Signal Processing 93 (2013) 2316–2327

Contents lists available at SciVerse ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Residual enhanced visual vector as a compact signature for mobile visual search

David Chen [a,*], Sam Tsai [a], Vijay Chandrasekhar [a], Gabriel Takacs [b], Ramakrishna Vedantham [b], Radek Grzeszczuk [b], Bernd Girod [a]

[a] Department of Electrical Engineering, Stanford University, CA 94305, USA
[b] Nokia Research Center, Palo Alto, CA 94304, USA

ARTICLE INFO

Article history:
Received 5 November 2011
Received in revised form
27 April 2012
Accepted 6 June 2012
Available online 16 June 2012

Keywords:
Mobile visual search
Compact signatures
Database compression

ABSTRACT

Many mobile visual search (MVS) systems transmit query data from a mobile device to a remote server and search a database hosted on the server. In this paper, we present a new architecture for searching a large database directly on a mobile device, which can provide numerous benefits for network-independent, low-latency, and privacy-protected image retrieval. A key challenge for on-device retrieval is storing a large database in the limited RAM of a mobile device. To address this challenge, we develop a new compact, discriminative image signature called the Residual Enhanced Visual Vector (REVV) that is optimized for sets of local features which are fast to extract on mobile devices. REVV outperforms existing compact database constructions in the MVS setting and attains similar retrieval accuracy in large-scale retrieval as a Vocabulary Tree that uses 25× more memory. We have utilized REVV to design and construct a mobile augmented reality system for accurate, large-scale landmark recognition. Fast on-device search with REVV enables our system to achieve latencies around 1 s per query regardless of external network conditions. The compactness of REVV allows it to also function well as a low-bitrate signature that can be transmitted to or from a remote server for an efficient expansion of the local database search when required.

1. Introduction

In the last few years, many mobile visual search (MVS) applications have been developed for automatic recognition of different objects of interest, such as outdoor landmarks [1–3], media covers [4,5], artwork [6], printed documents [7], and plants [8]. In each case, the user snaps a photo of the object with a mobile device to retrieve information about the object. Robust visual recognition is typically achieved using scale- and rotation-invariant local features like SIFT [9], SURF [10], GLOH [11], CHoG [12], and RIFF [13]. For large-scale visual search, there is also an equally important problem of indexing the billions of local features extracted from a database containing millions of images. Sivic and Zisserman developed the popular Bag-of-Features (BoF) framework [14]. Nister and Stewenius subsequently extended the BoF framework to generate much larger codebooks using tree-structured vector quantization [15]. Their Vocabulary Tree (VT) and subsequent variants [16–18] are widely used today.

Existing MVS applications typically transmit query data, either a JPEG-compressed image or compressed image features, from a mobile device to a remote server and then query a database hosted on the server. These applications employ the architecture depicted in Fig. 1(a). Central to the optimization of these applications is the development of a compact image signature, which

* Corresponding author. Tel.: +1 650 996 8352; fax: +1 650 724 3648.
E-mail address: dmchen@stanford.edu (D. Chen).

0165-1684/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.sigpro.2012.06.005

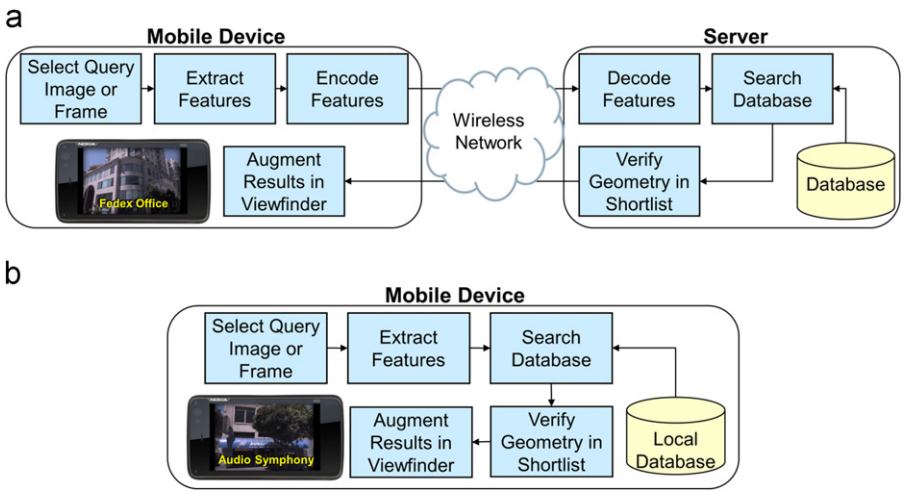

Fig. 1. (a) Features are extracted on the mobile device and transmitted to the server. Subsequently, the server compares the received features against the database (Architecture A). (b) All of the recognition is performed on the mobile device (Architecture B).

enables the query data to be transmitted faster over the network and more database images to be indexed in a limited amount of memory. Yeo et al. [19], Torralba et al. [20], and Weiss et al. [21] generated short hashes from the image or local features extracted from the image. Chandrasekhar et al. [22] applied transform coding on SIFT and SURF descriptors. Chen et al. [23] developed tree histogram coding to efficiently encode the BoF histogram for a large VT. Ji et al. [24] created low bitrate signatures with their multiple channel coding scheme, taking advantage of contextual cues like GPS, barcode, or RFID tags. Jegou et al. [25] utilized product quantizers to compactly represent SIFT descriptors. To create a custom descriptor which is inherently compressible, Chandrasekhar et al. [12] developed the CHoG descriptor. In a similar aim to create a compact descriptor, Calonder et al. [26] proposed BRIEF, a binary descriptor formed from simple intensity difference tests rather than histogram of gradients. The complementary problem of encoding the feature keypoint locations has been addressed by Tsai et al. [27] in their location histogram coding. Inspired by these recent developments, MPEG is currently pursuing the standardization of Compact Descriptors for Visual Search (CDVS) [28].

As mobile computing power and other hardware resources are improving rapidly, operations typically performed on a server can instead be performed directly on the mobile device. Therefore, the alternative architecture of on-device matching depicted in Fig. 1(b) is gaining greater interest now and enables several key advantages: (1) In regions with unreliable or no cellular service, a visual query can still be performed. (2) Traffic and computation can be reduced on a server that is handling many other queries. (3) Querying the locally stored database can be faster than querying a database on a remote server, because data transmission delays are avoided. (4) When no data is transmitted, the privacy of photos is protected.

Previous efforts to pursue effective on-device image matching include the landmark recognition system developed by Takacs et al. [1]. Their system partitions geolocation into loxels, uses the smartphone's GPS to selectively download features belonging to landmarks in nearby loxels, and quantizes the features using small k–d trees. Chen et al. [29] compress the inverted index of a VT and reduce the index's memory usage by 6×, allowing the index to fit in a smartphone's random access memory (RAM). Similarly, Schroth et al. [3] select quantizers and inverted files to transmit from a server to the mobile device based on vague prior locations like cell IDs. These previous on-device matching schemes all use tree-structured indexing schemes and suffer from the large memory usage of the database, which has severely constrained the number of images that can be indexed in the limited RAM of the mobile device.

In this paper, we develop a compact image signature called the Residual Enhanced Visual Vector (REVV) which further significantly reduces the database's memory usage. Like the Vector of Locally Aggregated Descriptors (VLAD) [30] and the Compressed Fisher Vector (CFV) [31], the REVV signature is formed from visual word residuals to build a compact representation. However, in contrast to VLAD and CFV, REVV is designed and optimized particularly for fast on-device image matching, using several hundred features like SURF and CHoG that can be computed in around 1 s on a mobile platform [4], and provides several key enhancements in residual aggregation, dimensionality reduction, and signature comparison. With these important new enhancements, REVV can attain similar retrieval accuracy as a VT with 1 million visual words, while using 25× less memory than the VT.

Our first contribution in this paper is a systematic optimization of the different processing stages for REVV: residual aggregation, power law, dimensionality reduction, and signed binarization. We demonstrate REVV's excellent recognition performance in large-scale retrieval for both SURF and CHoG features. Then, our second contribution is the design and construction of a new mobile augmented reality system for landmark recognition, which uses REVV to form a compact on-device database representation. A large database of landmark

images is queried entirely on the mobile device in around 1 s to produce nearly instantaneous augmentations in the viewfinder. This low latency can be achieved regardless of external network conditions. Finally, our third contribution is the use of REVV as a very discriminative low-bitrate image signature. This role for REVV is important for the scenario where a local on-device query is expanded onto a remote server. Due to REVV's compactness, the same REVV signature can be efficiently transmitted over a wireless network to the server.

The rest of the paper is organized as follows. In Section 2, the design of REVV is presented. Section 3 then describes the application of REVV for on-device database search, where large-scale retrieval results, memory usage comparisons, and an actual on-device implementation are reported. Section 4 discusses the application of REVV for low-bitrate visual queries in the MPEG CDVS framework, where additional large-scale performance evaluations are conducted.

2. Design of the residual enhanced visual vector

Fig. 2(a) shows the pipeline for generating a query REVV signature and then comparing it against pre-computed database REVV signatures for large-scale image retrieval. Like the VT, REVV can accurately and quickly generate a ranked list for a large database. In contrast to the VT, REVV achieves a very compact database representation and so allows many more images to be indexed in the database in the same amount of limited RAM. Now, we will examine the detailed operations of each processing block of the REVV pipeline in the following sections.

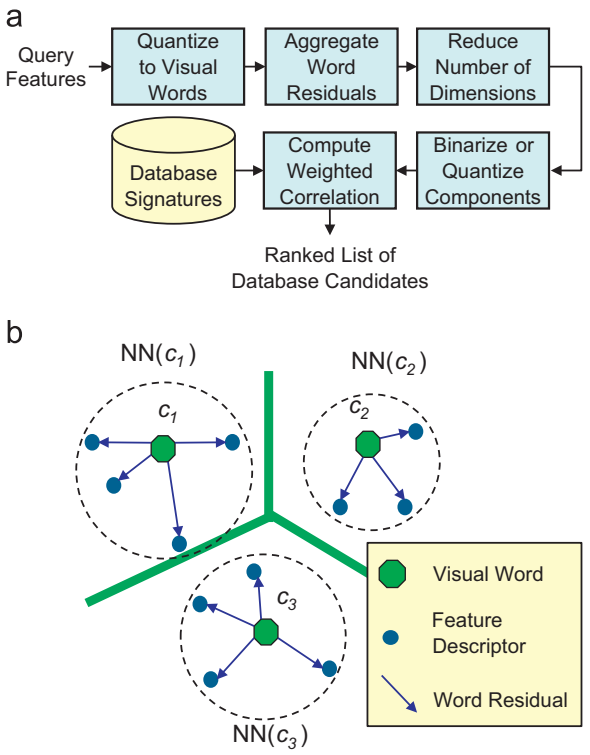

Fig. 2. (a) Pipeline for generating a query REVV signature and comparing against database REVV signatures. (b) Toy example of three visual words, feature descriptors classified to the nearest visual words, and resulting visual word residuals.

2.1. Image-level receiver operating characteristic

Many algorithmic choices and parameters must be optimized in the blocks shown in Fig. 2(a) to achieve the best performance for REVV. In computer vision [12,32], pattern recognition [33], and signal detection [34], the receiver operating characteristic (ROC) is commonly used to evaluate the performance of a system or algorithm. In this section, we employ an image-level ROC, which is a plot of the true positive rate versus the false positive rate for image matching, to accomplish the systematic optimization of REVV. Later in Sections 3 and 4, we will also validate the design choices generated by ROC-based optimization on large-scale retrieval tests.

In the ROC comparisons, for training, 16,000 matching and 16,000 non-matching image pairs are taken from the Oxford Buildings Data Set [35], which contains images of buildings around the Oxford campus, and the University of Kentucky Benchmark Data Set [36], which contains images of indoor objects like books, CDs, paintings, plates, and many other household objects. For testing, 8000 matching and 8000 non-matching image pairs are taken from the Zurich Buildings Data Set [37], which contains images of buildings around Zurich, and the Stanford Media Cover Data Set [38], which contains images of books, CDs, and DVDs. A mix of similar outdoor landmarks and indoor objects are contained in both training and testing data sets to ensure similar feature statistics of the two data sets for effective learning. Our work focuses on low-latency MVS, so we extract around 600 SURF features or CHoG features per image, which takes about 1 s on a mobile device [4].

2.2. Aggregation methods

Aggregation is the first step in forming a REVV signature. Let $\mathbf{c}_1,\ldots,\mathbf{c}_k$ be a set of d-dimensional visual words. As illustrated in the toy example of Fig. 2(b), after each descriptor in an image is quantized to the nearest visual word, a set of word residual (WR) vectors will surround each visual word. For a given image, let $V(\mathbf{c}_i)=\{\mathbf{v}_{i,1},\mathbf{v}_{i,2},\ldots,\mathbf{v}_{i,N_i}\}$ represent the set of N_i residual vectors around the ith visual word $\mathbf{c}_i$, where $\mathbf{v}_{i,j}\in\Re^d$. To aggregate the residuals, several different approaches are possible:

- Sum aggregation: This is the approach used by VLAD [30]. Here, the aggregated residual for the ith visual word is $\mathbf{S}_i=\sum_{j=1}^{N_i}\mathbf{v}_{i,j}$.
- Mean aggregation: We normalize the sum of residual vectors by the cardinality of $V(\mathbf{c}_i)$, so the aggregated residual becomes $\mathbf{S}_i=1/N_i\cdot\sum_{j=1}^{N_i}\mathbf{v}_{i,j}$.
- Median aggregation: This is similar to mean aggregation, except we find the median along each dimension, i.e., $\mathbf{S}_i(n)=\text{median}\{\mathbf{v}_{i,j}(n):j=1,\ldots,N_i\}$, $n=1,\ldots,d$.

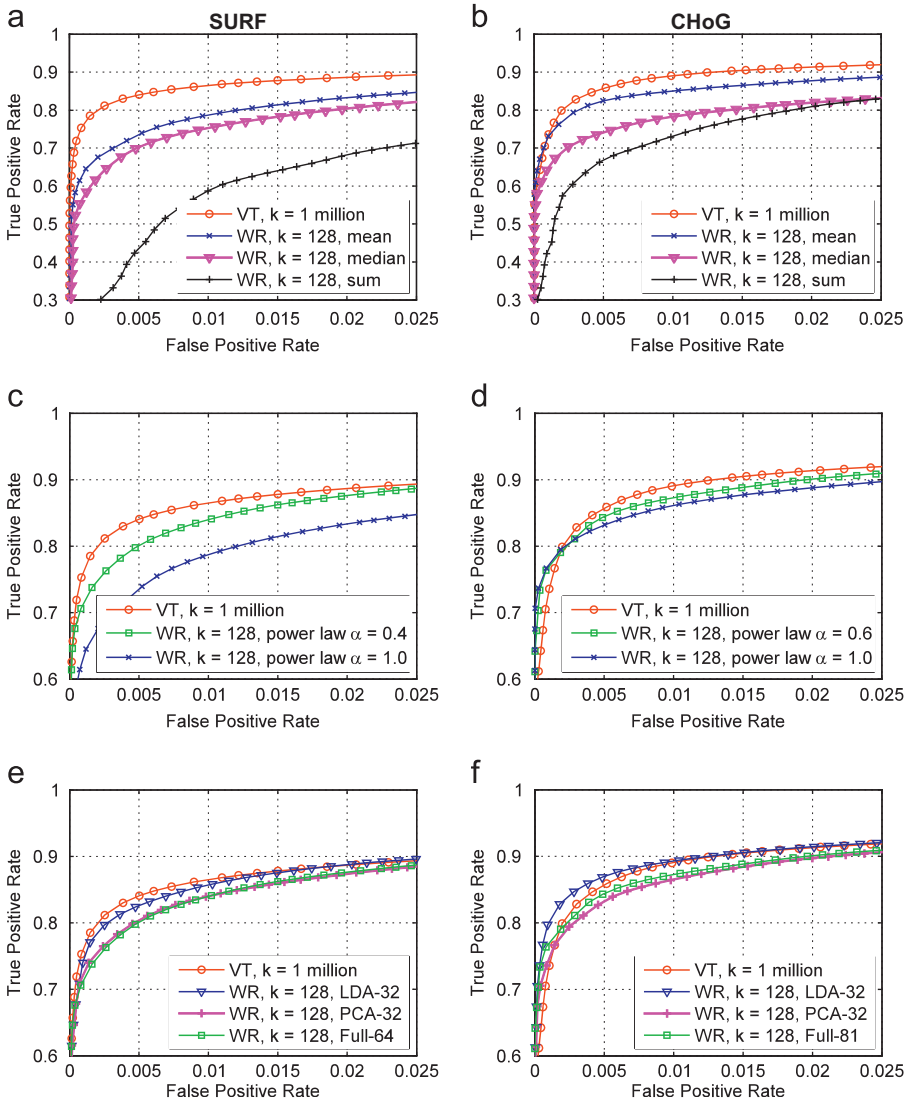

Fig. 3. VT stands for Vocabulary Tree and WR stands for a Word Residual method. (a–b) ROC performance for different aggregation methods. (c–d) ROC performance with and without applying a power law. (e–f) ROC performance with different dimensionality reduction techniques.

Next, let $\mathbf{S}=[\mathbf{S}_1\mathbf{S}_2\ldots\mathbf{S}_k]\in\Re^{kd}$ be the concatenation of the aggregated word residuals at all the visual words. A normalized image signature $\overline{\mathbf{S}}$ is formed as $\overline{\mathbf{S}}=\mathbf{S}/||\mathbf{S}||_2$. To compare two normalized image signatures $\overline{\mathbf{S}}_q$ and $\overline{\mathbf{S}}_d$, we can compute $\|\overline{\mathbf{S}}_q-\overline{\mathbf{S}}_d\|_2$, or equivalently $\langle\overline{\mathbf{S}}_q,\overline{\mathbf{S}}_d\rangle$.

The ROCs of the three aggregation methods are shown in Fig. 3(a–b) for $k=128$ visual words. Also plotted for comparison is the ROC for a VT with $k=1$ million words. We use settings that allow the VT to attain excellent retrieval performance: greedy search with 10 best paths [16], soft binning with three nearest centroids [17], and TF-IDF weighting. We measure the ROCs separately for a 64-dimensional SURF descriptor and an 81-dimensional CHoG descriptor.

From the comparison in Fig. 3(a–b), we observe that the sum-aggregated WR, which is a version of VLAD without compression as reported in [30], has a performance gap from the VT in this current setting which is favorable to low-latency MVS. The mean-aggregated WR, which requires just one additional division per visual word compared to the sum-aggregated WR, performs substantially better. Furthermore, the mean-aggregated WR also outperforms the median-aggregated WR, which is more expensive to compute. We have performed an analysis based on hypothesis testing explaining why mean aggregation outperforms sum aggregation.[1]

2.3. Power law

Applying a power law can reduce the influence of peaky components which are difficult to match [31]. The unnormalized residual vector becomes $\mathbf{S}=[\mathbf{S}_1^{\alpha}\ \mathbf{S}_2^{\alpha}\ \cdots\ \mathbf{S}_k^{\alpha}]$,

[1] Analysis: ⟨http://tinyurl.com/3llop3u⟩.

where $\mathbf{S}_i^{\alpha} = [\mathbf{S}_i(1)^{\alpha}\mathbf{S}_i(2)^{\alpha}\cdots\mathbf{S}_i(d)^{\alpha}]$ for $i = 1,2,\ldots,k$ and $\alpha \in [0,1]$. The normalized vector is defined identically as before. Experimentally, we found the optimal exponent value is $\alpha = 0.4$ and $\alpha = 0.6$ for SURF and CHoG, respectively. Fig. 3(c–d) shows the positive improvement in the ROC when we apply a power law. Since SURF omits the max-clipping functionality used by SIFT, some of the values in the SURF descriptor are relatively peaky, so applying a power law is important and a smaller value of α is employed. In contrast, CHoG descriptors are probability mass functions (PMFs) with soft binning applied and adequate amounts of bias added, so extremely peaky bin counts are rare [12] and a relatively large value of α is required.

2.4. Dimensionality reduction

Since the memory usage of the database is directly proportional to the residual vector's dimensionality, we want to reduce the dimensionality as much as possible without adversely impacting retrieval accuracy. VLAD [30] and CFV [31] both use principal component analysis (PCA) for dimensionality reduction. In our approach, we incorporate the knowledge about matching and non-matching image pairs via linear discriminant analysis (LDA). For each visual word, we solve the following problem:

$S_j = \text{word residual for image } j$

$J_{\mathrm{m}} = \{(j_1,j_2) : \text{ images } j_1 \text{ and } j_2 \text{ are matching}\}$

$J_{\mathrm{nm}} = \{(j_1,j_2) : \text{ images } j_1 \text{ and } j_2 \text{ are non-matching}\}$

$$\underset{\mathbf{w}}{\text{maximize}} \frac{\sum_{(j_1,j_2)\in J_{\mathrm{nm}}} \langle \mathbf{w}, \mathbf{S}_{j_1} - \mathbf{S}_{j_2} \rangle^2}{\sum_{(j_1,j_2)\in J_{\mathrm{m}}} \langle \mathbf{w}, \mathbf{S}_{j_1} - \mathbf{S}_{j_2} \rangle^2} \tag{1}$$

The objective in Eq. (1) is to maximize the ratio of inter-class variance to intra-class variance over the projection direction $\mathbf{w}$. LDA has also been used previously in [32] to reduce the dimensionality of custom feature descriptors. Eq. (1) has the following solution:

$$\mathbf{R}_{\mathrm{nm}}\mathbf{w}_i = \lambda_i \mathbf{R}_{\mathrm{m}}\mathbf{w}_i, \quad i = 1,2,\ldots,d_{\mathrm{lda}}$$

$$\mathbf{R}_{\theta} = \sum_{(j_1,j_2)\in J_{\theta}} (\mathbf{S}_{j_1} - \mathbf{S}_{j_2})(\mathbf{S}_{j_1} - \mathbf{S}_{j_2})^T, \quad \theta \in \{\mathrm{m},\mathrm{nm}\} \tag{2}$$

For each visual word, we retain the d_{lda} most energetic components after projection. In Fig. 3(e–f), we plot the ROC for (1) the original WR with 64 or 81 dimensions/word for SURF and CHoG, respectively; (2) WR reduced by PCA to 32 dimensions/word; and (3) WR reduced by LDA to 32 dimensions/word. The PCA-reduced WR performs similarly as the original WR, while the LDA-reduced WR outperforms the two other WR schemes. With LDA, we can reduce the image signature's dimensionality by 2× and 2.5× for SURF and CHoG, respectively, while boosting the retrieval performance.

Our LDA objective function in Eq. (1) is the ratio trace. Other authors have achieved improved performance by optimizing instead the trace ratio [39,40], a problem for which there is no closed-form solution but there does exist an iterative procedure for finding the solution. We have also computed the LDA projection matrix by optimizing the trace ratio, and we found the trace ratio optimization performs 0.1–0.4% better in true positive rate at very low false positive rates compared to the ratio trace optimization.

2.5. Signed binarization and correlation weights

Following dimensionality reduction by LDA, each component of the transformed residual is binarized to $+1$ or -1 depending on the sign. As in [31], this signed binarization creates a compact signature that just requires at most $k \cdot d_{\mathrm{lda}}$ bits to represent the residuals and k bits to indicate which visual words have been visited. For $d_{\mathrm{lda}} \le 32$, the d_{lda} binarized residual components at a visual word can be conveniently packed into a 32-bit unsigned integer.

A second benefit of signed binarization is fast score computation. The correlation between binary vectors $\mathbf{S}_q^{\mathrm{bin}}$ and $\mathbf{S}_d^{\mathrm{bin}}$ is given by

$$\frac{1}{N_q^{\mathrm{bin}} N_d^{\mathrm{bin}}} \sum_{i\in I_q \cap I_d} C(\mathbf{S}_{q,i}^{\mathrm{bin}}, \mathbf{S}_{d,i}^{\mathrm{bin}}) \tag{3}$$

where the quantities involved have the following meaning:

- $\mathbf{S}_{q,i}^{\mathrm{bin}}$ and $\mathbf{S}_{d,i}^{\mathrm{bin}}$ are the binarized residuals at the ith visual word
- $C(\mathbf{S}_{q,i}^{\mathrm{bin}}, \mathbf{S}_{d,i}^{\mathrm{bin}}) = d_{\mathrm{lda}} - 2H(\mathbf{S}_{q,i}^{\mathrm{bin}}, \mathbf{S}_{d,i}^{\mathrm{bin}})$ and H is Hamming distance
- I_q and I_d are the sets of visual words visited by the query and database images, respectively
- $N_q^{\mathrm{bin}} = \sqrt{d_{\mathrm{lda}}|I_q|}$ and $N_d^{\mathrm{bin}} = \sqrt{d_{\mathrm{lda}}|I_d|}$ are normalization factors

Importantly, the Hamming distance can be computed very quickly using bitwise XOR and POPCNT (population count) instructions.

When comparing two binarized residuals, a discriminative weighting can be applied to further improve performance. Fig. 4(a–b) plots two distributions: (1) the distribution $\Pr\{C|\text{match}\}$ of binary correlation per visual word for matching images pairs, and (2) the analogous distribution $\Pr\{C|\text{non-match}\}$ for non-matching image pairs. Large correlation values are more likely to come from matching pairs than from non-matching pairs. This property is captured nicely in the probability $\Pr\{\text{match}|C\}$, which increases as C increases. Thus, we design a weighting function $w(C) = \Pr\{\text{match}|C\}$. Assuming $\Pr\{\text{match}\} = \Pr\{\text{non-match}\}$, then by Bayes' Rule

$$w(C) = \frac{\Pr\{C|\text{match}\}}{\Pr\{C|\text{match}\} + \Pr\{C|\text{non-match}\}} \tag{4}$$

This weighting function is plotted in Fig. 4(c–d). Using this weighting function, the score changes from Eq. (3) to

$$\frac{1}{N_q^{\mathrm{bin}} N_d^{\mathrm{bin}}} \sum_{i\in I_q \cap I_d} w(C(\mathbf{S}_{q,i}^{\mathrm{bin}}, \mathbf{S}_{d,i}^{\mathrm{bin}})) \cdot C(\mathbf{S}_{q,i}^{\mathrm{bin}}, \mathbf{S}_{d,i}^{\mathrm{bin}}) \tag{5}$$

After binarization and weighting, we obtain the ROC plotted in Fig. 5(a–b), where it can be seen that the

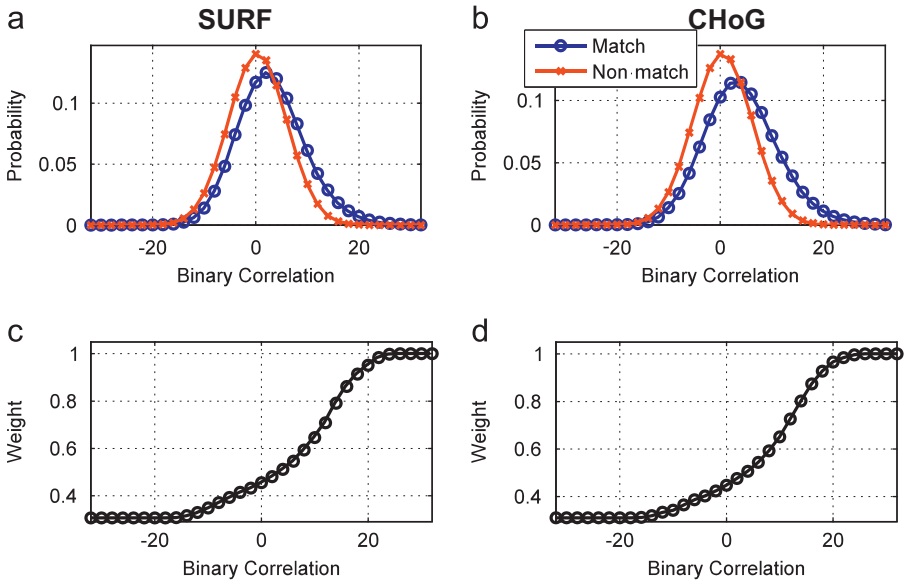

Fig. 4. (a–b) Distribution of binary correlations for matching and non-matching image pairs. (c–d) Weights for different binary correlation values.

binarized WR with $k=128$ visual words performs comparably well as the VT with $k=1$ million words.

2.6. Asymmetric signatures

REVV can be made even more discriminative if we create an asymmetric signature [25,30]. For on-device image matching, it makes sense to binarize the database signatures so that they can fit compactly in the limited RAM of the mobile device, whereas the query signature should not be compressed to avoid information loss and additional computation. For low-bitrate visual queries sent to a remote server, the opposite is true: the query signature should be binarized for efficient transmission over a wireless link, while the database signatures can be more finely quantized because they reside on a server with ample RAM.

Suppose we have an uncompressed query signature $\mathbf{S}_q$ and a binarized database signature $\mathbf{S}_d^{\text{bin}}$. We change the correlation score from Eq. (5) to the following:

$$\frac{1}{\|\mathbf{S}_q\|_2 \cdot N_d^{\text{bin}}} \sum_{i \in I_q \cap I_d} w(C(\mathbf{S}_{q,i}^{\text{bin}}, \mathbf{S}_{d,i}^{\text{bin}})) \cdot C(\mathbf{S}_{q,i}, \mathbf{S}_{d,i}^{\text{bin}}) \tag{6}$$

where the second correlation is computed by

$$C(\mathbf{S}_{q,i}, \mathbf{S}_{d,i}^{\text{bin}}) = \sum_{j=1}^{d_{\text{lda}}} \mathbf{S}_{q,i}(j) \cdot \mathbf{S}_{d,i}^{\text{bin}}(j), \quad i \in I_q \cap I_d \tag{7}$$

Since $\mathbf{S}_{d,i}^{\text{bin}}(j) \in \{-1,+1\}$, Eq. (7) can be computed using solely additions and subtractions. In (Appendix A), we further develop an efficient tree-based algorithm that speeds up these distance computations for large-scale retrieval. Fig. 5(c–d) plots the ROCs of REVV with asymmetric and binarized signatures, where it is evident that the asymmetric signatures provide a boost in matching performance for both SURF and CHoG features.

2.7. Fusion of multiple features

Fusion of information from multiple features has been discovered to improve performance in many recognition and retrieval tasks [41,42]. Assuming that both SURF and CHoG features are extracted on the mobile device, we can generate two different REVV signatures $\mathbf{S}_q^{\text{SURF}}$ and $\mathbf{S}_q^{\text{CHoG}}$ for the query image and then compute the correlations $C^{\text{SURF}} = \langle \mathbf{S}_q^{\text{SURF}}, \mathbf{S}_d^{\text{SURF}} \rangle$ and $C^{\text{CHoG}} = \langle \mathbf{S}_q^{\text{CHoG}}, \mathbf{S}_d^{\text{CHoG}} \rangle$ with respect to a database image. We tested several fusion methods (min, max, weighted sum) and found that weighted sum fusion [43,44] generated the best result, where a joint score is formed as $C^{\text{Joint}} = \beta C^{\text{SURF}} + (1-\beta) C^{\text{Joint}}$ and $\beta \approx 0.4$. Fig. 5(e) shows that the joint scheme can provide several percent gain in matching accuracy compared to using CHoG or SURF alone. The cost for utilizing both SURF and CHoG features to improve matching accuracy is that the query latency and the database's memory usage both roughly double. Other local features like SIFT and RIFF have also been tested with REVV [45,46] and could be used to further improve matching performance, but would likewise incur higher costs in computation and memory.

A combination of global and local features has also been used in near-duplicate video retrieval for improved recognition accuracy [42,41]. Unlike local features, global features are generally not designed to be invariant against the types of severe geometric distortions typically encountered in MVS applications, including arbitrary rotations of the camera and large viewpoint changes. Extensive evaluations on a variety of data sets have shown that global features like GIST [47] or color histograms are significantly outperformed by local feature representations when strong geometric distortions are present [48]. By design, REVV aggregates local features and so inherits the geometric and photometric invariances that are carefully built into the local features. Since we are pursuing low-latency MVS with a memory-efficient database, for the remainder of the paper, we focus on experiments

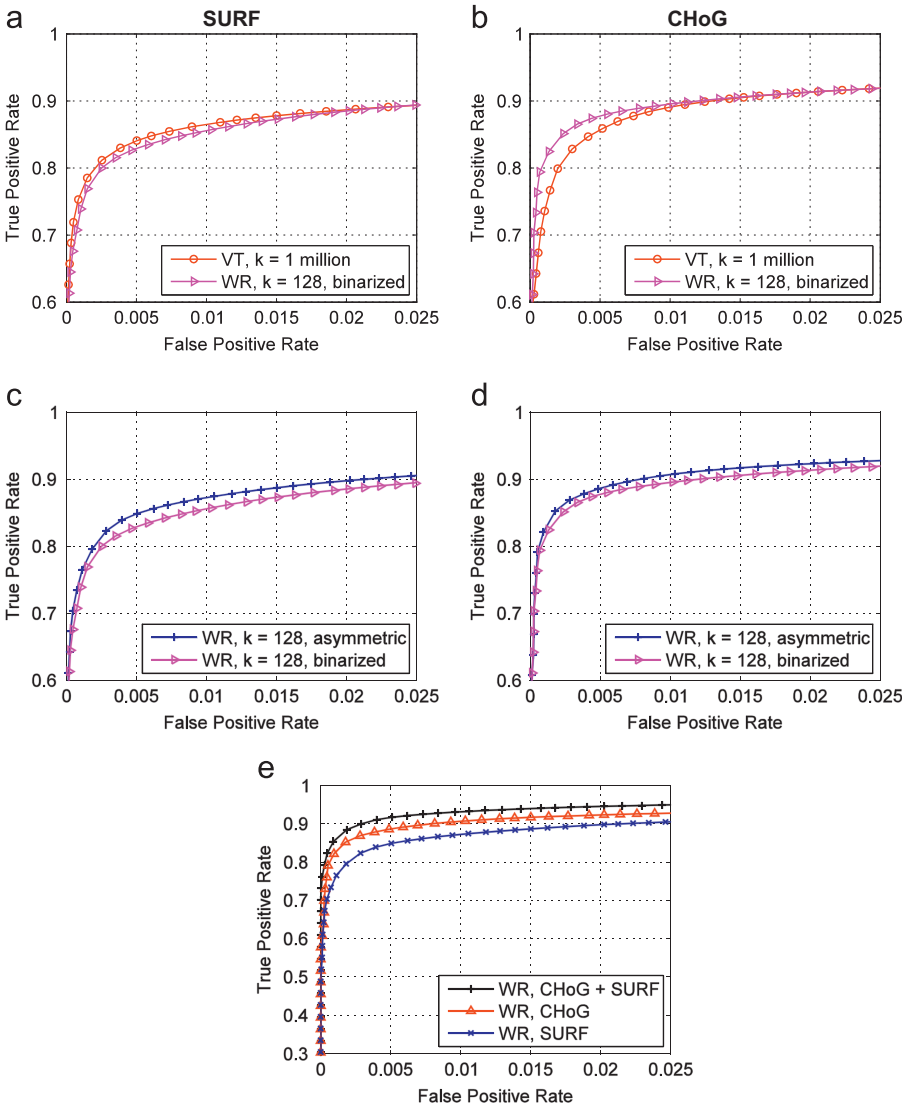

Fig. 5. (a–b) ROC performance of binarized REVV versus VT. (c–d) ROC performance of binarized and asymmetric REVV. (e) ROC performance after fusion of information from multiple features.

where just a single local feature type is used to generate REVV for on-device MVS applications.

3. Compact signatures for on-device database search

In the previous section, we designed REVV as a discriminative and compact image signature for MVS. Now, for REVV to be useful for on-device image matching, we first verify that it can attain excellent performance in large-scale retrieval while using a very small amount of memory. Subsequently, we incorporate the REVV signature into an on-device landmark recognition system and demonstrate an actual implementation on a smartphone.

3.1. Large-scale retrieval

We test the retrieval performance of REVV versus the VT on two large data sets. First, we test on the Stanford YouTube Data Set [49], where the database is 1 million keyframes taken from over 2000 YouTube video clips, and the query set contains 1224 viewfinder frames captured by a camera phone. Unlike many video retrieval systems which match a group of keyframes in one video to a group of keyframes in another video, the video retrieval system in [49] matches a single query keyframe by itself against a database of keyframes to achieve a very fast response for interactive mobile applications. For our current experiments on the Stanford YouTube Data Set, we also match each query keyframe by itself against the database. Second, we test on the Stanford MVS Data Set [50], where the database consists of 1200 labeled "clean" images of objects in eight different categories and 1 million distractor images, and the query set contains 3300 images of the same objects taken with different camera phones. Fig. 6(a–b) plots the recall versus database size for both data sets. SURF features are used, the VT or REVV is used to generate an initial ranked list, and database images in a

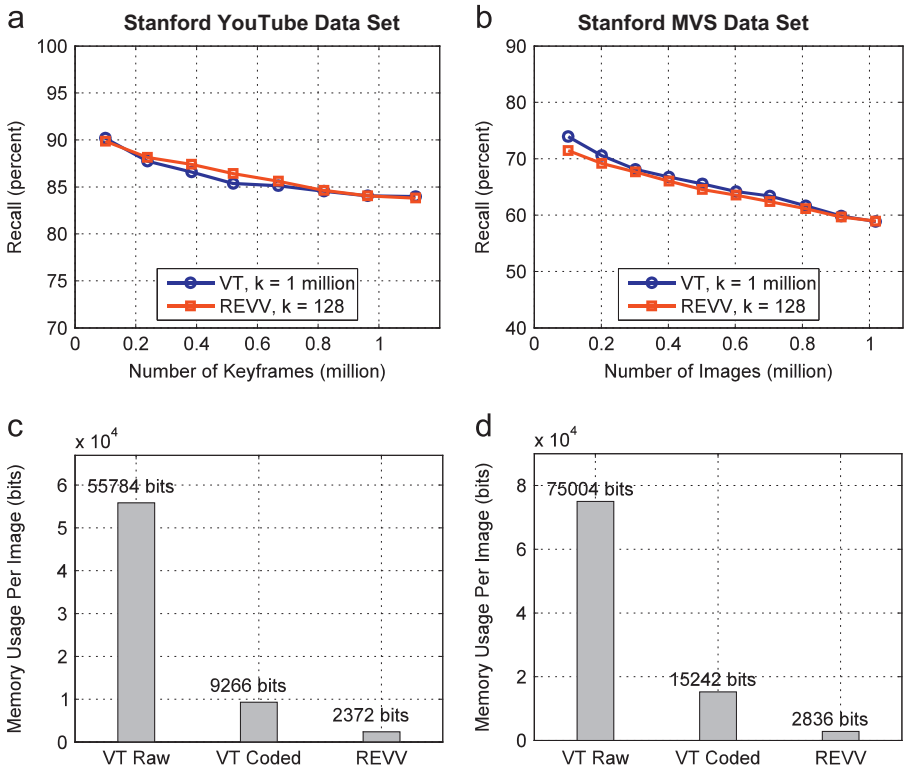

Fig. 6. (a–b) Recall for image retrieval from large databases. (c–d) Memory usage per image.

shortlist of the top 50 candidates are then geometrically verified with a distance ratio test [9] and RANSAC. For both data sets, REVV with $k=128$ words achieves similar recall as the VT with $k=1$ million words. Importantly, the recall rates for REVV and VT closely track one another closely as database size increases.

3.2. Memory usage

We compare the size of the database for three schemes: (1) VT with an uncompressed inverted index, (2) VT with a compressed inverted index [29], and (3) our newly designed REVV. Fig. 6(c–d) plots the memory usage per image in the database for the Stanford YouTube Data Set and the Stanford MVS Data Set, where again we used SURF features. Memory usage is generally lower for the YouTube Data Set versus the MVS Data Set because fewer features are extracted per image. Index compression for VT yields 5–6 $\times$ memory savings relative to the VT with uncompressed index. In contrast, REVV provides far greater savings: memory usage is shrunk 24–26 $\times$ from that of the VT with uncompressed index.

3.3. Multi-round database scoring

The comparisons between the query REVV signature and all database REVV signatures can be made significantly faster, with essentially no effect on retrieval accuracy, if we utilize a multi-round scoring algorithm similar to partial index traversal [51,52]. We score the database signatures in three steps: (1) Calculate partial correlations between the query signature and all database signatures using just k_{part} out of the k visual words overall. Typically, we use $k_{\text{part}} \approx 0.3k$ and choose the k_{part} visual words randomly. (2) Sort the database images by the computed partial correlations. (3) Out of the N_{total} database images in total, finish computing the full correlations for the N_{part} images attaining the highest partial correlations from Step 1. Typically, we set $N_{\text{part}} \approx 0.2N_{\text{total}}$.

The retrieval results for REVV in Fig. (a–b) were generated with multi-round scoring, where the database search consumed about 0.9 s per query on a 2.3 GHz Intel Xeon processor. A slower exhaustive comparison method achieves the same accuracy but consumes 2.7 s per query. We apply multi-round scoring in our mobile landmark recognition system to achieve very low query latencies, as we discuss in the subsequent section.

3.4. On-device landmark recognition system

Existing mobile landmark recognition systems [1–3] frequently send data between the mobile device and a remote server. Using REVV, we have developed and implemented an augmented reality (AR) application for landmark recognition that works entirely on a Nokia N900 smartphone, without needing assistance from an external server. The AR application has two modes. The default camera mode (left side of Fig. 7(a)) shows a live viewfinder stream of the physical world. During periods of low motion, new queries are initiated [53]. Almost instantly thereafter, relevant information of any recognized landmark – including the name, address, and phone

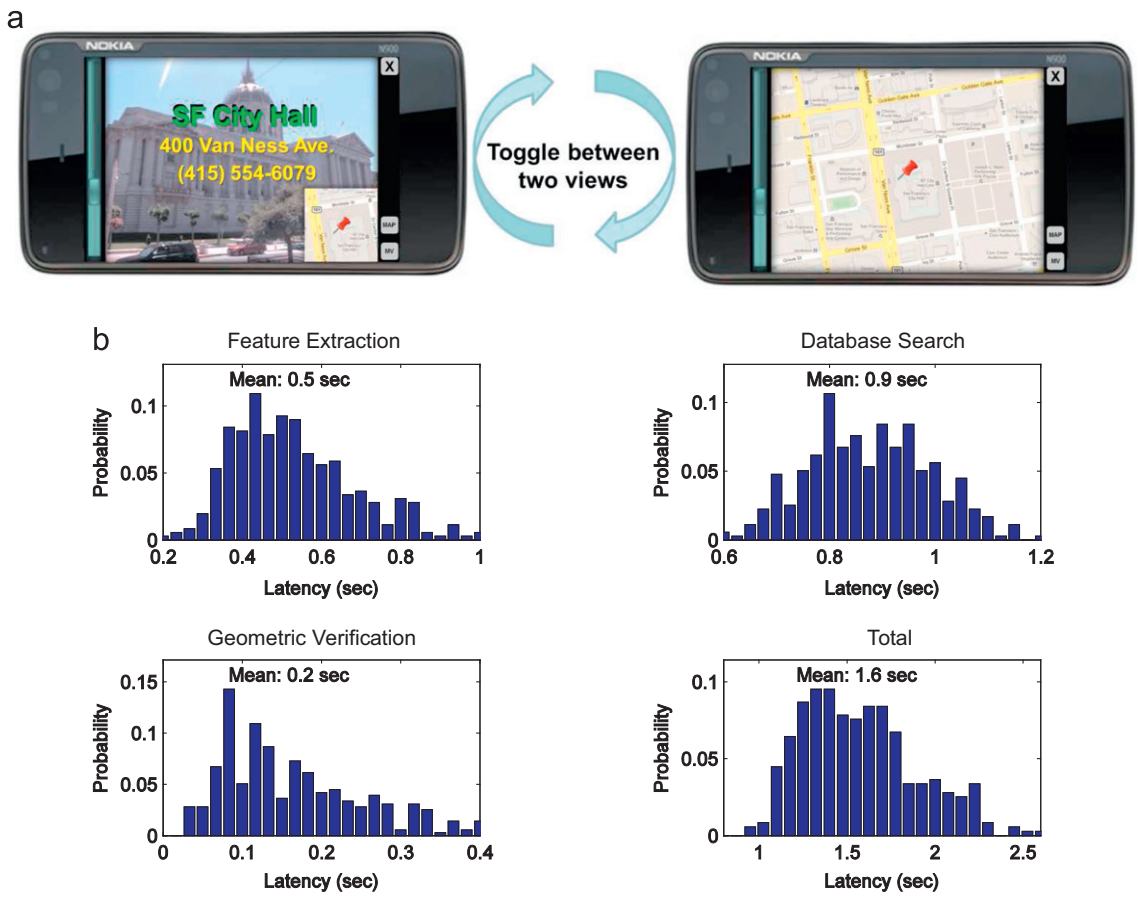

Fig. 7. (a) Screenshot of the two modes – camera mode on the left and map mode on the right – in our mobile landmark recognition application. (b) Distribution of latencies in our mobile landmark recognition system, measured over 350 actual queries.

number – is augmented in the viewfinder. The user can also toggle to a larger map (right side of Fig. 7(a)) of the local neighborhood. A demo video of the AR system is available online.[2]

During each query, the image-based recognition is performed entirely on the mobile device in the following steps: (1) SURF features are extracted from a viewfinder frame in a low-motion interval. (2) A query REVV signature is generated and compared against database REVV signatures to generate a ranked list. (3) A shortlist of top database candidates is geometrically verified.

At any given time, our system stores REVV signatures for 10,000 database images in RAM, where the images are taken from the San Francisco Landmark Data Set [44]. This quantity of images could represent (1) the most famous landmarks in the city or (2) landmarks in the local vicinity as estimated by GPS or cell ID. The N900 smartphone has a 600 MHz ARM processor and 256 MB of RAM, but much of this RAM is occupied by the operating system and other core applications, leaving just 50–60 MB of RAM for our AR application. Due to REVV's compactness, the 10,000 REVV database signatures and REVV's data structures only consume about 3.5 MB in total. In contrast, a VT with 1 million visual words requires about 70 MB to store a large tree [23] and an additional 18 MB to store a compressed inverted index [29].

Fig. 7(b) plots the distribution of latencies for 350 actual queries. The vast majority of queries take between 1 and 2 s, with a mean total latency of 1.6 s. Since we perform the recognition entirely on the device, this low latency can be achieved regardless of network conditions.

4. Compact signatures for low-bitrate visual queries

In the previous section, we used REVV as a memory-efficient signature for on-device image matching. If the best matching images are currently not located on the mobile device but instead reside on a remote server, we can expand the search by querying a remote server. Having already generated a REVV signature for the on-device search, we can reuse the same compact signature to perform a low-bitrate visual query. In this section, we first briefly describe the MPEG CDVS framework for evaluating low-bitrate image signatures and then report competitive results for the lowest target bitrate specified in the framework.

[2] Demo video: 〈http://www.youtube.com/watch?v=CwqtZ9364rE〉.

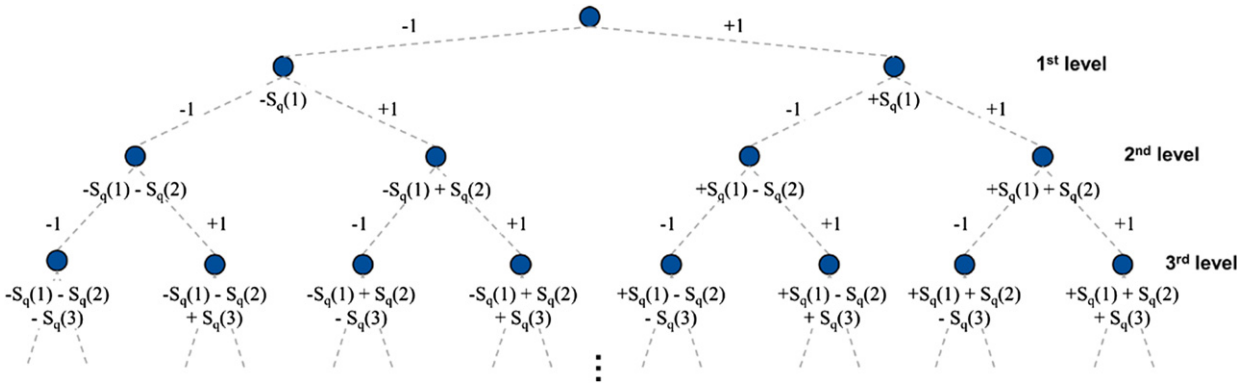

Fig. 8. Tree for progressively computing correlation values between an uncompressed residual vector and all possible binarized residual vectors.

4.1. Overview of MPEG CDVS framework

MPEG has designed a rigorous evaluation consisting of pairwise image matching and large-scale retrieval experiments on five categories of images: (1) text and graphics, (2) museum paintings, (3) video frames, (4) landmarks, and (5) common objects. The pairwise image matching experiments focus on the true positive rate (TPR) for each category, subject to the constraint that the overall false positive rate (FPR) is below 1%. The retrieval experiments also include 1 million distractor images collected from Flickr and focus on the mean average precision (MAP) and the precision at rank 1 (P@1). MPEG CDVS is concerned about how TPR, MAP, and P@1 vary as the query bitrate changes from 512 bytes to 16,384 bytes per image.

4.2. Image matching results in MPEG CDVS framework

We aim for the lowest target bitrate of 512 bytes in the MPEG CDVS framework. At such a low bitrate, very few features can be transmitted even if the features are compressed. On the query side, we generate a binarized REVV residual using CHoG descriptors, $k=192$ visual words, and $d_{\text{lda}}=28$ dimensions after LDA. On the server side, since the MPEG CDVS framework allows the database to use up to 16 GB of RAM, we quantize the database residuals to 8 bits per dimension to get an additional boost in retrieval accuracy from asymmetric signatures.

We compare REVV against another low-bitrate image signature method, Tree Histogram Coding [23], which losslessly compresses a VT histogram by run-length encoding. For THC, we use the same CHoG descriptors that are used for REVV, a tree with $k=1$ million leaf nodes, and hard binning. Soft binning amongst the m nearest nodes for each feature would approximately increase the bitrate for THC by a factor m, so we do not apply soft binning. Tables 1–3 record the pairwise matching and retrieval results for REVV and THC. For nearly every experiment, REVV attains better performance than THC at about half the bitrate. Another advantage of REVV is that the binarized residuals can be compared directly in the compressed domain, whereas THC requires the tree histogram to be fully decoded before a comparison can be made.

Table 1
True Positive Rate: MP and NMP refer to number of matching and non-matching image pairs, respectively. REVV uses 533 bytes and THC uses 1029 bytes.

Image Categories	MP	NMP	REVV	THC [23]
Text and graphics	3000	30,000	0.55	0.39
Museum paintings	364	3640	0.37	0.27
Video frames	400	4000	0.71	0.75
Landmarks	4005	48,675	0.62	0.30
Common objects	2550	25,500	0.58	0.47

Table 2
Mean Average Precision: REVV uses 533 bytes and THC uses 1029 bytes.

Image Categories	No. queries	REVV	THC [23]
Text and graphics	1500	0.55	0.43
Museum paintings	364	0.45	0.32
Video frames	400	0.87	0.79
Landmarks	3499	0.55	0.48
Common objects	2550	0.66	0.46

Table 3
Precision at Rank 1: REVV uses 533 bytes and THC uses 1029 bytes.

Image Categories	No. queries	REVV	THC [23]
Text and graphics	1500	0.61	0.53
Museum paintings	364	0.43	0.30
Video frames	400	0.86	0.76
Landmarks	3499	0.67	0.60
Common objects	2550	0.76	0.61

5. Conclusion

In this paper, we have demonstrated two important uses of a compact image signature that is designed for large-scale mobile visual search (MVS): (1) storing a large database on a memory-limited mobile device to achieve fast local queries, and (2) transmitting a low-bitrate signature over a wireless network to query a remote server when a local query needs to be expanded. Our

new signature, the Residual Enhanced Visual Vector (REVV), can be used effectively for both purposes. REVV has been optimized to work well with features like SURF, CHoG, or RIFF which can be computed efficiently on mobile devices for low-latency MVS. The general design of REVV also enables easy integration with other features like SIFT. While attaining the same retrieval accuracy as the VT, REVV uses 25× less memory than a VT. For on-device database search, we have developed a new mobile augmented reality system for landmark recognition which uses REVV to search a large database stored in the phone's RAM. For low-bitrate visual queries, we have shown that REVV can serve a compact signature which is efficient to transmit over a wireless network and that REVV achieves excellent results in the MPEG CDVS framework. In future work, the benefits of using REVV for efficiently updating a database over time will be studied.

Acknowledgment

We thank Dr. Hervé Jégou for very helpful suggestions on how to use his group's image retrieval software and data sets. We also thank the reviewers for their insightful comments and suggestions, which greatly improved the quality of this paper.

Appendix A. Fast computation of asymmetric correlations

Given an uncompressed residual $\mathbf{S}_{q,i}$ at the ith visual word of dimensionality d_{lda}, our goal is to efficiently evaluate Eq. (7) for every possible binary residual $\mathbf{S}_{d,i}^{\mathrm{bin}}$. For simplicity, we drop the dependence on i for the remainder of this section. We exploit the special structure amongst the binarized residual patterns and develop an efficient tree-based algorithm, which is depicted in Fig. 8. Here are the steps of the algorithm:

1. In the first level of the tree, place the two possible values of $\mathbf{S}_q(1)\cdot\mathbf{S}_d^{\mathrm{bin}}(1)$ at the two nodes.
2. In the second level of the tree, compute the four possible values of the sum $\sum_{j=1}^{2}\mathbf{S}_q(j)\cdot\mathbf{S}_d^{\mathrm{bin}}(j)$.
3. In the third level of the tree, compute the eight possible values of the sum $\sum_{j=1}^{3}\mathbf{S}_q(j)\cdot\mathbf{S}_d^{\mathrm{bin}}(j)$ by carefully reusing the partial sums already computed in the second level.
4. Repeat this down going down the tree, and at the leaf nodes, all possible values of $\sum_{j=1}^{d_{\mathrm{lda}}}\mathbf{S}_q(j)\cdot\mathbf{S}_d^{\mathrm{bin}}(j)$ will be available.

At the ith level, 2^i additions or subtractions are required, so this method requires $O(2^{d_{\mathrm{lda}}+1})$ operations, which is significantly less than the $O(d_{\mathrm{lda}}\cdot 2^{d_{\mathrm{lda}}})$ operations required for a naive approach for typical values of d_{lda} like $d_{\mathrm{lda}}=32$.

References

[1] G. Takacs, V. Chandrasekhar, N. Gelfand, Y. Xiong, W.-C. Chen, T. Bismpigiannis, R. Grzeszczuk, K. Pulli, B. Girod, Outdoors augmented reality on mobile phone using loxel-based visual feature organization, in: ACM Multimedia Information Retrieval, 2008.
[2] R. Ji, L.-Y. Duan, J. Chen, H. Yao, Y. Rui, W. Gao, Location discriminative vocabulary coding for mobile landmark search, International Journal of Computer Vision 96 (2012) 290–314.
[3] G. Schroth, R. Huitl, D. Chen, M. Abu-Alqumsan, A. Al-Nuaimi, E. Steinbach, Mobile visual location recognition, IEEE Signal Processing Magazine 28 (2011) 77–89.
[4] S.S. Tsai, D. Chen, V. Chandrasekhar, G. Takacs, N.-M. Cheung, R. Vedantham, R. Grzeszczuk, B. Girod, Mobile product recognition, in: ACM International Conference on Multimedia, 2010.
[5] Google, Google Goggles, 2011. 〈http://www.google.com/mobile/goggles〉.
[6] PlinkArt, PlinkArt, 2010. 〈http://www.androidtapp.com/plinkart〉.
[7] S.S. Tsai, H. Chen, D. Chen, G. Schroth, R. Grzeszczuk, B. Girod, Mobile visual search on printed documents using text and low bit-rate features, in: IEEE International Conference on Image Processing, 2011.
[8] P. Belhumeur, D. Chen, S. Feiner, D. Jacobs, W. Kress, H. Ling, I. Lopez, R. Ramamoorthi, S. Sheorey, S. White, L. Zhang, Searching the world's herbaria: a system for visual identification of plant species, in: European Conference on Computer Vision, 2008.
[9] D.G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2004) 91–110.
[10] H. Bay, A. Ess, T. Tuytelaars, L.V. Gool, Speeded-up robust features (SURF), Computer Vision and Image Understanding 110 (2008) 346–359.
[11] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (2005) 1615–1630.
[12] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, Y. Reznik, R. Grzeszczuk, B. Girod, Compressed histogram of gradients: a low bitrate descriptor, International Journal of Computer Vision 96 (2012) 384–399.
[13] G. Takacs, V.R. Chandrasekhar, S.S. Tsai, D.M. Chen, R. Grzeszczuk, B. Girod, Unified real-time tracking and recognition with rotation-invariant fast features, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010.
[14] J. Sivic, A. Zisserman, Video Google: a text retrieval approach to object matching in videos, in: IEEE International Conference on Computer Vision, 2003.
[15] D. Nister, H. Stewenius, Scalable recognition with a vocabulary tree, in: IEEE Conference on Computer Vision and Pattern Recognition, 2006.
[16] G. Schindler, M. Brown, R. Szeliski, City-scale location recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007.
[17] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Lost in quantization: improving particular object retrieval in large scale image databases, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008.
[18] H. Jégou, M. Douze, C. Schmid, Improving bag-of-features for large scale image search, International Journal of Computer Vision 87 (2010) 316–336.
[19] C. Yeo, P. Ahammad, K. Ramchandran, Rate-efficient visual correspondences using random projections, in: IEEE International Conference on Image Processing, 2008.
[20] A. Torralba, R. Fergus, Y. Weiss, Small codes and large image databases for recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008.
[21] Y. Weiss, A. Torralba, R. Fergus, Spectral hashing, in: Neural Information Processing Systems, 2008.
[22] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, J. Singh, B. Girod, Transform coding of image feature descriptors, in: SPIE Conference on Visual Communications and Image Processing, 2009.
[23] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, J.P. Singh, B. Girod, Tree histogram coding for mobile image matching, in: IEEE Data Compression Conference, 2009.
[24] R. Ji, L.-Y. Duan, J. Chen, H. Yao, Y. Rui, S.-F. Chang, W. Gao, Towards low bit rate mobile visual search with multiple-channel coding, in: ACM International Conference on Multimedia, 2011.
[25] H. Jégou, M. Douze, C. Schmid, Product quantization for nearest neighbor search, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (2011) 117–128.
[26] M. Calonder, V. Lepetit, C. Strecha, P. Fua, BRIEF: binary robust independent elementary features computer vision, in: European Conference on Computer Vision, 2010.
[27] S.S. Tsai, D.M. Chen, G. Takacs, V. Chandrasekhar, J.P. Singh, B. Girod, Location coding for mobile image retrieval, in: International ICST Mobile Multimedia Communications Conference, 2009.

[28] Y. Reznik, G. Cordara, M. Bober, Evaluation framework for Compact Descriptors for Visual Search, in: ISO/IEC JTCI/SC29/WG11 N12202.
[29] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, R. Vedantham, R. Grzeszczuk, B. Girod, Inverted index compression for scalable image matching, in: IEEE Data Compression Conference, 2010.
[30] H. Jégou, M. Douze, C. Schmid, P. Pérez, Aggregating local descriptors into a compact image representation, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010.
[31] F. Perronnin, Y. Liu, J. Sánchez, H. Poirier, Large-scale image retrieval with compressed fisher vectors, in: IEEE Conference on Computer Vision and Pattern Recognition, 2010.
[32] M. Brown, G. Hua, S. Winder, Discriminative learning of local image descriptors, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (2011) 43–57.
[33] T. Fawcett, An introduction to ROC analysis, Pattern Recognition Letters 27 (2006) 861–874.
[34] J. Swets, Signal Detection Theory and ROC Analysis in Psychology and Diagnostics, Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.
[35] J. Philbin, A. Zisserman, Oxford Building Data Set, 2007.
[36] H. Stewenius, D. Nister, University of Kentucky Benchmark Data Set, 2006.
[37] H. Shao, T. Svoboda, L.V. Gool, ZuBuD—Zurich Buildings Data Set, 2003.
[38] D. Chen, S. Tsai, B. Girod, Stanford Media Cover Data Set, 2009.
[39] Y. Yang, F. Nie, D. Xu, J. Luo, Y. Zhuang, Y. Pan, A multimedia retrieval framework based on semi-supervised ranking and relevance feedback, IEEE Transactions on Pattern Analysis and Machine Intelligence 34 (2012) 723–742.
[40] H. Wang, S. Yan, D. Xu, X. Tang, T. Huang, Trace ratio versus ratio trace for dimensionality reduction, in: IEEE Conference on Computer Vision and Pattern Recognition, 2007.
[41] M. Wang, X.-S. Hua, R. Hong, J. Tang, G.-J. Qi, Y. Song, Unified video annotation via multigraph learning, IEEE Transactions on Circuits and Systems for Video Technology 19 (2009) 733–746.
[42] J. Song, Y. Yang, Z. Huang, H.T. Shen, R. Hong, Multiple feature hashing for real-time large scale near-duplicate video retrieval, in: ACM International Conference on Multimedia, 2011.
[43] S.S. Tsai, D. Chen, H. Chen, C.-H. Hsu, K.-H. Kim, J. P. Singh, B. Girod, Combining image and text features: a hybrid approach to mobile book spine recognition, in: ACM International Conference on Multimedia, 2011.
[44] D. Chen, G. Baatz, K. Koser, S. Tsai, R. Vedantham, T. Pylvanainen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, R. Grzeszczuk, City-scale landmark identification on mobile devices, in: IEEE Conference on Computer Vision and Pattern Recognition, 2011.
[45] D. Chen, V. Chandrasekhar, G. Takacs, S. Tsai, M. Makar, R. Vedantham, R. Grzeszczuk, B. Girod, Compact Descriptors for Visual Search: improvements to the test model under consideration with a global descriptor, in: ISO/IEC JTCI/SC29/WG11 M23578.
[46] G. Takacs, V. Chandrasekhar, D. Chen, S. Tsai, R. Vedantham, R. Grzeszczuk, B. Girod, Compact Descriptors for Visual Search: Stanford Nokia integrated fast features, in: ISO/IEC JTCI/SC29/WG11 M22553.
[47] A. Oliva, A. Torralba, Modeling the shape of the scene: a holistic representation of the spatial envelope, International Journal of Computer Vision 42 (2001) 145–175.
[48] M. Douze, H. Jégou, H. Sandhawalia, L. Amsaleg, C. Schmid, Evaluation of GIST descriptors for web-scale image search, in: International Conference on Image and Video Retrieval, 2009.
[49] D. Chen, N.-M. Cheung, S. Tsai, V. Chandrasekhar, G. Takacs, R. Vedantham, R. Grzeszczuk, B. Girod, Dynamic selection of a feature-rich query frame for mobile video retrieval, in: IEEE International Conference on Image Processing, 2010.
[50] V. Chandrasekhar, D. Chen, S. Tsai, N.-M. Cheung, H. Chen, G. Takacs, Y. Reznik, R. Vedantham, R. Grzeszczuk, J. Bach, B. Girod, The Stanford mobile visual search data set, in: ACM Conference on Multimedia Systems, 2011.
[51] M. Persin, Document filtering for fast ranking, in: ACM SIGIR Conference on Research and Development in Information Retrieval, 1994.
[52] V.N. Anh, O. de Kretser, A. Moffat, Vector-space ranking with effective early termination, in: ACM SIGIR Conference on Research and Development in Information Retrieval, 2001.
[53] D. Chen, S. Tsai, R. Vedantham, R. Grzeszczuk, B. Girod, Streaming mobile augmented reality on mobile phones, in: International Symposium on Mixed and Augmented Reality, 2009.