Multi-dimensional Keyword-based Image Annotation and Search[*]

Ilaria Bartolini
DEIS, University of Bologna, Italy
i.bartolini@unibo.it

Paolo Ciaccia
DEIS, University of Bologna, Italy
paolo.ciaccia@unibo.it

ABSTRACT

Effective keyword search on image databases is a major open problem, due to the inherent imprecision of keywords (*tags*) used to describe images' content. In this paper we present a novel approach to deal with this problem, as implemented in the Scenique image retrieval and browsing system. Scenique is based on a multi-dimensional model, where each dimension is a tree-structured taxonomy of concepts, also called *semantic tags*, that are used to describe the content of images. We first describe an original algorithm, called Ostia (Optimal Semantic Tags for Image Annotation), that, by exploiting low-level visual features, tags, and metadata associated to an image, is able to predict a high-quality set of semantic tags for that image. Then, we describe how semantic tags can be effectively used for the purpose of improving the precision of keyword search.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Image Search and Retrieval

General Terms

Algorithms

Keywords

Taxonomies, Keywords, Semantic tags, Annotation

1. INTRODUCTION

Automatic image annotation aims to enable text-based techniques (search, browsing, clustering, classification, etc.) to be applied also to objects that otherwise could only be dealt with by relying on feature-based similarity assessment, which is known to be inherently imprecise [18]. Approaches to automatic image annotation include a variety of techniques, and they even differ in what "annotation" actually means, ranging from enriching images with a set of keywords (or *tags*) [10, 13, 1, 7, 11], to providing a rich semantic description of image content through the concepts of a full-fledged RDF ontology [16]. Further, solutions may differ in what kind of tags/concepts they ultimately provide, in this case the difference being among general-purpose systems and others that are tailored to discover only specific concepts/classes [15, 20].

It is however a fact that even text-based techniques, as exemplified by the image search extensions of Google and Yahoo!, and by systems like Microsoft's Photo Gallery, Google Picasa, and Yahoo's Flickr, yield a highly variable retrieval accuracy. This is due to the imprecision and the incompleteness of the manual annotation process (in the case of Photo Gallery, Picasa, and Flickr), or to the poor correlation that often exists between surrounding text of Web pages and the visual image content (for the case of Google and Yahoo!). Nonetheless, since keyword search allows data to be retrieved in a simple way and without the knowledge of data schema and query languages [6, 14], it is not easy to find alternatives that are both effective and user-friendly.

Complementary to the possibility of searching images by keyword and/or visual features is that of browsing them. In this case, images are typically organized in a hierarchical way, and users can focus on the parts of interest of the database by navigating through the hierarchy. The inadequacy of a single hierarchy has been demonstrated by systems like Flamenco [22], where *multi-faceted hierarchies* allow users to explore a data collection across multiple, orthogonal classification criteria. Along this direction in [2] we have introduced Scenique, an integrated image search and browsing system that allows images to be searched and explored using *both* tags and visual features.

In this paper we present a novel approach for the problems of image annotation and keyword-based search that: (i) Predicts for an image a set of so-called *semantic tags*, i.e., concepts taken from a set of tree-structured taxonomies; (ii) exploits such semantic annotations for improving the precision of keyword search. Semantic tags can be seen as a means to annotate images that is more precise than free tags (that have no inherent semantics), yet not so complex to be derived as concepts of RDF-like ontologies (whose semantics might not be so easy to grasp by end-users).

Figure 1 provides an intuition on the annotation problem we deal with: Given an image, possibly coming with some textual description, and a set of taxonomies, the objective

[*]This work is partially supported by the CoOPERARE MIUR Project.

KEYS'10, June 6, 2010, Indianapolis, Indiana, USA.

is to predict which are the concepts in such taxonomies that better describe the image.

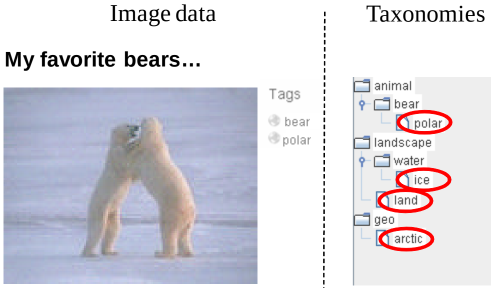


Figure 1: For the image on the left, predicted semantic tags (on the right) are `animal/bear/polar`, `landscape/water/ice`, `landscape/land/`, and `geo/artic`.

We have implemented our approach in the Scenique system and tested over real data. Preliminary results demonstrate that it can be highly effective in retrieving relevant images with respect to traditional keyword image search.

Our main contributions can be summarized as follows: (i) An effective algorithm, called Ostia (Optimal Semantic Tags for Image Annotation), able to automatically annotate images with semantic tags; (ii) a keyword-based search algorithm, called Ostia-KWS, that exploits semantic tags for improving the precision of results.

The rest of the paper is organized as follows. In Section 2 we briefly describe the model on which Scenique is based and introduce the problems we deal with. Section 3 presents and details the proposed algorithms. Section 4 experimentally evaluates the proposed techniques. Section 5 overviews related work and Section 6 concludes.

2. THE PROBLEMS

Scenique [2] is an integrated searching and browsing system that allows images to be organized and searched along a set of orthogonal *dimensions* (also called *facets*). Each dimension is organized as a tree and can be viewed as a particular coordinate used to describe the content of images. Scenique supports both *semantic* and *visual* facets, the latter being used to organize images according to their low-level features and not relevant in this paper.

A semantic dimension D_h, $h = 1, \ldots, M$, is a tree-structured taxonomy of concepts, also called *semantic tags*. More precisely, a semantic tag st_j is a path in D_h, $st_j = n_0/n_1/\ldots/n_k \in D_h$, where each n_i is a node of the taxonomy. Node n_i has a label that, for the sake of simplicity, we also denote as n_i.[1] The label of the root node is the dimension name (e.g., `location`, `subject`, etc.)

In the scenario we consider, Scenique manages an image database $\mathcal{DB} = \{I_1, \ldots, I_N\}$ and a set of M dimensions $D_1, \ldots, D_M$. In the more general case, an image $I_i \in \mathcal{DB}$ has the following components: A source file P_i (e.g., a JPEG picture); a set of low-level *visual features* F_i automatically extracted from P_i; the image *metadata* M_i, of which for the purpose of this paper we consider only the title, a textual description and a set of free tags (some, or even all, of these metadata might be missing for an image); a set of *keywords* $K_i = \{kwd_{i,j}\}$, automatically derived from M_i; and a set of *semantic tags* $ST_i = \{st_{i,j}\}$. Thus, each image I_i can be concisely represented as $I_i = (P_i, F_i, M_i, K_i, ST_i)$. The first problem we consider concerns annotation of images:

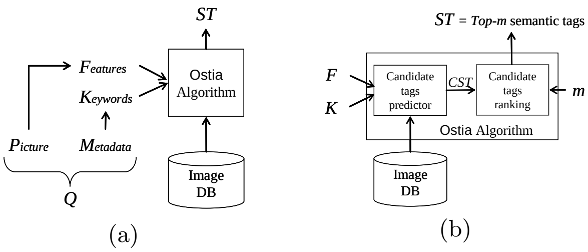


Figure 2: Image annotation: Overall approach (a) and the modules of the Ostia algorithm (b).

Problem 1 (Annotation) *Given an image database $\mathcal{DB}$ and a (query) image $Q = (P, M)$ (i.e., $F = K = ST = \emptyset$), determine the set of m $(m \geq 1)$* semantic tags *ST that better describe the content of the image Q.*

Solving Problem 1 for the database images provides semantic annotations which can be profitably exploited for solving keyword-based queries:[2]

Problem 2 (Keyword search) *Given an annotated image database $\mathcal{DB}$ and a keyword query $K = \{kwd_1, kwd_2, \ldots, kwd_n\}$, determine the set of ki $(ki \geq 1)$* images *that better match the keywords K.*

3. THE APPROACH

Sections 3.1 and 3.2 provide the solutions for Problem 1 and Problem 2, respectively.

3.1 The Ostia Algorithm

We adopt a 2-step approach to solve Problem 1, as illustrated in Figure 2 (a). First, a set of low-level visual features F and high-quality keywords K are extracted from $Q = (P, M)$. To this end we use, respectively, the feature extraction algorithm of the Windsurf library [3], which characterizes an image with color and texture features, and text analysis procedures, such as stemming, stoplist, and NLP [4] techniques,[3] not further described here for lack of space.

Once both F and K have been extracted, they are input to an algorithm, called Ostia (Optimal Semantic Tags for Image Annotation), that exploits information associated to images in the $\mathcal{DB}$ that are *similar* to Q either at the visual or the textual level (or both), to predict a set of semantic tags for Q. Ostia consists of two main modules, see Figure 2 (b). A first module is in charge of predicting a superset of ST, which are hereafter called *candidate semantic tags* (or simply candidates) and denoted CST. A second module organizes, for each dimension D_h, the candidates into a *candidate tree* $CT_h \subseteq D_h$, ranks them, and returns the top-m ones.

[1] This is only to simplify the presentation: Scenique allows the same label to be attached to multiple nodes, e.g., `activity/sport/soccer/Italy` and `activity/sport/basket/Italy`.

[2] For the purposes of this paper we focus on keyword search only, being understood that our approach can also be used if visual features of an input image are available.

[3] OpenNLP: `http://opennlp.sourceforge.net/`

3.1.1 Generating Candidate Semantic Tags

The first module of Ostia predicts, for each dimension D_h, a set of candidate semantic tags CST_h, with $CST = \bigcup_{h=1}^{M} CST_h$. The basic rationale of CST_h computation is to exploit available information of the query Q (i.e., K and F) in order to find images $I_i \in \mathcal{DB}$ that might contain tags relevant for Q.

We exploit query keywords K by applying a *co-occurrence* search on $\mathcal{DB}$ image keywords. The search provides a set of images that share at least e terms with K. We rank the images on the base of the co-occurrence value and, for the top-p images only, their keywords are added to a set RK of *relevant keywords* (which by default includes all keywords in K), and all the semantic tags are used to initialize CST. For example, if $K = \{\texttt{beach}, \texttt{sea}, \texttt{sun}\}$, $e = 2$, and there is an image I_i with $K_i = \{\texttt{beach}, \texttt{sea}, \texttt{sky}\}$ and $ST_i = \{\texttt{landscape/water/sea}\}$, then `sky` is added to RK and `landscape/water/sea` to CST.

Starting from the query features F, a *nearest-neighbors* search is performed on the $\mathcal{DB}$, which determines the set of the g images most similar to Q. For all keywords kwd_j (resp. semantic tags st_j) associated to at least one of such images, a frequency score is computed as the number of top-g images annotated with kwd_j (resp. st_j). Such annotations are then ranked based on their frequency and the top-s ones are added to RK and CST, respectively.

After the above-described steps, each relevant keyword $kwd_j \in RK$ is processed, since it can provide new candidate semantic tags. For each kwd_j we check if there is any path (i.e., semantic tag) st_j in the taxonomy of some dimension D_h terminating with a label equal to kwd_j (we call this step *joining phase*). If this is the case, st_j is added to CST_h and kwd_j deleted from RK.

For keywords that, after the joining phase, still populate RK, we apply a *keyword expansion* step in order to verify if it is possible to collect further semantic tags by means of *correlated* terms (namely, synonyms) available from WordNet.[4] For instance, if $\texttt{sea} \in RK$ and the label `sea` is not part of any dimension, whereas the semantic tag $st_j = \texttt{landscape/water/ocean}$ appears in some D_h, then st_j will be added to CST_h. For each keyword $kwd_j \in RK$, we find the matching lexical concept in WordNet, collect the synonyms of the associated synsets, add them to RK, and then apply to them the joining phase. Figure 3 shows possible ways to join a keyword to a semantic tag.

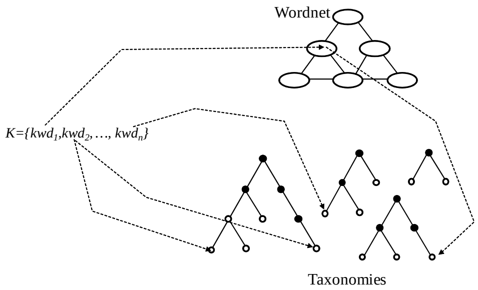


Figure 3: Ways to join keywords to semantic tags.

Algorithm 1 summarizes the above steps. Notice that, since in general a semantic tag st_j can be predicted multiple times, we keep trace of its *frequency*, $freq_j$, which will be used by the second module of Ostia.

Algorithm 1 Ostia: Candidate Semantic Tags Predictor

```
Input: $Q = (F, K)$: query image, $\mathcal{DB}$: image database,
    $e, p, g, s$: integer
Output: $CST$: candidate semantic tags
$CST \leftarrow \emptyset$, $RK \leftarrow K$;
COImg $\leftarrow$ Top(KwdSearch($K, \mathcal{DB}, e$), $p$);
                  ▷ Top-$p$ images sharing $\geq e$ kwd's with $Q$
$RK \leftarrow RK \cup \{kwd_{i,j} : I_i \in$ COImg$\}$;
$CST \leftarrow CST \cup \{st_{i,j} : I_i \in$ COImg$\}$;
NNImg $\leftarrow$ NNImgSearch($F, \mathcal{DB}, g$);
                  ▷ Top-$g$ most similar images to $Q$
$RK \leftarrow RK \cup$ Top($\{kwd_{i,j} : I_i \in$ NNImg$\}$, $s$);
                  ▷ Top-$s$ freq.-based keywords
$CST \leftarrow CST \cup$ Top($\{st_{i,j} : I_i \in$ NNImg$\}$, $s$);
                  ▷ Top-$s$ freq.-based semantic tags
$CST \leftarrow CST \cup$ Joining($RK, \{D_h\}$);
                  ▷ join keywords in RK to paths in some $D_h$
$RK \leftarrow$ GetSynonyms($RK$);
$CST \leftarrow CST \cup$ Joining($RK, \{D_h\}$);
return $CST = \{(st_j, freq_j)\}$.
```

3.1.2 Ranking the Candidates

The second module of Ostia organizes, for each dimension D_h, the candidate semantic tags CST_h into a *candidate tree* $CT_h \subseteq D_h$, and then computes the overall top-m results. Ranking is based on *weights*. The weight w_j of st_j is computed as $w_j = freq_j \cdot util_j$, where $freq_j$ is the frequency of st_j and $util_j$ is the so-called *utility* of st_j wrt *all* other candidates $st_i \in CST_h$, defined as:

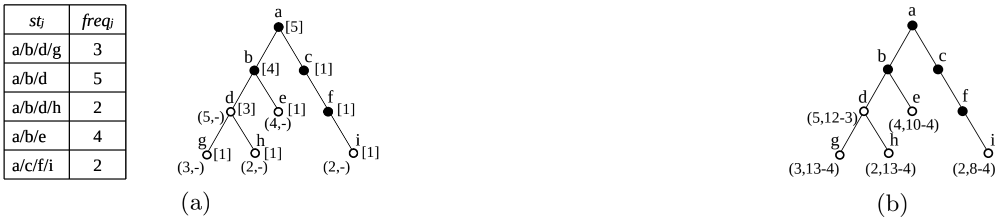
$$util_j = \sum_{st_i \in CST_h, i \neq j} \frac{\text{len}(st_j \cap st_i)}{MaxP_h} \qquad (1)$$

where $\text{len}(st_j \cap st_i)$ is the length of the common (prefix) path between st_j and st_i, whereas $MaxP_h$ is the maximum path length within the dimension D_h. Utility measures the amount of overlap between st_j and all other st_i's, and aims to score higher: a) longer (i.e., more specific) semantic tags (since for such candidates the degree of overlap with the other candidates is likely to be high), and/or b) candidates occurring in a "dense" part of the candidate tree. On the other hand, the frequency tends to be higher for more generic semantic tags because it is more common to provide generic annotations than specific ones.

Computing all the utilities by directly applying Equation 1 would require $O(N_h^2 \cdot MaxP_h)$ time, with N_h being the cardinality of CST_h. To reduce the computational overhead, we present an equivalent, but more efficient (linear), algorithm. For a semantic tag $st_j = n_0/n_1/\ldots/n_k$, whether st_j is a candidate or not, let us say that the *count* cnt_j of st_j is the number of candidates $st_i \in CST_h$ that contain st_j as a prefix (i.e., of which st_j is an *ancestor*): $cnt_j = \sharp$*candidate semantic tags* st_i *of type* $n_0/\ldots/n_k/\ldots/n_p, p \geq k$.

Figure 4 (a) shows an example. For instance, the candidate `a/b/d` has frequency 5 (as given) and count 3, since the number of candidates whose prefix is `a/b/d` is 3, i.e., `a/b/d/g`, `a/b/d/h`, and `a/b/d` itself.

[4] WordNet: `http://wordnet.princeton.edu`.



st_j	$freq_j$
a/b/d/g	3
a/b/d	5
a/b/d/h	2
a/b/e	4
a/c/f/i	2

Figure 4: Candidate tree example (a). Blank circles denote candidate semantic tags (e.g., the one labelled `d` corresponds to the candidate semantic tag `a/b/d`). Close to each candidate st_j, the pair $(freq_j, util_j)$ is shown ($util_j$ is initially undefined), whereas count values $[cnt_i]$ are shown for each node n_i. The tree is completed in (b) with the utility values of the candidates. For clarity of exposition, in this figure we do not normalize utility values by $MaxP_h$.

Theorem 1 *The utility $util_j$ of the candidate semantic tag $st_j = n_0/n_1/\ldots/n_k$ can be computed as:*

$$util_j = \frac{\sum_{l=0}^{k} cnt_l - len(st_j)}{MaxP_h} = \frac{\sum_{l=0}^{k}(cnt_l - 1)}{MaxP_h} \qquad (2)$$

where cnt_l is the count of the semantic tag $n_0/n_1/\ldots/n_l$, ancestor of st_j.

Proof. By grouping together the contribution of all candidate semantic tags st_i for which the value of $len(st_j \cap st_i)$ is the same, Equation 1 can be rewritten as:

$$util_j = \frac{\sum_{l=0}^{k}(l+1) \cdot ov_{l+1}}{MaxP_h}$$

in which ov_{l+1} is the number of semantic tags that share with st_j a prefix path of length exactly equal to $l+1$. From the definition of cnt_l it is also derived that:

$$cnt_l = 1 + \sum_{p=l}^{k} ov_{p+1}$$

since every candidate st_i for which $len(st_j \cap st_i) \geq l+1$ contributes 1 to cnt_l, and the 1 term accounts for the fact that cnt_l also counts st_j itself. By substituting in Equation 2 it is obtained:

$$util_j = \frac{\sum_{l=0}^{k}\sum_{p=l}^{k} ov_{p+1}}{MaxP_h} = \frac{\sum_{l=0}^{k}(l+1) \cdot ov_{l+1}}{MaxP_h}$$

□

Figure 4 (b) completes the example of Figure 4 (a) showing the utility values of all candidates. For instance, the utility of the semantic tag `a/b/d/g` is $((5+4+3+1) -$ len(`a/b/d/g`)$)/MaxP_h = (13-4)/MaxP_h = 9/MaxP_h$. The same result is obtained from Eq. 1, which would compute the utility as (len(`a/b/d/g` ∩ `a/b/d`) + len(`a/b/d/g` ∩ `a/b/d/h`)+len(`a/b/d/g`∩`a/b/e`)+len(`a/b/d/h`∩`a/c/f/i`)$/MaxP_h$ $= (3+3+2+1)/MaxP_h = 9/MaxP_h$.

The utilities of all candidates in CST_h can be computed in $O(N_h \cdot MaxP_h)$ time if counts are available. Counts are incrementally obtained while generating the candidate tree CT_h, by adding 1 to the count of a semantic tag st_l whenever a new candidate st_j of which st_l is an ancestor is added to CT_h, as detailed in Algorithm 2.

3.2 The Ostia Keyword Search Algorithm

The traditional image keyword search (KWS) approach is described in Figure 5 (a): Given a set of keywords K, the set of *relevant images*, *RI*, i.e., those images sharing at least one keyword with K, is determined. Images in *RI* are then ranked and the top-*ki* ones returned. In this paper, we assume a simple ranking schema based on how many keywords of K an image has. Alternative ranking approaches (e.g., TF*IDF, number of edges, size normalization) can be applied [17, 21].

Algorithm 2 Ostia: Optimal Set of Semantic Tags for Q

```
Input: $CST$: candidate semantic tags, $m$: integer
Output: $ST$: top-$m$ predicted semantic tags for $Q$
for all $D_h$ do
    $CT_h \leftarrow \emptyset$;
    while $\exists$ a candidate semantic tag $st_j \in CST_h$ do
        addCandidateTagToTree($(st_j, freq_j)$, $CT_h$);
        for all $n_i \in st_j = n_0/n_1/\ldots/n_k$ do
            if $n_i$ is a newly added node in $CT_h$ then
                $cnt_i \leftarrow 1$
            else $cnt_i \leftarrow cnt_i + 1$;
    computeUtilities($CT_h$);          ▷ utilities of candidates
    computeWeights($CT_h$, $CST_h$);   ▷ weights of cand.'s
    $ST_h \leftarrow$ Top($CST_h, m$);
            ▷ optimal set of semantic tags for dimension $D_h$
return $ST \leftarrow$ Top($\bigcup_{h=1}^{M} ST_h, m$).
```

Algorithm Ostia-KWS, illustrated in Figure 5 (b) and detailed in Algorithm 3, approaches the image keyword search problem by exploiting the ability of the Ostia annotator in predicting high-quality semantic tags, with the purpose of improving the overall quality of the results.

More in detail, Ostia-KWS first applies Ostia to the query keywords K, thus obtaining a set ST of top-m semantic tags for K. The set *RI* of relevant images is then defined as consisting of those images that are annotated with at least one of the semantic tags in ST. Finally, images in *RI* are ranked and the best *ki* ones returned. For homogeneity with KWS, even for Ostia-KWS a simple ranking schema based on counting the number of semantic tags $st_j \in ST$ associated to each image is applied.

4. EXPERIMENTAL RESULTS

We have implemented Ostia, Ostia-KWS and KWS algorithms within our Scenique system, which makes use of the Windsurf library[5] for low-level feature management (e.g., image segmentation and support for k-NN queries, see [3]

[5] Windsurf: http://www-db.deis.unibo.it/Windsurf/

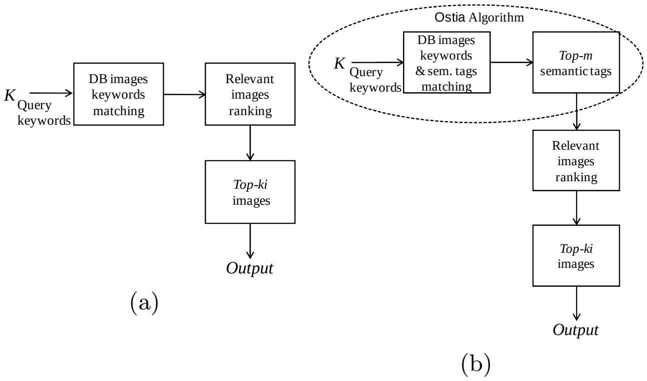


Figure 5: Illustration of KWS (a) and Ostia-KWS (b) search algorithm.

Algorithm 3 Ostia-based Keyword Search (Ostia-KWS)

```
Input: $K$: query, $\mathcal{DB}$: image database, $e, p, m, ki$: integer
Output: $\mathcal{I}$: top-$ki$ most relevant images for $K$
$RI \leftarrow \emptyset$;
$ST \leftarrow \mathsf{Ostia}_{Optimal}(\mathsf{Ostia}_{Candidate}(K, \mathcal{DB}, e, p), m)$;
        ▷ compute the set of optimal semantic tags
for all $st_j \in ST$ do
    $RI \leftarrow RI \cup \{I_i : st_j \in ST_i\}$;
        ▷ compute the set of relevant images
rankImages($RI$);        ▷ rank relevant images
return $\mathcal{I} \leftarrow$ Top($RI, ki$).        ▷ most relevant images
```

for more details). For experiments, we used real datasets of about 10,000 and 100,000 images extracted from the CoPhIR collection [5].[6] For the dimensions, we imported portions of open-access ontologies from Swoogle,[7] for a total of 15 dimensions.

Experiment 1: The aim of our first experiment is to measure the accuracy of Ostia in terms of classical precision (i.e., % of relevant predicted semantic tags) and recall (i.e., % of relevant predicted term with respect to those in the ground truth) metrics. The workload consisted of 50 randomly chosen images to be annotated. Each of such images was also assigned a set of semantic tags (3, on the average) by a set of volunteers so as to obtain a ground truth to evaluate the effectiveness of Ostia. The experiment was performed in the worst-case scenario, where each image $I_i \in \mathcal{DB}$ has no semantic tag at all, i.e., $ST_i = \emptyset$.

Figure 6 shows a visual result of Ostia for the picture Q_{912} with associated keywords $K_{912} = \{\texttt{photo}, \texttt{shangai}\}$. As we can observe, the predicted semantic tags (pointed by arrows in the figure), are all relevant for the image. Note that none of them contains keywords in K_{912}.

Figure 7 shows the annotation accuracy of Ostia in terms of precision and recall when varying the number of predicted semantic tags m for the dataset of 10,000 and 100,000 images. It can be observed that Ostia reaches higher levels of precision for the larger dataset: This is expected because, with more images in the database, it is easier to find images that are similar to the query from both the points of view of low-level features and keywords. The precision of Ostia is high for low values of m (about 85% when $m = 1$) and is maintained even for higher m values, by guaranteeing, at the same time, a good level of recall (around 70% for $m \in [6, 12]$).

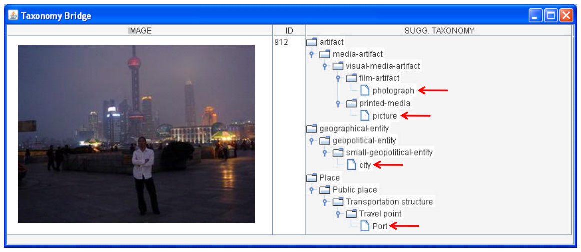


IMAGE	ID	SUGG. TAXONOMY
	912	artifact

- artifact
  - media-artifact
    - visual-media-artifact
      - film-artifact
        - photograph
      - printed-media
        - picture
- geographical-entity
  - geopolitical-entity
    - small-geopolitical-entity
      - city
- Place
  - Public place
    - Transportation structure
      - Travel point
        - Port

Figure 6: A visual example of Ostia in action.

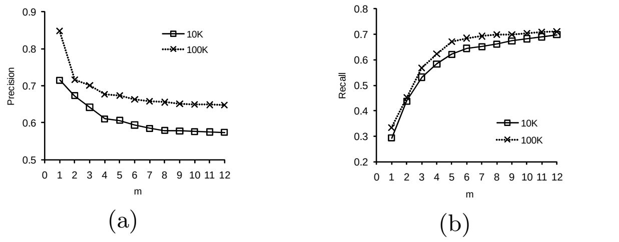


Figure 7: Annotation precision (a) and recall (b) vs. no. of predicted semantic tags.

Experiment 2: In this second experiment, our goal is to compare KWS and Ostia-KWS in terms of retrieval precision. The experiment was performed after automatically annotating only 100 images with $m = 3$ semantic tags per image. Results are averaged over 10 keyword queries, each with either 1 or 2 keywords.

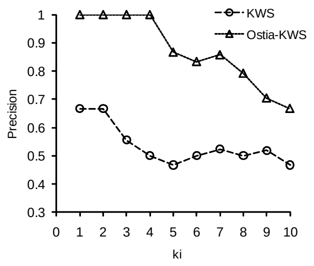


Figure 8: Precision vs no. of retrieved images.

As Figure 8 makes it evident, the precision of Ostia-KWS is consistently higher than that obtainable from KWS. This behavior is essentially due to the good quality of Ostia annotations (see Figure 7), which increases the probability that an image annotated with one of the semantic tags obtained from Ostia-KWS is also relevant to the keyword query.

5. RELATED WORK

The annotation process can be completely manual (this is the case for social Web services like Yahoo's Flickr) or take the advantage from some background knowledge, including the file name, the title, and the surrounding text. Systems that fit this scenario include the image search extensions of Google and Yahoo, which take into account the Web context

[6] The smaller dataset was obtained by sampling 10,000 images from the larger one.

[7] Swoogle: http://swoogle.umbc.edu/

of images to infer their relevance. On the other hand, approaches to automatic image annotation are typically based on machine learning techniques, that are used to train a set of concept classifiers [13, 7]. The limit of this approach is that it requires a new classifier to be built from scratch whenever a new class/concept is needed. On the other hand, Ostia does not require a learning phase, thus concepts can be freely added. Among solutions which uses both visual features and text annotations without pre-defined classes, [11] exploits the query visual features and its *geotags* to derive a set of similar images in the database from which, by means of a frequency-based procedure, geographically relevant tags are predicted. A similar approach is followed in [1], even if not restricted to the geographical case. [7] adds the use of Wordnet to prune uncorrelated tags. However, all these approaches predict free tags only, rather than concepts in a taxonomy as Ostia does.

Image annotations aims to enable keyword-based search techniques to be applied to image collections. In this context, traditionally an image constitutes one unit of information and is considered a result of a query if it contains a subset of the query's keywords (this is the case for Yahoo's Flickr and the image search extensions of Google and Yahoo). Recently, keyword search over structured and semi-structured data has been extensively investigated, since it allows data to be retrieved in a simple way and without the knowledge of the data schema and complex structured languages, such as SQL [6, 14]. The research results include retrieval and ranking approaches [17, 21], which exploit the structure of the data in order to provide relevant objects, and the interpretation of input keywords in order to deal with the ambiguity problem of keyword search [9, 19, 8]. With respect to image retrieval systems in literature, which are based on free tags, our approach maintains the advantage offered by the query paradigm of keyword search and tackles the keywords disambiguation problem by means of the notion of semantic tags.

6. CONCLUSIONS

In this paper we introduced a novel approach for keyword search on image collections based on multi-dimensional semantic tags. The core of the approach is the Ostia algorithm that takes the advantages of both visual features and keywords in order to predict for an image a high-quality set of concepts, here called semantic tags, taken from "light-weight" ontologies. Ostia can work in a focused way, i.e., predicting semantics tags only for a subset of user-specified dimensions. Further, it can work in an incremental way, i.e., by predicting semantic tags for an image with semantic tags (e.g., because a new dimension has been added). Unlike traditional keyword search, the Ostia-KWS algorithm first predicts the top-m semantic tags for the input keywords and then derives from such semantic tags the most relevant images. Preliminary results on real data demonstrate that our approach can be highly effective.

Future work will deal with the problem of exploiting the hierarchical nature of dimensions and of Wordnet concepts to improve the search of correct synonyms for a given keyword, and of reasoning on the correlation of predicted semantic tags. Further, we plan to investigate alternative ranking schemas and to study the correlation between the number of keywords in a query and that of semantic tags to predict for them. Finally, since semantic tags naturally lead to a hierarchical clustering of query results, we intend to inspect effective alternatives for the presentation of the results [12, 21].

7. REFERENCES

[1] I. Bartolini and P. Ciaccia. Imagination: Accurate Image Annotation Using Link-analysis Techniques. In AMR 2007, pages 32–44.

[2] I. Bartolini. Multi-faceted Browsing Interface for Digital Photo Collections. In CBMI 2009, pages 65–72.

[3] I. Bartolini, P. Ciaccia, and M. Patella. Query Processing Issues in Region-based Image Databases. *Knowledge and Information Systems*, 2010. To appear.

[4] M. Bates. Models of Natural Language Understanding. *National Academy of Sciences of the U.S.A*, 92(22):9977–9982, 1995.

[5] P. Bolettieri, A. Esuli, F. Falchi, C. Lucchese, R. Perego, T. Piccioli, and F. Rabitti. CoPhIR: a Test Collection for Content-Based Image Retrieval. *CoRR*, abs/0905.4627v2, 2009.

[6] Y. Chen, W. Wang, Z. Liu, and X. Lin. Keyword Search on Structured and Semi-structured Data. In SIGMOD 2009, pages 1005–1010.

[7] R. Datta, W. Ge, J. Li, and J. Z. Wang. Toward Bridging the Annotation-Retrieval Gap in Image Search. *IEEE MultiMedia*, 14(3):24–35, 2007.

[8] E. Demidova, I. Oelze, and P. Fankhauser. Do We Mean the Same? Disambiguation of Extracted Keyword Queries for Database Search. In KEYS 2009, pages 33–38.

[9] V. Hristidis, N. Koudas, Y. Papakonstantinou, and D. Srivastava. Keyword Proximity Search in XML Trees. *IEEE Trans. Knowl. Data Eng.*, 18(4):525–539, 2006.

[10] Y. Jin, L. Khan, L. Wang, and M. Awad. Image Annotations by Combining Multiple Evidence & WordNet. In MM 2005, pages 706–715.

[11] J. Kleban, E. Moxley, J. Xu, , and B. S. Manjunath. Global Annotation of Georeferenced Photographs. *ACM Conference on Image and Video Retrieval*, 2009.

[12] G. Koutrika, Z. M. Zadeh, and H. Garcia-Molina. DataClouds: Summarizing Keyword Search Results over Structured Data. In EDBT 2009, pages 391–402.

[13] J. Li and J. Z. Wang. Real-time Computerized Annotation of Pictures. In MM 2006, pages 911–920.

[14] A. Markowetz, Y. Yang, and D. Papadias. Keyword Search over Relational Tables and Streams. *ACM Trans. Database Syst.*, 34(3), 2009.

[15] A. Payne and S. Singh. A Benchmark for Indoor/Outdoor Scene Classification. In ICAPR 2005, pages 711–718.

[16] A. Penta, A. Picariello, and L. Tanca. Multimedia Knowledge Management using Ontologies. In MS 2008, pages 24–31.

[17] M. Sayyadian, H. LeKhac, A. Doan, and L. Gravano. Efficient Keyword Search Across Heterogeneous Relational Databases. In ICDE 2007, pages 346–355.

[18] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-Based Image Retrieval at the End of the Early Years. *IEEE TPAMI*, 22(12):1349–1380, 2000.

[19] Y. Tao and J. X. Yu. Finding Frequent Co-occurring Terms in Relational Keyword Search. In EDBT 2009, pages 839–850.

[20] R. Tye, G. Nathaniel, and N. Mor. Towards Automatic Extraction of Event and Place Semantics from Flickr Tags. In SIGIR 2007, pages 103–110.

[21] S. Wang, Z. Peng, J. Zhang, L. Qin, S. Wang, J. X. Yu, and B. Ding. NUITS: A Novel User Interface for Efficient Keyword Search over Databases. In VLDB 2006, pages 1143–1146.

[22] K.-P. Yee, K. Swearingen, K. Li, and M. A. Hearst. Faceted Metadata for Image Search and Browsing. In CHI 2003, pages 401–408.