Towards Low Bit Rate Mobile Visual Search with Multiple-Channel Coding

Rongrong Ji[*†‡]
[*]Peking University,
Beijing, 100871, China
rrji@pku.edu.cn

Ling-Yu Duan[*], Jie Chen[*]
[*]Peking University,
Beijing, 100871, China
{lingyu,cjie}@pku.edu.cn

Hongxun Yao[†]
[†]Harbin Institute of Technology,
Harbin, 150001, China
yhx@vilab.hit.edu.cn

Yong Rui[‡]
[‡]Microsoft China
Beijing, 100190, China
yongrui@microsoft.com

Shih-Fu Chang[♯]
[♯]Columbia University
New York, 10027, USA
sfchang@ee.columbia.edu

Wen Gao[*†]
[*]Peking University
Beijing, 100871, China
wgao@pku.edu.cn

ABSTRACT

In this paper, we propose a multiple-channel coding scheme to extract compact visual descriptors for low bit rate mobile visual search. Different from previous visual search scenarios that send the query image, we make use of the ever growing mobile computational capability to directly extract compact visual descriptors at the mobile end. Meanwhile, stepping forward from the state-of-the-art compact descriptor extractions, we exploit the rich contextual cues at the mobile end (such as GPS tags for mobile visual search and 2D barcodes or RFID tags for mobile product search), together with the visual statistics at the reference database, to learn multiple coding channels. Therefore, we describe the query with one of many forms of high-dimensional visual signature, which is subsequently mapped to one or more channels and compressed. The compression function within each channel is learnt based on a novel robust PCA scheme, with specific consideration to preserve the retrieval ranking capability of the original signature. We have deployed our scheme on both iPhone4 and HTC DESIRE 7 to search ten million landmark images in a low bit rate setting. Quantitative comparisons to the state-of-the-arts demonstrate our significant advantages in descriptor compactness (with orders of magnitudes improvement) and retrieval mAP in mobile landmark, product, and CD/book cover search.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; I.4 [**Image Processing and Computer Vision**]: [Scene Analysis, Object recognition]

General Terms

Algorithms, System, Measurement

*Area chair: Marcel Worring

MM'11, November 28–December 1, 2011, Scotsdale, Arizona, USA.

Keywords

Mobile Visual Search, Compact Descriptor, Contextual Learning, Wireless Communication, Data Compression

1. INTRODUCTION

Handheld mobile devices, such as smart camera phones, have great potential for emerging mobile visual search and augmented reality applications, such as location recognition, scene retrieval, product search, or CD/book cover search. In real-world mobile visual search applications, the reference database typically has millions of images and can only be stored on the remote server(s). Therefore, online querying involves transferring the query image from the mobile device to the remote server. This query delivery is often over a relatively slow wireless link. As a result, the quality of the user experience heavily depends on how much information has to be transferred. This issue becomes even more crucial when we move towards streaming augmented reality applications.

Indeed, this time consuming query delivery is often unnecessary, since the server only performs similarity search rather than query image reconstruction. With the ever growing computational power on mobile devices, recent works have proposed to directly extract compact visual descriptor(s) from the query image on the mobile end [1][2][3][4][5], and then send this descriptor over the wireless link with a low bit rate. This descriptor is expected to be compact, discriminative, and meanwhile efficient in extraction to reduce overall query delivery latency. In particular, the research, development, and standardization of compact descriptors for visual search are involving big industry efforts from Nokia, Qualcomm, Aptina, NEC, *etc.* [6].

Aside from low latency wireless query delivery, the direct extraction of compact descriptors on the mobile end may also lighten the overcrowding network communication in the remote server infrastructure. Different from processing textual queries, the online visual search systems face a bandwidth challenge when simultaneously receiving tens of thousands of image queries. How to optimize the bandwidth consumption for scalable visual search remains yet to be explored. On the other hand, a mobile device is constrained by the energy of the battery, hence saving energy is critical. However, sending the entire image or high-dimensional signature throughout 3G network connections will result in serious energy consumption. On the contrary, as quanti-

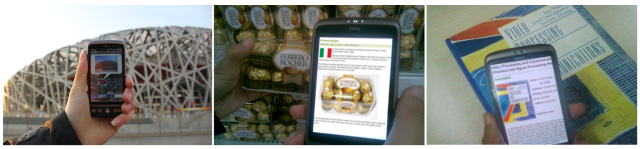

Figure 1: The developed mobile visual search system with Multiple-channel Coding based compact Visual Descriptor (MCVD). Our system is deployed in both the HTC DESIRE G7 and the iPhone4 smart phones, working for applications such as mobile landmark search, mobile product search, and mobile CD/Book cover search.

tatively reported in this paper, it is more power saving to extract and send a very compact descriptor instead.

Towards low bit rate mobile visual search, previous local descriptors *e.g.* SIFT [7], SURF [8], and PCA-SIFT [9] are *over size*: Sending hundreds of such local descriptors typically costs over the size of the original image. On the other hand, aiming for zero latency wireless query delivery, the state-of-the-art compact local descriptors [2][3] are still not compact enough. For instance, to the best of our knowledge, the state-of-the-art CHoG descriptor [2] still involves $50 \times n$ bits, where n is the number of local features per image, typically over hundreds. Intuitively, a possible extremely "compact" solution seeks to compress these local descriptors into a Bag-of-Words (BoW) histogram [1]. But to maintain sufficient discriminability, the BoW histogram typically involves over thousands of codewords (as demonstrated in most existing works [1][2][10]), which degrades the compactness potential of the sent descriptor.

We argue that such a high-dimensional image signature (*e.g.* BoW histogram) can be further compressed, without any serious loss in its search performance. Given the state-of-the-art effective high-dimensional image signatures, we aim to learn a compression function to further convert these signatures into an extremely compact descriptor (say, hundreds of bits) at the mobile end. Our main idea is to exploit the rich contextual cues inherent to the mobile end (*e.g.* GPS, 2D barcode, or RFID tags) to supervise the design of descriptor compression, which can significantly reduce the descriptor size[1] by a supervised compression function learning to preserve the ranking capability of the raw high-dimensional signature.

We propose a **M**ultiple-channel **C**oding based compact **V**isual **D**escriptor (**MCVD**) to achieve the above goal. The MCVD consists of both offline learning and online extraction phases: Its offline learning involves learning an optimal channel division, as well as an optimal descriptor compression with respect to each channel:

- The channel division adaptively maps each reference database image into its best-matched "channel", either based on its contextual cues (*e.g.* GPS) or based solely on its visual statistics (*e.g.* BoW or GIST).

- Within each channel, we introduce a supervised robust PCA scheme for descriptor compression, with two contributions: (1) it captures the main visual statistics within reference images in this channel by a low rank matrix recovery, which filters out indiscriminative object appearances from the BoW histograms within this channel (*e.g.* intruding foreground objects and irrelevant background trees in landmark search, background clutters in product search); (2) it incorporates the ranking precision loss of these reference images to supervise the dimension reduction procedure, which preserves the retrieval capability of the original BoW.

The online extraction of MCVD converts an initial high-dimensional visual signature (*e.g.* BoW histogram) into a compact descriptor, typically hundreds of bits for zero-latency wireless transferring. It consists of two steps:

- map the query image into its corresponding channel based on either its visual statistics, or its related contextual cues, or both;

- compress the visual signature using a channel-specific function to transform the high dimensional signature into a low-dim, hit/non-hit binary code.

Finally, the MCVD outputs a compact, discriminative, and channel-specific binary code sequence with further entropy coding. It is decoded to regenerate the original BoW again at the server end. In essence, MCVD undergoes a sort of lossy compression but still preserves a high search precision with extremely small amount of bits, which can incorporate the contextual supervision into both the channel division and the compression learning stages.

Taking mobile landmark search for instance, we maintain a visual vocabulary of moderate size to search worldwide landmarks: Once a mobile user enters a given city, the server alerts the mobile to receive a downstream adaption (*e.g.* the channel division function with multiple intra-channel compression functions) to "teach" the mobile how to adaptively extract MCVD in this city. In online search, the mobile device firstly extracts an original high-dimensional descriptor, and compresses it into a compact MCVD descriptor for delivery. Figure 2 shows our scheme with contributions in the following three-fold:

(1) the extracted visual descriptor is extremely compact (*e.g.* typically 100 bits per query in landmark search). Comparing with the widely used BoW for mobile visual search, we achieve almost identical mAP at an 1:4000 compression rate. Comparing with the state-of-the-art CHoG descriptors [2], we achieve an 1:64 compression rate. Comparing with sending a small-size query image (typically 20KB), we even achieve an 1:1500 to 1:3125 compression rate.

(2) the channel division and intra-channel compression learning merit context aware scenarios, which well exploits the pervasive contextual cues in mobile computing (such as GPS, 2D barcodes, Base station, as well as RFID tags).

(3) in intra-channel compression learning, we pay specific attentions to the computational efficiency and storage feasibility on the mobile end, achieved by an offline supervised robust PCA which uses singular vector decomposition.

We introduce our low bit rate mobile visual search pipeline in Section 2. We then present our channel division and intra-channel compression in Section 3. Section 4 demonstrates our real-world implementations, including mobile landmark search and product/CD/book cover search. The former is validated in a ten million web landmark image database, as well as a landmark search benchmark; while the latter is validated on public available product/CD/book cover benchmarks. Finally, we review related work in Section 5 and

[1]We will show in Section 4 that in some cases it could even improve the search discriminability

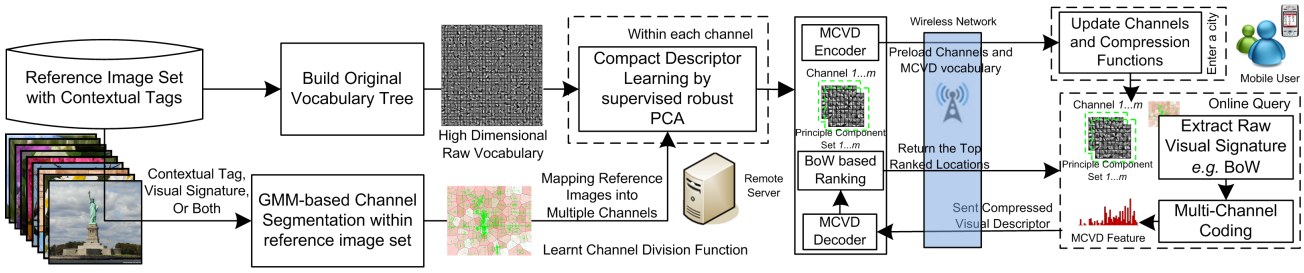

Figure 2: The proposed low bit rate mobile visual search framework using multiple-channel coding. Different from previous works in near-duplicated visual search, this framework emphasizes on extremely compact descriptor extraction directly on the mobile end. Towards zero-latency query delivery, approximate hundreds of bits visual descriptor is typically extracted from each query. To the best of our knowledge, it is the most compact descriptor with comparable discriminability to the state-of-the-arts [1][2][10][11].

discuss future research topics of the low bit rate mobile visual search in Section 6.

2. A LOW BIT RATE MOBILE VISUAL SEARCH FRAMEWORK

The proposed low bit rate mobile visual search framework follows the pipeline in Figure 2: At the mobile end, we extract the raw visual signature (such as local features or a Bag-of-Words histogram) from the query photo and compress (encode) this signature into a very compact descriptor. Instead of sending the query photo, we send this compact descriptor through a wireless link like a 3G network connection or WLAN at a low bit rate. At the remote server, this compact descriptor is decoded to form the raw visual signature, based on which we rank the image similarity using scalable indexing techniques such as vocabulary tree [10]. The key design of the above pipeline is our Multiple-channel Coding based compact Visual Descriptor (MCVD) in Section 3, which in a general sense integrates visual analysis, machine learning, and coding to provide an intelligent functionality in descriptor compression. In this section, we first introduce the pipeline of our low bit rate mobile visual search.

2.1 The Search Pipeline

Our MCVD consists of four phases as in Figure 2:

Step (1): The first phase is a "channel selection" in the mobile end. Its input can be any kind of initial high-dimensional visual descriptors, possibly with context cues. A typical instance is to use the Bag-of-visual-Words (BoW) histogram [10]. By comparing visual similarity, or contextual tag (such as GPS) based similarity, or both, the "channel selection" decides which channel(s) to use for mapping the raw BoW for the subsequent intra-channel compression.

Step (2): The second phase is to encode the raw BoW of an image with respect to the selected channel, which outputs a compact descriptor as our final MCVD result. This descriptor is further compressed into a codeword occurrence (hit/non-hit) histogram with Huffman coding.

Step (3): The encoded signature (with entropy coding), together with its channel identification, are transmitted over the wireless network.

Step (4): In the remote server, the decoding procedure is performed, following the pipeline of Huffman decoding, MCVD decoding, and BoW regeneration. The decompressed BoW is subsequently used for inverted indexing based search. In addition, BoW based spatial re-ranking can be applied, which however involves additional sending and matching the spatial layout coding of interest points [12].

2.2 Raw BoW Extraction with VT

Towards scalable visual search in a million scale database, vocabulary tree (VT) [10] based techniques were well exploited in previous works [1][13][14][15]. VT employs hierarchical K-means to partition local descriptors into quantized codewords. An h-depth VT with b-branch produces $m = b^h$ codewords, and the scalable search typically settles $h = 5$ and $b = 10$ [10]. Given a query photo I_q with J local descriptors $\mathbf{L}(q) = [L_1(q), ..., L_J(q)]$, VT quantizes $\mathbf{L}(q)$ by traversing the vocabulary hierarchy to find the nearest codeword, which converts $\mathbf{L}(q)$ to a BoW histogram $\mathbf{V}(q) = [V_1(q), ..., V_m(q)]$. In retrieval, an optimal ranking is supposed to minimize the following loss with respect to the ranking position $R(x)$ of each I_x (BoW feature $\mathbf{V}(x)$):

$$Loss_{Rank}(q) = \sum_{x=1}^{n} R(x)\ \mathbf{W}_x \cdot (\mathbf{V}(q) - \mathbf{V}(x)) \qquad (1)$$

where TF-IDF weighting is calculated in a way similar to its original form [16] in document retrieval:

$$\mathbf{W}_x = [\frac{n_1^x}{n^x} \times log(\frac{n}{n_{V_1}}), ..., \frac{n_m^x}{n^x} \times log(\frac{n}{n_{V_m}})] \qquad (2)$$

where n^x denotes the number of local descriptors in I_x; $n_{V_i}(x)$ denotes the number of local descriptors in I_x quantized into V_i; n denotes the total number of images in the database; n_{V_i} denotes the number of images containing V_i; $\frac{n_i^x}{n^x}$ serves as the term frequency of V_i in I_x; and $log(\frac{n}{n_{V_i}})$ serves as the inverted document frequency of V_i.

3. THE CHANNEL LEARNING PRINCIPLE

The state-of-the-art visual descriptors are typically high-dimensional and cost large amount of bits in wireless delivery. For instance, the well known SIFT local descriptor is 128×8 bits, which involves $128 \times 8 \times n$ bits per query (with n local features; n typical over hundreds). The state-of-the-art CHoG descriptor also involves $50 \times n$ bits. To further reduce the delivery size, the "bag" of local descriptors can be further quantized into a BoW, followed by a histogram compression such as [1]. However, to maintain sufficient discriminability, the histogram dimension of BoW is at least

10,000 (typically contains 0.1 to 1 million codewords in the state-of-the-art settings), which is against the compactness.

We divide and conquer this issue using *Channel Coding*, which maps each descriptor into a corresponding channel and then compress them correspondingly, making an optimal tradoff between compression rate and discriminability. The learnt channels and coding functions are of small size and can be online downloaded to the mobile or can be also easily pre-installed in the mobile end.

Learning Goal: Given database images $\mathbf{I} = \{I_i\}_{i=1}^n$, we offline extract n m-dim BoW [10][17] $\mathbf{V} = \{V_i\}_{i=1}^n$, typically is with dimensionality $m = 0.1 - 1$ million. All images may be bounded with contextual information such as GPS or base station tags. We aim to (1) learn a channel division function $\mathbf{C} = \{C_j\}_{j=1}^M$ to partition $\mathbf{I} = \{I_i\}_{i=1}^n$ into M channels, where $I_i \in C_j$ denotes I_i is assigned to channel C_j. The division function learning typically involves the contextual supervision *e.g.* GPS tags. (2) learn a codebook $\mathbf{U}_j \in \mathbb{R}_k$ for compact descriptor extraction in each C_j from $\mathbf{V} \in \mathbb{R}_m$, such that $k \ll m$, all of which are either pre-installed, or online updated to the mobile device once the mobile device is assigned to a given channel C_j, for example, the mobile user enters a geographical region.

3.1 Channel Definition

A channel refers to a way to subdivide (partition) the reference database, based on which a raw high-dimensional visual signature (such as BoW histogram) is mapped into one of these channels[2]. Within each channel, this signature is extremely compressed to form a channel dependent compact descriptor. For each database (such as a city scale landmark database or a product/CD/book cover database), there is a channel division function to partition the database images. Each channel has its own compression function. An example of channel constitution is shown in Figure 3.

The channel learning can be based on contextual tags (*e.g.* GPS or RFID) or can be based solely on the visual statistics of the high-dimensional signatures. Even when the original signature is delivered into a wrong channel, this signature is still compressed, but would result in less discriminative but compact descriptors.

3.2 Adaptive Channel Division Learning

Taking mobile landmark search as an example, we assume there are GPS tags $\mathbf{X} = \{x_i\}_{i=1}^n$ available. We then adopt a light but durable Gaussian Mixture Model (GMM) [18] to segment $\mathbf{I}$ into M channels. The geo-tagged database photos are assumed to be drawn from M landmark regions. We denote the ith component as w_i, with mean vector μ_i. We regard the GPS tags $\mathbf{X}$ of those photos belonging to the ith component as generated from the ith Gaussian with mean μ_i and covariance matrix $\sum$, following a normalized distribution $N(\mu, \sum)$.

Hence, we assign the jth photo into the ith channel based on its Bayesian posterior probability:

$$p(y = i|x_j) = \frac{p(x_j|y = i)P(y = i)}{p(x_j)} \quad (3)$$

where $p(x_j|y = i)$ is the probability that the channel (region)

[2]Here, the definition of channel differs from that in communication, which refers either to a physical transmission medium such as a wire, or to a logical connection over a multiplexed medium such as a radio channel.

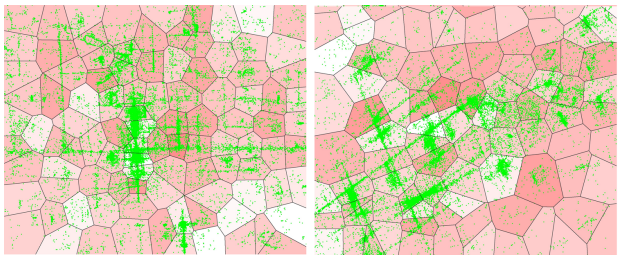

Figure 3: The channel division effects in mobile landmark search. GPS are employed in channel division and selection. In the above subfigures, the green nodes denote the spatial distribution of photos, and the black lines delineate the regions of different channels. The different saturation of red color reflects the length of MCVD. The left subfigure is the channel division in Beijing, the right subfigure is the channel division in New York City.

y of x_j is C_i, which follows the normalized distribution:

$$p(x_j|y = i) = \frac{exp\left[-\frac{1}{2}(x_j - \mu_i)^T \sum_i^{-1}(x_j - \mu_i)\right]}{(2\pi)^{\frac{M}{2}}||\sum_i||^{\frac{1}{2}}} \quad (4)$$

where $(x_j - \mu_i)$ in such a case denotes the geographical distance between the jth photo and the ith region centroid.

Neither the component parameters nor the channel assignments for photos are known. We adopt an expectation maximization for segmentation: First, we estimate a Gaussian Mixture Model at the tth iteration denoted by (t) as:

$$\lambda_t = \left\{\mu_1(t), ...\mu_M(t), \sum_1(t), ... \sum_M(t), P_1(t), ...P_M(t)\right\} \quad (5)$$

Subsequently, we utilize an Expectation Maximization procedure to learn the optimal Gaussian component assignment $p(y = i|x_j, \lambda_t)$ for each x_j and the optimal Gaussian component parameters (P_i and μ_i) respectively.

In addition, we may lighten the influence of contextual supervision in channel division, since in some cases these tags are unavailable. In such case, the $p(x_j|y = i)$ in Equation 3 is refined based solely on the visual similarity between the jth photo and the ith Gaussian component as:

$$p(x_j|y = i) = \frac{exp\left[-\frac{1}{2}(F_j - F_{\mu_i})^T \sum_i^{-1}(F_j - F_{\mu_i})\right]}{(2\pi)^{\frac{M}{2}}||\sum_i||^{\frac{1}{2}}} \quad (6)$$

F_j can refer to any easily computed set of visual features *e.g.* GIST or color histogram. In addition, features in Equation 6 can be (if needed) also a normalized combination of both contextual and visual features *e.g.* $< F_j, x_j >$.

3.3 Learning Intra-Channel Compression

To learn an optimal compression function within each channel, we have to address three practical issues: (1) the initial BoW histogram contains lots of irrelevant or noisy codewords, probably arising from intruding foreground objects and irrelevant background objects in mobile landmark search, or background clutters in mobile product search; (2) the storage of compression function should be minimal, for instance it is impossible to store an $1,000,000 \times 1000$ dimension reduction matrix, unless it is really sparse or has low rank (the former can be stored as a sparse matrix, and the latter can be efficiently approximated by using its singular

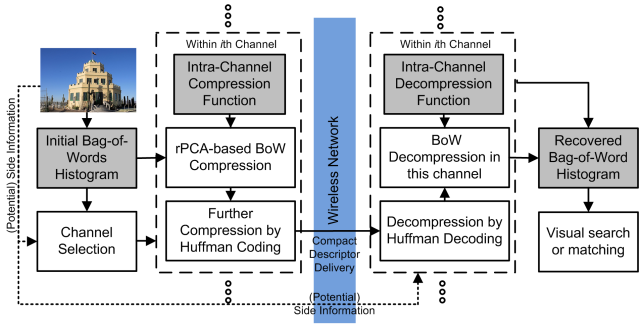

Figure 5: The channel learning principle consists of two phases: The first phase is to learn an optimal channel division based on either contextual information or visual statistics, or both. The second phase is to learn an optimal compression function with respect to each channel, based on the robust PCA learning with (possible) contextual supervision.

vectors); (3) the learnt dimension reduction function should be sufficiently discriminative in search, or in other words can preserve the ranking loss of the original high-dimensional BoW (without any compression) as much as possible.

We achieve the above goals based on a novel supervised robust PCA learning, with three basic considerations: (1) we learn a sparse error matrix to separate the noisy and unrepresentative visual elements from the raw BoW signatures in a given channel, based on which only the most representative and discriminative codewords are preserved; (2) we learn a low rank matrix for PCA extraction, based on which we only need to store the resulting singular vectors in the mobile storage card; (3) we allow to integrate the contextual supervision into the low rank matrix learning, which models the image ranking loss after compression to supervise the robust PCA extraction to preserve its ranking capability.

Ranking List for Learning: Given a channel C containing n' photos $[I_1, I_2, ..., I_{n'}]$, we randomly sample n_{sample} photos $[I'_1, I'_2, ..., I'_{n_{sample}}]$ as queries, which generates the following ranking list for this channel C:

$$\begin{aligned} Query(I'_1) &= [A_1^1, A_2^1, ..., A_R^1] \\ &... \\ Query(I'_{n_{sample}}) &= [A_1^{n_{sample}}, A_2^{n_{sample}}, ..., A_R^{n_{sample}}] \end{aligned} \tag{7}$$

A_i^j is the ith element in the ranked result of the jth query; R is the number of result returned. By compressing the initial BoW, the resulting MCVD descriptor is expected to preserve the ranking order $[A_1^j, A_2^j, ..., A_R^j]$ for the jth query as much as possible. Hence, we inject the above ranking loss to supervise intra-channel learning[3].

Supervised Robust PCA learning: Given in total n_{sample} ranking lists with m-dimensional BoW in each channel, we formulate the original signature matrix as $\mathbf{D} \in \mathbb{R}^{m \times n}$. Typically, there are lots of noisy words in $\mathbf{D}$ from patches like foreground cars, people, and trees (for landmark search), or background clusters (for product/CD/book cover search). These words are meaningless in retrieval and can be modeled as additive errors $\mathbf{E} \in \mathbb{R}^{m \times n}$. Therefore, similar to [19], we consider $\mathbf{D}$ as being generated from a purified, low-rank matrix $\mathbf{P} \in \mathbb{R}^{m \times n}$[4] with an additive error matrix $\mathbf{E}$:

$$\mathbf{D} = \mathbf{P} + \mathbf{E} \tag{8}$$

The solution of the above question can be exactly learnt through optimizing the following loss with $\mathcal{L}_0$ form:

$$Min_{(\mathbf{P},\mathbf{E})} rank(\mathbf{P}) + \gamma ||\mathbf{E}||_0 \quad subj \ \ to \ \ \mathbf{D} = \mathbf{P} + \mathbf{E} \tag{9}$$

Due to the nonconvex nature of Equation 9, following [19], we relax Equation 9 into the reformulation of the nuclear form of the low rank matrix $||\mathbf{P}||_* = \sum_i^k \sigma_i(\mathbf{P})$ with k singular vectors, together with the $\mathcal{L}_1$ form of the sparse error matrix $\mathbf{E}$. We then approximate the optimal $\mathbf{P}$ and $\mathbf{E}$ as:

$$Min_{(\mathbf{P},\mathbf{E})} ||\mathbf{P}||_* + \gamma ||\mathbf{E}||_1 \quad subj \ \ \mathbf{D} = \mathbf{P} + \mathbf{E} \tag{10}$$

where $||\mathbf{P}||_* + \gamma||\mathbf{E}||_1$ is the convex envelope of $rank(\mathbf{P}) + \gamma||\mathbf{E}||_0$. As stated in [19], for almost all pairs $(\mathbf{P}_0; \mathbf{E}_0)$ consisting of a low-rank matrix $\mathbf{P_0}$ for compression matrix learning and a sparse matrix $\mathbf{E}_0$ for error removal, there is a uniquely definite minimizer:

$$\begin{aligned} (\mathbf{P}_0, \mathbf{E}_0) = \arg Min_{(\mathbf{P},\mathbf{E})} &||\mathbf{P}||_* + \gamma||\mathbf{E}||_1 \\ subj \quad &\mathbf{D} = \mathbf{P} + \mathbf{E} \end{aligned} \tag{11}$$

Following the setting of [19], we also have $\gamma = m^{-\frac{1}{2}}$. Then the PCA is learnt from $\mathbf{P}$ by calculating a singular value decomposition matrix $\mathbf{M}_{m \times k}$ with k singular vectors to represent the original m dimensional signature. Such PCA decompositions subsequently produce the new $\mathbf{U}$ based on $\mathbf{U} = \mathbf{M}^T\mathbf{P}$, where $\mathbf{M}$ is built upon k singular vectors of $\mathbf{P}$. Subsequently, we only store the k singular vectors $\{\sigma'_i\}_{i=1}^k$ of $\mathbf{U}$ ($k \leq rank(\mathbf{U})$) on the mobile to compress BoW $\mathbf{V}$.

Beyond [19], we further incorporate the ranking loss into our objective function. Such loss comes from whether or not the learnt dimension reduction matrix ($M_{m \times k}$ from $\mathbf{P}$) can preserve the ranking list $[A_1^j, ..., A_R^j]$ for the jth query. To quantize this, we penalty the ranking loss for ith transformed BoW $\mathbf{U} \in \mathbb{R}_k$ from $\mathbf{V}_m\mathbf{M}_{m \times k}$:

$$\begin{aligned} &Loss(I'_i, \mathbf{P}) = \\ &\sum_{r=1}^{R} R(A_r^i) \mathbf{W}_{A_r^i} ||\mathbf{M}\mathbf{U}_j(I'_i), \mathbf{V}(A_r^i)||_{L2} \end{aligned} \tag{12}$$

where $i \in [1, n_{sample}]$; $R(A_r^i)$ is the position of rth returning for ith query I'_i. The overall ranking loss $Loss_{Rank}\mathbf{P}$ is:

$$\sum_{i=1}^{n_{sample}} \sum_{r=1}^{R} R(A_r^i) \mathbf{W}_{A_r^i} \cdot (\mathbf{M}^{t-1}\mathbf{U}_j(I'_i) - \mathbf{V}(A_r^i)) \tag{13}$$

The ideal compression in a given channel C is then formulated as the low rank recovery of $\mathbf{P}$, with respect to the sparse error matrix $\mathbf{E}$ together with the ranking loss, where the compromising parameter θ controls both factors:

$$\begin{aligned} Min_{(\mathbf{P},\mathbf{E})} &||\mathbf{P}||_* + \gamma||\mathbf{E}||_1 + \theta Loss_{Rank}\mathbf{P} \\ subj \quad &\mathbf{D} = \mathbf{P} + \mathbf{E} \end{aligned} \tag{14}$$

Optimization Procedure: Similar to [19], we adopt the proximal gradient approach in [20] to learn the optimal low-rank matrix $\mathbf{P_0}$ and a sparse matrix $\mathbf{E}_0$. In that sense,

[3] In the future, active sensing techniques can be also considered to sample queries for the subsequent learning.

[4] Our learning does not target at absolute low rank recovery of $\mathbf{P}$, instead our goal is to preserve the loss of ranking list distortions with the lowest number of singular vectors of $\mathbf{P}$.

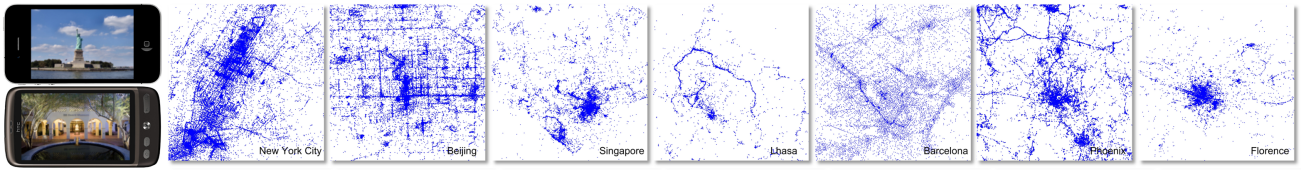

Figure 4: (a) Implementation platform of iPhone4 and HTC DESIRE G7; (b) Geographical photo distribution of *10MLandmarkWeb* in New York City, Beijing, Singapore, Lhasa, Barcelona, Phoenix, and Florence.

Table 1: Mobile memory (mb) and time (second) comparison (MCVD is with a vocabulary tree pruning to further reduce the storage space).

	Local Feature	BoW Generate	Mapping & Compression	Memory Cost
BoW	1.2s	0.14s	N/A	59MB(H=5, B=10)
MCVD	1.2s	0.038s	~ 0	0.69MB $\sim$ 3.44MB

the algorithm has a theoretical convergence rate of $O(t^{-2})$ where t is the iteration number. In each iteration, a Singular Value Decomposition is first calculated, then both $||\mathbf{P}||_*$ and loss in Equation 14 are evaluated. Learning is converged or stopped once the convergence rate is small enough, or the iteration reaches a given threshold (typically < 100)[5].

We summarize the input/output and pipeline of our rPCA based compression as three steps (for a higher-level search procedure please refer to Section 2.1): (1) Offline learn supervised rPCA and store the k principle components within each channel, which is online updated to the mobile; (2) Given an m-dim BoW histogram V extracted from the I_Q, the mobile device calculates the k principle components and binarizes them into k-bit hit/non-hit. (3) The k-bit MCVD descriptor is sent to the server, where the m-dim BoW is restored in a lossy manner.

Inter-Channel Coding: To address marginal queries or queries assigned to wrong channels, we may prefer to compress the raw signature by leveraging multiple channels, which is referred to *Inter-Channel Coding*, simulated to the Greedy N-Best Paths [15] in VT. Note that in such case, the resulting descriptor is in general N times larger than the single channel coding. In the remote server, the descriptor decoding follows a late fusion principle, which fuses the ranking list based on different decoded descriptors.

4. EXPERIMENTAL VALIDATIONS

Data Collection and Ground Truth Labeling: We have validated our proposed Multiple-channel Compact Visual Descriptor (MCVD) on both mobile landmark search and mobile product, CD/book cover search tasks. Our evaluations are conducted on the three following datasets:

(1) *The Ten Million Landmark images crawled from the Web* (*10MLandmarkWeb*): We have collected over ten million geographical tagged photos from both Flickr and Paranomia using their APIs. Our photo collection covers the following seven cities: New York City, Beijing, Singapore, Lhasa, Barcelona, Phoenix, and Florence. In each city, to label the queries and their ground truths, we choose the 30 most dense regions and 30 random regions from its geographical map. For each region, we ask volunteers to manually identify one or more dominant views. Then, all near-duplicated photos to a given view are picked out in both the current region and its nearby regions. Finally, we randomly sample 5 labeled photos from each region as the query, which forms 300 queries in total with ground truths in each city.

(2) *The PKU landmark Benchmark* (*PKUBench*): We also test our MCVD on the publicly available mobile landmark search benchmark dataset (the largest) from MPEG CDVS requirement subgroup [21], which contains 13,179 scene photos, organized into 198 landmark locations from the Peking University Campus. There are in total 6193 photos captured from professional cameras (SONY DSC-W290, Samsung Techwin <Digimax S830 / Kenox S830>, *etc.* with resolution 2592×1944) and 6986 from phone cameras (Nokia E72-1, HTC Desire, Nokia 5235, *etc.* with resolution 640×480, 1600×1200 and 2048×1536) respectively. Over 20 volunteers are involved in data acquisition, and each landmark is captured by a pair of volunteers: One using a professional camera and the other using a mobile phone, with a portable GPS. The average viewing angle variations are 10 degrees in nearby photos. Both blurring and shaking are more frequent occur in the mobile phone. There are 381 mobile queries from 198 landmarks (each contains 2 images), 33 reference images on average for each query. Each image in the reference list must be visually and truly matched against at least one of the queries of that object.

(3) *The Product and CD/book Cover Database* (*PCBD*) contains Stanford mobile visual search database [22], UK-Bench near-duplicated benchmark [10], and a book cover dataset collected by the authors. Each product/CD/book has a category tag, which in practice could be acquired in the mobile device through 2D barcode, logo, or RFID, *etc.* However, to simplify more targeted evaluation, we leverage PCBD for unsupervised channel learning only.

Memory and Time Cost: We deploy our MCVD descriptor on HTC DESIRE G7 and iPhone4, as shown in Figure 4 (a). HTC DESIRE G7 is equipped with an embedded camera with maximal 2592×1944 resolution, a Qualcomm ARMV7 processor with 998.4MHz frequency, a 512M ROM + 576M RAM memory, 4G SD extended storage, and an embedded GPS. iPhone4 is equipped with similar memory setting. Table 1 further shows the memory and time cost of MCVD on the mobile phone. In extracting MCVD descriptor, the most time-consuming job is the local feature extraction, which could be further accelerated by random sampling, instead of detecting interest points [2][7]. Basically, zero latency is a subjective concept, involving time costs of MCVD, query delivery, search, and results returning. MCVD cost is listed in Table 1. Search costs about 30 ms per query. About query delivery, for quantitative and fair comparisons, we prefer the query bit rate, as the exact

[5] It is worth to mention that we target a lossy compression. That is, we aim to learn P with lower rank; meanwhile the negative effects of enforcing low rank learning on the retrieval ranking loss is minimized. Therefore, our learning phase is to find out the compromise between low rank and ranking discriminability. In addition, as BoW is by nature sparse we can assume matrix $\mathbf{E}$ is sparse. The above assumptions have been empirically proved with its amazing performance in real-world settings.

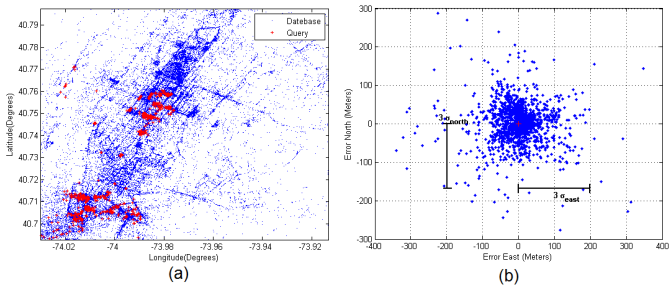

Figure 6: (a) the geographical distribution of the 600 queries in New York City; (b) the geographical error distribution of the search results to these queries (Queries are aligned in the red centroid point).

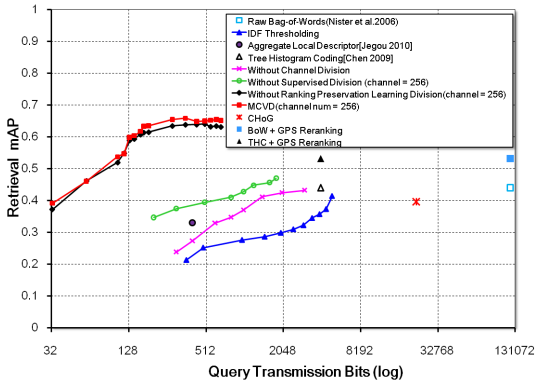

Figure 7: Rate distortion analysis of two typical cases of multiple-channel compact visual descriptor schemes with and without contextual information, with comparison to previous works in [1][2][10][11].

latency measurement in time heavily depends on the variable wireless connection.

Parameters: We extract SIFT [7] from each photo. A Vocabulary Tree [10] is built to generate the initial vocabulary $\mathbf{V}$, which outputs a BoW $V(i)$ for each photo I_i. We denote the hierarchical level as H and branching factor as B. In a typical setting, we have $H = 5$ and $B = 10$, producing approximately 0.1 million words. We maintain an initial identical vocabulary in the mobile end. To reduce the memory cost of VT, tree pruning operation is applied.

For each city (or each application), we learn its channel division and channel coding scheme respectively, which is online downloaded to the mobile device once the user enters a given city (for mobile landmark search), or the user activates the product search functionality (for product search). This downstream adaption costs small bandwidth but brings about great benefits in zero-latency query delivery, involves typically 80KB downstream data and can be downstream in less than 2s in a typical 3G network. In addition, with the channel division, tree pruning may contribute to very small memory cost of the tree as well as efficient quantization as shown in Table 1. Regarding the effects of different channel numbers and marginal queries located near channel boarders, we provide evaluations in "Insights into Channel Learning" and Figure 10 and 11 subsequently.

Evaluation Criterion: We use mean Average Precision (mAP) to evaluate the system performance. Given in total Q queries, mAP is defined as follows:

$$mAP = \frac{1}{N_q}\sum_{i=1}^{N_q}\left(\frac{\sum_{r=1}^{N} P(r)}{\#-of-relevant-images}\right) \qquad (15)$$

N_q is the number of queries; N the number of relevant images for the ith query; $P(r)$ is the precision at rank r.

Comparison Baselines: (1) *Raw Bag-of-Words*: Transmitting the entire BoW has the lowest compression rate. However, it provides an upper bound in mAP with any BoW compression strategies. (2) *IDF Thresholding*: As a straightforward scheme, we only transmit the IDs of codewords with the highest IDF values as an alternative solution for vocabulary Compression. (3) *Aggregating Local Descriptors [11]*: The work in [11] adopted aggregate quantization to obtain compact signature. Its output is also a compressed codeword histogram originated from an initial vocabulary V. (4) *Tree Histogram Coding [1]*: Chen *et al.* used residual coding to compress the BoW histogram, which is one of the most related works to our channel learning scheme. (5) *Without Channel Division*: To quantize the advantages of our channel division, we also degenerate our approach by ignoring any channel division, *i.e.*, performing compression function learning over only one channel within each entire database. (6) *Without Supervised Channel Division*: We further lighten the need of using contextual supervision in channel division to meet the scenarios of missing any contextual cues. We directly use visual clustering to subdivide each database into multiple channels, based on the L2 distance of the BoW signature of database images. To speed up the division process, only a small subset (sampling less than 10% of reference database images) is used to find out the cluster centroids. (7) *Without Ranking Preservation Learning*: We validate our ranking sensitive learning in each channel by removing the ranking cost in compression learning (Equation 14). As a result, we learn the dimension reduction in its original form, capturing only the visual statistics rather than its ranking preservation capability. As shown latter, it would also degenerate the descriptor effectiveness. (9) *CHoG [2]*: Finally, we compare our BoW compression based descriptor with the state-of-the-art compact local descriptor CHoG presented in [2].

Rate Distortion Analysis: Figure 7 compares the rate distortion with the state-of-the-arts in [1][2][10][11] in *10MLandmarkWeb*. We achieve the highest compression rate with equalized mAP distortions (horizontal view), or in other words maintain the highest search mAP with equalized compression rates (vertical view). We achieve less than 256 bits per image. Comparing with the raw high-dimensional BoW, we achieve identical mAP by 1:1,000 compression rate; comparing with the state-of-the-arts [1][2], we achieve 1:64; comparing with sending the JPEG query image (typically over 40 KB), we even achieve 1:2,500-1:3,125 compression rate. We also outperform Baseline (6) and (7) that are without supervised channel division or ranking preservation learning.

Insights into Channel Learning: *Impacts of Channel Number Variations*: One straightforward concern with MCVD might be the proper channel division by using contextual tags as well as the reliance on the context cues. So we provide two groups of quantitative evaluations.

First, we evaluate the impacts of different channel division numbers on MCVD in mobile landmark search. By setting the channel number identical to the database image number, we degenerate MCVD to GPS based location recognition. Figure 9 shows that directly using GPS cannot achieve the

Figure 8: Visualized Examples of the location recognition search results, as well as the product and book cover search results. Left image is the mobile photo query, the right images are the ranking list (the upper line: the ranking results using the initial Bag-of-Words; the lower line: the ranking results using our compact descriptor). The left subfigure shows the product, book search identities.

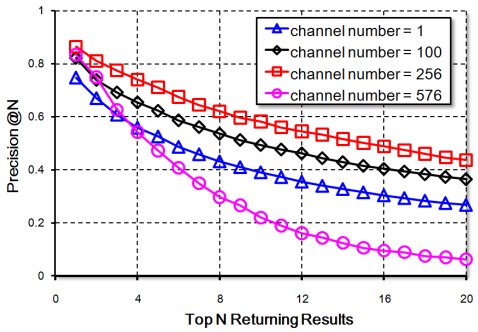

Figure 9: Precision@N with respect to the increasing of channel numbers in New York City. Note that neither too coarse nor too precise channel division is optimal, the later is partially due to some imprecise GPS tags that could result in assigning the test query into mistaken channels.

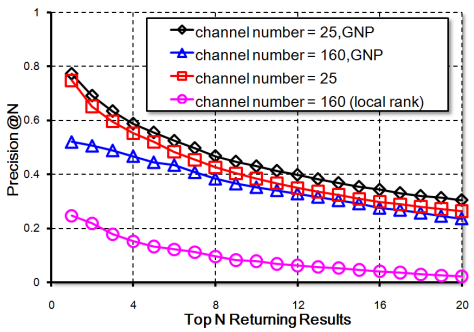

Figure 10: MCVD robustness in PKUBench by (a). adding contextual distractor. (b). for geographical marginal queries. "local ranking" denotes to search images in an individual channel only - seriously affected by marginal queries.

highest performance. One limitation comes from the noisy GPS acquisition, *e.g.* within dense buildings.

Second, without channel division (setting channel = 1), our scheme is degenerated into a solution without any contextual supervision. In such a case, our MCVD has a performance degeneration comparing with the well-tuned channel number (setting channel = 256), which reveals the advantages of contextual cues in helping visual descriptor extraction. In mobile landmark search experiments, given the best-tuned channel number, we have on average 3200 images within a radius range of on average 0.9km in each channel[6].

Channel Division Robustness: Our channel division faces another challenge from the marginal queries: It is natural to imagine that the channel division would affect the queries happening at the margin of nearby channels (such as geographical nearby partitions in mobile landmark search). To demonstrate our robustness, we choose a landmark search scenario and sample a set of 100 queries located within 50 meters of channel margins (ten for each channel). Simulating the Greedy N-Best Paths (GNP) [15] in vocabulary search, we also incorporate multiple channels in coding, followed by a late fusion at the client end after getting N ranking lists of images from the N channels (in such case the coding length would be approximate N times larger). However, Figure 10 shows that, even without GNP, considerable robustness is still obtained with the marginal query sampling. Indeed, performance is moderately acceptable comparing with entering the correct channel. It is because we are doing global ranking at the entire image database, rather than doing the local ranking only within the image subset in each channel.

[6]Regarding to the top 1-4 returning, different channel numbers have less negative affects. In practice, we use cross validation to assign a best channel for each scenario

Robustness in Mobile Photographing Scenarios: Based on PKUBench, we evaluate MCVD robustness in five real-world mobile photographing cases. Figure 11 shows the visualized query results as well as the mAP degeneration for each photographing scenario:

(1) *Occlusive Query* contains 20 mobile queries and 20 corresponding digital camera queries, occluded by foreground cars, people, and buildings.

(2) *Background Clutters* contains 20 mobile queries and 20 corresponding digital camera queries taken far away from the landmark, probably containing nearby buildings.

(3) *Night and Cloudy Query Set* contains 9 mobile phone queries and 9 corresponding digital camera queries. The night query is seriously affected by the lightings.

(4) *Blurring and Shacking Query Set* contains 20 mobile queries taken when the phone shacking and 20 corresponding digital camera queries with blurring or shacking.

(5) *Adding Distractors into Database* is to inject a set of distractor photos (6630 photos from Summer Palace provided in [21], visually similar to the landmark buildings in PKUBench) to PKUBench with random GPS tag within PKU compus.

Quantitative Validations in Product Search and CD/book Cover Search: We further validate our MCVD on the Product and CD/book Cover Database (PBDC), including three public available datasets including UKBench [10], Stanford MVS Benchmark [1], and a publicly available Book Cover database. For UKBench and Book Cover database, we sample the first image as the query and use the remaining images to build up the ground truth reference database. For Stanford MVS Benchmark, for each object category a query is matched to one top ground truth in reference dataset. Our method also reports promising mAP comparing with Baselines as shown in Figure 8 and Table 2.

Where the MCVD Codewords are Matched: Figure 12 further investigates where our MCVD descriptor is

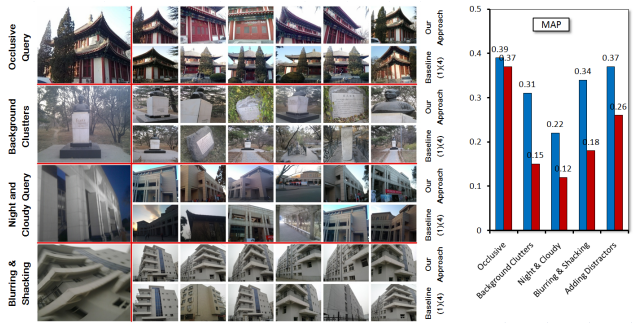

Figure 11: The robustness test of MCVD with respect to (1) Occlusive Query, (2) Background Clutters, (3) Night and Cloudy Query, (4) Blurring and Shacking Query, (5) Adding distractors. We compare our scheme with Baselines (1)(4) (blue bar: MCVD; red bar: BoW).

Table 2: mAP with respect to the compression rate for the product/CD/book cover databases.

	UKBench	Stanford MVS Benchmark	Book Cover
BoW	0.27/14.5KB	0.80/14.5KB	0.485/14.5KB
THS [1]	0.27/0.5KB	0.80/0.5KB	0.485/0.5KB
CHoG [2]	0.26/2.6KB	0.75/3.2KB	0.46/2.1KB
MCVD	0.35/0.4KB	0.88/0.2KB	0.62/200bits

matched from the query image to the database images, in which circles are associated with the matched local descriptors (recovered from MCVD) and different colors indicate different nonzero codewords (quantized from local descriptors). It is easy to see that, our MCVD only preserves a very small subset of local features detected in the query (originally approximate 300-400 features), which are dealt with as the most discriminative features.

Average Energy Consumption: On a mobile device, we are constrained by the battery life. Therefore, energy conserving is critical for mobile applications. One interesting investigation is the comparison of average mobile energy consumption in: (1) extracting and sending compact descriptors, (2) extracting and sending the BoW signature, and (3) sending the original query image. We empirically test the number of image queries the mobile can send before the battery runs out of power for 3G network connection. A typical phone battery has a voltage of 4.0V and a capacity of 1400mAH(or 20.2K Joules). Hence, for 3G connections, the maximum number of images that the mobile can send is 20.2k joules / 52.4 joules = 385 total queries. For the extraction and transmitting of our proposed multiple-channel compact visual descriptor, we would be able to perform 20.2k joules / 8.1 joules = 2494 total queries, which is 6× as many queries as transmitting the entire query image.

Our evaluations in Figure 13 reveals that sending both original query image and the high-dimensional descriptor would cost serious energy consumption, comparing with performing visual descriptor compression on the mobile and then sending the compact descriptor instead.

5. RELATED WORK

Combining with the ever growing wireless Internet services, mobile visual search has offered a wide range of applications, such as location recognition [14][23][24][25][26],

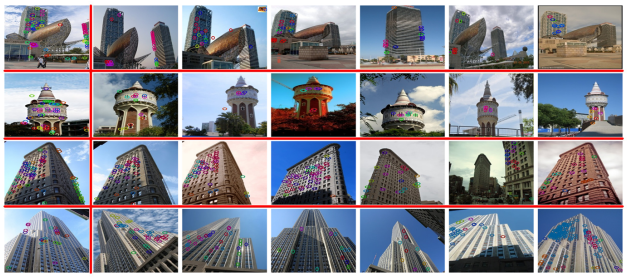

Figure 12: Case study of the spatial matching for our proposed low bit rate descriptors between query (left photo) and the top returning results. Different colors denote different nonzero codewords in the recovered BoW histogram from MCVD.

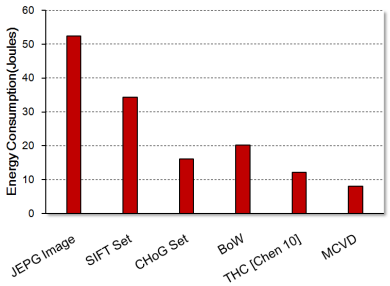

Figure 13: Average energy consumption comparison through the 3G wireless link, between transmitting the entire query image and the extraction and transmitting of MCVD and other compact descriptors.

landmark retrieval [27][28][29] product search, and CD/book cover search [5][10][12]. In general, most state-of-the-art mobile visual search systems are deployed based on approximate similarity matching techniques such as bag-of-words models [10][14][15] or hashing systems, typically with inverted indexing system to be scalable.

Coming with the ever growing computational power of mobile devices, recent works [1][2][3][4][5] have proposed to directly extract visual descriptors on the mobile devices for the subsequent low cost transmission. Although there are previous works in compact local descriptors such as SURF [8], GLOH [30], and PCA-SIFT [9], their compression rates and transmission efficiencies are not sufficient in the context of low bit rate wireless transmission. Consequently, recent works have focused on more compact visual content descriptions [1][2][3][4] that are specially designed for the mobile visual search scenarios.

A representative group of works come from compressing the local visual descriptors [2][3][5][31]. For instance, Chandrasekhar *et al.* proposed a Compressed Histogram of Gradients (CHoG) [2] for compressive local feature description in the mobile device, which adopts both Huffman Tree and Gagie Tree coding to compress each local feature into approximate 50 bits. The same authors in [3] proposed to compress the SIFT descriptor with Karhunen-Loeve Transform for mobile visual search, resulting into approximately 2 bits per SIFT dimension (128 in total). Tsai *et al.* [12] proposed to transmit the spatial layouts of descriptors. While local feature detectors typically extract $\sim 1,000$ points per image, the overall transmission is about 8KB, less than sending the query photo (typically over 20KB).

The recent work in [1] stepped forward by sending the bag-of-features histogram instead of the original local feature descriptors with position difference encoding of non-zero bins, which gained much higher compression rates with only slight loss in discriminability. The work in [1] reported an approximate 2 KB code per image for a vocabulary with 1M words, much less than directly sending the compact local descriptors (more than 5KB to the best of our knowledge). A recent review can be found at [5].

6. CONCLUSIONS AND FUTURE WORKS

In this paper, we present a low bit rate mobile visual search framework towards zero-latency query delivery even using a relatively slow wireless link. Our main idea is to directly extract a compact visual descriptor on the mobile end, with possible supervision from contextual cues such as GPS, barcode, or RFID. To this end, state-of-the-art visual descriptors such as BoW histograms have a relatively high dimensionality although effective. We propose to learn a multiple-channel coding scheme to further compress these high-dimensional signatures efficiently at the mobile end: First, we learn a channel division to subdivide the database images with possible contextual supervision; Second, in each channel, we learn its respective compression function based on a supervised robust PCA, which separates the noisy visual contents in compression, reduces the storage cost at mobile end, and preserves the ranking capability of the original high-dimensional BoW histogram. We evaluate MCVD on a million-scale mobile landmark search system and a mobile product and CD/book cover search system, with applications on Apple and Android cell phones. We report superior performances over the state-of-the-arts [1][2][10][11].

We envision the promising usage of compact visual descriptors in the ongoing industry standardizations and state-of-the-art research efforts, which possibly starts more practical scenarios in real world low bit rate applications. The proposed descriptor as well as the zero-latency search scheme would be at the very beginning of a significant and potentially huge activity, which is towards challenging but exciting research, development, and standardization of mobile search as well as mobile reality augmentation applications.

7. ACKNOWLEDGEMENT

This work was supported by National Basic Research Program of China (2009CB320902), in part by Chinese National Nature Science Foundation (60902057, 61071180) and CADAL Project Program. Correspondence should be addressed to lingyu@pku.edu.cn.

8. REFERENCES

[1] D. Chen, S. Tsai, V. Chandrasekhar, G. Takacs, J. Singh, and B. Girod. Tree histogram coding for mobile image matching. In *DCC*, 2009.

[2] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod. Chog: Compressed histogram of gradients a low bit-rate feature descriptor. In *CVPR*, 2009.

[3] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, J. Singh, and B. Girod. Transform coding of image feature descriptors. In *VCIP*, 2009.

[4] M. Makar, C. Chang, D. Chen, S. Tsai, and B. Girod. Compression of image patches for local feature extraction. In *ICASSP*, 2009.

[5] B. Girod, V. Chandrasekhar, D. Chen, N.-M. Cheung, R. Grzeszczuk, Y. Reznik, G. Takacs, S. Tsai, and R. Vedantham. Mobile visual search. In *IEEE Signal Processing Magazine*, 2011.

[6] Y. Reznik. Compact descriptors for visual search. In *MPEG N11529*, 2010.

[7] D. Lowe. Distinctive image features from scale invariant keypoints. In *IJCV*, 2004.

[8] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *ECCV*, 2006.

[9] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *CVPR*, 2004.

[10] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[11] H. Jegou, M. Douze, C. Schmid, and P. Perez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010.

[12] S. Tsai, D. Chen, G. Takacs, V. Chandrasekhar, J. Singh, and B. Girod. Location coding for mobile image retrieval. In *MobileMedia*, 2010.

[13] D. Chen, S. Tsai, V. Chandrasekhar, *et al.* Inverted index compression for scalable image matching. In *DCC*, 2010.

[14] A. Irschara, C. Zach, J. Frahm, and H. Bischof. From structure-from-motion point clouds to fast location recognition. In *CVPR*, 2009.

[15] G. Schindler and M. Brown. City-scale location recognition. In *CVPR*, 2007.

[16] G. Salton and C. Buckley. Term-weighting approaches in text retrieval. In *Inform. Proc. and Manage.*, 1988.

[17] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *ICCV*, 2003.

[18] C. Stauffer and W. Grimson. Learning patterns of activity using real-time tracking. In *PAMI*, 2000.

[19] J. Wright, A. Ganesh, S. Rao, Y. Peng, and Y. Ma.

[20] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. In *SIAM Journal on Imaging Science*, 2009.

[21] R. Ji, L.-Y. Duan, T. Huang, H. Yao, and W. Gao. Pukbench: A contextual rich benchmark for mobile visual search. In *95th MPEG Meeting*. CDVS AD HOC Group Input Contribution, 2010.

[22] V. Chandrasekhar, D. Chen, S. Tsai, N.-M. Cheung, H. Chen, G. Takacs, Y. Reznik, R. Vedantham, R. Grzeszczuk, J. Bach, and B. Girod. The stanford mobile visual search dataset. In *ACM Multimedia Systems Conference*, 2011.

[23] W. Zhang and J. Kosecka. Image based localization in urban environments. In *3DVT*, 2006.

[24] J.-A. Lee, K.-C. Yow, and A. Sluzek. Image based information guide on mobile devices. In *Advances in Visual Computing*, 2008.

[25] J. Philipin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabulary and fast spatial matching. In *CVPR*, 2007.

[26] D. Crandall, L. Backstrom, and J. Kleinberg. Mapping the world's photos. In *WWW*, 2009.

[27] J. Hays and A. Efros. Img2gps: estimating geographic information from a single image. In *CVPR*, 2008.

[28] Y. Zheng, M. Zhao, Y. Song, and H. Adam. Tour the world: building a web-scale landmark recognition engine. In *CVPR*, 2009.

[29] X. Li, C. Wu, C. Zach, S. Lazebnik, and J.-M. Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *ECCV*, 2008.

[30] K. Mikolajczyk and C. Schmid. Performance evaluation of local descriptors. In *PAMI*, 2005.

[31] V. Chandrasekhar, D. Chen, A. Lin, G. Takacs, S. Tsai, N. Cheung, Y. Reznik, R. Grzeszczuk, and B. Girod. Comparison of local feature descriptors for mobile visual search. In *ICIP*, 2010.