Multiple Instance Classification or: How I Learned to Evaluate Local Image Descriptors

Ilaria Bartolini, Pietro Pascarella, and Marco Patella
DISI - Alma Mater Studiorum, Università di Bologna, Italy
Email: {ilaria.bartolini,pietro.pascarella,marco.patella}@unibo.it

Abstract—**Multiple instance classification (MIC) is a kind of supervised learning, where data are represented as *bags* and each bag contains many *instances*. Training bags are given a label and the system tries to learn how to label bags, without necessarily learning how to label each instance individually. In this paper, we apply concepts drawn from MIC to the realm of content-based image retrieval, where images are described as bags of visual *local descriptors*. In particular, we purport the use of classifiers, following the different MIC paradigms, to evaluate the effectiveness of *any* local descriptor.**

1. Introduction

Content-based image retrieval (CBIR) consists in searching for images of interest in large databases, exploiting their visual content, as opposed to concept-based image indexing, which applies text-based techniques for indexing images, using image captions, surrounding text, keywords, and so on [15]. CBIR can be used per se, e.g., to search for a particular image in an image dataset (a notable example is the Google Images system, images.google.com), or as a building block for other image-related tasks, like browsing [5], annotation [4], classification [1], and so on.

The fundamental concept in CBIR is that of similarity, which is used to compare the image content. Evaluating the similarity between two images involves: (1) automatically extracting relevant *features/descriptors*, summarizing visual content of each image, and (2) compare such features to assess a *similarity score* in $[0, 1]$, with the understanding that higher values indicate high degrees of similarity between images' content.

Approaches to extraction of image features can be broadly classified in global (where descriptors represent visual characteristics of the image as a whole) and local (where features describe visual characteristics of a small set of image pixels), with local features having a major prevalence in recent approaches. Another fundamental ingredient for CBIR is efficient indexing, due to the facts that image databases are usually (very) large and that often a real-time query processing is required. However, as acknowledged in [11], "research on efficient ways to index images by content has been largely overshadowed by research on efficient visual representation and similarity measures".

One of the most common ways to measure the effectiveness of any CBIR technique is *classification*, i.e., identifying to which of a set of categories a new image belongs, given a (training) set of images for which the category membership is known. Indeed, as acknowledged in [12]: "A feature that performs well for the task of classification on a certain data set, it will most probably be a good choice for retrieval of images from that data set, too." Despite the importance of this task, most of the approaches have only focused on establishing the accuracy of image content descriptors (*features*), with a negligent lack of emphasis on classifying techniques and almost no interest to efficiency. The former issue is even more prominent, due to the increasing usage of local features, which opens the way to a plethora of more advanced classification techniques. The latter problem is of utmost importance for those approaches using *lazy learning*, i.e., for which the training data are generalized only (or mostly) when a query is made to the system (when a new image is to be classified). In such cases, the training phase is quite fast, while evaluation is the more costly part of classification (as opposed to *eager learning*, where the opposite happens). We will show that methods for retrieval and classification are inherently entangled and that all approaches presented here are applicable to any local image descriptor, making them a valid tool for establishing the effectiveness and efficiency of their proposed features.

To overcome the deficiencies outlined above, we propose to combine the realms of multiple instance classification (MIC) and content-based image retrieval, by applying multiple MIC techniques to the task of image classification. To the best of our knowledge, this is the first attempt to combine the world of multiple instance classification to the task of image classification *in a comprehensive way*. Indeed, a few previous attempts [1], [9], [18], [20], [21] have used approaches drawn from MIC for classifying images, but without putting them into the proper context. Moreover, among the approaches using lazy learning, we are the first to put an emphasis on efficiency of the classification. Our goal here is not to propose a novel technique for image classification; rather, we would like to show how the introduction of concepts drawn from Artificial Intelligence could help researchers working in CBIR to evaluate their proposed features and/or indexing techniques in a more structured way, by showing them the existing alternatives.

All approaches we introduce here have been implemented on top of the WINDSURF framework [7], providing us a number of algorithms and indexing data structures for

978-1-7281-5606-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ISM.2019.00031

efficient query processing. In this way, we have been able to abstract from the underlying details of feature extraction, data indexing, etc., and to focus on MIC algorithms.

2. Background on the WINDSURF Framework

WINDSURF [7] is a framework (and software library[1]) for the management of multimedia (MM) hierarchical data. The fundamental concept of the WINDSURF framework is that of MM documents that are made of several component *elements*. Although the WINDSURF model is general enough to encompass several types of MM documents (e.g., videos, time series, web pages), for the scope of this paper, we shall restrict ourselves to the domain of still images. The retrieval model of WINDSURF can then be described as follows: we have a database $\mathcal{D}$ of N images, $\mathcal{D} = \{I_1, \ldots, I_N\}$, where each image I is composed of n_I *elements*, $I = \{R_1, \ldots, R_{n_I}\}$. Each element R is described by way of *features* that represent, in an appropriate way, the content of R. Given a query image $Q = \{Q_1, \ldots, Q_n\}$ composed of n elements, and an element distance function δ, that measures the dissimilarity of a given pair of elements (using their features), we want to determine the set of *best* images in $\mathcal{D}$ with respect to Q. Clearly, WINDSURF is also able to handle the simpler case of global features, when images are described by way of a single (global) descriptor.

In the original incarnation of WINDSURF [3], elements correspond to regions, i.e., sets of image pixels R that share the same visual content (color & texture). Such regions are obtained by clustering together pixels of the original image and a 37-D descriptor (feature) is extracted for each region. Region descriptors are then compared using the metric Bhattacharyya distance. Recently, WINDSURF has also been extended to deal with salient point descriptors [6]. The concept of salient point descriptors has been originated in the field of computer vision [17], where an image is summarized by way of a set of local features to allow fast matching between the image itself and existing models/patterns. Features are extracted from a subset of the image pixels (the *salient points*) that are considered relevant as they contain most of the image information.

In order to allow efficient resolution of similarity queries, WINDSURF supports indices built on top of both elements and documents: this allows the definition of alternative query processing algorithms (like those described in this paper). In particular, an implementation of the M-tree index [10] is included, allowing efficient resolution of k-NN and range queries and providing sorted access to indexed elements/documents, i.e., to output data in increasing order of distance with respect to the object with which the index is queried.

3. Multiple Instance Classification and its Application to Images

Multiple instance learning [13] is a branch of supervised learning where, instead of a training set of objects, the learner receives a training set of *bags*, each containing multiple *instances*. Multiple instance classification (MIC) [2] is the name given to the sub-field of MIL focused on classification. MIC includes a number of classification techniques that exploit the fact that the class of each individual bag can be transferred to all (or to some) of its instances.

In image classification, the problem tackled in this paper, bags correspond to *images*, containing n *features* (instances) R_i: $I = \{R_1, R_2, \ldots, R_n\}$. The objective is to estimate a classification function $C(I)$ providing the class of I. To this end we are given a *training set* $\mathcal{T}$ of M images with corresponding class, $\mathcal{T} = \{(I_1, C_1), (I_2, C_2), \ \ldots, (I_M, C_M)\}$, where C_i is the class of image I_i. From the point of view of MIL, techniques differ in the assumption regarding how the class of each bag is related to the bag instances [14]. The excellent review paper [2] provides the following taxonomy:

- Instance Space (IS): it is assumed that the discriminative information lies at the instance level, so that classification is performed on instances and the overall classification is performed by aggregating scores obtained at the instance level.
- Bag Space (BS): the main assumption is that discriminative information lies at the bag level, and this cannot be distributed to instances.
- Embedded Space (ES): each bag is mapped to a single feature vector, summarizing the relevant information included at the instance level, then a vector-based classifier is exploited.

As we will show in the following sections, each of the three alternatives suggests a retrieval model, based on instances (features), bags (images), and vectors, respectively. This also helps researchers proposing novel local features for characterizing image content to define their appropriate model.

3.1. Instance Space Classification

3.1.1. Collective Assumption. These methods are based on the assumption that all instances in a bag contribute equally to the bag class. In this case the bag class can be estimated by choosing the class maximizing a simple (weighted) average.

This was implemented using a two-step confidence-rated IS classifier:

1) First, each image feature R is classified using a feature-level classifier $c(R)$; the classifier also computes a value $\nu(R)$ representing the confidence that c has on its choice $c(R)$.
2) Then, the whole image is classified taking into consideration the class assigned to each of its features.

[1] The Windsurf library is written in Java and is released under the "QPL" license, being freely available at URI www-db.disi.unibo.it/Windsurf/ for education and research purposes only.

In particular, for any class C_j a score $s_j(I)$ is computed for image I as the sum of confidences of features classified to each class:

$$s_j(I) = \sum_{R \in I : c(R) = C_j} \nu(R) \tag{1}$$

Then I is classified to the class maximizing the value in Equation 1:

$$C(I) = \arg\max_j s_j(I) \tag{2}$$

1-NN classifier - $\mathtt{IS}$:. The classifier of feature R and the corresponding score are defined as follows, taking into account the nearest neighbor of R only:

$$c(R) = c(NN_1(R)) \qquad \nu(R) = sim(R, NN_1(R))$$

This classifier equals the one called Φ^f in [1]. Efficient retrieval of $NN_1(R)$ is guaranteed by oerforming a 1-NN query on the feature-based indices.

Local classifier - $\mathtt{IS}_{\mathtt{L}}$:. The only difference with respect to the previous classifier is the score which is defined as follows:

$$c(R) = c(NN_1(R)) \quad \nu(R) = \begin{cases} 1 & \text{if } 1 - \frac{sim(R, NN_{\bar{1}}(R))}{sim(R, NN_1(R))} > c \\ 0 & \text{otherwise} \end{cases}$$

where $NN_{\bar{1}}(R)$ is the nearest neighbor of R of a class different than that of $NN_1(R)$ ([1] calls this classifier Φ^m). The efficient evaluation of this approach is obtained through a sorted access for each query feature R, retrieving instances until a result is obtained whose class is different to that of $NN_1(R)$.

Weighted local classifier - $\mathtt{IS}_{\mathtt{WL}}$:. The classifier of feature R and the corresponding score are defined as follows, taking into account the nearest neighbor of R only:

$$c(R) = c(NN_1(R)) \qquad \nu(R) = 1 - \frac{sim(R, NN_{\bar{1}}(R))}{sim(R, NN_1(R))}$$

With respect to the previous classifier, here the confidence value is not binary, but uses the original "fuzzy" ratio between similarities of the NN of R and of the NN of a different class. In [1], this classifier was denoted as Φ^w. Again, sorted access on a region index guarantees efficient evaluation.

Weighted k-NN classifier - $\mathtt{IS}_{\mathtt{W}}$:. First, a score for each class is defined according to classes of the nearest k neighbors of feature R:

$$s_j(R) = \sum_{i=1,\ldots,k : c(NN_i(R)) = C_j} sim(R, NN_i(R))$$

Then, the classifier and the score are defined as:

$$c(R) = \arg\max_j s_j(R) \qquad \nu(R) = 1 - \frac{\overline{\max} s_j(R)}{\max s_j(R)}$$

where $\overline{\max} s_j(R)$ is the score of the second best class for R. Note that this definition of confidence is coherent with the one defined for the weighted local classifier (the two definitions coincide for $k = 1$), but it is different with respect to the one given for the 1-NN classifier. This classifier was named Φ^k in [1]. Efficient evaluation is obtained by way of a k-NN query on the feature index.

3.1.2. Standard Multiple Instance Assumption. Methods in this category suppose that instances of each class are only contained in bags of the same class and that every bag contains at least one instance of its class (SMI assumption). This is equal to say that, for each bag, one of the instances possesses some "desirable" property making the whole bag of that class, thus we are trying to identify which instance type characterizes each class.

The SMI assumption ($\mathtt{SMI}$) was implemented by removing, from the training set, all those instances that would lead to a wrong classification, i.e., whose NN belongs to a different class.[2] This approach has also the advantage of reducing the ground truth size, as also acknowledged in [1], where this approach was called "local features cleaning". Finally, any IS classifier can be exploited on the reduced ground truth to classify each instance.

3.2. Bag Space Classification

Techniques following this paradigm consider each bag as a whole, so that the classification is performed in the space of bags. Typically, methods in this category exploit a distance $d(I_i, I_j)$ obtained by appropriately aggregating distances between correspondent features $\delta(R_{i,h}, R_{j,k})$. Examples, all of which have been implemented here, include the EMD distance ($\mathtt{BS}_{\mathtt{EMD}}$) [16], [19], the Hausdorff distance ($\mathtt{BS}_{\mathtt{Haus}}$), and the Chamfer distance ($\mathtt{BS}_{\mathtt{Cham}}$) [8]. It has to be noted that the distances also differ in their time complexity, since the Chamfer and Hausdorff distances are quadratic in the number of instances, while EMD is super-cubic [16] thus it is extremely time consuming, particularly for large bags (as in the case of SIFT salient point descriptors).

Alternatively, one can use a kernel function $K(I_i, I_j) \in [0, 1]$ assessing the similarity between images I_i and I_j. Kernel- and distance-based classifiers can be used interchangeably by transforming distance to similarity and vice versa. In our case, since all distances are normalized in $[0, 1]$, the transformation needed to use a kernel-based classifier is $K(I_i, I_j) = 1 - d(I_i, I_j)$.

This classifier is called Φ^s in [1]. The retrieval model is the classical one used in CBIR, where images are retrieved for decreasing values of their similarity to the query image and efficient evaluation is obtained exploiting an image-based index. A 1-NN classifier has been used for all implemented alternatives.

3.3. Embedded Space Classification

Methods in this category map each image I to a K-dimensional vector v then exploit a K-dimensional classifier on the so-obtained vector. Mapping from I to v is usually performed by way of a vocabulary V, i.e., a set of K words $V = \{(w_1, p_1), (w_2, p_2), \ldots, (w_K, p_K)\}$, where each word is characterized by an identifier w and a prototype

[2] Actually, we removed the instance from the training set if its NN in a different bag is in a different class. This was required because it could happen that the NN of an instance belongs to the same bag.

instance p. The mapping function $\mathcal{M}$, given an image I and a vocabulary V, produces a K-dimensional vector v, $\mathcal{M}(I, V) = v$.

The ES approach makes sense basically whenever the number of instances in a bag is so high to make IS and BS classification (and retrieval) impractical. The retrieval model here consists of comparing image histograms using a vectorial distance (we implemented the simple Euclidean metric) and indexing histograms using a spatial index.

3.3.1. Histogram-Based Methods. These techniques consider that each component of the vector v is obtained as the average value (for that component) of features in I:

$$v_j = \frac{1}{N} \sum_{R \in I} f_j(R), \quad j \in [1, K] \tag{3}$$

where $f_j(R)$ measures the probability that feature R corresponds to word w_j.

Possible implementations for f_j are:

- Bag-of-words with hard assignment: $f_j(R) = 1 \Leftrightarrow j = \arg\min_i \delta(R, p_i)$, otherwise $f_j(R) = 0$. This way, each feature is assigned to one and only one word and the j-th component of v counts how many features of I are assigned to word w_j.
- Bag-of-words with soft assignment: $f_j(R) = 1 - \delta(R, p_i)$. $f_j(R)$ represents the similarity between feature R and word w_j and the j-th component of v represents the average similarity of features in I to word w_j.[3]

3.3.2. Distance-Based Methods. These techniques consider that each component of the vector v is obtained as the matching degree between features in I and the corresponding word:

$$v_j = \min_{R \in I} \delta(R, p_j), \quad j \in [1, K] \tag{4}$$

We implemented all three different methods: histogram-based with hard assignment ($\mathtt{ES}_\mathtt{H}$), histogram-based with soft assignment ($\mathtt{ES}_\mathtt{S}$), and distance-based ($\mathtt{ES}_\mathtt{D}$). Finally, we used the so-called bag-of-visual-words ($\mathtt{ES}_\mathtt{BOVW}$) approach, where the size K of the vocabulary is (much) higher than the number of instances in each bag. In this scenario, which is the one commonly used for salient point descriptors, the hard assignment is used, each image is represented as a sparse vector and the Hamming distance is used to compare histograms. Note that this is the de-facto standard for salient point descriptors, for which vector quantization is used to deal with the high number of descriptors in each image (usually, in the order of hundreds) and with the high dimensionality (64-128) of each descriptor.

[3] In the general case of non-normalized distances, it is $f_j(R) = e^{-\frac{\delta(R,p_i)^2}{\sigma^2}}$, where σ is a parameter.

4. Final Discussion

We introduced a number of concepts drawn from multiple instance learning, showing how they can be successfully applied to the important task of image classification. In the future, we plan to expand the presented approaches to the task of image annotation, where multiple labels are (semi-)automatically assigned to an image [4]. Such task is a multi-class image classification, where the number of categories can be very large (as large as the vocabulary size). To this end, it would be interesting to pursue the approach of [20], where a dual multiple instance assumption is adopted, since each image is considered a bag of both features *and* labels.

References

[1] G. Amato and F. Falchi. kNN Based Image Classification Relying on Local Feature Similarity. *SISAP '10.*

[2] J. Amores. Multiple Instance Classification: Review, Taxonomy and Comparative Study. In *Art. Int.*, 201, 2013.

[3] S. Ardizzoni, I. Bartolini, and M. Patella. Windsurf: Region-Based Image Retrieval Using Wavelets. *IWOSS '99.*

[4] I. Bartolini and P. Ciaccia. Imagination: Exploiting Link Analysis for Accurate Image Annotation. *AMR '07.*

[5] I. Bartolini, P. Ciaccia, and M. Patella. Adaptively Browsing Image Databases with PIBE. In *MTAP*, 31(3), 2006.

[6] I. Bartolini and M. Patella. Windsurf: The Best Way to SURF (and SIFT/BRISK/ORB/FREAK, too). In *Mult. Sys.*, 24(4), 2018.

[7] I. Bartolini, M. Patella, and G. Stromei. The Windsurf Library for the Efficient Retrieval of Multimedia Hierarchical Data. *SIGMAP '11.*

[8] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. In *TPAMI*, 24(24), 2002.

[9] Y. Chen, J. Bi, and J.Z. Wang. MILES: Multiple-Instance Learning via Embedded Instance Selection. In *TPAMI*, 28(12), 2006.

[10] P. Ciaccia, M. Patella, and P. Zezula. M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. *VLDB '97.*

[11] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image Retrieval: Ideas, Influences, and Trends of the New Age. In *Comp. Sur.*, 40(2), 2008.

[12] T. Deselaers, D. Keysers, and H. Ney. Features for Image Retrieval: An Experimental Comparison. In *Inf. Retr.*, 11(2), 2008.

[13] T.G. Dietterich, R.H. Lathrop, and T. Lozano-Pérez. Solving the Multiple Instance Problem with Axis-Parallel Rectangles. In *Art. Int.*, 89 (1–2), 1997.

[14] J. Foulds and E. Frank. A Review of Multi-Instance Learning Assumptions. In *Know. Eng. Rev.*, 25(1), 2010.

[15] M.S. Lew, N. Sebe, C. Djeraba, and R. Jain. Content-Based Multimedia Information Retrieval: State of the Art and Challenges. In *TMCCA*, 2(1), 2006.

[16] H. Ling and K. Okada. An Efficient Earth Mover's Distance Algorithm for Robust Histogram Comparison. In *TPAMI*, 29(5), 2007.

[17] D.G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. In *IJCV*, 60(2), 2004.

[18] O. Maron and A.L. Ratan. Multiple-Instance Learning for Natural Scene Classification. *ICML '98.*

[19] Y. Rubner and C. Tomasi. Perceptual Metrics for Image Database Navigation. Kluwer, Boston, MA, 2000.

[20] J. Wu, Y. Yu, C. Huang, and K. Yu. Deep Multiple Instance Learning for Image Classification and Auto-Annotation. *CVPR '15.*

[21] C. Yang and T. Lozano-Pérez. Image Database Retrieval with Multiple-Instance Learning Techniques. *ICDE '00.*