Object-based Tag Propagation for Semi-Automatic Annotation of Images

Ivan Ivanov, Peter Vajda, Lutz Goldmann, Jong-Seok Lee, Touradj Ebrahimi
Multimedia Signal Processing Group – MMSPG
Institute of Electrical Engineering – IEL
Ecole Polytechnique Fédérale de Lausanne – EPFL
CH-1015 Lausanne, Switzerland
{ivan.ivanov, peter.vajda, lutz.goldmann, jong-seok.lee, touradj.ebrahimi}@epfl.ch

ABSTRACT

Over the last few years, social network systems have greatly increased users' involvement in online content creation and annotation. Since such systems usually need to deal with a large amount of multimedia data, it becomes desirable to realize an interactive service that minimizes tedious and time-consuming manual annotation. In this paper, we propose an interactive online platform that is capable of performing semi-automatic image annotation and tag recommendation for an extensive online database. First, when the user marks a specific object in an image, the system performs an object duplicate detection and returns the search results with images containing similar objects. Then, the annotation of the object can be performed in two ways: (1) In the tag recommendation process, the system recommends tags associated with the object in images of the search results, among which, the user can accept some tags for the object in the given image. (2) In the tag propagation process, when the user enters his/her tag for the object, it is propagated to images in the search results. Different techniques to speed-up the process of indexing and retrieval are presented in this paper and their effectiveness demonstrated through a set of experiments considering various classes of objects.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services—*Data sharing, Web-based services*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*

General Terms

Algorithms, Experimentation, Performance

Keywords

social networks, tag propagation, image annotation, tag recommendation, object duplicate detection

MIR'10, March 29–31, 2010, Philadelphia, Pennsylvania, USA.

1. INTRODUCTION

In recent years, social networks, digital photography and web-based personal image collections have gained popularity. A social network service typically focuses on building online communities of people who share interests and activities, or are interested in exploring the interests and activities of others. At the same time, they have become a popular way to share and to disseminate information, creating new challenges for access, search and retrieval. For example, users upload their personal photos and share them through online communities, asking other people to comment or rate them. This trend has resulted in a continuously growing volume of publicly available photos on multimedia sharing websites like Flickr[1], or social networks like FaceBook[2]. For instance, Flickr contains over 3.6 billion photos [2], and every month more than 2 billion photos are uploaded to FaceBook [1].

In these environments, photos are usually accompanied with metadata, such as comments, ratings, information about the uploader and their social network. Moreover, a recent trend is also to "tag" them. Tags are short textual annotations used to describe photos in order to provide meaningful information about them.

The most popular tags in photo sharing sites such as Flickr are usually related to the location where the photo was taken (e.g., San Francisco or France), the objects/persons appearing in the photo (e.g., baby, car or house), or the event/time when the photo was taken (e.g., wedding or summer) [3, 17].

Annotations and their association with images provide a powerful cue for their grouping and indexing. This cue is also essential for image retrieval systems to work in practice. The current state-of-the-art in content-based image retrieval systems has not yet delivered widely accepted solutions, except for some very narrow application domains, mainly because of the semantic gap problem, i.e., it is hard to extract semantically meaningful information using just low-level features [19]. In social networks, the success of Flickr and FaceBook proves that users are willing to provide this semantic context ("subjective" users' impressions) through manual annotations [17], which can help to bridge the semantic gap, and therefore improve the results of visual content search engines. Different users who annotate the same photo can provide different annotations, which is one more advantage of these systems.

However, tagging a lot of photos by hand is a time-consuming task. Users typically tag a small number of shared photos

[1] http://www.flickr.com
[2] http://www.facebook.com

only, leaving most of them with incomplete metadata. This lack of metadata seriously impairs search, as photos without proper annotations are typically much harder to find. Therefore, robust and efficient algorithms for automatic or semi-automatic tagging (or tag propagation) are desirable to help people organize and browse large collections of personal photos in an efficient way.

The main novelty of the paper comes from the application which realizes an interactive service that minimizes the users' tedious and time-consuming manual annotation process, and the evaluation of the object duplicate detection part of the system. We propose an interactive online platform which is capable of performing semi-automatic image annotation and tag recommendation for an extensive online database of images containing various object classes. Since the most salient regions in images usually correspond to specific objects, we consider object-based tagging within the system. First, when the user marks a specific object in an image, the system performs an object duplicate detection and returns the search results with images containing similar objects. Then, the annotation of the object can be performed in two ways, i.e., tag recommendation and tag propagation. In the tag recommendation mode, the system recommends tags for the object within the query image. The corresponding tags of all matched objects within the retrieved images are shown to the user who can then select appropriate tags among them. In the tag propagation mode, when the user enters his/her tag for the object, it is propagated to other images containing similar objects.

The remaining sections of this paper are organized as follows. We introduce related work in Section 2. Section 3 describes our approach for the interactive online platform and discusses the two modes, tag recommendation and tag propagation. Experiments and results are discussed in Section 4. Finally, Section 5 concludes the paper with a summary and some perspectives for future work.

2. RELATED WORK

The proposed system is related to different research fields including visual content analysis, social networking and tagging. The goal of this section is to review the most relevant works on human tagging and various approaches for tag recommendation.

Tagging images is a very time consuming process and tagging objects within images even more. Therefore, it is necessary to understand and increase the motivation of users to annotate images. Ames and Naaman [5] have explored different factors that motivate people to tag photos in mobile and online environments. One way is to decrease the complexity of the tagging process through tag recommendation which derives a set of possible tags from which the user can select suitable ones. Another way is to provide incentives for the user in form of entertainment or rewards. The most famous examples are the ESP Game and Peekaboom, developed for collecting information about image content. The *ESP Game* [21] randomly matches two players who are not allowed to communicate with each other. They are shown the same image and asked to enter a textual label that describes it. The aim is to enter the same word as your partner in the shortest possible time. *Peekaboom* [22] takes the ESP Game to the next level. Unlike the ESP Game, it's asymmetrical. To start, one player is shown an image and the other sees an empty black space. The first user is given a word related to the image, and the aim is to communicate that word to the other player by revealing portions of the image. Peekaboom improves on the data collected by the ESP Game and for each object in the image determines precise location information. Unlike these games, *LabelMe* is a web-based tool that allows easy image annotation and sharing of such annotations [16]. Using this tool, a large variety of annotations are collected spanning many object categories (cars, people, buildings, animals, tools, etc.).

However, manually tagging a large number of photos is still a tedious and time-consuming task. Thus automatic image annotation has received a lot of attention recently. Automatic image annotation is a challenging task which has not been solved in a satisfactory fashion for real-world applications. Most of the solutions are developed for a specific application and usually consider only one tag type, e.g., faces, locations, or events.

Berg *et al.* [8] propose an approach to label *people*, i.e. assign names to faces within newspapers (images with captions). They cluster face images visually in appropriate discriminant coordinates and apply natural language processing techniques to the caption to check whether the person whose name appears in a caption is depicted in the associated image or not. *Picasa*[3] also provides a service for name tagging which automatically finds similar faces in a photo collection.

Annotating images with geographical information such as landmarks and locations is a topic which has recently gained increasing attention. Ahern *et al.* [4] have developed a mobile application called *ZoneTag*[4] which enables the upload of context-aware photos from mobile phones equipped with a camera to Flickr. In addition to automatically supplying the location metadata for each photo (provided by a GPS device or mobile phone), ZoneTag supports context-based tag suggestions in which tags are provided from different sources including past tags from the user, the user's social network, and external geo-referenced data sources like Yahoo! Local, and Upcoming.org. Through the combination of textual information with vision based methods efficient systems for tag recommendation and propagation can be developed. Kucuktunc *et al.* [10] incorporate visual and textual cues of semantically relevant photos to recommend tags and to improve the quality of the suggested tags. First, the system requires the user to add a few initial tags for each uploaded photo. These initial tags are used to retrieve related photos which contain other tags besides the initial ones. Then the set of candidate tags collected from a pool of images is weighted according to the similarity of the target photo to the retrieved photo. Similarity is based on visual features including color histograms and SIFT features. Finally, a scoring function is applied to the list of candidate tags and the tags with the highest scores are used to automatically expand the tags of the target photo. This approach has its limitations, since users typically do not provide initial tags for each uploaded photo, but rather tag only a small number of shared photos.

Another application that combines textual and visual techniques has been proposed by Quack *et al.* [15]. They developed a system that crawls photo collections on the internet to identify clusters of images referring to a common object

[3] `http://picasa.google.com`

[4] `http://zonetag.research.yahoo.com`

(physical items on fixed locations), and events (special social occasions taking place at certain times). The clusters are created based on the pair-wise visual similarities between the images, and the metadata of the clustered photos is used to derive a label for the clusters. Finally, Wikipedia[5] articles are attached to the images and the validity of these associations is checked. Gammeter *et al.* [9] extends this idea towards object-based auto-annotation of holiday photos in a large database that includes landmark buildings, statues, scenes, pieces of art, with help of external resources such as Wikipedia. In both papers, [9] and [15], GPS coordinates are used to pre-cluster objects which limit classes of objects to landmarks and buildings. Another limitation is that GPS coordinates may not be always available. In contrast, our work considers extensive online database of various object classes, such as landmark buildings, cars, cover or text pages of newspapers, shoes, trademarks and different gadgets like mobile phones, cameras, watches.

Lindstaedt *et al.* [11] developed *tagr* a tag recommendation system for pictures which depict fruits and vegetables. They combine three types of information: visual content, text and user context. At first, they group annotated images into classes using global color and texture features. The user defined annotations are then linked with the images. The resulting set of tags for visually similar images is then extended with synonyms derived from WordNet. When the user uploads an untagged image, it is assigned to one of the classes and corresponding tags are recommended to the user. In addition, this system analyzes tags which the user assigns to the images and returns the profiles of users with similar tagging preferences. This method has been proven to be effective to recommend tags for a set of selected fruits and vegetables, but it cannot be applied to other classes of objects, which limits its applicability. Other approaches for automatic image annotation consider only the context. Sigurbjörnsson and van Zwol [17] developed a system which recommends a set of tags based on collective knowledge extracted from Flickr. Given a photo with user-defined tags, a new list of candidate tags is derived for each of the user-defined tags, based on tag co-occurrence. The lists of candidate tags are aggregated, tags are ranked, and a new set of recommended tags is provided to the user.

The tag propagation and tag recommendation are nowadays very important in environment such as social network, since they provide efficient information for grouping or retrieving images. The system proposed in this paper provides these functionalities in an interactive way. The novelty is that image annotation is performed at the object level by making use of content based processing. It does not consider context, such as text or GPS coordinates, which may limit its applicability. This approach is suitable for all kinds of objects, such as trademarks, books, newspapers, and not just buildings or landmarks.

3. SYSTEM OVERVIEW

In this section, we present our method for object-based tag recommendation and propagation. Since the manual annotation of all the instances of an object within a large set of images is very time consuming, the system offers tag propagation of marked and tagged objects. Image annotation is performed at the object level, outlining the object with a bounding-box. The system architecture is illustrated in Figure 1.

[5]http://www.wikipedia.org

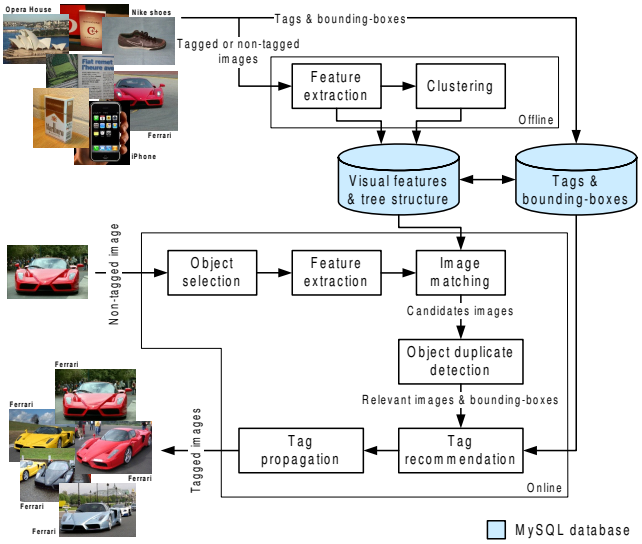

Figure 1: Overview of the system for semi-automatic annotation of objects in images.

3.1 Offline part

The goal of the offline processing is to preprocess uploaded images in order to allow efficient and interactive object tagging. It starts by describing each image with a set of sparse local features. In order to speed up the feature matching, the features of all images are grouped hierarchically into a tree representation.

3.1.1 Feature extraction

For a robust and efficient object localization sparse local features are adopted to describe the image content. Salient regions are detected using the Fast-Hessian detector [7] which is based on approximation of Hessian matrix detector. The position and scale are computed for each of the regions and will be used for the object duplicate detection (described in Section 3.2.3).

The detected regions are described using Speeded Up Robust Features (SURF) [7], which can be extracted very efficiently and are robust to arbitrary changes in viewpoints. The goal of SURF is to approximate the popular and robust features based on the Scale Invariant Feature Transform (SIFT) [12].

3.1.2 Clustering

For the object tagging, features of a selected object have to be matched against all the images in the database. Therefore a fast matching algorithm is required to ensure interactivity of the application. Hierarchical clustering is applied to group the features according to their similarity. This improves the efficiency of the feature matching since a fast approximation of the nearest neighbor search can be used.

Hierarchical k-means clustering is used to derive the vocabulary tree, similar to the one described in [13]. Within the tree, parent nodes correspond to the cluster centers derived from the features of all its children nodes and leaf nodes correspond to the real features within the images. The clus-

tering leads to a balanced tree with a similar depth for all the leaves.

Since the importance of the individual visual words (i.e., nodes in the tree structure) may differ among the images in the database, weights w_i are assigned to each of the corresponding nodes i. These weights are equivalent to the inverse document frequency (IDF) commonly used in text retrieval which is defined as

$$w_i = \log\left(\frac{N}{N_i}\right) \quad (1)$$

where N is the number of images in the database and N_i is the number of images which have features in the subtree, if the i-th node is considered as a root of this subtree. The basic idea behind IDF is that the importance of a visual word is higher if it is contained in only a few images. Furthermore, the importance of a visual word i in relation to an individual image j is considered using the term frequency (TF) which is defined as

$$m_{ij} = \frac{N_{ij}}{\sum_k N_{kj}} \quad (2)$$

where N_{ij} is the number of occurrences of a visual word i within an image j and the denominator is the number of occurrences of all features within image j.

Given this TF-IDF weighting scheme the overall weight d_{ij} for a visual word i within an image j is given as

$$d_{ij} = m_{ij} \cdot w_i \quad (3)$$

which can be combined into a vector $\mathbf{d}_j$. This vector will be matched to the one extracted from the query image to compute the similarity within the image matching step described in Section 3.2.2.

The computational complexity of the complete offline phase is $O((n \cdot N \cdot log(n))^2)$, where n is the size of the image and N is the number of images, since the clustering, which is the most time consuming part of this method, uses limited number of iterations.

3.2 Online part

The goal of the online processing is to recommend a tag for an object in a given image and then to propagate this tag to other images containing the same object. The user marks a desired object in the image by selecting a bounding-box around it. The system performs image matching by making use of local features and selects a reduced set of candidate images which are most likely to contain the target object. The object duplicate detection is applied to detect and to localize the target object within the reduced set of images. The corresponding tags of all matched objects are shown to the user who can then select an appropriate tag among them. Once an object has been tagged the user can ask the system to propagate it automatically to other images within the database.

3.2.1 Object selection

The user can annotate any photo in the database, which is either uploaded by himself/herself or by any other user. Images are annotated on the object level. The database used in this work covers a wide range of different classes, which will be described in more details in Section 4.1. Once the user chooses a photo which he/she wants to annotate, the user is free to label as many objects depicted in the image as he/she chooses. The user interface used in this work is shown in Figure 2. By clicking on the button "add note", the user marks an object by selecting object's boundaries as a bounding-box. This process is commonly used in many photo sharing services, such as FaceBook. When a user enters the page with particular image from the dataset, tags which are previously entered by other users accompanied with the corresponding bounding-boxes, will already appear on the image. If there is a mistake in the annotation (either the outline or the text of the label is not correct), the user may either edit the label by renaming the object, redrawing along the object's boundary or deleting labels for the chosen image. Once the desired object is marked, the tag recommendation process can start.

3.2.2 Image matching

In order to speed up the object duplicate detection process image matching is used to select a reduced set of candidate images which are most likely to contain the target object. Since the more complex object duplicate detection is only applied to this reduced set, the overall speed is considerably increased. By making use of the local features, target images can be distinguished from non target images even if the target object is just a small part of it.

Given the local features within the selected region in the query image and the vocabulary tree, a weighting vector $\mathbf{q}$ is computed in the same way as the weighting vector $\mathbf{d}_j$ for image j, described in Section 3.1.2.

Based on these weighting vectors, the query image is matched to all the images j in the database and the individual matching scores s_j are computed in the same way as in [13]:

$$s_j = \|\mathbf{q} - \mathbf{d}_j\| = 2 - 2 \cdot \sum_{\forall i: q_i \neq 0 \cap d_{ij} \neq 0} \frac{q_i \cdot d_{ij}}{\|\mathbf{q}\| \cdot \|\mathbf{d}_i\|}. \quad (4)$$

Image j whose scores s_j exceed a predefined threshold T_I are discarded and will not be considered for the object duplicate detection.

The complexity of the search step for similar images is $O(n \cdot log(n))$, where n is the size of the query image, as the feature extraction creates $O(n \cdot log(n))$ features by making use of pyramids for detection of scale invariant features [7].

3.2.3 Object duplicate detection

The goal of the object duplicate detection step is to detect and to localize the target object within the reduced set of images returned from the image matching step. The outcome of this step is a set of predicted objects described through their bounding-boxes for each of the images.

Local features are used for object duplicate detection in [12]. General Hough Transformation is then applied for object localization. Our object duplicates detection method is based on this algorithm and the detection accuracy is improved by using inverse document frequency. Inverse document frequency has been used for the similar purpose in [18]. Descriptors are extracted from local affine-invariant regions and quantized into visual words, reducing the noise sensitivity of the matching. Inverted files are then used to match the video frames to a query object and retrieve those which are likely to contain the same object. Different techniques for the object duplicate detection have been proposed in the literature. Vajda *et al.* [20] proposed to use sparse features which are robust to arbitrary changes in viewpoints. Spatial graph model matching is then applied to improve the

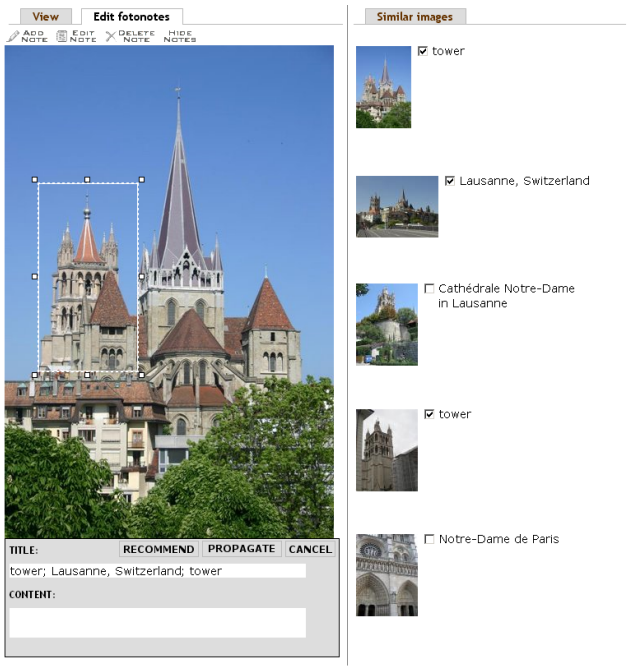

Figure 2: Screenshot of the web application during the tag recommendation step. Based on the selected object the system automatically proposes tags from which the user can select suitable ones.

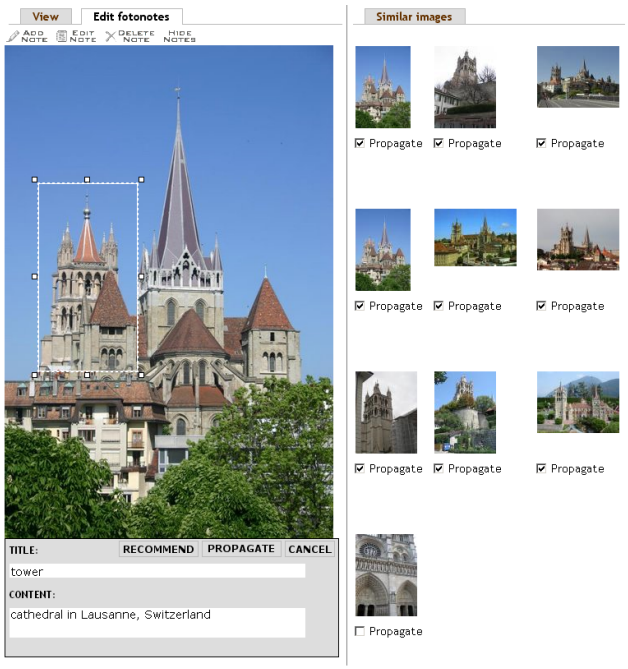

Figure 3: Screenshot of the web application during the tag propagation step. The system automatically propagates the tagged object to the images in the database and asks the user to verify the result.

accuracy of the detection, which considers the scale, orientation, position and neighborhood of the features. Philbin *et al.* [14] applied the Bag of Words method for detecting buildings in a large database. To resolve the problem of large database they use a forest of 8 randomized k-d trees as a data structure for storing and searching features.

The detection and localization starts by matching the features within the selected region of the query image to the features with the candidate image. Again the hierarchical vocabulary tree is used to speed up the nearest neighbor search. Matches whose distance is larger than a predefined threshold T_F are discarded.

In order to detect and to localize target objects based on these matched features, the general Hough transform [6] is applied. Each matched feature within the candidate image votes for the position (center) and the scale of a bounding-box based on the position and scale of the corresponding feature within the query image. Since unique features may provide a more reliable estimate of the bounding-box, the vote of a feature is equal to its inverse document frequency (IDF) value w_i already described in Section 3.1.2. This leads to a 3-dimensional histogram that describes the distribution of the votes across the bounding-box parameters (position, scale). To obtain the set of predicted objects the local maxima of the histogram are searched and thresholded with T_O.

The complexity of our method for object duplicate detection is $O(n \cdot log(n))$, where n is the size of the query image, since the SURF feature extraction uses image pyramids for detection of scale invariant features [7] and the general Hough transformation has the same computational complexity, since we do not consider rotated objects within the database.

3.2.4 *Tag recommendation*

The main idea of the tag recommendation step is to assist the user by suggesting probable tags for the marked object as shown in Figure 2. This is comparable to the autocomplete feature in text editors that suggests relevant words based on already typed characters.

After selecting an object within the current image the user can press the "recommend" button to ask for suitable tags for this object. The system tries to find duplicates of the selected object using the algorithms described in the previous sections. The bounding-boxes of the predicted objects are compared to those of already tagged objects. If there is more than 50 % overlap, objects are considered as a match. The corresponding tags of all matched objects are combined into a recommendation list which is displayed to the user in form of tags and associated thumbnails. Duplicate tags may appear in the recommendation list, when multiple images contain visually similar objects accompanied by the same tag, which can be seen in Figure 2.

The user has then the choice to select one or more of the recommended tags or to provide his own tags. At the end of this step, the bounding-box of the objects and the associated tags are stored in the database.

3.2.5 *Tag propagation*

Since the manual annotation of all the instances of an object within a large set of images is very time consuming, the system offers tag propagation of marked and tagged objects as shown in Figure 3. Thereby duplicates of the tagged object are detected within the database and the result is shown to the user.

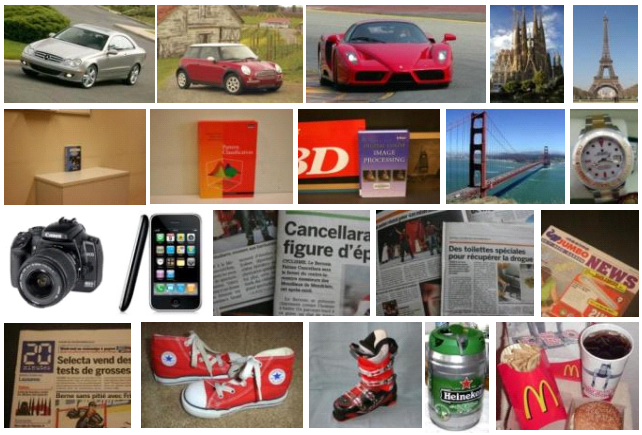

Figure 4: Samples from the 160 objects within the dataset.

Once an object has been marked and tagged, a user can ask the system to propagate it automatically to the other images in the database by pressing the "propagate" button. The system performs object duplicate detection in the way explained in previous sections, and returns images containing object duplicates. Considering matches between propagated and already tagged objects one has to distinguish two cases:

- If an object duplicate does not match any already tagged object both its bounding-box and tag can be automatically propagated to the corresponding image. However since the object duplicate detection may return a few non-relevant objects the user can verify the propagated tags.
- If an object duplicate matches an already tagged object the two bounding-boxes and sets of tags have to be merged. The system can either ask the user to resolve the conflict and merge the two objects, or this can be done automatically using some heuristics. Since manually tagged objects are usually more reliable than automatically propagated ones, the bounding-box of the object duplicate will be discarded but the tags will be combined.

4. EXPERIMENTS

In this section the performance of the proposed tag propagation and tag recommendation methods are evaluated and analyzed in two application scenarios. The considered dataset is described in Section 4.1. In Section 4.2 the evaluation is presented, and finally the results are discussed for both scenarios in Section 4.3.

4.1 Dataset

A new dataset was created in order to evaluate the tag propagation and tag recommendation methods. Part of the dataset is obtained from Google Image Search[6], Flickr and Wikipedia by querying the associated tags for different classes of objects. The rest of the dataset is formed by manually taking photos of particular objects using digital camera Canon EOS 400D.

[6]`http://images.google.com`

Table 1: Summary of the classes and some example objects

Classes	Example objects
Cars	BMW Mini Cooper, Citroen C1, Ferrari Enzo, Jeep Grand Cherokee, Lamborghini Diablo, Opel Ampera, Peugeot 206, Rolls Royce Phantom
Books	"Digital Color Image Processing", "Image Analysis and Mathematical Morphology", "JPEG2000", "Pattern Classification", "Speech Recognition"
Gadgets	Canon EOS 400D, iPhone, Nokia N97, Sony Playstation 3, Rolex Yacht-Master, Tissot Quadrato Chronograph
Buildings	Sagrada Familia (Barcelona), Brandenburg Gate (Berlin), Tower Bridge (London), Golden Gate Bridge (San Francisco), Eiffel Tower (Paris)
Newspapers	MobileZone, Le Matin Bleu, 20 Minutes, EPFL Flash
Text	Titles, paragraphs and image captions in newspapers
Shoes	Adidas Barricade, Atomic Ski Boot, Converse All Star Diego, Grubin Sandals, Merrell Moab, Puma Unlimited
Trademarks	Coca Cola, Guinness, Heineken, McDonald's, Starbucks, Walt Disney

The resulting dataset consists of 3200 images: 8 classes of objects, and 20 objects for each of them. For each object, 20 sample images are collected. Summary of the considered classes and some example objects are shown in Table 1. Figure 4 shows a single image for a single object from some of the 160 objects, while Figure 5 provides several images for 3 selected objects (e.g., Merrell Moab hiking shoes, Golden Gate Bridge, and Starbucks trademark).

As it can be seen, images with a large variety of view points and distances, as well as with different background environments, are considered for each object. The dataset is split into training and test subsets. Training images are chosen carefully so that they provide a frontal wide angle view of the objects depicted in images. Objects are selected using bounding-boxes. One sample image from each object

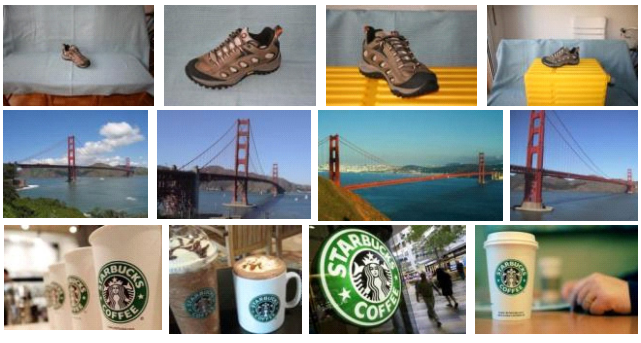

Figure 5: Selected objects for 3 different objects: Merrell Moab hiking shoe, Golden Gate Bridge (San Francisco), and Starbucks trademark.

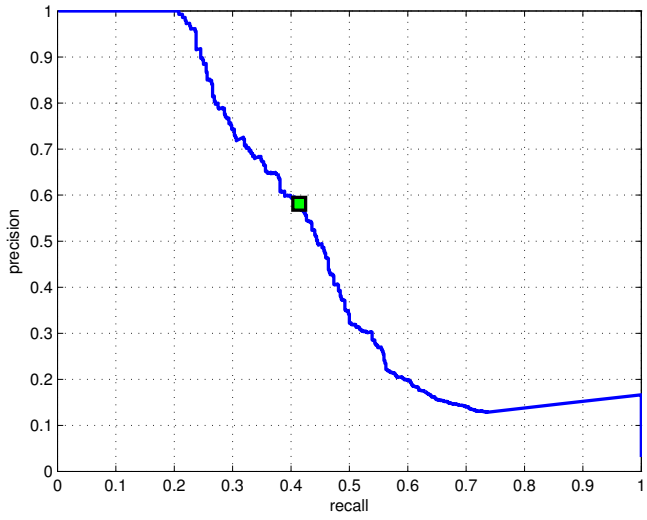

Figure 6: Performance of the object duplicate detection as precision vs. recall (PR) curve averaged over all the classes.

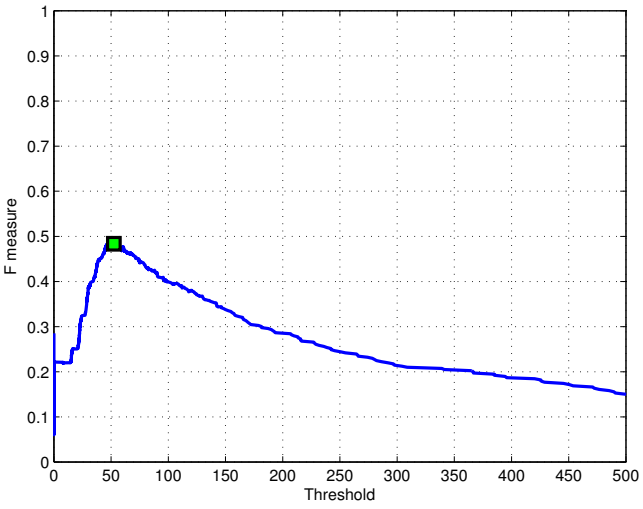

Figure 7: Performance of the object duplicate detection as average F-measure vs. object duplicate detection threshold T_O.

is chosen as a training image. All other images from the dataset are test images.

4.2 Evaluation

Since both the tag recommendation and tag propagation rely on the performance of the object duplicate detection only this part of the whole system has been assessed. It can be evaluated as a typical detection problem where the set of predicted objects is compared against a set of ground truth objects.

Objects are matched against each other based on the overlap of their bounding-boxes. If the ratio between the overlapping area and the overall area exceeds 50 % it is considered as a match. Based on that, a confusion matrix is computed, which contains the number of true positives (TP), false positives (FP) and false negatives (FN). For the evaluation precision-recall (PR) curves can be derived from this confusion matrix. PR curves plot the recall (R) versus the precision (P) with:

$$P = \frac{TP}{TP+FP}, \quad (5)$$

$$R = \frac{TP}{TP+FN}. \quad (6)$$

The F-measure is calculated to determine the optimum thresholds for the object duplicate detection. It can be computed as the harmonic mean of P and R values:

$$F = \frac{2 \cdot P \cdot R}{P+R}. \quad (7)$$

Thus, it considers precision and recall equally weighted.

4.3 Results

Figure 6 shows the performance of the object duplicate detection in form of the average PR curve computed over all the classes within the dataset. It provides a good visualization of the opposing effects (high precision versus high recall) which are inherent to any detection task. The results show that if both effects are considered with equal importance, the optimum is achieved at $R = 0.4$ and $P = 0.6$. However the precision can be greatly increased if $R = 0.2$ is considered enough for the tag recommendation and propagation.

In order to determine the optimal threshold for the object duplicate detection, the F-measures across the different thresholds has been computed. Figure 7 shows the threshold versus F-measure curve. The optimal threshold of 50 is chosen for the maximum F-measure of 0.49 and shown in all related figures by green markers. The final F-measure averaged over the whole dataset is 0.48. The tag recommendation will be more supported than the tag propagation because the propagation is more sensitive to the performance of the object duplicate detection.

In order to compare the different classes with each other, the F-measure is computed for each of the object classes as shown in Figure 8. Trademarks perform best, thanks to the large number of salient regions. In the case of text or cover pages of newspapers, books or gadgets, the proposed tag propagation scenario performs worse because the objects do not have enough discriminative features. Shiny or rotated objects, such as cars, shoes or buildings, are hard to detect due to the changing reflections and varying viewpoint.

5. CONCLUSION

Social networks are gaining popularity for sharing interests and information. Especially photo sharing and tagging is becoming increasingly popular. Among others, tags of people, locations, and objects provide efficient information for grouping or retrieving images. Since the manual annotation of these tags is quite time consuming automatic tag propagation based on visual similarity offers a very interesting solution.

In this work we have developed an efficient system for semi-automatic object tagging in images. After marking desired object in an image, the system performs object duplicate detection in the whole database and returns the search results with images containing similar objects. Then, the

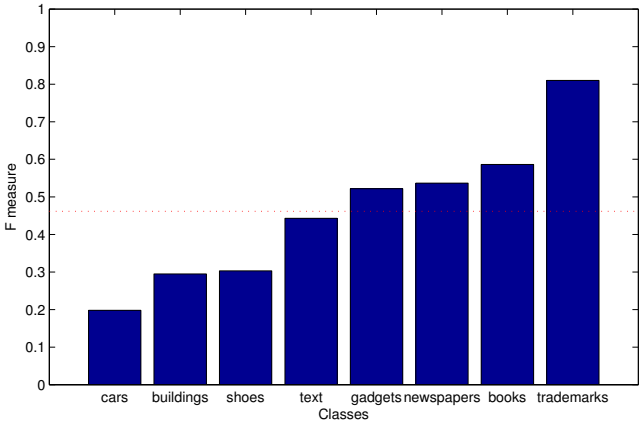

Figure 8: Performance of the object duplicate detection as F-measure across the different classes.

annotation can be performed through a tag recommendation process, in which the system recommends tags associated with the object in the images of the search results, or through tag propagation process, when the user enters his/her tag for the object and it is propagated to the images in the search results.

The performance of the system has been assessed by evaluating the performance of the object duplicated detection step, since both tag recommendation and propagation rely on its outcome. It has been shown that the detection works reliably for salient objects such as trademarks, books, newspapers, and gadgets.

Our semi-automatic tag propagation system has the potential to be improved in many ways. As a future study, we will extend it to support other classes of objects and consider evaluation of the system in the view point of the database size and latency in the system because it is important for the system to be interactive. Since our interactive system supports interaction between users, future work can also focus on modeling users' trust in such a manner that only tags from trusted users are finally propagated.

6. ACKNOWLEDGMENTS

This work was supported by the Swiss National Foundation for Scientific Research in the framework of NCCR Interactive Multimodal Information Management (IM2), the Swiss National Science Foundation Grant "Multimedia Security" (number 200020-113709), and partially supported by the European Network of Excellence PetaMedia (FP7/2007-2011).

References

[1] FaceBook Statistics. Available at `http://www.facebook.com/press/info.php?statistics`.

[2] Wikipedia – Flickr. Available at `http://en.wikipedia.org/wiki/Flickr`.

[3] Flickr – All time most popular tags. Available at `http://www.flickr.com/photos/tags/`.

[4] S. Ahern, S. King, M. Naaman, R. Nair, and J. H.-I. Yang. ZoneTag: Rich, community-supported context-aware media capture and annotation. In *Proceedings of the Mobile Spatial Interaction Workshop (MSI) at the SIGCHI Conference on Human Factors in Computing Systems (CHI 2007)*, 2007.

[5] M. Ames and M. Naaman. Why we tag: Motivations for annotation in mobile and online media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2007)*, pages 971–980, 2007.

[6] D. H. Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2): 111–122, 1981.

[7] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded-up robust features. In *Proceedings of the 9th European Conference on Computer Vision*, pages 404–417, 2006.

[8] T. L. Berg, A. C. Berg, J. Edwards, M. Maire, R. White, Y. W. Teh, E. Learned-Miller, and D. A. Forsyth. Names and faces in the news. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, pages 848–854, 2004.

[9] S. Gammeter, L. Bossard, T. Quack, and L. Van Gool. I know what you did last summer: object level auto-annotation of holiday snaps. In *Proceedings of the 20th International Conference on Computer Vision (ICCV 2009)*, 2009.

[10] O. Kucuktunc, S. Sevil, A. Tosun, H. Zitouni, P. Duygulu, and F. Can. Tag suggestr: Automatic photo tag expansion using visual information for photo sharing websites. *Springer Lecture Notes in Computer Science: Semantic Multimedia*, 5392/2008:61–73, 2008.

[11] S. Lindstaedt, V. Pammer, R. Mörzinger, R. Kern, H. Mülner, and C. Wagner. Recommending tags for pictures based on text, visual content and user context. In *Proceedings of the 3rd International Conference on Internet and Web Applications and Services (ICIW 2008)*, pages 506–511, 2008.

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[13] D. Nister and H. Stewenius. Robust scalable recognition with a vocabulary tree. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, pages 2161–2168, 2006.

[14] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, pages 1–8, 2007.

[15] T. Quack, B. Leibe, and L. Van Gool. World-scale mining of objects and events from community photo collections. In *Proceedings of the International Conference on Content-based Image and Video Retrieval (CIVR 2008)*, pages 47–56, 2008.

[16] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008.

[17] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th International Conference on World Wide Web (WWW 2008)*, pages 327–336, 2008.

[18] J. Sivic and A. Zisserman. Video Google: Efficient Visual Search of Videos. *Springer Lecture Notes in Computer Science: Toward Category-Level Object Recognition*, 4170/2006:127–144, 2006.

[19] 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 Transactions on Pattern Analysis and Machine Intelligence*, 22(12): 1349–1380, 2000.

[20] P. Vajda, F. Dufaux, T. H. Minh, and T. Ebrahimi. Graph-based approach for 3D object duplicate detection. In *Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2009)*, pages 254–257, 2009.

[21] L. von Ahn and L. Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2004)*, pages 319–326, 2004.

[22] L. von Ahn, R. Liu, and M. Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2006)*, pages 55–64, 2006.