Semantic lattices for multiple annotation of images

Anne-Marie Tousch, Stéphane Herbin, Jean-Yves Audibert

Semantic Lattices for Multiple Annotation of Images

Anne-Marie Tousch
ONERA
&
Université Paris-Est
Ecole des Ponts ParisTech
CERTIS
Marne-La-Vallée, France
tousch@certis.enpc.fr

Stéphane Herbin
ONERA
Châtillon, France
stephane.herbin@onera.fr

Jean-Yves Audibert
CERTIS
&
Willow - ENS / INRIA
Paris, France
audibert@certis.enpc.fr

ABSTRACT

We address the problem of describing precisely an object present in an image. The starting point is a semantic lattice defining all possible coherent object descriptions through inheritance and exclusion relations. This domain knowledge is used in a learning process which outputs a set of coherent explanations of the image valued by their confidence level. Our first contribution is to design this method for multiple complexity level image description. Our secondary focus is to develop rigorous evaluation standards for this computer vision task which, to our knowledge, has not been addressed in the literature despite its possible use in symbolic annotation of multimedia database. A critical evaluation of our approach under the proposed standards is presented on a new appropriate car database that we have collected.

1. INTRODUCTION

1.1 Problem statement

This work is about generating multiple image annotations corresponding to various levels of semantic precision. The origin of the problem we address lies in the nature of semantic interpretation of data. Indeed, describing the content of an image is an ill-posed problem: the type of relevant description depends on the context of use which is not univocally defined by the image alone. For instance, when observing an image containing a car — the type of data that will illustrate our approach (see Fig. 1) — one may be interested in finding its brand or in characterizing its shape or its size. One may also be interested in identifying the car model name, or even its version.

In image retrieval problems, one solution to solve the inherent ambiguousness of data description is to make use of image content description based techniques, i.e. to rely only on universal image features such as color, shape or texture models. The claim of this kind of approach is that it is able to get around the semantic gap issue, using for instance relevance feedback querying strategies [35].

Another trend to carry out semantic analysis is to exploit knowledge representations such as ontologies on symbolic metadata. The use of semantic tools is expected to master the polysemy or imprecision of both symbolic annotations and queries. In this family of approaches, two subproblems need to be solved: the construction of relevant annotations, and the design of a similarity measure between the annotations and a compatible form of the research query [13].

The target context of this study is domain specific applications. This introduces several peculiarities:

- meaningful differences between data rely on very specific details which are hard to guess without expert knowledge;
- users usually master the specific concepts and vocabulary of the domain;
- the size of the database is large, but typically less than 10^5 items;
- annotated or reference data are scarce.

Content based image retrieval may not be the adequate framework for dealing with this kind of constraints, since efficient methods often rely on a rather large learning database, or address too coarse classification for domain specific applications. We propose to base our approach on the following features:

- image indexing is automatic and is based on few reference annotated data;
- annotations should be coherent with a domain knowledge representation;
- research queries are symbolic or textual;
- semantic ambiguousness is solved offline by generating multiple annotations,

this last point being the main contribution of our work.

Figure 1: images of cars from 7 different classes.

Table 1: A possible sequence of outputs in our working scheme: the more precise the description, the less reliable.

Description	Confidence
Hatchback	0,78
Supermini	0,66
Hatchback, Supermini	0,6
Peugeot	0,55
Hatchback, Supermini, Peugeot	0,5
Hatchback, Supermini, Peugeot, 206	0,45
Hatchback, Supermini, Peugeot, 206, 3 doors	0,4

A multiple annotation has to be understood as a distribution of consistent annotations each valued by a confidence coefficient (Tab. 1). Each element in this list is assumed to address a given level of semantic precision roughly characterized by the number of labels.

Retrieval is easy when using this kind of metadata: a data is retrieved if all the query terms are contained in one of the annotations. This scheme is therefore a simple string matching procedure, and does not rely on a sophisticated semantic similarity measure. However, each matched single annotation comes with a confidence coefficient computed offline in the indexing phase that can be used for sorting or filtering out the retrieved data.

In this framework, the indexing or annotation process is automatic and possibly unfaithful, due to lack of learning or reference data. This means that all annotations may not have the same level of confidence, especially if they are scarce and characterize only a small number of samples. Thus, the confidence coefficient measures the quality of the indexing step, not the online matching score: since it is a priori, it can be computed offline.

Not all lists of labels are meaningful. Each label has a signification — it refers to an object or a property — and is related to other labels in some fashion. One possible way to decide the consistency of a list of labels is to embed it into a logical environment encoding knowledge about the true states of world. Our goal is not to make logical inferences, however. Since we are interested in calculating confidence coefficients associated with *all* consistent label lists in order to sort them, we use a simple way of representing domain knowledge by defining two types of relations between subset of labels: *inheritance* and *exclusion*.

Domain knowledge is equivalent to introducing the constraints able to define the subset of consistent lists of labels. Figure 2 shows an example of inheritance or `is-A` relations between labels used for the description of cars. Once the set of consistent labels have been identified, they can be organized by subset order in a global structure organizing the descriptions in various complexity levels: a *semantic lattice*. In the general case, such as depicted in Fig. 2, the lattice does not reduce to a tree and cannot be considered as a mere taxonomy: this constitutes one of the main difficulties to master in our problem.

The work presented in this paper explains how such a semantic lattice is used for describing images with lists of labels valued with a confidence coefficient. The main contributions are:

- design of a processing chain for multiple complexity level image description;
- definition of an operating criterion, an error/complexity curve, for multiple description management;
- careful evaluation of the chain on a database of car images.

1.2 Related work

Most of the processing chains developed for the description of image content follow the same outline: extraction of reliable informative features, and construction of a mapping from those features to an interpretation space.

1.2.1 Image features

Our approach is based on the extraction of local features, as is now quite usual in object recognition approaches [1].

One policy for computing localized image features is first choosing a location using interest, regularly spaced or random points, then characterizing the image locally. Marszalek and Schmid [23] use patches located by salient region and point detectors. Lazebnik et al. [17] use dense sampling on a regular grid. Nowak et al. [25] show that random sampling achieve comparable performance for bag-of-features type image classification. Chen et al. [12] use the regions texture and color properties of the segmented image coupled with a multiple-instance learning algorithm.

Another local image feature computing policy, closer to the one we are following in this work, is to build detectors adapted to very specific though simple characteristics. This has been used for a long time in face recognition, for instance, where eye, nose or mouth detectors form a basis used to detect or normalize face appearances [1].

1.2.2 Interpretation of features

Images may be described along different directions: Hollink et al. [15] propose to distinguish between non visual (date, photographer's name, digital format ...), perceptual (color, shape, texture, spatial relations ...) or conceptual (event, name of person, place ...) levels. The fundamental issue of data annotation, often referred to as the *semantic gap* [34, 18], addresses the problem of relating the perceptual and the conceptual levels. The perceptual level, somehow likened to what is called the image *content*, may also be described using sophisticated symbolic representations [21].

The majority of studies have addressed image description either as detection or as categorization problem, i.e. the descriptions belong to conceptual spaces with simple topology. More recently, image retrieval issues have requested more flexible or semantically colored types of description inspired by document classification applications [14, 4, 8] and have addressed multi-label description [3, 19, 7, 9] or ontology based annotations reduced to taxonomic or hierarchical descriptions [29, 5, 30, 23, 27]. [20] introduce uncertainty on the labelling and compute a probability for each concept given an image. However, they treat each concept regardless of their semantic relations. To the best of our knowledge, semantic hierarchies or taxonomies have been used mostly to improve performance in recognition tasks [24], or have

[1] See the classification method description of the Pascal VOC Challenge 2007
`http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/workshop/everingham_cls.pdf`

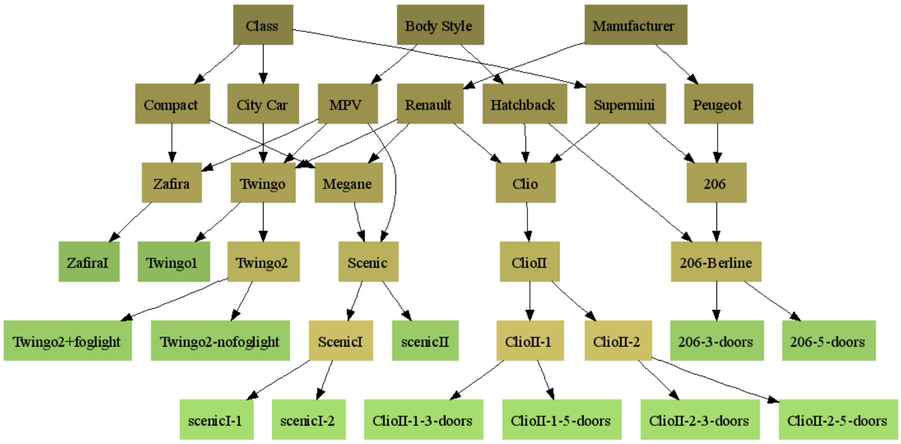


Figure 2: Example of an `is-A` graph. Arrows show inheritance relations. Some labels have multiple inheritance. Labels sharing a common parent are exclusive.

been generated as a by-product of a classification scheme [32]; they have not been used to generate multiple description level characterization.

Our approach intensively uses machine learning techniques. In this field, several recent studies have addressed learning in complex or structured spaces extending kernel-based approaches or graphical models to structured spaces such as multilabels, strings or hierarchies [2, 11, 10, 31].

The conceptual level studied in this paper has a lattice topology, and is not organized as a true hierarchy: a description, i.e. a list of labels, may have more than one parent, i.e. may have more than one simpler consistent description. This peculiarity motivated the specific developments presented in following.

1.3 Overview of the approach

The objective of this work is to build a process for the description of images or objects with multiple levels of semantic precision. The output is a series of lists of labels, each valued with a reliability or confidence coefficient. The consistency of each list of labels is guaranteed by a semantic lattice aiming at representing domain knowledge.

Our global processing chain is divided into four main tasks (Fig. 3):

1. Extracting image information by computing a *signature*.
2. Calculating a probability for each list of labels based on the image signature.
3. Ensuring global coherence of the probabilities using a semantic lattice.
4. Issuing the series of consistent lists of labels ordered by their probability.

The rest of the article is organized as follows: the computation of the signature is detailed in section 2; the image annotation step using signatures is presented in section 3; section 4 is about designing an adequate criterion for performance evaluation; experiments and results are presented in section 5 along with the methods we compared with.

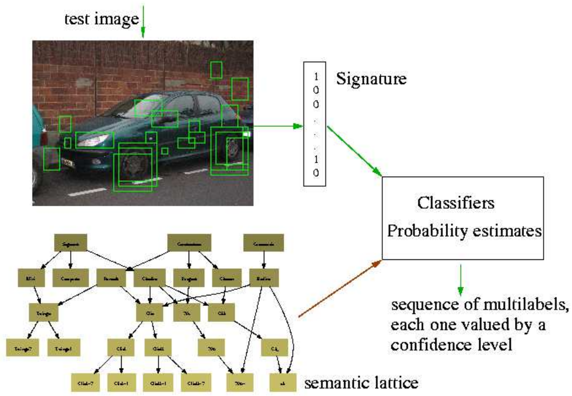


Figure 3: Synopsis of the processing chain for the multiple complexity level image description of objects.

2. IMAGE FEATURES

Image representation based on the detection of local features have proven successful. Zhang et al. [33] have provided an in-depth study of the state-of-the-art methods using local features and kernels for object categorization.

We choose to represent an image as a vector of binary values where each binary value corresponds to the presence or not of a particular local patch detector in the image, thus yielding a low-level semantic vocabulary. As in the bag of words approach, we do not represent spatial relationship between features. However, contrarily to the latter technique, the vocabulary is predefined through labeling in the training images of small patches potentially discriminant for identification of each class (logo, lights, . . . ; see Fig. 4).

A detector is designed for each word of the vocabulary. It is based on

- representing the patches by a SIFT-like descriptor [22] which is known to cope well with illumination and contrast variations: we use 4×4 local histograms corresponding to a 4×4 square grid, each histogram containing 8 bins corresponding to possible orientations of the gradient in one of the 4×4 squares.
- a one-class-SVM [28] with an histogram intersection

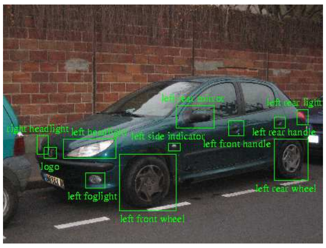


Figure 4: Example of a labeled car. A word is defined using small patches of one part for one car model.

Figure 5: Examples of detections. Last column is a false detection, as a rear mirror of the wrong model is detected. This kind of false detection is expected to happen repeatedly and is a reason for the noise on the signature.

kernel [6] which has already been successfully used with the SIFT descriptors [17]. The advantage of using one-class-SVMs is essentially to avoid the definition of the "negative" class through instances: an object is better defined by what it is than what it is not.

Due to the limited size of the training set and in order to improve the one-class-SVM classifier, we artificially increase the training set size by applying small affine transformations of the training images. In order to control the false alarms, we look for the patch corresponding to a particular word only at scales and positions similar to the one observed in the training set.

To find the presence in an image of a word of this vocabulary, we use a sliding window technique, that is we look for this patch at different scales and locations. If the word is found at one scale or one location, the corresponding binary value is put to 1, otherwise it is put to 0. This signature is simple, and may be improved in many ways. Since the emphasis of the work presented in this article is on multiple description level management, we did not spend too much time in optimizing the feature extraction step. As a number of detectors may fail (see Fig. 5), the signatures used in the next section may be highly noisy.

3. MULTIPLE SEMANTIC ANNOTATIONS

The output metadata built by our procedure has to be understood as a probabilistic distribution on annotations. The underlying ideas governing its design can be summarized the following way:

- use a fixed domain specific vocabulary = set of labels;
- an annotation is reduced to a conjunction of labels, i.e. a multilabel;
- not all multilabels are consistent;
- some multilabels are less general than others, i.e. yield to a higher semantic precision.

3.1 Consistent multilabels

Knowledge representations have been widely studied for *semantic web* or Web 2.0 applications. WordNet is one of the most popular one. However, our probabilistic formalism relies on both inheritance and exclusion properties of labels, which are not explicitly represented in WordNet. For instance, the entry for *car* in WordNet indifferently returns the concepts *compact car*, *minicar*, which describe the size of a car, *hatchback*, *station wagon*, which describe its shape, or *taxi*, *ambulance*, which describe its use. Thus the simple hierarchy of WordNet cannot inform us that *taxi* and *ambulance* are conflicting concepts, no more than that *taxi* and *station wagon* are compatible. Moreover, as we are concentrating on cars, we would like to have a more precise description level than the one provided by WordNet. For those two reasons — no explicit exclusion between labels and no specific enough vocabulary — we were forced to design our own knowledge representation.

In our setting, annotation consists in assigning probabilities to a series of lists of labels or multilabels (we will use equivalently both expressions in the following). The consistency of multilabels is assessed by specifying inheritance and exclusion relationships between single labels.

Inheritance is defined using an `is-A` graph on labels such as the one shown on Fig. 2. Multiple inheritance is allowed: for instance, the label `Clio` has two parents, `Renault` and `Supermini`. This has to be contrasted with taxonomic organization where each category may not have more than one parent.

Exclusion is also described using the same `is-A` graph. We apply the underlying rule stating that "labels sharing a common parent are exclusive", as is also true for taxonomies.

A list of labels will be said consistent if 1/ each single label contained in the list has all its ancestors in the list and 2/ no two single labels in the list are exclusive. The global set of consistent multilabels can be organized by set order, leading to a "*semantic lattice*".

3.2 Complexity of multilabels

The notion of complexity of description or semantic precision plays a central role in our problem. Let's introduce a few notations to define it. The `is-A` graph is a directed graph denoted by $\mathcal{G}$. Each of its nodes is associated with a single label. Let N_g be the number of labels or nodes. If i is a node, we write $par(i)$ the set of parents of i, i.e. the nodes connected to i, and $anc(i)$ the set of its ancestors. A *root* or *base node* is a node without parent. A *leaf* is a node without child.

A multilabel is a subset of labels. It is represented as a binary vector $\mathbf{y} \in \{0,1\}^{N_g}$ where the j-th coordinate y_j equals 1 if label j belongs to the multilabel. The set of consistent multilabels, defined by inheritance and exclusion constraints, is denoted by $\mathcal{Y}_g$. The set $\mathcal{Y}_g$ can be easily enumerated due to the low degree of connectivity of the graph. In our application, we used 54 labels, and found 92 consistent multilabels.

Nodes from the graph $\mathcal{G}$ and multilabels from the set $\mathcal{Y}_g$ are related. A given node i can be mapped to a consistent list of labels $\mathbf{y}(i)$ by aggregating the labels associated with all its ancestors $anc(i)$.

Multilabels associated with the leaves of graph $\mathcal{G}$ have the highest description complexity. They characterize the data with the highest degree of precision. These leaves define exclusive classes. We assume that each data ultimate ground truth is such a multilabel. In the application tested, we have 20 leaves.

We state that the description complexity is equal to the number of labels used to describe the data, and define:

- node complexity: $\mathcal{C}(i) = |anc(i)| + 1$
- multilabel complexity: $\mathcal{C}(\mathbf{y}) = |\mathbf{y}| = \sum_{j=1}^{N_g} y_j$

Node and multilabel complexities are equivalent since we have $\mathcal{C}(i) = \mathcal{C}(\mathbf{y}(i))$.

3.3 Multilabel probability computation

The computation of a probability for each consistent multilabel is done in two steps: computation of a probability for each node of the graph $\mathcal{G}$; propagation of those probabilities to all the other consistent multilabels, ensuring global coherence.

The computation of a probability for node i, or equivalently for multilabel $\mathbf{y}(i)$, is achieved using a binary SVM with gaussian kernel applied on the data signature. For each node, the database is divided into positive and negative samples in a one-versus-rest approach. A sample will be considered positive if its ground truth contains all the labels of the multilabel associated with the node.

This scheme leads to highly unbalanced problems when going down the graph. We handled this by using different SVM C parameters for positive and negative data and using an adapted error. After training, we followed Platt's method [26] for converting SVM outputs into probabilities by fitting a sigmoid on the classifier output.

Let $\{p_1, \dots, p_{N_g}\}$ be the set of the SVM probabilistic outputs for all nodes in the graph given an input signature. We need to estimate probabilities for the multilabels that are not associated with a node due to multiple inheritance.

The idea is to build the global distribution of multilabel probabilities on the probabilities assigned to the leaves, i.e. the most complex descriptions. Indeed, the leaves make a partition of the data — they are exhaustive and mutually exclusive — so that each multilabel should verify:

$$p(\mathbf{y}) = \sum_{j \in \hat{\mathcal{G}}} y_j p_j. \tag{1}$$

where $p(\mathbf{y})$ is the probability assigned to multilabel $\mathbf{y}$ and $\hat{\mathcal{G}}$ is the set of leaves. This equality must hold also for multilabels associated with each node $\mathbf{y}(i)$:

$$p_i = p(\mathbf{y}(i)) = \sum_{j \in \hat{\mathcal{G}}} y(i,j) p_j. \tag{2}$$

where $y(i,j)$ is the j-th coordinate of multilabel $\mathbf{y}(i)$.

The probabilities obtained using the direct computation of the SVMs may not satisfy the constraint (2) for every node. We seek a regularized approximation $\tilde{p}_i$ of the probabilities assigned to each leaf, optimizing the criterion :

$$\min_{\tilde{\mathbf{p}}} \sum_{i \notin \hat{\mathcal{G}}} w_i \cdot (p_i - \sum_{j \in \hat{\mathcal{G}}} y(i,j)\tilde{p}_j)^2 + \sum_{j \in \hat{\mathcal{G}}} w_j \cdot (p_j - \tilde{p}_j)^2, \tag{3}$$

$$\text{s.t.} \quad \sum_{j \in \hat{\mathcal{G}}} \tilde{p}_j = 1, \quad \text{and } \forall j, \tilde{p}_j > 0, \tag{4}$$

where $\tilde{\mathbf{p}}$ is the estimated probabilities on the leaves, and w_i is a weight on node i such that probabilities on low complexity nodes are favored. We choose to take $w_i = \frac{1}{\mathcal{C}(i)^\gamma}$, and after experiments, we set γ to 0.01. This is a simple convex quadratic programming problem of dimension the number of leaves of the semantic graph. The two members in eq. (3) correspond respectively to (a) trying to reach the constraint (2) for nodes and (b) keeping probabilities for leaves the nearest possible to the probabilistic outputs.

The probability of any multilabel in $\mathcal{Y}_g$ is then computed using equation 1 where p_i is replaced by $\tilde{p}_i$. Using this global computation scheme — SVM on every node + regularization — we have assigned a probability to each consistent multilabel. The next section describes how to exploit this probability distribution, and how to evaluate its descriptive capacity.

4. EVALUATION

An algorithm solving our multiple complexity level image description task should output confidence levels for any possible description. To be consistent, these probabilities need to be decreasing along any chain of multilabels $\mathbf{y}_1, \dots, \mathbf{y}_k$ such that $\mathbf{y}_1 \subsetneq \dots \subsetneq \mathbf{y}_k$.

For a test sample with true multilabel $\mathbf{t}$, any multilabel $\mathbf{y} \subseteq \mathbf{t}$ is a correct answer. The semantic precision of the final answer of the algorithm is controlled by an input confidence parameter, denoted hereafter p, that the user can tune. The idea is to make the algorithm output the most complex (or precise) explanation of the image which has a confidence level greater than p. The choice of the multilabel of maximum complexity with probability larger than p is done through the following steps:

1. Threshold the set of multilabels to keep only multilabels having confidence coefficients larger than p:
$$\mathcal{Y}_g^p = \{\mathbf{y} \in \mathcal{Y}_g | p(\mathbf{y}) \geq p\}, \tag{5}$$
2. Build the set of maximal (in complexity) multilabels among the high confidence multilabel set $\mathcal{Y}_g^p$:
$$\partial\mathcal{Y}_g^p = \{\mathbf{y} \in \mathcal{Y}_g^p | \forall \mathbf{y}' \in \mathcal{Y}_g, \text{ s.t. } \mathbf{y} \subsetneq \mathbf{y}', \mathbf{y}' \notin \mathcal{Y}_g^p\}, \tag{6}$$
3. Choose the multilabel $\hat{\mathbf{y}}$ maximizing the probability $p(\mathbf{y})$ in $\partial\mathcal{Y}_g^p$.

To evaluate the classification efficiency of algorithms solving our multiple complexity level image description task, we plot an error/complexity curve. This curve is parameterized by the confidence factor $p \in [0,1]$, a point $(c(p), \varepsilon(p))$ being the mean complexity and mean error of answers for p on the test set:

$$c(p) = \frac{1}{N} \sum_{i=1}^{N} \mathcal{C}(\hat{\mathbf{y}}_i), \tag{7}$$

$$\varepsilon(p) = \frac{1}{N} \sum_{i=1}^{N} \ell(\hat{\mathbf{y}}_i, \mathbf{t}_i), \tag{8}$$

where $\mathbf{t}_i$ is the ground truth for sample i and ℓ is the 0/1-loss function :

$$\ell(\mathbf{y}_1, \mathbf{y}_2) = \begin{cases} 0 & \text{if } \mathbf{y}_1 \subseteq \mathbf{y}_2, \\ 1 & \text{otherwise.} \end{cases} \tag{9}$$

This error/complexity curve is the fundamental tool used to evaluate and compare multiple description algorithms.

5. EXPERIMENTS

5.1 Dataset

Our dataset is composed of 644 images of 20 classes of cars with varied inter-class visual and semantical differences. The dataset is divided into two separate sets namely "$\mathcal{L}_a$"

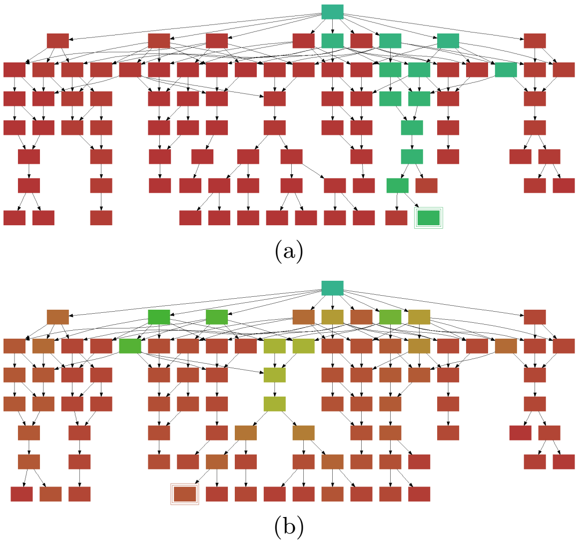


Figure 6: Each graph shows the probability of each multilabel for a given test image. Ground truth multilabel has multiple borders. (a) shows a favorable case. (b) shows a more ambiguous scenario.

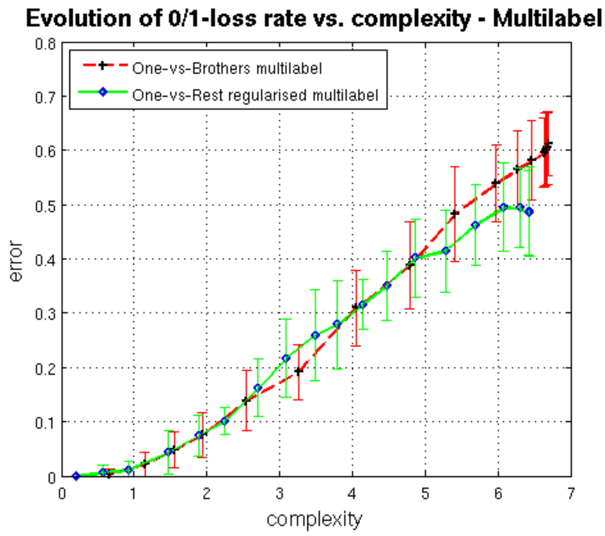


Figure 7: Mean error rate vs. mean multilabel complexity computed on the 318 test images in $\mathcal{L}_b$ using 0/1-loss.

where all images were richly annotated (326 images) and "$\mathcal{L}_b$" where only the class was given (318 images). The distribution of examples per class is highly variable depending on their real global statistical distribution (the photos were taken in the streets) varying from 3 to 29 photos for a class in $\mathcal{L}_a$ or $\mathcal{L}_b$. Illumination changes, reflections and car colors create intra-class variation. The viewpoint angle was limited to a relatively small range included in "3/4 left-front view". Yet the viewpoint change is high enough to have important differences on the front of the car, such as the right headlight not always being visible.

The set $\mathcal{L}_a$ was used to design the individual detectors. We divided it into 5 folds to cross-validate the one-class SVM parameter C. The best parameter was used to train the classifier on the whole database. The signatures were then computed for all images in $\mathcal{L}_b$ using the detectors thus trained.

5.2 Probabilities

We used SVMs with RBF kernels and a two-level cross-validation on $\mathcal{L}_b$. The first level is used to find the optimal SVM parameters C and σ. The second level is used to generate the probabilistic outputs for error estimation. All probabilities are regularized using (3), and a probability is computed for each consistent multilabel. These probabilities are shown in the graphs Fig. 6. Each node in this graph corresponds to a multilabel, and each link from $\mathbf{y}_i$ to $\mathbf{y}_j$ denotes the fact that multilabel $\mathbf{y}_i$ is included in multilabel $\mathbf{y}_j$ with only one more label in $\mathbf{y}_j$. A green node means probability near to 1, whereas a red node means probability 0. The figure shows different scenarios : in graph 6(a), there is strong confidence on the output; in graph 6(b), the result is more ambiguous, and even low confidence thresholds are likely to give a low complexity output.

5.3 Classification Results

As a first step, we test the algorithm performance in a classification framework. The error/complexity curve obtained with our method is shown Fig. 7 along with our adaptation of Marszalek and Schmid's algorithm [23]. The algorithm proposed by [23] is adapted to trees or taxonomies. At node A they train a binary SVM for each of its child nodes B_i using positive and negative sets P and N:

$$P = \text{supp}(B_i) \quad N = \text{supp}(A) - \text{supp}(B_i), \qquad (10)$$

where supp(X) is the set of samples belonging to category X. The structure we are working on is the combination of different trees sharing some of their nodes (see Fig. 2). Thus their algorithm can be applied in each underlying tree to get a set of multilabels. The confidence threshold p is used with SVM probabilistic outputs as a stopping criteria : starting at each base node r, we descend the hierarchy while the classifier associated with the link returns a probability bigger than p, giving eventually an output $\mathbf{y}_r(p)$. For confidence p, taking as multilabel the union of the outputs from the different hierarchies might not be consistent. In this case, we impose a consistent multilabel output by taking the one of greatest complexity in the union of the multilabels $\mathbf{y}_r(p)$:

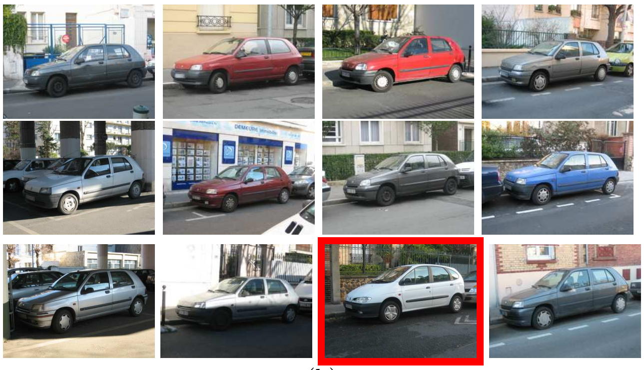
$$\hat{\mathbf{y}}(p) = \underset{\mathbf{y}}{\text{argmax}}\{\mathcal{C}(\mathbf{y})|\mathbf{y} \subseteq \bigvee_{r \text{ root}} \mathbf{y}_r(p)\}. \qquad (11)$$

We compute the mean complexity and mean loss $(c(p), \varepsilon(p))$ on the test set for several values of p to draw the curve in figure 7. The results show that our algorithm performs better, especially for higher complexities. For a mean complexity of 6 on the database, our algorithm gives a mean 0/1-loss rate of 49%, compared to 54% for [23].

5.4 Retrieval results

The principle of multiple annotation is tested on an image retrieval problem. The protocole conforms to a standard Google-like session. Queries are conjunctions of keywords and results consist of ranked lists of data. Since we are interested in a domain specific context with moderate size database, evaluation can rely on the knowledge of the entire database and on the computation of precision/recall curves.

(a)



(b)

(c)

Figure 8: First 12 images retrieved (from left to right and top to bottom) respectively for multilabels (a) Renault, (b) Supermini,Hatchback,Renault,Clio,ClioI and (c) Supermini,Hatchback,Renault,Clio,ClioII,ClioII-2. The system is able to retrieve objects from classes with relatively large intra-class variation as in (a) as well as distinguishing small variations. The only falsely retrieved image is marked with a red border.

The retrieval algorithm is a simple string matching comparing the query and the annotations. The possible queries are equal to the set of consistent annotations as explained in section 3. The global retrieval performance evaluation is based on examining the returned lists of data for various thresholds on the confidence coefficients.

Figure 8 shows the first 12 images found for queries of increasing semantic precision. The computations being done offline, these results are obtained instantaneously for the whole database. The corresponding Precision-Recall curves for the same multilabels are shown in Fig. 9, along with other related queries. The thick green curve is the average over all possible queries where each point is obtained by thresholding the returned confidence coefficients.

Those curves show large variations in retrieval behavior. The equal precision-recall points vary from 50% to 90% on Fig. 9, with an average at 68%. The performance decreases with the complexity of the query, although not strictly montonically. It is also related to the number of items in each class, as was noticed also in [16], though some classes with few examples give also good results. This is not surprising, since the quality of annotations depends on learning, and therefore on the amount of available data in each class.

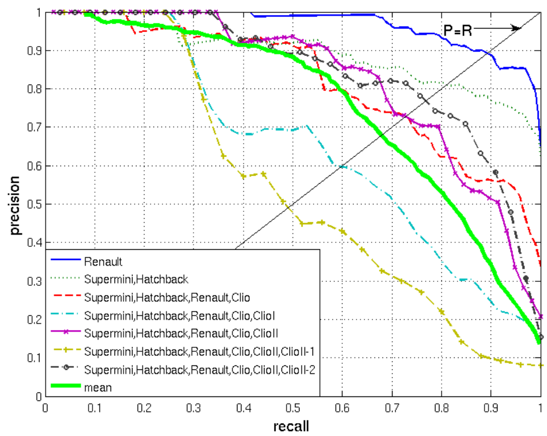


Figure 9: Precision-Recall for multilabels at different semantic level. Retrieval is done on the entire database.

6. CONCLUSION AND FUTURE WORK

This article settles a framework for dealing with multiple level semantic annotation of images, allowing the setting of a trade-off between confidence and semantic precision. A complete processing chain to describe images with confidence-rated multilabels was presented. We defined a criterion for the evaluation of such multilabel classifiers. Both the comparison with an algorithm adapted from a hierarchical classification task and the tests in image retrieval showed promising results.

The work presented in this paper can be developed in several directions. The image signature can be improved using other types of features or using geometrical relations between local descriptors. A more interesting approach would be to use the domain knowledge to control the type of discriminant features in order to build a more “intelligent” signature.

7. ACKNOWLEDGEMENTS

This work was supported in part by the Agence Nationale de la Recherche project “Modèles Graphiques et Applications”.

8. REFERENCES

[1] O. Arandjelovic and A. Zisserman. Automatic face recognition for film character retrieval in feature-length films. In *CVPR05*, pages I: 860–867, 2005.

[2] G. H. Bakir, T. Hofmann, B. Schölkopf, A. J. Smola, B. Taskar, and S. V. N. Vishwanathan. *Predicting Structured Data (Neural Information Processing)*. The MIT Press, 2007.

[3] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. Blei, and M. Jordan. Matching words and pictures. *JMLR*, 3:1107–1135, 2003.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[5] S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, I. Kompatsiaris, S. Staab, and M. Strintzis. Semantic annotation of images and videos for multimedia analysis. In *ESWC2005*, pages 592–607, 2005.

[6] S. Boughorbel, J. Tarel, and N. Boujemaa. Generalized histogram intersection kernel for image recognition. In *ICIP05*, pages III: 161–164, 2005.

[7] M. R. Boutell, J. Luo, S. Xipeng, and C. M. Brown. Learning multi-label scene classification. *Patt. Recog.*, 37(9):1757–1771, Sept. 2004.

[8] L. Cai and T. Hofmann. Hierarchical document categorization with support vector machines. In *CIKM '04*, pages 78–87, 2004.

[9] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *PAMI*, 29(3):394–410, 2007.

[10] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni. Hierarchical classification: combining bayes with svm. In *ICML '06*, pages 177–184. ACM Press, 2006.

[11] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni. Incremental algorithms for hierarchical classification. *JMLR*, 7:31–54, 2006.

[12] Y. Chen, J. Bi, and J. Wang. MILES: Multiple-instance learning via embedded instance selection. *PAMI*, 28(12):1931–1947, December 2006.

[13] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2):1–60, 2008.

[14] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99*, pages 50–57. ACM, 1999.

[15] L. Hollink, G. Schreiber, B. Wielinga, and M. Worring. Classification of user image descriptions. *Int. J. Hum.-Comput. Stud.*, 61(5):601–626, 2004.

[16] D. P. Huijsmans and N. Sebe. How to complete performance graphs in content-based image retrieval: Add generality and normalize scope. *PAMI*, 27(2):245–251, 2005.

[17] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR06*, pages II: 2169–2178, 2006.

[18] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain. Content-based multimedia information retrieval: State of the art and challenges. *ACM Tr. Multim. Comp. Com. Appl.*, 2(1):1–19, 2006.

[19] J. Li and J. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *PAMI*, 25(9):1075–1088, 2003.

[20] J. Li and J. Z. Wang. Real-time computerized annotation of pictures. In *MULTIMEDIA '06*, pages 911–920, New York, NY, USA, 2006. ACM.

[21] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma. A survey of content-based image retrieval with high-level semantics. *Patt. Recog.*, 40(1):262–282, 2007.

[22] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[23] M. Marszałek and C. Schmid. Semantic hierarchies for visual object recognition. In *CVPR'07*, 2007.

[24] A. McCallum, R. Rosenfeld, T. M. Mitchell, and A. Y. Ng. Improving text classification by shrinkage in a hierarchy of classes. In *ICML '98*, pages 359–367, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[25] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*. Springer, 2006.

[26] J. Platt. *Probabilistic outputs for support vector machines and comparison to regularize likelihood methods*, chapter 5, pages 61–74. MIT Press, 2000.

[27] A. Popescu, C. Millet, and P.-A. Moëllic. Ontology driven content based image retrieval. In N. Sebe and M. Worring, editors, *CIVR*, pages 387–394. ACM, 2007.

[28] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Comput.*, 13(7):1443–1471, 2001.

[29] M. Srikanth, J. Varner, M. Bowden, and D. Moldovan. Exploiting ontologies for automatic image annotation. In *SIGIR '05*, pages 552–558, 2005.

[30] G. Stamou, J. van Ossenbruggen, J. Pan, G. Schreiber, and J. Smith. Multimedia annotations on the semantic web. *Multimedia, IEEE*, 13(1):86–90, Jan.-March 2006.

[31] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 6:1453–1484, 2005.

[32] G. Wang, Y. Zhang, and L. Fei-Fei. Using dependent regions for object categorization in a generative framework. In *CVPR '06*, pages 1597–1604, 2006.

[33] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2):213–238, 2007.

[34] R. Zhao and W. I. Grosky. Bridging the semanitic gap in image retrieval. In *Distributed multimedia databases: techniques & applications*, pages 14–36. IGI Publishing, Hershey, PA, USA, 2002.

[35] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Syst.*, 8(6):536–544, 2003.