

Hierarchical Annotation of Medical Images

Ivica Dimitrovski^{a,b,*}, Dragi Kocev^a, Suzana Loskovska^b, Sašo Džeroski^a

^aDepartment of Knowledge Technologies, Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia

^bDepartment of Computer Science and Computer Engineering, Faculty of Electrical Engineering and Information Technologies, Rugjer Boshkovik bb, 1000 Skopje, Republic of Macedonia

Abstract

This paper presents a hierarchical multi-label classification (HMC) system for medical image annotation. HMC is a variant of classification where an instance may belong to multiple classes at the same time and these classes/labels are organized in a hierarchy. Our approach to HMC exploits the annotation hierarchy by building a single predictive clustering tree (PCT) that can simultaneously predict all of the annotations of an image. Hence, PCTs are very efficient: a single classifier is valid for the hierarchical semantics as a whole, as compared to other approaches that produce many classifiers, each valid just for one given class. To improve the performance, we construct ensembles of PCTs. We evaluate our system on the IRMA database that consists of X-ray images. We apply four feature extraction techniques commonly used in the context of X-ray images. Moreover, we investigate whether combinations of these techniques increase the predictive performance. The experiments show that our system outperforms the best-performing approach from the literature (a collection of SVMs, each predicting one label at the lowest level of the hierarchy), both in terms of error and efficiency. This holds across a range of descriptors and descriptor combinations, regardless of the type of feature fusion used. To stress the generality of the proposed approach, we have also applied it for automatic annotation of a large number of consumer photos with multiple annotations organized in semantic hierarchy. The obtained results show that this approach is general and easily applicable in different domains offering state-of-the-art performance.

Keywords: Automatic Image Annotation, Hierarchical Multi-Label Classification, Predictive Clustering Trees, Feature Extraction from Images

1. Introduction

Digital imaging in medicine is in constant growth due to the increasing availability of imaging equipment in the hospitals. Average-sized radiology departments now produce several tera-bytes of data annually. This prompts for efficient systems for image annotation, storage, retrieval and mining. Typically, the medical image databases are accessible by textual information through the standard Picture Archiving and Communication System (PACS) [1], [2]. The PACS integrates

*Corresponding author (telephone: +389 2 3099 159)

Email addresses: ivicad@feit.ukim.edu.mk (Ivica Dimitrovski), Dragi.Kocev@ijs.si (Dragi Kocev), suze@feit.ukim.edu.mk (Suzana Loskovska), Saso.Dzeroski@ijs.si (Sašo Džeroski)

imaging modalities and interfaces with hospital and departmental information systems to manage storage and distribution of images to medical personal, researchers, clinics, and imaging centers. The important requirement of PACS is an efficient search function to access required images.

An universal format for PACS image storage and retrieval is the Digital Imaging and Communications in Medicine (DICOM) standard [3]. DICOM is a well known standard for handling, storing, printing, and transmitting information in medical imaging. The DICOM header contains tags to decode the body part examined, the patient position and the acquisition modality. Some of the tags are automatically set by the digital system according to the imaging protocol used to capture the pixel data. Other part of the tags are set manually by the physicians or radiologists during the routine documentation. This procedure cannot always be considered very reliable, since frequently some entries are either missing, false, or do not describe the anatomic region precisely [4]. Thus, an automated categorization relying only on DICOM tags is impossible. Furthermore, manual annotation of images is an expensive and time-consuming procedure, especially given the large and constantly growing databases of medical images.

Automatic image annotation or image classification is an important step in image retrieval. In the medical domain, using information directly extracted from images to annotate/categorize them will improve the quality of image annotation in particular, and more generally the quality of patient care. Properly classified medical image data can help medical professionals in fast and effective access to data in their teaching, research, training, and diagnostic problems. The results of the classification step can also be used for multilingual image annotations as well as for DICOM (Digital Imaging and Communications in Medicine) header corrections [5].

Automatic image annotation uses a computer system which automatically assigns metadata in the form of captions or keywords to a digital image. Typically, image analysis first extracts feature vectors. Together with the training annotations, they are then used by a machine learning algorithm to learn to automatically assign annotations. The performance of the computer system largely depends on the availability of strongly representative visual features, able to characterize different visual properties of the images, and the use of effective algorithms for training classifiers for automatic image annotation.

A single image may contain different meanings organized in hierarchical semantics: hence, hierarchical multi-label classification (HMC) is strongly recommended for obtaining multi-label annotations. The task of multi-label classification is to assign multiple labels to each image. The assigned labels are a subset of a previously defined set or hierarchy of labels. HMC is used in various domains [6], such as text classification, scene and video classification, medical imaging and biological applications. One of the main issues involved in multi-label classification is the importance of detecting and incorporating the connections between the labels into the multi-labeling process. A second and related issue is the additional complexity involved in multi-label learning, as compared to single-label learning.

In this paper, we present a HMC system for medical image annotation. This system consists of the two standard parts of image annotation systems, i.e., processing (feature extraction) and classification of images. The image processing part uses state-of-the-art approaches to convert an image to a set of numerical features extracted directly from the pixel values. The image classification part, which labels and groups the images, contains the main novelty of our approach: The labels can be organized in a hierarchy and an image can be labeled with more than one label (an image can belong to more than one group).

First, we generate four different types of descriptors most suitable for X-Ray medical images: raw pixel representation (RPR) [7], local binary patterns (LBP) [8], edge histogram descriptors (EHD) [9], and scale-invariant feature transform (SIFT) [10]. The features are generated using

the medical X-ray images from the ImageCLEF2009 medical image annotations task [5]. Next, we use these features together with the annotations to train the classifiers. In particular, we use ensembles (bags and random forests) of PCTs for HMC and SVMs for single-label classification, the most widely used classifier in the area of image annotation. At the end, we assess the predictive performance of the classifiers using the hierarchical error measure (HEM) from [5] and overall recognition rate (RR), commonly used for assessing the predictive performance over the database we use.

The main question that we address in our research is whether exploiting the semantic knowledge about the inter-class relationships among the image labels organized in hierarchical structure can improve the predictive performance of a system for automatic image annotation. To this end, we compare the predictive performance of the ensembles of PCTs for HMC (that predict all labels simultaneously) to that of the SVMs (each of them predicting a single label). We do this across all feature extraction techniques, thus evaluating the different feature extraction techniques and their use in HMC of medical X-ray images. Moreover, we investigate whether (and which type of) combination of these techniques yields better predictive performance. We consider low level (LL) and high level (HL) feature fusion/combination schemes [7].

To emphasize the generality of our approach we have also tested it on the image database from the ImageCLEF@ICPR 2010 photo annotation task [11]. The images in this database are annotated with 53 visual concepts organized in a classification scheme with hierarchical structure which we used to build ensembles of PCTs for HMC as classifiers. The 53 concepts include abstract categories (like partylife), the time of day (like day or night), persons (like no person visible, small or big group) and quality (like blurred or underexposed). For a complete overview of the task see [11].

The remainder of the paper is organized as follows. Section 3 introduces predictive clustering trees and their use for HMC. Section 4 describes the techniques for feature extraction from images. In Section 5, we explain the experimental setup. The obtained results and a discussion thereof are given in Section 6. Section 8 concludes the paper and points out some directions for further work.

2. Related work

Regardless of the number of visual concepts that have to be learned and their mutual connections, most of the present systems for annotation of general images and medical images in particular learn a separate model for each visual concept (label), i.e., they treat the classes as completely separate and independent both visually and semantically.

The multi-label classification problems are usually transformed into several binary classification problems. For example, the methods that have high predictive performance on the current challenges/competitions detection and annotation tasks as PASCAL Visual Object Classes challenge [12], ImageCLEF medical image annotation task [13], [5] and ImageCLEF visual concept detection and annotation task [14] perform multi-label classification by building binary classifiers for each label. The instances associated with particular label are in one class and the rest are in another class. For solving the binary classification problems its common to use a SVMs with a χ^2 kernel [15]. This means that the increase of the number of labels used for annotation will increase the complexity of such an approach.

To deal with a large number of categories/classes, many approaches combine binary classifiers using class hierarchies [16], [17]. This usually results in logarithmic complexities. The class

hierarchies can be automatically constructed through analysis of visual similarities in top-down manner by recursive partitioning of the set of classes [18] or the hierarchy can be built bottom-up by agglomerative clustering [19]. The hierarchies could also be found by exhaustive search or random sampling followed by cross-validation [20].

Alternative method to automatic construction of hierarchies is to query an external semantic network with class labels [17]. Since semantic networks model concepts and relations between them, a subgraph in a form of hierarchy can be easily extracted. Such an approach allows to incorporate prior knowledge about object identity into the visual recognition system. Our approach for automatic image annotation is based on this idea, by exploiting the semantic knowledge about the inter-class relationships among the image labels organized in hierarchical structure we build one classifier that can simultaneously predict all of the annotations of an image instead of building binary classifiers for each node in the hierarchy.

Another popular approach for image annotation is TagProp [21]. TagProp is a discriminatively trained nearest neighbor model. Tags of test images are predicted using a weighted nearest-neighbor model to exploit labeled training images. Neighbor weights are based on neighbor rank or distance. TagProp allows the integration of metric learning by directly maximizing the log-likelihood of the tag predictions in the training set. However, in a recent study, Mensink et al.[22] showed that a per label trained linear SVM classifiers outperform TagProp.

3. Ensembles of PCTs for HMC

3.1. The task of HMC

Hierarchical multi-label classification is a variant of classification where (1) a single example may belong to multiple classes at the same time and (2) the possible classes are organized in a hierarchy. An example that belongs to some class c automatically belongs to all the super-classes of c : This is called the hierarchical constraint.

Problems of this kind can be found in many domains including text classification, functional genomics, and object / scene classification. For a more detailed overview of the possible application areas we refer the reader to [6].

In medical image classification, the application domain on which we focus, an important problem is the development of an automatic image annotation system, which can specify the image modality, body orientation, body region, or the biological system examined. In this domain, the predefined set of labels might be organized in a semantic hierarchy, such as the one shown in Fig. 1. Each image is represented with: (1) a set of descriptors (in this example, the descriptors are histograms of five types of edges encountered in the image) and (2) a set of labels/annotations. A single image can be annotated with multiple labels at different levels of the predefined hierarchy.

For example, the image in the second row of the table in Fig. 1 has two labels, middle abdomen and renal pelvis, listed explicitly. Note that this image is also implicitly labeled with the labels: anatomy, abdomen, kidney, uropoietic and bio-system. These labels are all the ancestors of the explicitly listed labels in the given hierarchy.

The data, as presented in the table in the left-hand side of Fig. 1, constitute a data set for HMC. This set can be used by the machine learning algorithm to train a classifier for HMC. The testing set of images contains only the set of descriptors and has no *a priori* annotations.

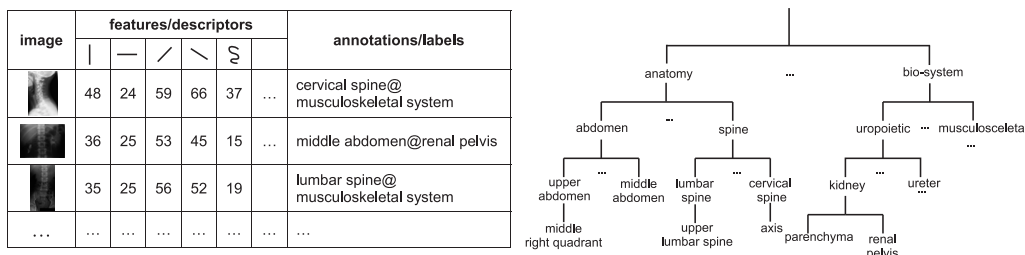


Figure 1: An example task of HMC in medical domain. The table (on the left-hand side) contains set of images with their visual descriptors and annotations. The annotations are part of the IRMA [23] hierarchical classification scheme (of which a small part is shown on right hand side).

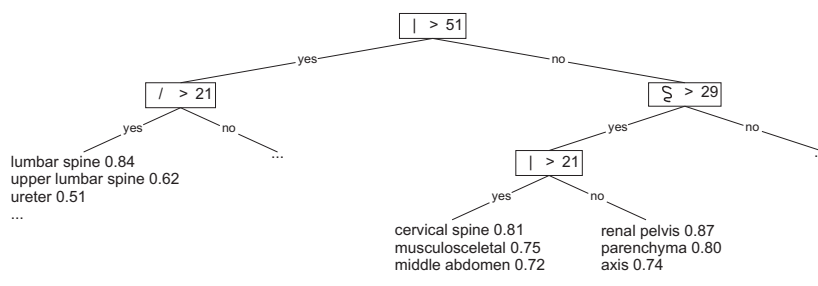


Figure 2: An example of a predictive clustering tree constructed using the descriptors from Fig. 1. The internal nodes contain tests on the descriptors, while the leafs store the probabilities that an image is annotated with a given label.

3.2. Predictive clustering trees

Predictive Clustering Trees (PCTs) [24] ¹ generalize decision trees [25] and can be used for a variety of learning tasks including different types of prediction and clustering. The PCT framework views a decision tree as a hierarchy of clusters: the top-node of a PCT corresponds to one cluster containing all data, which is recursively partitioned into smaller clusters while moving down the tree. The leaves represent the clusters at the lowest level of the hierarchy and each leaf is labeled with its cluster's prototype (prediction). Note that the hierarchical structure of the PCT (Fig. 2) does not necessary reflect the hierarchical structure of the annotations (Fig. 1).

PCTs are built with a greedy recursive top-down induction (TDI) algorithm, similar to that of C4.5 [26] or CART [25]. The learning algorithm starts by selecting a test for the root node. Based on this test, the training set is partitioned into subsets according to the test outcome. This is recursively repeated to construct the subtrees. The partitioning process stops when a stopping criterion is satisfied (e.g., the number of records in the induced subsets is smaller than some predefined value; the length of the path from the root to the current subset exceeds some predefined value etc.). In that case, the prototype is calculated and stored in a leaf.

One of the most important steps in the TDI algorithm is the test selection procedure. For each node, a test is selected by using a heuristic function computed on the training examples. The goal

¹The PCT framework is implemented in the CLUS system, which is available at <http://www.cs.kuleuven.be/~dtai/clus>.

of the heuristic is to guide the algorithm towards small trees with good predictive performance. The heuristic used in this algorithm for selecting the attribute tests in the internal nodes is the reduction in variance caused by partitioning the instances, while the variance $Var(S)$ is defined by (1). Maximizing the variance reduction maximizes cluster homogeneity and improves predictive performance.

The main difference between the algorithm for learning PCTs and a TDI algorithm (for example, see the algorithms proposed in [26], [25]) is that the former considers the variance function and the prototype function, that computes a label for each leaf, as parameters that can be instantiated for a given learning task. So far, the PCTs have been instantiated for the following tasks: multiple targets prediction [27], [28], prediction of time-series [29] and hierarchical-multi label classification [30]. In this article, we focus on the last task.

3.3. PCTs for hierarchical-multi label classification

To apply PCTs to the task of HMC, the example labels are represented as vectors with Boolean components. Components in the vector correspond to labels in the hierarchy traversed in a depth-first manner. The i -th component of the vector is 1 if the example belongs to class c_i and 0 otherwise. If $v_i = 1$, then $v_j = 1$ for all v_j 's on the path from the root to v_i .

The variance of a set of examples (S) is defined as the average squared distance between each example's label v_i and the mean label \bar{v} of the set, i.e.,

$$Var(S) = \frac{\sum_i d(v_i, \bar{v})^2}{|S|} \quad (1)$$

The higher levels of the hierarchy are more important: an error at the upper levels costs more than an error at the lower levels. Considering this, a weighted Euclidean distance is used:

$$d(v_1, v_2) = \sqrt{\sum_i w(c_i)(v_{1,i} - v_{2,i})^2} \quad (2)$$

where $v_{k,i}$ is the i 'th component of the class vector v_k of an instance x_k , and the class weights $w(c_i)$. The class weights decrease with the depth of the class in the hierarchy, $w(c_i) = w_0 \cdot w(c_j)$, where c_j is the parent of c_i . Each leaf in the tree stores the mean \bar{v} of the vectors of the examples that are sorted in that leaf (Fig. 2). Each component of \bar{v} is the proportion of examples \bar{v}_i in the leaf that belong to class c_i . An example arriving in the leaf can be predicted to belong to class c_i if \bar{v}_i is above some threshold t_i . The threshold can be chosen by a domain expert.

The PCTs are also extended for predicting hierarchies organized as directed acyclic graphs (DAGs). In this case, then the depth of a class is not unique: classes do not have single path from the hierarchy's root. To resolve this issue, Vens et al. [30] suggest four aggregation schemes of the possible paths from the top-node to a given class: average, maximum, minimum and sum. After an extensive experimental evaluation, they recommend to use the average as aggregation function. For a detailed description of PCTs for HMC the reader can refer to [30]. Next, we explain how PCTs are used in the context of an ensemble classifier, in order to further improve the performance of PCTs.

3.4. Ensemble methods

An ensemble classifier is a set of (base) classifiers. A new example is classified by the ensemble by combining the predictions of the member classifiers. The predictions can be combined

by taking the average (for regression tasks), the majority vote (for classification tasks) [31],[32], or more complex combinations.

We use PCTs for HMC as base classifiers. Average is applied to combine the predictions of the different trees: the leaf's prototype is the proportion of examples of different classes that belong to it. Just like for the base classifiers, a threshold should be specified to make a prediction.

We consider two ensemble learning techniques that have primarily been used in the context of decision trees: bagging and random forests. Bagging [31] constructs the different classifiers by making bootstrap replicates of the training set and using each of these replicates to construct one classifier. Each bootstrap sample is obtained by randomly sampling training instances, with replacement, from the original training set, until a number of instances is obtained equal to the size of the training set. Bagging is applicable to any type of learning algorithm.

A random forest [32] is an ensemble of trees, obtained both by bootstrap sampling, and by randomly changing the feature set during learning. More precisely, at each node in the decision tree, a random subset of the input attributes is taken, and the best feature is selected from this subset (instead of the set of all attributes). The number of attributes that are retained is given by a function f of the total number of input attributes x (e.g., $f(x) = x$, $f(x) = \sqrt{x}$, $f(x) = \lfloor \log_2 x \rfloor + 1$, ...). By setting $f(x) = x$, we obtain the bagging procedure.

4. Feature extraction from images

Collections of medical images can contain various images obtained using different imaging techniques. Different feature extraction techniques are able to capture different aspects of an image (e.g., texture, shapes, color distribution...). In this study, we focus on X-ray images, hence, we use texture (LBP and EHD) and local (SIFT) features as most promising for describing X-ray images [5],[33].

Texture is especially important, because it is difficult to classify medical images using shape or gray level information. Effective representation of texture is needed to distinguish between images with equal modality and layout. Local image characteristics are fundamental for image interpretation: while global features retain information on the whole image, the local features capture the details. They are thus more discriminative concerning the problem of inter and intra-class variability, an open challenge in automatic annotation of medical images [7].

4.1. Raw pixel representation

The most straightforward approach to image classification is the direct use of the image pixel values as features. The images are scaled to a common size and represented by a feature vector that contains image pixel values. It has been shown that for classification and retrieval of medical radiographs, this method serves as a reasonable baseline [34]. We used a 32x32 down-sampled representation of the images as recommended in [7]. The obtained 1024 pixel values were then used as input features. Fig. 3 shows how we built the raw pixel representation for each image.

4.2. Local binary patterns

Local binary patterns (LBP) are one of the best representations of texture content in images [8]. They are invariant to monotonic changes in gray-scale images and fast to compute. Furthermore, they are able to detect different micro patterns including edges, points, constant areas etc.

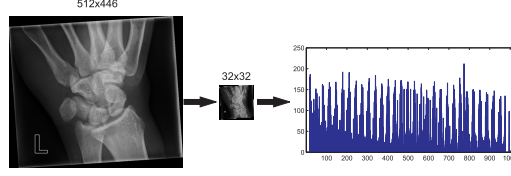


Figure 3: Down-sampling for raw pixel representation

The basic idea behind the LBP approach is to use the information about the texture from a local neighborhood. First, we define the radius R of the local neighborhood under consideration. The LBP operator then builds a binary code that describes the local texture pattern in the neighborhood set of P pixels. The binary code is obtained by applying the gray value of the neighborhood center as a threshold. The binary code is then converted to a decimal number which represents the LBP code. Formally, given a pixel at position (x_c, y_c) the resulting LBP code can be expressed as follows:

$$LPB_{(P,R)}(x_c, y_c) = \sum_{n=0}^{P-1} S(i_n - i_c) 2^n \quad (3)$$

where n ranges over the P neighbors of the central pixel (x_c, y_c) , i_c and i_n are the gray-level values of the central pixel and the neighbor pixel, and $S(x)$ is defined as:

$$S(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (4a)$$

$$(4b)$$

The image is traversed with the LBP operator pixel by pixel and the outputs are accumulated into a discrete histogram. However, not all LBP codes are informative. Certain LBP codes capture fundamental properties of the texture and are called uniform patterns because they constitute the vast majority, sometimes over 90 percent, of all patterns present in the observed textures [8]. These patterns have one thing in common, namely, a uniform circular structure that contains very few spatial transitions. They function as templates for micro-structures such as bright spot, flat area or dark spot.

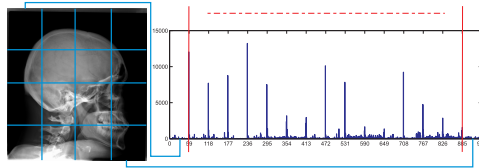


Figure 4: The image is divided into 4x4 non-overlapping sub-images from which LBP histograms are extracted and concatenated into a single, spatially enhanced histogram

In our experiments, we used the patterns $LPB_{8,1}^{u2}$, where the superscript $u2$ reflects the use of uniform patterns that have a U value of at most 2 on a neighborhood of size 8 and radius 1. The U value is the number of spatial transitions (bitwise 0/1 changes) in the pattern. The non-uniform patterns (patterns that have U value larger than 2) are grouped under one bin in the

resulting histogram. With the $LBP_{8,1}^{u2}$ operator, the number of bins in the histogram is reduced from 256 to 59 (58 bins for uniform patterns and one bin for non-uniform/noisy patterns).

To spatially enhance the descriptors and improve the performance, it has been suggested to repeatedly sample predefined sub-regions of an image (e.g. 1x1, 2x2, 4x4, 1x3 etc.) [35]. The different resolutions are then aggregated into a spatial pyramid which allows for region-specific weighting. Following these approaches, we divide the images into 4x4 non-overlapping sub-images (blocks) and concatenate the LBP histograms extracted for each sub-image into a single, spatially enhanced feature histogram. This approach aims at obtaining a more local description of the images. Fig. 4 shows how we build the LBP histogram with 944 bins in total for each image (16 blocks with 59 bins each).

4.3. Edge histogram descriptors

Edge detection is a fundamental problem of computer vision and has been widely investigated [36]. The goal of edge detection is to mark the points in a digital image at which the luminous intensity changes sharply. An edge representation of an image drastically reduces the amount of data to be processed, yet it retains important information about the shapes of objects in the scene. Edges in images constitute important features to represent their content.

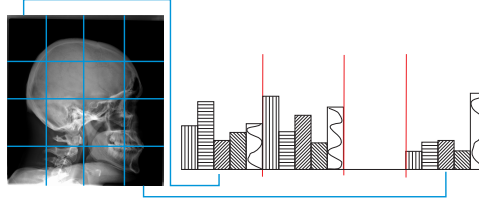


Figure 5: The image is divided into 4x4 non-overlapping sub-images. For each sub-image five types of edge bins are calculated and concatenated into a single, spatially enhanced histogram

The edge histogram in the image space represents the frequency and the directionality of the brightness changes in the image. To represent it, the MPEG-7 standard defines the edge histogram descriptor (EHD) [9]. The edge histogram descriptor basically represents the distribution of five types of edges in each local area/sub-image. The image space is divided into 4x4 non-overlapping blocks, yielding 16 equal-sized sub-images (Fig. 5).

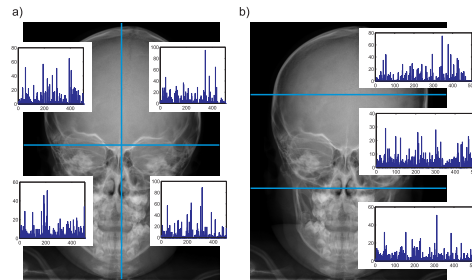


Figure 6: Two different spatial pyramids used in our experiments, a) 2x2, b) 1x3. The spatial pyramid constructs feature vectors for each of the specific parts of the image

To characterize the sub-images, a histogram of edge distribution for each sub-image is generated. Edges in the sub-images are categorized into five types: vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edges, as presented in Fig. 5. The histogram for each sub-image represents the relative frequency of occurrence of the five types of edges in the corresponding sub-image and thus contains five bins.

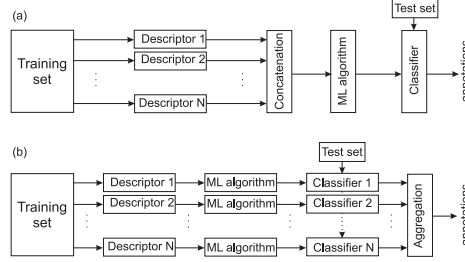


Figure 7: Fusion schemes for the different descriptors. a) Low level fusion, b) High level fusion.

Since there are 16 sub-images in the image and 5 types of edges, a total of 80 histogram bins are required. Note that each of the 80-histogram bins has its own semantics in terms of location and edge type. In our experiments, the edge detection is performed using the Canny edge detection algorithm [37].

4.4. SIFT descriptors

We employ the bag of features approach commonly used in many state of the art approaches in image classification [38]. The basic idea of this approach is to sample a set of local image patches using some method (densely, randomly or using a key-point detector) and calculate a visual descriptor on each patch (SIFT descriptor, normalized pixel values). The resulting distribution of descriptors is then quantified against a pre-specified visual codebook which converts it to a histogram. The main issues that need to be considered when applying this approach are: sampling of the patches, selection of the visual patch descriptor and building the visual codebook.

We use dense sampling of the patches, which samples an image grid in a uniform fashion using a fixed pixel interval between patches. We use an interval distance of 6 pixels and sample at multiple scales ($\sigma = 1.2$ and $\sigma = 2.0$). Due to the low contrast of the radiographs, it would be difficult to use any detector for points of interest. Also, it has been pointed in [38], that a dense sampling is always superior to any strategy based on detectors for points of interest. We calculate a SIFT descriptor [10] for each image patch.

The crucial aspects of a codebook representation are the codebook construction and assignment. An extensive comparison of the codebook representation variables is presented in [39]. We employ k -means clustering in the R environment for statistical computing and graphics [40] on 400000 randomly chosen descriptors from the set of images available for training. k -means partitions the visual feature space by minimizing the variance between a predefined number of k clusters. Here, we set k to 500, thus we define a codebook with 500 codewords [7].

The dense sampling gives an equal weight to all key-points, irrespective of their spatial location in the image. To overcome this limitation we follow the spatial pyramid approach which we applied for the LBP descriptor. For this descriptor, we used a spatial pyramid of 1x1, 2x2, and 1x3 regions. Since every region is an image in itself, the spatial pyramid can easily be used in combination with dense sampling. The resulting vector with 4000 bins (8x500) was obtained by

concatenation of the eight histograms. Fig. 6 shows an example of the extracted histograms for spatial pyramids of 2x2 and 1x3.

4.5. Feature fusion schemes

Different visual features bringing different information about the visual content of the images clearly outperform single feature approaches as can be seen in [5], [7]. Following these findings, in our research we investigate two different feature fusion schemes: low level (LL) and high level (HL). These fusion schemes are depicted in Fig. 7.

For the low level feature fusion scheme, the descriptors are concatenated in a single feature vector and a classifier is trained on the joint feature vector. The high level fusion scheme averages the predictions from the individual classifiers trained on the separate descriptors.

5. Experimental setup

In this section, we present the experimental setup we used to validate and compare the proposed system. First, we present the databases of images that we use. Next, we describe the evaluation metrics we use to assess the predictive performance of the classifiers. Then, we state the experimental questions that we investigate in this study. At the end, we show the parameter instantiation for the algorithms and the design of the experiments.

5.1. The IRMA database

We evaluated our system by applying it to the database for the ImageCLEF2009 medical image annotations task [5]. This database is provided by the IRMA group from the University Hospital of Aachen, Germany [23]. The database contains 12677 fully annotated radiographs, taken randomly from medical routine, which should be used to train a classifier. The dataset contains two parts: ImageCLEF2007 (12339 training and 1353 testing images) and ImageCLEF2008 (12667 training and 1733 testing images). These datasets present a difficult classification problem. First, the classes in the training set are extremely imbalanced (e.g. there are classes with less than 10 images and classes with more than 2000 images). Second, the distribution of the classes in the training set is different from the one on the testing set.

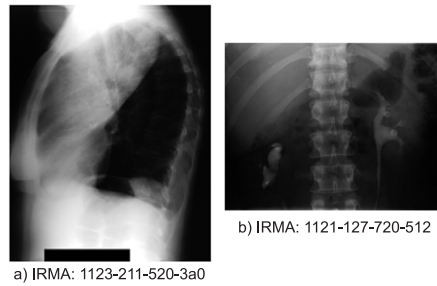


Figure 8: IRMA-coded chest and abdomen radiograph. For instance, the code for the biological axis (512) on the sub-figure b) is translated as follows: 5 is for uropoietic system, 51 is for uropoietic system, kidney and 512 is uropoietic system, kidney, renal pelvis. The renal pelvis is an element of the kidney, which in turn is an element of the uropoietic system

The images are labeled according to the four annotation label sets [5]. We used the ImageCLEF2007 label set with 116 IRMA codes and the ImageCLEF2008 label set with 193 IRMA

codes because of the hierarchical nature of the coding scheme [23]. The goal is to correctly annotate 1353 (for 2007) and 1733 (for 2008) images that are provided without labels, using the different respective annotation label sets in turn.

The IRMA coding scheme consists of four axes with three to four positions, each position taking a value from the set 0,..., 9, a,..., z, where '0' denotes 'unspecified' and determines the end of a path along an axis. The four axes are: technical axis (T, image modality), directional axis (D, body orientation), anatomical axis (A, body region examined) and biological axis (B, biological system examined). This allows a short and unambiguous notation (IRMA: TTTT-DDD-AAA-BBB), where T, D, A, and B denotes a coding or sub-coding digit of the respective axis. A small part of the IRMA coding hierarchy is presented in Fig. 1. Fig. 8 gives two examples of unambiguous image classification using the IRMA code.

The IRMA code is hierarchical in its nature and it allows us to exploit the hierarchy of the code. This means that we can construct an automatic image annotation system based on predictive clustering trees for HMC.

5.2. Evaluation metrics

In this study, we use two evaluation metrics: the ImageCLEF hierarchical evaluation measure and overall recognition rate. The ImageCLEF hierarchical evaluation measure is proposed in [5]. This measure takes into account the depth and the difficulty of the predictive problem ('branching factor') at which an error has occurred (Equation 5). It can be calculated using the following formula:

$$\sum_{i=1}^I \frac{1}{b_i} \frac{1}{i} \delta(v_i, \bar{v}_i), \quad (5)$$

$$\delta(v_i, \bar{v}_i) = \begin{cases} 0, & \text{if } v_j = \bar{v}_j \forall j \leq i \\ 0.5, & \text{if } v_j = * \exists j \leq i \\ 1, & \text{if } v_j \neq \bar{v}_j \exists j \leq i \end{cases} \quad (6a)$$

$$(6b)$$

$$(6c)$$

where I is the depth of the hierarchy, b_i is the number of possible labels at the error ('branching factor') and i is the depth at which the error occurred. This measure allows the classifier not to predict the complete code/annotation, that is, the classifier can predict the first 2 nodes of the code (level of the hierarchy) and then to say 'don't know' (encoded by *) for the next node/level. The ImageCLEF evaluation measure can range from 0 to the number of testing images. If this measure is closer to 0, then the classifier is more accurate.

The overall recognition rate is a very common and widely used evaluation measure. It is the fraction of the test images whose complete IRMA code was predicted correctly.

5.3. Experimental questions

The goal of this study is to answer the following questions:

1. Does the use of the hierarchy (in ensembles of PCTs) improve the predictive performance over flat classification (SVMs)?
2. How is the relative performance of the two techniques affected by the:
 - (a) Usage ensembles of PCTs over a single PCT in the domain of image annotation?
 - (b) Different ensemble methods: bagging or random forest?
 - (c) Various feature extraction techniques for medical X-Ray images?

- (d) Schemes for fusion of the descriptors from the feature extraction techniques?
3. Is the proposed system with ensembles of PCTs scalable and efficient?

For the first three questions (1, 2a and 2b), we evaluate the performance of PCTs for HMC and ensembles (bagging and random forest) of PCTs. After that, we compare the best method for HMC with SVMs. It has been shown [30] that exploiting the structure of the hierarchy in tree classifiers yields better predictive performance in the domain of functional genomics. Here, we compare the performance of the ensemble classifiers with SVMs for flat classification - the most widely used classifiers for medical image annotation [7].

To check which feature extraction technique is most suitable for medical X-Ray images (question 2c), we compare the performance of the classifiers on each type of visual descriptors. For this purpose, we discuss only the results from the separate runs of the descriptors (first four rows from Table 1 and Table 2).

The various feature extraction techniques capture different aspects of an image. Here, we investigate whether combination of feature extraction techniques can increase the predictive performance (question 2d). The results from the fusion schemes are presented in the last 10 rows in Table 1 and Table 2.

We compare the execution times of the different classifiers to assess the efficiency and scalability of the system (question 3). We measure the time needed to train the classifiers; for SVMs this includes also the time needed to optimize the parameters.

5.4. Experimental design

In this section, we describe the experimental setup that we used. First, we describe an adaptation of the hierarchy of the IRMA code and then the parameter instantiations of the learning algorithms. Note that we stated the parameters for the feature extraction techniques while explaining them (see Section 4).

The IRMA coding scheme was proposed by Lehmann et al. in [23] and it consists of four axes which are strictly hierarchical (tree-shaped hierarchies). The literature (for example, see [5],[23]) suggests that these four axes are independent. We conducted series of experiments predicting the four axes simultaneously (combined in a single hierarchy) and axis-wise prediction. The predictive performance when using all four axes simultaneously was higher as compared using each axis separately. This leads us to believe that these axes are not-independent. In a separate study, Tommasi et al. [7] came to a similar conclusion. To address this issue, we adapted the IRMA coding hierarchy as follows.

We take the code of the first position for the biological axis and add it in front of the codes for the anatomical and directional axes. The inclusion of the biological code in the first level in the hierarchy helps us to initially filter the images resulting in large visual differences in the first level of the hierarchy. In the context of the axis A, the first level of axis B is necessary because the examined body region insufficiently describes the content and structure of the images. For example, fluoroscopy of the abdominal region may access the vascular or the gastrointestinal system depending on the way the contrast agent is administered, which results in different image textures. For the directional axis, this is even more obvious. For instance, an image of a chest and an image of a hand can have the same directional code, but are visually very different.

The adaptation of the hierarchy of the IRMA code was done to increase the inter-class variability and decrease the intra-class variability of the images. Fig. 9 shows the adapted hierarchy of the classes that we use. Note that this hierarchy was only used to train the classifier. The evaluation was performed by using the original IRMA hierarchy.

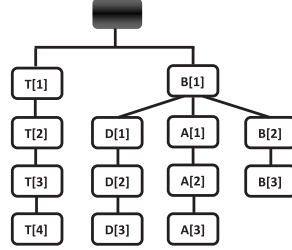


Figure 9: The adapted hierarchy of the classes in the IRMA code

In the following, we state the parameter instantiations that we used to train the classifiers: PCTs, ensembles and SVMs. The algorithm for learning PCTs requires as input the weight of the depth in the hierarchy. We set w_0 to 0.75 to force the algorithm to make better predictions on the upper levels of the hierarchy. Also, we performed F-test pruning to prevent over-fitting of the trees [30].

We trained ensembles of 100 un-pruned trees (PCTs). For the base PCTs, we used the same weight (0.75) as when training the single PCTs. The size of the feature subset that is retained at each node, when training a random forest, was set to 10% of the number of descriptive attributes. Remember that the output of the classifier is a probability that a given example is annotated with a given label. If the probability is higher than a given threshold (obtained during the training of the classifier), then the example is annotated with the given label. Since the hierarchical evaluation measure allows the classifier to predict a portion of the code, different thresholds for the different levels of the hierarchy were selected. If a probability for a given code was lower than the threshold then for this code and its sub-codes the classifier predicts ‘don’t know’.

For training the SVMs, we used a custom developed application. This application uses the LIBSVM library [41]. We apply the *One-against-All* (OvA) approach to solve the partial binary classification problems. Each of the SVMs was trained with a χ^2 kernel. We optimize the cost parameter C of the SVMs using an automated parameter search procedure. For the parameter optimization, we separate 20% of the training set and use it as validation set. After finding the optimal C value, the SVM was finally trained on the whole set of training images.

For the evaluation of the SVMs using the hierarchical error measure, we applied the confidence based opinion fusion from [7]. Let us assume that there are N classes. Then, using the OvA approach, N SVMs are trained – each separating a single class from all remaining ones. The decision is based on the distances of the test sample to the N hyperplanes. The prediction then corresponds to the hyperplane for which the distance is largest. The confidence based opinion fusion, however, takes into account the difference of the predictions with the two largest distances reported from the SVMs classifiers. This difference is computed only if their distances differ less than a threshold value (obtained during training using the validation data set). In that case, the final prediction will contain ‘don’t know’ starting from the position where the two underlying predictions begin to differ. For example, if the two predictions for the anatomical axis are 411 and 421 then the final prediction will be 4**. This approach improves the hierarchical error measure for the SVMs classifier by 10 to 20 points depending on the used descriptors.

Table 1: Predictive performance of the feature extraction algorithms and their combinations (best results are shown in boldface) using the ImageCLEF hierarchical evaluation measure (better performance is for smaller values of HEM)

Table 2: Predictive performance of the feature extraction algorithms and their combinations (best results are shown in boldface) using recognition rate (better performance is for larger values of RR)

6. Results and discussion

Table 1 and Table 2 present the results obtained using the experimental setup described in Section 5. First, we compare the performance of single PCTs and ensembles of PCTs. Then, we compare the performance of the best ensemble method (random forests) and SVMs. In the discussion of the results, we focus on the HEM (Table 1), since the two evaluation measures show similar behavior; the conclusions for HEM are valid also for RR.

The results clearly show that ensemble methods outperform single PCTs on all of the datasets: random forests are significantly better (according to the non-parametric Wilcoxon test for statistical significance) than single PCTs ($p < 4 \cdot 10^{-6}$) and bagging is better than single PCTs ($p < 4 \cdot 10^{-6}$). A comparison between the two ensemble methods shows that random forest outperforms bagging and that the difference is statistically significant ($p < 1 \cdot 10^{-4}$).

We next compare the performance of random forests to the performance of the SVMs. On all datasets the random forests perform better than the SVMs; the difference on average is ~ 17 points for the ImageCLEF2007 and ~ 20 points for ImageCLEF2008 datasets (note that a point in the HEM roughly corresponds to one completely misclassified image). The difference in the performance is statistically significant (with $p < 4 \cdot 10^{-6}$). This shows that exploiting the structure of the hierarchy does help in improving the predictive performance.

Next, we analyze the results for the individual feature extraction algorithms (top 4 rows from Table 1 and Table 2). We can note the high predictive performance of the SIFT histogram: it is most capable of capturing the hierarchical structure of the X-ray images. The other feature extraction algorithms follow after a gap of ~ 37 points for random forests and 50 points for SVMs. They are ordered by performance as follows: LBP, then EHD and the simplest descriptor RPR has the worst performance. The difference of the performance to the LBP operator (random forests and SVMs) is very noticeable: on the ImageCLEF2007 dataset, random forests are better by ~ 30 points and on ImageCLEF2008 by ~ 50 points. The LBP descriptors are able to capture information that is more easily utilized by the random forests than by the SVMs.

We also compare the results of the experiments conducted with different feature fusion schemes. Inclusion of more than one type of features in the classification process contributes to better representation of the hierarchical nature of the images and helps to further improve the predictive performance. The low level fusion scheme (concatenation) yields better predictive performance than the high level fusion. This is valid for all algorithms used in this study.

The best results are achieved by using random forests on the concatenated SIFT, LBP and EHD descriptors (boldface in Table 1 and Table 2). This holds for both datasets, ImageCLEF2007 and ImageCLEF2008. Moreover, our best results are better than the best results reported so far on this database (see [5]). Our score of 153.2 for ImageCLEF2008 is by 16.3 points better than the best result, and the score of 51.9 for ImageCLEF2007 is by 12.4 points better than the best result.

From the results, we can also notice the worse performance of all algorithms on the ImageCLEF2008 dataset, as compared to the ImageCLEF2007 dataset. This is mainly due to the

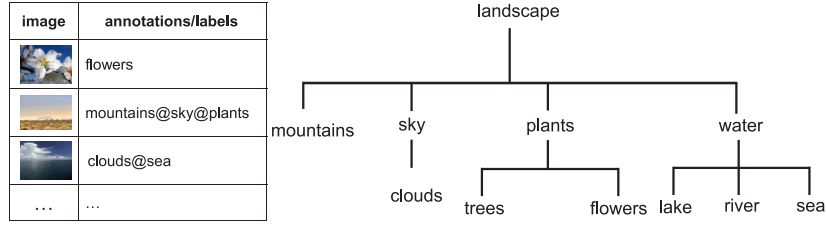


Figure 10: A fragment of the hierarchy for image annotation. The annotations are part of the hierarchical classification scheme for the ICPR 2010 photo annotation task (right). The table contains set of images with their annotations (left).

larger hierarchy of the ImageCLEF2008 dataset (195 nodes as compared to 140 nodes for ImageCLEF2007 dataset). In addition, the difference of the distribution of images in the training and the testing set is bigger for the ImageCLEF2008 dataset than for the ImageCLEF2007 dataset.

Additionally, we assess the efficiency of the algorithms by measuring the time needed to learn the classifier. The random forests are the fastest method; they are ~ 10 times faster than bagging and ~ 4 times than the SVMs (including the optimization of the SVM parameters). Recall that the random forests are ensembles of PCTs that predict the complete hierarchy (a single model), while the SVMs construct a classifier for each node of the hierarchy separately. Hence, the increase of the hierarchy will significantly increase the training time of SVMs (additional classifiers should be trained), while the training time for random forests will increase only slightly.

7. Experiments on photo annotation

In order to show the generality of the proposed system, we perform experiments on annotation of general images. In this section, we present first the experimental setup that we used (the data, evaluation metrics and the experimental design). Afterwards, we give the results and compare them to the state-of-the-art approaches used in image annotation.

7.1. Experimental setup

This set of experiments was performed using the database from ImageCLEF@ICPR photo annotation task [42]. The database consists of 8000 train images and 10000 test images annotated with 53 visual concepts organized in a small hierarchy with tree structure (see Fig. 10 for an example). The average number of annotations per image is 8.68. The visual concepts also contain abstract categories like Family/Friends, Partylife, Quality (blurred, underexposed, ...) and etc., thus making the annotation/classification task very challenging.

The most widely used evaluation measure in the area of ‘general photo annotation’/‘visual concept detection’ is the mean average precision (MAP). Thus, we use MAP to evaluate the performance of the algorithms on this database². For a given target visual concept, the average precision can be calculated as the area under the precision-recall curve for that target. Hence, it combines both precision and recall into a single performance value. The average precision is calculated for each visual concept separately and the obtained values are then averaged to

²Why not also overall recognition rate? Can we calculate also ORR for this case???

obtain the mean average precision. Because the true labels of the test images from the Image-CLEF@ICPR 2010 database are not publicly available we report the MAP value obtain using 3-fold cross validation on the train set³.

For the images from this database, we use SIFT features, since in the previous experiments they were best performing features (also SIFT features are typically used in this type of problem [14]). The SIFT features for this set of experiments were constructed using visual codebook with 4000 instead of 500 words (see section 4.4). This modification was made because most of the state-of-the-art approaches for image classification of general photos use visual codebook with 4000 words [14], [12]. From the previous experiments, random forests were the best performing method, so again we train random forests with 100 un-pruned PCTs for HMC. For the base PCTs, we used the same weight (0.75) and the size of the feature subset that is retained at each node was set to 10% of the number of descriptive attributes same as in the experiments from the section 5.

Here, to train the SVMs we use the LibSVM implementation with probabilistic output [43]. To solve the multiple classification problems we employ again the *One-against-All* approach. For each visual concept, we build a binary classifier so instances associated with that visual concept are in one class (positive) and the rest are in another class (negative). To handle the imbalance in the number of positive versus negative training examples, we fix the weights of the positive and negative class. The weight of the positive class is set to $\frac{\#pos+\#neg}{\#pos}$ and the weight of the negative class is set to $\frac{\#pos+\#neg}{\#neg}$, with $\#pos$ the number of positive instances in the train set and $\#neg$ the number of negative instances⁴. Similarly as in the previous experiments, we optimize the value of the cost parameter C of the SVMs.

7.2. Results and discussion

The results from the photo annotation experiments are shown in Table 3. The table also contains the total training time and testing time per image for both SVMs and random forests of PCTs for HMC. From the presented results we can note that the random forests of PCTs for HMC outperform the SVMs both in terms of predictive performance and efficiency. The latter holds especially for the time needed to produce an annotation for a given test image: our approach is more than 500 times faster than the SVMs.

Table 3: Results of the photo annotation experiments evaluated using Mean Average Precision (better performance is for larger values of MAP).

	MAP	Train time (sec)	Test time per image (sec)
RF	0.452	9113.516	0.002
SVMs	0.429	11821.227	1.078

Following the results from the study performed by Mensink et al. [22], this means that our system also outperforms the TagProp [21] approach for image annotation. The results show that our system offers better predictive performance and efficiency than systems that are most widely used for annotation of images. All in all, the proposed system is general and easily applicable to other domains.

³Why 3-fold and not 10-fold??? We will have to defend this somehow???

⁴A reference for this??? Where is this recommended???

8. Conclusions

Hierarchical multi-label classification (HMC) problems are encountered increasingly often in image annotation. However, flat classification machine learning approaches are predominantly applied in this area. In this paper, we propose to exploit the annotation hierarchy in image annotation by using ensembles of trees for HMC. Applied on a benchmark task of hierarchical annotation of medical (X-ray) images, our approach achieves the best results so far. The sources of this high performance are explored in detail and include the exploitation of the hierarchy, the use of ensemble methods, the use of state-of-the-art local features, such as SIFT, and feature fusion.

Our approach to HMC exploits the annotation hierarchy by building a single classifier that simultaneously predicts all of the labels in the hierarchy. Further improvement is achieved by building ensembles of HMC trees, such as random forests. Compared to the best-performing approach so far (a collection of SVMs, each predicting one label at the lowest level of the hierarchy), our approach has superior performance, both in terms of accuracy/error and especially in terms of efficiency.

Ensembles of trees for HMC perform consistently better than SVMs over a range of image descriptors and combinations thereof. For efficient image representation, we consider a raw pixel representation, local binary patterns, edge histogram descriptors and SIFT histograms, as well as combinations thereof. The SIFT histograms are the best individual descriptors. Moreover, combination of descriptors lifts the predictive performance of the classifiers. We consider both low-level and high-level fusion of the image descriptors, the former yielding better performance.

Also, our approach performs better than the state-of-the-art in the area of photo annotation (i.e., visual concept detection). In terms of predictive performance, it outperforms SVMs trained with a χ^2 kernel and the TagProp system. Moreover, it outperforms the SVMs in terms of efficiency. Thus, we show that the proposed approach is easily applicable to other domains.

We expect it is possible to further improve the predictive performance of our system. We could try to adapt our tree-learning approach to tackle the shift in distribution of images between the training and the testing set. Better performance may also be obtained by including high level feature extraction algorithms able to give more understandable and compact representation of the visual content of the images (segmented objects with relations among them).

Let us conclude by emphasizing the scalability of our approach. Decision trees are one of the most efficient machine learning approaches and can handle large numbers of examples. The ensemble approach of random forests scales very well for large numbers of features. Finally, trees for HMC scale very well as the complexity of the annotation hierarchy increases, being able to handle very large hierarchies organized as trees or directed acyclic graphs. Combining these, our approach is scalable along all three dimensions.

Acknowledgment

The authors would like to thank Donato Malerba, Department of Computer Science, University of Bari for his valuable comments and suggestions.

References

- [1] R. Choplin, J. Boehme, C. Maynard, Picture archiving and communication systems: an overview, *Radiographics* 12 (1) (1992) 127–129.

- [2] S. Becker, R. Arenson, Costs and benefits of picture archiving and communication systems, *Journal of the American Medical Informatics Association* 1 (5) (1994) 361–371.
- [3] N. E. M. Association, Digital imaging and communications in medicine - dicom, <http://dicom.nema.org/> (2009). URL <http://dicom.nema.org/>
- [4] M. O. Guld, M. Kohnen, D. Keysers, H. Schubert, B. B. Wein, J. Bredno, T. M. Lehmann, Quality of dicom header information for image categorization, in: *SPIE*, Vol. 4685, 2001, pp. 280–287.
- [5] T. Tommasi, B. Caputo, P. Welter, M. O. Guld, T. M. Deserno, Overview of the clef 2009 medical image annotation track, in: *Working notes - CLEF 2009 Workshop*, 2009.
- [6] C. Silla, A. Freitas, A survey of hierarchical classification across different application domains, *Data Mining and Knowledge Discovery* doi:10.1007/s10618-010-0175-9.
- [7] T. Tommasi, F. Orabona, B. Caputo, Discriminative cue integration for medical image annotation, *Pattern Recognition Letters* 29 (15) (2008) 1996–2002.
- [8] T. Ojala, M. Pietikainen, T. Maenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24 (7) (2002) 971–987.
- [9] D. K. Park, Y. S. Jeon, C. S. Won, Efficient use of local edge histogram descriptor, in: *International Multimedia Conference, ACM workshops on Multimedia*, 2000, pp. 51–54.
- [10] D. G. Lowe, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision* 60 (2) (2004) 91–110.
- [11] S. Nowak, Imageclef@icpr contest: Challenges, methodologies and results of the photo annotation task, in: *International Conference on Pattern Recognition*, 2010, pp. 489–492.
- [12] M. Everingham, L. V. Gool, C. Williams, A. Zisserman, The PASCAL Visual Object Classes Challenge 2009 (VOC2009) Results (2009).
- [13] T. Deselaers, H. Muller, P. Clough, H. Ney, T. M. Lehmann, The clef 2005 automatic medical image annotation task, *International Journal of Computer Vision* 74 (1) (2005) 51–58.
- [14] S. Nowak, P. Dunker, Overview of the clef 2009 large-scale visual concept detection and annotation task, in: *Multilingual Information Access Evaluation II. Multimedia Experiments*, 10th Workshop of the Cross-Language Evaluation Forum, 2010, pp. 94–109.
- [15] K. E. A. van de Sande, T. Gevers, C. G. M. Snoek, Evaluating color descriptors for object and scene recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32 (9) (2010) 1582–1596.
- [16] M. Marszałek, C. Schmid, Constructing category hierarchies for visual recognition, in: *European Conference on Computer Vision*, Vol. IV of LNCS, Springer, 2008, pp. 479–491.
- [17] M. Marszałek, C. Schmid, Semantic hierarchies for visual object recognition, in: *Conference on Computer Vision & Pattern Recognition*, 2007.
- [18] L. Song, Y. Haoran, C. Liang-Tien, R. Deepu, Adaptive hierarchical multi-class svm classifier for texture-based image classification, in: *IEEE International Conference on Multimedia and Expo*, 2005.
- [19] L. Zhigang, S. Wenzhong, Q. Qianqing, L. Xiaowen, X. Donghui, Hierarchical support vector machines, in: *IEEE International Geoscience and Remote Sensing Symposium*, 2005.
- [20] X. Yuan, W. Lai, T. Mei, X.-S. Hua, X. qing Wu, S. Li, Automatic video genre categorization using hierarchical svm, in: *IEEE International Conference on Image Processing*, 2006, pp. 2905–2908.
- [21] M. Guillaumin, T. Mensink, J. Verbeek, C. Schmid, Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation, in: *International Conference on Computer Vision*, 2009, pp. 309–316.
- [22] T. Mensink, G. Csurka, F. Perronnin, J. Sanchez, J. J. Verbeek, Lear and xrcr’s participation to visual concept detection task - imageclef 2010, in: *CLEF (Notebook Papers/LABs/Workshops)*, 2010.
- [23] T. M. Lehmann, H. Schubert, D. Keysers, M. Kohnen, B. B. Wein, The irma code for unique classification of medical images, in: *SPIE*, Vol. 5033, 2003, pp. 440–451.
- [24] H. Blockeel, L. D. Raedt, J. Ramong, Top-down induction of clustering trees, in: *International Conference on Machine Learning*, Morgan Kaufmann, 1998, pp. 55–63.
- [25] L. Breiman, J. Friedman, R. Olshen, C. J. Stone, *Classification and Regression Trees*, Chapman & Hall/CRC, 1984.
- [26] R. J. Quinlan, *C4.5: Programs for Machine Learning*, 1st Edition, Morgan Kaufmann, 1993.
- [27] D. Kocov, C. Vens, J. Struyf, S. Džeroski, Ensembles of multi-objective decision trees, in: *European conference on Machine Learning, Lecture Notes In Artificial Intelligence*, 2007, pp. 624–631.
- [28] J. Struyf, S. Džeroski, Constraint based induction of multi-objective regression trees, in: *Proc. of the 4th International Workshop on Knowledge Discovery in Inductive Databases KDID - LNCS 3933*, Springer, 2006, pp. 222–233.
- [29] I. Slavkov, V. Gjorgjioski, J. Struyf, S. Džeroski, Finding explained groups of time-course gene expression profiles with predictive clustering trees, *Molecular BioSystems* 6 (4) (2010) 729–740.
- [30] C. Vens, J. Struyf, L. Schietgat, S. Džeroski, H. Blockeel, Decision trees for hierarchical multi-label classification, *Machine Learning* 73 (2) (2008) 185–214.
- [31] L. Breiman, Bagging predictors, *Machine Learning* 24 (2) (1996) 123–140.

- [32] L. Breiman, Random forests, *Machine Learning* 45 (1) (2001) 5–32.
- [33] I. Dimitrovski, S. Loskovska, Content-based retrieval system for x-ray images, in: *International Congress on Image and Signal Processing*, 2009, pp. 2236–2240.
- [34] D. Keysers, T. Deselaers, C. Gollan, H. Ney, Deformation models for image recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29 (8) (2007) 1422–1435.
- [35] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, in: *IEEE conference on Computer Vision and Pattern Recognition*, Vol. 2, 2006, pp. 2169–2178.
- [36] D. Ziou, S. Tabbone, Edge detection techniques an overview, *Pattern Recognition and Image Analysis* 8 (24) (1998) 537–559.
- [37] J. Canny, A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8 (6) (1986) 679–698.
- [38] J. Zhang, M. Marszalek, S. Lazebnik, C. Schmid, Local features and kernels for classification of texture and object categories: A comprehensive study, *International Journal of Computer Vision* 73 (2) (2007) 213–238.
- [39] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, J. M. Geusebroek, Visual word ambiguity, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 99 (1). doi:10.1109/TPAMI.2009.132.
- [40] R. D. C. Team, R: A language and environment for statistical computing (2009).
URL <http://www.R-project.org>
- [41] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm> (2001).
- [42] S. Nowak, Imageclef@icpr2010 – photo annotation task, <http://www.imageclef.org/2010/ICPR/> (2010).
URL <http://www.imageclef.org/2010/ICPR/>
- [43] H.-T. Lin, C.-J. Lin, R. C. Weng, A note on platt’s probabilistic outputs for support vector machines, *Machine Learning* 68 (2007) 267–276.