Supervised Hashing based on the Dimensions' Value Cardinalities of Image Descriptors

Dimitrios Rafailidis

Abstract—**This letter presents a hashing method, where the key idea is to exploit the discriminative power of image descriptors' Dimensions' Value Cardinalities (DVC), that is, the number of distinct values that occur at the dimensions of the descriptors. DVC are an inherit characteristic of image descriptors, capable of boosting the search accuracy for approximate nearest neighbor search. However, previous DVC-based search strategies function in an unsupervised manner. To account for the fact that the semantic information of images can significantly leverage the search accuracy, this letter proposes an efficient supervised hashing strategy based on DVC. Given a set of training data, the proposed approach first calculates a consensus sparse matrix, to consider both the DVC-based similarities and the sparse semantic information of images. Then, it formulates an objective function as a joint minimization problem, to jointly compute (i) the binary codes of the training data; and (ii) the projection matrix to map external queries to the Hamming space. The joint problem is solved via an efficient alternating optimization algorithm. Experiments on a benchmark dataset demonstrate the superiority of the proposed approach over other state-of-the-art supervised hashing and DVC-based search strategies.**

Index Terms—**Supervised hashing, visual search, dimensions' value cardinalities.**

I. Introduction

HASHING strategies have been widely used for *Approximate Nearest Neighbor* (ANN) search, due to their low storage cost and fast query speed [1]. The key idea is to compute binary/hash codes, by efficiently mapping the original data to the Hamming space. Hashing strategies can be roughly divided into *data-independent* [1]–[3] and *data-dependent* [4]–[12]. *Data-independent* approaches calculate binary codes by random projections, to construct similarity preserving hash functions based on different distance/similarity metrics [1]–[3]. As random projections are performed, a long length of binary codes is required to achieve high search accuracy in *data-independent* approaches [4].

Instead of performing random projections, *data-dependent* methods learn binary codes based on a training set. *Data-dependent* hashing can be further categorized into *unsupervised* and *supervised*. *Unsupervised* hashing strategies, such as Spectral Hashing [5], Iterative Quantization [6] and Anchor Graph Hashing [4], try to preserve the similarities of the training data using linear or nonlinear functions, when projecting the data to the Hamming space. However, *unsupervised* hashing strategies do not consider the images' semantic information, thus having limited search accuracy due to the well known semantic gap problem [13].

In the effort to bridge the semantic gap, *Supervised* hashing methods incorporate prior information, when learning the binary codes. Such prior information is usually expressed as pairwise labels of semantically similar and dissimilar data pairs. For instance, Minimal Loss Hashing (MLH) generates binary codes in a supervised manner, based on a structural SVM framework with latent variables [7]; Kernel-based Supervised Hashing (KSH) formulates an objective function for supervised hashing based on code inner products [8]; Fast supervised Hashing with decision trees (FastH) learns hash functions by training boosted decision trees and a GraphCut-based block search method [9]; Boosted Shared Hashing (BSH) formulates a unified objective function to learn both the hash functions and the semantically similarities of the images and follows a query adaptive retrieval strategy [10]; Neighborhood Discriminant Hashing (NDH) learns a discriminant hashing function by exploiting the images' local information, provided that an image and its visually similar neighbour have semantically similar labels [11]; Supervised Discrete hashing (SDH) defines a joint learning objective which integrates binary codes and linear classifier training to leverage the semantic label information [12]. Compared to *unsupervised* strategies, *supervised* hashing can significantly increase the search accuracy by exploiting the semantic information of images [7]–[12].

Recently, a new ANN search strategy has been been introduced, based on the Dimensions' Value Cardinalities (DVC) of image descriptor vectors [14]–[17]. DVC are the unique values that occur at the dimensions of the descriptors in an image collection (Section II). Various DVC-based approaches have been introduced, such as ANN search on a single machine [14] and distributed frameworks [15], or stream processing of image data [17]. For instance, in [14] Multi-Sort IndeXing (MSIDX) prioritizes the dimensions with high DVC in the search strategy, assuming that these dimensions have more discriminative power. In doing so, MSIDX achieves high search accuracy, compared to unsupervised hashing methods. However, all the aforementioned DVC-based strategies do not work in a supervised manner, thus having a glass ceiling on the ANN search accuracy.

Therefore, a pressing challenge resides on designing a hashing method, which considers both the semantic information of images and the discriminative power of DVC. The main contributions of this letter are summarized as follows, (i) a weighting scheme based on DVC is proposed to calculate the image similarities using a kernel function; (ii) the DVC-based weighted similarities and the semantic information of

D. Rafailidis is with the Department of Informatics, Aristotle University of Thessaloniki, GR-54124, Thessaloniki, Greece, e-mail: draf@csd.auth.gr.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2016.2599901, IEEE Signal Processing Letters

images in the form of must-link and cannot-link constraints are incorporated into a consensus sparse matrix; finally (iii) the objective function is formulated as a joint minimization problem, which is solved via alternating optimization [18], so as to jointly learn the binary codes and the projection matrix to map external queries to the Hamming space. In the experiments on a benchmark dataset, the proposed approach outperforms several state-of-the-art search strategies.

II. Dimensions' Value Cardinalities

Definition 1 (DVC)**.** *"The DVC of a dimension $i \in 1, \ldots, D$ is the number of unique values at the i-th dimension of all the descriptors in an image collection [14]–[16]."*

Property 1. *"DVC distributions highly depend on the descriptors' extraction strategies [16]."*

Property 2. *"Each descriptor extraction method tends to produce similar DVC distributions for different dataset sizes; therefore, ANN search strategies that exploit descriptors' DVC can scale, as the DVC distributions over the dimensions are preserved, irrespective of the dataset sizes. [16]."*

Property 3. *"Descriptors' DVC have a strong impact on the ANN search strategies, both in terms of search accuracy and query speed [16]."*

III. Problem Formulation

Given a training set with N samples, let $\mathbf{X} \in \Re^{D \times N}$ be a data matrix, where each column corresponds to a D-dimensional descriptor vector $\mathbf{x}_j = [x_1, x_2, \ldots, x_D] \in \Re^D$ of sample j, with $j = 1 \ldots N$. In our supervised approach, we assume that L distinct labels have been used, where l_j denotes the label of sample j. Based on the semantic label information, we generate must-link $\mathcal{ML} = \{(x, y)|l_x = l_y\}$ and cannot-link constraints $\mathcal{CL} = \{(x, y)|l_x \neq l_y\}$, with $x, y = 1 \ldots N$. The constraints are stored in a sparse matrix $\mathbf{S}_{label} \in \Re^{N \times N}$, with $[S_{label}]_{xy} = 1$, if $(x, y) \in \mathcal{ML}$; $[S_{label}]_{xy} = -1$, if $(x, y) \in \mathcal{CL}$; and 0 otherwise. The matrix with the binary codes of the training set is denoted by $\mathbf{B} \in \{0, 1\}^{C \times N}$, where the j-th column is the C-dimensional binary code $\mathbf{b}_j = [b_1, b_2, \ldots, b_C] \in \{0, 1\}^C$ of sample j. The problem that the proposed approach faces is formally defined as follows:

Definition 2 (Problem)**.** *"The goal of the proposed approach is to calculate the projection matrix $\mathbf{P} \in \Re^{D \times C}$ and the binary matrix $\mathbf{B}$, by considering the semantic information in $\mathbf{S}_{label}$ and preserving the original similarities, when mapping the data to the Hamming space."*

IV. Proposed Approach

The proposed hashing method consists of the following steps: (i) an efficient way is proposed to weigh the similarities based on the image descriptors' DVC; (ii) a consensus sparse matrix is calculated by considering both the semantic label information and the DVC-based weighted similarities; (iii) the projection matrix $\mathbf{P}$ and the binary matrix $\mathbf{B}$ are computed based on an efficient alternating optimization algorithm.

A. Weighting Strategy based on DVC

The training data matrix $\mathbf{X} \in \Re^{D \times N}$ is represented as follows:

$$\mathbf{X} = \begin{cases} X_{11}, X_{12}, \ldots, X_{1N} \\ X_{21}, X_{22}, \ldots, X_{2N} \\ \qquad \vdots \\ X_{D1}, X_{D2}, \ldots, X_{DN} \end{cases} \tag{1}$$

Based on the images descriptors' value types, that is, *integer*, *normalized real* and *real values* [14]–[16], $\forall\ i = 1, \ldots, D$, the i-th DVC is calculated for each row i of matrix $\mathbf{X}$, corresponding to the unique values that the i-th dimension has in the whole training set. The D different DVC are stored in a vector $\mathbf{u} = \{u_1, u_2, \ldots, u_D\} \in \Re^D$. Provided that dimensions with high DVC have more discriminative than those with lower ones, we weigh the DVC accordingly, by generating a D-dimensional vector $\mathbf{w} \in \Re^D$, where each element is calculated as follows:

$$w_i = \frac{u_i}{\max\{u_i | i = 1 \leq i \leq D\}} \log \frac{N}{q_i} \tag{2}$$

where q_i denotes the total number that the unique values X_{ij} appear in the i-th dimension of the N descriptors. The weighting vector $\mathbf{w}$ has real values in $(0, 1]$, with $w_i = 1$, if the i-th dimension has the highest DVC value. According to Property 2, DVC come from the same distribution family for different dataset sizes, which means that the relative differences between the DVC are preserved, thus producing similar weights in $\mathbf{w}$ for different dataset sizes. Following the notation of p-norm, the weighted distance $\mathcal{F}(\cdot)$ between two image descriptors $\mathbf{x}$ and $\mathbf{y}$ is computed as follows:

$$\mathcal{F}(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{D} w_i |x_i - y_i|^p \right)^{1/p} \tag{3}$$

In this letter, we consider the Euclidean distance ($p = 2$). The weighted distance function $\mathcal{F}$ amplifies the differences between two descriptors, when they have different values at dimensions with high DVC ($w_i \simeq 1$), and downweigh the differences at the dimensions with lower ones ($w_i \simeq 0$), respectively. The DVC-based weighted similarities are calculated based on the Gaussian kernel, which are stored in a matrix $\mathbf{S}_{DVC} \in \Re^{N \times N}$, with $[S_{DVC}]_{xy} = \exp(-\mathcal{F}(\mathbf{x}, \mathbf{y})/\sigma^2)$, and σ being the bandwidth of the kernel.

B. Consensus Sparse Matrix

The goal of this step is to compute a consensus sparse matrix $\mathbf{S} \in \Re^{N \times N}$, by combining the semantic information in $\mathbf{S}_{label}$ with the DVC-based weighted similarities in $\mathbf{S}_{DVC}$. We formulate the problem of computing $\mathbf{S}$ as a minimization problem, where we have to bring the consensus matrix $\mathbf{S}$ as close as possible to matrices $\mathbf{S}_{label}$ and $\mathbf{S}_{DVC}$, expressed by the approximation errors $||\mathbf{S} - \mathbf{S}_{label}||_F^2$ and $||\mathbf{S} - \mathbf{S}_{DVC}||_F^2$, respectively. In addition, as $\mathbf{S}_{label}$ is sparse (Section III), we

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2016.2599901, IEEE Signal Processing Letters

force the solution of the consensus matrix $\mathbf{S}$ to be sparse using the $l_{2,1}$-norm $||S||_{2,1}$, which is defined as [19]:

$$||\mathbf{S}||_{2,1} = \sum_{i=1}^{N} \sqrt{\sum_{j=1}^{N} S_{ij}^2} = \sum_{i=1}^{N} ||S_{i,*}||_2 \tag{4}$$

where $S_{i,*}$ denotes the i-th row of matrix $\mathbf{S}$. Hence, we have to solve the following minimization problem with respect to the consensus matrix $\mathbf{S}$:

$$\min_{\mathbf{S}} \mathcal{O} = ||\mathbf{S} - \mathbf{S}_{label}||_F^2 + ||\mathbf{S} - \mathbf{S}_{DVC}||_F^2 + \lambda ||\mathbf{S}||_{2,1} \tag{5}$$

where $\lambda > 0$ is a regularization parameter. By setting the gradient of $\mathcal{O}$ with respect to $\mathbf{S}$ equal to zero we have:

$$2(\mathbf{S} - \mathbf{S}_{label}) + 2(\mathbf{S} - \mathbf{S}_{DVC}) + 2\lambda(\mathbf{K}^{-1}\mathbf{S}) = 0 \tag{6}$$

with $\mathbf{K} \in \Re^{N \times N}$ being a diagonal matrix, whose entries are calculated as follows, $K_{ii} = 2||S_{i,*}||_2$ for the i-th diagonal entry. According to (6), we compute the following closed form solution of the consensus matrix $\mathbf{S}$:

$$\mathbf{S} = (2\mathbf{I} + \lambda \mathbf{K}^{-1})^{-1}(\mathbf{S}_{label} + \mathbf{S}_{DVC}) \tag{7}$$

where $\mathbf{I}$ is the identity matrix.

C. Projection Matrix and Binary Codes

Having computed the consensus matrix $\mathbf{S}$, we have to calculate the projection matrix $\mathbf{P}$ and the binary matrix $\mathbf{B}$. Based on graph-based strategies [5], we formulate the following joint minimization problem:

$$\min_{\mathbf{B},\mathbf{P}} \mathcal{G} = ||\mathbf{X} - \mathbf{P}\mathbf{B}||_F^2 + Tr(\mathbf{B}\mathbf{L}\mathbf{B}^T) + \alpha(||\mathbf{B}||_F^2 + ||\mathbf{P}||_F^2) \tag{8}$$

$$\text{subject to } \mathbf{B} \in \{0,1\}^{C \times N}$$

where $Tr(\cdot)$ is the trace operator. Matrix $\mathbf{L} \in \Re^{N \times N}$ is the graph Laplacian constructed based on the consensus matrix $\mathbf{S}$, which is calculated as $\mathbf{L} = \mathbf{D} - \mathbf{S}$, and $\mathbf{D} \in \Re^{N \times N}$ is the degree matrix, a diagonal matrix with $D_{ii} = \sum_j S_{ij}$. The first term of (8) expresses the reconstruction error between $\mathbf{PB}$ and the original data matrix $\mathbf{X}$; the second term $Tr(\mathbf{BLB}^T)$ is the matrix form of $\sum_{ij} S_{ij}||\mathbf{b}_i - \mathbf{b}_j||^2$, which expresses if two samples i and j are similar in the consensus matrix $\mathbf{S}$, then they should have similar binary codes $\mathbf{b}_i$ and $\mathbf{b}_j$ in the Hamming space; and the last one is the regularization term on variables $\mathbf{B}$ and $\mathbf{P}$ to avoid model overfitting, with α being the regularization parameter.

The minimization problem of $\mathcal{G}$ in (8) is intractable because of the discrete constraint in $\mathbf{B}$, that is, taking only the values 0 and 1. Thus, at the current stage we relax the constraint by taking continuous real values in $\mathbf{B}$. Nonetheless, the minimization problem of (8) remains intractable, as we have to jointly minimize $\mathcal{G}$ with respect to $\mathbf{B}$ and $\mathbf{P}$. We solve the minimization problem of (8) using alternating optimization [18], that is, we fix matrix $\mathbf{P}$ and optimize with respect to $\mathbf{B}$, and then we fix the updated matrix $\mathbf{B}$ and optimize with respect to $\mathbf{P}$.

-Fix $\mathbf{P}$ and update $\mathbf{B}$ (Step 1): Using the trace operator $Tr(\cdot)$, Eq. (8) can be rewritten as follows:

$$\begin{aligned}\mathcal{G}(\mathbf{B},\mathbf{P}) = Tr\big[(\mathbf{X} - \mathbf{PB})(\mathbf{X} - \mathbf{PB})^T\big] \\ + Tr(\mathbf{BLB}^T) + \alpha Tr(\mathbf{BB}^T) + \alpha Tr(\mathbf{PP}^T)\end{aligned} \tag{9}$$

The partial derivative of $\mathcal{G}$ with respect to $\mathbf{B}$ in (9) is calculated as follows:

$$\frac{\partial \mathcal{G}(\mathbf{B},\mathbf{P})}{\partial \mathbf{B}} = -2\mathbf{P}^T\mathbf{X} + 2\mathbf{P}^T\mathbf{PB} + 2\mathbf{BL} + 2\alpha\mathbf{B} \tag{10}$$

As there is no closed form solution for $\mathbf{B}$ in (10), we calculate the matrix $\mathbf{B}$ based on the L-BFGS Quasi-Newton method [20], using the `libLBFGS` library [21].

-Fix $\mathbf{B}$ and update $\mathbf{P}$ (Step 2): Having fixed matrix $\mathbf{B}$, we set the gradient of $\mathcal{G}$ with respect to $\mathbf{P}$ equal to zero:

$$\frac{\partial \mathcal{G}(\mathbf{B},\mathbf{P})}{\partial \mathbf{P}} = -2\mathbf{XB}^T + 2\mathbf{PBB}^T + 2\alpha\mathbf{P} = 0 \tag{11}$$

As $(\mathbf{BB}^T + a\mathbf{I})$ is positive definite, based on (11) we can obtain a closed form solution for $\mathbf{P}$:

$$\mathbf{P} = \mathbf{XB}^T(\mathbf{BB}^T + \alpha\mathbf{I})^{-1} \tag{12}$$

The two-step alternating optimization is an iterative algorithm, which is repeated until convergence, concluding in real valued matrices $\mathbf{B}$ and $\mathbf{P}$. At the first iteration, we use the identity matrix to initialize $\mathbf{P}$.

Generate binary codes: As aforementioned, we relaxed the constraint in (8) over the alternating optimization algorithm, which results in a matrix $\mathbf{B}$ with real values. To generate the final binary codes $\mathbf{B} \in \{0,1\}^{C \times N}$, we follow the maximum entropy principle, based on which a binary bit $c = 1, \ldots, C$ that gives balanced partitioning of the training set should provide maximum information [22]. We apply the following thresholding strategy, for the c-th bit we quantize the real values in the c-th row of $\mathbf{B}$ using the respective mean value $M_c = 1/N \sum_j B_{c,j}$, with $B_{cj} = 1$, if $B_{cj} > M_c$, and 0 otherwise.

Out-of-sample extension: Given an external sample $j \notin \mathbf{X}$, we have to compute the binary codes of $\mathbf{b}_j \in \Re^C$. Based on the projection matrix $\mathbf{P} \in \Re^{D \times C}$ we create a real valued vector $\mathbf{b}_j = \mathbf{x}_j\mathbf{P}$, where $\mathbf{x}_j$ is the D-dimensional image descriptor vector of the external sample j. The final binary codes are quantized in $\{0,1\}$ according to the aforementioned thresholding strategy.

V. Experiments

A. Data Analysis

The evaluation is performed on the NUS-WIDE dataset [23], which consists of 269,648 images with 81 concept labels, and each image is represented by a 500-dimensional Bag-of-Words vector. In Fig. 1, we plot the DVC for each dimension, where the "picks" represent the dimensions with high DVC. To verify that the values in the weighting vector $\mathbf{w}$ of (2) are similar for different dataset sizes (Section IV-A), the NUS-WIDE dataset was uniformly down-sampled from 100% to 20%, by a step of 20%. In Fig. 2(a)-(b), we report the respective five empirical cumulative distribution functions of (a) the descriptors' DVC

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2016.2599901, IEEE Signal Processing Letters

and (b) the weights in $\mathbf{w}$ for each down-sampled dataset, with $F(x) = P(X \leq x)$, and $x = \text{DVC}_i$, $x = w_i$. Using the Kolmogorov-Smirnov test ($p \leq 0.01$), we verified that the five DVC distributions in Fig. 2(a) come from the same distribution family, which confirms that DVC have Property 2, as in the analysis of [16]. Fig. 2(b) shows that the five different down-sampled datasets generate similar weight distributions, thus having similar values in the weighting vector $\mathbf{w}$ of (2) for the different dataset sizes.

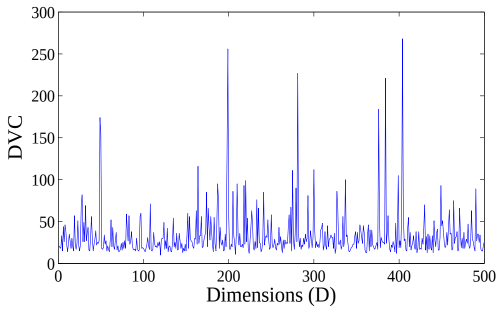

Fig. 1. DVC of the 500-dimensional Bag-of-Words vectors in NUS-WIDE.

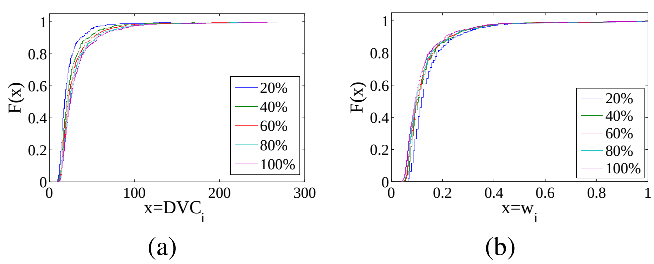

Fig. 2. Empirical distributions functions of (a) DVC_i and (b) weights w_i for different dataset sizes.

B. Settings

The ANN accuracy is measured in terms of Mean Average Precision (MAP), where the true neighbors of a query are defined as the images that share at least one label with the query image [7], [8], [12]. Following [4], [12], the 21 most frequent labels are kept, where 100 images from each label are randomly selected to form the query set, and the results are averaged over ten runs. In the proposed DVC-based Supervised Hashing method, namely DVC-SH, parameters λ=1e-1 and α=1e-3 are determined based on cross validation. The complexity of DVC-SH depends on the number of iterations that the alternating optimization algorithm needs to converge. A predefined convergence threshold is set to 1e-4 and the maximum number of iterations is fixed to 20. In the ten runs, the algorithm needs up to 14 iterations to converge.

C. Results

DVC-SH is compared with the unsupervised strategies LSH [1] and MSIDX [14]. The search radius in MSIDX is varied so as to produce the same bit budget with the hashing methods [14]. In addition, two variants of the proposed approach are used, namely SH and SHL. The SH variant ignores the DVC-based weighting strategy of Section IV-A, with $\mathbf{w} = \mathbf{1}$, and the SHL variant uses only the $\mathbf{S}_{label}$ matrix in (5). Fig. 3 shows that the supervised approaches of SH, SHL and DVC-SH clearly outperform the unsupervised methods of MSIDX and LSH. The SHL variant has limited performance, when comparing with DVC-SH and SH, as SHL uses only the sparse semantic information of images when computing $\mathbf{S}$, thus generating a sparse similarity graph. Meanwhile, the proposed DVC-SH method beats the SH variant, by considering the discriminative power of DVC, when weighting the image similarities.

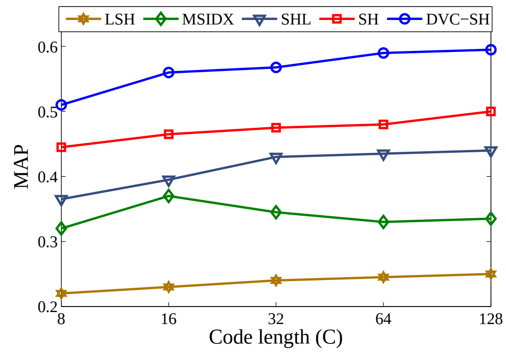

Fig. 3. Comparison with LSH, MSIDX, and the variants SH and SHL.

Next, the proposed DVC-SH approach is compared with the supervised hashing methods MLH [7], KSH, [8], FastH [9] and SDH [12], using their implementations at [24], [25], [26] and [27], respectively. In FastH, the tree depth is fixed to 4 [9], and in SDH the number of anchor points is set to 1,000 [12]. Fig. 4 shows that DVC-SH outperforms the competitive supervised hashing methods, achieving a relative improvement of 8.5-12.4%, when varying the code lengths. This happens because DVC-SH considers both the DVC-based weighted similarities and the semantic information of images, when generating the binary codes.

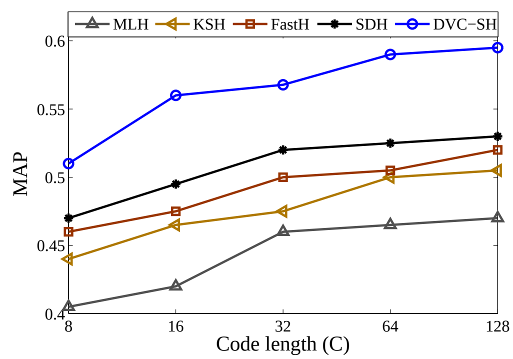

Fig. 4. Comparison with supervised hashing methods.

VI. Conclusion

This letter presented DVC-SH, an efficient supervised hashing method. DVC-SH computes a consensus sparse matrix to capture both the DVC-based weighted similarities and the images' semantic information when learning the binary codes. The experimental results showed that DVC-SH achieves high search accuracy, compared to other DVC-based strategies and supervised hashing methods. As future work, we will consider the impact of DVC on cross-view hashing [28], [29].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LSP.2016.2599901, IEEE Signal Processing Letters

References

[1] A. Gionis, P. Indyk, and R. Motwani, "Similarity search in high dimensions via hashing," in *25th International Conference on Very Large Data Bases, VLDB*, 1999, pp. 518–529.

[2] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *ACM Symposium on Computational Geometry*, 2004, pp. 253–262.

[3] M. Raginsky and S. Lazebnik, "Locality-sensitive binary codes from shift-invariant kernels," in *Advances in Neural Information Processing Systems 22, Proceedings of the Annual Conference on Neural Information Processing Systems, NIPS*, 2009, pp. 1509–1517.

[4] W. Liu, J. Wang, S. Kumar, and S. Chang, "Hashing with graphs," in *28th International Conference on Machine Learning, ICML*, 2011, pp. 1–8.

[5] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Advances in Neural Information Processing Systems 21, Proceedings of the Annual Conference on Neural Information Processing Systems, NIPS*, 2008, pp. 1753–1760.

[6] Y. Gong and S. Lazebnik, "Iterative quantization: A procrustean approach to learning binary codes," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2011, pp. 817–824.

[7] M. Norouzi and D. J. Fleet, "Minimal loss hashing for compact binary codes," in *28th International Conference on Machine Learning, ICML*, 2011, pp. 353–360.

[8] W. Liu, J. Wang, R. Ji, Y. Jiang, and S. Chang, "Supervised hashing with kernels," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2012, pp. 2074–2081.

[9] G. Lin, C. Shen, Q. Shi, A. van den Hengel, and D. Suter, "Fast supervised hashing with decision trees for high-dimensional data," in *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 2014, pp. 1971–1978.

[10] X. Liu, Y. Mu, B. Lang, and S. Chang, "Mixed image-keyword query adaptive hashing over multilabel images," *TOMCCAP*, vol. 10, no. 2, p. 22, 2014.

[11] J. Tang, Z. Li, M. Wang, and R. Zhao, "Neighborhood discriminant hashing for large-scale image retrieval," *IEEE Trans. Image Processing*, vol. 24, no. 9, pp. 2827–2840, 2015.

[12] F. Shen, C. Shen, W. Liu, and H. T. Shen, "Supervised discrete hashing," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2015, pp. 37–45.

[13] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1349–1380, 2000.

[14] E. Tiakas, D. Rafailidis, A. Dimou, and P. Daras, "Msidx: Multi-sort indexing for efficient content-based image search and retrieval," *IEEE Transaction on Multimedia*, vol. 15, no. 6, pp. 1415–1430, 2013.

[15] S. Antaris and D. Rafailidis, "Similarity search over the cloud based on image descriptors' dimensions value cardinalities," *ACM Transactions on Multimedia Computing, Communications & Applications*, vol. 11, no. 4, p. 51, 2015.

[16] T. Semertzidis, D. Rafailidis, M. Strintzis, and P. Daras, "The influence of image descriptors' dimensions' value cardinalities on large-scale similarity search," *International Journal of Multimedia Information Retrieval*, pp. 1–18, 2014.

[17] D. Rafailidis and S. Antaris, "Indexing media storms on flink," in *2015 IEEE International Conference on Big Data*, 2015, pp. 2836–2838.

[18] P. Jain, P. Netrapalli, and S. Sanghavi, "Low-rank matrix completion using alternating minimization," in *Symposium on Theory of Computing Conference, STOC*, 2013, pp. 665–674.

[19] F. Nie, H. Huang, X. Cai, and C. H. Q. Ding, "Efficient and robust feature selection via joint l2,1-norms minimization," in *Advances in Neural Information Processing Systems 23, Proceedings of the Annual Conference on Neural Information Processing Systems, NIPS*, 2010, pp. 1813–1821.

[20] J. L. Morales and J. Nocedal, "Automatic preconditioning by limited memory quasi-newton updating," *SIAM Journal on Optimization*, vol. 10, no. 4, pp. 1079–1096, 2000.

[21] The libLBFGS library, [Online]. Available: https://github.com/chokkan/liblbfgs, accessed Feb. 10, 2016.

[22] R. Lin, D. A. Ross, and J. Yagnik, "SPEC hashing: Similarity preserving algorithm for entropy-based coding," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2010, pp. 848–854.

[23] The NUS-WIDE dataset, [Online]. Available: http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm, accessed Jan. 10, 2016.

[24] MLH source code, [Online]. Available: https://github.com/marklar/min-loss-hashing, accessed Feb. 10, 2016.

[25] KSH source code, [Online]. Available: http://www.ee.columbia.edu/ln/dvmm/downloads/WeiKSHCode/dlform.htm, accessed Feb. 10, 2016.

[26] FastH source code, [Online]. Available: https://bitbucket.org/chhshen/fasthash/, accessed May. 22, 2016.

[27] SDH source code, [Online]. Available: https://github.com/bd622/DiscretHashing, accessed Feb. 10, 2016.

[28] M. M. Bronstein, A. M. Bronstein, F. Michel, and N. Paragios, "Data fusion through cross-modality metric learning using similarity-sensitive hashing," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2010, pp. 3594–3601.

[29] G. Ding, Y. Guo, and J. Zhou, "Collective matrix factorization hashing for multimodal data," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2014, pp. 2083–2090.