Towards Simultaneous Image Compression and Indexing for Scalable Content-Based Retrieval in Remote Sensing

Gencer Sumbul, Graduate Student Member, IEEE, Jun Xiang, and Begüm Demir, Senior Member, IEEE

Abstract—Due to the rapidly growing remote-sensing (RS) image archives, images are usually stored in a compressed format for reducing their storage sizes. Thus, most of the existing content-based RS image retrieval systems require fully decoding images (i.e., decompression) that is computationally demanding for large-scale archives. To address this issue, we introduce a novel approach devoted to simultaneous RS image compression and indexing for scalable content-based image retrieval (denoted as SCI-CBIR). The proposed SCI-CBIR prevents the requirement of decoding RS images before image search and retrieval. To this end, it includes two main steps: 1) deep-learning-based compression and 2) deep-hashing-based indexing. The first step effectively compresses RS images by employing a pair of deep encoder and decoder neural networks and an entropy model. The second step produces hash codes with a high discrimination capability for RS images by employing pairwise, bit-balancing, and classification loss functions. For the training of the SCI-CBIR approach, we also introduce a novel multistage learning procedure with automatic loss weighting techniques to characterize RS image representations that are appropriate for both RS image indexing and compression. The proposed learning procedure enables automatically weighting different loss functions considered for the proposed approach instead of computationally demanding grid search. Experimental results show the effectiveness of the proposed approach when compared to widely used approaches in RS. The code of the proposed approach is available at https://git.tu-berlin.de/rsim/SCI-CBIR.

Index Terms—Deep-learning-based compression, hashing-based indexing, image retrieval, remote sensing (RS).

I. INTRODUCTION

RECENT advances in satellite technologies lead to a significantly increased volume of remote-sensing (RS) image archives. Thus, in recent years, increasing attention has been devoted to the development of accurate and scalable content-based image retrieval (CBIR) methods for such archives [1], [2], [3]. For large-scale CBIR, fast and accurate indexing methods that allow approximate nearest neighbor search are fundamental. In this perspective, hashing-based indexing has recently attracted attention to solving the large-scale approximate nearest neighbor search problems for RS CBIR due to its high time-efficient (in terms of both storage and speed) and accurate search capability within huge image archives. Hashing methods map high-dimensional image features into compact binary hash codes [4]. Then, image retrieval can be achieved by calculating the Hamming distances with simple bitwise XOR operations [5]. Several hashing methods are presented in RS [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16]. The traditional hashing methods extract hand-crafted image features and map them into low-dimensional binary codes by using hashing functions [6], [7], [8]. In these methods, image feature extraction and hash code generation are separately applied. Thus, they are not capable of simultaneously optimizing feature learning and hash code learning, which results in the limited capability of generated hash codes to represent the high-level semantic content of RS images. Recently, several deep-hashing-based indexing methods are introduced in RS to address this issue. As an example, in [10] a deep hashing neural network (DHNN) is introduced to learn high-level semantic features and compact hash codes in an end-to-end manner. To improve the training stability of deep neural networks (DNNs) while learning hash codes, DHNN generates the continuous approximations of hash codes during training while exploiting quantization loss to push the approximated hash codes towards the discrete values. In greater detail, the likelihood pairwise loss is utilized in DHNN to preserve the similarity of images on their hash codes. However, the pairwise loss can lead similar images to cluster together in a small portion of the Hamming space that prevents generating discriminative hash codes. To avoid this problem, in [11], a deep-hashing convolutional neural network (DHCNN) is introduced to employ image labels for learning more discriminative hash codes. To this end, DHCNN learns to predict image labels together with generating hash codes by jointly optimizing cross-entropy loss with pairwise and quantization losses. Despite the success of pairwise loss in these methods, triplet loss has been found more effective than pairwise loss by introducing a margin threshold between the similar and dissimilar images [17]. Accordingly, in [12], a metric-learning-based deep hashing network (MiLaN) is introduced to combine quantization loss with triplet loss. In addition, MiLaN also employs bit-balancing loss for maximizing code variance and information by forcing each bit to have an equal chance of being 0 or 1. Unlike the
above-mentioned methods, which utilize pretrained CNNs, in [13], a semisupervised hashing adversarial autoencoder (SSHAAE) is introduced to employ an adversarial autoencoder network for generating the discriminative and similarity preserved hash codes with low quantization errors by end-to-end training. In addition to losses used by DHCNN, SSHAAE also employs bit-balancing and reconstruction losses. In [14], a generative adversarial network is exploited for hash code learning, while similar losses to DCHHN for the generator and a sigmoid function for the discriminator are used to determine whether the generated codes are true codes that comply with the bit-balancing rule. In [15], a meta-hashing algorithm is introduced to increase the generalization capability of DNNs utilized for hash code generation under a small number of training samples. To this end, this algorithm employs few-shot meta-learning for hash code generation by dividing a learning objective into multiple subtasks and using all training samples multiple times. In [16], an asymmetric hash code learning (AHCL) method is proposed to increase the training efficiency of DNNs for hash code learning. To this end, AHCL learns a deep hashing function only for query images, while hash codes of archive images are obtained from query hash codes based on class label similarity. We refer the reader to [2] for a detailed review of hashing-based indexing methods in RS.

The above-mentioned hashing-based indexing methods are potentially effective for RS CBIR. RS images are usually stored in compressed format in archives to reduce their storage sizes [18]. Thus, image decoding (i.e., decompression) is required before applying any hashing method. This is computationally demanding and impractical in the case of large-scale CBIR problems. According to our knowledge, there is no hashing-based indexing method in RS that can be applied in the compressed domain efficiently and effectively. To address this issue, in this article, we introduce a novel approach devoted to simultaneous RS image compression and indexing for scalable content-based image retrieval (denoted as SCI-CBIR). Unlike the existing CBIR approaches in RS, the proposed approach simultaneously indexes RS images with hash codes while effectively compressing them. To this end, the proposed SCI-CBIR is made up of two main steps: 1) deep-learning (DL)-based compression and 2) deep-hash-based indexing. The first step applies image feature extraction and image reconstruction based on a pair of encoder and decoder DNNs, while a probabilistic entropy model is employed to optimize the length of the compressed bitstream. The second step employs pairwise, bit-balancing, and classification loss functions for the generation of hash codes based on image features characterized by the first step. To effectively characterize image features for both image indexing and compression, we propose a novel multistage learning procedure for the training of the proposed SCI-CBIR approach, allowing to automatically weight different loss functions considered in both steps. Note that this study aims to introduce neither compression nor hashing algorithm but to propose a novel approach that simultaneously indexes and compresses RS images. Due to the proposed approach, the need for decompressing images before indexing, unlike the existing CBIR approaches in RS, is fully eliminated. The proposed approach has been briefly presented in [19] with limited experimental analysis. This article extends our work by introducing a detailed description of the proposed approach with a detailed experimental analysis of two large-scale benchmark archives. Furthermore, several new experiments are conducted and their results are commented on. The main contributions of this work are summarized as follows.

- As a first time in RS, the proposed SCI-CBIR approach simultaneously applies RS image compression and indexing and thus does not require RS image decoding before CBIR that can save a significant amount of time for operational applications.
- The proposed multistage learning procedure automatically weights all the considered loss functions that allow us to: 1) learn appropriate RS image representations for both image compression and indexing; 2) eliminate computationally demanding grid search; and 3) automatically achieve different rate-distortion tradeoff points.
- The proposed SCI-CBIR approach is independent of image compression and indexing methods being selected and can operate with any DNN-based method.

The rest of this article is organized as follows. Section II presents the related works on RS image compression and RS CBIR in the compressed domain. Section III introduces the proposed SCI-CBIR approach. Section IV describes the considered RS image archives and the experimental setup, while Section V provides the experimental results. Section VI concludes our article.

II. RELATED WORK

In this section, we survey the existing methods for RS image compression and RS CBIR on a compressed domain. Traditional RS image compression methods are categorized into three groups: 1) prediction-based methods, which predict each spectral band based on the other bands and encode the prediction residuals to bitstreams (e.g., CCDCS-123 multi- and hyperspectral image compression standard [20]); 2) vector quantization methods, which independently reduce the clusters of image pixels with similar characteristics by grouping them together (e.g., mean-normalized vector quantization [21]); and 3) transform-based methods, which map RS images to transform domain (e.g., Karhunen-Loève transform [22], discrete cosine transform [23], discrete wavelet transform [24], etc.) representations and thus reduce the correlation among image pixels. Although prediction-based compression methods apply lossless compression and embody a low computational complexity, their compression ratio is generally low, which makes them infeasible for large-scale RS archives. Vector quantization methods provide a higher compression ratio than prediction-based methods. However, training these methods and generating required codebooks can be computationally demanding. Transform-based methods generally provide a high compression ratio and speed of computation and thus are widely used for RS image compression on operational archives. Among several transform-based methods, JPEG 2000 [25] became very popular in RS due to its multiresolution paradigm, scalability, and high compression ratio. JPEG 2000 algorithm is widely used to compress...
RS images acquired by most of the recent satellites (such as Sentinel-2 [26]).

Recent studies on learning-based compression show that DL-based compression methods preserve the perceptual quality of images at lower bit rates compared to traditional methods such as JPEG2000 [27]. DL-based image compression methods usually consist of a pair of encoder and decoder DNNs for feature extraction and image reconstruction, and an entropy model for bit-rate optimization. According to the type of the DNN, recent DL-based image compression methods can be divided into one-time feed-forward and multistage recurrent-based compression methods [28]. One-time feed-forward DNNs (e.g., CNNs) employ only one time of image encoding and decoding and thus require to be trained multiple times for different bit rates. However, for multistage recurrent DNNs (e.g., recurrent neural networks), image encoding is iteratively applied, while the number of iterations determines a variable range of bit rates within a single training. In RS, few DL-based image compression methods have been proposed in the framework of a standard CNN-based image compression, where a piecewise linear approximation to the occurrences of pixel values is used as an entropy model. In [29], a residual network framework is introduced to adapt the standard CNN-based compression for multispectral RS images by characterizing RS image representations with residual blocks and a weighted feature channel module. In [27], spectral–spatial feature partitioned extraction is integrated into the standard CNN-based compression to characterize the spatial and spectral content of RS images in a parallel fashion. In [30], polydirectional CNNs are introduced in the standard CNN-based compression to separately extract the spectral and spatial RS image features for preventing the dominance of either spatial or spectral content. In the computer vision community, generalized divisive normalization [31], residual blocks [32], attention modules [33], and nonlocal networks [34] have been employed in the context of CNN-based compression to further reduce the spatial redundancy when characterizing image latent that results in lower bit rate for entropy encoding. To further improve the compression ratio, hyperpriors [35], autoregressive context models [36], and discretized Gaussian mixture likelihoods [33] are incorporated into the entropy model for more accurate bit-rate optimization. The reader is referred to [28] for recent advances in DL-based image compression.

According to our knowledge, there is only one study in RS that is devoted to applying CBIR in compressed domain [37]. To reduce the time required for fully-decoding images, in [37], a coarse to fine progressive RS image description and retrieval system in the partially decoded JPEG 2000 compressed domain is proposed. In that system, the code blocks associated only with the coarse wavelet resolution are initially decoded. Then the most irrelevant images to the query image are discarded based on the similarities computed on the coarse resolution wavelet features of the query and archive images. The processes of code blocks decoding and elimination of the irrelevant images are iterated until the codestreams associated with the highest wavelet resolution are decoded. Finally, the most similar images to the query are chosen. Although that system reduces significantly the retrieval time compared to those that require full decoding, it still requires a partial decomposition that may require significant time for operational CBIR applications.

As mentioned above, DL-based image compression methods are much more successful to preserve the perceptual quality of images at lower bit-rate values compared to JPEG2000 [27]. According to our knowledge, our SCI-CBIR approach is the first study in the framework of the scalable CBIR on the DL-based compressed domain in RS.

### III. Proposed SCI-CBIR Approach

Let \( \mathcal{X} = \{x_1, \ldots, x_M\} \) be an RS image archive that includes \( M \) uncompressed images, where \( x_i \) is the \( i \)-th image in the archive. We assume that a training set \( \mathcal{T} \subset \mathcal{X} \) is available, where \( \forall x_i \in \mathcal{T} \) is associated with a set of class labels \( I_i \in \{0, 1\}^K \) and \( K \) is the number of classes.

The proposed SCI-CBIR approach aims to achieve accurate CBIR in a scalable way without any need for decompression of RS images before CBIR. Accordingly, SCI-CBIR simultaneously: 1) compresses each image \( x_i \in \mathcal{X} \) into a bitstream and 2) indexes each image through a \( q \) bit hash code \( b_i \) (which is stored in a hash table for scalable CBIR). This is achieved based on two steps: 1) DL-based compression and 2) deep-hashing-based indexing. For the training of SCI-CBIR, we introduce a multistage learning procedure to automatically define different loss weights and rate-distortion trade-off points. Fig. 1 shows an illustration of the proposed SCI-CBIR approach, which is explained in detail in Sections III-A–III-C.

#### A. First Step: DL-Based Compression

The DL-based compression step of the proposed SCI-CBIR approach aims to compress each RS image to a minimum length bitstream, which is efficiently stored and utilized for reconstructing the image with a minimum amount of distortion. By following the recent advances in DL-based image compression, this step employs a pair of encoder–decoder DNNs for learning to reconstruct RS images and an entropy model for reducing the length of bitstreams (i.e., bit-rate optimization). Accordingly, this step includes three main blocks:
1) image encoding; 2) compression decoding; and 3) entropy modeling.

Let \( f : \mathcal{X} \mapsto \mathcal{Y} \) be an image encoder that maps the image \( x_i \) to its latent \( y_i \), where \( \mathcal{Y} \) is the set of all latents for \( \mathcal{X} \). The first block of this step transforms \( x_i \) into its quantized latent representation \( \hat{y}_i \) as follows:

\[
y_i = f(x_i; \theta_f);
\hat{y}_i = Q(y_i)
\]  

where \( Q(a) = \lfloor a \rfloor \) is a rounding function that converts \( a \) into its nearest integer (i.e., quantization) and \( \theta_f \) is the encoder parameters. During training, \( Q(a) \) is replaced by \( U(a - (1/2), a + (1/2)) \), where \( U \) is a uniform distribution. Let \( g : \mathcal{Y} \mapsto \mathcal{X} \) be a decoder that maps the latent \( y_i \) into the reconstructed image \( \hat{x}_i \), where \( \hat{X} \) is the set of reconstructed images. The second block of this step reconstructs \( x_i \) from its quantized representation as follows:

\[
\hat{x}_i = g(\hat{y}_i; \theta_g) = g(Q(f(x_i; \theta_f)); \theta_g)
\]  

where \( \theta_g \) is the decoder parameters. The third block of this step estimates the required number of bits to encode \( \hat{y}_i \), which is defined according to the mutual information between \( x_i \) and \( \hat{x}_i \). Since the actual distribution of image latents \( p_{\hat{y}} \) is unknown, its inference is intractable. Accordingly, the entropy modeling block estimates \( p_{\hat{y}} \) with an entropy model \( q_{\hat{y}} \), where \( \theta_q \) is the entropy model parameters. This block also employs arithmetic coding algorithm \( A \), which consists of arithmetic encoder \( A_e \) and arithmetic decoder \( A_d \) for generating compressed bitstreams from quantized representations.

To achieve minimum image compression distortion at a minimum length of the bitstream, the image compression objective \( L_C \) is defined according to a rate-distortion optimization problem [38] as follows:

\[
L_C = L_R + \lambda L_D
\]

\[
L_C = E_{x_i \sim p_x}[-\log(q_{\hat{y}}(\hat{y}_i))] + \lambda E_{x_i \sim p_x}[d(x_i, \hat{x}_i)]
\]  

where \( p_x \) is approximated over the images of \( T \). The rate term \( L_R \) is the cross entropy between the entropy model \( q_{\hat{y}} \) and the marginal distribution of quantized image latents \( E_{x_i \sim p_x}P_{\hat{y}} \). \( d \) is the distortion metric, for which we utilize multiscale structural similarity index (MS-SSIM) [39] as \( d(x_i, \hat{x}_i) = 1 - \text{MS-SSIM}(x_i, \hat{x}_i) \). \( \lambda \) controls the rate-distortion tradeoff points.

It is worth noting that the proposed SCI-CBIR approach is independent of the image compression method utilized in the first step of our approach as soon as it employs a pair of encoder and decoder DNNs. Recent studies on DL-based image compression have focused on enhancing the capacity of the considered entropy model for an accurate estimation of \( p_{\hat{y}} \), while operating image reconstruction based on encoder–decoder architectures. In this article, for the entropy modeling block of this step, we consider context-adaptive hyper-prior-based Gaussian mixture models introduced in [33] as the entropy model due to its proven success for spatial redundancy reduction. In this entropy model, the probability estimation of quantized image latents is conditioned on a hyper-prior (which is defined by a factorized density model) and an autoregressive context model to capture the spatial dependencies among the elements of quantized latents. The reader is referred to [33] for the details of this entropy model.

\section*{B. Second Step: Deep-Hashing-Based Indexing}

The deep-hashing-based indexing step of the proposed SCI-CBIR approach aims to map the latent representation of each RS image (which is characterized in the first step) into its discriminative hash code, which preserves the semantic image content. Then, hash codes are indexed in a hash table for all RS images in the archive, where semantically similar images are in the same hash bucket. To this end, this step includes three main blocks: 1) index decoding; 2) hash code generation; and 3) class prediction. Let \( t : \mathcal{Y}, \theta_t \mapsto \mathcal{E} \) be a decoder that maps the latent \( y_i \) into the corresponding image embedding for indexing \( e_i \) associated with the image \( x_i \) (i.e., \( t(y_i) = e_i \)), where \( \theta_t \) is the decoder parameters. The index decoding block employs \( t \) for characterizing image embeddings by extracting and decoding semantically informative features specific to indexing based on the latent representations of images. Accordingly, \( t \) is composed of the attention layer introduced in [33] followed by convolutional layers. Let \( b : \mathcal{E}, \theta_b \mapsto [-1, 1]^q \) be a binarizer that maps the image embedding \( e_i \) into the binary hash code \( b_i \) of \( x_i \), where \( \theta_b \) is the binarizer parameters. Let \( k : \mathcal{E}, \theta_k \mapsto [0, 1]^K \) be a classifier that maps the image embedding \( e_i \) into the class prediction \( \hat{I}_i \) of \( x_i \), where \( \theta_k \) is the classifier parameters. Once the image embedding \( e_i \) is characterized for \( x_i \), the class prediction and hash code generation blocks operate on \( e_i \) to generate corresponding class prediction and hash code, respectively.

To characterize discriminative hash codes that preserve the semantic similarity of images, we employ the soft pairwise loss (SPL) [40] \( L_P \), bit-balancing loss [41] \( L_B \), and a classification loss \( L_N \). SPL considers the rank difference of semantic pairwise similarities of images. To this end, image pairs are grouped into images with hard similarity and images with soft similarity. An image pair shares either no common labels or all its labels for hard similarity, while an image pair shares some of its labels for soft similarity. Let \( J = \{(x_i, x_j) | x_i \in T, x_j \in T, i \neq j \} \) be a set of all image pairs in \( T \). The SPL function is defined as follows:

\[
L_P = \sum_{(x_i, x_j) \in J} m_{ij} \left[ \log(1 + e^{sh_{ij}}) - s_{ij}^o \right] \\
+ \gamma (1 - m_{ij}) \left[ \frac{1}{2} (s_{ij}^q + q) - s_{ij}^q q \right]
\]

\[
s_{ij}^q = \frac{\langle I_i, I_j \rangle}{\|I_i\|_2 \|I_j\|_2}, \quad s_{ij}^o = \langle b_i, b_j \rangle
\]

\[
m_{ij} = \begin{cases} 
1, & s_{ij}^o \in [0, 1] \\
0, & 0 < s_{ij} < 1
\end{cases}
\]  

where \( s_{ij}^o \) and \( s_{ij}^q \) are the pairwise similarities between \( x_i \) and \( x_j \) and their hash codes, respectively. \( m_{ij} \) defines whether \( (x_i, x_j) \) is associated with soft similarity (\( m_{ij} = 0 \)) or hard similarity (\( m_{ij} = 1 \)). \( \gamma \) is a weighting parameter between different types of similarities. For balancing the distribution of hash code bits by maximizing their variance, we adapt the
bit-balancing loss [41] for image pairs as follows:

$$L_B = \sum_{(x_i,x_j) \in J} \| (b_i^T 1) \|^2_2 + \| (b_j^T 1) \|^2_2$$  \hspace{1cm} (5)

where $1$ is a vector with all elements 1. $L_B$ enforces the hash codes to contain the equal numbers of $-1$ and 1. To further enhance the discriminative capability of hash codes, we formulate the classification loss over image pairs as follows:

$$L_N = \sum_{(x_i,x_j) \in J} \| \hat{l}_i - l_i \|^2_2 + \| \hat{l}_j - l_j \|^2_2.$$  \hspace{1cm} (6)

By considering the above-mentioned losses defined for the first step of our SCI-CBIR approach, the final hashing objective is formulated as follows:

$$L_H = w_P L_P + w_B L_B + w_N L_N$$  \hspace{1cm} (7)

where $w_P, w_B, w_N$ are the loss weights. We note that the proposed SCI-CBIR approach is independent of the DNN architecture utilized in this step, and thus hash codes can be obtained through different DNN architectures.

C. Proposed Multistage Learning Procedure

The objectives of both steps of the proposed approach $L_C, L_H$ enforce to encode different information through the image encoder $f$ on image latents $Y$. The first step enforces image latents to embody maximum information required for reconstructing images, while the second step enforces image latents to preserve the most discriminative image features for hash code learning. For the training of the proposed SCI-CBIR, one could optimize the aggregation of different losses considered for both steps in a single learning procedure that is widely utilized for combining different objectives in DL. However, due to different characteristics of image compression and indexing tasks, this learning procedure may lead to: 1) the competition of the learning objectives of image compression and indexing tasks; 2) the dominance of one of the objectives; and 3) limited characterization of each task compared to separately learning each objective. In this case, either multiple instances of the considered DNN need to be trained with different $\lambda$ values or recurrent models need to be integrated to achieve a variable range of rate-distortion tradeoff points [28]. Accordingly, to prevent this limitation, we propose a multistage learning procedure for the training of our SCI-CBIR that aims to: 1) learn RS image latents compatible for both RS image compression and indexing; 2) automatically weights different losses; and 3) automatically achieve different rate-distortion tradeoff points for compression without applying computationally demanding grid search of $\lambda$. To this end, the proposed learning procedure is made up of three consecutive stages: 1) learning reconstruction; 2) bit-rate optimization; and 3) learning hash codes.

1) Learning Reconstruction: The compression objective in (3) involves the conflict of bit rate and distortion terms that leads to decreasing bit-rate term increases the distortion term, and vice versa (i.e., rate-distortion tradeoff). Accordingly, to achieve an effective reconstruction capability without being affected by the rate-distortion tradeoff, in the first stage, only the distortion loss $L_D$ is optimized until its convergence with a learning rate $\eta_1$, which is gradually decreased based on the value of $L_D$.

2) Bit-Rate Optimization: To accurately achieve different rate-distortion points, in the second stage, $L_D$ is continued to optimize together with a bit-rate loss $L_R$ with a learning rate $\eta_2$. Most of the existing DL-based image compression methods require multiple trainings with different $\lambda$ values to achieve different tradeoff points for (3). Unlike them, in this stage, we reformulate (3) as a multiobjective optimization problem and employ a multiple-gradient descent algorithm (MGDA) [42] for automatically achieving the set of optimal tradeoffs points as the set of Pareto optimal solutions. Let $g_D = \nabla_{\theta_D} L_D$ and $g_R = \nabla_{\theta_R} L_R$ be the gradient vectors of $L_D$ and $L_R$, respectively, over the parameters $\theta_C = \theta_f \cup \theta_D \cup \theta_R$. The gradient descent direction for a Pareto optimal solution (which leads to an optimal tradeoff point) is obtained by optimizing the following problem:

$$\min \{ \| u \|^2_2 | u = w_D g_D + w_R g_R, w_D + w_R = 1 \}$$

where $w_D$ and $w_R$ are estimated as follows:

$$w_D = \begin{cases} 1, & g^T_D g_R \geq g^T_D g_D \\ 0, & \| g_R - g_D \|^2_2, \text{ otherwise} \end{cases}$$  \hspace{1cm} (9)

After obtaining $u$ by solving (8), the parameters $\theta_C$ are updated as follows:

$$\theta_C = \theta_C - \eta_2 u = \theta_C - \eta_2 (w_D g_D + w_R g_R).$$  \hspace{1cm} (10)

Since the distortion loss is converged in the learning reconstruction stage, $w_D \approx 1$ and $w_R \approx 0$ at the beginning of this stage. As this stage continues, $L_R$ decreases until the first Pareto solution is found by (9). Then, by increasing $\eta_2, w_R$ is gradually increased to reach another Pareto solution. Thus, by adjusting the learning rate itself, this stage allows obtaining the set of optimal rate-distortion tradeoff points without operating multiple trainings and applying computationally demanding grid search of $\lambda$.

3) Learning Hash Codes: The last stage involves optimizing all the losses associated with both steps of our approach to learning RS image latents compatible with both RS image indexing and compression. To this end, this stage employs two learning rates $\eta_3^C$ and $\eta_3^R$ for the losses of the first and second steps, respectively. It is worth noting that since the losses $L_D$ and $L_R$ are optimized in the first two stages, we keep $\eta_3^C < \eta_3^R$ to prevent the domination of image compression over image indexing. Since the different rate-distortion points are achieved in the second stage, the overall objective is written for a rate-distortion point as follows:

$$L = L_C + L_H$$
$$L_C = w_D L_D + w_R L_R$$
$$L_H = w_P L_P + w_B L_B + w_N L_N$$  \hspace{1cm} (11)
where $w_D$ and $w_R$ are estimated for the specific rate-distortion point in the previous step. To automatically find the weights $w_p, w_b, w_N$ instead of time demanding grid search, we utilize automatic loss weighting techniques. Accordingly, the update rules for the SCI-CBIR are written as follows:

$$\theta_C = \theta_C - \eta^C_2 (\nabla_{\theta_C} L_C + \nabla_{\theta_b} L_H)$$
$$\theta_H = \theta_H - \eta^H_2 \nabla_{\theta_H} L_H$$

(12)

where $\theta_H = \theta_1 \cup \theta_2 \cup \theta_3$.

IV. DATASET DESCRIPTION AND EXPERIMENTAL SETUP

A. Dataset Description

To evaluate the proposed approach, experiments were conducted on two RS image archives: 1) BigEarthNet-S2 [43] and 2) MLRSNet [44]. BigEarthNet-S2 is a large-scale multilabel benchmark archive that consists of 590326 Sentinel-2 images across ten different European countries. In the experiments, we considered a subset of BigEarthNet-S2 acquired over Serbia and summer season that includes 14832 images, each of which is made up of 120 × 120 pixels for 10-m bands, 60 × 60 pixels for 20-m bands, and 20 × 120 pixels for 60 m bands. In the experiments, cubic interpolation was applied to 20- and 60-m bands that leads to 120 × 120 pixels for each band. Each image is annotated with multilabels provided by the CORINE Land Cover Map (CLC) database of the year 2018. In the experiments, the 19 class nomenclature introduced in [43] was exploited. MLRSNet is a multilabel RS image archive that contains 109161 images selected from aerial orthoimagery with varying spatial resolutions from 10 to 0.1 m. For the experiments, we randomly selected a subset of MLRSNet that consists of 15302 images, each of which has the size of 256 × 256 pixels. Each image is annotated with multilabels from 60 classes. For the experiments, we divided BigEarthNet-S2 and MLRSNet into training, validation, and test sets with ratios of 50%, 25%, 25%, and 50%, 10%, 40%, respectively. To apply CBIR, we selected queries from the validation set, while images were retrieved from the test set of each archive.

B. Experimental Setup

For the first step of the proposed approach, we utilize the autoencoder DNN architecture presented in [33]. The indexing decoder within the second step of proposed SCI-CBIR includes the attention layer from [33] followed by two convolutional layers, each of which includes 512 hidden units with ReLU activation function, while their filter sizes are $5 \times 5$ and $3 \times 3$. The class prediction and hash code generation blocks of the second step include single convolutional layers with the filter size of $1 \times 1$. We tested different activation functions for the hash code generation block among sigmoid, tanh, softsign [45], and greedy hash [46] functions. The parameter $\gamma$ was set to 0.1/q, while the hash code length $q$ was varied as $q = 16, 32, 64$. The mini-batch size was selected as 32 for both archives. While training the second step, horizontal and vertical flipping were randomly applied to the training set. We trained the proposed approach by using a stochastic gradient descent algorithm.

As discussed in Section III-C, the training of the proposed approach is divided into three stages. In the first stage, the first step of the proposed approach was optimized for the distortion loss only and $\eta_1$ was updated according to the MS-SSIM value averaged on the validation set $V$ as follows:

$$\eta_1 = \begin{cases} 
10^{-4}, & \text{MS-SSIM}(V, \hat{V}) < 24 \\
5 \times 10^{-5}, & 24 \leq \text{MS-SSIM}(V, \hat{V}) \leq 29 \\
10^{-5}, & \text{MS-SSIM}(V, \hat{V}) > 29.
\end{cases}$$

(13)

The second stage starts when the distortion loss value reaches its convergence. The learning rate $\eta_2$ was set to $10^{-5}$ at the beginning of the stage. After the first Pareto point was obtained, $\eta_2$ is increased to $9 \times 10^{-5}$. In the third stage, the second step of the proposed approach was jointly trained with the first step, while the learning rate $\eta_3^H$ was set to $10^{-4}$. $\eta_3^C$ was varied as $\eta_3^C = 0, 10^{-8}, 10^{-4}$, while automatic loss weighting technique was varied among projecting conflicting gradients (PCGrad) [47], dynamic weight average (DWA) [48], and equal weighting. All the experiments were conducted on NVIDIA Tesla V100 GPUs. Experimental results were provided in terms of MS-SSIM and bit rate (bpp) for compression performances, while precision ($P$), recall ($R$), mean average precision (mAP), and retrieval time were used for comparing retrieval performances. It is worth noting that we mapped MS-SSIM values into decibel (dB) scale as suggested in [33]. The retrieval metrics $P$, $R$, and mAP were averaged on the 15 most similar images.

We conducted experiments to: 1) perform a sensitivity analysis and 2) compare the proposed SCI-CBIR approach with standard approaches. In detail, we compare the results of the first step of SCI-CBIR with those obtained by applying image compression with a recurrent neural network (denoted as IC-RNN) [49] and JPEG 2000 [25]. We compare the results of the second step of SCI-CBIR with those obtained by the second step of our approach trained on fully decompressed data (denoted as SI-CBIR). We compare the results of the proposed SCI-CBIR trained by using our multistage learning procedure with those trained by using a standard learning procedure. For IC-RNN, we utilized MS-SSIM as the distortion measure and updated the learning rate using (13). It was trained with six RNN iterations for 280 epochs. For SI-CBIR, we trained the second step of our approach followed by the image encoder of the first step with the same hyperparameters and the loss functions $L_P, L_B$, and $L_N$. SI-CBIR is not capable of simultaneous compression and indexing and thus requires decoding before indexing. For the standard learning procedure, we jointly trained all the losses required for compression and indexing in a single learning procedure. For the loss weights, we varied the weight of the distortion loss $L_P$ and kept the rest equal to the control rate-distortion tradeoff.

V. EXPERIMENTAL RESULTS

A. Sensitivity Analysis of the Proposed SCI-CBIR Approach

In this section, the results of the sensitivity analysis for the proposed SCI-CBIR approach are presented in terms of: 1) different values of the learning rate $\eta_3^C$; 2) the effectiveness of the attention layer applied in the second step; 3) different
activation functions of the hash code generation block within the second step; 4) different automatic loss weighting techniques applied in the third stage of our multistage learning procedure; and 5) different values of \( q \). It is worth noting that during the sensitivity analysis, we set default values for the following hyperparameters: 1) \( q = 64 \) and 2) the bpp value as 0.63 and 0.33 on BigEarthNet-S2 and MLRSNet, respectively, for the first two stages of our learning procedure. We also set PCGrad as the default automatic loss weighting technique and Greedy hash as the default activation function.

In the first set of trials, we analyzed the effect of the learning rate \( \eta^3 \) (which is utilized in the third stage of the proposed multistage learning procedure). Table I shows the corresponding results for the BigEarthNet-S2 archive when different values of \( \eta^3 \) are used and the first two stages of our learning procedure are achieved at different bpp values. By analyzing the table, one can observe that using a higher value of \( \eta^3 \) as \( 10^{-8} \) leads to a significant reduction in compression results while providing the highest retrieval scores. One can see from the table that when the first step of our approach is not optimized (\( \eta^1 = 0 \)), our approach achieves the lowest retrieval scores compared to using \( \eta^1 > 0 \). However, when \( \eta^1 \) is set a small value higher than zero (\( \eta^1 = 10^{-8} \)), the proposed SCI-CBIR approach achieves comparable compression and retrieval performances. This shows that our approach is capable of simultaneously learning image representations for both indexing and compression in the third stage of our multistage learning procedure when \( \eta^1 \) is properly set. Accordingly, we set \( \eta^3 \) to \( 10^{-8} \) for the rest of the experiments. We observed a similar effect of \( \eta^3 \) for the MLRSNet archive (not reported for space constraints).

In the second set of experiments, we assessed the effectiveness of the attention layer, which is used in the second step of our approach. Table II shows the retrieval results obtained by the proposed SCI-CBIR approach with and without the attention layer for BigEarthNet-S2 when the different bpp values are achieved in the first two stages of our learning procedure. From the table, one can see that the overall scores obtained with the attention layer are significantly higher than those without the attention layer independent of the bpp values. This shows the effectiveness of the attention layer that increases the capability of our approach to accurately decode image representations for indexing in the second step and thus to learn discriminative hash codes. Similar behavior of the attention layer has been observed for the MLRSNet archive (not reported for space constraints).

In the third set of trials, we analyzed the effect of different activation functions of the hash code generation block. Table III shows the corresponding retrieval results for BigEarthNet-S2. One can observe from the table that using the Greedy hash activation function achieves the highest precision and MAP scores with comparable recall scores. It is due to the fact that the Greedy hash function does not require applying the quantization loss on the discrete hash codes. Accordingly, this function minimizes the quantization error compared to other activation functions [46]. Thus, we set Greedy hash as the activation function for the rest of the experiments. We observed similar behavior for the MLRSNet archive (not reported for space constraints).

In the fourth set of experiments, we assessed the effect of different automatic loss weighting techniques (which are applied in the third stage of our learning procedure) on retrieval performance. Table IV shows the corresponding
Table IV

| Automatic Loss Weighing Technique | P (%) | R (%) | MAP (%) |
|----------------------------------|-------|-------|---------|
| DWA [48]                         | 73.3  | 69.7  | 72.9    |
| PCGrad [47]                      | 74.2  | 70.1  | 74.1    |
| Equal Weighing                   | 73.3  | 69.3  | 73.2    |

Table V

| q    | BigEarthNet-S2 | MLRSNet |
|------|----------------|---------|
|      | P (%) | R (%) | MAP (%) | P (%) | R (%) | MAP (%) |
| 16   | 72.2  | 69.0  | 70.6    | 46.7  | 45.0  | 44.7    |
| 32   | 72.5  | 70.9  | 72.6    | 57.7  | 57.3  | 56.5    |
| 64   | 74.2  | 70.1  | 74.1    | 60.6  | 59.8  | 58.9    |

retrieval performances for the BigEarthNet-S2 archive. From the table, one can observe that the proposed SCI-CBIR approach achieves the highest scores when PCGrad is chosen as the automatic loss weighting technique. When DWA and the equal weighting technique (which equally weights different losses) are used, SCI-CBIR leads to similar retrieval performances. It is worth noting that the hashing objective in (7) is made up of different types of losses. PCGrad is capable of projecting the gradient of a loss function onto the normal plane of the gradient of another loss function. This reduces gradient interference among different loss functions that allow for more effective optimization of hashing objective compared to DWA. Accordingly, for the rest of the experiments, we utilized PCGrad as the automatic loss weighting technique applied in the third stage of our learning procedure. Similar behavior of these techniques on our approach has been observed for the MLRSNet archive.

In the fifth set of trials, we analyzed the effect of hash code length. Table V shows the corresponding retrieval performances at different values of $q$ for BigEarthNet-S2 and MLRSNet archives. One can observe from the table that by increasing $q$, most of the metric values monotonically increase for both archives. Accordingly, the proposed SCI-CBIR achieves the highest scores under all the metrics when $q = 64$ compared to other values of $q$. As an example, the proposed SCI-CBIR with $q = 64$ achieves almost 14% higher precision and 15% higher recall compared to SCI-CBIR with $q = 16$ for the MLRSNet archive. Thus, for the rest of the experiments, we set $q$ as 64.

B. Comparison With Standard Approaches

In this section, we compare the performance of the first and second steps of our approach and our multistage learning procedure with the standard approaches. Accordingly, we evaluated the effectiveness of: 1) the first step compared to JPEG2000 [25] and IC-RNN [49]; 2) the second step compared to SI-CBIR; and 3) the multistage learning procedure compared to the standard learning procedure.

In the first set of trials, we compare the DL-based compression step of our approach with JPEG2000 and IC-RNN. Fig. 2 shows the compression results at different bpp values for BigEarthNet-S2 and MLRSNet archives. By assessing the figure, one can observe that our SCI-CBIR approach achieves the highest MS-SSIM at each bpp value for both archives. This shows that the first step of our approach is capable of effectively decoding RS images with varying rate-distortion points, while RS image compression and indexing are simultaneously learned in our approach. In greater detail, the proposed SCI-CBIR approach and IC-RNN significantly outperform the JPEG2000 algorithm. This shows the effectiveness of DL-based compression compared to conventional methods for RS images. Figs. 3 and 4 show an example of reconstructed
Fig. 5. MAP versus bpp obtained by the proposed SCI-CBIR approach and SI-CBIR for (a) BigEarthNet-S2 and (b) MLRSNet archives.

RS images after they are compressed by proposed SCI-CBIR, IC-RNN, and JPEG2000 for the BigEarthNet-S2 and MLRSNet archives, respectively. One can see from the figures that the proposed SCI-CBIR approach is as capable as IC-RNN for reconstructing images without significant loss of spatial information. When compared to JPEG2000, our approach provides higher reconstruction quality. As an example, when JPEG2000 is utilized to compress the original image given in Fig. 3(a) at 0.7 bpp, it is not able to reconstruct the spatial details of the original image [see Fig. 3(b)] in contrast to our approach.

In the second set of experiments, we assessed the effectiveness of the deep-hashing-based indexing step of our approach compared to SI-CBIR. Fig. 5 shows the corresponding CBIR results for both archives when the first step of our approach was used to decode RS images at different bpp values for SI-CBIR. By analyzing the figure, one can see that the proposed SCI-CBIR approach achieves similar CBIR performance compared to SI-CBIR under different bpp values. In greater detail, one can also observe that the CBIR performance of our approach is not significantly affected by the changes in bpp values. This shows that when compression and indexing are simultaneously learned, the proposed SCI-CBIR approach is capable of indexing RS images as accurately as without learning image compression during training as in SI-CBIR. Figs. 6 and 7 show an example of RS images retrieved by both approaches for the BigEarthNet-S2 and MLRSNet archives, respectively. One can see from the figures that the proposed SCI-CBIR approach retrieves similar images to the query images compared to SI-CBIR independent of the bpp values. This is in line with our conclusion from Fig. 5. Table VI shows the required CBIR time for both approaches. It can be seen from the table that the required retrieval time per image of proposed SCI-CBIR is almost one-tenth of the time for both archives compared to SI-CBIR under similar CBIR scores. This is due to the fact that the retrieval time of SI-CBIR includes also the image decoding time, which is not required for the proposed SCI-CBIR approach. In detail, since RS image compression and indexing are simultaneously learned by our approach during training, hash codes (which are generated by our deep-hashing-based indexing step) are
TABLE VII
RESULTS OBTAINED BY PROPOSED SCI-CBIR TRAINED WITH OUR MULTISTAGE LEARNING PROCEDURE AND STANDARD LEARNING PROCEDURE ASSOCIATED WITH SIMILAR BIT RATES (BIGEARTHNET-S2 ARCHIVE)

| Our Multi-Stage Learning Procedure | Standard Learning Procedure |
|-----------------------------------|-----------------------------|
| MS-SSIM (dB) | bpp | P (%) | R (%) | MAP (%) | λ | MS-SSIM (dB) | bpp | P (%) | R (%) | MAP (%) |
| 26.7 | 0.62 | 74.2 | 70.1 | 74.1 | 150 | 22.3 | 0.63 | 70.8 | 67.6 | 70.2 |
| 27.9 | 0.78 | 74.5 | 70.0 | 74.3 | 200 | 23.0 | 0.71 | 70.8 | 67.8 | 70.3 |
| 29.0 | 0.94 | 73.9 | 70.0 | 73.7 | 500 | 26.3 | 0.87 | 70.1 | 68.0 | 70.0 |
| 29.5 | 1.05 | 73.8 | 69.5 | 73.4 | 700 | 26.9 | 1.08 | 70.6 | 68.2 | 70.0 |
| 30.3 | 1.34 | 74.2 | 69.7 | 73.8 | 1000 | 27.6 | 1.29 | 70.0 | 68.0 | 69.3 |
| 30.5 | 1.38 | 73.8 | 69.9 | 73.4 | 1250 | 27.9 | 1.44 | 70.5 | 67.9 | 70.1 |
| 30.8 | 1.56 | 73.8 | 69.9 | 73.5 | 1500 | 28.0 | 1.64 | 70.1 | 67.2 | 69.6 |

Fig. 7. (a) Query image, and images retrieved by (b) SI-CBIR, (c) proposed SCI-CBIR at 0.33 bpp, (d) proposed SCI-CBIR at 0.56 bpp, (e) proposed SCI-CBIR at 0.69 bpp, and (f) proposed SCI-CBIR at 0.85 bpp (MLRSNet archive).

directly utilized for CBIR without any need for decompressing RS images. Due to this, during large-scale RS image indexing, the proposed SCI-CBIR approach saves the significant amount of time required for computationally demanding decompression of images.

In the third set of trials, we analyzed the effectiveness of the proposed multistage learning procedure by comparing it with the standard learning procedure. Table VII shows the compression and retrieval results obtained by the proposed SCI-CBIR approach trained with the proposed multistage and standard learning procedures for the BigEarthNet-S2 archive. By assessing the table, one can see that the proposed SCI-CBIR approach with our multistage procedure provides higher scores of CBIR metrics and MS-SSIM values compared to SCI-CBIR with the standard learning procedure at similar bpp values. This is due to the fact that when a single learning procedure with equal loss weights is utilized as in standard learning procedure, learning objectives for indexing and compression are conflicting with each other independent of the different rate-distortion tradeoff points (which is controlled by $\lambda$ in the standard learning procedure). This prevents accurately learning RS image compression together with RS image indexing. Unlike the standard learning procedure, due to the proposed multistage learning procedure, our approach is capable of simultaneously learning both tasks in an effective way by automatically: 1) weighting different loss functions; and 2) finding rate-distortion tradeoff points. Similar behavior of the proposed multistage learning procedure has been observed for the MLRSNet archive (not reported for space constraints).

VI. CONCLUSION

This article introduces a novel approach (denoted as SCI-CBIR) to simultaneously compress and index RS images for scalable CBIR. The SCI-CBIR approach is characterized by two steps that are simultaneously applied based on a novel multistage learning procedure. The first step is the DL-based compression step, where RS images are first mapped into their latent representations and then reconstructed back from the latents by exploiting a pair of encoder and decoder DNNs. An entropy model is utilized to generate bitstreams for a rate-distortion tradeoff point. The second step is the deep-hashing-based indexing step, where hash codes of RS images are generated from their latent representations. With the proposed multistage learning procedure, all the parameters of SCI-CBIR are learned within three consecutive stages:
1) minimizing a distortion loss to model reconstruction; 2) finding the Pareto optimal solutions of a multiobjective optimization problem to achieve a variable range of bit rates; and 3) minimizing soft pairwise, bit balancing, and classification losses with automatic loss weighting techniques to characterize hash codes. This allows the proposed SCI-CBIR approach to: 1) obtain different bit rates without a need for training the considered DNN multiple times and 2) automatically find the weights for the five different losses considered in both steps without any need for a computationally expensive grid search.

Experimental results obtained on two benchmark archives show that the proposed approach provides high compression performance, while resulting in high retrieval accuracy without any need for decompressing the images before the indexing (which is required for most of the CBIR systems in RS). We underline that this is a very important advantage, particularly for large-scale CBIR and thus the proposed approach is convenient for possible operational applications. It is worth noting that the archives used in our experiments are benchmarks. However, in many real applications, we expect that the CBIR is applied to much larger archives. For large-scale CBIR, by using our approach, the gain in retrieval time is expected to be increased significantly compared to the existing approaches. In the case of compressing and indexing very large size RS image scenes, we suggest utilizing lightweight DNNs (such as Zoom-In [50] and ESPNetv2 [51]) that allow applying training and inference of our approach in a computationally efficient manner.

It is worth noting that the proposed approach can be easily adapted to the CBIR problems for which: 1) images are compressed by other DL-based compression algorithms; and also 2) hash codes are obtained through different DL-based architectures. As a final remark, we would like to point out that the development of DL-based image compression methods is becoming a more and more important topic. In this context, the proposed approach is very promising as it allows RS CBIR for the case that images are compressed by using DNNs. As a future development, we plan to study the development of DL-based 3-D compression models were not only spatial, but also spectral redundancies are compressed. Moreover, we plan to explore RS CBIR in the 3-D compressed domain, which is expected to be particularly relevant for search and retrieval from hyperspectral image archives.

ACKNOWLEDGMENT

The authors thank Nimisha Thekke Madam for the valuable discussions on image compression and indexing.

REFERENCES

[1] Y. Li, J. Ma, and Y. Zhang, “Image retrieval from remote sensing big data: A survey,” Inf. Fusion, vol. 67, pp. 94–115, Mar. 2021.
[2] G. Sumbul, J. Kang, and B. Demir, “Deep learning for image search and retrieval in large remote sensing archives,” in Deep Learning for the Earth Sciences: A Comprehensive Approach to Remote Sensing, Climate Science and Geosciences. Hoboken, NJ, USA: Wiley, 2021, ch. 11, pp. 150–160.
[3] G. Sumbul and B. Demir, “Plasticity-stability preserving multi-task learning for remote sensing image retrieval,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–16, 2022.
[4] J. Lin, Z. Li, and J. Tang, “Discriminative deep hashing for scalable face image retrieval,” in Proc. Int. Joint Conf. Artif. Intell., 2017, pp. 2266–2272.
[5] Z. Li, J. Tang, L. Zhang, and J. Yang, “Weakly-supervised semantic guided hashing for social image retrieval,” Int. J. Comput. Vis., vol. 128, no. 8, pp. 2265–2278, 2020.
[6] 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, Sep. 2016.
[7] 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.
[8] T. Reato, B. Demir, and L. Bruzzone, “An unsupervised multicode hashing method for accurate and scalable remote sensing image retrieval,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 2, pp. 776–786, Feb. 2019.
[9] Y. Li, Y. Zhang, X. Huang, and J. Ma, “Learning source-invariant deep hashing convolutional neural networks for cross-source remote sensing image retrieval,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 11, pp. 6521–6536, Nov. 2018.
[10] Y. Li, Y. Zhang, X. Zhu, H. Zhu, and J. Ma, “Large-scale remote sensing image retrieval by deep hashing neural networks,” IEEE Trans. Geosci. Remote Sens., vol. 58, no. 6, pp. 2857–2869, Jun. 2020.
[11] W. Song, S. Li, and J. A. Benediktsson, “Deep hashing learning for visual and semantic retrieval of remote sensing images,” IEEE Trans. Geosci. Remote Sens., vol. 59, pp. 9661–9672, 2021.
[12] S. Roy, E. Sangineto, B. Demir, and N. Sebe, “Metric-learning-based deep hashing network for content-based retrieval of remote sensing images,” IEEE Geosci. Remote Sens. Lett., vol. 18, no. 2, pp. 226–230, Feb. 2021.
[13] X. Tang, C. Liu, X. Zhang, J. Ma, C. Jiao, and L. Jiao, “Remote sensing image retrieval based on semi-supervised deep hashing learning,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., Jul. 2019, pp. 879–882.
[14] C. Liu, J. Ma, X. Tang, X. Zhang, and L. Jiao, “Adversarial hash-code learning for remote sensing image retrieval,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., Jul. 2019, pp. 4324–4327.
[15] X. Tang et al., “Meta-hashing for remote sensing image retrieval,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–11, 2022.
[16] W. Song, Z. Gao, R. Dian, P. Ghamsiri, Y. Zhang, and J. A. Benediktsson, “Asymmetric hash code learning for remote sensing image retrieval,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–14, 2022.
[17] G. Sumbul, M. Ravanbakhsh, and B. Demir, “Informative and representative triplet selection for multilabel remote sensing image retrieval,” IEEE Trans. Geosci. Remote Sens., vol. 60, pp. 1–11, 2022.
[18] H. Kramer, Observation of the Earth and Its Environment: Survey of Missions and Sensors, Berlin, Germany: Springer, 2019.
[19] G. Sumbul, J. Xiang, N. T. Madam, and B. Demir, “A novel framework to jointly compress and index remote sensing images for efficient content-based retrieval,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., Jan. 2022, pp. 1–4.
[20] E. Augé, J. E. Sánchez, A. Kiely, I. Blanes, and J. Serra-Sagristá, “Performance impact of parameter tuning on the CCSDS-123 lossless multi- and hyperspectral image compression standard,” J. Appl. Remote Sens., vol. 7, no. 1, pp. 1–16, 2013.
[21] M. J. Ryan and J. F. Arnold, “The lossless compression of AVIRIS images by vector quantization,” IEEE Trans. Geosci. Remote Sens., vol. 35, no. 3, pp. 546–550, May 1997.
[22] P. Hao and Q. Shi, “Reversible integer KLT for progressive-to-lossless compression of multiple component images,” in Proc. Int. Conf. Image Process., vol. 1, 2003, pp. 1–633.
[23] G. P. Abousleman, M. W. Marcellin, and B. R. Hunt, “Compression of hyperspectral imagery using the 3-D DCT and hybrid DPCM/DCT,” IEEE Trans. Geosci. Remote Sens., vol. 33, no. 1, pp. 26–34, Jan. 1995.
[24] W. Sweldens, “The lifting scheme: A custom-design construction of biorthogonal wavelets,” Appl. Comput. Harmon. Anal., vol. 3, no. 2, pp. 186–200, Apr. 1996.
[25] A. Skodras, C. Christopoulos, and T. Ebrahimi, “The JPEG 2000 still image compression standard,” IEEE Signal Process. Mag., vol. 18, no. 5, pp. 36–58, Sep. 2001.
...