Toward Multilabel Image Retrieval for Remote Sensing

Toward Multilabel Image Retrieval for Remote Sensing

Raffaele Imbriaco, Clint Sebastian, Egor Bondarev, and Peter H. N. de With, *Fellow, IEEE*

***Abstract*— The availability of large-scale remote sensing (RS) data facilitates a wide range of applications, such as disaster management and urban planning. An approach for such problems is image retrieval, where, given a query image, the goal is to find the most relevant match from a database. Most RS literature has been focused on single-label retrieval, where we assume an image has a single label. The primary challenge in single-label RS retrieval is that performance in most datasets is saturated, and it has become difficult to compare the performance of different methods. In this work, we extend the major multilabel classification datasets to the multilabel retrieval problem. We also define protocols, provide evaluation metrics, and study the impact of commonly used loss functions and reranking methods for multilabel retrieval. To this end, a novel multilabel loss function and a reranking technique are proposed, which circumvent the challenges present in conventional single-label image retrieval. The developed loss function considers both class and feature similarity. The proposed reranking technique achieves high performance with computation cost that is well-suited for fast online retrieval.**

***Index Terms*— Image retrieval, loss function, multilabel, query expansion, remote sensing (RS).**

I. Introduction

THE increasing availability of large remote sensing (RS) image collections has greatly contributed to explorations in the image retrieval domain. Given a query image, an image database should be sorted based on its similarity or relevance to the content of the query. A large body of work exists, which proposes solutions to this problem in the context of RS. Conventional systems rely on handcrafted features [1], [2], but more modern alternatives deploy convolutional neural networks (CNNs), to produce compact representations that encode the semantic content of the images [3]–[6]. In these systems, the CNNs act as feature extraction producing highly descriptive vectors from the images. More advanced systems exploit either local features [4], [7]–[9] or more sophisticated metric learning techniques [6], [10]–[12]. However, it seems that the datasets and the benchmark protocols have not evolved as rapidly as the retrieval systems themselves. The latter became significantly more powerful, where even several different systems saturate the performance on popular benchmarks. This complicates the direct comparison between systems since the advantages of one solution over another become hard to evaluate. Furthermore, various works report on different evaluation metrics and inconsistent dataset splits. Therefore, a fair comparison between different methods becomes even more challenging.

Another challenge is that the studied datasets may be insufficiently large to properly assess how solutions would behave in large-scale scenarios. The datasets frequently studied in the literature range from 2100 to 31 500 images [13], [14]. In addition, the focus has been centered around single-label retrieval, with few exceptions for multilabel retrieval [15]–[17]. This is an additional simplification of the image retrieval problem in the context of RS imagery. Considering the nature of the images, different semantic classes typically occur within a single picture. Therefore, to foster research on multilabel retrieval, we explore the challenges with single-label retrieval and extend the existing datasets to multilabel problems. This extension of the RS image retrieval problem not only provides a challenging definition of the retrieval problem but also remains useful for applications, such as urban planning, agricultural management, and crisis aversion. Furthermore, multilabel retrieval can be useful for difficult problems, such as cross-domain retrieval, where finer labels provide improved results [18]. Compared to single-label retrieval, multilabel retrieval enables more accurate results that are representative of a query image. In this work, several datasets of various sizes and labels are explored such that it is easier to verify the strength of a method. To this end, various evaluation metrics are also investigated, which are suited for multilabel retrieval. We propose standardized protocols to assess retrieval performance. To establish baselines, we study the existing loss functions and reranking methods commonly used in the RS literature. Finally, a novel multilabel loss function and a reranking method are proposed. Our contributions can be summarized as follows.

1) A new set of annotations and evaluation protocols for multilabel image retrieval is generated on four different multilabel datasets. Our protocols provide a clear definition of challenging examples and easy ones.
2) New baselines and benchmark of several existing loss functions and reranking techniques for multilabel retrieval are established. Since most metric learning

Manuscript received March 26, 2021; revised May 28, 2021; accepted June 21, 2021. Date of publication July 20, 2021; date of current version January 14, 2022. *(Raffaele Imbriaco and Clint Sebastian contributed equally to this work.) (Corresponding author: Raffaele Imbriaco.)*

Raffaele Imbriaco and Egor Bondarev are with the Video Coding and Architectures Group, Department of Electrical Engineering, Eindhoven University of Technology, 5612 Eindhoven, The Netherlands (e-mail: r.imbriaco@tue.nl).

Clint Sebastian and Peter H. N. de With are with Cyclomedia Technology, 5301 Zaltbommel, The Netherlands, and also with the Video Coding and Architectures Group, Department of Electrical Engineering, Eindhoven University of Technology, 5612 Eindhoven, The Netherlands.

Digital Object Identifier 10.1109/TGRS.2021.3095957

loss functions are originally intended for single-label retrieval, we extend these such that they are applicable for multilabel retrieval.

3) A novel loss function called ordered multilabel (OML) loss is proposed, which accounts for both classwise and order information, offering high performance.
4) A fast and effective reranking method called Jaccard Affinity reranking is proposed, and we study commonly used reranking techniques in the RS literature.

The remainder of this article is organized as follows. Section II provides an overview of the existing methods in the RS literature. Section III describes the new evaluation protocols, loss functions, and reranking techniques. Section IV provides the experimental setup and results followed by the conclusion in Section V.

II. Related Work

This section provides a summary of recent single-label remote sensing image retrieval (RSIR) systems. This is followed by an overview of existing multilabel RSIR systems and datasets. Afterward, reranking techniques in the context of RSIR are briefly discussed.

A. Single-Label RSIR

The earliest retrieval systems employ handcrafted textural features to describe the information present in aerial images [1], [2]. More advanced systems exploit high-level features by aggregating local descriptors into a single representation. Combinations of low- (e.g., textural) and medium-level features, such as Bag-of-Words [22], have also been studied in [13], [23]–[25]. However, the advent of CNNs has dramatically increased the performance of RSIR systems. Convolutional descriptors provide high-level semantic features in a compact representation. Furthermore, CNN-based descriptors are often trained to further enhance their performance and robustness [5], [7], [8], [26], [27].

The work of Sumbul *et al.* [28] provides an overview of RSIR systems using deep learning. The authors categorize them according to the strategies used for training and model deployment. For example, a system can be characterized by its network architecture, learning strategy, or loss function. We adhere to this categorization and discuss other works based on their approach to feature learning for RSIR. Three different ways of learning features exist in the literature, which is briefly elaborated in the following. The first approach is training networks using classification losses to produce semantically enriched representations. The second employs metric learning to improve the distance between similar images in feature space. Finally, the third uses reconstruction losses for training without supervision. An example study of this last category is [7], where patches are reconstructed and used inside the Bag-of-Words framework. In this work, we focus on supervised systems and solutions for RSIR.

1) Classification: CNN features have become increasingly popular in RS since they provide high-level semantic information that can be adapted for specific tasks [29]. In this case, the assumption is that each image $\mathbf{X}_i$ in a dataset is assigned a label l_i belonging to one of C different classes. These networks are trained using a classification loss, such as the cross-entropy loss. The results are descriptive and compact feature representations that can be extracted at different network depths [7]. In [10], Schuman *et al.* present a masking loss that enables simultaneous training of multiple datasets. In this way, a single network can learn from diverse datasets avoiding the pitfalls of semantic association between disjunct classes. In [8], a CNN is trained to extract local descriptors following a two-stage procedure. First, the network features are fine-tuned with RS imagery. Then, it learns an attention mask for local feature selection. In both stages, a classification loss is employed. The local descriptors are aggregated and converted into a single global representation as a vector of locally aggregated descriptors (VLAD) [30]. Although training these networks is relatively straightforward, there is no guarantee that the descriptors will be close in the feature space. Metric learning provides a solution to this problem. By explicitly learning similarity and dissimilarity, better representations are obtained which are particularly suited for retrieval applications.

2) Metric Learning: Since its introduction in [31], metric learning has become instrumental in image retrieval tasks, such as landmarks [32] and person reidentification [33]. In metric learning, the CNNs learn to quantify image similarity using the contrastive loss [31], the triplet loss [34], or other more advanced loss functions. In [35], an in-depth review on metric learning is provided.

In RSIR, metric learning is commonly used and generates highly descriptive image representations [3]. The work of Liu *et al.* [6] focuses on improving the loss used for metric learning. They propose a global optimal structured embedding concept that considers the distribution of positive and negative samples in each training batch. Thereby, the authors decrease the intraclass variability and increase the interclass descriptiveness. Similarly, [12] proposes a novel loss function that utilizes an informative set of image samples. This set is dynamically weighted to enhance the resulting descriptors, hence improving the retrieval performance. A different proposal is introduced in [4], where a Siamese graph convolutional network is deployed to encode the relationships occurring in each image. The authors generate superpixels for each image and produce a region adjacency graph. This approach takes a finer look at the details in the images and learns graph similarity. While metric learning improves the performance of RSIR systems, similarity learning does not explicitly capture the relative sorting between images. Loss functions that enable explicit rank-learning are discussed as follows.

3) Rank Learning: In this paradigm, the network learns the relative ordering between a query image and a collection of database images. Commonly, the optimized metric is the average precision (AP, the area under the precision–recall curve). However, the AP metric uses a nondifferentiable indicator function. This indicator function is a unit step function that has either discontinuous or zero derivatives. Hence, training with gradient methods is impossible [36]. There are different approximations to the AP function in the literature. In [37], the AP metric is approximated in a histogram fashion that considers the precision and recall as parametric functions to

be learned. An alternative histogram relaxation of the AP function is presented by the authors of [38]. Here, the indicator is replaced by a smooth quantization function that produces a soft assignment. In addition, the ranks of all images in the dataset are efficiently computed and used during training. While AP losses consider a histogram for the approximation of the AP function, recent work provides a simpler differentiable approach to learning with AP [36]. The authors substitute the indicator function with a parameterized sigmoid function. This sigmoid contains a temperature parameter that allows increasing or decreasing the sharpness of the derivative.

A caveat of the previously discussed work is that it deals exclusively with single-label images. However, RS imagery captures large areas where several semantic categories occur [39], [40]. This is a significant increase in complexity because losses and methods based on binary image correspondences (an image is either a positive or a negative match) need to be extended to consider nonbinary image matching. While the single-label RSIR literature is abundant, multilabel RSIR has received little attention.

B. Multilabel RSIR

In recent years, the multilabel image retrieval problem has attracted the interest of the computer vision community. As a result, various solutions have been proposed, which exploit region proposals to identify object instances [41], [42] or directly exploit label information [43]–[45]. Satellite and aerial images span a much larger geographical area and present significant appearance variations even across objects of the same class. The former is due to seasonal and environmental changes, whereas the latter stems from the intrinsic variability of the imaged objects. Nevertheless, some of the modules and proposals from image retrieval literature can be adapted for ML-RSIR.

In [16], Dai *et al.* present a multilabel RSIR (ML-RSIR) system based on both spectral and spatial features (extracted with scale-invariant feature transform (SIFT) [46]). Their system constructs a codebook as in BoW [22] for each of the extracted features. Then, they are combined into a single descriptor. The proposed retrieval pipeline learns the probability of a label appearing in an image, given its spatial and spectral content. The system of [16] is tested and trained with a multispectral dataset [47]. Chauduri *et al.* [48] propose a novel graph-based framework for ML-RSIR. In addition, they also provide image-level annotations for the UC Merced dataset [13]. The authors of [15], [17] consider that the absence of dedicated ML-RSIR datasets hinders research in the field and proposes to remedy this by annotating the UC Merced dataset. Using a semi-automated system, they generate pixel-level annotations for 17 different classes. Two other datasets (Wuhan dense labeling dataset (WHDLD) [49], aerial image dataset (AID) [50]) have been adapted from single-label to enable multilabel processing. Semantic segmentation-based solutions to the ML-RSIR problem are evaluated in [51] and [49]. The networks are trained for semantic segmentation, and features are extracted from the various predicted masks.

Unfortunately, some of the previously mentioned work does not provide a thorough enough explanation of the metrics employed. These metrics have been directly uprooted from the single-label task without providing explanations on how they are being computed for the multilabel case. For example, the mean Average Precision (*m*AP) metric is commonly employed to evaluate the performance of ML-RSIR systems. As mentioned above, the *m*AP metric employs a ranking function based on a binary indicator. However, it is not explained how the new multilabel notation of the datasets affects this metric. An exception is [49], where the authors specify that images should have at least one label in common to be considered correct. This binarizes the image correspondences but is an oversimplification of the ML-RSIR problem.

A common component of image retrieval systems is reranking, which involves postprocessing techniques exploiting similarities in the database to improve retrieval performance.

C. Reranking

A popular technique for reranking is average query expansion (AQE) [52], [53], wherein a new representation of the query vector is generated by averaging the descriptors of the query and the top-*k* database matches. An improvement of AQE is presented in [32], where the database descriptors are weighted by their distance to the query image.

Other reranking techniques do not reconstruct a query descriptor but exploit additional information instead. For example, the work of Ye *et al.* [21] proposes to compute the CNN-generated label vector to that of the database after retrieving the top matches. Pedronette *et al.* [54] present a scalable technique that uses truncated rank lists and efficient data structures to leverage the contextual information present in ranked lists. A trainable reranking technique is developed in [55]. After generating the initial ranking, the authors train a classifier to identify similar and dissimilar images. Furthermore, a secondary step of visual reranking is employed to increase retrieval performance.

D. Dataset Evaluation Protocols

In the current RSIR landscape, there exist a variety of datasets for the evaluation of RSIR methods. Unfortunately, across this vast literature, there appears to be little consensus on dataset splits and evaluation protocols. Hence, replicating results and direct comparisons across systems are difficult. The latter is further complicated when different systems saturate the performance on the available datasets. Table I lists a summary of the retrieval results of several state-of-the-art solutions. The results presented in Table I come from systems using different train and test splits, ranging from 50%–50% to 80%–20% per dataset. The inconsistencies in the current evaluation protocols complicate fairly assess RSIR systems. For example, this occurs when comparing the performance of the systems proposed in [20] and [3]. Across both datasets, their performances are almost identical and consistent. However, when looking at the second-best systems ([5] and [11]), we observe roughly a 27-point *m*AP difference in the UC-Merced dataset. In our opinion, these differences

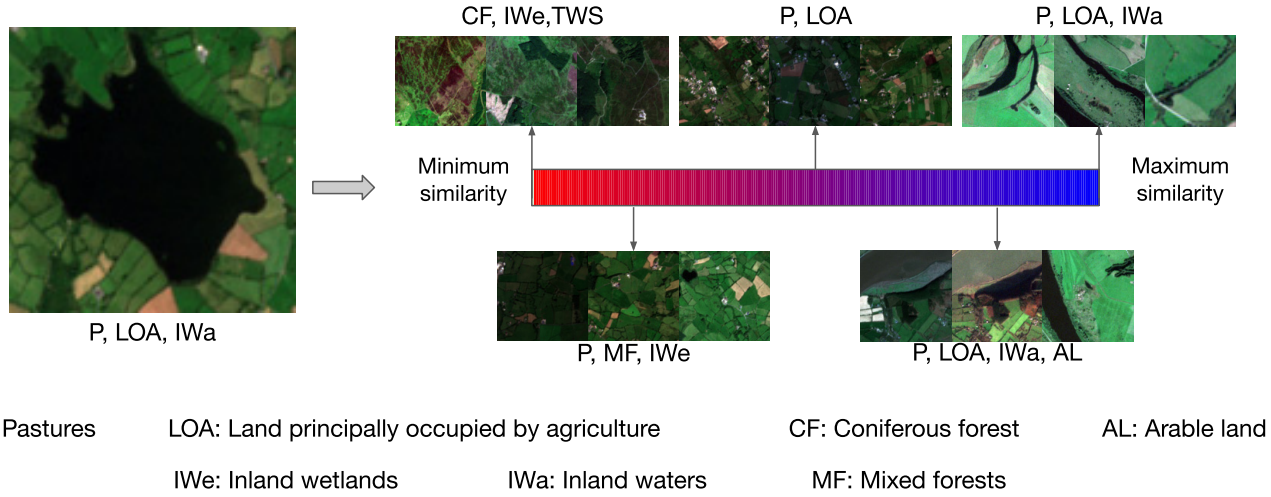

Fig. 1. Visual representation of the Jaccard label distance on BigEarthNet19 images. The leftmost image is an example query, while the other images are used to compute the Jaccard distance. Visual content similarity can occur with little to no label similarity.

TABLE I
RETRIEVAL PERFORMANCE COMPARISON OF STATE-OF-THE-ART RSIR SYSTEMS ON TWO POPULAR BENCHMARKS. LARGER *m*AP, MP@10 VALUES, AND SMALLER AVERAGE NORMALIZED MODIFIED RETRIEVAL RANK (ANMRR) VALUES ARE BETTER. SEVERAL OF THE PRESENTED SYSTEMS SATURATE PERFORMANCE ON COMMON RSIR DATASETS, WHICH INDICATES THE NEED FOR A MORE CHALLENGING RSIR PROTOCOL

Method	PatternNet [19]			UC Merced [13]		
	*m*AP	ANMRR	mP@10	*m*AP	ANMRR	mP@10
CSN+DF [20]	99.6	0.0027	99.6	96.7	0.0211	97.6
DCL [12]	99.4	-	100	98.8	-	100
EM-CNN [5]	99.5	0.0021	99.6	97.0	0.0203	97.7
RR-ICD [21]	98.5	0.0118	-	95.6	0.0359	-
MLMD [10]	98.4	0.0106	-	87.2	0.0972	-
SGCN [4]	81.8	0.2100	97.1	69.9	0.3000	93.6
DML [3]	99.6	0.0030	99.6	96.6	0.0223	97.6
T-T-C Net[11]	99.5	0.0025	99.6	70.4	0.2344	89.6

stem from the lack of consensus between evaluation and dataset splits.

The authors of [27] also identify a similar problem and propose a method for generating challenging RSIR datasets from the existing sets. However, ML-RSIR has received little attention. The existing benchmarks consist of relatively small datasets containing pixel-level multilabel information. We propose to leverage the large, multilabel dataset BigEarthNet19 [56] and provide a standardized protocol for the evaluation of ML-RSIR systems using metrics common to multilabel retrieval and a flexible image correspondence methodology. By doing this, we aim at a solution that facilitates a fair comparison between systems and preserves reproducibility. The code and models will be made available upon acceptance.

III. METHODOLOGY

As previously discussed, current ML-RSIR systems use evaluation metrics from single-label problems where image correspondences are binary (matches are either positive or negative). However, when images can belong to more than a single class, correspondences can be interpreted as a real number in a predetermined range. An example is depicted in Fig. 1, where images are ranked within a label-similarity interval between zero and unity. To employ metrics, such as *m*AP, the multilabel matching should be converted into a binary correspondence. This relaxation of the multilabel image correspondence drastically changes the difficulty of the ML-RSIR problem. If only the exact label matches are positives, a minuscule number of the database images will be relevant. This is especially true for large datasets with more varying class nomenclatures. Alternatively, one could consider the case of sharing at least one class in common, as in [49]. Under this convention, images that share more than one label are equally relevant as images that share a single class. To solve these problems and inspired by recent work in landmark retrieval [57], we propose a more flexible relaxation of the multilabel correspondence for ML-RSIR.

The methodology is organized as follows. The new evaluation protocols are described in Section III-A. The novel loss function and reformulated loss functions for multilabel retrieval are elaborated in Section III-B. The novel and existing reranking techniques are discussed in Section III-C.

A. Evaluation Protocol

To determine how similar the labels of an image $\mathbf{X}_i$ are to the labels of any other image $\mathbf{X}_j$, we consider the Jaccard Index of their multiclass labels l_i and l_j, respectively. By doing so, the label similarity is defined as

$$J(l_i, l_j) = \frac{\|l_i \cap l_j\|}{\|l_i \cup l_j\|}. \tag{1}$$

Using the Jaccard index, the label similarity will span the interval between zero (total mismatch) and unity (perfect match). In this manner, it is possible to select a cutoff threshold τ_c to determine whether images should be considered matched or mismatched. The advantage of this approach is the inherent flexibility allowed by the value of τ_c. Thus, we define the set $\mathcal{S}_i$ of correct matches for image $\mathbf{X}_i$ by

$$S_i = \{\mathbf{X}_j \mid J(l_i, l_j) \geq \tau_c\}. \tag{2}$$

By adopting different values of τ_c, the ML-RSIR problem can be posed as strictly as desired. In this work, we propose three different evaluation protocols with an increasing level of difficulty, as in [57]. These protocols are denoted as *Easy*, *Medium*, and *Hard*, corresponding to the thresholds 0.40, 0.60, and 0.80, respectively.

These thresholds are selected to highlight how well the features encode the semantic differences present in the data. Better performing models will also be able to learn representations for similar images with only slight label differences. Furthermore, this evaluation protocol will facilitate comparisons between different proposals. As discussed above, many RSIR systems employ increasingly advanced techniques and more complex models. However, the datasets used for training and testing do not seem to increase in complexity at the same pace. Therefore, we consider it relevant to develop a range of protocols that enable the evaluation of problems of varying complexity.

In the existing literature, machine learning systems are evaluated on problems of different complexity by testing them across several datasets. However, there are some disadvantages to this approach. First, there is no apparent consensus regarding the fraction of test and training splits. Common ratios range from 50/50 to 80/20. These do not generally include validation sets and are likely different in each work. In addition, different datasets have different nomenclatures, or even different semantic categories altogether. Hence, the image content and the number of classes can vary. Second, the CNNs acting as feature extraction units need to be trained in each dataset. In most cases, this is a time-consuming and computationally intensive task. By considering different evaluation protocols using the same dataset, we can alleviate the aforementioned issues. In addition, this may enable better and fair comparisons across RSIR systems.

Needless to say, binarization of the multilabel image correspondence leads to information loss. Images below the cutoff threshold are considered negatives, but this does not encode how far apart the labels actually are. This also holds for the positive samples. Better representations should be obtained by systems capable of exploiting the full multilabel information. After obtaining a suitable dataset split and appropriate nomenclatures, the model should be trained with a loss function that is in accordance with the new protocols.

B. Loss Functions

This section formulates the existing loss functions for multilabel retrieval and describes the proposed multilabel loss. The conventional approach to learning descriptors can be divided into two types. The first approach deploys metric learning or ranking loss functions that separate the descriptors by using feature distances. The second approach is to learn to discriminate descriptors by classifying them. In the latter case, the descriptors are implicitly separated in feature space via a classification loss. In an ideal scenario, the class information and the feature distance are essential to improve performance. In the context of multilabel retrieval, the current loss functions that are used in single-label retrieval cannot be directly applied. Therefore, we reformulate them such that they are applicable to multilabel retrieval systems.

The two most popular metric learning-based loss functions are contrastive and triplet loss functions. Both of these loss functions utilize mining strategies to select anchor, positive, and negative samples to maximize distances between the positive and negative samples. In the case of single-label retrieval, the definition of what is positive or negative is well established. Given a sample anchor $a \in \mathcal{C}_i$, a sample is defined as positive when $p \in \mathcal{C}_i$ and negative when the sample $n \in \mathcal{C}_j$, where $j \neq i$ and $\{\mathcal{C}_i, \mathcal{C}_j\} \in \mathcal{C}$. The set $\mathcal{C}$ is the set of all class labels. This formulation is challenging for multilabel retrieval, as each sample could have a variable number of labels.

The existing protocol does not provide a clear distinction between positive and negative images, for a given anchor. To create this separation, we utilize the Jaccard Index between the labels of the anchor image a and another image i. For example, an image is positive (p), when $J(l_a, l_i) > 0.5$, and negative (n), when $J(l_a, l_i) < 0.5$, where l_a and l_i are the labels of the anchor image and another image in the minibatch. Hence, during mining in metric learning losses, such as contrastive and triplet loss, the label correspondences are checked to generate positives and negatives. Therefore, the contrastive loss can now be formulated as

$$\mathcal{L}_{\text{cont}} = \begin{cases} T(z) \cdot D(x_a, x_p), & z > 0.5 \\ (1 - T(z)) \cdot \max(0, m - D(x_a, x_n)), & \text{otherwise.} \end{cases} \tag{3}$$

Here, D denotes the cosine distance between a pair of embeddings x_i, x_j. The parameter m is the margin, and $T(z)$ is the step function given by

$$T(z) = \begin{cases} 1, & \text{for } z > 0.5 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

where $z = J(l_a, l_i)$. Similarly, the triplet loss is given as

$$\mathcal{L}_{\text{triplet}} = \max(0, m + D(x_a, x_p) - D(x_a, x_n)). \tag{5}$$

The metric that we use for determining positives and negatives is the Jaccard Index. We choose a Jaccard Index of 0.5, as other settings add additional challenges. For example, a value of 0.8 would make batch mining very strict. This would require the anchor and positive sample to have a high degree of label similarity. In practice, such samples may not be present in the minibatch. On the other hand, a value of 0.2 would make the mining too lenient and may consider sample pairs with low label similarity to be positives. To preserve the simplicity and avoid overtuning, we choose a Jaccard Index of 0.5 as the boundary for separating positives and negatives.

For single-label multiclass problems, the cross-entropy loss is the most widely used objective function. The cross-entropy objective is a feasible loss function for multilabel classification. In such a scenario, the sigmoid activation replaces the softmax function moving from a multinomial to a binomial distribution. Applying the cross-entropy loss function over the binomial distribution of each classification label creates a one-versus-all objective. Therefore, the cross-entropy function will

be applied across each label of a sample. The cross-entropy loss ($\mathcal{L}_{\text{xent}}$) specified by

$$\mathcal{L}_{\text{xent}} = -(y \cdot \log(\hat{p}) + (1 - y) \cdot \log(1 - \hat{p})) \tag{6}$$

where y is the target class probability and $\hat{p}$ the predicted probability.

The benefit of the metric learning approaches is that they discriminate embeddings by separating them in feature space while ignoring class-based separation. The classification losses ignore feature distances between embeddings and segregate features by classification labels. Although a strict feature-based separation is absent in classification losses, this is implicitly applied.

OML loss: From Section III-B, it is evident that both feature distances and class-based information would be an effective approach. However, multilabel retrieval has an additional challenge: the ordering of the samples. Unlike single-label tasks where any image from the same class as the query is considered a correct match, a multilabel problem would require the image with a Jaccard Index of unity to be the best match. The consecutive ranks would be images with lower Jaccard Indices in descending order. However, inducing the ordering of embeddings often requires quantization of scores, which is not differentiable. Therefore, to obtain an approximation of the ordering, we follow the same method as in [36] such that the final loss becomes differentiable. Given these constraints, we propose a loss that considers ordering and class information called OML loss. The OML loss is given as

$$\mathcal{L}_{\text{OML}} = \frac{1}{N_{\mathcal{C}}} \sum_{c \in \mathcal{C}} \left(1 - \frac{1}{N_{X_c}} \sum_{i \in X_c} \frac{S(i, \tau)}{S(\psi, \tau)} \right) \tag{7}$$

where $\mathcal{C}$ is the set of classes present in the minibatch, $N_{\mathcal{C}}$, the cardinality of set $\mathcal{C}$. Similarly, X_c is the set of all the elements with label c in the minibatch. N_{X_c} denotes the number of elements in X_c. The variable ψ denotes all the elements in the minibatch. The scoring function $S(\cdot, \tau)$ is the temperature-scaled sigmoid shifted by unity, which is given by

$$S(i, \tau) = 1 + \sum_{j \in X_i} \frac{1}{1 + \exp\left(\mathbf{A}^d_{ij} / \tau\right)}. \tag{8}$$

The affinity matrix $\mathbf{A}_{ij}$ is the similarity between the ith and jth embeddings. The matrix $\mathbf{A}^d$ at i, j represents the sum of differences of the similarity of an element i with every other element j in the matrix $\mathbf{A}_{ij}$. The output of the scoring function is the accumulation of the distance affinity matrix $\mathbf{A}^d{}_{ij}$ after normalization.

The scoring function generates a score for each element in the minibatch. The highest scoring element indicates the least correlated sample within the given class. Conversely, the sample will have the highest similarity with itself, and the scoring function will yield a value of zero. Therefore, the scoring function approximates an ordering to all elements in the set. The scoring function is applied across all elements of a class c and the entire sample space ψ in the minibatch. The output of $S(\psi, \tau)$ denotes the true ordering of an element in the minibatch, whereas $S(i, \tau)$ indicates the ordering within the class. The OML loss enforces each classwise ordering to be close to the true order. In a multilabel scenario, each classwise ordering is pushed toward the true order. This effectively means that each sample is implicitly reweighed with the number of associated labels. For example, in Class B of Fig. 2, the triangle in red has the first position locally. However, it has the third rank in the global ordering. The OML loss pushes the local first rank toward a global third.

C. Reranking

Reranking is often applied to improve retrieval performance by recomputing similarity with an enhanced representation or other improvements. The most common approach for reranking images in RS and landmark retrieval literature is using query expansion [8]. The conventional query expansion technique is the AQE. Given a query q and its top matches $d_1, d_2, \ldots, d_m$, AQE renews the query to an improved representation r_{qe} into

$$\hat{r}_{\text{QE}} = q + \sum_{i=0}^{k} d_i \tag{9}$$

and

$$r_{\text{QE}} = \frac{\hat{r}_{\text{QE}}}{\|\hat{r}_{\text{QE}}\|}. \tag{10}$$

Similarly, the query representation is improved by considering the power (α) of the query distance to each of its top matches. This is known as α query expansion (αQE), which is defined as

$$\hat{r}_{\alpha\text{QE}} = q + \sum_{i=0}^{k} d_i \cdot \text{sim}_{\cos}(q, d_i)^{\alpha} \tag{11}$$

and

$$r_{\alpha\text{QE}} = \frac{\hat{r}_{\alpha\text{QE}}}{\|\hat{r}_{\alpha\text{QE}}\|} \tag{12}$$

where $\text{sim}_{\cos}$ is the cosine similarity between a pair of embeddings. The reformulated query using αQE is $r_{\alpha\text{QE}}$, and parameter k is number of top matches considered for query renewal. Both these methods are online methods and are calculated on the fly. However, both these methods also rely on feature similarity to improve reranking. This is reliable in the cases where the ranks are indifferent to ordering. As discussed in Section III-B, it is evident that multilabel retrieval performance depends on well-ordered results.

Jaccard Affinity Reranking: To improve efficiency and performance, we propose to use a novel reranking scheme based on label similarity. This scheme relies on the label rather than feature similarity since label similarity provides a more accurate representation of the correct ordering. To utilize the label similarity, we propose a graph-based approach for reranking. The benefit of this approach is that it requires minimal online computation because the database graph can be constructed offline. Let $\mathbf{L} = [\ell_1, \ell_2, \ell_3, \ldots, \ell_m]$ be the overall vector representing the labels vectors corresponding to each image in the database. Then, the label affinity matrix $\mathbf{A}^L_{ij}$ is

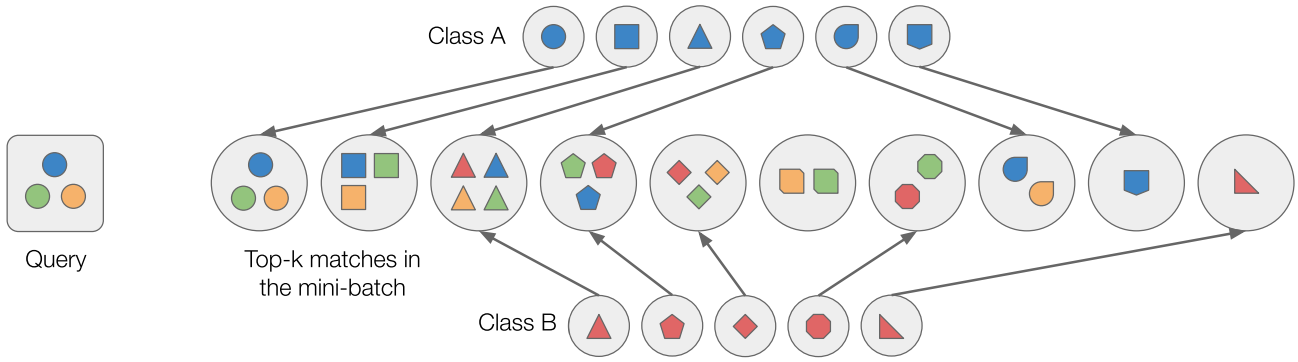

Fig. 2. Visualization of OML loss. Given a batch of images, we sample a query image (shown in the circle) and its corresponding top matches to obtain global ordering (mid-row). We select a single label from each sample and order them per class by feature distances to get local ordering (shown in top and bottom rows). Each shape represents a sample, and each color denotes a class. Through OML loss, the local ordering per class is pushed toward a global ordering.

defined by the Jaccard function of the overall vector and its transpose such that

$$\mathbf{A}_{ij}^{L} = J(\mathbf{L}, \mathbf{L}^{T}) \tag{13}$$

where $\mathbf{L} \in \mathbb{R}_{m \times n}$, parameter m is the number of label vectors in $\mathbf{L}$, and n is the number of labels associated with each query image. The affinity matrix is then ranked with respect to the Jaccard Index to obtain a graph $\mathbf{G}$ corresponding to each element d_i in the database. Given a query descriptor q, with matches $d_1, d_2, \ldots, d_m$, we replace the matches by top matches corresponding to query d_1 from $\mathbf{G}$. It should be noted that both $\mathbf{A}_{ij}^{L}$ and $\mathbf{G} \in \mathbb{R}_{m \times m}$. The key advantage is that the nearest neighbor graph $\mathbf{G}$ can be computed offline, which results in fast online retrieval. Besides this, the label similarity graph is far more accurate than a feature similarity graph as the former induces ordering through the Jaccard Index.

D. Deep Hashing

In the RSIR literature, deep hashing is a common technique for addressing large-scale datasets [58]. While we consider deep hashing to be outside the scope of this work, it is important to address how such a popular technique can be integrated into ML-RSIR. Fortunately, the proposed system requires minimal changes in order to enable deep hashing. In order to obtain short binarized descriptors, the last few layers of the CNN need to be changed to fully connected layers for dimensionality reduction. This would introduce additional parameters but would not fundamentally alter the computation of OML. In addition, our system does not preclude the deployment of regularizing losses, such as the binary quantization loss in [59]. However, there exists the possibility that employing the Hamming distance directly within OML could yield subpar results as it currently computes the Euclidean distance.

IV. Experiments

A. Datasets

We evaluate our proposed loss and reranking technique across four different multilabel datasets, including a large-scale dataset. These datasets are described as follows.

1) DLRSD: This dataset is presented in [17], and it is a multilabel adaptation of the UCM [13] dataset. The authors provide annotations for 17 classes obtained from semantic segmentation. It contains 2100 images with a spatial resolution of 0.3 m, and each image is 256 × 256 pixels.

2) WHDLD: The dataset first was introduced in [49] and contains 4940 RGB images with a spatial resolution of 2 m. Their image sizes are identical to those of the dense labeling remote sensing dataset (DLRSD) dataset. However, the number of classes present in WHDLD is drastically smaller. Only six semantic labels are annotated: bare soil, building, pavement, road, vegetation, and water. Similar to DLRSD, pixel-level annotations are available.

3) ML-AID: The ML-AID dataset [50] consists of 3000 images, which is adapted from [60] and was originally used for multilabel classification. Unlike the previous datasets, it contains images with various spatial resolutions ranging between 0.5 and 8 m. ML-AID uses the same 17-class nomenclature of DLRSD.

4) BigEarthNet19: Originally published for multilabel classification [61], this large-scale dataset contains 590 236 multispectral images belonging to 45 classes. A recent revision to the nomenclature [56] collapses many of the original 45 classes into 19 complex semantic labels. BigEarthNet19 provides hyperspectral bands in addition to the RGB bands present in the previous datasets.

B. Evaluation Metrics

The results are evaluated using the standard metrics in retrieval. We consider the *m*AP under different thresholds of the Jaccard Index to assess the relevance of label overlap between a query and its top matches. To evaluate the quality of the matching order, we use normalized Discounted Cumulative Gain (*n*DCG). To obtain global performance, we use weighted Average Precision (*w*AP) [62]–[64].

1) Mean Average Precision: The *m*AP is defined as

$$m\text{AP} = \frac{1}{Q} \sum_{q}^{Q} \text{AP}(q) \tag{14}$$

where AP is the average precision and is defined as

$$\mathrm{AP}(q) = \frac{1}{N_{\mathbb{1}}(q)@k} \sum_{i}^{k} \mathbb{1}(q,i) \frac{N_{\mathbb{1}}(q)@i}{i} \tag{15}$$

where $\mathbb{1}(q,i)$ is the indicator function for the query q and ith image, and $N_{\mathbb{1}}(q)@k$ is the number of positive images in the top-k ranks. As mentioned above, we study three different settings for the indicator function for assessing the difficulty of image matching. These protocols, denoted as *Easy*, *Medium*, and *Hard*, use increasingly strict Jaccard Index thresholds of 0.40, 0.60, and 0.80, respectively. These protocols quantify how well the method performs under the condition a variable number of labels. This enables intuitive interpretation of the results. Furthermore, it enhances the importance of the label similarity in ML-RSIR and proper ordering when ranking the database (higher Jaccard Index should be ranked higher).

2) Normalized Discounted Cumulative Gain: The *n*DCG is well-suited for multilabel retrieval problems, as the metric can measure the relevance of the top-ranked results. The *n*DCG at k is defined as

$$n\mathrm{DCG}@k = \frac{\mathrm{DCG}@k}{\mathrm{IDCG}@k}. \tag{16}$$

The DCG@k for a query q is defined as

$$\mathrm{DCG}@k = \sum_{i=1}^{k} \frac{2^{J(q,i)} - 1}{\log_2(i+1)} \tag{17}$$

where $J(q,i)$ is the Jaccard Index between query and the ith ranked image. The ideal discounted cumulative gain (IDCG@k) is specified by

$$\mathrm{IDCG}@k = \sum_{i=1}^{k} \frac{2^{J(q,i)_r} - 1}{\log_2(i+1)} \tag{18}$$

where $J(q,i)_r$ is the number of common labels between the query and the ith ranked image sorted by decreasing order of relevance r. The advantage of *n*DCG is that it constrains the value between zero and unity, leading to more interpretable results and comparisons.

3) Weighted Average Precision: The weighted Average Precision (*w*AP) metric is similar to the *m*AP. However, it employs the number of shared classes between the query and each retrieved image in its construction. Hence, it is suitable for ML-RSIR without needing to define a specific threshold for the indicator function. Therefore, the *w*AP can be calculated by

$$w\mathrm{AP} = \frac{1}{Q} \sum_{q}^{Q} \left(\frac{1}{N_{\mathbb{1}}(q)@k} \sum_{i}^{k} \mathbb{1}(q,i) \left(\sum_{j}^{i} \frac{J(q,j)}{i} \right) \right). \tag{19}$$

C. Implementation Details

We deploy the ResNet50 architecture pretrained on ImageNet and train it on each of the previously mentioned datasets [65]. We employ the Adam optimizer with a learning rate of 10^{-4} [66]. The network is trained with contrastive, triplet, cross-entropy, or OML losses. Except for BigEarthNet19, we use a reproducible data split of 70/10/20 for training, validation, and testing. On BigEarthNet19, we use the splits provided by the authors. Each minibatch contains 72 images and is trained for 100 epochs on the DLRSD, WHDLD, and ML-AID datasets. The BigEarthNet19 dataset is trained with a batch size of 256 for 60 epochs. During the evaluation, the test set is separated into disjoint sets of queries and databases. Query images with no positive matches are discarded due to this separation (all positives belong to the query set). We compute the *m*AP over the whole database for the different protocols (*Easy, Medium, and Hard*), and the *w*AP and *n*DCG metrics with $k = 100$.

TABLE II

RETRIEVAL RESULTS ON THE DLRSD DATASET. BOLD INDICATES THE HIGHEST SCORE

	*m*AP			nDCG@100	*w*AP@100
	Easy	Medium	Hard		
Contrastive	49.5	30.5	22.5	78.7	1.72
Triplet	86.3	70.5	54.4	92.9	2.45
Cross-Entropy	84.5	70.8	56.9	92.9	2.34
OML	**89.2**	**75.2**	**59.8**	**94.9**	**2.53**
with Jaccard Affinity Re-ranking					
Contrastive	67.5	48.3	33.7	81.6	1.93
Triplet	90.6	76.7	58.8	94.2	2.54
Cross-Entropy	**93.6**	78.9	**64.9**	95.1	2.56
OML	93.4	**81.6**	64.3	**95.7**	**2.59**

D. Results

This section compares and discusses the performance of the proposed OML loss against losses commonly used in single-label and multilabel RSIRs. The retrieval performance is evaluated before and after Jaccard Affinity reranking on four different datasets. Afterward, different reranking methods are compared on the descriptors generated by the OML loss.

1) Multilabel Retrieval: Table II showcases the retrieval performance on the DLRSD dataset. We observe that the OML loss consistently outperforms other losses across all protocols and yields a significant *m*AP gain. A similar trend is noticeable both for *n*DCG and *w*AP, each presenting an increase of approximately 2% and 8%. A surprising outcome is a high performance produced by the cross-entropy loss and the poor-quality matches generated by the contrastive loss. In several metrics, the performance of cross-entropy loss rivals or exceeds that of the triplet loss. The likely cause for this is the simple relaxation used for determining positive and negative matches during mining. When training with the metric losses, only partial class information is present. This is due to the binarization of the multilabel image correspondences. However, OML learns the relative ordering between samples from class and metric information, which improves the performance.

When Jaccard Affinity reranking is applied, another significant *m*AP gain is observed. However, when using this reranking technique, the performance gap across three losses (OML, Cross-entropy, Triplet) is significantly reduced. This can be explained by the strong dependence of the Jaccard

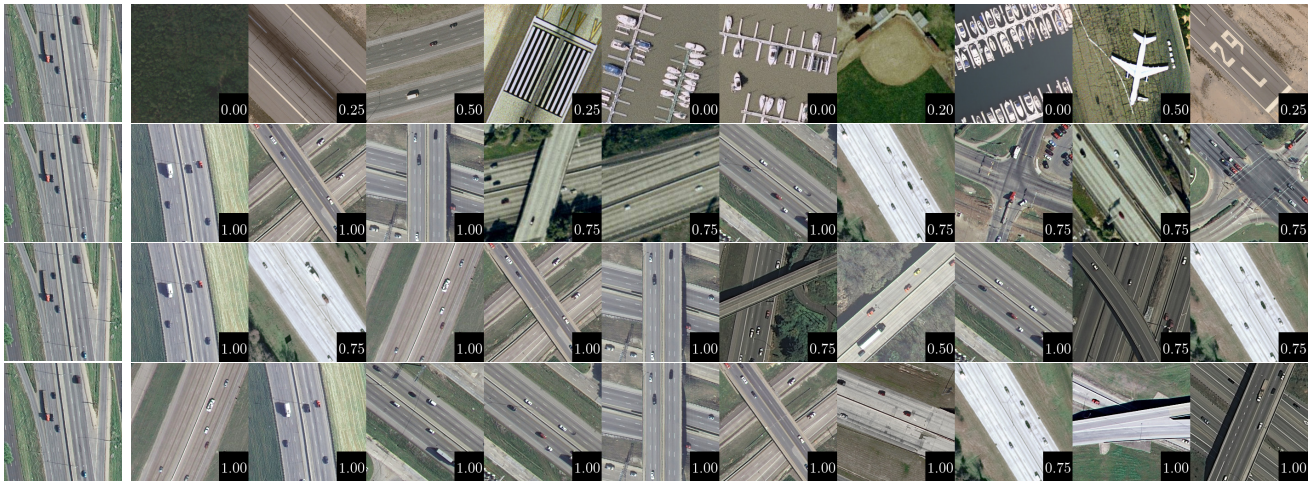

Fig. 3. Example of retrieved images on the DLRSD dataset with different losses. Top to bottom: contrastive, triplet, cross-entropy, and OML. The query image is the leftmost image. The black boxes at the bottom right of the matched images contain the Jaccard Index in relation to the query (higher is better).

TABLE III
RETRIEVAL RESULTS ON THE ML-AID DATASET. BOLD INDICATES THE HIGHEST SCORE

	mAP			nDCG@100	wAP@100
	Easy	Medium	Hard		
Contrastive	79.2	62.7	31.5	84.2	3.93
Triplet	88.2	75.3	45.7	89.3	4.17
Cross-Entropy	87.8	76.1	50.3	88.9	4.16
OML	**88.8**	**77.5**	**50.5**	**92.9**	**4.44**
with Jaccard Affinity Re-ranking					
Contrastive	88.9	75.4	49.4	89.1	4.16
Triplet	96.2	86	64.9	92.6	4.37
Cross-Entropy	95.4	85.4	61.5	91.4	4.33
OML	**97.4**	**87.8**	**66.7**	**94.3**	**4.51**

TABLE IV
RETRIEVAL RESULTS ON THE WHDLD DATASET. BOLD INDICATES THE HIGHEST SCORE

	mAP			nDCG@100	wAP@100
	Easy	Medium	Hard		
Contrastive	94.3	81.1	55.4	94.6	4.17
Triplet	94.9	84.9	61.2	95.1	4.20
Cross-Entropy	**96.5**	**87.6**	**65.3**	94.7	4.19
OML	95.1	85.3	63.6	**96.7**	**4.34**
with Jaccard Affinity Re-ranking					
Contrastive	97	87.5	65.4	96.4	4.26
Triplet	**98.1**	**90.9**	**75.1**	97	4.30
Cross-Entropy	97.9	90.7	73.2	96.6	4.29
OML	97.8	90.3	73.4	**97.5**	**4.34**

Affinity reranking technique and the top-ranked images. If the best match across the different models is correctly identified, then the reranking technique will yield similar results. Hence, if all losses succeed in retrieving the most relevant image, their performance will be equalized. An example prior to reranking of the top matches for each loss is depicted in Fig. 3. It should be noted for the three best performing losses that the retrieved images have high relevance (Jaccard Index closer to unity).

For the ML-AID dataset, results in Table III, a trend similar to that of DLRSD is observable. This dataset has the same 17 semantic classes but shows a smaller gap between OML and the other losses in mAP. However, OML still obtains a higher nDCG and wAP indicating that more relevant images are found in the top-100 ranks (better overlap between query and database labels). This also highlights the increased difficulty of the ML-AID dataset, as the mAP on the *Hard* protocol barely exceeds 50% without reranking, whereas most losses achieve higher performance on the same protocol for DLRSD.

A possible explanation for this performance drop is the nature of the annotations for each dataset. ML-AID uses exclusively human annotations, while DLRSD used segmentation maps reviewed by human annotators. This, in conjunction with the smaller size of DLRSD, largely explains the difference in performance. Evidence for this claim is present in the nDCG and wAP metrics. For DLRSD, the wAP ranges from 1.5 to 2.6, whereas these values range between 3.9 and 4.5 for ML-AID. This indicates that each image in the ML-AID dataset has more classes than the DLRSD dataset. The average number of labels assigned per image is 3.31 and 5.17 for the DLRSD and ML-AID datasets, respectively. This means that it is hard to provide a good separation of the descriptors since it should largely remain similar to several classes at once. However, this peculiarity of the dataset construction significantly increases the impact of reranking. Deploying Jaccard Affinity reranking on ML-AID yields an increase of roughly ten mAP points. We conjecture that the large overlap in labels enables the reranking technique to retrieve a higher number of relevant samples.

The results on the WHDLD dataset are summarized in Table IV. In this case, cross-entropy provides a slightly higher performance on the mAP. The contrastive loss, which lags in previous experiments, offers similar performance to the other loss functions. While the WHDLD dataset is also annotated by pixel-level segmentation, it only contains six semantic classes, and the mean number of labels per image is 4.63, which is similar to ML-AID. We particularly note that cross-entropy offers strong performance when the number of classes in the dataset is low and has several labels per image. By further applying Jaccard Affinity reranking, all the methods observe a significant gain in performance.

The experimental results on BigEarthNet19 are given in Table V. While cross-entropy loss has the best overall

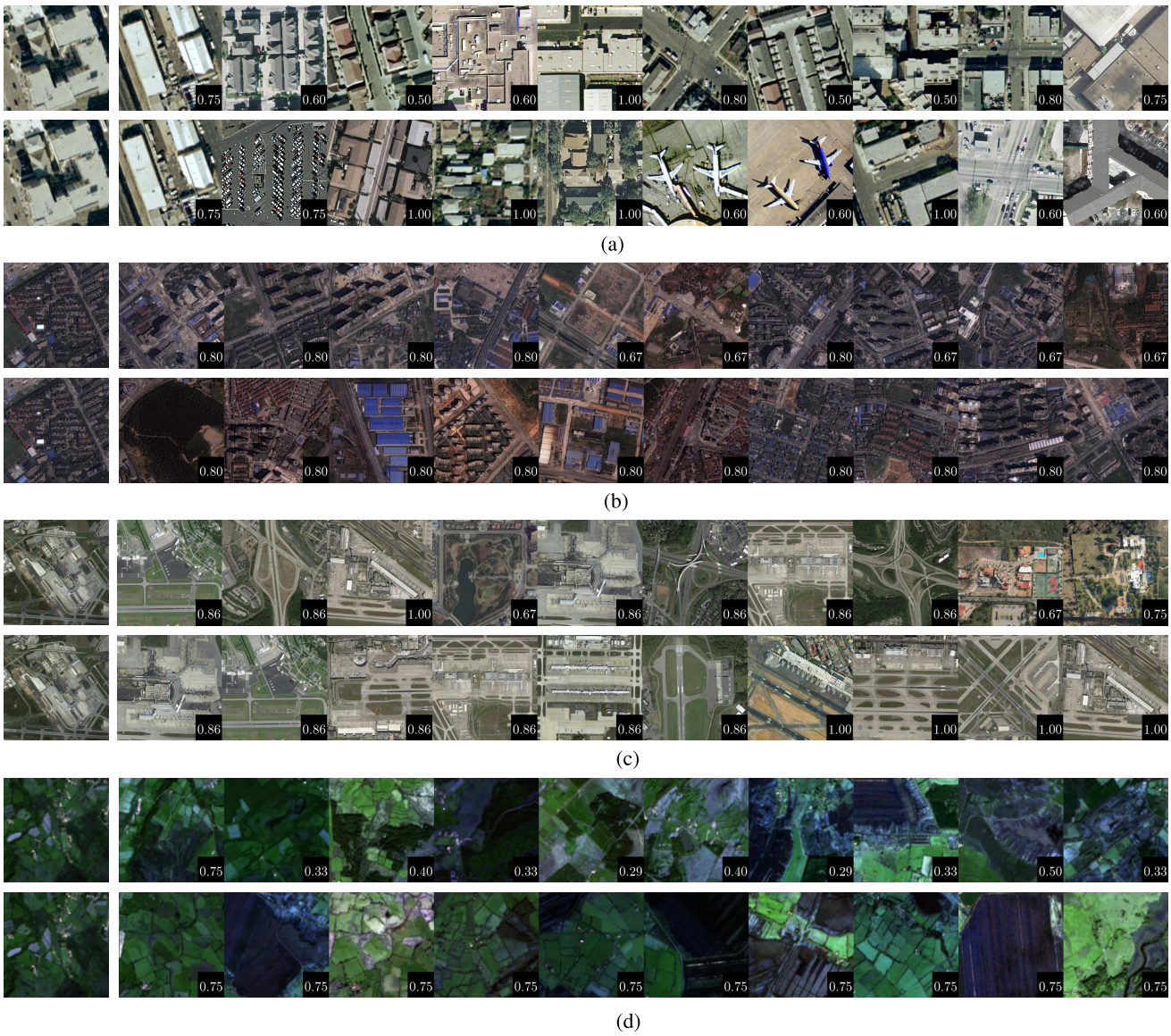

Fig. 4. Example of retrieval and reranking (lower row) with Jaccard Affinity on all the datasets. Descriptors are obtained with the OML loss. The query image is the leftmost image. The black boxes at the bottom right of the matched images contain the Jaccard Index in relation to the query (higher is better). (a) DLRSD. Query labels: "buildings," "cars," "pavement," and "trees." (b) WHDLD. Query labels: "building," "pavement," "vegetation," and "bare soil." (c) AID. Query labels: "airplane," "bare soil," "buildings," "cars," "grass," and "pavement." (d) BigEarthNet19. Query labels: "pastures," "inland wetlands," "land principally occupied by agriculture," and "moors, heathland, and sclerophyllous vegetation."

performance, the OML loss offers competitive performance. Both OML and cross-entropy outperform all metric learning-based losses by a large margin. Nevertheless, the BigEarthNet19 dataset remains the most challenging, and it is of particular interest, as it has low retrieval performance across all losses. Unlike the previous three datasets, no loss function obtains an mAP score of over 50% on the more challenging protocols. The strong cross-entropy loss performance is remarkable in this case. We conjecture that the high performance is due to the low number of labels associated with each image. On average, BigEarthNet19 has only 2.88 labels per image compared to 3.13, 4.63, and 5.15 labels for DLRSD, WHDLD, and ML-AID, respectively. We consider that, in this case, classification losses offer better performance as each image needs to associate only a few labels per image. In essence, this is closer to a classification task. Qualitative results are found in Fig. 4.

Similarly, with Jaccard Affinity reranking, the performance improves in all cases. However, on losses such as contrastive and triplet, the gains are not as large as OML or cross-entropy. Like most reranking techniques, Jaccard Affinity reranking is sensitive to the initial set of ranks. A poor set of initial ranks acts as an insufficient baseline for the reranking technique. In the cases of contrastive and triplet losses, the *m*AP on the *Hard* protocol is low, thereby leading to lower gains when Jaccard Affinity reranking is applied.

2) Reranking: For these experiments, we have evaluated two popular reranking techniques in retrieval literature, AQE, and alpha query expansion (αQE). Unlike Jaccard Affinity reranking, AQE and αQE possess hyperparameters that should

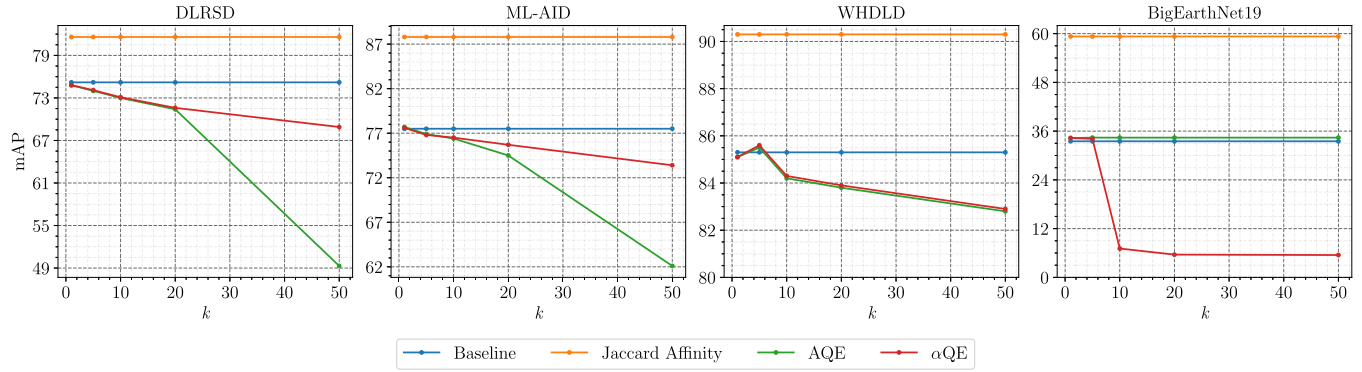

Fig. 5. *m*AP Performance on the *Medium* protocol across the four datasets. The baseline performance of OML loss is in blue, AQE in green, αQE in red, and Jaccard Affinity in orange. The Jaccard Affinity consistently improves performance, whereas both AQE and αQE lower performance to the baseline for different values of k.

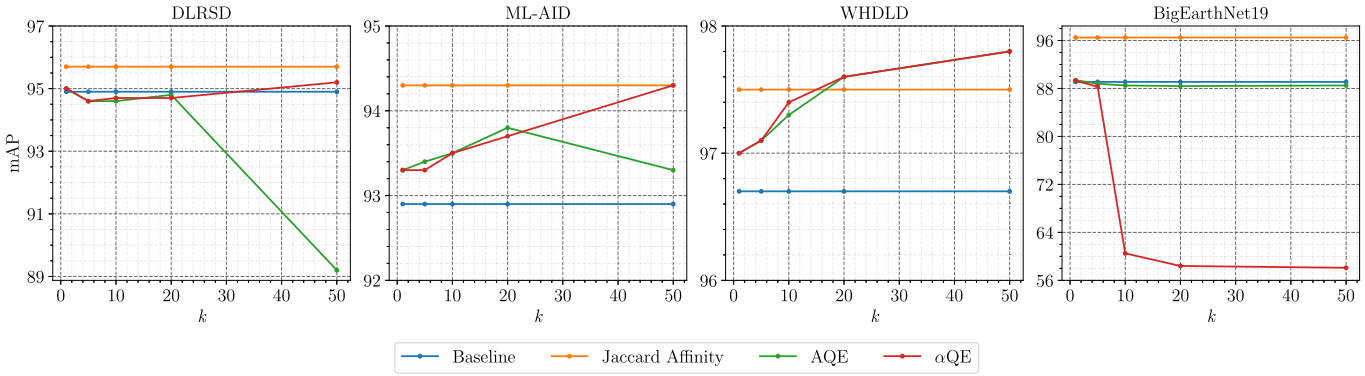

Fig. 6. *n*DCG performance of the four datasets. The baseline performance of OML loss is in blue, AQE in green, αQE in red, and Jaccard Affinity in orange. The Jaccard Affinity consistently improves performance, whereas both AQE and αQE offer inconsistent and unreliable performance.

TABLE V
RETRIEVAL RESULTS ON THE BIGEARTHNET19 DATASET. BOLD INDICATES THE HIGHEST SCORE

	mAP			nDCG@100	wAP@100
	Easy	Medium	Hard		
Contrastive	34.1	19.8	13.9	75.7	1.20.
Triplet	45.2	27.9	18.0	85.1	1.71
Cross-Entropy	**52.9**	**35.9**	**24.1**	**89.7**	**1.97**
OML	52.1	33.5	21.7	89.1	**1.97**
with Jaccard Affinity Re-ranking					
Contrastive	39.6	24.9	17.4	74.7	1.10
Triplet	62.9	48.6	39.0	94.0	1.92
Cross-Entropy	**82.0**	**73.2**	**66.5**	**97.8**	**2.39**
OML	72.0	59.3	49.6	96.5	2.20.

be carefully selected to ensure the best possible performance. AQE is controlled by a single parameter k that determines the number of images used in the construction of the new query. Meanwhile, αQE also requires tuning the α parameter, which serves to regulate the impact of image similarity when generating the query. In Figs. 5 and 6, the *m*AP for the *Medium* protocol and the *n*DCG are plotted for various values of k. We have selected an α value of 2, as it yields the best overall performance. Regarding the *m*AP, we observe a significant decline with growing values of k for both QE techniques. This is due to the introduction of unrelated classes mixing with the original query. This aspect reduces the descriptiveness of the query vector since it becomes polluted with unwanted information that lowers performance.

Furthermore, AQE shows a marked decrement in performance for larger k values in the DLRSD and ML-AID datasets, unlike its αQE counterpart. The weight assigned by the similarity of query and top matches powered by αQE adds a strong penalization to the renewed query descriptor. This reduces the impact of less relevant matches. We conjecture that this behavior is not present in the WHDLD due to the low number of classes and the large average number of classes per image. As several images have large label overlap, the vector becomes more descriptive, improving the overall retrieval performance. Regardless, both query expansion methods are ill-suited, as they perform below the baseline. It should be noticed that the performance using the QE family of techniques requires tuning of their hyperparameters. They vary depending on the dataset, whereas Jaccard Affinity reranking is dataset-agnostic. The caveat with all of these methods is that low retrieval performance in the top ranks will produce poor results as they are highly sensitive. Nevertheless, the Jaccard Affinity technique yields significant *m*AP gains in comparison to QE techniques.

On BigEarthNet19, we observe that both query expansion methods perform marginally better than baseline or lower. The αQE method performs significantly lower for large values of k. This is because, even though feature similarity with the query is high, the label similarity is low. When the query is renewed with irrelevant top matches, this scaling with feature similarity significantly lowers performance. As AQE is not scaled with

feature similarity, the impact of the performance drop is not as high as with αQE.

The *m*AP metric provides a holistic view of performance. However, in a realistic setting, the user of the retrieval system may not be interested in a perfect ranking of the dataset but in receiving relevant matches in the top positions. Fig. 6 depicts how the *n*DCG@100 changes for increasing values of k. This metric quantifies how much the ranking of the top-100 images deviates from the ideal sorting. We observe that Jaccard Affinity reranking consistently outperforms other reranking methods in almost all cases. Apart from retaining high *m*AP, Jaccard Affinity maintains the high-quality ordering of the retrieved results. Again, we observe the inconsistent performance of query expansion methods. As an explanation, we speculate that the poor performance of the QE techniques is due to their construction of a new query based on the top-k matches. Neither AQE nor αQE considers the correspondences between images to be nonbinary and can introduce features from classes unrelated to the original query. Such features reduce the descriptiveness of the new vector and can even cause significant topic drift. To assess the performance further, we also evaluated the computation costs on the largest available dataset. The BigEarthNet dataset, with 120 000 images, is utilized for the timing experiments. The Jaccard Affinity reranking took approximately 3.5 ms for retrieving 10 000 queries. The Jaccard Affinity method is fast as it requires only lookup to the top match from the precomputed graph.

While we have proposed an effective reranking technique, it is clear that reranking in ML-RSIR is a complicated problem requiring custom-made solutions. The direct application of existing techniques may not be sufficient for the ML-RSIR case, as evident by the low performance of AQE and αQE. However, exploiting the semantic relationships between different classes may improve the retrieval performance considering that certain objects are likely to appear together (e.g., buildings and pavement).

V. Conclusion

As image retrieval is one of the most vital tasks in RS image understanding, we have extended the DLRSD, ML-AID, WHDLD, and BigEarthNet19 datasets for multilabel image retrieval. We propose a framework that clearly defines the multilabel retrieval task and provides updated evaluation metrics to account for a variable number of labels. We also extend popular metric loss functions such that contrastive and triplet losses are compatible with the multilabel image retrieval task. Apart from extending these loss functions, we also propose a novel differentiable multilabel loss function that accounts for the class and ordering top-ranks based on feature distances. To the best of our knowledge, this is the first multilabel rank learning loss. Furthermore, we develop the Jaccard Affinity reranking technique and compare it against the popular reranking methods in image retrieval. We demonstrate that the OML loss outperforms, or remains competitive with, existing losses popular in the RSIR literature across four different datasets with varying semantic complexity and scale. In addition, it is shown that the Jaccard affinity reranking technique consistently improves the retrieval performance without hyperparameter tuning. We consider that ML-RSIR is an understudied field in RS due to the lack of standardized protocols and its intrinsic complexity. We hope that the framework and techniques presented in this article will foster the interest of the community for multilabel RSIR.

References

[1] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[2] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996.

[3] R. Cao *et al.*, "Enhancing remote sensing image retrieval using a triplet deep metric learning network," *Int. J. Remote Sens.*, vol. 41, no. 2, pp. 740–751, Jan. 2020.

[4] U. Chaudhuri, B. Banerjee, and A. Bhattacharya, "Siamese graph convolutional network for content based remote sensing image retrieval," *Comput. Vis. Image Understand.*, vol. 184, pp. 22–30, Jul. 2019.

[5] Y. shu Liu, Z. Han, C. Chen, L. Ding, and Y. Liu, "Eagle-eyed multitask CNNs for aerial image retrieval and scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 9, pp. 6699–6721, Sep. 2020.

[6] P. Liu, G. Gou, X. Shan, D. Tao, and Q. Zhou, "Global optimal structured embedding learning for remote sensing image retrieval," *Sensors*, vol. 20, no. 1, p. 291, Jan. 2020.

[7] X. Tang, X. Zhang, F. Liu, and L. Jiao, "Unsupervised deep feature learning for remote sensing image retrieval," *Remote Sens.*, vol. 10, no. 8, p. 1243, Aug. 2018.

[8] R. Imbriaco, C. Sebastian, E. Bondarev, and P. de With, "Aggregated deep local features for remote sensing image retrieval," *Remote Sens.*, vol. 11, no. 5, p. 493, Feb. 2019.

[9] R. Imbriaco, T. Alkanat, E. Bondarev, and P. de With, "Multi-branch convolutional descriptors for content-based remote sensing image retrieval," in *Proc. VISIGRAPP*, 2020, pp. 242–249.

[10] A. Schumann, L. Sommer, M. Vogler, and J. Beyerer, "Onthology-based masking loss for improved generalization in remote sensing semantic image retrieval," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.

[11] Y. Liu, Y. Liu, C. Chen, and L. Ding, "Remote-sensing image retrieval with tree-triplet-classification networks," *Neurocomputing*, vol. 405, pp. 48–61, Sep. 2020.

[12] L. Fan, H. Zhao, and H. Zhao, "Distribution consistency loss for large-scale remote sensing image retrieval," *Remote Sens.*, vol. 12, no. 1, p. 175, Jan. 2020.

[13] Y. Yang and S. D. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. GIS*, 2010, pp. 270–279.

[14] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.

[15] Z. Shao, K. Yang, and W. Zhou, "A benchmark dataset for performance evaluation of multi-label remote sensing image retrieval," *Remote Sens.*, vol. 10, no. 6, p. 964, 2018.

[16] O. E. Dai, B. Demir, B. Sankur, and L. Bruzzone, "A novel system for content based retrieval of multi-label remote sensing images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 1744–1747.

[17] Z. Shao, K. Yang, and W. Zhou, "Performance evaluation of single-label and multi-label remote sensing image retrieval using a dense labeling dataset," *Remote Sens.*, vol. 10, no. 6, p. 964, Jun. 2018.

[18] G. Hoxha, F. Melgani, and B. Demir, "Toward remote sensing image retrieval under a deep image captioning perspective," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 4462–4475, 2020.

[19] W. Zhou, S. Newsam, C. Li, and Z. Shao, "PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval," 2017, *arXiv:1706.03424*. [Online]. Available: http://arxiv.org/abs/1706.03424

[20] Y. Liu, C. Chen, Z. Han, L. Ding, and Y. Liu, "High-resolution remote sensing image retrieval based on classification-similarity networks and double fusion," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1119–1133, 2020.

[21] F. Ye, M. Dong, W. Luo, X. Chen, and W. Min, "A new re-ranking method based on convolutional neural network and two image-to-class distances for remote sensing image retrieval," *IEEE Access*, vol. 7, pp. 141498–141507, 2019.

[22] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, pp. 1470–1477.
[23] Q. Bao and P. Guo, "Comparative studies on similarity measures for remote sensing image retrieval," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, vol. 1, Oct. 2004, pp. 1112–1116.
[24] C.-R. Shyu, M. Klaric, G. J. Scott, A. S. Barb, C. H. Davis, and K. Palaniappan, "GeoIRIS: Geospatial information retrieval and indexing system—content mining, semantics modeling, and complex queries," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 4, pp. 839–852, Apr. 2007.
[25] V. Risojević and Z. Babić, "Fusion of global and local descriptors for remote sensing image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 10, no. 4, pp. 836–840, Jul. 2013.
[26] W. Xiong, Z. Xiong, Y. Cui, and Y. Lv, "Deep multi-feature fusion network for remote sensing images," *Remote Sens. Lett.*, vol. 11, no. 6, pp. 563–571, Jun. 2020.
[27] W. Xiong, Y. Lv, Y. Cui, X. Zhang, and X. Gu, "A discriminative feature learning approach for remote sensing image retrieval," *Remote Sens.*, vol. 11, no. 3, p. 281, Feb. 2019.
[28] G. Sumbul, J. Kang, and B. Demir, "Deep learning for image search and retrieval in large remote sensing archives," 2020, *arXiv:2004.01613*. [Online]. Available: http://arxiv.org/abs/2004.01613
[29] D. Marmanis, M. Datcu, T. Esch, and U. Stilla, "Deep learning Earth observation classification using ImageNet pretrained networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 1, pp. 105–109, Jan. 2016.
[30] H. Jegou, F. Perronnin, M. Douze, J. Sánchez, P. Perez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, Sep. 2012.
[31] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 1735–1742.
[32] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1655–1668, Jul. 2019.
[33] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*. [Online]. Available: http://arxiv.org/abs/1703.07737
[34] E. Hoffer and N. Ailon, "Deep metric learning using triplet network," in *Proc. SIMBAD*, 2015, pp. 84–92.
[35] M. Kaya and H. S. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, Aug. 2019.
[36] A. Brown *et al.*, "Smooth-ap: Smoothing the path towards large-scale image retrieval," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020.
[37] F. Cakir, K. He, X. Xia, B. Kulis, and S. Sclaroff, "Deep metric learning to rank," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1861–1870.
[38] J. Revaud, J. Almazán, R. S. Rezende, and C. R. D. Souza, "Learning with average precision: Training image retrieval with a listwise loss," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 5107–5116.
[39] S. Srivastava, J. E. Vargas-Muñoz, D. Swinkels, and D. Tuia, "Multilabel building functions classification from ground pictures using convolutional neural networks," in *Proc. 2nd ACM SIGSPATIAL Int. Workshop AI Geographic Knowl. Discovery (GeoAI)*, New York, NY, USA: Association for Computing Machinery, 2018, pp. 43–46, doi: 10.1145/3281548.3281559.
[40] Y. Hua, S. Lobry, L. Mou, D. Tuia, and X. X. Zhu, "Learning multi-label aerial image classification under label noise: A regularization approach using word embeddings," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Sep. 2020, pp. 525–528.
[41] H. Lai, P. Yan, X. Shu, Y. Wei, and S. Yan, "Instance-aware hashing for multi-label image retrieval," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2469–2479, Jun. 2016.
[42] Z. Zhang, Q. Zou, Y. Lin, L. Chen, and S. Wang, "Improved deep hashing with soft pairwise similarity for multi-label image retrieval," *IEEE Trans. Multimedia*, vol. 22, no. 2, pp. 540–553, Feb. 2020.
[43] Y. Xie, Y. Liu, Y. Wang, L. Gao, P. Wang, and K. Zhou, "Label-attended hashing for multi-label image retrieval," in *Proc. IJCAI*, Jul. 2020, pp. 955–962.
[44] D. Wu, Z. Lin, B. Li, M. Ye, and W. Wang, "Deep supervised hashing for multi-label and large-scale image retrieval," in *Proc. ACM Int. Conf. Multimedia Retr.*, 2017, pp. 150–158.
[45] G. Song and X. Tan, "Deep code operation network for multi-label image retrieval," *Comput. Vis. Image Understand.*, vol. 193, Apr. 2020, Art. no. 102916.
[46] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, vol. 2, Sep. 1999, pp. 1150–1157.
[47] F. Omruuzun, B. Demir, L. Bruzzone, and Y. Y. Cetin, "Content based hyperspectral image retrieval using bag of endmembers image descriptors," in *Proc. 8th Workshop Hyperspectral Image Signal Process. Evol. Remote Sens. (WHISPERS)*, Aug. 2016, pp. 1–4.
[48] 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.
[49] Z. Shao, W. Zhou, X. Deng, M. Zhang, and Q. Cheng, "Multilabel remote sensing image retrieval based on fully convolutional network," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 318–328, 2020.
[50] Y. Hua, L. Mou, and X. X. Zhu, "Relation network for multilabel aerial image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4558–4572, Jul. 2020.
[51] W. Zhou, X. Deng, and Z. Shao, "Region convolutional features for multi-label remote sensing image retrieval," 2018, *arXiv:1807.08634*. [Online]. Available: https://arxiv.org/abs/1807.08634
[52] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman, "Total recall: Automatic query expansion with a generative feature model for object retrieval," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[53] O. Chum, A. Mikulík, M. Perdoch, and J. E. S. Matas, "Total recall II: Query expansion revisited," in *Proc. CVPR*, 2011, pp. 889–896.
[54] D. C. Guimarães Pedronette, J. Almeida, and R. da S. Torres, "A scalable re-ranking method for content-based image retrieval," *Inf. Sci.*, vol. 265, pp. 91–104, May 2014.
[55] X. Tang, L. Jiao, W. J. Emery, F. Liu, and D. Zhang, "Two-stage reranking for remote sensing image retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 10, pp. 5798–5817, Oct. 2017.
[56] G. Sumbul *et al.*, "BigEarthNet dataset with a new class-nomenclature for remote sensing image understanding," 2020, *arXiv:2001.06372*. [Online]. Available: https://arxiv.org/abs/2001.06372
[57] F. Radenović, A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Revisiting Oxford and Paris: Large-scale image retrieval benchmarking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5706–5715.
[58] 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.
[59] Y. Li, Y. Zhang, X. Huang, H. Zhu, and J. Ma, "Large-scale remote sensing image retrieval by deep hashing neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 950–965, Feb. 2018.
[60] G.-S. Xia *et al.*, "AID: A benchmark data set for performance evaluation of aerial scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 7, pp. 3965–3981, Jul. 2017.
[61] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, "Bigearthnet: A large-scale benchmark archive for remote sensing image understanding," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 5901–5904.
[62] R. Baeza-Yates *et al.*, *Modern Information Retrieval*, vol. 463. New York, NY, USA: ACM Press, 1999.
[63] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Trans. Inf. Syst.*, vol. 20, no. 4, pp. 422–446, Oct. 2002.
[64] F. Zhao, Y. Huang, L. Wang, and T. Tan, "Deep semantic ranking based hashing for multi-label image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1556–1564.
[65] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[66] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: https://arxiv.org/abs/1412.6980

Raffaele Imbriaco received the B.Sc. degree in 2013, M.Sc. degree from the Eindhoven University of Technology (TU/e), Eindhoven, The Netherlands, after concluding his research project on X-ray imaging at Philips Healthcare, where he is pursuing the Ph.D. degree in electrical engineering.

His research interests include deep learning, image retrieval, and visual place recognition. He is one of the researchers involved in the Public Safety and Crisis Management Service Orchestration (PSCRIMSON) Project.

Clint Sebastian received the M.Sc. degree in electrical engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2016, where he is pursuing the Ph.D. degree in computer vision with the Video Coding and Architectures Research Group.

Following this, he worked with Cyclomedia, Zaltbommel, The Netherlands, as a Computer Vision Engineer working on object detection, image segmentation, and generative modeling on large-scale datasets. He is the author and a coauthor of several scientific publications in journals and international conferences. His research interests include object detection, scene segmentation, image retrieval, and reidentification using deep learning.

Egor Bondarev received the M.Sc. degree in robotics and informatics from State Polytechnic University, Minsk, Belarus Republic, in 1997, and the Ph.D. degree in computer science from the Eindhoven University of Technology (TU/e), Eindhoven, The Netherlands, in 2009, in the research domain of performance predictions of real-time component-based systems on multiprocessor architectures.

He is an Assistant Professor with the Video Coding and Architectures Group, TU/e, focusing on such research areas as multimodal sensor fusion, smart surveillance with multicamera systems, and photorealistic 3-D reconstruction of environments. He has written and coauthored over 50 publications on real-time computer vision and image/3-D processing algorithms. He is the Leader of an internal research cluster on real-time data fusion from multimodal sensors, such as thermal, depth, laser, RGB, and acoustic. He is involved in several European research projects, and he is a TU/e Project Leader in the large international PANORAMA, Public Safety and Crisis Management Service Orchestration (PSCRIMSON), and APPS projects, all addressing challenges of multimodal multicamera smart surveillance.

Peter H. N. de With (Fellow, IEEE) is an International Expert on video compression and video and image analysis for health, surveillance, and multimedia. He has been active in the Research and Development of video algorithms and systems design for about 35 years and held positions at Philips Research, Logica Consulting Management Company (CMG) (now CGI), CycloMedia, Zaltbommel, and The Netherlands and was/is an Advisor to other companies. Since 2000, he has been leading the Video Coding Architectures Group [Signal Processing Systems (SPS)-VCA], which focuses on video/image signal processing, signal transformation, feature analysis, and machine learning techniques for the application areas of health, surveillance, and automotive. For safety and security, he has been involved in multiple EU projects on video analysis, object, and behavior recognition and in surveillance projects with the Dutch Defense, Bosch Security Systems, TKH-Security, ViNotion, and more.

Dr. de With is a member of the Royal Holland Society of Academic Sciences and Humanities, has coauthored over 600 articles on video coding, analysis, architectures, and 3-D processing, and has received multiple articles awards. He is the Program Committee Member of the IEEE Consumer Electronic Show (CES) and the International Conference on Image Processing (ICIP) and holds some 30 patents.