Thomas Reato, Begüm Demir, Lorenzo Bruzzone

An Unsupervised Multicode Hashing Method for Accurate and Scalable Remote Sensing Image Retrieval

Journal article | Accepted manuscript (Postprint)
This version is available at https://doi.org/10.14279/depositonce-9327

Reato, T., Demir, B., & Bruzzone, L. (2019). An Unsupervised Multicode Hashing Method for Accurate and Scalable Remote Sensing Image Retrieval. IEEE Geoscience and Remote Sensing Letters, 16(2), 276–280. https://doi.org/10.1109/lgrs.2018.2870686

An Unsupervised Multicode Hashing Method for Accurate and Scalable Remote Sensing Image Retrieval

Thomas Reato, Begüm Demir, *Senior Member, IEEE*, and Lorenzo Bruzzone, *Fellow, IEEE*

***Abstract*— Hashing methods have recently attracted great attention for approximate nearest neighbor search in massive remote sensing (RS) image archives due to their computational and storage effectiveness. The existing hashing methods in RS represent each image with a single-hash code that is usually obtained by applying hash functions to global image representations. Such an approach may not optimally represent the complex information content of RS images. To overcome this problem, in this letter, we present a simple yet effective unsupervised method that represents each image with primitive-cluster sensitive multi-hash codes (each of which corresponds to a primitive present in the image). To this end, the proposed method consists of two main steps: 1) characterization of images by descriptors of primitive-sensitive clusters and 2) definition of multi-hash codes from the descriptors of the primitive-sensitive clusters. After obtaining multi-hash codes for each image, retrieval of images is achieved based on a multi-hash-code-matching scheme. Any hashing method that provides single-hash code can be embedded within the proposed method to provide primitive-sensitive multi-hash codes. Compared with state-of-the-art single-code hashing methods in RS, the proposed method achieves higher retrieval accuracy under the same retrieval time, and thus it is more efficient for operational applications.**

***Index Terms*— Content-based image retrieval, image information mining, multicode hashing, remote sensing (RS).**

I. Introduction

DUE to explosive growth of remote sensing (RS) image archives, development of content-based image search and retrieval (CBIR) methods has recently attracted a great deal of research in RS. Simple nearest neighbor (NN) search algorithms that exhaustively compare the query image with each image in the archive (e.g., k-NN algorithm) are time demanding, and thus impractical for operational RS CBIR applications. A solution to avoid an exhaustive comparison is to employ approximate NN (ANN) search algorithms. ANN algorithms aim at identifying images from a large archive that have a high probability to be NNs of the query image with a sublinear, or even constant time complexity [1]. Recently, hashing-based ANN search techniques have been used in RS due to their high time-efficient (in terms of both storage and speed) and accurate search capability within huge data archives [2].

Hashing methods map the original feature space into a low-dimensional Hamming (binary) space by a set of hash functions [1]. By this way, the images are represented by binary hash codes, and therefore image retrieval can be achieved by calculating the Hamming distances with simple bit-wise exclusive-OR (XOR) operations. The most well-known hashing method is locality-sensitive hashing (LSH), in which random projections are exploited to produce binary hash codes. LSH is applied to many CBIR problems due to its simplicity and efficiency. However, it needs to generate long binary codes to achieve accurate results. This issue makes it inappropriate for complex RS CBIR problems. Recently, we presented, adapted to RS data properties and tested two kernel-based hashing methods [2]: 1) kernel-based unsupervised LSH (KULSH) [3] and 2) kernel-based supervised LSH (KSLSH) [4]. The KULSH defines hash functions in the kernel space by using only unlabeled images (and thus it is unsupervised), whereas the KSLSH leverages on the semantic similarity extracted by annotated images to describe much distinctive hash functions in the kernel space (and thus it is supervised). In [5], to perform fast parallel RS image retrieval a parallelization of KULSH using graphical processing units is presented. A partial randomness hashing method, which improves the accuracy of the LSH by considering a weight matrix defined based on the labeled training RS images is presented in [6]. Unlike RS, in the computer vision and multimedia communities, the use of hashing is widely studied. An extensive analysis comparing various hashing methods is given in [7].

The few hashing methods presented in RS are commonly designed for representing each image by a single-hash code obtained by applying hash functions to global image representations (which can be defined through the assembly of local features) [2], [5], [6]. Accordingly, while defining the hashing functions, these hashing methods do not take into account the possible primitives (such as different land-cover classes) present in the images. However, there are usually several regions within each RS image associated with different land-

This work was supported by the European Research Council under the ERC Starting Grant BigEarth-759764. *(Corresponding author: Begüm Demir.)*

T. Reato and L. Bruzzone are with the Department of Computer Science and Information Engineering, University of Trento, 38123 Trento, Italy.

B. Demir is with the Faculty of Electrical Engineering and Computer Science, Technische Universitat Berlin, D-10587 Berlin, Germany (e-mail: demir@tu-berlin.de).

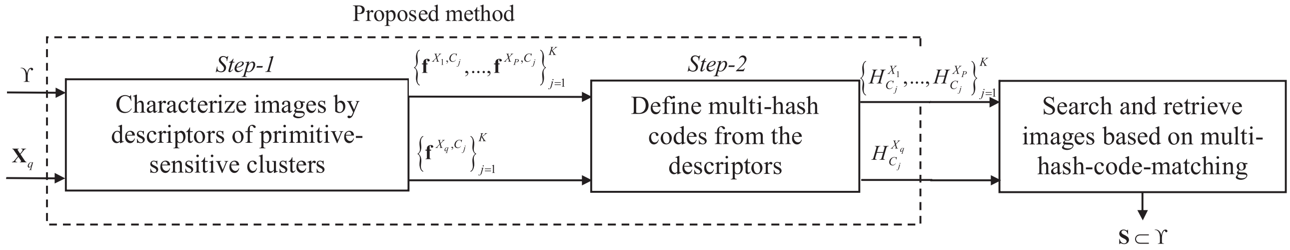

Fig. 1. Block diagram of the proposed method (this diagram assumes that $\mathbf{X}_q \notin \Upsilon$).

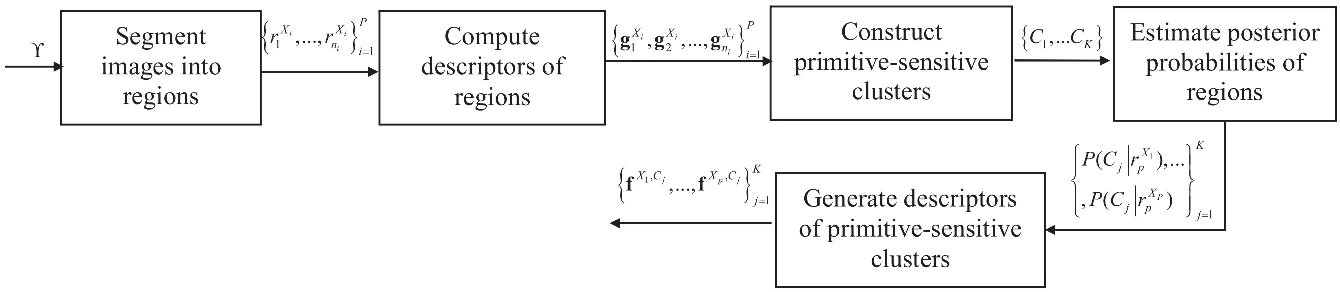

Fig. 2. Block diagram of the task of generation of the descriptors of the primitive-sensitive clusters of the archive images.

cover classes. Thus, representing an RS image with a global image descriptor and thus with a single-hash code without explicitly modeling semantic of different regions may lead to inaccurate results (particularly when specific semantic content is present in the query images). Thus, the characterization of primitives present images in the construction of binary hash codes can be very useful to accurately model the complex semantic content of RS images. To address these issues and accurately characterize the primitives present in RS images using hashing, in this letter, we introduce a novel unsupervised strategy, which represents each image with multi-hash codes, where each code corresponds to a primitive. The contribution of this letter, which significantly extends the work presented in [8], consists in introducing a novel strategy to describe each image by a set of descriptors of primitive-sensitive clusters and their hash codes for large-scale RS retrieval problems. The experimental results obtained on a benchmark archive demonstrate that the proposed method outperforms the state-of-the-art single-code unsupervised hashing method.

II. Proposed Method

Let $\Upsilon = \{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_P\}$ be an archive that consists of a very large number P of RS images, where $\mathbf{X}_i$ is the ith image. Given a query image $\mathbf{X}_q (\mathbf{X}_q \in \Upsilon$ or $\mathbf{X}_q \notin \Upsilon)$, we aim to: 1) improve the capability of the traditional hashing methods in describing semantic content of RS images without using any annotated training image and 2) retrieve a set $\mathbf{S} \subset \Upsilon$ of images semantically very similar to $\mathbf{X}_q$ from the archive Υ with a time efficient and accurate search. To this end, in this letter, we propose a novel unsupervised strategy that describes each RS image by primitive-sensitive multi-hash codes. The proposed method, which can be used with any kind of RS images (e.g., multispectral, hyperspectral, and SAR images), is defined based on two-steps: 1) characterization of images by descriptors of primitive-sensitive clusters and 2) definition of multi-hash codes from the descriptors of the primitive-sensitive clusters. After obtaining the multi-hash codes, retrieval of RS images is achieved based on the similarities estimated between the multi-hash codes of the query image and those of each image in the archive. Fig. 1 shows the block scheme of the proposed method. Each step is explained in detail in the following.

A. Step 1: Characterization of Images by Descriptors of Primitive-Sensitive Clusters

This step is devoted to represent each RS image with descriptors of primitive-sensitive clusters in order to capture its semantic content. To this end, the primitive-sensitive clusters are initially computed based on a novel unsupervised algorithm that includes three substeps (see Fig. 2). In the first substep, images $\mathbf{X}_i, i = 1, 2, \ldots, P$ in the archive are segmented into a set of regions $\{r_1^{X_i}, r_2^{X_i}, \ldots, r_{n_i}^{X_i}\}$ (where $r_p^{X_i}$ is the pth region of $\mathbf{X}_i$ and n_i is the total number of regions in $\mathbf{X}_i$). Segmentation is achieved by using the parametric kernel graph cut algorithm as suggested in [9]. In the second substep, a region descriptor $\mathbf{g}_p^{X_i}$ (i.e., a feature vector that models the region) is computed for each region $r_p^{X_i}, p = 1, 2, \ldots, n_i$. Region descriptors should be defined taking into account the specific properties of the considered RS images. In the third substep, a set of regions are randomly selected and then clustered into K clusters $\{C_1, C_2, \ldots C_K\}$. Each cluster $C_j, j = 1, 2, \ldots, K$ (where C_j is the jth cluster) is treated as the jth primitive and called as jth primitive-sensitive cluster. Clustering is an effective way to characterize primitives without using any training image. In this letter, it is accomplished by using Gaussian mixture models and parameters of the mixture models with K components are estimated by the expectation–maximization algorithm.

After defining the primitive-sensitive clusters, a relationship between the regions $r_p^{X_i}, p = 1, 2, \ldots, n_i$ of each image $\mathbf{X}_i$ and the set of primitive-sensitive clusters $\{C_1, C_2, \ldots C_K\}$ is built. This is achieved based on the posterior probabilities $P(C_j|r_p^{X_i})$ of each region $r_p^{X_i}$ within $\mathbf{X}_i$ to belong to the jth cluster C_j. To this end, posterior probabilities $P(C_j|r_p^{X_i})$ are estimated from the parameters of the mixture models. Then, the descriptor $\mathbf{f}^{X_i,C_j}$ of the jth primitive for $\mathbf{X}_i$ is defined as the average of the descriptors of the regions belonging to the jth primitive-sensitive cluster with a probability higher than a threshold Th as follows:

$$\mathbf{f}^{X_i,C_j} = \frac{1}{nr} \sum_{\forall P(C_j|r_p^{X_i}) \geq \mathrm{Th}} \mathbf{g}_p^{X_i} \tag{1}$$

where nr is the number of regions for which the posterior probabilities are greater than Th. Equation (1) relies on the fact that the regions associated with a probability higher than a threshold Th are highly representative of that cluster. If there is no region with posterior probability greater than or equal to the given threshold Th (i.e., $P(C_j|r_p^{X_i}) < \mathrm{Th}, \forall r_p^{X_i}, p = 1, 2, \ldots, n_i$), the jth primitive is considered as not existing in $\mathbf{X}_i$. Thus, the related descriptor is defined as $\mathbf{f}^{X_i,C_j} = \mathbf{z}$, where $\mathbf{z}$ is a vector of all zero entries. It is worth noting that the number of nonnull image descriptors is equal or smaller than the total number K of primitive-sensitive clusters and it does not depend on the number of regions present in the images. It is important to observe that the result of this step depend on the value of threshold Th and the number K of primitive-sensitive clusters. Selecting a relatively high Th leads to a low sensitivity to the different semantic classes possibly present in the image. On the one hand, a relatively high values of K increases the number of descriptors per image and thus also increases the computational time and storage complexity. On the other hand, if a relatively small K value is selected, one single cluster may represent more than one primitive class. In other words, descriptors of regions belonging to different primitive classes can be assigned to the same cluster, reducing the capability of the descriptor to accurately model the considered primitive. It is worth noting that this step can be accomplished by: 1) any unsupervised segmentation algorithm that accurately divides images into semantically meaningful regions and 2) any clustering algorithm that provides posterior probabilities to build an accurate relationship between the regions and the primitive-sensitive clusters.

B. Step 2: Definition of Multi-hash Codes From the Descriptors of the Primitive-Sensitive Clusters

Once the descriptors $\{\mathbf{f}^{X_1,C_j}, \ldots, \mathbf{f}^{X_P,C_j}\}_{j=1}^{K}$ of primitive-sensitive clusters are estimated, a standard single-code hashing method is applied to the descriptors associated with each primitive-sensitive cluster independently of each other. We use the KULSH method [3], which is able to define effective hash functions for nonlinearly separable high-dimensional RS image descriptors in the kernel space [2]. Differently from [2], in this letter, the KULSH is applied to descriptors $\{\mathbf{f}^{X_1,C_j}, \ldots, \mathbf{f}^{X_P,C_j}\}$ of the jth primitive-sensitive cluster independently of those of the other primitive-sensitive clusters. Thus, the rth hash function h_r^j associated with the descriptors of the jth primitive-sensitive cluster is defined as:

$$h_r^j(\mathbf{f}^{X_i,C_j}) = \mathrm{sign}\left(\sum_{p=1}^{P} \boldsymbol{\omega}_r^{C_j}(j) K(\mathbf{f}^{X_p,C_j}, \mathbf{f}^{X_i,C_j})\right) \tag{2}$$

where $K(\cdot,\cdot)$ is a kernel function and $\boldsymbol{\omega}_r^{C_j}$ is the coefficient vector defined as follows:

$$\boldsymbol{\omega}_r^{C_j} = \mathbf{K}_{C_j}^{-1/2}\mathbf{e} \tag{3}$$

where $\mathbf{K}_{C_j} \in \mathbb{R}^{m\times m}$ is the kernel matrix formed by m descriptors randomly selected associated with the jth primitive-sensitive cluster and $\mathbf{K}_{C_j}^{-1/2}$ is obtained as $\mathbf{K}_{C_j}^{-1/2} = V\Lambda^{-1/2}V^T$, where $V\Lambda V^T$ is the Eigen-decomposition of $\mathbf{K}_{C_j}$ and $\mathbf{e}$ is a vector with ones in the components corresponding to the indices of a subset of t descriptors randomly selected among the m descriptors. The choice of sampling t descriptors is for simplifying the computations as suggested in [3]. The same process defined in (2) is applied to a total of b hash functions $[h_1^j, h_2^j, \ldots, h_b^j]$. This leads to a b-bits hash code $H_{C_j}^{X_i} = [h_1^k(\mathbf{f}^{X_i,C_j})h_2^k(\mathbf{f}^{X_i,C_j})\ldots h_b^k(\mathbf{f}^{X_i,C_j})]$ that represents the descriptor of the jth primitive-sensitive cluster in $\mathbf{X}_i$. All these processes are repeated for descriptors of other primitive-sensitive clusters and the set of hash codes $\{H_{C_j}^{X_i}\}_{j=1}^{K}$ is obtained for each image. The set $\{H_{C_j}^{X_q}\}_{j=1}^{K}$ of hash codes for a query image $\mathbf{X}_q$ is also computed by applying the same b hash functions associated with each primitive cluster.

It is worth noting that according to our knowledge, this is the first time that the KULSH is used in the framework of primitive-sensitive unsupervised multi-hash codes generation. This requires definition of a series of hash functions related to each primitive-sensitive cluster, each of which generates one hash bit to represent the descriptor of the considered primitive-sensitive cluster. Note that any hashing method that provides single-hash code can be adopted here to provide multi-hash codes after properly applying the first step of the proposed method. After the images are represented by multi-hash codes, a multi-hash-code-matching scheme is utilized to retrieve the images based on the similarities among the multi-hash codes. To this end, the similarity between $\mathbf{X}_q$ and $\mathbf{X}_i, i = 1, 2, \ldots, P$ is estimated as follows:

$$d^{X_q,X_i} = \sum_{j=1}^{K} H_{C_j}^{X_q} \otimes H_{C_j}^{X_i} \quad \text{if } \mathbf{f}^{X_q,C_j} \neq \mathbf{z} \tag{4}$$

where d^{X_q,X_i} is the total Hamming distance between the hash codes of $\mathbf{X}_q$ and $\mathbf{X}_i$ and $\otimes$ represents the XOR operation. Then, the set $\mathbf{S} \subset \Upsilon$ of images that have the lowest distance is retrieved.

III. Data Set Description and Design of Experiments

In our experiments, the UCMERCED benchmark archive, which consists of 2100 images selected from aerial orthoimagery with a spatial resolution of 30 cm [10], was utilized. To assess the performance of the proposed method, we have used the annotations of primitive classes of images that are described in [9] and available at "http://bigearth.eu/datasets." The total number of considered primitive class labels is 17 (which are: airplane; bare soil; buildings; cars; chaparral; court; dock; field; grass; mobile home; pavement; sand; sea; ship; tanks; trees; and water), while the number of labels associated with each image varies between 1 and 7. In the

TABLE I

AVERAGE RECALL, RETRIEVAL TIME, AND STORAGE COMPLEXITY OF THE PROPOSED AND STANDARD METHODS OBTAINED WHEN $b = 8$ AND $b = 32$

Results	Standard k-nn	Proposed k-nn	Single-code hashing [2]	Proposed multi-code hashing	Single-code hashing [2]	Proposed multi-code hashing
			b=8		b=32	
Recall	66.29 %	68.26 %	55.76 %	61.12%	58.85	65.29 %
Time (in seconds)	128×10^{-4}	1475×10^{-4}	10×10^{-4}	10×10^{-4}	41×10^{-4}	41×10^{-4}
Storage Complexity	1.640 KB	30.140 KB	0.0083 KB	0.0610 KB	0.0330 KB	0.0680 KB

experiments, each region is described by the concatenation of following features: 1) shape features (which are Fourier descriptors and contour-based shape descriptors); 2) texture features (which are entropy and spectral histograms); and 3) intensity features (which are mean and standard deviation of the samples within each region). The reader is referred to [9] for further details on the considered features. In the experiments, we compared the proposed multicode hashing with the KULSH that is a state-of-the-art unsupervised single-code hashing method [3]. The results obtained by the KULSH are denoted as standard single-code hashing in the experiments. In addition to the complete use of the proposed multicode hashing method, we have also assessed the effectiveness of the descriptors of primitive-sensitive clusters (which are obtained by only applying step 1). To this end, the similarities between the descriptors of primitive-sensitive clusters of the query image (when $\mathbf{f}^{X_q,C_k} \neq \mathbf{z}$) and those of all archive images are estimated by using the summation of the pairwise Gaussian radial basis function (RBF) kernel similarities (denoted as proposed k-NN). Results achieved by the proposed method are compared with those obtained by the standard k-NN algorithm (which computes the similarity between the query image and all archive images based on the global image descriptors). In our experiments, we have considered spectral histogram features extracted from the entire image as the global image descriptor. In the experiments, Gaussian RBF kernel was used also for the KULSH and the standard k-NN. Results of each method are provided in terms of: 1) average recall; 2) storage complexity; and 3) average computational time obtained in 2100 trials performed with 2100 selected query images from the archive. Average recall is defined as the average ratio of the total number of intersecting labels between the query and the retrieved images to the number of labels associated with the query image [9]. The retrieval performance was assessed on top-20 retrieved images. All the experiments were implemented via MATLAB on a standard PC with Intel Xeon CPU E5-1650 v2 @ 3.50 GHz, 16 GB RAM. While estimating the storage complexity of image and region descriptors, we consider that each feature is stored as a double floating-point number. The total number of hash bits considered in the retrieval phase of the proposed method depends on the number of primitive-sensitive clusters existing in $\mathbf{X}_q$ [see (4)]. In the experiments, to have a fair comparison, the number of hash bits for the standard single-code hashing method was selected as the average number of hash bits employed in the proposed method when querying the 2100 images. The number K of primitive clusters was tested within the range [15–102] with a step size of 3, while the value for the threshold Th was varied in the range of [0.1–0.9] with a step size increment of 0.1. Then, the best parameters were selected according to the cross-validation procedure. Specifically, the highest recall was obtained when Th $= 0.5$ and $K = 17$. From the validation assessments, we observed that given an appropriate Th value varying the number K of clusters slightly changes the average recall of the proposed method. However, this change is not statistically significant.

IV. EXPERIMENTAL RESULTS

Table I shows the average recall, average computational time, and storage complexity required for the proposed and the standard methods with 8 and 32 hash bits. From the Table I, one can see that the proposed k-NN provides higher recall than the standard k-NN at the cost of higher computational time and higher storage complexity. As an example, the improvement achieved by the proposed k-NN is 2% (see Table I). The computational and the storage complexities of the proposed k-NN are sharply reduced by the proposed multicode hashing strategy. In greater details, time reduction is more than one order of magnitude when $b = 8$, while it is half an order of magnitude when $b = 32$. Moreover, the proposed multicode hashing strategy yields an average recall 1% lower with respect to the standard k-NN, when $b = 32$, with significantly reduced retrieval time and storage complexity. Moreover, it provides 5% higher recall when $b = 8$ and more than 6% higher recall when $b = 32$ with respect to the standard single-code hashing method under the same retrieval time. By analyzing the Table I, one can also see that the storage complexity and retrieval time of single-code and proposed multicode hashing methods associated with $b = 8$ are increased when $b = 32$, while the average recall is improved. It is worth noting that the storage complexity of the proposed multicode hashing strategy is slightly higher than that of the standard single-code hashing method. This is due to storing also the hash codes of the descriptors of primitive clusters with all zero entries. Fig. 3 shows an example of images retrieved by using the proposed multicode hashing strategy and the standard single-code hashing method. The retrieval order of each image is given above the related image, while the primitive class labels associated with the image are given below the related image. By visually analyzing the results, one can observe that the proposed method leads to retrieval of semantically more similar images from the archive. As an example, the second retrieved image by the proposed method contains buildings

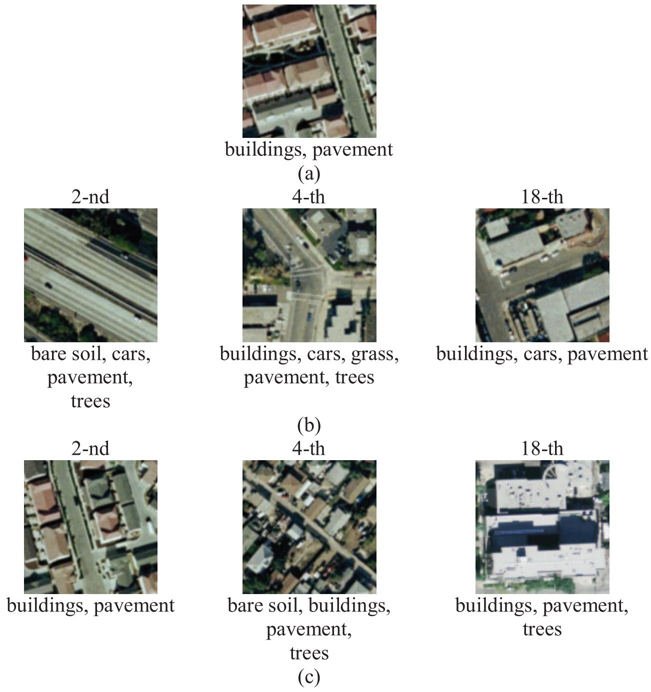

Fig. 3. (a) Query image. (b) Examples of images retrieved by the standard single-code hashing method. (c) Examples of images retrieved by the proposed multicode hashing method when $b = 32$.

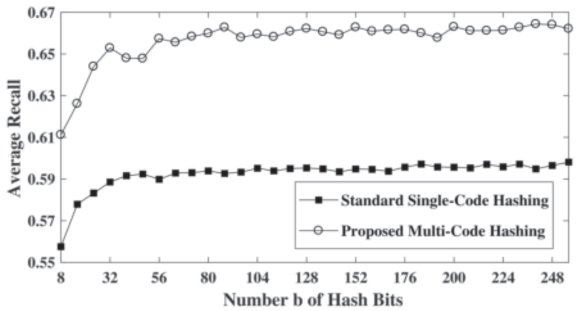

Fig. 4. Average recall versus the number of hash bits obtained by the standard single-code and the proposed multicode hashing methods.

that are present in the query image, whereas that retrieved by the standard method does not contain any building. The same relative behavior is also observed in the results obtained by many other query images. It is important to note that the proposed multicode hashing provides much higher recall than the standard single-code hashing method independently on the number b of hash bits considered (see Fig. 4). However, as expected, by increasing b both the retrieval time and the amount of memory required for storing the hash codes increase for both methods. All these results show that the proposed method is suitable to be used on real RS image retrieval scenarios where the images have highly complex semantic content.

V. Conclusion

In this letter, a novel primitive-sensitive multicode hashing method has been presented for large-scale complex CBIR problems in RS. The proposed method consists of a two-steps algorithm for learning multi-hash codes, each of which represents a primitive. The first step models images by descriptors of primitive-sensitive clusters, while the second step maps the descriptors of primitive-sensitive clusters into multi-hash codes. It is worth emphasizing that the proposed method is unsupervised, and thus it does not require any labeled annotated image to construct the hash codes. The proposed method enables a more detailed characterization of the semantic content of an image with respect to the standard single-code unsupervised hashing methods. Moreover, it is general and can be embedded in any unsupervised single-code hashing method. Our experiments have proven the effectiveness and efficiency of the proposed method: 1) to describe the complex content of RS images and 2) to achieve fast image search and retrieval with low storage complexity. As a future development of this letter, we plan to extend the proposed method to the framework of deep primitive-sensitive hashing, which can learn region features and hash codes through deep networks instead of exploiting hand-crafted features. In addition, we also plan to extend the proposed method to the framework of multicode supervised hashing.

References

[1] M. Hu, Y. Yang, F. Shen, N. Xie, and H. T. Shen, "Hashing with angular reconstructive embeddings," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 545–555, Feb. 2018.
[2] B. Demir and L. Bruzzone, "Hashing-based scalable remote sensing image search and retrieval in large archives," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 2, pp. 892–904, Feb. 2016.
[3] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1092–1104, Jun. 2012.
[4] W. Liu, J. Wang, R. Ji, Y.-G. Jiang, and S.-F. Chang, "Supervised hashing with kernels," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 2074–2081.
[5] N. Luka, B. Žalik, S. Cui, and M. Datcu, "GPU-based kernelized locality-sensitive hashing for satellite image retrieval," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Milan, Italy, Jul. 2015, pp. 1468–1471.
[6] P. Li and P. Ren, "Partial randomness hashing for large-scale remote sensing image retrieval," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 3, pp. 464–468, Mar. 2017.
[7] J. Wang, W. Liu, S. Kumar, and S.-F. Chang, "Learning to hash for indexing big data—A survey," *Proc. IEEE*, vol. 104, no. 1, pp. 34–57, Jan. 2016.
[8] T. Reato, B. Demir, and L. Bruzzone, "Primitive cluster sensitive hashing for scalable content-based image retrieval in remote sensing archives," in *Proc. Int. Conf. Geosci. Remote Sens. Symp.*, Fort Worth, TX, USA, Jul. 2017, pp. 2199–2202.
[9] B. Chaudhuri, B. Demir, S. Chaudhuri, and L. Bruzzone, "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 1144–1158, Feb. 2018.
[10] Y. Yang and S. Newsam, "Geographic image retrieval using local invariant features," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 2, pp. 818–832, Feb. 2013.