Neurocomputing 175 (2016) 768–781

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Improving the BoVW via discriminative visual *n*-grams and MKL strategies

A. Pastor López-Monroy [a,*], Manuel Montes-y-Gómez [a], Hugo Jair Escalante [a], Angel Cruz-Roa [b], Fabio A. González [b]

[a] Instituto Nacional de Astrofísica, Óptica y Electrónica, Computer Science Department, Luis Enrique Erro No. 1, Sta. Ma. Tonantzintla, C.P. 72840 Puebla, Mexico

[b] MindLab, Computing Systems and Industrial Engineering Department, Universidad Nacional de Colombia, Cra 30 No 45 03-Ciudad Universitaria, Bogotá DC, Colombia

ARTICLE INFO

Article history:
Received 11 May 2015
Received in revised form
6 September 2015
Accepted 20 October 2015
Communicated by Haowei Liu
Available online 10 November 2015

Keywords:
Visual words
Visual *n*-grams
Image classification
Sequences of visual words
Multiple kernel learning

ABSTRACT

The Bag-of-Visual-Words (BoVW) representation has been widely used to approach a number of different high-level computer vision tasks. The idea behind the BoVW representation is similar to the Bag-of-Words (BoW) used in Natural Language Processing (NLP) tasks: to extract features from the dataset, then build feature histograms that represent each instance. Although the approach is simple and effective facilitating its applicability to a wide range of problems, it inherits a well-known limitation from the traditional BoW; the disregarding of spatial information among extracted features (sequential information in text), which could be useful to capture discriminative visual-patterns. In this paper, we alleviate this limitation with the joint use of visual words and multi-directional sequences of visual words (visual *n*-grams). The contribution of this paper is twofold: (i) to build new simple-effective visual features inspired in the popular idea of *n*-gram representations in NLP and (ii) to propose the Multiple Kernel Learning (MKL) strategies to better exploit the joint use of visual words and visual *n*-grams in Image Classification (IC) tasks. For the former, we propose building a codebook of visual *n*-grams, and use them as attributes to represent images by means of the BoVW representation. For the second point, we consider the visual words and visual *n*-grams as different feature spaces, then we propose MKL strategies to better integrate the visual information. We evaluate our proposal in the image classification task using five different datasets: Histopathology, Birds, Butterflies, Scenes and a subset of 6 classes of CalTech-101. Experimental results show that the proposed strategies exploiting our visual *n*-grams, outperforms or is competitive with (i) the traditional BoVW, (ii) the BoVW using visual *n*-grams under traditional fusion schemes (e.g., ensemble based classifiers) and (iii) other approaches in the literature for IC that consider the spatial context.

1. Introduction

Nowadays the huge amount of digital information available is constantly growing. Much of this information are images generated by image-capturing devices under a wide variety of different domains. All this vast amount of images could be exploited for the benefit of several practical applications, which makes important to have automated tools to assist their analysis. In general, Image Classification (IC) aims to organize images according to predefined categories. IC is one of the most important tasks regarding the organization and analysis of visual information. There are several methods for IC, but the most traditional approach consist in representing images with vectors of visual features, then build classification models using supervised learning algorithms [1].

The representation of images is a key procedure for IC, then a number of different approaches have been proposed so far. The Bag-of-Visual Words (BoVW) [2,1] representation is one of the most used approaches because it is simple and effective, achieving outstanding performance in several computer vision tasks, for example: medical image classification [3,4], category level scene classification [5], object recognition [6], video retrieval [2], image retrieval [7] and human-activity recognition [8]. The core idea behind the BoVW is very similar to the Bag-of-Words (BoW) representation used in text mining tasks (see, e.g., [9]). On one hand, BoW represent documents with vectors, taking each word in the vocabulary as an attribute. On the other hand, the BoVW precomputes a vocabulary of visual words from the training

* Corresponding author at: Instituto Nacional de Astrofísica, Óptica y Electrónica, Computer Science DepartmentDepartment, Luis Enrique Erro No. 1, Sta. Ma. Tonantzintla, C.P. 72840 Puebla, Mexico.
E-mail address: pastor@ccc.inaoep.mx (A.P. López-Monroy).

http://dx.doi.org/10.1016/j.neucom.2015.10.053
0925-2312/

dataset (e.g., clustering vectors of relevant visual features representing parts of images), then represent images with vectors that account for the presence/absence of visual words in images (e.g., histograms of visual words). In computer vision tasks, visual words can play the role of words to identify a particular class/topic [6]. The pure presence/absence of specific visual words can provide valuable information for discriminating between target classes. For example, in face recognition, an eye (or part of it) could be highly informative to recognize a face.

Notwithstanding the success of BoVW, it has a well-known drawback: the disregarding of spatial context among visual words. In specific computer vision tasks, spatial context properly exploited, has been useful to improve the performance of several approaches [10,11]. Thus it is promising to enrich the BoVW by using spatial information. Most of the time at the cost of requiring higher computational resources, the spatial context have been captured in several ways; for example, computing relative [12] or absolute [13] spatial configurations of visual words, or integrating the distance and angle information among specific visual words [11,14]. To capture the spatial context, in this paper we propose an effective feature inspired by one of the most used solutions in NLP for incorporating sequential information in documents representation: *n*-grams (sequences of *n*-words to capture compound word patterns). This type of representation can capture compound item-patterns; for example, in text mining; *united-states, very-good*, etc. In the case of visual imagery, we intend to capture frequent local co-occurrence of visual elements. In this regard, we propose the jointly use of codebooks of visual words and visual *n*-grams (multidirectional sequences of visual words) to represent images under a bag of features formulation. In other words, we propose the extension of the BoVW to the Bag-of-Visual *n*-grams (BoVN),[1] which can be seen as representing images under different feature spaces (e.g., visual words and visual *n*-grams). These different feature spaces could be used together to enhance the performance of classification models, nonetheless this is not a trivial tasks [16]. In this work we propose to exploit two fusion strategies to combine information through Multiple Kernel Learning (MKL) methods. The first one consist in represent images under individual feature spaces (e.g., visual words and visual *n*-grams), then use MKL strategies to exploit the information in the different spaces. The second one consist in representing images under the whole feature space (e.g., visual words and visual *n*-grams), but using different kernel functions to produce different notions of similarity that can be exploited by the proposed MKL strategy. MKL uses similarity kernel functions to delegate the construction of a new combined kernel function to an algorithm [17,18]. Using the latter strategies to fuse information, we perform an extensive experimental work in order to establish a solid framework to exploit the proposed visual *n*-grams.

In this work, we focus on automatic classification using five different image collections; Histopathology, Birds, Butterflies, Scenes and a subset of CalTech-101. These image collections have special particularities like heterogeneous rich visual content, high intra-class variability and complex mixtures of non-localized structural patterns. In particular, a BoVWs representation assumes that there are localized patterns (visual words) which could characterize high-level concepts in the image, but in this paper we go beyond exploiting the usefulness of spatial context captured by sequences of visual words (visual *n*-grams), which encompass a complex mixture of visual patterns that allow us to decide about the class. Experimental results suggest that the proposed framework is a good alternative to other approaches reported in the literature.

The main contributions of this work are twofold: (i) to introduce the effective visual *n*-grams inspired in NLP and (ii) to propose MKL strategies to exploit the joint use of visual words and visual *n*-grams for Image Classification (IC) tasks. This paper is organized as follows. The next section reviews the most relevant work to our proposed solution. Section 3 introduces the proposed methodology to extract visual *n*-grams, whereas Section 3.3 presents the proposed strategies to take advantage of our visual *n*-grams for IC. Sections 4 and 5 present the image collections and experimental settings respectively. Section 6 reports the experimental results we obtained, and Section 7 outlines our conclusions and future avenues of inquiry.

2. Related work

The BoVW approach was introduced by Sivic and Zisserman for tackling the problem of video retrieval [2]. The outstanding performance and simplicity of BoVW quickly became popular and began to expand into computer vision including image classification [4,19,20], image retrieval [1,7,21], object recognition [6], human-activity recognition [8], etc. According to the literature there are several ways to implement the BoVW framework, but the general strategy is as follows: (i) a set of extracted image regions/parts of the training dataset are represented by feature vectors (i.e., SIFT descriptors); (ii) feature vectors are clustered, then cluster centers are taken as the visual words; (iii) visual regions of each image are replaced by the closest visual words in the dictionary, building attribute vectors that represent images (e.g., an histogram of the visual words in the image).

A shortcoming of the BoVW like representations is the overlook of spatial relationships among words. In other words, histograms that account for visual word frequencies do not hold any spatial information about the occurrence of each visual word into the image. The spatial information, correctly exploited, has proven to be useful in several computer vision tasks [12]. Given this scenario, a lot of work has been devoted to capture such spatial relationships among visual words [13,22]. For example, spatial relations have been represented using a graph of visual words in order to describe logos in sports photos [23]. Other efforts have brought ideas from other areas such as NLP. For example, in image retrieval, Zheng et al. [24] proposed the idea of visual phrases by pairwise grouping close or overlapping (according to a threshold) keypoint regions. Since the latter implies to test all keypoints in a one-vs.-rest fashion, they test only on those frequent keypoints in the image dataset. In other works, Yuan et al. [25,26], took advantage of the use of *k*-nearest neighbors algorithm to group visual words and building visual phrases of different lengths in order to get relevant information. In video data mining, visual phrases have also been used for obtaining the principal objects and characters in a video by clustering on viewpoint invariant configurations [27]. Quack et al. [28] have explored local sets of visual words to detect frequent and distinctive features for object classes, this provides the option to use the method for object recognition or as a feature selector. Other approaches have used Language Models (LMs) in order to capture spatial information. A language model is a popular technique used in NLP to model sequences of words. Previous works use LMs for computer vision tasks and perform several steps before training the LM [29], for example, the use of co-occurrence and proximity information of neighbor visual words. The latter is because a LM needs to "read" the visual words in some direction. For example, Tirilly et al. [12] used principal component analysis to project visual descriptors in a particular

[1] In a previous work we introduced an initial idea of building simple sequences of visual words [15]. In this paper we present the full description of our approach studying the generality of visual *n*-grams for several image domains, and appropriated MKL intermediate fusion strategies to jointly exploit the use of visual words and our version of visual *n*-grams).

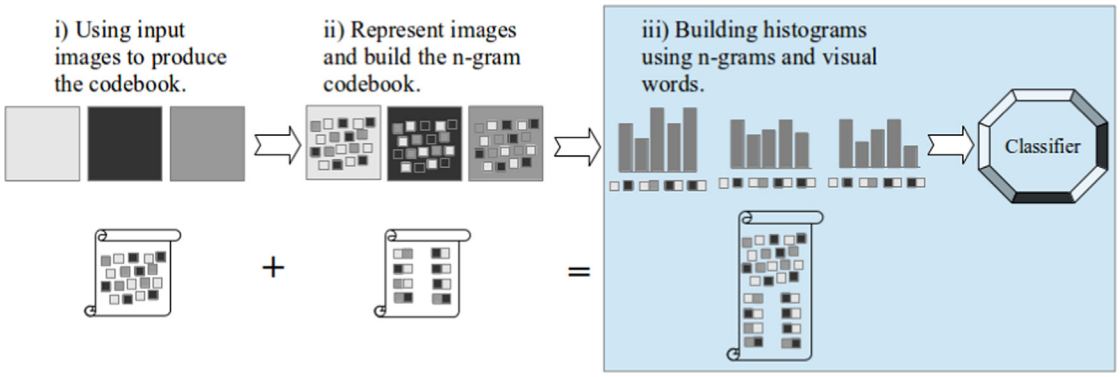

Fig. 1. Image representation through Bag-of-Visual *n*-grams.

direction-axes, then induce a sequence of visual words [12]. Word sequences are classified using a Language Model Classifier (LMC). The LMC builds a LM for each class using the training documents. For testing, they measure the probability of belonging to each LM, and predict the most probable class. Another effective approach to capture spatial relationships among visual elements is the Spatial Pyramid Representation (SPR) proposed in [13]. The core idea of SPR relies in generating sub-windows of an image by using a sequence of increasingly coarser grids defined by a pyramid. For example, in a pyramid of three levels, there are three grids with sizes of 4×4, 2×2 and 1×1 cells. In this way, SPR computes a local BoVW in each cell of the image. The final representation of each image consists of an arrangement of its local histograms. Thus, the comparison of two images is done by using an intersection kernel computed between the representation vectors.

In all previous works authors have proposed interesting extensions to the use of visual words, reporting improvements over the standard BoVW representation. However, these proposals do not necessarily correspond to the way in which sequences of words are processed in NLP tasks for boosting the performance [30]. For instance, LMs are rarely used for text categorization, also it is known that LMC can have problems to handle imbalanced class problems [31]. Other previous works have proven that adding information about occurrences of relative spatial information in visual words can enhance the performance, but at the cost of higher computational complexity, especially when relative distance and angles are considered [12,32]. In this paper we adopt a representation that has proven to be very helpful for text categorization. We focus on the idea of word *n*-grams, which are sequences of *n* words [33] to demonstrate the usefulness and generality of contextual information that could be captured using the analogy of visual *n*-grams. Extracting other kind of features (in our case visual *n*-grams), raises new issues about the way to properly use them inside a classification system. In most previous works, authors have combined the extracted spatial-visual-features in order to improve the performance of their methods, then it is interesting to exploit this kind of information into the final representation. Nonetheless, the most used ways to combine heterogeneous attributes are simple fusion approaches; *early fusion* and *late fusion* [34,33,35]. The main idea of *early fusion* is to concatenate the different feature spaces (e.g., words and *n*-grams) into single vectors, which are fed to a learning method [35,16]. The Support Vector Machine (SVM) has shown to be effective using the *early* BoVW representation [36,4,20]. On the other hand, *late fusion* strategies consider each feature space independently and build an ensemble learning system to combine the outputs of classifiers trained on different inputs (e.g., weighting vote ensemble classifier) [37,35]. The underlying idea is to represent instances using vectors corresponding to each feature space in order to provide different perspectives/views of each instance. The problem of *early–late* fusion approaches is that they can be affected if the feature spaces are not diverse enough [35,16]. Multiple Kernel Learning (MKL), also known as *intermediate fusion*, is an attractive fusion scheme that has shown improvements over typical *early–late* fusion approaches [17], in part for performing the combination of information at a different level; at a *kernel* level. MKL methods build more accurate models using kernel functions that represent different similarity notions of the feature spaces [17]. In the literature there are number of ways to perform the combination of kernel functions, for example, rule based operations (mean or product) over kernel matrices [38], alignment training techniques to weight the contribution of each kernel [39,40], projected gradient updates [41], linear and conic analytical solutions for determining kernel weights [42], etc. In this paper, using different image collections, we evaluate the proposed BoVN representation and propose to exploit different fusion alternatives based on Multiple Kernel Learning to combine features.

3. Image Classification through visual *n*-grams and MKL

In this section we describe the proposed Bag-of-Visual *n*-grams (BoVN) representation for image classification, as well as the proposed MKL fusion strategies. In Fig. 1 we show the general process for generating the BoVN. In the first step we take the whole (training) images and extract the visual words using a standard procedure outlined in Section 3.1. In the second step we extract visual *n*-grams to build a visual *n*-gram codebook (explained in Section 3.2). In the third and final step we merge the visual words codebook and the visual *n*-gram codebook in order to get a final codebook. We use our final codebook to build histograms, which are fed to the proposed strategies through MKL (Section 3.3). Each of these steps are described in the rest of this section.

3.1. Construction of the visual words codebook

In this section we explain the first stage before building the Bag-of-Visual *n*-grams (BoVN) representation. In this context, we first need to build our visual words from the image collection. Such visual words will be the initial features used to generate the visual *n*-grams. In Fig. 2, we show the process to extract the visual words for an image collection using the standard BoVW formulation. We start extracting small patches from the images. For this, we use a regular-grid-based extraction. This is done by partitioning images using a regular grid, and taking each grid item as a patch of fixed size, see step ii) in Fig. 2. The next step consists in representing each extracted patch by a set of features (a visual descriptor). The fourth and last step in the process is the construction of the visual dictionary or visual word codebook. The

codebook is built by clustering all patch descriptors extracted from the image collection. In this process, all similar patch descriptors in the training set are grouped together independently of the source image. The *k*-means algorithm is used in this work to find a set of centroids which represent our visual words, which are labeled by an id and placed in the codebook.

To represent images using the latter codebook, each image is gridded and each image patch is replaced by its closest visual word in the codebook (see Fig. 3). In this way, each image is represented by a histogram that accounts for the occurrence of visual words (from the learned codebook) in the image. In the next section, we show how to use the aforementioned codebook in order to construct visual *n*-grams.

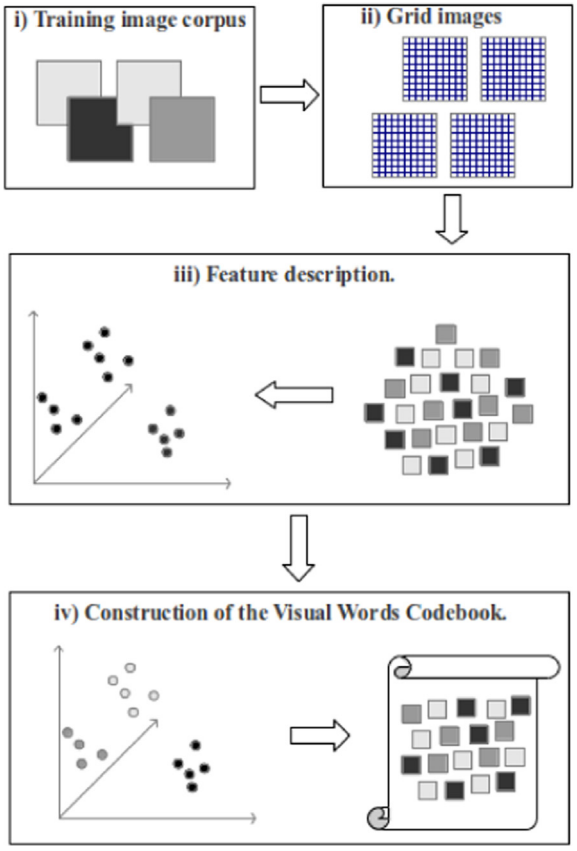

Fig. 2. The process to build a visual word codebook.

3.2. Extracting visual n-grams

In this section we present the second stage to build our visual *n*-grams. As already mentioned, we assume that there is a visual codebook which we will use to represent images. To capture spatial relationships among visual words, we inspired ourselves in the way word *n*-grams are used for text-classification. In NLP, *n*-grams are sequences of *n* consecutive words that help to maintain semantic relationships between words, which allows us to represent compound concepts like “bus stop” with a single attribute. Nonetheless in image domain, the extraction of visual *n*-grams face some additional issues. For example, a document can be read only in one direction, but sequences of image descriptors can be extracted horizontally, vertically, or diagonally). Another problem is to determine the right direction to interpret each visual *n*-gram. For example, 3-grams in text normally can be interpreted correctly only in one direction (say, “the human being”, but not “being human the”). On the other hand, visual 3-grams can have the same order but different orientation if the image is rotated. Therefore, the two descriptor sequences d_a–d_b–d_c and d_c–d_b–d_a might be the same pattern. In this work, we consider such patterns the same, making them rotation invariant.

In order to construct visual *n*-grams we apply the following effective approach. First of all remember that we have each instance represented as the codeword matrix for each image (see Fig. 4). Thus, let A be the $a \times b$ codebook matrix of a given image. The main idea is to produce *n*-grams ignoring the orientation in which they appear. To construct *n*-grams we iterate over each element $a_{i,j}$ of the matrix A and we extract the neighbors in a straight fashion. That is, we extract sequences using items between the items $a_{i,j}$ and $a_{i+k,j+h}$, if and only if they are part of the straight line joining $a_{i,j}$ and $a_{i+k,j+h}$. In Fig. 4 we illustrate the process to extract visual bigrams using a sliding window on each visual word to build its neighbors. This leads to obtain *n*-grams in horizontal, vertical and diagonal directions. The latter condition leaves us with eight possible *n*-grams for each position in the matrix. Finally, each *n*-gram is normalized to be read just in one way and consequently indexed as the same item in our new visual *n*-gram codebook. We use these normalized visual *n*-gram codebook to proceed with the image representation. For this, each image is represented using visual words and visual *n*-grams through histograms of the occurrence of visual *n*-grams found in the image.

Fig. 3. Example of a represented image using the visual word codebook. Left: original image. Right: visual words representation.

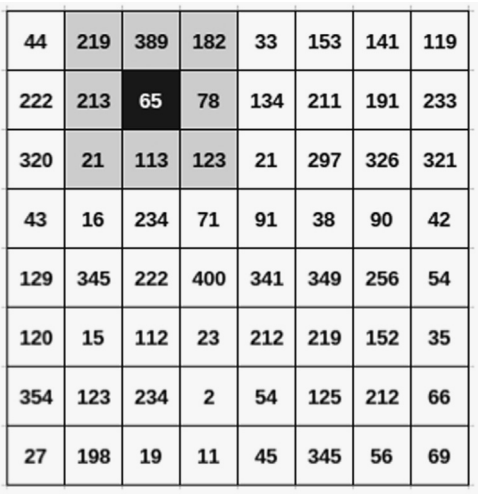

Fig. 4. The process to build a visual *n*-grams using a sliding window. For the dark path (65) the extracted *n*-grams are 65–389, 65–219, 65–213, 65–21, 65–113, 65–123, 65–78, and 65–182.

3.3. Exploiting the jointly use of visual words and visual n-grams

According to previous sections, at this point there are at least two sets of visual features: visual words and the visual *n*-grams. These visual features can be already used to feed a wide range of different classification algorithms. Nonetheless, when instances can be represented under different sets of attributes, there exist several ways to take advantage of those different feature spaces [35]. In this context, we are interested in the following question: *How features coming from visual words and visual n-grams spaces be used together to enhance the performance of classification models?*

The appropriated use of several feature spaces to improve the discriminative power of a system is not a trivial task [16]. Two of the most popular strategies for combining information from different sources are *early-fusion* and *late-fusion* [16,37,43,35]. The former consists of merged attributes from two spaces into single space, then use standard supervised learning methods to build classification models. On the other hand, *late fusion* strategies use classifier ensembles to train individual models on each feature space, then perform a joint prediction using a voting decision or trained combiner. More specifically, the combination of features in early fusion consists in extending the space of visual words *VW* using the visual *n*-grams *VN* space to produce a new space with $|VW|+|VN|$ dimensions. The intuitive idea is that the learning algorithm (in our case SVM) will be able to learn the important properties of the target problem in such space. On the other hand, in late fusion we build an SVM for each space *VW* and *VN*. For this purpose, we implemented each classifier to make a prediction using a vector of the probabilities of belonging to each class. For the final decision, we aggregated such vectors to determine the label as the *i* element with the maximum value. Nonetheless, as shown in experimental results of Section 6.2, sometimes several kinds of textual features could not be diverse enough to build accurately such ensemble models. For example, consider the following example keeping in mind our analogy visual–textual word and visual–textual *n*-gram. In text mining tasks, a wide variety of different kinds of textual features (e.g., words, word *n*-grams, frequent maximal sequences, collocations, etc.) are extracted just from one rigorous modality: the text (sequences of tokens). Thus, the space features could not be totally independent from others especially when one space was used as the base of a new one (e.g., *n*-grams are built from words). Thus, in some classification tasks, ensemble *early/late fusion* methods could not receive truly multimodal features, which degrades the diversity in the feature space and makes difficult to built an accurate ensemble system [16,37,43,35,44]. For this reason, we propose to use two alternative strategies to integrate visual *n*-grams through a more appropriated state-of-the-art scheme fusion. The *intermediate fusion* makes use of Multiple Kernel Learning (MKL) techniques to delegate the construction of a new combined kernel function to an algorithm [17]. Using the latter strategy to fuse information, we establish a solid framework for the use of the proposed visual *n*-grams making it a good alternative to other approaches reported in the literature.

3.3.1. MKL strategies to exploit visual n-gram spaces

According to the literature, the most common/effective classifier under bag of features formulation is the Support Vector Machine (SVM) [34,2,1,36,17]. SVM is a learning algorithm that aims to find an optimal separating hyperplane between instances belonging to two different classes [17]. Let $\{\mathbf{x}_i, y_i\}$ be the training instance-class pairs examples, where $\mathbf{x}_i \in \mathbb{R}^d$ and $y \in \{-1, +1\}$, with d dimensionality of the problem (say the size of the vocabulary). SVMs aim to determine a mapping from training examples to classes using the following linear function:

$$f(\mathbf{x}) = \text{sign}\left(\sum_i \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) - b\right) \quad (1)$$

where α_i and y_i are the weight and label of training example i. To map the $(\mathbf{x}_i, \mathbf{x}_j)$ input vectors into the feature space, the $k(\mathbf{x}_i, \mathbf{x})$ kernel function is applied. Intuitively, $k(\mathbf{x}_i, \mathbf{x})$ measures the similarity between instances x_i and x_j.[2] Selecting the kernel function is an important issue in the training.

Multiple Kernel Learning (MKL) methods are popular solutions to face the problem of combining different feature spaces. The core idea relies in kernel functions; instead of choosing a single kernel function for a specific problem, it is better to have a set and let an algorithm to learn the best combination of them [17]. To better explain this idea consider the following expression which represent combined kernels:

$$k\eta(\mathbf{x}_i, \mathbf{x}_j) = f\eta(\{k_m(\mathbf{x}_i^m, \mathbf{x}_j^m)\}_{m=1}^P | \eta) \quad (2)$$

the core is the combination function, $f\eta : \mathbb{R}^P \to \mathbb{R}$, may be linear or not, the kernel functions, $\{k_m : \mathbb{R}^{D_m} \times \mathbb{R}^{D_m} \to \mathbb{R}\}_{m=1}^P$, take P feature representations of data instances: $\mathbf{x}_i = \{\mathbf{x}_i^m\}_{m=1}^P$ where $\mathbf{x}_j^m \in \mathbb{R}^{D_m}$, and D_m is the dimensionality of the m feature space. η parametrizes the combination function and usually are fixed parameters without any optimization during training.

In order to put MKL framework in context of our extracted visual features, let $V_k = \{w_1, \ldots, w_d\}$ denote the d extracted features in space k (e.g., the codebook of visual words), $\Psi = \{V_1, \ldots, V_P\}$ the set of m considered feature spaces in the whole collection (e.g., codebooks for visual words and visual *n*-grams). In this paper, we propose to fed the $f\eta$ function in Equation (2), using input instances represented through the following two strategies:

(a) Strategy 1: Single kernel – several spaces: A fixed kernel function uses inputs coming from P different space representations (e.g., visual words and visual *n*-grams). The intuitive idea is to represent instances using vectors corresponding to each individual feature space in order to provide different perspectives/views of each image, then use MKL to learn a general perspective. Thus, we end up with $P = |\Psi|$ representations for each data instance: $\mathbf{x}_i = \{\mathbf{x}_i^m\}_{m=1}^P$ where $\mathbf{x}_j^m \in \mathbb{R}^{D_m}$, and D_m is the dimensionality of the m feature space. Having instances represented under P feature spaces is useful to generate diversity in the search space using the fixed kernel

[2] The parameters α and b are learned using optimization techniques [17].

Table 1
Representative MKL algorithms [17].

MKL algorithm	Description
1. SimpleMKL [41]	Iterative MKL algorithm that uses projected gradient updates and trains SVMs at each iteration to solve the optimization problem
2. RBMKL [38]	Rule based MKL trains an SVM with the (mean or product) of the combined kernels
3. NLMKL [45]	A nonlinear MKL algorithm using an SVM is the base learner and a quadratic kernel
4. LMKL [46]	Localized MKL algorithm using the softmax gating model uses the concatenations of all feature representations in the gating model
5. GMKL [47]	The generalized MKL algorithm learns a kernel function instead of kernel matrix defining a kernel funtion in the space of kernels called *hyperkernel*, this uses a convex combination of base kernels
6. GLMKL [39,40]	The group Lasso-based MKL algorithms updates the kernel weights to learn a conic combination of the kernels
7. CABMKL [42]	Centered-alignment-based MKL algorithm. The first step uses a linear analytical solution for determining the kernel weights. The second step trains an SVM with the kernel calculated with these weights
8. ABMKL [48]	Alignment-based MKL algorithms determine kernel weights using a heuristic, then train an SVM with the kernel calculated with these weights

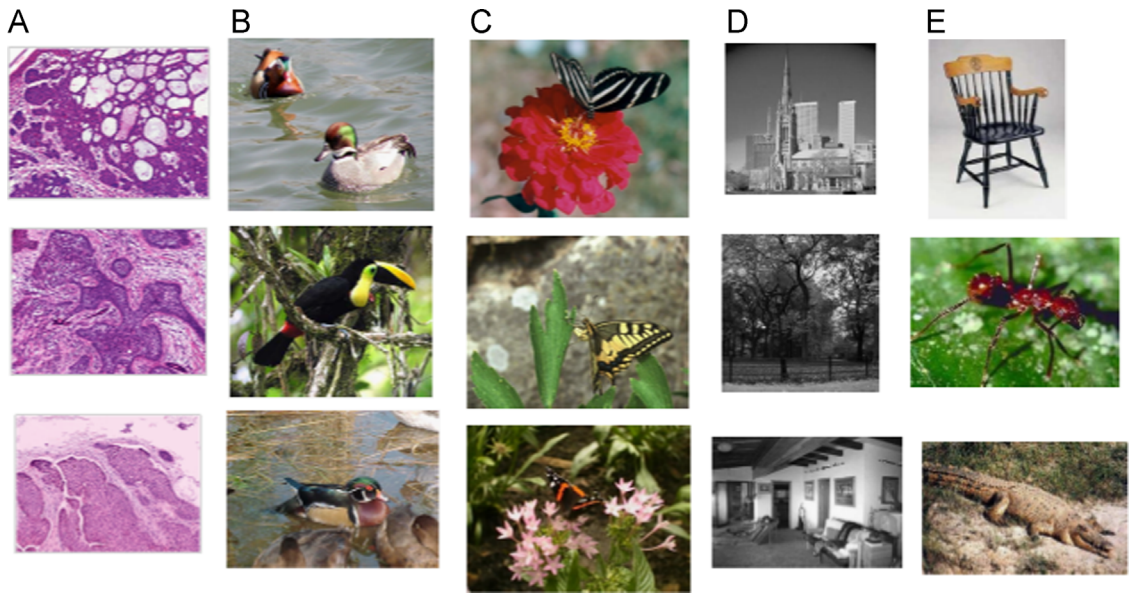

Fig. 5. Image samples of the image collections: (A) Histopathology, (B) Birds, (C) Butterflies, (D) Scenes and (E) 6-Caltech.

function. This allows us to have P different representations that can be used by MKL to build the $k\eta$ general kernel.

(b) Strategy 2: Several kernels – single space: A set of s different kernels functions correspond to different notions of similarity. The whole feature space (visual words and visual n-grams) are used to represent each instance using $P=s$ vector representations. The intuitive idea is that, instead of trying to find which is the best kernel function, a learning method do the picking or combination. In this strategy we represent data instances using vectors of $D=|\bigcup_{V_j \in \Psi} V_j|$ features. Thus, we end up with a vector for each data instance, but using the s kernel functions we produce the $P=s$ representations for each data instance: $\mathbf{x}_i = \{\mathbf{x}_i^m\}_{m=1}^P$ where $\mathbf{x}_j^m \in \mathbb{R}^D$, and D is the dimensionality of the whole feature space. We use those P representations in the $f\eta$ function and build the final kernel $k\eta$.

In order to solve the $f\eta$ for building the final kernel $k\eta$, we use the MKL methods outlined in Table 1. In this way, we analyze the performance of several MKL algorithms to produce a new kernel method that accurately describe each image collection exploiting our precalculated visual n-grams.

4. Image collections

In order to perform the evaluation and demonstrate the generality of visual n-grams we used five different image collections that expose special particularities, which could be captured by visual sequences. Fig. 5 and Table 2 briefly describes each collections. For example, the Histopathology image collection [20,19] is class-imbalanced and contain complex visual patterns in tissues structures (healthy or pathological); the classification is related to pathological lesions and morphological–architectural features which can be captured by visual n-grams. Other collections like Birds, Butterflies and Scenes also have features of texture and structure not only in the target object (e.g., the bird), but also in the other surrounding visual elements like the grass, sky, water; which could play a role to determine or not the class label.

5. Experimental settings

We have performed several experiments for each dataset. In those experiments, we gridded images in patches of 8 pixels.[3] Among the wide variety of image descriptors in the literature, we use the Scale Invariant Feature Transform (SIFT) [52] descriptor extracting edge points at two scales and eight orientations. We also use the discrete cosine transform (DCT) applied to each channel of the RGB color space by patch. The descriptor is built merging the 64 coefficients from each one of the three channels.

[3] We experimentally test patches of size 8×8 and 16×16. The 8×8 size patch is an appropriated option, which has been also confirmed by other authors in the Histopathology dataset [20].

Table 2
Image collections used for the evaluation of visual *n*-grams. Histopathology collection is the only one with multi-label, which was approached as seven binary problems of the 1417 Histopathology image collection. The positive instances are images belonging to a target category. Basal cell carcinoma is the only one related with cancer diagnosis.

Dataset	Classes	Distribution	Total
1. Histopathology [4]	7	carcinoma (518), collgen (1238), epidermis (147), hair folicle (118), eccrine glands (126), sebaceous glands (136), inflammatory infiltrate (99).	1417
2. Birds [49]	6	egret (100), mandarin (100), owl (100), puffin (100), toucan (100), wood duck (100),	600
3. Butterflies [50]	7	admiral (111), black-swallowtail (42), machaon (83), monarch-closed (74), monarch-open (84), peacock (134), zebra (91)	619
4. Scenes [13]	15	bedroom (216), suburb (241), industrial (311), kitchen (210), livingroom (289), coast (360), forest (328), highway (260), insidecity (308), mountain (374), opencountry (410), street (292), tallbuilding (356), office (215), store (315)	4485
5. 6-Caltech [51]	6	anchor (42), ant (42), camera (50), chair (62), crocodile (50), dollar-bill (52)	298

This strategy produces visual words that takes into account color and texture. We considered these features because in previous studies they have shown outstanding (e.g., DCT best descriptor found in [4,19,20] for Histopathology dataset) or at least competitive performance than other more complicated alternatives [52]. However, other types of feature-descriptors could be considered as well. It is worth noting that, in our *n*-gram experiments a setting of order *n* includes all *n*-grams of lower or equal order than *n*. The feature combination was done in that way because that is the way that *n*-grams have shown to improve text classification tasks [34,33,30] (we also performed experiments with separated representations but we obtained worse results). Furthermore, we have 400 unigrams[4] and different number of *n*-grams for each different value of *n* (from 1 to 3). The latter means that, in an experiment of 3-grams (1+2+3grams), we have combined 400 unigrams plus *x*-top-frequent 2-grams and the *x*-top-frequent 3-grams features for our BoVN. Even though there is a number of ways to select generated *n*-grams (e.g., information gain), we are interested in observing if the simple top-frequent features can improve the performance, this is also a common practice in several text mining tasks[34,30]. Moreover, it is worth mentioning that we have normalized each space of attributes in an individual way (we represent each space as a probability distribution). In the evaluation we used stratified 10 fold cross validation (10FCV) for each dataset. For the Histopathology dataset we report the average of the *F*-measure obtained on each binary problem. For the rest of image collections we report the micro-*F*-measure (FM), which weights the *F*-measure performance in each class according to the number of instances in the class. The FM reflects the performance considering the precision and recall.[5] Eq. (3) defines the *F*-measure in terms of the precision and recall:

$$FMeasure = 2 \times \frac{precision \times recall}{precision + recall} \tag{3}$$

For the proposed MKL strategies we report the average time required to perform the learning and classification steps. In our experiments we used a computer with a CPU Intel-Corei7 3.6 GHz and 64 GB of RAM. In Section 6 we will explain more details about each experiment such as the way we measured the performance and other specific conditions for each experiment.

[4] We chose 400 visual words as a fixed *k* value for all databases because of two reasons: for the Histopathology dataset is a good configuration [4], and for the other datasets, the number of visual words was between 100 and 400, nonetheless 400 features did not present a severe impact in the performance [49,13].

[5] Precision is the fraction of instances that are relevant, this is the number of true positives over the number of true positives plus the number of false positives. Recall is the fraction of relevant instances that are classified, this is the number of true positives over the number of true positives plus the number of false negatives [53].

5.1. Statistical significance of results

In this paper, we are interested in deciding if two approaches are statistically significantly different; these two approaches will be the proposed method (BoVN) and each of the proposed baselines (e.g., BoVW). For this reason we used the Wilcoxon signed-ranks (Wsr) [54] test for determining the statistical significance of differences in results. Wsr is the test recommended by J. Demšar for comparisons between two algorithms [55]. The Wsr is a non-parametric test, that makes no assumption that the differences between two random variables compared are distributed normally.

6. Experiments and results

In this section we explain the purpose and details of each experiment. We have chosen the most relevant experiments to show the different properties of the use of visual-*n*-grams for IC. The best result of each set of experiments is put in bold.

6.1. Bag-of-Visual-Words versus Bag-of-Visual n-grams

The goal of this experiment is to show how the proposed visual *n*-grams could improve the classification performance. For this, we present experimental results comparing the performances of a traditional Bag-of-Visual-Words (BoVW) and our proposed Bag-of-Visual *n*-grams (BoVN). In this first experiment, for the proposed BoVN, we extended the set of visual features by adding a set of 2-grams (multidirectional sequences of two visual words). Since the number of possible 2-grams are of hundreds of thousands we have fixed it to the top-frequent 2500.[6] For this experiment we represented images under the BoVN through the *early fusion* scheme (called early BoVN). Experiments reported in Table 3 show the performance for *early* BoVN, and the traditional BoVW using SVM with a linear kernel [56].

The experimental results presented in Table 3 suggest that, independently of using the DCT or SIFT descriptor, the use of visual 2-grams outperforms the average classification performance of 1-grams in every image collection. The averaged better *F*-measure obtained by the early BoVN, against the simple BoVW (which is the traditional BoVW using 1-grams), is in part due to the pairs of visual words representing structural visual patterns, which in some way reinforce some evidence in text mining [34,30]. It is worth noting that using DCT descriptor for the Histopathology dataset produces better classification performance than SIFT descriptor. This is because DCT descriptor considers important

[6] We analyze how the dimensionality influences the performance of a Bag-of-Visual 2-grams (testing incrementally from 1000 to 10 000 of features), getting that the 2500 top frequent bigrams are a good balance (sightly better than experiments using less and more features) between the dimensionality and the performance of our approach.

Table 3
F-measure results for visual words vs. visual *n*-grams. For image preprocessing in these collections, settings from Section 5 were used.

Results BoVW vs. BoVN *Averaged F-measure per collection*						
Descriptor	Model	Histopathology	Birds	Butterflies	Scenes	6-Caltech
DCT	Early BoVN	**64.54**	47.91	**62.19**	**63.40**	**54.10**
DCT	BoVW	58.54	**52.90**	61.10	61.01	53.48
SIFT	Early BoVN	**61.71**	**54.79**	52.31	**77.19**	**72.29**
SIFT	BoVW	53.41	53.12	**55.82**	74.10	70.51

Table 4
F-measure results for visual words vs. visual *n*-grams. For image preprocessing in these collections, settings from Section 5 were used.

BoVN *Averaged F-measure per collection*					
Config	Histopathology	Birds	Butterflies	Scenes	6-Caltech
1grams	58.54	53.12	61.10	74.10	70.51
1+2grams	**64.54**	**54.79**	**62.19**	**77.19**	**72.29**
1+2+3grams	62.69	53.31	62.09	76.12	71.11
1+2+3+4grams	61.34	51.11	61.05	74.32	69.31

properties of texture and color, which are relevant for histopathology images [4]. Furthermore, the histopathology images are captured in a more controlled environment, which makes possible to have images in the same scale and resolution. The DCT descriptor also obtained better results than SIFT for the Butterflies dataset. This is also due to the color and texture properties of the images. Moreover, most images in Butterflies dataset have the object in similar positions, which alleviates problems related with rotation. On the other hand, for Birds, Scenes and 6-Caltech datasets, using SIFT descriptor leads to a better classification performance than DCT. This is mainly because natural images have some properties (e.g., different scales, resolutions, and orientations) that SIFT descriptor can handle in a more appropriated way [52]. It is worth mentioning that under the same visual descriptor, computing the Wsr test over the outputs of the 10CFV in each dataset, we obtained more than 98% of statistical confidence in results comparing early BoVN and BoVW. In the following sections, we present more detailed experimental results. Given the evidence of the performance using DCT and SIFT descriptors in Table 3, for the *remainder of experiments* we used DCT descriptors for experiments in the Histopathology and Butterflies datasets, but SIFT descriptors for the rest of collections.

6.1.1. Longer sequences of visual words

The purpose of these experiments is to expose whether considering *n*-grams of higher order than 2 could improve the performance of the classifier. Table 4 presents the results of the experiments of the BoVN approach for visual *n*-grams using unigrams (which are the traditional visual words and one of our baselines) to tetragrams.[7] From results in Table 4 we can figure out that the best setting is 1+2grams. This can be due to the following reasons. The first one is related with the size of the sequences: it is well known that the higher *n* for *n*-grams, the higher number of instances are required to find that sequences of length *n* [33]. The second one is related with the high dimensionality: using longer sequences produces large vocabularies, which also produce sparse feature vectors (long sequences are more difficult to find [30]). According to the Wsr test, only the difference between using 1+2grams and 1+2+3grams is not statistical significantly. Nonetheless, using 1+2grams seems to be a better option given the compromise between the effectiveness and the required computational resources. For this reason, in our following experiments we used 1+2grams as visual features.

6.2. Strategies to exploit visual n-grams

Visual words and visual *n*-grams can be seen as two different sets of visual features. These visual features can be already used together (e.g., *early or late fusions*) to feed a wide range of different classification algorithms. Nevertheless, as explained in Section 3.3 it is possible to exploit these feature spaces using more appropriated fusion methods. The purpose of experiments in this section is to show that the MKL strategies can improve the classification performance taking advantage of the joint use of visual words and visual *n*-grams. For this we analyze the proposed visual *n*-grams under a MKL formulation, solving the kernel combination using a wide variety of MKL methods outlined in Table 1. We presents the general obtained results by Strategies 1 and 2 under the following specific kernel combinations[8]:

1. Linear $k_{LIN}(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$
2. Intersection $k_{INT}(\mathbf{x}_i, \mathbf{x}_j) = \sum_h^d \min(x_{i,h}, x_{j,h})$
3. The fusion of k_{LIN} and k_{INT} under MKL schemes.

In Tables 5 and 6 we report experimental results per collection. We also report the time required by each method to perform the 10CFV over all datasets. The time required to build the kernel matrix is what varies from one MKL strategy to another. Once the matrix kernel is learned/built, it is fed into a standard SVM. For these results, the proposed BoVN using *early* or *late* fusion strategies outperforms the traditional BoVW in each dataset. For the sake of comparison, in Table 5 we also evaluate other approach in the literature that also take advantage of contextual information; a Language Model Classifier (LMC). As explained in Section 2, language models have been used in previous works [12,29] for building classifiers. Thus, we have implemented a Language Model Classifier (LMC) as the one used in [12], which is based on the *CMU-Cambridge Statistical Language Modeling Toolkit v2* [57]. The language model classifier uses 1+2+3grams (configurations up to 10-grams were tested) remaining parameters of the software were left by default (e.g., smoothing good turing discount and backoff).[9] From Table 5 it can be seen that MKL Strategy 1 (specially RBMKL) is a better option than BoVN and LMC in most datasets, except for Caltech dataset where an *early* BoVN seems to obtain competitive results. Nevertheless, from the considered strategies and kernels to evaluate MKL using visual *n*-grams, the most competitive seems to be results in Table 6, which corresponds to Strategy 2: "several

[7] We selected the 2500 top frequent features for each *n*-gram space in the same way that in Section 6.1. Thus the experiment 1+2+3+4grams uses the information of the 400 visual words (1-grams) and 7500 sequences of visual words (*n*-grams).

[8] We study other basic kernel functions, as polynomial and gaussian, and their combination. Nevertheless, obtained results are much lower than the performance of linear and intersection kernel. This is due in part to that linear and intersection kernels are more appropriated for working with data represented using histograms.

[9] The language model classifier works as follows: (i) For each binary problem, it takes the training documents and builds two model languages (one for positive class and one for the negative) and (ii) for each test document, it measures the distance (using the probability chain rule) against the positive and negative models and it assigns the closest category.

Table 5
Strategy 1: Single kernel (linear | intersection) – several spaces. This table shows experiments using sequences of visual words (Uni-Bi-grams) early and late fusion. For these experiments we compute the *F*-measure for the positive class in each category on 10-fold cross validation using unigrams and bigrams in each of the problems. Simple BoVW is the only experiment using just the 400 visual words.

Strategy 1: Single kernel (linear | intersection) – several spaces
F-measure performance by collection

	Dataset					Time (h)
	Histopathology k_{LIN} | k_{INT}	Birds k_{LIN} | k_{INT}	Butterflies k_{LIN} | k_{INT}	Scenes k_{LIN} | k_{INT}	6-Caltech k_{LIN} | k_{INT}	
SimpleMKL	65.12 | 66.11	55.67 | 55.18	62.82 | 62.14	77.22 | 77.62	72.16 | 71.07	7.91
RBMKL	**66.53** | 66.43	55.01 | **55.98**	62.53 | **63.48**	76.12 | **77.79**	73.22 | **73.82**	3.71
NLMKL	62.41 | 61.11	54.33 | 55.43	62.31 | 62.24	75.11 | 75.53	71.87 | 70.52	12.81
LMKL	61.17 | 60.54	54.74 | 52.23	60.31 | 62.29	76.24 | 77.71	72.91 | 71.21	9.21
GMKL	65.20 | 67.42	54.54 | 55.33	61.28 | 58.22	73.32 | 74.03	72.34 | 70.93	6.56
GLMKL	65.69 | 65.21	54.73 | 54.37	61.44 | 60.51	77.01 | 76.44	73.12 | 72.17	6.58
CABMKL	60.55 | 61.31	53.53 | 53.02	62.28 | 62.89	76.98 | 75.38	72.12 | 72.44	5.97
ABMKL	60.91 | 59.88	53.50 | 54.73	62.55 | 62.15	73.91 | 72.21	69.13 | 67.12	4.21
Early BoVN	64.31 | 63.71	54.79 | 54.51	62.19 | 61.13	77.19 | 76.62	72.29 | 72.94	2.21
Late BoVN	60.31 | 61.31	53.05 | 54.34	61.51 | 61.28	76.04 | 75.25	71.34 | 72.01	2.36
Simple BoVW	58.59 | 57.21	53.12 | 54.10	61.10 | 62.66	74.10 | 73.11	70.51 | 69.38	1.16
LMC	53.00	54.76	61.14	62.12	68.41	5.24

Table 6
Strategy 2: several kernels (linear + intersection) – single space. This table shows experiments using sequences of visual words (Uni-Bi-grams) early and late fusion. For these experiments we compute the *F*-measure for the positive class in each category on 10-fold cross validation using unigrams and bigrams in each of the problems. Simple BoVW is the only experiment using just the 400 visual words.

Strategy 2: Several kernels (linear + intersection) – single space
F-measure performance by collection

	Dataset					Time (h)
model	Histopathology	Birds	Butterflies	Scenes	6-Caltech	
SimpleMKL	67.12	56.12	63.82	78.01	72.34	17.54
RBMKL	**68.53**	**56.41**	**64.00**	**78.22**	**74.12**	7.23
NLMKL	62.41	55.31	63.89	78.10	73.21	28.71
LMKL	61.17	53.71	63.83	77.21	73.84	23.34
GMKL	66.20	55.76	64.15	77.13	72.41	14.21
GLMKL	66.69	53.30	63.05	78.02	73.92	14.56
CABMKL	60.55	54.33	63.29	76.72	73.01	11.10
ABMKL	60.91	54.96	63.25	77.34	72.74	9.92
	k_{LIN} | k_{INT}	k_{LIN} | k_{INT}	k_{LIN} | k_{INT}	k_{LIN} | k_{INT}	k_{LIN} | k_{INT}	k_{LIN} | k_{INT}
Early BoVN	64.31 | 63.71	54.79 | 54.51	62.19 | 61.13	77.19 | 76.62	72.29 | 72.94	6.74
Late BoVN	60.31 | 61.31	53.05 | 54.34	61.51 | 61.28	76.04 | 75.25	71.34 | 72.01	6.98
Simple BoVW	58.59 | 57.21	53.12 | 54.10	61.10 | 62.66	74.10 | 73.11	70.51 | 69.38	2.24

(linear + intersection) kernels – single space)". In this strategy, linear and intersection kernels correspond to different notions of similarity of the whole space (visual words plus 2-grams). Then, instead of trying to find which kernel is the best, MKL method performs combination. Experimental results show improvements when using several MKL strategies, but the bests results in Table 6 were obtained by RBMKL. We individually validate RBMKL-Strategy-2 using the Wilcoxon Signed Rank test against: simple BoVW, early-BoVN, late-BoVN and RBMKL-Strategy-1. The output obtained by this test is above of 98% of statistical confidence. The best outcomes of the RBMKL method confirm what is reported in [17], which in similar domains obtains competitive performance or overcomes other approaches. The RBMKL method combines kernels performing a linear operation using the similarity matrices of each kernel, in our case the mean of the linear and intersection kernel matrices. RBMKL methods, through a simple-effective kernel operation can derive a kernel that better reflects similarities among instances represented under histograms of visual words and visual bigrams. This suggest that without using elaborated kernel learning techniques (note the time required by RBMKL), and under the proposed visual *n*-grams space, it is possible to compute useful similarity kernel matrices. We hypothesize that the main reason of this result is that instances are under a space of high dimensionality (400 visual words + 2500 2-grams), which allows us to obtain more appropriated similarity measures. The LMC does not provide better performance than BoVN. This can be due in part to the highly unbalanced data in some collections, which provides very few documents to build accurate language models for some positive classes. Moreover, since language models rely in probabilistic bases, the unbalanced data represents a common problem.

6.2.1. *Specific detailed results by class for MKL*

In this section we present some of the most relevant results by class obtained by MKL strategies. The purpose is to expose the performance of visual *n*-grams in some specific interesting classes.

Table 7
Strategy 2: Several (linear + intersection) kernels – single space. The table shows detailed experiments in the Histopathology dataset using sequences of visual words (Uni-Bi-grams) under RBMKL and the traditional BoVW. The class 1 is the only one related with cancer diagnosis.

Experiments for Histopathology dataset using Strategy 2
Detailed F-measure by class

Model	*Class*							Avg
	1	2	3	4	5	6	7	
RBMKL	**96.46**	**99.08**	**84.68**	**55.28**	**52.41**	**56.30**	**35.52**	**68.53**
BoVW	86.10	94.80	74.40	36.80	35.80	48.00	34.20	58.59

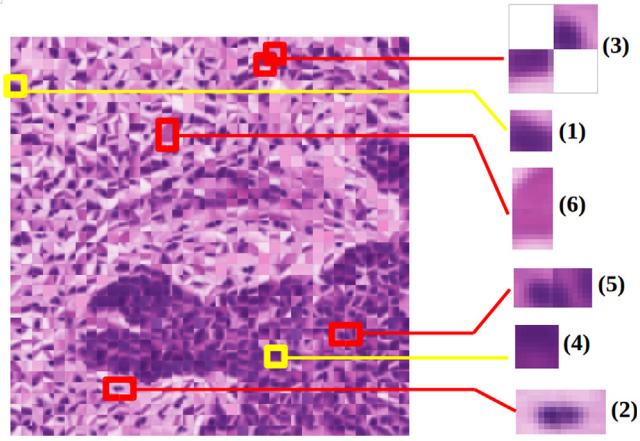

Fig. 6. Example of an image related with cancer diagnosis (class 1). The image is represented under the computed visual words codebook using DCT descriptor. According to information gain implemented in [58], we rank the 6 most discriminative visual features found in this image. We highlight in yellow and red, the most discriminative visual words and visual *n*-grams respectively. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 8
Strategy 2: Several (linear + intersection) kernels – single space. The table shows detailed experiments in the Birds dataset using sequences of visual words (Uni-Bi-grams) under early, late fusion and RBMKL (MKL intermediate fusion). The *F*-measure value of each class for BoVW corresponds to the best kernel configuration we found (linear or intersection).

Experiments for Birds dataset using Strategy 2
Detailed F-measure performance by class

Model	*Classes*						Avg
	Egret	Mandarin	Owl	Puffin	Toucan	Wood duck	
RBMKL	52.32	**47.71**	**68.03**	**55.53**	**64.32**	**50.55**	**56.41**
BoVW	**52.62**	41.54	66.75	54.37	60.21	49.15	54.10

Histopathology dataset: Results in Table 7 show that most methods using 1+2grams overcome 1-grams methods in most classes. This is more visible in classes 1 and 3–5. The class 1 is the most important, because it is the only one related with cancer diagnosis. Images in class 1 present structural tumor cells having large and darker nuclei, which are accurately characterized by visual bigrams (see Fig. 6). Visual words (1-grams) are competitive in classes 2, 6 and 7 (none of them related with cancer diagnosis). Such classes are in opposite ends, either by the lack of structured spatial visual elements (classes 2 and 6) that make bigrams to lose their advantage, or because the contextual information of visual words are much more global rather than local (class 6). We think those problems need more instances and explore other parameters (e.g., patch sizes, size of sequences, or alternative descriptors).

Birds and Butterflies datasets: Table 8 presents experimental results using the Birds dataset. Methods using 1+2grams outperform simple BoVW (1-grams) in some specific classes. This is more visible in classes Mandarin, Puffin, Toucan, and Wood duck. In a similar way that the Histopathology dataset, we think those results are in part due to the complexity of each class image. Fig. 7 shows one instance of the Egret class (left) and one of the mandarin class (right). Results suggest that, simple BoVW, in some way can solve the Egret class because the target object contains low variety of visual words and there are less structural local visual patterns (captured by the *n*-grams). On the other hand, the image belonging to the mandarin class, expose more visual spatial patterns that could be extracted (see example in Fig. 8).[10] A similar situation is presented for results in Table 9 for the Butterflies image collection (see the example in Fig. 9).

Finally, experimental results using visual *n*-grams for the Scenes datasets also showed similar properties for specific classes. For example, in this collection there are classes where results of experiments using visual *n*-grams are closer to the pure use of visual words (1-grams), having low gain/lost performance. Some classes with more difference in performance are kitchen, living room, bedroom and store. These kinds of indoor classes appear to be the more complicated given the high variety of objects that could be found (other interesting classes are street and suburb). In those classes, visual *n*-grams provided an improvement in the performance. On the other hand, simple visual words get better results in natural scenes like mountain, forest, open country and coast have more plain unstructured visual elements like sky, grass, water, etc. We think such images are better classified by a simple BoVW because there are structural visual elements that need to be captured in a more global way (visual *n*-grams capture local visual patterns).

6.3. Bag-of-Visual n-grams and the Spatial Pyramid Representation (SPR)

In this section we present an experimental evaluation to deepen the analysis of the proposed visual *n*-grams and the BoVW. In spite of the simplicity and effectiveness of the BoVW approach, there have been several efforts to incorporate spatial information to it. In addition to the language models used in Section 6.2, there have been other approaches to capture spatial information at different levels. The Spatial Pyramid Representation (SPR) [13] is one of the most notable works to improve the performance of BoVW. In the following subsections we evaluate and compare the performance of the proposed BoVN and the SPR. We explain the key differences between the two approaches and discuss some properties of each dataset that makes possible the outstanding performance of each approach. Furthermore, we found that both approaches can be easily integrated to obtain an improvement in the classification performance.

6.3.1. BoVN vs. SPR

The purpose of this first experiment is to compare the classification performance of the proposed strategy and the SPR. For this purpose we evaluate our best strategy using the visual 1+2grams and RBMKL. We also evaluate the performance of the SPR representation on each dataset.[11] Some interesting findings can be

[10] Analogous characteristics present other classes like Owl (mostly a white bird) and toucan (a bird with more contrast and structural characteristics).

[11] We use the following experimental settings: 400 visual words and 8x8 size patches, besides our best descriptor for each collection; DCT descriptor for

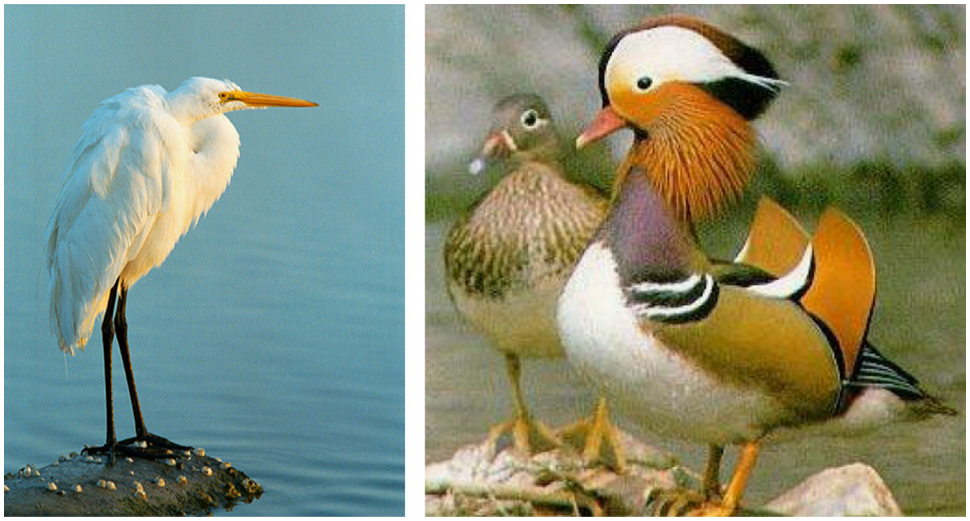

Fig. 7. Left *Egret* class and Right *Mandarin* of the Birds dataset. Sample instances to expose the image characteristics of those two classes and their performance when using visual *n*-grams to capture the context.

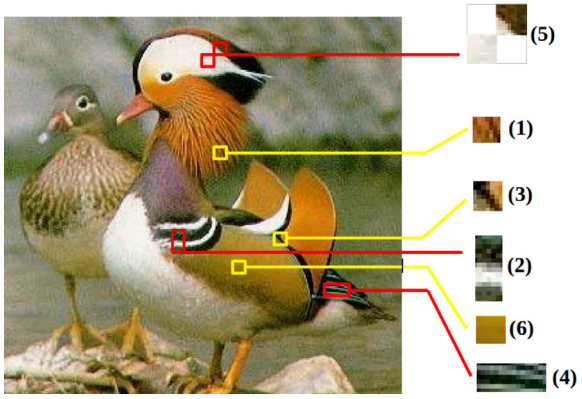

Fig. 8. Example of a Mandarin duck image. According to information gain implemented in [58], we rank the 6 visual regions that produced visual features with most discriminative information. We highlight in yellow and red, the regions that using SIFT descriptors produced discriminative visual words and visual 2-grams respectively. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

highlighted from results in Table 10. For example, it is interesting to see that, according to the Wsr test, the proposed RBMKL approach significantly outperforms the SPR in the Birds and the Butterflies datasets. Regarding the 6-Caltech dataset, RBMKL also outperforms SPR, however, the difference is not statistically significant. Finally for the 15-scenes dataset, SPR significantly outperforms our approach. We think such results on each dataset are due to specific characteristics of each domain. More specifically, the proposed visual *n*-grams extract very *local* visual patterns. For example, visual bigrams are useful to capture some characteristic lines of the mandarin duck (see Fig. 8), but may fail to capture more global visual patterns (e.g., the mandarin ducks are usually surrounded by visual elements similar to water or grass). On the other hand, SPR captures more *global* and *absolute* visual patterns, which can be matched by the intersection kernel according to each region in corresponding levels of the pyramid. We believe that this is one of the reasons of the high performance of SPR in natural scenes dataset, where images belonging to the same class share more visual words in each pyramid level (e.g., in buildings images the top part usually is the sky and clouds). Nonetheless, SPR may fail to account for very local and relative patterns, especially when an image presents the target object in a wide range of positions and rotation variants (the case of Birds and Butterflies datasets). The latter scenario hinder to match several levels of the pyramid (except for the coarser level), which produce a more noisy representation.

6.3.2. Extending SPR using visual n-grams and MKL

The purpose of this second experiment is to evaluate the classification performance of visual *n*-grams when they are integrated into the SPR. For this, we evaluate SPR using the experimental settings of Section 6.3.1. In Table 11, there are three SPR strategies. The first one (SPR) corresponds to the standard implementation as described in [13]. The second one (SPR+VN) is a simple extension where local histograms of visual 1+2grams are computed from each cell in the pyramidal representation. Thus, the final pyramidal representation of an image is the arrangement of histograms of 1+2grams corresponding to each cell. Finally, SPR+VN+RBMKL is an approach, that under the Strategy 2, uses the representation vectors of SPR+VN to learn a new kernel.[12] From results in Table 11, we can observe that integrating visual 2grams into the SPR results in an improvement of the classification performance. We can also note that by using RBMKL to learn the kernel, the impact in the performance is positive. It is worth mentioning that SPR+VN and SPR+VN+RBMKL, according to the Wsr, are significantly better than SPR. We think that the improvement in the results id due to the complementary *local* and *global* information carried by each of the combined methods.

7. Conclusions

The interest of this research lies at the intersection of the fields of computer vision and natural language processing. The underlying motivation is to improve the state-of-the-art BoVW through fusion strategies to integrate the visual *n*-grams (multi-directional sequences of visual words) as attributes. This takes the analogy

(*footnote continued*)
histology and Butterflies datasets, while SIFT descriptor for the rest. An intersection kernel as defined in [13] is used into an SVM. By using these fixed experimental settings, we experimental determine to 3 the number of the pyramid levels in SPR by exploring values between 2 and 5.

[12] A linear and an intersection kernel is built using image representation of SPR+VN, then RBMKL is used to learn a new kernel function.

Table 9
Strategy 2: Several (linear + intersection) kernels - single space. The table shows detailed experiments in the Butterflies dataset using sequences of visual words (Uni-Bi-grams) under early, late fusion and RBMKL (MKL intermediate fusion). The *F*-measure value of each class for BoVW corresponds to the best kernel configuration we found (linear or intersection).

Experiments for Butterflies dataset using Strategy 2 *Detailed F-measure performance by class*								
Model	*Classes*							Avg
	Admiral	Swallow tail	Machaon	Monarch 1	Monarch 2	Peacock	Zebra	
RBMKL	57.31	**44.73**	**70.61**	**57.61**	**67.14**	**72.32**	**65.11**	**62.11**
BoVW	**58.82**	42.05	68.02	54.95	65.15	70.72	64.10	60.45

Fig. 9. Left *Black Swallowtail* class and Right *Peacock* of the Butterflies dataset. Sample instances to expose the image characteristics of those two classes and their performance when using visual *n*-grams to capture the context.

Table 10
F-measure results for RBMKL vs. SPR. For image preprocessing in these collections, settings from Section 5 were used. For each dataset, we use the asterisk to indicate statistical significance of differences in results.

Results RBMKL vs. SPR *Averaged F-measure per collection*					
Model	Histopathology	Birds	Butterflies	Scenes	6-Caltech
RBMKL	**68.53***	**56.41***	**64.00***	78.2	**74.12**
SPR	67.32	53.38	61.97	**80.10***	72.13

Table 11
F-measure results for SPR extended with visual *n*-grams and MKL. For image preprocessing in these collections, settings from Section 5 were used.

Results of integrating visual *n*-grams and MKL into SPR *Averaged F-measure per collection*					
Model	Histopathology	Birds	Butterflies	Scenes	6-Caltech
SPR	67.32	53.38	61.97	80.10	72.13
SPR+VN	68.90	57.71	63.15	80.78	74.83
SPR+VN+RBMKL	**70.02**	**58.19**	**65.31**	**81.29**	**76.17**

visual–textual words into a new higher level combining contextual (visual *n*-grams) and non-contextual (visual words) information through alternative fusion strategies. Motivated by the analogy visual–textual, we consider the fusion of the contextual and non-contextual information in NLP tasks. Thus, we evaluate visual *n*-grams to consider visual spatial information in NLP. Regarding the typical fusion strategies, simple *early fusion* strategy showed better/similar performance than *late fusion* approach. This is due in part to that in text classification most of textual feature spaces are derived from one rigorous modality: the text. This condition degrades the diversity among the search space, which is one of the most important aspects for building ensembles. The results show evidence of the usefulness of integrate visual *n*-grams under the proposed MKL strategies, showing that, every experiment using visual bigrams outperforms unigrams and other methodologies. Our results suggest evidence of the usefulness of BoVN under MKL strategies (in particular for RBMKL) in different image collections. We believe this is because the method is finding better notions of similarity for each space, which could be difficult to obtain with other typical approaches like early and late fusion. To the best of our knowledge, the usefulness of visual *n*-grams under different fusion strategies has never been studied for different image domains. In this paper we study the *early*, *late* and proposed *intermediate* fusion approaches to provide a solid framework using visual words and the proposed visual *n*-grams. Future research paths include bringing ideas to capture contextual information in a more global way, and extracting higher level information among visual words considering the co-occurrence of the elements to model semantic information to improve the discriminative power in more complex images like the ones in the Scenes dataset.

Acknowledgments

This research was supported by CONACyT under Research Grant CB-241306: “Clasificación y Recuperación de Imágenes Mediante Técnicas de Minerí a de Textos”. López-Monroy thanks

for doctoral scholarship CONACyT-Mexico 243957. Finally, Cruz-Roa also thanks for doctoral grant supports Colciencias 528/2011 and "An Automatic Knowledge Discovery Strategy in Biomedical Images" DIB-UNAL/2012.

References

[1] G. Csurka, C.R. Dance, L. Fan, J. Willamowski, C. Bray, Visual categorization with bags of keypoints, In: International Workshop on Statistical Learning in Computer Vision, ECCV, vol. 1, 2004, p. 22.
[2] J. Sivic, A. Zisserman, Video Google: a text retrieval approach to object matching in videos, In: Proceedings of the International Conference on Computer Vision, ICCV, 2003.
[3] T. Tommasi, F. Orabona, B. Caputo, Image annotation task: an svm-based cue integration approach, In: Working Notes of the 2007 CLEF Workshop, 2007.
[4] A. Cruz-Roa, J.C. Caicedo, F.A. González, Visual pattern mining in histology image collections using bag of features, Artif. Intell. Med. 52 (2011) 91–106.
[5] L. Fei-Fei, P. Perona, A Bayesian hierarchical model for learning natural scene categories, In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, CVPR 2005, vol. 2, IEEE, San Diego, CA, 2005, pp. 524–531.
[6] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: a comprehensive study, Int. J. Comput. Vis. 73 (2007) 213–238.
[7] P. Tirilly, V. Claveau, P. Gros, A Review of Weighting Schemes for Bag of Visual Words Image Retrieval, Technical Report, TEXMEX – INRIA – IRISA, 2009.
[8] H. Wang, M.M. Ullah, A. Klaser, I. Laptev, Evaluation of local spatio-temporal features for action recognition, In: Proceedings of the British Machine Vision Conference, 2009, pp. 1–11.
[9] P. Turney, From frequency to meaning: vector space models of semantics, J. Artif. Intell. Res. 37 (2010) 141–188.
[10] C. Galleguillos, S. Belongie, Context based object categorization: a critical survey, Comput. Vis. Image Underst. 114 (2010) 712–722.
[11] J. Krapac, J. Verbeek, F. Jurie, Modeling spatial layout with Fisher vectors for image categorization, In: 2011 IEEE International Conference on Computer Vision, ICCV, IEEE, Barcelona, Spain, 2011, pp. 1487–1494.
[12] P. Tirilly, V. Claveau, P. Gros, Language modeling for bag-of-visual words image categorization, In: ACM Proceedings of the 2008 International Conference on Content-Based Image and Video Retrieval, 2008, pp. 249–258.
[13] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, vol. 2, IEEE, New York City, NY, 2006, pp. 2169–2178.
[14] Y. Cao, C. Wang, Z. Li, L. Zhang, L. Zhang, Spatial-bag-of-features, In: 2010 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, IEEE, San Francisco, CA, 2010, pp. 3352–3359.
[15] A. Pastor López-Monroy, M. Montes-y-Gómez, H.J. Escalante, A. Cruz-Roa, F.A. González, Bag-of-visual-ngrams for histopathology image classification. Proc. SPIE 8922, IX International Seminar on Medical Information Processing and Analysis, SPIE, Mexico City, México, 2013, 89220 p, http://dx.doi.org/10.1117/12.2034113.
[16] L. Kuncheva, Combining Pattern Classifiers, Wiley Press, New York, 2005, pp. 241–259.
[17] M. Gönen, E. Alpaydın, Multiple kernel learning algorithms, J. Mach. Learn. Res. 12 (2011) 2211–2268.
[18] M. Alioscha-Pérez, H. Sahli, I. González, A. Taboada-Crispi, Sparse and non-sparse multiple kernel learning for recognition, Comput. Sist. 16 (2) (2012) 167–174.
[19] A. Cruz-Roa, G. Díaz, E. Romero, F.A. González, Automatic annotation of histopathological images using a latent topic model based on non-negative matrix factorization, J. Pathol. Inf. 2 (4) (2011).
[20] G. Díaz, E. Romero, Micro-structural tissue analysis for automatic histopathological image annotation, Microsc. Res. Tech. 75 (2012) 343–358.
[21] H.J. Escalante, L.E. Sucar, M. Montes-y Gómez, Semantic cohesion for image annotation and retrieval, Comput. Sist. 16 (1) (2012) 121–126.
[22] Y.-T. Zheng, M. Zhao, S.-Y. Neo, T.-S. Chua, Q. Tian, Visual synset: towards a higher-level visual representation, In: IEEE Conference on Computer Vision and Pattern Recognition, 2008, CVPR 2008, IEEE, Anchorage, AL, 2008, pp. 1–8.
[23] M. Jamieson, A. Fazly, S. Dickinson, S. Stevenson, S. Wachsmuth, Learning structured appearance models from captioned images of cluttered scenes, In: IEEE 11th International Conference in Computer Vision, 2007, ICCV 2007, 2007, pp. 1–8.
[24] Q.F. Zheng, W. Wang, W. Gao, Effective and efficient object-based image retrieval using visual phrases, In: ACM Proceedings of the 14th Annual ACM International Conference on Multimedia, 2006, pp. 77–80.
[25] J. Yuan, Y. Wu, M. Yang, Discovery of collocation patterns: from visual words to visual phrases, In: IEEE in Computer Vision and Pattern Recognition, 2007, CVPR 2007, 2007, pp. 1–8.
[26] J. Yuan, M. Yang, Y. Wu, Mining discriminative co-occurrence patterns for visual recognition, In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, IEEE, Providence, RI, 2011, pp. 2777–2784.
[27] J. Sivic, A. Zisserman, Video data mining using configurations of viewpoint invariant regions, In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004, CVPR 2004, vol. 1, IEEE, Washington, DC, 2004, pp. 488–495.
[28] T. Quack, V. Ferrari, B. Leibe, L. Van Gool, Efficient mining of frequent and distinctive feature configurations, In: IEEE 11th International Conference on Computer Vision, 2007, ICCV 2007, IEEE, Rio de Janeiro, Brazil, 2007, pp. 1–8.
[29] L. Wu, M. Li, Z. Li, W.Y. Ma, N. Yu, Visual language modeling for image classification, In: ACM Proceedings of the International Workshop on Workshop on Multimedia Information Retrieval, 2007, pp. 115–124.
[30] S. Wang, C.D. Manning, Baselines and bigrams: simple, good sentiment and topic classification, In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, 2009, pp. 90–94.
[31] A. McCallum, K. Nigam, et al., A comparison of event models for naive Bayes text classification, In: AAAI-98 Workshop on Learning for Text Categorization, vol. 752, Citeseer, 1998, pp. 41–48.
[32] R. Khan, C. Barat, D. Muselet, C. Ducottet, Spatial histograms of soft pairwise similar patches to improve the bag-of-visual-words model, Comput. Vis. Image Underst. 132 (0) (2015) 102–112 http://dx.doi.org/10.1016/j.cviu.2014.09.005, URL ⟨http://www.sciencedirect.com/science/article/pii/S1077314214001878⟩.
[33] C.M. Tan, Y.F. Wang, C.D. Lee, The use of bigrams to enhance text categorization, Inf. Process. Manag. 38 (2002) 529–546.
[34] R. Bekkerman, J. Allan, Using Bigrams in Text Categorization, Technical Report, Department of Computer Science, University of Massachusetts, Amherst, 2004.
[35] L. Rokach, Ensemble-based classifiers, Artif. Intell. Rev. 33 (2009) 1–39.
[36] O. Boiman, E. Shechtman, M. Irani, In defense of nearest-neighbor based image classification, In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2008, CVPR 2008, 2008, pp. 1–8.
[37] L. Breiman, Bagging predictors, Mach. Learn. 24 (1996) 123–140.
[38] A. Ben-Hur, W.S. Noble, Kernel methods for predicting protein–protein interactions, Bioinformatics 21 (Suppl 1) (2005) i38–i46.
[39] Z. Xu, R. Jin, H. Yang, I. King, M.R. Lyu, Simple and efficient multiple kernel learning by group lasso, In: Proceedings of the 27th International Conference on Machine Learning, ICML-10, 2010, pp. 1175–1182.
[40] M. Kloft, U. Brefeld, S. Sonnenburg, A. Zien, Non-sparse regularization and efficient training with multiple kernels, Arxiv preprint arXiv 1003 (0079), 2010, p. 186.
[41] A. Rakotomamonjy, F. Bach, S. Canu, Y. Grandvalet, Simplemkl, J. Mach. Learn. Res. 9 (2008) 2491–2521.
[42] C. Cortes, M. Mohri, A. Rostamizadeh, Two-stage learning kernel algorithms, In: Proceedings of the 27th International Conference on Machine Learning, ICML-10, 2010, pp. 239–246.
[43] G. Brown, J. Wyatt, R. Harris, X. Yao, Diversity creation methods: a survey and categorisation, Inf. Fusion 6 (2005) 5–20.
[44] R.O. Chávez, M. Montes, L.E. Sucar, Using a Markov random field for image re-ranking based on visual and textual features, Comput. Sist. 14 (4) (2011) 393–404.
[45] C. Cortes, M. Mohri, A. Rostamizadeh, Learning non-linear combinations of kernels, In: Advances in Neural Information Processing Systems, 2009, pp. 396–404.
[46] M. Gönen, E. Alpaydin, Localized multiple kernel learning, In: Proceedings of the 25th International Conference on Machine Learning, ACM, Helsinki, Finland, 2008, pp. 352–359.
[47] M. Varma, B.R. Babu, More generality in efficient multiple kernel learning, In: Proceedings of the 26th Annual International Conference on Machine Learning, ACM, Montreal, Quebec, Canada, 2009, pp. 1065–1072.
[48] S. Qiu, T. Lane, A framework for multiple kernel support vector regression and its applications to sirna efficacy prediction, IEEE/ACM Trans. Comput. Biol. Bioinform. 6 (2) (2009) 190–199.
[49] S. Lazebnik, C. Schmid, J. Ponce, A maximum entropy framework for part-based texture and object recognition, In: 10th IEEE International Conference on Computer Vision, 2005, ICCV 2005, vol. 1, IEEE, Beijing, China, 2005, pp. 832–838.
[50] S. Lazebnik, C. Schmid, J. Ponce, et al., Semi-local affine parts for object recognition, In: British Machine Vision Conference, BMVC'04, 2004, pp. 779–788.
[51] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories, Comput. Vis. Image Underst. 106 (1) (2007) 59–70.
[52] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[53] D.M. Powers, Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.
[54] F. Wilcoxon, Individual comparisons by ranking methods, Biometr. Bull. 1 (6) (1945) 80–83 , URL ⟨http://www.jstor.org/stable/3001968⟩.
[55] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.
[56] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, ACM Trans. Intell. Syst. Technol. 2 (2011) 55 27:1–27, Software available at ⟨http://www.csie.ntu.edu.tw/cjlin/libsvm⟩.
[57] P. Clarkson, R. Rosenfeld, Statistical language modeling using the CMU-Cambridge toolkit, In: Proceedings of EUROSPEECH, International Speech Communication Association Rhodes, Greece, 1997, vol. 97, pp. 2707–2710.

[58] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. Witten, The Weka Data Mining Software: An Update, SIGKDD Explorations 11.

A. Pastor López-Monroy is a Ph.D. student in the Computer Science Department at the National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico, where he is part of the Language Technologies Laboratory. His research focuses on, but not limited to, the intersection of the fields of computer vision and natural language processing. In his research he studies image analysis and text mining algorithms to design novel methods to improve the use of specific visual features in computer vision tasks. He obtained his B. Eng. degree from the Technological Institute of Celaya, and M.Sc. degree from INAOE in 2009 and 2012 respectively.

Manuel Montes-y-Gómez is part of the Laboratory of Language Technologies (LabTL) at the National Institute of Astrophysics, Optics and Electronics (INAOE) México. He received his Ph.D. in Computer Science from the Center of Computing Research at the National Polytechnic Institute (IPN), México, 2002. He also received the M.Sc. in Computer Science from the National Institute of Astrophysics, Optics and Electronics (INAOE), 1998. Among his research topics of interest are text mining, information retrieval and computer vision.

Hugo Jair Escalante is a part of the Laboratory of Language Technologies (LabTL) at the National Institute of Astrophysics, Optics and Electronics (INAOE) México. He received his Ph.D. and M.Sc. in Computer Science from National Institute of Astrophysics, Optics and Electronics (INAOE) in 2010 and 2006 respectively. Among his research topics of interest are machine learning, text mining, information retrieval and computer vision.

Angel Cruz-Roa belongs to Machine Learning, Perception and Discovery Lab (MindLab) from National University of Colombia, before he was with Bioingenium Research Group from National University of Colombia since 2008 and he finished the Master in Biomedical Engineering and started the Doctorate in Systems and Computer Science in 2010 at the National University of Colombia. Among his research topics of interest are medical image representation and understanding, image processing, machine learning, computer vision, computer graphics and applications.

Fabio A. González is the head of the Machine Learning, Perception and Discovery Lab (MindLab) from National University of Colombia. He received his Ph.D. and M.Sc. in Computer Science from the University of Memphis, 2003. He also received the M.Sc. in Maths from the Universidad Nacional de Colombia, 1998. Among his research topics of interest are machine learning, information retrieval and computer vision.