Artistic Image Analysis using Graph-based Learning Approaches

Gustavo Carneiro

***Abstract*—We introduce a new methodology for the problem of artistic image analysis, which among other tasks, involves the automatic identification of visual classes present in an art work. In this paper we advocate the idea that artistic image analysis must explore a graph that captures the network of artistic influences by computing the similarities in terms of appearance and manual annotation. One of the novelties of our methodology is the fact that the proposed formulation is a principled way of combining these two similarities in a single graph. Using this graph, we show that an efficient random walk algorithm based on an inverted label propagation formulation produces more accurate annotation and retrieval results compared to the following baseline algorithms: bag of visual words, label propagation, matrix completion, and structural learning. We also show that the proposed approach leads to a more efficient inference and training procedures. This experiment is run on a database containing 988 artistic images (with 49 visual classification problems divided into a multi-class problem with 27 classes and 48 binary problems), where we show the inference and training running times, and quantitative comparisons with respect to several retrieval and annotation performance measures.**

***Index Terms*—Content-based image retrieval, Art image analysis, Graph-based learning methods.**

I. Introduction

Artistic image analysis is an interdisciplinary field of work involving computer vision researchers and art historians. We consider an artistic image to be an artistic expression represented on a flat surface (e.g., canvas or sheet of paper) in the form of a painting, printing or drawing. The analysis of artistic image deserves special attention by computer vision scientists for several reasons. First, current dominant visual classification methods in the field based on inductive models and the bag of features (BoF) representation are not adequate for the classification of visual classes in artistic images because of the lack of consistent texture, color and geometry features to represent robustly those visual classes. For instance, notice in Fig. 1 the different types of color and textures that can be found for the visual class "sea". In fact, the representation of visual classes in artistic images is so inconsistent that color and texture have hitherto been used only for characterizing artists [1] and style [2] rather than visual classes. The second reason is that, compared to digital photographic images, artistic images have several orders of magnitude less training images. This is important because current visual classification methodologies in the field need large training sets to work robustly. The third reason is the potential to strengthen the links between the fields of computer vision and art history, which can open opportunities in terms of research, education and applications. Finally, the fourth reason is that the analysis of man-made artistic images may enable the development of new approaches that can help the field of computer vision solve more general image analysis problems.

a) b) c)

Fig. 1. Different paintings showing the visual class "sea" with remarkably different patterns of color and texture. In (a), we show Pieter Brueghel il Giovane's *Christ on the Storm on the Sea of Galilee*; in (b) we have Claude Monet's *Shadows on the Sea"*; and (c) displays John Marin's *Sea Piece*.

Even though there are several types of artistic images, in this paper we focus on art works produced via printmaking techniques. Printmaking is a method of replicating paintings based on intaglio printing (e.g., etching), relief printing (e.g., engraving) or planographic printing (e.g., lithography) [3]. Even though printmaking is considered to be a creative work of art, the print produced still has clear connections to the original painting, as evidenced in Fig. 2, which shows a painting in (a) and its print (b), with noticeable similarities and modifications. Focusing on the analysis of artistic prints is important given that they have influenced and served as one of the major sources of inspiration for generations of artists. This substantial influence happened because of the fast and cheap production of paper and advancements in graphical arts that have happened in the last five centuries. Specific examples can be found in the influence of Japanese art prints on impressionist artists in the XIX^{th} century [4], the influence of the Flemish and Italian masters in the paintings of the Madrilean Baroch [5], and the influence of prints on artistic tile painters in Portugal [6], [7] (see Fig. 3). Currently, the process of discovering the influences between different works of art is a central task in the field of art history, which can only be performed by an experienced art historian.

Compared to photographic images, artistic images have several characteristics that can be explored. For instance, they usually follow composition rules that can be exploited during the analysis process (e.g., rule of thirds, rule of odds, etc.).

This work was funded by the FCT (ISR/IST plurianual funding) through the PIDDAC Program funds and Project PRINTART (PTDC/EEA-CRO/098822/2008). G. Carneiro was also supported by the European Commission; Marie Curie IIF, contract number PIIF-GA-2009-236173 (IMASEG3D). G. Carneiro is with the Australian Centre for Visual Technologies, the University of Adelaide, Australia.

a) Painting b) Print

Fig. 2. Examples of different artworks of the theme "The Assumption of Mary into Heaven". In (a) it is displayed the painting by Rubens (1626), and (b) shows a print by Willem Panneels (1630).

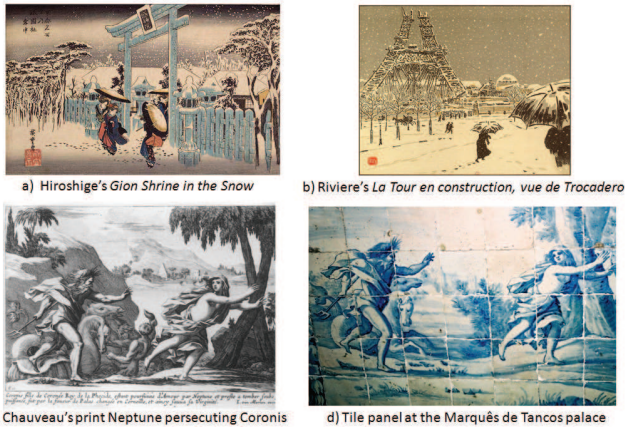


Fig. 3. Influence of Japanese art prints (a) on impressionist paintings (b), and of monochromatic art prints (c) on tile panel paintings (d).

Also, during the analysis process, several constraints can be enforced if they are available, such as author, theme, style, period and origin. Finally, one of the most important properties that can be explored in an artistic image analysis process is the network of artistic influence, where each new art work can be visually similar to other previous works and this visual similarity usually implies label similarity.

In this paper, we introduce a new method for the annotation and retrieval of artistic prints, consisting of a simple and effective transductive inference procedure. In general, when compared to inductive models, transduction explores explicitly visual and label similarities among training images and visual similarity between training and test images, which has the following advantages: 1) more adaptable to new training sets, 2) better results with training sets of small sizes [8], and 3) natural choice to capture the network of artistic influences described above. The transductive inference proposed in this paper is an extension of the inverted label propagation methodology that we propose in [9], which is a graph-based approach, where the nodes are represented by the images and the edge weights are computed using a measure of appearance and label similarities between images. Label propagation methods usually produce annotations given a test image, while inverted label propagation produces a ranked list of training images, and the final annotation is then estimated based on the combination of the annotations of the ranked training images. This inverted label propagation methodology [9] has been shown to produce empirically better annotation and retrieval results than several baseline methods. The technical novelty of this paper compared to [9] lies in the manner that we combine the top ranked images to produce an annotation for the test image, which provides considerably more accurate annotation and retrieval results. Additionally, we compare the annotation and retrieval results produced by our system to more baseline methods (when compared to [9]), such as: bag of features (BoF) with support vector machine (SVM) classifier [10], label propagation [11], label propagation with label correlation [12], [13], matrix completion [14], and structural learning [15]. Another novelty of this paper is with respect to the database used, which contains 988 images with 49 visual classes (with one multi-class problem with 27 classes and 48 binary problems) instead of the 307 images with 22 classes (one multi-class with 7 classes and 21 binary problems) of [9]. With this dataset, we are able to compare the performance of all approaches with respect to the number of training images and the dimensionality of the feature representation. Finally, we also provide a running time complexity analysis of the training and inference procedures of all methods discussed in this paper.

II. Literature Review

In this section, we provide a brief review of the works in art and photographic image retrieval and annotation. We also review graph-based learning methods that are relevant to our proposal.

The area of art image annotation and retrieval has attracted the attention of researchers in the fields of computer vision and machine learning [16], [17], [18]. The majority of these works focuses on the artistic identification problem, where the goal is to classify original and fake paintings of a given artist [19], [20], [21] or to produce stylistic analysis of paintings [2], [22], [23]. Most of the methods above can be regarded as adaptations from the content-based image retrieval systems [24], where the emphasis is placed on the characterization of brush strokes using texture or color. Multi-class classification has been explored in other artistic image analysis. For example, the ancient Chinese painting classification studied by Li and Wang [1] deals with the multi-class classification of painting styles, and the automatic brushwork annotation by Yelizaveta et al. [25] handles the multi-class classification of brush strokes. Nevertheless, our problem involves not only a multi-class, but also a multi-label classification [26].

Currently in photographic image annotation and retrieval, the most successful methods are based on the bag of visual words framework using a multiple kernel learning (MKL) classifier [27], which is an extension of the SVM classifier that allows the combination of several kernels. This methodology is effective when used in a retrieval setting where the number classes is relatively small (with a large number of training images per class), the set of visual classes is fixed, and the color and texture features are consistent across samples of each visual class. Unfortunately, these constraints do not apply for artistic images because the number of visual classes can be quite large (with each visual class containing relatively small

number of training images), the introduction of new images to the database may happen often, and the visual classes are poorly characterized by their texture and color patterns. In photographic image analysis, there is a trend to get around the problem of the high number of visual classes with the use of compressed sensing [28], which finds a sub-space of smaller dimensionality for classification. However, the dynamic nature of this learning problem, where new classes are regularly introduced into the training database, is still an issue in this area of research. Finally, except for a few studies in art image analysis [1], we are not aware of methodologies that can deal with problems presenting visual classes with inconsistent color and texture representations.

Graph-based learning (or network link analysis) has been thoroughly studied by the information retrieval community to rank web pages [29], [30], [31]. Essentially, a graph is built where the vertexes represent the web pages and the edge weights denote the existence of hyper-links. Analysis algorithms based on random walks in this graph have been designed to rank the nodes (i.e., web pages) according to their importance in this network. These graph-based techniques have also received considerable attention from the machine learning community for the problems of semi-supervised learning [32], unsupervised image segmentation [33] and multi-class classification [34]. In the area of image retrieval [35], [36], [37], the approaches based on random walk procedures scale gracefully, can handle training sets of small sizes, allow the combination of visual and non-visual cues, and tackle dynamic problems where new images and annotations are continuously introduced into the database. However, all these approaches generally use different graphs for different types of information. In contrast, our formulation introduces a new approach that joins all these similarities into a single graph.

III. Database and Problem Formulation

The artistic database used in this paper contains 988 images with global annotations (see Fig. 4). These images were collected from the Artstor digital image library [38] and annotated by art historians. The global annotation contains one multi-class problem (theme with 27 classes) and 48 binary problems. In the experiments, we divide this database into a training set $\mathcal{D}$, and a test set $\mathcal{T}$.

The training set of annotated images is defined as $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{D}|}$, with $\mathbf{x}_i$ being the representation of image I_i and $\mathbf{y}_i$ denoting the global annotation of that image representing the M multi-class and binary problems, so $\mathbf{y}_i = [\mathbf{y}_i(1), ..., \mathbf{y}_i(M)] \in \{0,1\}^Y$, where each problem is denoted by $\mathbf{y}_i(k) \in \{0,1\}^{|\mathbf{y}_i(k)|}$ with $|\mathbf{y}_i(k)|$ representing the dimensionality of $\mathbf{y}_i(k)$ (i.e., $|\mathbf{y}_i(k)| = 1$ for binary problems, $|\mathbf{y}_i(k)| > 1$ with $\|\mathbf{y}_i\|_1 = 1$ for multi-class problems), $Y = 75$, and $M = 49$ with one multi-class problem (with 27 classes) and 48 binary problems. This means that binary problems involve an annotation that indicates the presence or absence of a visual class, while multi-class annotation regards problems that one and only one of the possible classes is present. The test set is represented by $\mathcal{T} = \{(\widetilde{\mathbf{x}}_i, \widetilde{\mathbf{y}}_i)\}_{i=1}^{|\mathcal{T}|}$, with $\mathcal{D} \bigcap \mathcal{T} = \emptyset$. The union of $\mathcal{D}$ and $\mathcal{T}$ produces the full dataset with 988 images (i.e., $|\mathcal{D} \bigcup \mathcal{T}| = 988$). The label cardinality of the database, computed as $LC = \frac{1}{|\mathcal{D}|+|\mathcal{T}|} \sum_{i=1}^{|\mathcal{D}|+|\mathcal{T}|} \|\mathbf{y}_i\|_1$, is 4.22, while the label density $LD = \frac{1}{(|\mathcal{D}|+|\mathcal{T}|)Y} \sum_{i=1}^{|\mathcal{D}|+|\mathcal{T}|} \|\mathbf{y}_i\|_1$, is 0.05.

Calvary (theme), Christ, Cross | Holy Family (theme), Christ child, Mary, St. Joseph | Resurrection (theme), Christ, Soldier

Fig. 4. Example of annotation of printmaking images (from Artstor [38]) produced by an art historian.

The images are represented with the spatial pyramid [39], where the local descriptors are extracted with the scale invariant feature transform (SIFT) [40] using a uniform grid over the image and scale space in order to have 10000 descriptors per image. The spatial pyramid representation is achieved by tiling the image in three levels, as follows [41]: the first level comprises the whole image, the second level divides the image into 2×2 regions, and the third level breaks the image into 3×1 regions. The visual vocabulary is built by gathering the descriptors from all images and running a hierarchical clustering algorithm with three levels, where each node in the hierarchy has 10 descendants [42]. This results in a directed tree with $1+10+100+1000 = 1111$ vertexes, and the image feature is formed by using each descriptor of the image to traverse the tree and record the path (note that each descriptor generates a path with 4 vertexes). The histogram of visited vertexes is weighted by the node entropy (i.e., vertexes that are visited more often receive smaller weights). Using this hierarchical tree (with a total of 1111 vertexes) and the tiling described above (with 8 tiles), an image is represented by 8 histograms as in $\mathbf{x} \in \Re^X$, where $X = 8 \times 1111$. Note that in order to test the robustness of the methodologies to image representations of different dimensionalities, we also build two additional vocabularies, where each vertex of the hierarchy has 4 or 7 descendants, resulting in $X = 8 \times (1+4+16+64)$ and $8 \times (1+7+49+343)$, respectively.

A. Proposed Annotation and Retrieval Formulation

The annotation of a test image represented by $\widetilde{\mathbf{x}}$ from the test set $\mathcal{T}$ is achieved by finding the annotation vector $\mathbf{y}^*$ that solves the following optimization problem:

$$\begin{aligned} &\text{maximize } p(\mathbf{y}|\widetilde{\mathbf{x}}) \\ &\text{subject to } \mathbf{y} = [\mathbf{y}(1), ..., \mathbf{y}(M)] \in \{0,1\}^Y, \\ &\qquad \|\mathbf{y}(k)\|_1 = 1 \text{ for } \{k \in \{1, ..., M\} | |\mathbf{y}(k)| > 1\}, \end{aligned} \tag{1}$$

where $p(\mathbf{y}|\widetilde{\mathbf{x}})$ is a probability function that computes the confidence of annotating the test image $\widetilde{\mathbf{x}}$ with a vector $\mathbf{y} \in \mathcal{Y}$, with the set $\mathcal{Y}$ consisting of all the possible label annotations

present in the training set $\mathcal{D}$, whose size $|\mathcal{Y}| \leq |\mathcal{D}|$. Note that this optimization function is quite different from the one we proposed in our previous work [9], which was based on a variation of the class mass normalization [43].

The proposed retrieval is done by building a set of test images $\widetilde{\mathbf{x}}$ (with annotation $\widetilde{\mathbf{y}}$) present in the set $\mathcal{T}$ that are relevant to a query $\mathbf{q} \in \{0,1\}^Y$, as follows:

$$\mathcal{Q}(\mathbf{q},\tau) = \{(\widetilde{\mathbf{x}},\widetilde{\mathbf{y}}) | (\widetilde{\mathbf{x}},\widetilde{\mathbf{y}}) \in \mathcal{T}, (\mathbf{q}^\top \mathbf{y}^*) > 0, p(\mathbf{y}^*|\widetilde{\mathbf{x}}) > \tau\}, \tag{2}$$

where $\mathbf{y}^*$ is computed with (1), and $\tau \in [0,1]$ is a threshold.

IV. Inverse Label Propagation

We first describe the general problem of inverse label propagation, which takes a test image $\widetilde{\mathbf{x}}$ and ranks the most relevant training images $(\mathbf{x},\mathbf{y}) \in \mathcal{D}$ via a random walk process. We consider three solutions, each making different assumptions about the random walk process, as follows: 1) plain random walk (assumes a large number of independent random walks with a large number of steps); 2) stationary solution (assumes a random walk with an infinite number of steps, which generates a stationary distribution); and 3) combinatorial harmonics (assumes a one-step random walk).

All these solutions use a graph defined by $\mathcal{G} = (\mathcal{V},\mathcal{E})$, built with the training set $\mathcal{D}$, where the nodes $\mathcal{V}$ represent the images and the weights of each edge in $\mathcal{E}$ are computed based on the appearance and label similarities between the training images. Given an image $\widetilde{\mathbf{x}}$ from the test set $\mathcal{T}$, we start a random walk process in this graph by taking into account the appearance similarity between the test image and training images, and the process continues using the edge weights of the graph. Following the notation introduced by Estrada et al. [44], we define a random walk sequence of U steps as

$$\mathbf{t} = [(\mathbf{x}^{(1)},\mathbf{y}^{(1)}), ..., (\mathbf{x}^{(U)},\mathbf{y}^{(U)})], \tag{3}$$

where each $(\mathbf{x}^{(u)},\mathbf{y}^{(u)}) \in \mathcal{D}$. A core goal of our algorithm is to provide a label $\mathbf{y}$ for a test image $\widetilde{\mathbf{x}}$ by finding the argument that maximizes $p(\mathbf{y}|\widetilde{\mathbf{x}})$ in (1), which is estimated from the result of the random walk algorithm, as follows:

$$p(\mathbf{y}|\widetilde{\mathbf{x}}) = Z \sum_{r=1}^{R} p(\mathbf{y}|\mathbf{t}_r) p(\mathbf{t}_r|\widetilde{\mathbf{x}}), \tag{4}$$

where r indexes a random walk (3), R represents the total number of different (and independent) random walks, Z is a normalization factor, and $p(\mathbf{y}|\mathbf{t}_r)$ varies depending on the solution of the inverse label propagation (see below Sections IV-A to IV-C), but its goal is to estimate the likelihood of the annotation $\mathbf{y}$ given the visited nodes during the random walk (3). The computation of (4) assumes that each step of the random walk is independent of all previous steps given the one immediately before (i.e., a Markov process), so the probability of a sequence of random steps, given the test image, is

$$\begin{aligned} p(\mathbf{t}|\widetilde{\mathbf{x}}) &= p([(\mathbf{x}^{(1)},\mathbf{y}^{(1)}), ..., (\mathbf{x}^{(U)},\mathbf{y}^{(U)})]|\widetilde{\mathbf{x}}) \\ &= \left[\prod_{u=2}^{U} p((\mathbf{x}^{(u)},\mathbf{y}^{(u)})|(\mathbf{x}^{(u-1)},\mathbf{y}^{(u-1)}),\widetilde{\mathbf{x}})\right] p((\mathbf{x}^{(1)},\mathbf{y}^{(1)})|\widetilde{\mathbf{x}}) \end{aligned} \tag{5}$$

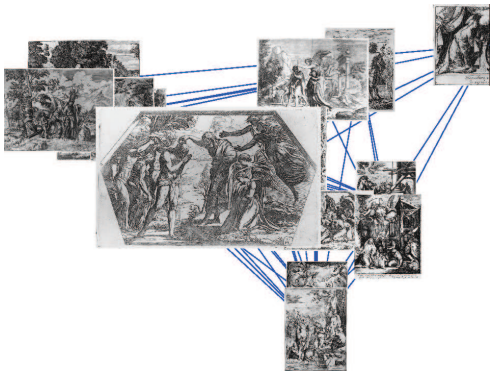


Fig. 5. Network structure in the training set built using the adjacency matrix in (7) and shown using a variant of the multidimensional scaling algorithm [45]. The large image in the center is a training image with its most similar images, in terms of visual and annotation content, appearing closer in the graph.

where $p((\mathbf{x}^{(u)},\mathbf{y}^{(u)})|(\mathbf{x}^{(u-1)},\mathbf{y}^{(u-1)}),\widetilde{\mathbf{x}})$ represents the probability of selecting training image $\mathbf{x}^{(u)}$ with annotation $\mathbf{y}^{(u)}$ for the u^{th} step of the random walk, given the test image $\widetilde{\mathbf{x}}$ and the database image $\mathbf{x}^{(u-1)}$ with annotation $\mathbf{y}^{(u-1)}$ selected by the algorithm at step $u-1$, and $p((\mathbf{x}^{(1)},\mathbf{y}^{(1)})|\widetilde{\mathbf{x}})$ denotes the probability of hopping from $\widetilde{\mathbf{x}}$ to $\mathbf{x}^{(1)}$ (with annotation $\mathbf{y}^{(1)}$) at step $u=1$.

Fig. 5 shows a part of the graph whose nodes are the training images and edges represent the similarity between image features and annotations, described by the adjacency matrix $\mathbf{W}$ defined below in (7), where we only take into account the similarities between training images. More specifically, we take a training image shown at the center (enlarged image in the figure), and display the graph structure with each node located in a 2-D space (note that only the closest 20 images are shown in this 2-D space).

A. Random Walk

The computation of $p(\mathbf{y}|\widetilde{\mathbf{x}})$ in (4) can follow a plain random walk (RW) strategy, where an adjacency matrix is used to build the probability transition matrix, as follows:

$$\mathbf{P} = \alpha \mathbf{D}^{-1}\mathbf{W} + (1-\alpha)\mathbf{1}\mathbf{v}^\top, \tag{6}$$

where $\alpha \in [0,1)$, $\mathbf{v}$ is a non-negative vector with $\|\mathbf{v}\|_1 = 1$, $\mathbf{1}$ is a column vector with ones, $\mathbf{D},\mathbf{W} \in \mathbb{R}^{|\mathcal{D}|\times|\mathcal{D}|}$, with

$$\mathbf{W}(i,j) = s_y(\mathbf{y}_i,\mathbf{y}_j) s_x(\mathbf{x}_i,\mathbf{x}_j) s_x(\mathbf{x}_i,\widetilde{\mathbf{x}}). \tag{7}$$

where the label similarity function is the Jaccard index defined by $s_y(\mathbf{y}_i,\mathbf{y}_j) = \frac{\mathbf{y}_i^\top \mathbf{y}_j}{\|\mathbf{y}_i\|^2 + \|\mathbf{y}_j\|^2 - \mathbf{y}_i^\top \mathbf{y}_j}$, and the feature similarity function is the histogram intersection defined as $s_x(\mathbf{x}_i,\mathbf{x}_j) = \sum_{d=1}^{X} \min(\mathbf{x}_i(d),\mathbf{x}_j(d))$ (i.e., this is the histogram intersection kernel over the spatial pyramid representation described in Sec. III, where $\|\mathbf{x}\|_1 = 1$). Note that the label and feature similarity functions in (7) obbey the four distance axioms [**?**], [**?**]. Also in (6), the diagonal matrix $\mathbf{D}(i,i) = \sum_j \mathbf{W}(i,j)$ normalizes the rows of $\mathbf{P}$. It is important to mention that the ergodicity [**?**] of $\mathbf{P}$ in (6) is equal to α (in this paper, we fix $\alpha = 0.85$ [30]).

The initial distribution vector (used to compute $p(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}|\widetilde{\mathbf{x}})$) takes into account only the appearance between the test image $\widetilde{\mathbf{x}}$ and all training images $\{\mathbf{x}_1, ..., \mathbf{x}_{|\mathcal{D}|}\}$, as in

$$\mathbf{p}_0 = [s_x(\mathbf{x}_1, \widetilde{\mathbf{x}}), ..., s_x(\mathbf{x}_{|\mathcal{D}|}, \widetilde{\mathbf{x}})]^\top, \quad (8)$$

where $\mathbf{p}_0$ is normalized to produce $\|\mathbf{p}_0\|_1 = 1$. Finally, to compute $p(\mathbf{y}|\widetilde{\mathbf{x}})$, we assume that the probability of $\mathbf{y}$ is independent of the random walk given the last node visited, which means that $p(\mathbf{y}|\mathbf{t}_r)$ in (4) is defined as follows:

$$p(\mathbf{y}|\mathbf{t}_r) = p(\mathbf{y}|(\mathbf{x}^{(U)}, \mathbf{y}^{(U)})) = \frac{\delta(\|\mathbf{y} - \mathbf{y}^{(U)}\|_1)}{\sum_{j=1}^{|\mathcal{D}|} \delta(\|\mathbf{y}_j - \mathbf{y}^{(U)}\|_1)} \quad (9)$$

where $\delta(.)$ is the Dirac delta function. The RW strategy consists of running R independent random walk processes, each with U steps, using the adjacency matrix in (7) and initial distribution (8). This strategy is referred to as **ILP-RW** in the experiments.

1) Running Time Complexity of the Random Walk: In terms of training, the computation of $\mathbf{W}$ involves $O(|\mathcal{D}|^2)$ operations, but by computing a sparse $\mathbf{W}$, we can reduce this complexity to $O(|\mathcal{D}| \log |\mathcal{D}|)$. For the inference, the main steps of **ILP-RW** are the computation of the distribution $\mathbf{p}_0$ with K nearest neighbors, which can be computed on average with complexity $O(\log |\mathcal{D}|)$, but has the worst case $O(|\mathcal{D}|)$ [46] . The random walk algorithm has complexity $O(R \times U)$, where $R, U << |\mathcal{D}|$ are fixed constants defined a priori, which means that the most expensive step of the inference is the initial K nearest neighbor search.

B. Stationary Solution

The stationary solution estimates the result of a random walk with an infinite number of steps [36] independently of the initial distribution. This method relies on the adjacency matrix $\mathbf{W}$ and diagonal matrix $\mathbf{D}$, both defined in (7), to build the normalized transition matrix:

$$\mathbf{T} = \mathbf{D}^{-\frac{1}{2}} \mathbf{W} \mathbf{D}^{-\frac{1}{2}}. \quad (10)$$

This solution exploits the eigenvector centrality (i.e., the eigenvector of $\mathbf{T}$ associated with the eigenvalue equals to 1) to determine the ranking of a node (recall that a node represents an image in the training database $\mathcal{D}$). This ranking denotes the likelihood of visiting the node after an infinite number of steps of a random walk process defined by $\mathbf{T}$ in (10).

Assuming a random initial distribution of the vertexes, denoted by the vector $\mathbf{v}^{(0)}$, and that, at each iteration of the random walk, the distribution underlying the decision process builds on the weighted edges and on the probability vector $\mathbf{p}_0$ in (8), we compute the stationary vector as follows [11]:

$$\mathbf{v}^{(u)} = (\alpha \mathbf{T})\, \mathbf{v}^{(u-1)} + (1-\alpha)\mathbf{p}_0 \Rightarrow \mathbf{v}^{(\infty)} = (\mathbf{I} - \alpha \mathbf{T})^{-1} (1-\alpha)\mathbf{p}_0 \quad (11)$$

where α is defined in (6) and $\mathbf{I}$ denotes the identity matrix.

The probability of annotation $\mathbf{y}$ given the test image is computed as follows:

$$p(\mathbf{y}|\widetilde{\mathbf{x}}) = Z \sum_{i=1}^{|\mathcal{D}|} \mathbf{v}^{(\infty)}(i) p(\mathbf{y}|(\mathbf{x}_i, \mathbf{y}_i)) \quad (12)$$

where $\mathbf{v}^{(\infty)}(i)$ is the i^{th} component of the stationary vector (11), $p(\mathbf{y}|(\mathbf{x}_i, \mathbf{y}_i))$ is defined as in (9) replacing $(\mathbf{x}^{(U)}, \mathbf{y}^{(U)})$ by $(\mathbf{x}_i, \mathbf{y}_i)$, and Z is a normalization factor. In the experiments, this approach is represented by the acronym **ILP-SS**.

1) Running Time Complexity of the Stationary Solution: The training is based on the computation of $\mathbf{T}$, which has run-time complexity of $O(|\mathcal{D}|^2)$, but if $\mathbf{T}$ is a sparse matrix with at most K non-zero values per row, this is reduced to $O(|\mathcal{D}| \log |\mathcal{D}|)$. For the inference, the most expensive operation stems from computing the inverse $(\mathbf{I} + \alpha \mathbf{T})^{-1}$, which in general has running time complexity $O(|\mathcal{D}|^3)$ (but efficient algorithms can reduce this complexity to $O(|\mathcal{D}|^{2.376})$ [47]).

C. Combinatory Harmonics

The combinatory harmonics solution estimates the probability of first reaching each of the database samples $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}$ in a random walk process starting at the test image $\widetilde{\mathbf{x}}$ [33]. The computation of this solution extends the adjacency matrix in (7), as in: $\widetilde{\mathbf{W}} = \begin{bmatrix} \mathbf{W} & \widetilde{\mathbf{w}} \\ \widetilde{\mathbf{w}}^T & 0 \end{bmatrix}$, where $\widetilde{\mathbf{w}}$ is the un-normalized initial distribution vector defined as $\mathbf{w} = [s_x(\mathbf{x}_1, \widetilde{\mathbf{x}}), ..., s_x(\mathbf{x}_{|\mathcal{D}|}, \widetilde{\mathbf{x}})]^\top$. Our goal is to find the distribution $\mathbf{g}^* \in \Re^{|\mathcal{D}|}$ ($\|\mathbf{g}^*\|_1 = 1$), representing the probability of first reaching each of the training images in a random walk procedure, where the labeling matrix $\mathbf{G} = \mathbf{I}$ (i.e., an $|\mathcal{D}| \times |\mathcal{D}|$ identity matrix) denotes a problem with $|\mathcal{D}|$ classes, with each training image representing a separate class. The estimation of $\mathbf{g}^*$ is based on the minimization of the following function:

$$E([\mathbf{G}, \mathbf{g}]) = \frac{1}{2} \left\| [\mathbf{G}, \mathbf{g}] \widetilde{\mathbf{L}} \begin{bmatrix} \mathbf{G}^T \\ \mathbf{g}^T \end{bmatrix} \right\|_2^2, \quad (13)$$

where $\widetilde{\mathbf{L}} = \widetilde{\mathbf{D}} - \widetilde{\mathbf{W}}$ is the Laplacian matrix computed from the the adjacency matrix $\widetilde{\mathbf{W}}$, where $\widetilde{\mathbf{D}}$ is a matrix that has the sum of the rows in the diagonal (i.e., it is a diagonal matrix). This Laplacian matrix can be divided into blocks of the same sizes as in $\widetilde{\mathbf{W}}$, that is $\widetilde{\mathbf{L}} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{B} \\ \mathbf{B}^T & \mathbf{L}_2 \end{bmatrix}$. Solving the following optimization problem produces $\mathbf{g}^*$ [33]:

$$\begin{array}{ll} \text{minimize} & E([\mathbf{G}, \mathbf{g}]) \\ \text{subject to} & \mathbf{G} = \mathbf{I}, \end{array} \quad (14)$$

which has the closed form solution [33]: $\mathbf{g}^* = (-\mathbf{L}_2^{-1} \mathbf{B}^T \mathbf{I})^\top$. Note that $\mathbf{g}^* \in [0, 1]^{|\mathcal{D}|}$ and $\|\mathbf{g}^*\|_1 = 1$.

Finally, we compute the probability of annotation $\mathbf{y}$ given the test image, as in

$$p(\mathbf{y}|\widetilde{\mathbf{x}}) = Z \sum_{i=1}^{|\mathcal{D}|} \mathbf{g}^*(i) p(\mathbf{y}|(\mathbf{x}_i, \mathbf{y}_i)) \quad (15)$$

where $\mathbf{g}^*(i)$ is the i^{th} component of the solution vector from (14), $p(\mathbf{y}|(\mathbf{x}_i, \mathbf{y}_i))$ is computed as in (12), and Z is a normalization factor. In the experiments, this approach is represented by the acronym **ILP-CH**.

1) Running Time Complexity of the Combinatorial Harmonics: The training needs to compute the adjacency matrix with complexity $O(|\mathcal{D}|^2)$, but by calculating a sparse adjacency matrix, this complexity is reduced to $O(|\mathcal{D}| \log |\mathcal{D}|)$. For the inference, the most expensive operation is the computation

of $\mathbf{B}$ and $\mathbf{L}_2$, which represents the last column of $\widetilde{\mathbf{L}}$. The computation of $\mathbf{B}$ can be simplified with a computation of its values only for the K nearest neighbors, which means that we only need to compute the relevant K rows of the matrix $\widetilde{\mathbf{W}}$. Therefore, the running time complexity is dominated by the K nearest neighbor search, which has average complexity $O(\log|\mathcal{D}|)$, but worst case $O(|\mathcal{D}|)$ followed by the computation of $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{D}}$, which means that the whole algorithm should run in $O(|\mathcal{D}|\log|\mathcal{D}|)$, on average.

V. Baseline Methodologies

The performance of our approach is compared with different visual classification methodologies that have recently shown state-of-the-art results in photographic image retrieval and annotation tasks. Specifically, we evaluate the performance of other inductive and transductive methodologies, in addition to random annotation and nearest neighbor approaches. For the inductive learning, we study the performance of bag of features and structural learning. The transductive methodology is tested with different types of label propagation methods.

A. Random Annotation

The random global annotation uses a random variable $w \sim \mathcal{U}(0,1)$, where $\mathcal{U}(0,1)$ denotes a uniform distribution between 0 and 1, and the optimal annotation $\mathbf{y}^*$ is defined based on the priors of the visual classes, as follows:

$$\begin{array}{lll} \text{Multi-class:} & \mathbf{y}^*(k) = \left\{ \begin{array}{ll} \pi_1, & w < p(\mathbf{y}(k)=\pi_1) \\ \vdots & \\ \pi_{|\mathbf{y}(k)|}, & \sum_{y=1}^{|\mathbf{y}(k)|-1} p(\mathbf{y}(k)=\pi_y) \leq w \end{array} \right. \\ \text{Binary:} & \mathbf{y}^*(k) = \left\{ \begin{array}{ll} 1, & w < p(\mathbf{y}(k)=1) \\ 0, & \text{otherwise} \end{array} \right. , \end{array} \tag{16}$$

where $p(\mathbf{y}(k)=\pi_y) = \frac{1}{|\mathcal{D}|}\sum_{i=1}^{|\mathcal{D}|}\mathbf{y}(k)_i^\top \pi_y$ denotes the class priors with $\pi_y = 1$ for binary problems and $\pi_y \in \{0,1\}^{|\mathbf{y}(k)|}$ with zeros everywhere except at the y^{th} position for multi-class problems. The probability of the random annotation $\mathbf{y}^*$ from (16) is computed with $p(\mathbf{y}^*|\widetilde{\mathbf{x}}) = \prod_{k=1}^{M} p(\mathbf{y}(k) = \mathbf{y}^*(k))$ using the class priors defined above. In the experiments, this approach is named **RND**.

B. Nearest Neighbor

The nearest neighbor annotation $\mathbf{y}^*$ for a test image represented by $\widetilde{\mathbf{x}}$ is produced by taking the annotation of the closest training image, as follows: $(\mathbf{y}^*, \mathbf{x}^*) = \arg\min_{(\mathbf{y}_i,\mathbf{x}_i)\in\mathcal{D}} \|\mathbf{x}_i - \widetilde{\mathbf{x}}\|_2$. In the experiments, this approach is represented with **NN**.

C. Bag of Features

The bag of features model is based on Y support vector machine (SVM) classifiers using the one-versus-all training method. Specifically, we train the Y classifiers (each classifier for each label) $p(\mathbf{y}(k) = \pi_y|\widetilde{\mathbf{x}}, \theta_{SVM}(k,y))$, for $k \in \{1,...,M\}$, $y \in \{1,...,|\mathbf{y}(k)|\}$, $\pi_y \in \{0,1\}^{|\mathbf{y}(k)|}$ (with the y^{th} element equal to one and rest are zero), and the annotation and retrieval use the same methods in (1) and (2), respectively, assuming $p(\mathbf{y}|\widetilde{\mathbf{x}}) = \prod_{k=1}^{M}\prod_{y=1}^{\mathbf{y}(k)} p(\mathbf{y}(k) = \pi_y|\widetilde{\mathbf{x}}, \theta_{SVM}(y))$. The penalty factor of the SVM for the slack variables is determined via cross-validation, where the training set $\mathcal{D}$ is divided into a training and validation sets of 90% and 10% of $\mathcal{D}$, respectively. This model roughly represents the state-of-the-art approach for image annotation and retrieval problems [48], but notice that the fact that it assumes one independent classifier for each class represents a disadvantage of this approach. This approach is represented by the acronym **BoF** in the experiments.

1) Running Time Complexity of the Bag of Features: The training stage is quite complex given that it it involves: 1) the implementation of the visual vocabulary with the hierarchical k-means (complexity of $O(|\mathcal{D}|^2)$); and 2) training of one SVM classifier per class (with worst case complexity $O(|\mathcal{D}|^3)$ due to the need of inverting a $|\mathcal{D}| \times |\mathcal{D}|$ matrix, but much more efficient algorithms have been proposed recently [49]). The inference only involves a linear product between the support vectors and the test image feature, which has complexity $O(Y \times S)$, with S being the number of support vectors.

D. Label Propagation

The label propagation encodes the similarity between pairs of images using the graph Laplacian, and estimate the annotations of test image using transductive inference. This method has been intensively investigated, but we only present the main developments proposed in this area of research. The main objective of label propagation techniques is to find the annotation matrix $\mathbf{F}^*$ using the following optimization problem [11]:

$$\begin{array}{ll} \text{minimize} & 0.5\ \text{tr}(\mathbf{F}^\top(\mathbf{D}-\mathbf{W})\mathbf{F}) \\ \text{subject to} & \mathbf{f}_i = \mathbf{y}_i,\ \text{for } i = 1,...,|\mathcal{D}| \end{array} , \tag{17}$$

where $\mathbf{F} \in \Re^{(|\mathcal{D}|+|\mathcal{T}|)\times Y}$, $\mathbf{W}, \mathbf{D} \in \Re^{(|\mathcal{D}|+|\mathcal{T}|)\times(|\mathcal{D}|+|\mathcal{T}|)}$, $\mathbf{W}_{ij} = s_x(\mathbf{x}_i, \mathbf{x}_j)$ with $s_x(.)$ defined in (7) such that the index for the training set is from 1 to $|\mathcal{D}|$ and for the test set from $|\mathcal{D}|+1$ to $|\mathcal{D}|+|\mathcal{T}|$, $\mathbf{D}$ is a diagonal matrix with its (i,i)-element equal to the sum of the i^{th} row of $\mathbf{W}$, and tr(.) is an operator that computes the trace of a matrix. This problem has the closed form solution $\mathbf{F}^* = \beta(\mathbf{I} - \alpha(\mathbf{D}-\mathbf{W}))^{-1}\mathbf{Y}$, where $\mathbf{I}$ denotes the identity matrix, $\mathbf{Y}^\top = [\mathbf{y}_1,...,\mathbf{y}_{|\mathcal{D}|},\mathbf{0},...,\mathbf{0}] \in \Re^{Y\times(|\mathcal{D}|+|\mathcal{T}|)}$ and α and β are regularization parameters such that $\alpha + \beta = 1$. In the experiments, this approach is named **LP**, as defined in Sec. IV-A, $\alpha = 0.85$, which means that $\beta = 0.15$. The problem in (17) has been extended in order to include label correlation [12], [13], as follows

$$\begin{array}{l} \text{minimize } 0.5\ \text{tr}(\mathbf{F}^\top(\mathbf{D}-\mathbf{W})\mathbf{F}) + \\ \quad (1-\mu)\text{tr}((\mathbf{F}-\mathbf{Y})\Lambda(\mathbf{F}-\mathbf{Y})) + \mu\text{tr}(\mathbf{F}\mathbf{C}\mathbf{F}^\top), \end{array} \tag{18}$$

where Λ is a matrix containing ones in the diagonal from indexes 1 to $|\mathcal{D}|$, and zero otherwise, and $\mathbf{C} \in [-1,1]^{Y\times Y}$ contains the correlation between classes. The problem in (18) has the closed form solution $\mathbf{F}^* = (\mathbf{D}-\mathbf{W})^{-1}\mathbf{Y}(\mathbf{I}-\mu\mathbf{C})$, where μ is a regularization parameter. We represent this approach by **LP-CC** in the experiments. After finding $\mathbf{F}^*$ using (17) or (18), the values for $\mathbf{y}_i^*$ for each test image is estimated with class mass normalization [43], which adjusts the class distributions to match the priors. Notice that with (17) and (18), followed by class mass normalization, it is possible

to obtain the optimal $\mathbf{y}^*$, but the retrieval problem can only build the set of test images that have $\mathbf{q}^\top \mathbf{y}^* > 0$ without any confidence measure (i.e., the image is either retrieved or not given a certain query). Hence, we still use (2), but we can no longer sort the test images from most to least relevant to a given query.

1) Running Time Complexity of the Label Propagation: The training involves pre-computing part of the matrix $\mathbf{W}$ related to the training set with complexity $O(|\mathcal{D}|^2)$, but similarly to the inverse label propagation methods, we can compute a sparse matrix $\mathbf{W}$ with run-time complexity $O(|\mathcal{D}| \log |\mathcal{D}|)$. The inference involves the computation of the inverses $(\mathbf{I} - \alpha(\mathbf{D} - \mathbf{W}))^{-1}$ in (17) or $(\mathbf{D} - \mathbf{W})^{-1}$ in (18), which in general has running time complexity $O((|\mathcal{D}| + |\mathcal{T}|)^3)$ (but efficient algorithms can reduce this complexity to $O((|\mathcal{D}| + |\mathcal{T}|)^{2.376})$ [47]).

E. Matrix Completion

The matrix completion formulation consists of forming a joint matrix with annotation and features, as follows [14]:

$$\begin{array}{ll} \text{minimize} & \text{rank}(\mathbf{Z}) \\ \text{subject to} & \mathbf{Z}_\mathbf{y} = [\mathbf{y}_1 ... \mathbf{y}_{|\mathcal{D}|}],\ \mathbf{Z}_\mathbf{x} = [\mathbf{x}_1 ... \mathbf{x}_{|\mathcal{D}|}], \\ & \mathbf{Z}_{\widetilde{\mathbf{x}}} = [\widetilde{\mathbf{x}}_1 ... \widetilde{\mathbf{x}}_{|\mathcal{T}|}]. \end{array} \quad (19)$$

where $\mathbf{Z} = \begin{bmatrix} \mathbf{Z}_\mathbf{y} & \mathbf{Z}_{\mathbf{y}^*} \\ \mathbf{Z}_\mathbf{x} & \mathbf{Z}_{\widetilde{\mathbf{x}}} \end{bmatrix}$, and goal is to find the values for $\mathbf{Z}_{\mathbf{y}^*} = [\mathbf{y}^*_1 ... \mathbf{y}^*_{|\mathcal{T}|}]$. In (19), the non-convex minimization objective function rank(.) is replaced by the convex nuclear norm $\|\mathbf{Z}\|_* = \sum_{k=1}^{min\{|\mathcal{D}|, Y+X\}} \sigma_k(\mathbf{Z})$, where the $\sigma_k(\mathbf{Z})$ represents the singular values of $\mathbf{Z}$. Moreover, the equality constraints for $\mathbf{Z}_\mathbf{x}$ and $\mathbf{Z}_{\widetilde{\mathbf{x}}}$ are replaced by squared losses, and the constraint for $\mathbf{Z}_\mathbf{y}$ is relaxed to a logistic loss. After finding $\mathbf{Z}_{\mathbf{y}^*}$, the values for $\mathbf{y}^*_i$ for each test image are estimated with class mass normalization [43]. Similarly to the label propagation methods described above, we are able to obtain the optimal $\mathbf{y}^*$, but the retrieval problem described in (2) cannot sort the test images because we never compute $p(\mathbf{y}|\widetilde{\mathbf{x}})$. This approach is represented by the acronym **MC** in the experiments.

1) Running Time Complexity of the Matrix completion: For the matrix completion, all test images are placed in matrix Z, and the resulting annotation is computed for all of them simultaneously at the inference procedure, which means that there is no training stage. The algorithm in [14] consists of a fixed point continuation method, whose main task is the computation of the SVD of $\mathbf{Z}$, which has a computational complexity of $O(|\mathcal{D}|^3)$.

F. Structural Learning

The structural learning formulation follows the structured SVM implementation [15], which is based on the margin maximization quadratic problem, defined by:

$$\begin{array}{ll} \text{minimize}_{\mathbf{w},\xi} & \|\mathbf{w}\|^2 + C \sum_{i=1}^{|\mathcal{D}|} \xi_i \\ \text{subject to} & \mathbf{w}^\top \Psi(\mathbf{y}_i, \mathbf{x}_i) - \mathbf{w}^\top \Psi(\mathbf{y}, \mathbf{x}_i) + \xi_i \geq \Delta(\mathbf{y}_i, \mathbf{y}), \\ & \quad i = 1...|\mathcal{D}|, \ \ \forall \mathbf{y} \in \{0,1\}^Y, \\ & \xi_i \geq 0, \ \ i = 1...|\mathcal{D}| \end{array} \quad (20)$$

where $\Delta(\mathbf{y}_i, \mathbf{y}) = \|\mathbf{y}_i - \mathbf{y}\|_1$, $\Psi(\mathbf{y}, \mathbf{x}) = \mathbf{x} \otimes \mathbf{y} \in \Re^{X \times Y}$ (i.e., this is a tensor product combining the vectors $\mathbf{x}$ and $\mathbf{y}$ by replication the values of $\mathbf{x}$ in every dimension $y \in \{1, ..., Y\}$ where $\mathbf{y}^\top \pi_y = 1$), C is penalty for non-separable points, and ξ_d denotes the slack variables to deal with non-separable problems. The inference problem is simply $\mathbf{y}^* = \arg\max_{\mathbf{y} \in \mathcal{Y}} \mathbf{w}^\top \Psi(\mathbf{y}, \widetilde{\mathbf{x}})$ using $\mathbf{w}$ learned with (20), and $\mathcal{Y}$ denoting the set of all possible annotations in the training set. Similarly to the cases above, we can estimate $\mathbf{y}^*$, but we do not compute $p(\mathbf{y}|\widetilde{\mathbf{x}})$, which means that retrieval problem described in (2) cannot sort the test images. We represent this approach with the acronym **SL** in the experiments.

1) Running Time Complexity of the Structural Learning: Structural learning is in general solved with large margin methods (e.g., cutting plane approaches), which has complexity $O(|\mathcal{D}|^2)$ with the use of kernels. The inference problem is based on testing all annotations in $\mathcal{Y}$ in order to determine the optimal $\mathbf{y}^*$, which in the worst case has complexity $O(|\mathcal{D}| \times X \times Y)$.

VI. Experiments

For the experiments, we run a 10-fold cross validation, where the database is divided into a training set $\mathcal{D}$ with 90% of the original data points (i.e., $|\mathcal{D}| = 889$), and a test set $\mathcal{T}$ with the remaining 10% of the points (i.e., $|\mathcal{T}| = 99$). We compute several retrieval and annotation errors and display the results using the average and standard deviation in this 10-fold cross validation experiment. We explain how the errors are computed in Sec. VI-A, and show the results for our methods and the baseline approaches in Sec. VI-B. Also, the values of several parameters in our models are defined by dividing the initial training set of 889 images into training and validation sets (with 90% and 10% of the original size of the training set, respectively), and the values achieved for each parameter are: number of nearest neighbors is $K = 20$ in Sec. IV-A and Sec. IV-C; length of random walk is $U = 10$ in (3); number of random walks is $R = 100$ in (4). Finally, in order to assess the scalability of the methodology with respect to the database size, we run the same set of experiments with a database of 307 images, where $|\mathcal{D}| = 276$ and $|\mathcal{T}| = 31$ (note that in this database the label cardinality is 5.45, the label density is 0.19, and the dimensionality of the label vector is 28, with one multiclass problem containing 7 classes and 21 binary problems); Furthermore, the scalability in terms of the image representation dimensionality is estimated by modifying the hierarchical clustering algorithm with three levels, where each node has $\{4, 7, 10\}$ descendants (Sec. III) [1].

A. Retrieval and Annotation Error Measures

We compare the performance of the annotation methodologies using three different types of measures. The first computes the *retrieval* performance by taking the mean (over all visual classes) of the mean average precision (MAP) [50], which is

[1]Recall that the acronyms for the methodologies tested in this section are defined as follows: inverse label propagation (ILP), combinatorial harmonics (CH), stationary solution (SS), random walk (RW), bag of features (BoF), label propagation (LP), class label correlation (CC), matrix completion (MC), structural learning (SL), random (RND), nearest neighbor (NN).

TABLE I
RETRIEVAL, LABEL-BASED AND EXAMPLE-BASED RESULTS OF ALL METHODOLOGIES USING THE MEAN AND STANDARD DEVIATION OF THE MEASURES DESCRIBED IN SEC. VI-A. THE HIGHLIGHTED VALUE IN EACH COLUMN INDICATES THE HIGHEST FOR EACH MEASURE.

	Retrieval	Label-based annotation			Example-based annotation			
Models	Label MAP	Average Precision	Average Recall	Average F1	Average Precision	Average Recall	Average F1	Average Accuracy
ILP-CH	$\mathbf{0.18 \pm .04}$	$\mathbf{0.26 \pm .05}$	$\mathbf{0.26 \pm .05}$	$\mathbf{0.26 \pm .05}$	$0.39 \pm .03$	$\mathbf{0.39 \pm .04}$	$\mathbf{0.38 \pm .03}$	$\mathbf{0.33 \pm .03}$
ILP-SS	$0.12 \pm .01$	$0.15 \pm .02$	$0.16 \pm .05$	$0.15 \pm .04$	$0.24 \pm .04$	$0.24 \pm .04$	$0.23 \pm .04$	$0.20 \pm .04$
ILP-RW	$0.10 \pm .01$	$0.10 \pm .03$	$0.13 \pm .02$	$0.11 \pm .03$	$0.33 \pm .03$	$0.36 \pm .03$	$0.34 \pm .03$	$0.26 \pm .03$
BoF	$0.12 \pm .05$	$0.14 \pm .11$	$0.10 \pm .06$	$0.11 \pm .08$	$\mathbf{0.47 \pm .05}$	$0.26 \pm .08$	$0.30 \pm .05$	$0.23 \pm .05$
LP	$0.11 \pm .01$	$0.12 \pm .02$	$0.12 \pm .02$	$0.12 \pm .02$	$0.32 \pm .03$	$0.28 \pm .02$	$0.26 \pm .02$	$0.19 \pm .01$
LP-CC	$0.11 \pm .01$	$0.13 \pm .02$	$0.14 \pm .02$	$0.13 \pm .02$	$0.27 \pm .03$	$0.26 \pm .03$	$0.25 \pm .03$	$0.18 \pm .02$
MC	$0.17 \pm .01$	$0.24 \pm .03$	$0.11 \pm .02$	$0.15 \pm .02$	$\mathbf{0.47 \pm .03}$	$0.28 \pm .02$	$0.32 \pm .02$	$0.25 \pm .02$
SL	$0.14 \pm .01$	$0.20 \pm .04$	$0.15 \pm .03$	$0.17 \pm .03$	$0.37 \pm .04$	$0.32 \pm .03$	$0.34 \pm .03$	$0.28 \pm .03$
RND	$0.08 \pm .06$	$0.06 \pm .01$	$0.07 \pm .01$	$0.06 \pm .01$	$0.26 \pm .02$	$0.21 \pm .01$	$0.22 \pm .01$	$0.15 \pm .01$
NN	$0.13 \pm .01$	$0.17 \pm .02$	$0.17 \pm .04$	$0.17 \pm .03$	$0.32 \pm .04$	$0.32 \pm .03$	$0.31 \pm .03$	$0.26 \pm .03$

defined as the average precision over all queries, at the ranks that the recall changes. The second computes the *label-based annotation* performance by taking the mean (over all visual classes) of the precision, recall and F1 values [50]. Finally, the third measure assesses the *example-based annotation* performance of the full annotation produced by each methodolgy by computing the mean (over all test examples) of the precision, recall, F1 and accuracy values [50].

B. Results

We first show in Fig. 6, an experiment about the scalability of the methodologies with respect to database size and to feature dimensionality. In the first row, it is displayed the retrieval, label-based and example-based annotation performances (error bars with mean and standard deviation) of all methodologies with respect to the training set size, where the database with 307 images has 28 visual classes and the one with 988 images has 75 classes. The second row shows the performance of all methodologies as a function of the dimensionality of image representation. Table VI shows the retrieval, label-based and example-based annotation performances (mean and standard deviation) of all methodologies using the database with 988 images and the hierarchical tree with 1111 visual words (see Sec. III). Figure 7 shows the annotation produced by **ILP-CH** in several test images.

C. Running Times

We measure how much time it takes, on average, for the training and inference procedures of each methodology, and the results are shown in Table II, where we state whether the method is transductive (T) or inductive (I). Note that for the methods **LP**, **LP-CC**, and **MC** the inference results are produced on the annotation of all 99 test images simultaneously, while for the other methods, the results are shown per image. Also, all running times have been evaluated using Matlab implementations of the algorithms on the following computer: MacBook Pro, 2.3GHz Intel Core i5 with 4GB of memory.

D. Discussion

According to the results in Sec. VI-B, we can conclude that for the database used in this paper, our proposed approach, called **ILP-CH**, leads to the most accurate annotation and retrieval results, when compared to the competing methodologies presented in Sec. V. We can also see that **ILP-CH** also leads to the more efficient training and inference procedures. The results for **ILP-SS** and **ILP-RW** are not competitive enough in this database. Structural learning and matrix completion methodologies produce relatively worse results than **ILP-CH**, but they are competitive for most measures. We suspect that larger training sets can improve the annotation and retrieval results of these approaches. However, the main issues with **SL** and **MC** are with respect to their training and inference running times, respectively. Label propagation is surprisingly worse than other approaches in terms of the retrieval and annotation results. Finally, **BoF** is competitive, but the time to train the models is much larger than for other approaches.

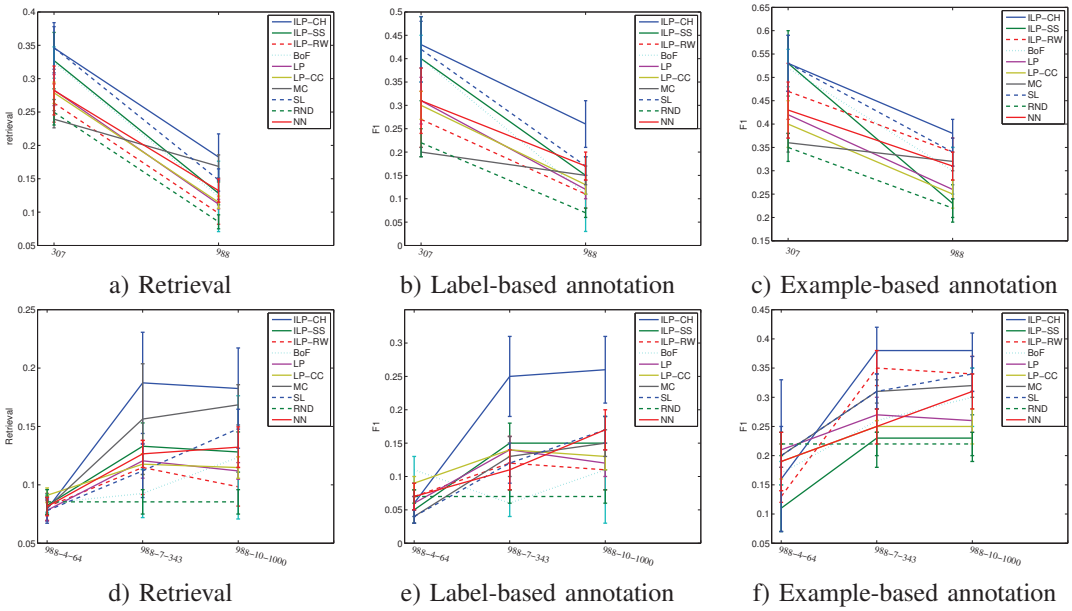
It is interesting to see the behavior of all approaches in terms of the database size and the feature dimensionality. In general, all methods produce more accurate results for smaller databases because the number of image classes is smaller. More specifically, the database of 307 images contains 28 classes, while the 988-image database has 75 classes. It is also worth observing that all methods present a similar degradation slope in terms of the database size, except for MC, which presents a better robustness (it remains to be studied the reason for that better robustness). Also, all methods have similar performances with the feature representations that use a hierarchical tree, where each node has either 7 or 10 descendants, but the performance deteriorates considerably for all approaches with a hierarchical tree, where each node has 4 descendants.

VII. CONCLUSIONS

In this paper, we introduce the problem of artistic image annotation and retrieval and propose several solutions using graph-based learning techniques. The methodologies proposed

TABLE II
RUNNING TIMES FOR THE TRAINING AND INFERENCE PROCEDURES OF EACH METHODOLOGY (IN BRACKETS, IT IS INDICATED IF THE METHODOLOGY IS A TRANSDUCTIVE (T) OR AN INDUCTIVE (I) MODEL).

Methodology	Training	Inference
ILP-CH (T)	119s	0.7s
ILP-SS (T)	119s	4s
ILP-RW (T)	119s	3.8s
BoF (I)	2×10^4s	1.9s
LP (T)	149s	0.85s
LP-CC (T)	149s	0.85s
MC (T)	0s	365s
SL (I)	2.5×10^5s	3s



a) Retrieval b) Label-based annotation c) Example-based annotation

d) Retrieval e) Label-based annotation f) Example-based annotation

Fig. 6. Scalability of the methodologies with respect to training set size (first row), and dimensionality of the image representation (second row). The hor. axis in (a)-(c) indicates the database size, which are 307 and 988 images. In (d)-(f), the hor. axis represents the dimensionality of the image representation with the first number indicating the database size (fixed at 988), number of descendants in $\{4, 7, 10\}$, and total number of leaves in the hierarchical tree in $\{64, 343, 1000\}$ (see Sec. III).

Manual:	Holy Family (theme), Christ child, Mary, St. Joseph	Circumcision of Christ (theme), Christ child, Mary, Priest, St. Joseph	Nativity (theme), Christ child, Mary	Judith (theme), Holofernes, Judith, Maidservant
Automated:	Holy Family (theme), Christ child, Mary, St. Joseph	Circumcision of Christ (theme), Christ child, Mary, Priest, St. Joseph	Nativity (theme), Christ child, Mary, St. Joseph	Judith (theme), Holofernes' head, Judith

Fig. 7. Annotation results from **ILP-CH** with the manually annotated ground truth for reference.

represent a relatively new transductive inference based on a random walk approach, where the technique that produced the best results in terms of accuracy and efficiency is an inverse label propagation approach based on combinatorial harmonics [33]. The annotation and retrieval results of the proposed method are compared to the results produced by several state-of-the-art approaches based on: bag of features, label propagation, matrix completion, and structural learning. These results are computed in a database of annotated artistic images containing 988 annotated images in a multi-label problem containing 49 visual classification problems divided into a multi-class problem with 27 classes and 48 binary problems. We plan to investigate further the role of composition of visual classes in artistic images, which implies the detection of objects in such images. We also plan to study the influence of human and animal poses in artistic image analysis. Finally, other interesting topics of investigation are the study of new image representations and new kernel functions to compute $\mathbf{W}$ in (7).

Acknowledgments: I would like to thank Duarte Lázaro and Rosário Carvalho for their help with the art history issues. I also would like to thank David Lowe for valuable suggestions on the development of this work and the anonymous reviewers for the helpful suggestions I would like to acknowledge the use of SVMLight by Thorsten Joachims, and the matrix completion code developed by Ricardo Cabral.

REFERENCES

[1] J. Li and J. Wang, "Studying digital imagery of ancient paintings by mixtures of stochastic models," *IEEE Trans. Image Processing*, vol. 13, no. 3, pp. 340–353, 2004.
[2] D. Graham, J. Friedenberg, D. Rockmore, and D. Field, "Mapping the similarity space of paintings: image statistics and visual perception," *Visual Cognition*, vol. 18, no. 4, pp. 559–573, 2010.
[3] Http://www.visual-arts cork.com/printmaking.htm.
[4] F. Baumann, K. Zurich, E. Degas, M. Karabelnik, and J. Boggs, *Degas Portraits: Portraits*. London: Merrell Holberton, 1995.
[5] B. N. Prieto, "Flandes e italia en la pintura barroca madrilea," in *Fuentes y Modelos de la Pintura Barroca Madrilea*, 2008.
[6] T. Campos, "Application des regles iconographiques aux azulejos portugais du xviieme siecle," in *Europalia*, 1991, pp. 37–40.
[7] R. Carvalho, "O programa artistico da ermida do rei salvador do mundo em castelo de vide, no contexto da arte barroca," *Artis -Revista do Instituto de Historia da Arte da Faculdade de Letras de Lisboa*, no. 2, pp. 145–180, 2003.
[8] V. N. Vapnik, *Statistical Learning Theory.* Wiley, 1998.

[9] G. Carneiro, "Graph-based methods for the automatic annotation and retrieval of art prints," in *Proceedings of the ACM International Conference on Multimedia Retrieval*, 2011.
[10] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *In Workshop on Statistical Learning in Computer Vision, ECCV*, 2004, pp. 1–22.
[11] X. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Scholkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems 16*, 2004.
[12] H. Wang, H. Huang, and C. Ding, "Image annotation using multi-label correlated green's function," in *ICCV*, 2009, pp. 2029–2034.
[13] Z. Zha, T. Mei, J. Wang, Z. Wang, and X. Hua, "Graph-based semi-supervised learning with multi-label," in *ICME*, 2008, pp. 1321–1324.
[14] A. B. Goldberg, X. Zhu, B. Recht, J. Xu, and R. D. Nowak, "Transduction with matrix completion: Three birds with one stone," in *NIPS*, 2010, pp. 757–765.
[15] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *JMLR*, vol. 6, pp. 1453–1484, 2005.
[16] C. Johnson, E. Hendriks, I. Berezhnoy, E. Brevdo, S. Hughes, I. Daubechies, J. Li, E. Postma, and J. Wang, "Image processing for artistic identification: Computerized analysis of vincent van goghs brushstrokes," *IEEE Signal Processing Magazine*, pp. 37–48, 2008.
[17] H. Maitre, F. Schmitt, and C. Lahanier, "15 years of image processing and the fine arts," in *IEEE Int. Conf. Image Processing*, 2001, pp. 557–561.
[18] D. Stork, "Computer image analysis of paintings and drawings: An introduction to the literature," in *Proceedings of the Image Processing for Artist Identification Workshop*, 2008.
[19] I. Berezhnoy, E. Postma, and H. van den Herik, "Computerized visual analysis of paintings," in *Int. Conf. Association for History and Computing*, 2005, pp. 28–32.
[20] S. Lyu, D. Rockmore, and H. Farid, "A digital technique for art authentication," *Proceedings of the National Academy of Sciences USA*, vol. 101, no. 49, pp. 17 006–17 010, 2004.
[21] G. Polatkan, S. Jafarpour, A. Brasoveanu, S. Hughes, and I. Daubechies, "Detection of forgery in paintings using supervised learning," in *International Conference on Image Processing*, 2009.
[22] J. Hughes, D. Graham, and D. Rockmore, "Stylometrics of artwork: uses and limitations," in *Proceedings of SPIE: Computer Vision and Image Analysis of Art*, 2010.
[23] S. Jafarpour, G. Polatkan, I. Daubechies, S. Hughes, and A. Brasoveanu, "Stylistic analysis of paintings using wavelets and machine learning," in *European Signal Processing Conference*, 2009.
[24] R. Datta, D. Joshi, J. Li, and J. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, no. 2, 2008.
[25] M. Yelizaveta, C. Tat-Seng, , and R. Jain, "Semi-supervised annotation of brushwork in paintings domain using serial combinations of multiple experts," in *ACM Multimedia*, 2006, pp. 529–538.
[26] O. Dekel and O. Shamir, "Multiclass-multilabel classification with more classes than examples," in *International Conference on Artificial Intelligence and Statistics*, 2010.
[27] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in *International Conference on Computer Vision*, 2009.
[28] D. Hsu, S. Kakade, J. Langford, and T. Zhang, "Multi-label prediction via compressed sensing," in *Neural Information Processing Systems*, 2009.
[29] A. Borodin, G. Roberts, J. Rosenthal, and P. Tsaparas, "Link analysis ranking: algorithms, theory, and experiments," *ACM Trans. Internet Techn.*, vol. 5, no. 1, pp. 231–297, 2005.
[30] S. Brin and L. Page, "The anatomy of a large-scale hypertextualweb search engine," in *Computer Networks and ISDN Systems*, 1998, pp. 107–117.
[31] A. Ng, A. Zheng, and M. Jordan, "Link analysis, eigenvectors and stability," in *IJCAI*, 2001, pp. 903–910.
[32] X. Zhu, "Semi-supervised learning literature survey," Computer Sciences, University of Wisconsin-Madison, Tech. Rep. 1530, 2005.
[33] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, 2006.
[34] M. Szummer and T. Jaakkola, "Partially labeled classification with markov random walks," in *NIPS*, 2001, pp. 945–952.
[35] X. He, W.-Y. Ma, and H.-J. Zhang, "Imagerank: Spectral techniques for structural analysis of image database," in *IEEE International Conference on Multimedia and Expo*, 2003, pp. 25–28.
[36] Y. Jing and S. Baluja, "Visualrank: Applying pagerank to large-scale image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1877–1890, 2008.
[37] J. Liu, M. Li, Q. Liu, H. Lu, and S. Ma, "Image annotation via graph learning," *Pattern Recognition*, vol. 42, no. 2, pp. 218–228, 2009.
[38] Http://www.artstor.org.
[39] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Object Categorization: Computer and Human Vision Perspectives*, 2009, pp. 1–37.
[40] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[41] Http://pascallin.ecs.soton.ac.uk/challenges/VOC/.
[42] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *CVPR*, 2006, pp. 2161–2168.
[43] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *ICML*, 2003, pp. 912–919.
[44] F. Estrada, D. Fleet, , and A. Jepson, "Stochastic image denoising," in *BMVC*, 2009.
[45] I. Borg and P. Groenen, *Modern Multidimensional Scaling: theory and applications*. Springer-Verlag, 2005.
[46] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms (3rd ed.)*. MIT Press, 2009.
[47] D. Coppersmith and S. Winograd, "Matrix multiplication via arithmetic progressions," *Journal of Symbolic Computation*, vol. 9, no. 3, pp. 251–280, 1990.
[48] K. V. de Sande, T. Gevers, and A. Smeulders, "The university of amsterdams concept detection system at imageclef 2009," in *CLEF working notes 2009*, 2009.
[49] E. Hazan, T. Koren, and N. Srebro, "Beating sgd: Learning svms in sublinear time," in *NIPS*, 2011, pp. 1233–1241.
[50] S. Nowak, H. Lukashevich, P. Dunker, and S. Rüger, "Performance measures for multilabel evaluation: a case study in the area of image classification," in *Multimedia Information Retrieval*, 2010, pp. 35–44.

Gustavo Carneiro received the BS and MSc degrees in computer science from the Federal University of Rio de Janeiro, and the Military Institute of Engineering, Brazil, in 1996 and 1999, respectively. Dr. Carneiro received the PhD degree in Computer Science from the University of Toronto, Canada, in 2004. Currently he is a senior lecturer at the School of Computer Science of the University of Adelaide in Australia. Previously, Dr. Carneiro worked at the Instituto Superior Técnico (IST), Technical University of Lisbon from 2008 to 2011 as a visiting researcher, and from 2006-2008, he worked at Siemens Corporate Research in Princeton, USA. He is the recipient of a Marie Curie International Incoming Fellowship and has authored more than 40 peer-reviewed publications in international journals and conferences. His research interests include medical image analysis, image feature selection and extraction, content-based image retrieval and annotation, and general visual object classification.