Exploring Consistent Preferences: Discrete Hashing with Pair-Exemplar for Scalable Landmark Search

Lei Zhu[†], Zi Huang[†], Xiaojun Chang[‡], Jingkuan Song[♯], Heng Tao Shen[♯]

[†]The University of Queensland, Brisbane, Australia
[‡]Carnegie Mellon University, Pittsburgh, USA
[♯]University of Electronic Science and Technology of China, Chengdu, China
leizhu0608@gmail.com,huang@itee.uq.edu.au,{cxj273,jingkuan.song}@gmail.com,shenhengtao@hotmail.com

ABSTRACT

Content-based visual landmark search (CBVLS) enjoys great importance in many practical applications. In this paper, we propose a novel discrete hashing with pair-exemplar (DHPE) to support scalable and efficient large-scale CBVLS. Our approach mainly solves two essential problems in scalable landmark hashing: 1) Intra-landmark visual diversity, and 2) Discrete optimization of hashing codes. Motivated by the characteristic of landmark, we explore the consistent preferences of tourists on landmark as pair-exemplars for scalable discrete hashing learning. In this paper, a pair-exemplar is comprised of a canonical view and the corresponding representative tags. Canonical view captures the key visual component of landmarks, and representative tags potentially involve landmark-specific semantics that can cope with the visual variations of intra-landmark. Based on pair-exemplars, a unified hashing learning framework is formulated to combine visual preserving with exemplar graph and the semantic guidance from representative tags. Further, to guarantee direct semantic transfer for hashing codes and remove information redundancy, we design a novel optimization method based on augmented Lagrange multiplier to explicitly deal with the discrete constraint, the bit-uncorrelated constraint and balance constraint. The whole learning process has linear computation complexity and enjoys desirable scalability. Experiments demonstrate the superior performance of DHPE compared with state-of-the-art methods.

CCS CONCEPTS

• **Information systems applications** → Multimedia information systems;

KEYWORDS

Landmark Search; discrete hashing; pair-exemplar

MM'17, October 23–27, 2017, Mountain View, CA, USA.
 ISBN 978-1-4503-4906-2/17/10…$15.00
DOI: https://doi.org/10.1145/3123266.3123301

Figure 1: Example of pair-exemplars of two landmarks *Rotunda of Mosta* and *St. Peter's Square*.

1 INTRODUCTION

With the prevalence of social multimedia and mobile devices, large quantities of user-generated landmark images are recorded and archived on social websites. Developing effective indexing methods to facilitate large-scale content-based visual landmark search (CBVLS) [44] enjoys great importance in real practice.

However, most existing techniques on indexing landmark images are specially designed for compressing particular visual-words based features [3, 7, 10, 41]. They cannot be directly applied to general landmark representations. Hashing [19, 20, 24, 29, 34, 36, 37, 42] is a more general indexing approach which can be promisingly applied to support large-scale CBVLS. With binary transformation by hashing, storage cost of high dimensional image representations can be significantly reduced. Moreover, the online search process on large-amounts of images can be greatly accelerated with efficient Hamming distance computations. Due to these desirable advantages, binary hashing has been receiving considerable attentions from researchers. Various supervised and unsupervised hashing approaches are proposed. Among them, supervised hashing learns binary codes by exploiting explicit semantic labels [24]. These approaches can achieve promising performance with strong semantic supervision. But they require high-quality labels that are usually hard and expensive to obtain in practical CBVLS. It inevitably results in a scalability issue for real-world applications.

Unsupervised hashing is designed without any dependence on semantic labels, mainly relying on visual contents to learn binary codes [35, 45, 46]. It can well support scalable CBVLS. However, directly applying existing unsupervised hashing methods for CBVLS suffers from two major limitations: 1) *Intra-landmark Visual Diversity.* Many landmarks are comprised of multiple attractive regions. It may lead to huge visual variations among their recoded images. Even for the landmark with single historical building or construction, it can be photographed by tourists from various viewpoints,

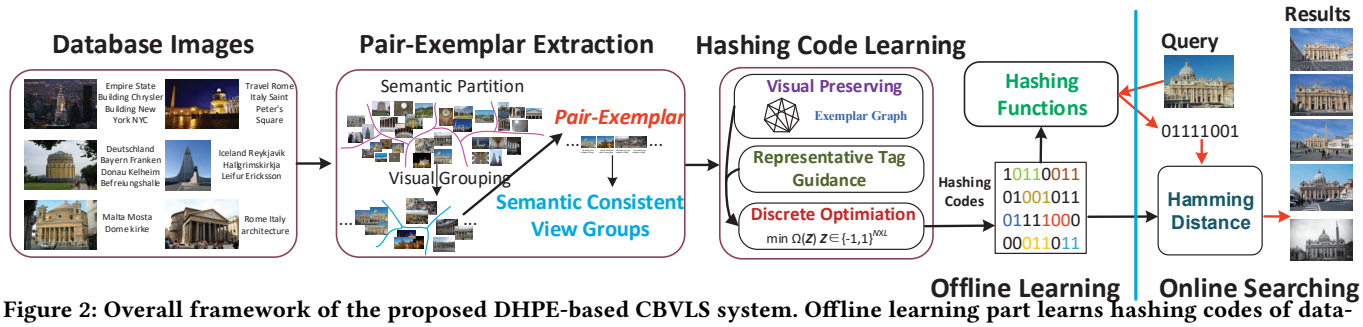


Figure 2: Overall framework of the proposed DHPE-based CBVLS system. Offline learning part learns hashing codes of database images and hashing functions for online query. This part mainly consists of three steps: pair-exemplar extraction, hashing code and function learning. The online searching part generates hashing codes for queries and performs efficient online similarity search in Hamming space.

lighting, and weather conditions, which will generate images with great visual diversity. Due to the intrinsic nature of image representation, unsupervised hashing codes learned on it could suffer from limited semantic representation capability. 2) *Discrete Optimization.* Hashing codes are binary (0 or 1). Hence, hashing learning is essentially a mixed-integer combinatorial optimization problem. To simplify the optimization, most existing approaches apply "relaxing+rounding" optimization framework [8, 11, 12, 18, 33, 39, 48]. This learning scheme may cause significant information loss and thus deteriorate the search performance. Recent literature develops several discrete hashing approaches [13, 19, 21, 24, 26]. Nevertheless, they are specially designed for particular hashing types and formulations. Moreover, many of them [13, 21, 23, 24] only deal with discrete constraint without considering bit-uncorrelated and balance constraint, which are essential for hashing learning.

Landmark images enjoy an important characteristic that general images do not possess: consistent preferences of users/tourists on both visual contents and descriptive semantic terms, which can be observed from the images captured by different tourists and the semantic tags assigned to these images. For a particular landmark, only the most famous and attractive views of the landmark will be photographed by various tourists spontaneously. When tourists share the captured images on social websites (e.g., Flickr), many of them would like to share those attractive views and label them with semantic tags. Besides, the semantic tags they are used to describe the images of a landmark usually concentrate on a small vocabulary, such as landmark names, locations, objects in the landmarks, etc. Hence, we can observe interesting phenomena on the accumulated landmark images shared on social websites: 1) Diverse landmark images of both query and database images visually concentrate on canonical views. 2) User assigned tags on the same landmark will focus on particular terms. These canonical views and user assigned tags reveal the consistent preferences of tourists on landmarks. On the perspective of technique, canonical views intrinsically characterize the view components of landmarks, and the accumulated user tags involve discriminative landmark-specific semantics that can correlate diverse images. They can be exploited to potentially assist landmark hashing learning and cope with visual variations.

Motivated by the aforementioned considerations, in this paper, we propose a novel hashing learning framework, discrete hashing with pair-exemplar (DHPE), to facilitate scalable CBVLS. DHPE extracts the aforementioned consistent preferences of different tourists on landmarks as pair-exemplars, and exploits them further to semantically assist visual hashing process and cope with visual variations of diverse landmark images. In this paper, a pair-exemplar is comprised of a canonical view and the corresponding representative tags. Examples are shown in Figure 1. In particular, canonical view captures the key visual characteristics of landmark, and representative tags are comprised of the accumulated landmark-specific textual tags posted by different tourists. Specifically, DHPE works with two stages: First, pair-exemplars are discovered from loosely organized landmark images with an efficient two-layer clustering. Then, hashing code learning is performed in a unified framework. An exemplar graph is constructed to efficiently preserve view structures of database images by determining canonical views in pair-exemplars as anchors [18]. Simultaneously, representative tags in pair-exemplars are exploited to semantically guide the visual hashing code learning process. Moreover, to guarantee direct semantic transfer to hashing codes and remove information redundancy, we propose an effective discrete optimization method based on augmented Lagrange multiplier (ALM) [1] that directly solves the discrete hashing codes. The basic framework of DHPE-based CBVLS system is illustrated in Figure 2. The main contributions of this paper are:

1. We extract consistent preferences of different tourists on landmarks as pair-exemplars for scalable hashing. Pair-exemplars pack canonical views and the representative tags. Canonical views characterize the key view characteristics of a landmark and reduce computation complexity of visual preserving, the representative tags involve discriminative terms that semantically correlate diverse views.
2. Based on the pair-exemplars, we develop a unified hashing framework to combine a visual preserving part with visual exemplar graph and a semantic guidance part from representative tags. These two parts interact with each other, so that the learned hashing codes are embedded with proper landmark-specific semantics and thus can cope with the visual variations
3. To guarantee direct semantic transfer for hashing codes and avoid information quantization loss, DHPE not only explicitly deals with the discrete constraint, but also considers the bit-uncorrelated constraint and balance constraint. A discrete optimization approach based on ALM is developed to effectively learn hashing codes. The whole learning process enjoys linear computation complexity and desirable scalability.

4. Comparative experimental results demonstrate the superiority of DHPE. The effects of pair-exemplar and discrete optimization of hashing methods are also validated to demonstrate the advantages of the proposed approach.

2 RELATED WORK

Efficient Landmark Search. Many approaches are developed to speedup the landmark search process. Most of them focus on compressing specific visual-words based representations into compact descriptor. Ji *et al.* [10] present a location discriminative vocabulary coding to compress bag-of-visual-words with location awareness. Duan *et al.* [7] explore multiple information sources to extract compact landmark image descriptor. Chen *et al.* [3] develop a soft bag-of-visual phrase to learn category-dependent visual phrases, by capturing co-occurrence features of neighbouring visual-words. Zhou *et al.* [41] propose scalable cascaded hashing to achieve codebook-free large-scale landmark search. These aforementioned methods are specially designed for compressing visual-words based features. Hence, they cannot be directly applied to general image features. Moreover, they are designed on low-level features with limited semantic discriminative capability. This disadvantage further limits its performance.

Hashing. Locality sensitive hashing (LSH) [22] is a data independent hashing approach. It generates hashing codes via random projection. As its learning process is performed without considering any image information, it requires more hashing bits to achieve a satisfactory performance, resulting in significant storage cost.

To enrich the hashing codes with semantics, various supervised hashing approaches are proposed for image indexing. Supervised hashing learns hashing codes by exploiting explicit semantic labels. These approaches can achieve better performance than unsupervised hashing methods. However, supervised hashing requires semantic labels that are hard and expensive to obtain in practical CBVLS. Unsupervised hashing generates binary codes without any semantic labels. Anchor graph hashing (AGH) [18], iterative quantization (ITQ) [9], bilinear projections (BP) [8], circulant binary embedding (CBE) [39], density sensitive hashing (DSH) [12], sparse embedding and least variance encoding (SELVE) [48], sparse projection (SP) [33], and scalable graph hashing (SGH) [11] are typical examples. AGH extends spectral hashing [30] by approximating the image relations with a low-rank matrix. ITQ reduces the quantization loss brought by dimension reduction based binary embedding. BP, CBE, and SP speedup the hashing projections for high-dimensional data. DSH extends LSH and learns projective hashing functions that best match the distribution of the data. SELVE embeds samples into sparse vector and learns least variance encoding model to generate binary hashing codes. SGH applies feature transformation to solve large-scale graph hashing. Due to the semantic gap between image features and high-level semantics, hashing codes learned by these approaches still suffer from significant semantic shortage.

Discrete Optimization. To cope with the discrete optimization challenges in binary hashing, a few approaches have been proposed. Discrete graph hashing (DGH) [19] reformulates the graph hashing with a discrete optimization framework and solves the problem within a tractable alternating maximization. Supervised discrete hashing (SDH) [24] learns discrete hashing codes with a supervised learning. A cyclic coordinate descent algorithm is applied to calculate discrete hashing codes in a closed form. Coordinate discrete hashing (CDH) [21] is designed for cross-modal hashing, and the discrete optimization proceeds in a block coordinate descent manner. Column sampling based discrete supervised hashing (COSDISH) is proposed in [13] to learn discrete hashing codes from semantic information by column sampling. Discrete proximal linearized minimization (DPLM) is presented in [25] to reformulate the hashing learning as minimizing the sum of a smooth loss term with a nonsmooth indicator function. Kernel-based supervised discrete hashing (KSDH) [26] solves discrete hashing codes via asymmetric relaxation strategy. These approaches can achieve good performance for particular hashing types and formulations, however, they cannot be easily and directly generalized to our problem. In addition, many discrete hashing approaches [13, 21, 24] only deal with discrete constraint without considering bit-uncorrelated and balance constraint, which are important for hashing learning.

3 THE PROPOSED APPROACH

3.1 Problem Definition

The main objective of DHPE is to learn $Z = [z_1, z_2, ..., z_N] \in \mathbb{R}^{l \times N}$, where $z_n = [z_{1n}, z_{2n}, ..., z_{ln}]^\top \in \mathbb{R}^{l \times 1}$ are the hashing codes of the n_{th} image, l is hashing code length, N is the number of database images. To generate hashing codes for query images that are out of the database, DHPE learns a group of hashing functions $H = \{h_1, h_2, .., h_l\}$. Each of them defines a mapping: $\mathbb{R}^{d_x} \mapsto \{0, 1\}$, d_x denotes the feature dimension of visual representation.

3.2 Pair-Exemplar Extraction

As illustrated in Section 1, due to the characteristics of landmarks, the recorded diverse landmark images visually concentrate on canonical views. Simultaneously, the tags associated with images of the same landmark are usually constraint to a small set of terms. These canonical views and tags promisingly reflect the consistent preferences of different tourists on landmarks. In this paper, we pack a canonical view and its representative tags into a pair-exemplar, which is leveraged to characterize the key semantic components of landmarks. In particular, we propose an efficient two-layer clustering method by jointly analysing visual and textual distributions of landmarks. As indicated in [4, 5], textual features extracted from the associated tags enjoy better discriminative capability for landmarks. Therefore, in the first layer, images are semantically partitioned into C_y coarse groups with clustering on textual features. These C_y coarse groups are semantically similar, but may vary on visual contents. In the second layer, for each semantic group, images are further grouped into C_x visually consistent view groups with clustering analysis on their visual representations. In this paper, clustering in two layers are efficiently implemented with k-means. Note that this part is flexible and can be substituted with other effective clustering methods [38].

Ideally, with the two-layer clustering, we obtain K semantic consistent view groups. For each view group, we compute the visual and textual distances between images. The image that has the smallest combined distance[1] to all the remaining images in the same view group are selected as the canonical view. Formally, we

[1] Average sum of visual and textual distances.

denote the canonical view selected from the k_{th} view group as CV_k. Meanwhile, we compute the occurrence frequency of each tag associated with images in this view group. In particular, the tags with the occurrence frequencies that are more than half of group size are determined as representative tags. They are selected to comprise the k_{th} representative tag set RT_k. Both the selected CV_k and RT_k jointly comprise the pair-exemplar PE_k of the k_{th} view group $PE_k = \{CV_k, RT_k\}$. Then, the pair-exemplars for all landmarks are $\{PE_k\}_{k=1}^{K}$. Typical examples of them are presented in Figure 1.

Landmark database contains redundant images concentrating on canonical views, which brings additional computation burden on subsequent hashing learning. In this paper, we restructure the database with the discovered pair-exemplars. For the k_{th} view group, we select M images that are most semantically similar to CV_k. These images comprise a new database images that can be visually characterized as $X = [x_1, ..., x_{KM}] \in \mathbb{R}^{d_x \times KM}$ and textually represented as $Y = [y_1, ..., y_{KM}] \in \mathbb{R}^{d_y \times KM}$, d_x and d_y denote corresponding feature dimensions. Representative tags of pair-exemplars are leveraged to adjust the textual representation of images that are concentrated on their corresponding canonical view. It is to avoid tag incompleteness and noise of landmark image that usually occur in social media [4]. Formally, the data dimension in y_i is set to 1 if the corresponding tag belongs to $RT_{i/M}$, and 0 vice versa. The visual representation of canonical views in pair-exemplars are $PE = [pe_1, ..., pe_K] \in \mathbb{R}^{d_x \times K}$. In the following, we still use N to denote KM for presentation convenience.

3.3 Hashing Code Learning

Hashing codes are learned based on the extracted pair-exemplars. Its formulation is comprised of two main parts: 1) Visual preserving, which ensures the similarities of hashing codes to be consistent with the original view structures. 2) Representative tag guidance, which aims at enriching the semantics of visual hashing codes with the assistance of representative tags in pair-exemplars.

Visual Preserving. CBVLS retrieves similar images for query [44]. Hence, one of the essential design principles of hashing for CBVLS is visual preserving, i.e., similar landmark images are mapped to binary codes with short Hamming distances. In this paper, we construct an exemplar graph to preserve image relations by considering canonical views in pair-exemplars as anchors [18] and images as graph vertices. We seek to minimize the weighted Hamming distance of hashing codes. Visual preserving is formulated as

$$\min_{\{z_i\}_{i=1}^{N} \in [-1,1]^{l \times 1}} \sum_{i,j=1}^{N} S_{ij} ||z_i - z_j||^2 \Rightarrow \min_{Z \in [-1,1]^{l \times N}} Tr(Z \Omega Z^{\top})$$

where $\Omega = D - S$ is the Laplacian matrix of exemplar graph, S characterizes the affinity similarities of images, $D = S\mathbf{1}$, $\mathbf{1}$ is column vector with all ones, and $Tr(\cdot)$ is the trace operator. The design principle of the above formula is to impose a heavy penalty if two similar images are projected far apart. Note that, explicitly computing S leads to $O(N^2)$ time complexity [15], which is unacceptable in real world application. With pair-exemplars, the affinity matrix S can be efficiently computed as $S = V \Lambda V^{\top}$ [17], where $\Lambda = \mathtt{diag}(V^{\top}\mathbf{1})$. $V = [v(x_1), .., v(x_N)]^{\top}$, $v(x)$ is data-to-exemplar mapping

$$v(x) = \frac{[\delta_1 \exp(\frac{-||x - pe_1||_2^2}{\sigma}), \ldots, \delta_K \exp(\frac{-||x - pe_K||_2^2}{\sigma})]^{\top}}{\sum_{k=1}^{K} \delta_k \exp(\frac{-||x - pe_k||_2^2}{\sigma})}$$

δ_k is set to 1 if pe_k belongs to the s^2 closest exemplars of x, and 0 vice versa, $\sigma > 0$ is the bandwidth parameter, which is calculated as the average distances between images and exemplars. As S is a doubly stochastic matrix that has unit row and column sums, so we obtain the resulting graph Laplacian as $\Omega = I - V \Lambda V^{\top}$. As shown in hashing learning, keeping this low-rank form decomposition will avoid explicit $O(N^2)$ matrix computation and enjoy $O(N)$ computational complexity.

Representative Tag Guidance. Hashing codes relying on pure visual contents suffer from limited semantics, which will deteriorate the search performance. Cross-modal hashing methods [14, 28, 40, 47] can exploit contextual tags to enrich the semantics of visual hashing codes. However, the main objective of cross-modal hashing is to discover the shared space for heterogenous search across different types of media. In this case, visual and textual representations are generally treated equally. The valuable information originally owned by visual features may not be comprehensively preserved as result of mandatory correlation. Motivated by the strong supervision capability of using assigned tags on landmark images [4, 5], we exploit textual representation Y obtained based on representative tags in pair-exemplars to guide the hashing learning. Specifically, we minimize the noisy linear classification errors based on hashing codes. We argue that the learned binary codes are expected to be optimal for weak classification. Since representative tags involve noise, we use $l_{2,1}$ norm [16] to measure the errors. In our design, it will adaptively and automatically select the most informative tags for semantic guidance. Formally, the semantic guidance is formulated as

$$\min_{Z \in [-1,1]^{l \times N}, U} ||Y - UZ||_{2,1}$$

where U is linear classification mapping matrix.

Overall Formulation. After comprehensively considering visual preserving, representative tag guidance, and hashing constraints [19], we obtain the overall formulation of DHPE.

$$\begin{aligned} &\min_{Z,U} ||Y - UZ||_{2,1} + \alpha Tr(Z(I - V\Lambda V^{\top})Z^{\top}) \\ &s.t.\ Z \in [-1, 1]^{l \times N},\ ZZ^{\top} = NI,\ Z\mathbf{1} = 0 \end{aligned} \tag{1}$$

where $\alpha > 0$ balances the regularization terms. $Z \in [-1, 1]^{l \times N}$ is discrete constraint on hashing codes, $ZZ^{\top} = NI$ is bit-uncorrelated constraint which guarantees that the learned hashing bits to be uncorrelated and removes information redundancy, $Z\mathbf{1} = 0$ is balance constraint which forces each bit to have equal chance to occur.

Note that solving Eq.(1) is essentially a challenging combinatorial optimization problem due to the three constraints. Most existing hashing approaches apply "relaxing+rounding" optimization framework [29]. Basically, they first relax discrete constraint to calculate continues values, and then binarize them to hashing codes via rounding. This two-step learning method can simplify the solving process, but it may cause significant information loss [19, 24]. In recent literature, several discrete hashing solutions are proposed.

[2]The optimal s is 5 in this paper.

However, they are developed for particular hashing types and formulations. For example, graph hashing [19], supervised hashing [13, 24], cross-modal hashing [21]. Hence, their designed learning strategies cannot be directly applied to solve our problem.

3.4 Discrete Optimization

In this paper, we propose an effective optimization algorithm based on augmented Lagrange multiplier (ALM) [1] to calculate the discrete solution within one step. Our idea is adding auxiliary variables A, B to separate constraints, and transforming the objective function to an equivalent one that can be solved more easily. Formally, we set $A = Y - UZ, B = Z$. Eq.(1) is reformulated as

$$\begin{aligned} &\min_{Z,U,A,B} \ ||A||_{2,1} + \frac{\beta}{2}||Y - UZ - A + \frac{E_y}{\beta}||_F^2) + \\ &\alpha Tr(Z(I - V\Lambda V^\top)B^\top) + \frac{\mu}{2}||Z - B + \frac{E_z}{\mu}||_F^2 \\ &s.t. \quad B \in [-1, 1]^{l\times N}, ZZ^\top = NI, Z\mathbf{1} = 0 \end{aligned} \tag{2}$$

where E_y, E_z measure the difference between the target and auxiliary variables, $\alpha, \beta, \mu > 0$ adjust the balance between terms. We adopt alternate optimization to iteratively solve the above equation. Specifically, we optimize the objective function with respective to one variable while fixing other variables.

Update A. The optimization formula is

$$\min_A \ ||A||_{2,1} + \frac{\beta}{2}||Y - UZ - A + \frac{E_y}{\beta}||_F^2 \tag{3}$$

Let us define $A = [A_1; A_i; A_{d_y}], T = Y - UZ + \frac{E_y}{\beta}, T = [T_1; T_i; T_{d_y}]$. The above equation can be rewritten as a sum form $\min_A \sum_{i=1}^{d_y} \frac{1}{\beta}||A_i||_2 + \frac{1}{2}||A_i - T_i||_F^2$. By taking the derivative of $||A_i||_2$ with respect to A_i, we have

$$\frac{\partial ||A_i||_2}{A_i} = \begin{cases} r & A_i = 0 \\ \frac{A_i}{\sqrt{A_i A_i^\top}} & A_i \neq 0 \end{cases} \tag{4}$$

where r is sub-gradient and $||r||_2 \leq 1$.

Therefore, by taking the derivative of $\frac{1}{\beta}||A_i||_2 + \frac{1}{2}||A_i - T_i||_F^2$ with respect to A_i and setting it to 0, we can obtain that 1) If $A_i = 0$, we get $-T_i + \frac{1}{\beta}r = 0 \Rightarrow \frac{1}{\beta} \geq ||T_i||_2$. 2) If $A_i \neq 0$, we get $A_i - T_i + \frac{1}{\beta}\frac{A_i}{\sqrt{A_i A_i^\top}} = 0 \Rightarrow A_i = \frac{||A_i||_2}{||A_i||_2 + \frac{1}{\beta}}T_i, A_i = (1 - \frac{\frac{1}{\beta}}{||T_i||_2})T_i$. As $\frac{||A_i||_2}{||A_i||_2 + \frac{1}{\beta}} > 0$, A_i and T_i have the same sign. Thus, $1 - \frac{\frac{1}{\beta}}{||T_i||_2} > 0 \Rightarrow \frac{1}{\beta} < ||T_i||_2$.

The i_{th} row of the optimal solution A is calculated as

$$A(i,:) = \begin{cases} \frac{||T_i||_2 - \frac{1}{\beta}}{||T_i||_2}T_i & ||T_i||_2 > \frac{1}{\beta} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Update U. Similarly, the optimization formula for U is

$$\min_U ||Y - UZ - A + \frac{E_y}{\beta}||_F^2 \tag{6}$$

By calculating the derivative of the objective function with respective to U, and setting it to 0, we can obtain that

$$UZ = Y - A + \frac{E_y}{\beta} \tag{7}$$

Since $ZZ^\top = NI$, we can further derive that

$$U = \frac{1}{N}(Y - A + \frac{E_y}{\beta})Z^\top \tag{8}$$

Update B. The optimization formula for B is

$$\min_{B\in[-1,1]^{l\times N}} \alpha Tr(Z(I - V\Lambda V^\top)B^\top) + \frac{\mu}{2}||Z - B + \frac{E_z}{\mu}||_F^2 \tag{9}$$

The objective function in Eq.(9) can be simplified as

$$\min_{B\in[-1,1]^{l\times N}} ||B - (Z + \frac{E_z}{\mu} - \frac{\alpha}{\mu}Z(I - V\Lambda V^\top))||_F^2 \tag{10}$$

The discrete solution of B can be directly represented as

$$B = \mathrm{Sgn}(Z + \frac{E_z}{\mu} - \frac{\alpha}{\mu}Z + \frac{\alpha}{\mu}ZV\Lambda V^\top) \tag{11}$$

where $\mathrm{Sgn}(\cdot)$ is signum function.

Update Z. The optimization formula for Z is

$$\begin{aligned} \min_Z \ &\frac{\beta}{2}||Y - UZ - A + \frac{E_y}{\beta}||_F^2 + \alpha Tr(Z(I - V\Lambda V^\top)B^\top) \\ &+ \frac{\mu}{2}||Z - B + \frac{E_z}{\mu}||_F^2 \\ &s.t. \ ZZ^\top = NI, Z\mathbf{1} = 0 \end{aligned} \tag{12}$$

The objective function in Eq.(12) can be transformed as

$$= \min_{ZZ^\top = NI, Z\mathbf{1}=0} -Tr(Z^\top C) \tag{13}$$

where $C = B - \frac{E_z}{\mu} - \frac{\alpha}{\mu}B + \frac{\alpha}{\mu}BV\Lambda V^\top + \frac{\beta}{\mu}U^\top(Y - A + \frac{E_y}{\beta})$. Eq.(12) is equivalent to the following maximization problem

$$\max_{ZZ^\top = NI, Z\mathbf{1}=0} Tr(Z^\top C) \tag{14}$$

Mathematically, with singular value decomposition (SVD), C can be decomposed as $C = P\Delta Q^\top$, where the columns of P and Q are left-singular vectors and right-singular vectors of C respectively, Δ is rectangular diagonal matrix and its diagonal entries are singular values of C. Then, the optimizing for Z becomes $\max_Z \ Tr(Z^\top P\Delta Q^\top) \Leftrightarrow \max_Z \ Tr(\Delta Q^\top Z^\top P)$. $\Delta \geq 0$ as Δ is calculated by SVD. On other hand, we can easily derive that $Q^\top Z^\top PP^\top ZQ = NI$. Therefore, according to the **Theorem 3.1**, the optimal Z can only be obtained when $Q^\top Z^\top P = \mathrm{diag}(\sqrt{N})$. Hence, the solution of Z is

$$Z = \sqrt{N}PQ^\top \tag{15}$$

Theorem 3.1. *Given any matrix G which meets $GG^\top = NI$ and diagonal matrix $\Delta \geq 0$, the solution of $\max_G Tr(\Delta G)$ is $\mathrm{diag}(\sqrt{N})$.*

Proof. Let us assume λ_{ii} and g_{ii} are the i_{th} diagonal entry of Δ and G respectively, $Tr(\Delta G) = \sum_i \lambda_{ii}g_{ii}$. Since $GG^\top = NI$, $g_{ii} \leq \sqrt{N}$. $Tr(\Delta G) = \sum_i \lambda_{ii}g_{ii} \leq \sqrt{N}\sum_i \lambda_{ii}$. The equality holds only when $g_{ii} = \sqrt{N}, g_{ij} = 0, \forall i, j$. $Tr(\Delta G)$ achieves its maximum when $G = \mathrm{diag}(\sqrt{N})$. □

Moreover, in order to satisfy the balance constraint $Z\mathbf{1} = 0$, we apply Gram-Schmidt process as [19] and construct matrices $\hat{P}$ and $\hat{Q}$, so that $\hat{P}^\top\hat{P} = I_{L-R}$, $[P, 1]^\top\hat{P} = 0$, $\hat{Q}^\top\hat{Q} = I_{L-R}$, $Q\hat{Q}^\top = 0$, R is the rank of C. The close form solution for Z is

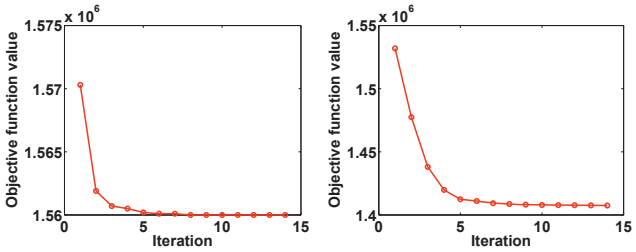
$$Z = \sqrt{N}[P, \hat{P}][Q, \hat{Q}]^\top \tag{16}$$

Update E_y, E_z, μ. The update rules are ($\rho > 1$ is learning rate that controls the convergence.)

$$E_y = E_y + \mu(Y - UZA_y), E_z = E_z + \mu(Z - B), \mu = \rho\mu \tag{17}$$

Table 1: mAP of all approaches on two datasets. The best result in each column is marked with bold.

Methods	*MMParis*					*Landmark-101*				
	16	32	64	128	256	16	32	64	128	256
AGH	0.2420	0.3880	0.4491	0.4890	0.4815	0.2535	0.4008	0.4592	0.4903	0.5049
ITQ	0.2656	0.4201	0.5415	0.6097	0.6681	0.3165	0.4586	0.5479	0.6146	0.6553
BP	0.1275	0.2658	0.3682	0.4901	0.5786	0.1495	0.2780	0.3931	0.4955	0.5718
CBE	0.1697	0.3077	0.4463	0.5534	0.6133	0.1741	0.3033	0.4307	0.5410	0.6067
DSH	0.1617	0.2691	0.3784	0.4670	0.5120	0.1905	0.3121	0.3871	0.4641	0.5062
SELVE	0.2760	0.3595	0.4355	0.4536	0.4450	0.2738	0. 3942	0.4629	0.4769	0.4853
SP	0.2534	0.4026	0.5113	0.5862	0.6467	0.2969	0.4175	0.5198	0.5882	0.6294
SGH	0.2488	0.4056	0.5231	0.6158	0.6600	0.2703	0.4068	0.5162	0.5925	0.6341
DPLM	0.2689	0.3638	0.4146	0.4480	0.4718	0.2734	0.3718	0.4233	0.4439	0.4669
DHPE	**0.3450**	**0.4731**	**0.5628**	**0.6362**	**0.6837**	**0.4063**	**0.5409**	**0.6438**	**0.7033**	**0.7309**



(a) *MMParis* (b) *Landmark-101*

Figure 3: Objective function value variations with iterations.

3.5 Hashing Function Learning

In this paper, we leverage linear projection to learn hashing functions for its high online efficiency. The formulation is $\min_W ||Z - W^\top V^\top||_F^2 + \eta||W||_F$, where V is feature transformation of X based on pair-exemplars, $W \in \mathbb{R}^{K\times l}$ denotes the projection matrix. The optimal W can be calculated as $W = \left(VV^\top + \eta I\right)^{-1} V^\top Z^\top$. The final hashing functions can be constructed as $H(x) = \frac{\text{sgn}(W^\top v(x))+1}{2}$.

3.6 Algorithm Analysis

Convergency Analysis. The updating of variables will decrease the objective function value. As indicated by ALM optimization theory [1, 2], the iterations will make the optimization process converged. We also conducted empirical experiment on the convergence property using *MMParis* [31, 32] and *Landmark-101* [43]. Figure 3 presents the results. We observe that the objective function value first decreases with the number of iterations and then becomes steady after around 10 iterations. This result demonstrates that the convergence of the proposed method and indicates the efficiency of our algorithm.

Computational Complexity Analysis. In the generation of pair-exemplars, k-means is applied for semantic partition and visual grouping, which takes $O(NC_x d_x)$ and $O(NC_y d_y)$, respectively. Hence, the time complexity of the two-layer clustering is $O(N)$. Pair-exemplars are discovered from $C_x C_y$ clusters and this process cost $O(C_x C_y)$. The main cost in exemplar graph construction is the distance computation between canonical views in pair-exemplars and training images, which costs $O(K^2 M)$. The computational complexity of discrete optimization is $O(\#iter(d_x KM + d_y KM + d_x l + d_y l + lKM))$, where $\#iter$ denotes the number of iterations. Given $KM \gg d_x(d_y) > L$, this process scales linearly with KM. The computation of hashing functions solves a linear system, whose time complexity is $O(KM)$. Calculating hashing codes of database images costs $O(KM)$. The overall computation complexity of offline learning is $O(N)$. In online retrieval, generating hashing codes for a query can be completed in $O((d_x + 1)l)$.

4 EXPERIMENTAL CONFIGURATION

Experimental Datasets and Setting. We conduct experiments on two real-world image datasets, *MMParis* [31, 32] and *Landmark-101* [43]. *MMParis* consists of 501,356 geo-tagged images of landmarks in Paris. They are collected from Flickr and Panoramio with geographic bounding box. In this paper, we use ground truth of 79 touristic landmarks covering 94,303 images. *Landmark-101* contains 101 worldwide landmarks involving 57,386 images crawled from Flickr with relevant keyword search. It includes the images photographed for various beauty spots, from various viewpoints, and under various weather conditions. For both datasets, as only visual images are provided, we re-crawled user tags with the Photo ID provided in datasets. Images with no tags are removed. Finally, 40,584 and 38,460 images are remained in *MMParis* and *Landmark-101*, respectively. For image representation, we extract 4096 dimensional feature vector from VGG-19 convolutional networks [27] for each image that contains the activations of the hidden layer immediately before the object classifier. For textual description, we use a 500-dimensional vector space model on tags that have the highest occurrence frequency. In this case, each dimension of feature vector for a photo is a value indicating occurrence frequency of each text tag. For both datasets, 20 query images for each landmark are randomly sampled to comprise testing images. The remaining samples are determined as training images and database images to be retrieved.

Evaluation Metric. In our experimental paper, mean average precision (mAP) [25, 33] is adopted as the evaluation metric. The top 50 images are returned and collected in retrieval. Furthermore, *Precision-Scope* curve is also reported to reflect the retrieval performance variations with respect to the number of retrieved images. For both datasets, as images are labelled into independent categories, they are considered to be relevant only if they belong to the same category.

Evaluation Baselines. The learning of DHPE is independent with explicit semantic labels. Therefore, we compare its performance with several state-of-the-art unsupervised hashing approaches. They include: anchor graph hashing (AGH) [18], iterative quantization (ITQ) [9], bilinear projections (BP) [8], circulant binary embedding (CBE) [39], density sensitive hashing (DSH) [12], sparse

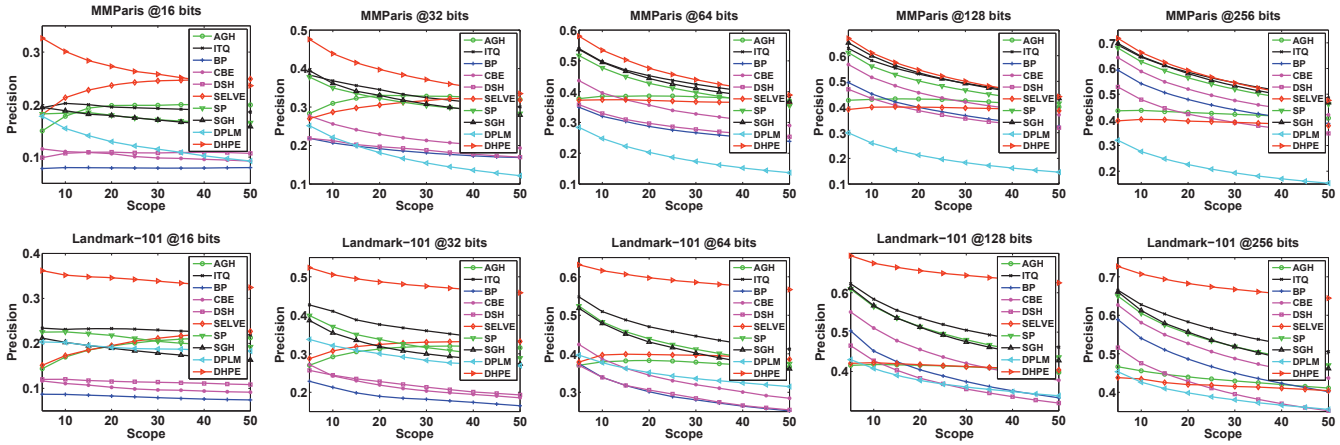


Figure 4: *Precision-Scope* curves on *MMParis* and *Landmark-101*.

Table 2: Effects of pair-exemplar and discrete optimization. DHPE-I denotes the competitor which only considers visual preserving with exemplar graph. DHPE-II denotes the variant of DHPE that removes discrete constraint. It first solves the relaxed codes and then generates hashing codes by mean thresholding. DHPE-III denotes the variant of DHPE that removes bit balance constraint.

Methods	*Landmark-101*				
	16	32	64	128	256
DHPE-I	0.3136	0.4342	0.5368	0.6097	0.6595
DHPE-II	0.2047	0.3952	0.5839	0.6929	0.7306
DHPE-III	0.3923	0.5341	0.6408	0.6977	0.7308
DHPE	**0.4063**	**0.5409**	**0.6438**	**0.7033**	**0.7309**

embedding and least variance encoding (SELVE) [48], sparse projection (SP) [33], scalable graph hashing (SGH) [11], discrete proximal linearized minimization (DPLM) [25][3]. All parameters in compared approaches are adjusted according to the relevant literatures and the best performance is reported in this paper.

Implementation Details. C_x and C_y on both datasets are set to 9 and 1000 respectively. In Eq.(2), there are three parameters: β, α, and μ, which adjust the balance between regularization terms. η in hashing function learning is designed with the same objective. 5-fold cross-validation is adopted to choose these parameters from $\{10^{-4}, 10^{-2}, 1, 10^2, 10^4\}$. The best performance is achieved when $\{\beta = 10, \mu = 10^4, \alpha = 10^{-4}, \eta = 10^{-4}\}$, $\{\beta = 10^4, \mu = 1, \alpha = 10^{-4}, \eta = 10^{-4}\}$ on *MMParis* and *Landmark-101* respectively. In experiments, hashing code length L on all datasets is varied in the range of $\{16, 32, 64, 128, 256\}$ to observe the performance.

5 RESULTS AND DISCUSSIONS

Performance Comparison Results. Table 1 presents the main mAP comparison results. Figure 4 reports *Precision-Scope* curves of all approaches on *Landmark-101* and *MMParis* respectively. These results clearly demonstrate that DHPE consistently outperforms the compared approaches on all datasets and hashing bits. On *Landmark-101*, DHPE outperforms the second best method by 11%. In addition, we find that, even with 64 bits, DHPE can obtain more accurate results than that of competitors on Landmark-101 with 256. This desirable advantage shows that DHPE can significantly improve the discriminative capability of hashing for CBVLS. In addition, we find that performance gap on small code length is more obvious than that on larger code length. This is because: short hashing codes involve less semantics in compared approaches, while for DHPE, more semantics of hashing codes can be compensated from auxiliary pair-exemplars.

[3]We choose the unsupervised graph hashing of DPLM for comparison.

Table 3: Effects of representative tag guidance. Retrieval performance comparison with cross-modal hashing methods.

Methods	*Landmark-101*				
	16	32	64	128	256
CVH	0.3281	0.4792	0.5762	0.6090	0.6194
IMH	0.2855	0.4191	0.5508	0.5913	0.5634
LCMH	0.2744	0.4760	0.5689	0.5818	0.5744
LSSH	0.2866	0.4653	0.5271	0.5459	0.5787
CMFH	0.2740	0.3879	0.4876	0.5665	0.6162
DHPE	**0.4063**	**0.5409**	**0.6438**	**0.7033**	**0.7309**

Table 4: Effects of $l_{2,1}$ norm in Eq.(1).

Methods	*Landmark-101*				
	16	32	64	128	256
No $l_{2,1}$ norm	0.3450	0.4742	0.6108	0.6932	0.7305
DHPE	**0.4063**	**0.5409**	**0.6438**	**0.7033**	**0.7309**

Effects of Pair-Exemplar. In this subsection, we investigate the effects of pair-exemplar. Specifically, we compare the performance of DHPE with the competitor which only considers visual preserving with exemplar graph. We denote it as DHPE-I for presentation convenience. Table 2 presents the main comparison results on *Landmark-101*. It can be clearly observed that DHPE consistently achieves better retrieval performance on all hashing code lengths. The largest absolute performance increase is more than 10%. The reason is that: with pair-exemplar assistance, the view structures of landmarks can be modelled more accurately. And discriminative semantics from representative tags can be effectively transferred to visual hashing codes. Hence, the generated hashing codes have better discriminative capability and search performance.

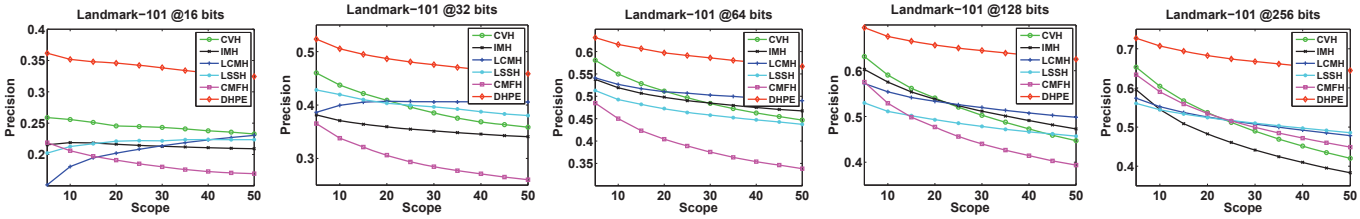
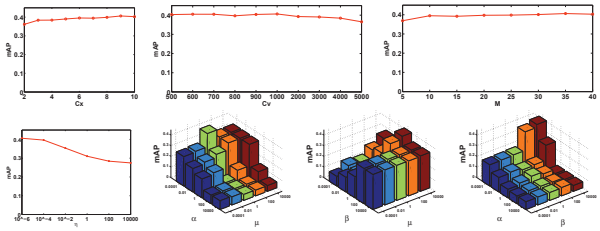


Figure 5: *Precision-Scope* curves of compared cross-modal hashing methods on *Landmark-101*.



(e) β is fixed to 10^4 (f) α is fixed to 10^{-4} (g) μ is fixed to 1

Figure 6: Performance variations of DHPE with parameters on *Landmark-101*. The figure is best viewed with PDF magnification

Effects of Representative Tag Guidance. To validate the effects of representative tag guidance in hashing learning, we compare DHPE with recent cross-modal hashing methods. These compared approaches exploit contextual tags as textual representations for semantic enrichment. They include: cross-view hashing (CVH) [14], inter-media hashing (IMH) [28], linear cross-modal hashing (LCMH) [47], latent semantic sparse hashing (LSSH) [40], and collective matrix factorization hashing (CMFH) [6]. Table 3 presents the main experimental results. Figure 5 demonstrates precision-scope curves. We can find that DHPE outperforms the compared cross-modal hashing methods. The potential reason is that: the main objective of cross-modal hashing is to discover the shared space and thus support heterogenous search across different media. In this case, visual and textual representations are treated equally. The valuable information originally owned by visual features has not been comprehensively preserved due to the mandatory correlation. This may also explain that the performance of cross-modal methods on CBVLS is different from that on cross-modal retrieval task.

Effects of Discrete Optimization. To evaluate the effects of discrete optimization, we compare the performance on *Landmark-101* between DHPE and the solutions which adopt conventional "relaxing+rounding" optimization in many existing hashing approaches. Specifically, we relax the discrete constraint and remove balance constraint in the Eq.(1). For comparison, we denote DHPE-II as the approach that relaxes discrete constraint. In this variant, the relaxed hashing values are solved with ALM, but the final binary hashing codes are generated by mean thresholding. We also compare the performance with the variant approach DHPE-III that removes balance constraint. Table 2 summarizes the performance comparison results. We can clearly observe that DHPE can consistently achieve better performance in all cases. These results validate the effects of direct discrete optimization on avoiding quantization errors that may be brought in non-discrete hashing approaches.

Effects of $l_{2,1}$ Norm on Hashing Performance. $l_{2,1}$ norm is used in Eq.(1) to eliminate noise. In this subsection, we compare the hashing performance of DHPE with the one which is formulated with Frobenius norm. Table 4 demonstrates the main results. It clearly shows that DHPE outperforms the competitor on all code lengths. In addition, we find that the performance gap is larger on smaller hashing code length. This is because: the involved noise in representative tags have more impact on semantic embedding when generating hashing codes with shorter length. In this case, the advantages of employed $l_{2,1}$ norm can be better performed on removing adverse noise and generating hashing codes.

Parameter Experiments. In this subsection, we evaluate the performance variations of DHPE with parameters. C_x, C_y, M are used in Section 3.2 to extract discriminative pair-exemplars for hashing learning. β, α, μ, and η are used in Eq.(2) and hashing function learning to play trade-off between regularization terms and empirical loss. We report the results on *Landmark-101* when hashing code length is 16. Figure 6 illustrates the main experimental results. From this figure, we can clearly find that the best performance can be achieved on certain point of C_x (9), $\alpha(10^{-4})$, $\beta(10^4)$, $\mu(1)$, a certain range of C_y (500∼1000) and M (10∼ 40), $\eta(10^{-6} \sim 10^{-4})$.

6 CONCLUSION

Existing hashing techniques suffer from great intra-landmark visual diversity and discrete optimization in CBVLS. In this paper, we explore the consistent preferences of tourists on landmark as pair-exemplars for scalable landmark hashing. The exemplar graph built on canonical views in pair-exemplars can efficiently capture the view topology of landmark and improve the efficiency of subsequent hashing learning. And representative tags in pair-exemplars involve landmark-specific semantics that can well cope with the visual variations of landmarks. Based on pair-exemplars, a unified hashing learning framework is formulated to combine view preserving and semantic guidance of representative tags. Further, we design a augmented Lagrange multiplier based optimization method to explicitly deal with the discrete constraint, the bit-uncorrelated constraint and balance constraint. It ensures direct semantic transfer for hashing codes and removes information redundancy. Experiments show the superior performance of the proposed approach and validate the advantages of our approach.

7 ACKNOWLEDGEMENTS

This work was supported in part by the ARC under Grant FT130101530 and DP170103954, and National Natural Science Foundation of China under Grant 61632007.

REFERENCES

[1] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.* 3, 1 (2011), 1–122.

[2] Rick Chartrand. 2012. Nonconvex Splitting for Regularized Low-Rank + Sparse Decomposition. *IEEE Trans. Signal Process.* 60, 11 (2012), 5810–5819.

[3] Tao Chen, Kim-Hui Yap, and Dajiang Zhang. 2014. Discriminative Soft Bag-of-Visual Phrase for Mobile Landmark Recognition. *IEEE Trans. Multimedia* 16, 3 (2014), 612–622.

[4] Zhiyong Cheng and Jialie Shen. 2016. On very large scale test collection for landmark image search benchmarking. *Signal Process.* 124 (2016), 13 – 26.

[5] David Crandall, Yunpeng Li, Stefan Lee, and Daniel Huttenlocher. 2016. Recognizing landmarks in large-scale social image collections. In *Visual Analysis and Geolocalization of Large Scale Imagery*, Asaad Hakeem, Richard Szeliski, Mubarak Shah, Luc Van Gool, and Amir Zamir (Eds.). Springer.

[6] G. Ding, Y. Guo, J. Zhou, and Y. Gao. 2016. Large-Scale Cross-Modality Search via Collective Matrix Factorization Hashing. *IEEE Trans. Image Process.* 25, 11 (2016), 5427–5440.

[7] Ling-Yu Duan, Jie Chen, Rongrong Ji, Tiejun Huang, and Wen Gao. 2013. Learning Compact Visual Descriptors for Low Bit Rate Mobile Landmark Search. *AI Magazine* 34, 2 (2013), 67–85.

[8] Y. Gong, S. Kumar, H. A. Rowley, and S. Lazebnik. 2013. Learning Binary Codes for High-Dimensional Data Using Bilinear Projections. In *CVPR*. 484–491.

[9] Yunchao Gong, S. Lazebnik, A. Gordo, and F. Perronnin. 2013. Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-Scale Image Retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 12 (2013), 2916–2929.

[10] Rongrong Ji, Ling-Yu Duan, Jie Chen, Hongxun Yao, Junsong Yuan, Yong Rui, and Wen Gao. 2012. Location Discriminative Vocabulary Coding for Mobile Landmark Search. *Int. J. Comput. Vision* 96, 3 (2012), 290–314.

[11] Qing-Yuan Jiang and Wu-Jun Li. 2015. Scalable Graph Hashing with Feature Transformation.. In *IJCAI*. 2248–2254.

[12] Z. Jin, C. Li, Y. Lin, and D. Cai. 2014. Density Sensitive Hashing. *IEEE Trans. Cybern.* 44, 8 (2014), 1362–1371.

[13] Wang-Cheng Kang, Wu-Jun Li, and Zhi-Hua Zhou. 2016. Column Sampling Based Discrete Supervised Hashing. In *AAAI*. 1230–1236.

[14] Shaishav Kumar and Raghavendra Udupa. 2011. Learning Hash Functions for Cross-View Similarity Search.. In *IJCAI*. 1360–1365.

[15] Jingjing Li, Yue Wu, Jidong Zhao, and Ke Lu. 2016. Low-rank discriminant embedding for multiview learning. *IEEE Trans. Cybern.* (2016).

[16] Jingjing Li, Jidong Zhao, and Ke Lu. 2016. Joint Feature Selection and Structure Preservation for Domain Adaptation.. In *IJCAI*. 1697–1703.

[17] Wei Liu, Junfeng He, and Shih-Fu Chang. 2010. Large Graph Construction for Scalable Semi-Supervised Learning. In *ICML*. 679–686.

[18] Wei Liu, Wang Jun, Sanjiv Kumar, and Shih-Fu Chang. 2011. Hashing with Graphs.. In *ICML*. 1–8.

[19] Wei Liu, Cun Mu, Sanjiv Kumar, and Shih-Fu Chang. 2014. Discrete Graph Hashing. In *NIPS*. 3419–3427.

[20] Yadan Luo, Yang Yang, Fumin Shen, Zi Huang, Pan Zhou, and Heng Tao Shen. 2017. Robust discrete code modeling for supervised hashing. *Pattern Recognit.* (2017). DOI: https://doi.org/10.1016/j.patcog.2017.02.034 doi:10.1016/j.patcog.2017.02.034.

[21] Yadong Mu, Wei Liu, Cheng Deng, Zongting Lv, and Xinbo Gao. 2016. Coordinate Discrete Optimization for Efficient Cross-View Image Retrieval. In *IJCAI*. 1860–1866.

[22] Maxim Raginsky and Svetlana Lazebnik. 2009. Locality-sensitive binary codes from shift-invariant kernels. In *NIPS*. 1509–1517.

[23] Fumin Shen, Wei Liu, Shaoting Zhang, Yang Yang, and Heng Tao Shen. 2015. Learning Binary Codes for Maximum Inner Product Search. In *ICCV*. 4148–4156.

[24] Fumin Shen, Chunhua Shen, Wei Liu, and Heng Tao Shen. 2015. Supervised Discrete Hashing. In *CVPR*. 37–45.

[25] F. Shen, X. Zhou, Y. Yang, J. Song, H. T. Shen, and D. Tao. 2016. A Fast Optimization Method for General Binary Code Learning. *IEEE Trans. Image Process.* 25, 12 (2016), 5610–5621.

[26] Xiaoshuang Shi, Fuyong Xing, Jinzheng Cai, Zizhao Zhang, Yuanpu Xie, and Lin Yang. 2016. Kernel-Based Supervised Discrete Hashing for Image Retrieval. In *ECCV*. 419–433.

[27] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556 (2014).

[28] Jingkuan Song, Yang Yang, Yi Yang, Zi Huang, and Heng Tao Shen. 2013. Inter-media Hashing for Large-scale Retrieval from Heterogeneous Data Sources. In *SIGMOD*. 785–796.

[29] J. Wang, T. Zhang, j. song, N. Sebe, and H. T. Shen. 2017. A Survey on Learning to Hash. *IEEE Trans. Pattern Anal. Mach. Intell.* (2017). DOI: https://doi.org/10.1109/TPAMI.2017.2699960

[30] Yair Weiss, Antonio Torralba, and Robert Fergus. 2008. Spectral Hashing. In *NIPS*. 1753–1760.

[31] Tobias Weyand and Bastian Leibe. 2013. Discovering Details and Scene Structure with Hierarchical Iconoid Shift. In *ICCV*. 3479–3486.

[32] Tobias Weyand and Bastian Leibe. 2015. Visual landmark recognition from Internet photo collections: A large-scale evaluation. *Comput. Vis. Image Underst.* 135 (2015), 1–15.

[33] Yan Xia, K. He, P. Kohli, and J. Sun. 2015. Sparse projections for high-dimensional binary codes. In *CVPR*. 3332–3339.

[34] Liang Xie, Jialie Shen, and Lei Zhu. 2016. Online Cross-Modal Hashing for Web Image Retrieval. In *AAAI*. 294–300.

[35] Liang Xie, Lei Zhu, and Guoqi Chen. 2016. Unsupervised multi-graph cross-modal hashing for large-scale multimedia retrieval. *Multimedia Tools Appl.* 75, 15 (2016), 9185–9204.

[36] Liang Xie, Lei Zhu, Peng Pan, and Yansheng Lu. 2016. Cross-Modal Self-Taught Hashing for large-scale image retrieval. *Signal Process.* 124 (2016), 81–92.

[37] Yang Yang, Yadan Luo, Weilun Chen, Fumin Shen, Jie Shao, and Heng Tao Shen. 2016. Zero-Shot Hashing via Transferring Supervised Knowledge. In *MM*. 1286–1295.

[38] Yang Yang, Fumin Shen, Zi Huang, and Heng Tao Shen. 2016. A Unified Framework for Discrete Spectral Clustering. In *IJCAI*. 2273–2279.

[39] Felix X. Yu, Sanjiv Kumar, Yunchao Gong, and Shih-Fu Chang. 2014. Circulant Binary Embedding. In *ICML*. 946–954.

[40] Jile Zhou, Guiguang Ding, and Yuchen Guo. 2014. Latent Semantic Sparse Hashing for Cross-modal Similarity Search. In *SIGIR*. 415–424.

[41] Wengang Zhou, Ming Yang, Houqiang Li, Xiaoyu Wang, Yuanqing Lin, and Qi Tian. 2014. Towards Codebook-Free: Scalable Cascaded Hashing for Mobile Image Search. *IEEE Trans. Multimedia* 16, 3 (2014), 601–611.

[42] Lei Zhu, Jialie She, Xiaobai Liu, Liang Xie, and Liqiang Nie. 2016. Learning Compact Visual Representation with Canonical Views for Robust Mobile Landmark Search. In *IJCAI*. 3959–3965.

[43] Lei Zhu, Jialie Shen, Hai Jin, Liang Xie, and Ran Zheng. 2015. Landmark Classification With Hierarchical Multi-Modal Exemplar Feature. *IEEE Trans. Multimedia* 17, 7 (2015), 981–993.

[44] Lei Zhu, Jialie Shen, Hai Jin, Ran Zheng, and Liang Xie. 2015. Content-Based Visual Landmark Search via Multimodal Hypergraph Learning. *IEEE Trans. Cybern.* 45, 12 (2015), 2756–2769.

[45] L. Zhu, J. Shen, L. Xie, and Z. Cheng. 2016. Unsupervised Topic Hypergraph Hashing for Efficient Mobile Image Retrieval. *IEEE Trans. Cybern.* (2016). DOI: https://doi.org/10.1109/TCYB.2016.2591068

[46] Lei Zhu, Jialie Shen, Liang Xie, and Zhiyong Cheng. 2017. Unsupervised Visual Hashing with Semantic Assistant for Content-Based Image Retrieval. *IEEE Trans. on Knowl. and Data Eng.* 29, 2 (2017), 472–486.

[47] Xiaofeng Zhu, Zi Huang, Heng Tao Shen, and Xin Zhao. 2013. Linear Cross-modal Hashing for Efficient Multimedia Search. In *MM*. 143–152.

[48] X. Zhu, L. Zhang, and Z. Huang. 2014. A Sparse Embedding and Least Variance Encoding Approach to Hashing. *IEEE Trans. Image Process.* 23, 9 (2014), 3737–3750.