Efficient and Rich Annotations for Large Photo Collections

Jayaguru Panda
jayaguru.panda@research.iiit.ac.in
IIIT Hyderabad, India

C V Jawahar
jawahar@iiit.ac.in
IIIT Hyderabad, India

***Abstract*—Large unstructured photo collections from Internet usually have distinguishable keyword tagging associated with the images. Photos from tourist and heritage sites can be described with detailed and part-wise annotations resulting in an improved automatic search and enhanced photo browsing experience. Manually annotating a large community photo collection is a costly and redundant process as similar images share the same annotations. We demonstrate an interactive web-based annotation tool that allows multiple users to add, view, edit and suggest rich annotations for images in community photo collections. Since, distinct annotations could be few, we have an easy and efficient batch annotation approach using an image similarity graph, pre-computed with instance retrieval and matching. This helps in seamlessly propagating annotations of the same objects or similar images across the entire dataset. We use a database of $20K$ images (Heritage-20K) taken from a world-famous heritage site to demonstrate and evaluate our annotation approach.**

I. Introduction

The widespread usage of digital photography along with the Internet boom has made available billions of photographs over the Internet, particularly on social networking platforms. The computer vision research community is busy in exploiting these large unstructured collections. Community photo collections from the Internet have been used for creating interactive 3D photo browsing tools [4], reconstructing dense 3D scenes using multi-view stereo [5], summarising scenes [6], segmenting scenes [7], hole-filling [8], learning object category models [9], estimating geo-location [10] and finding image paths [11]. Most of these efforts take advantages of the quantity, variety and distribution of photos in a collection. Another important dimension is the similarity of the objects and scenes within a photo collection. [12] demonstrates an object level auto-annotation framework for a community photo collection. We are interested in finer part-wise description at the object-level (See Figure 1).

Internet photos are tagged by certain distinguishing keywords, which helps their retrieval by search engines. A large percentage of community photos have interesting scenes and objects within the image which could be associated with more informative tags than just vague keywords. Many such photos may come from popular tourist and heritage sites around the world. With the help of detailed tags associated with each image, a better structure can induced into the unorganised photo collection. This helps the cause of search engines as they have a larger text description for an image to look up the query keywords. Also, self-explaining photos enhance the photo browsing experience of a user. Another use-case for a richly annotated photo collection could be instant on-the-spot auto-annotation applications such as [13], that perform on-device instance recognition for photo captured using the camera of a mobile phone. These applications aim to comprehensively educate the user about the scene and objects captured in the query image. When distinct objects in the training dataset are thoroughly annotated, important details associated with the matched image can describe and characterize the query.

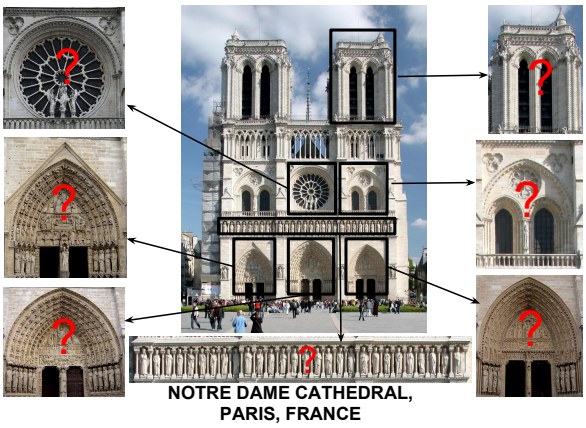

Figure 1. An Internet photo of a monument, tagged as "Notre Dame Cathedral" in "Paris, France" We are interested in details: its history, architecture, etc as well as descriptions of highlighted objects that are usually interesting to an enthusiastic tourist.

An image from a tourist site may pose several questions about the scene captured, the objects contained, the location specifics, the historical and/or archaeological significance, the architectural details, etc. All such necessary information can be associated with the images in the form of textual or audio/video annotations. There are two major challenges associated with annotating a large photo collection. Firstly, rich and precise information about objects in a photo is difficult to collect manually. Secondly, Internet photo collections may have sizes ranging millions of images. Annotating each photo one-by-one is a cumbersome task. However, the number of distinct scenes, buildings, or objects in the photos may be far lesser than the total collection i.e. many similar images may share the same annotations.

Figure 2. Scene instance annotations for a set of popular monuments.

We demonstrate a web-based community tool that facilitates an efficient batch annotation mechanism. Once a photo is annotated, the tool tries to identify all photos in the collection which are similar to this or, have the same object/scene and thus, propagate the annotation across the entire dataset.

II. Image Annotations

In this section, we discuss how to richly annotate a photo in a part-wise approach that stores finer object-level details.

Annotation Modes: We identify various types of annotations that could describe a photo captured from a tourist site: text, hyperlink, audio, video, localisation on a map of the tourist site, object boundaries and graphical illustrations in the form of arrows for direction purposes. While adding text annotations is straightforward, other form of annotations require a generalised framework. For instance, the scene depicted in the image can be marked on a map.

Scene and Object Instances: Scene instances are distinguished structures at a site which are of popular tourist and heritage interests. Annotations used to briefly describe the image scene may be called scene annotations (See Figure 2). However, when multiple distinct objects are captured in a single image, a single overall description may not always fit. Specific objects like an interesting artifact or structure occurring in a scene might have particular significance. It is useful to identify and localise such distinct objects within the image (See Figure 3). A rectangular boundary represents specific object regions, with distinct annotations.

Collaborative Sources: Multiple users like historians, researchers, students, tour guides or other third parties form a collaborative knowledge-based network to build rich useful annotations. While some have in-depth knowledge about the history of the site, we could also get useful annotations from enthusiastic tourists who wish to share their experience. The annotation system is designed to work in a client-server fashion, which allows users to add and modify annotations.

III. Efficient Batch Annotation Approach

In section II, we discuss the annotation structure for a photo. The idea is to have an efficient and seamless

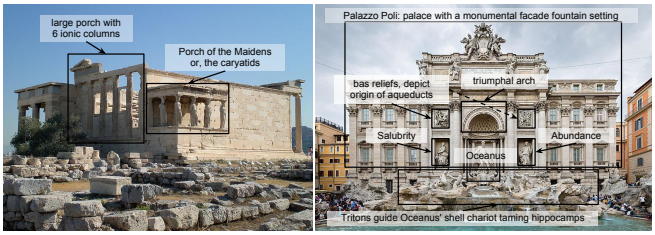

Figure 3. Object instance annotations for a couple of monuments.

annotation propagation approach to annotate a group of similar images simultaneously. Once a photo is annotated, we look for matching images and object regions and spread the annotation across the entire collection.

A. Image Similarity Graph

Image matching across datasets having millions of images is a highly expensive method as it involves an exhaustive pairwise matching step. However, during the online query process, pairwise matching of the query image with only a small subset of the entire dataset is necessary. This subset must contain all images from the dataset, that are similar to the query. An image similarity graph, mapping similarity and matching relations between photos in the collection, can be built to efficiently look-up large databases for a given annotated query photo. Each image in the similarity graph is a node and the edge between two similar images is weighted by the number of matching features. Further, the edges also store the pair-wise matches to speed-up the computations during online querying and matching the object regions.

Constructing the graph involves an offline step of computing similar images for each photo. This problem is related to the image retrieval problem, where a query image is given and the system returns a ranked list of similar images. We use the state-of-the-art instance retrieval approach (See section III-B) for this purpose. In the graph-based representation of a photo collection, each image has an adjacency list of other similar images from the collection. This allows the discovery of neighboring images from same scene (See Section III-C). Given a query, corresponding matches for the object regions are looked up (See Section III-D).

B. BoW Instance Retrieval for Similarity Relations

The similarity relations for each photo in a large dataset can be obtained by employing the Bag-of-Words (BoW) based instance retrieval approach [1], [3] to search similar images and objects with respect to a query image. This is followed by SIFT-based geometric verification of the retrieved images. The entire process, described in the following paragraphs, is applied to each image in the dataset to construct the Image Similarity Graph.

Given a set of images, local image feature descriptors like SIFT, SURF, etc are extracted at interest points of each image. These high-dimensional feature vectors are clustered

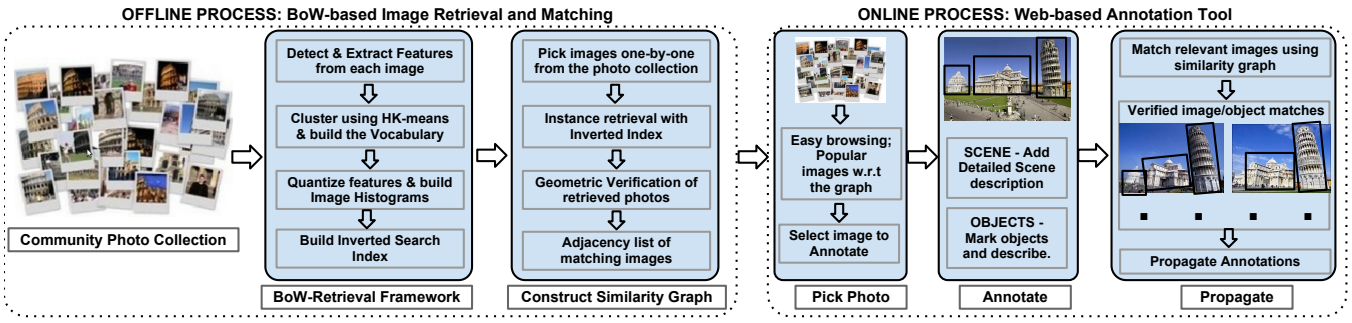

Figure 4. An overview of the annotation building framework. The offline process details the pipeline to construct an image similarity graph for a large photo collection. During the online process, we have a web-based annotation tool that allows easy browsing of the photos and selecting one to associate useful scene and object information. The annotations are then propagated across the collection.

using a clustering technique like K-Means to define a visual vocabulary for the image dataset. The features from each image are then, quantized and a histogram of visual word representation (also called BoW representation) is obtained for every image in the database. To speed-up the query process, an inverted search index is built, which maps the visual words to a posting list of images that contain them. During the retrieval process, the BoW representation is computed for the query image. For each visual word occurring in the query, a vote is given to all images that are mapped to it in the search index. A popular voting measure is the Term Frequency - Inverse Document Frequency (TF-IDF) based weighting of visual words. The database images are shortlisted and ranked based on these TF-IDF scores.

Once the similar images are efficiently retrieved, a filtering technique is needed to identify the false retrievals. For this purpose, geometric verification is used to remove errors due to outliers from mismatched or missing features, because of detector failure, occlusion, etc. The standard solution is to use the RANSAC algorithm [2], which involves generating affine transformation hypotheses using a minimal number of correspondences and then evaluating each hypothesis based on the number of "inliers" among all features. This helps in estimating a fundamental matrix fit between the query and each of the target retrievals. The verified retrievals are scored corresponding to the number the inlier matches. These pairwise verified matches are preserved in the edge information of the similarity graph. This later step typically improves the mean Average Precision (mAP) of the retrieval process i.e. we get a higher precision-recall values corresponding to the retrieval results.

C. *Neighborhood relations in Similarity Graph*

As shown in Figure 5, an image having the same scene annotation as another could be visually dissimilar to each other. This occurs frequently in case of stereo images of a building or architectural structure with significant viewpoint variations at a heritage site. However, when we have a dense image dataset, it is possible to have intermediary matching images to suggest that two visually dissimilar images may actually come from the same scene. That is, an image I_3 may

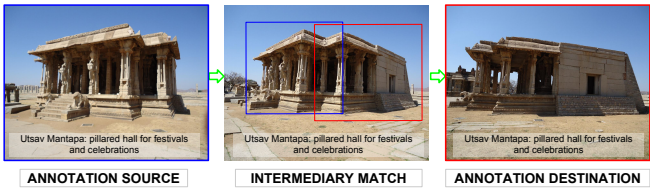

Figure 5. Identifying the same scene in two visually dissimilar images by exploiting the neighborhood relations in the Image Similarity Graph. As can be seen in the figure, the intermediary match helps to identify the similarity between the annotation source and destination.

not be a direct neighbor of image I_1. However, if image I_2 is a neighbor of I_1, and image I_3 is a neighbor of I_2, then we can verify by the intersection of matching features, whether I_1 and I_3 come from the same scene. Thus, the image similarity graph helps verifying the neighbors-of-neighbors relations with strong edge-weights to identify images from the same scene-instance. This helps in propagating the scene annotations to a larger number of images.

D. *Object-Boundary Correspondence*

The objects annotated in the source query I_Q need to be automatically identified in all the retrieved and verified target images $I_1, I_2, ..I_N$. Some objects in the retrieved matches may be partially occluded, which pose a difficulty in obtaining boundary correspondence in those images. The

Figure 6. Illustration of Object-Boundary correspondence for an annotated object query source image and its top-3 retrieved results.

list of matching feature keypoints between the source and the target, is obtained from the edge information in the similarity graph. These matches are already verified at the graph building stage. We now compute the homographies H_i's between I_Q and I_i, where $i = 1, 2, ..N$. Using perspective transformation, we can now project any point from I_Q onto the target images I_i's. The rectangular object boundary coordinates for the source image I_Q, are estimated in each of the target images I_i, to localise the corresponding annotated objects (See Figure 6). Now, the retrieved images also have rectangular boundaries over matching object regions. These can be again manually checked before transferring the object annotations to the target images.

IV. Results and Discussions

The web-based annotation tool is built using *HTML5* and advanced *javascript* technologies. The server end uses a *mysql* database to centrally store annotations by different authors (collaborative users). The image browsing interface has been developed keeping in view the large size of databases that could be required to handle. We demonstrate the usage of the tool on a $20K$ image dataset from a famous heritage site. The image similarity graph for the dataset is pre-computed and stored in the form of *json* object files.

A. Heritage-20K dataset

Figure 7. Random sampling from the Heritage-20K dataset.

We introduce a novel data set of images from a world heritage site. This dataset has 20,958 medium-resolution (480×360) images of specific buildings, architectural details, interiors and exteriors of heritage monuments, etc. A random sampling from the dataset is shown in Figure 7. The challenge is to collect precise annotations for each structure. There are many similar looking beautiful structures (mostly stone carvings on pillars, walls, etc) with different significance and an enthusiastic visitor expects to learn most of it during his tour of the place.

B. Annotation Transfer using Similarity Graph

We use SIFT as the local image descriptors to extract features from each image. These 128-dimensional feature vectors are then quantized using Hierarchical K-Means to get a vocabulary tree [14], whose leaves represent the visual words i.e. cluster centers. The inverted search index is constructed as discussed in Section III-B. Each image from the photo collection is then queried to get the Image Similarity Graph. During the online process, given a query image with annotated object regions, we perform object-boundary-correspondence for the database images.

C. Annotation Tool

The tool has a simple interface to browse image thumbnails. The large number of image thumbnails are fetched on-the-fly using *ajax* from the server, when the user scrolls through the collection. This makes the browsing experience fast, smooth and hassle-free. Annotated images are shown within a green border on the thumbnail. Also, listed are few popular images that are not yet annotated. A popular image is chosen from the graph, if it is closely related to many similar images. Such photos may represent scenes or objects that are of interest to many. On selecting an image thumbnail, the full image appears on a *HTML5 canvas* and we have the controls for adding, modifying and deleting annotations for this image (See Figure 8-(a)).

There are separate controls for scene and object annotations. To add an object annotation, a boundary must be drawn over the image region on the canvas. This has been made simple with just two mouse clicks to represent the opposite corner vertices of a rectangular boundary. Once the annotations are added, the user can click and search for images from the database that are similar to the annotated photo. A list of images with corresponding object boundaries are displayed to the user. One can manually identify the false positives before clicking a button to propagate the annotations (See Figure 8-(b)). Each verified image is annotated in the database with the scene and object instances annotated in the source photo.

D. Evaluation

The effectiveness of the web-based tool built for the Heritage-20K photo collection, is measured in terms of the manual efforts required to give rich and precise annotations to a set of images. A set of 30 distinct "popular" images were chosen for this purpose. For test purpose, only text annotations were considered for scene and object instances. The length of scene-instance text description for each of the 30 images was on average, 25-words or, 170-characters. Each image had 2 distinct object-instance annotations on an average, which implied that 2×2 clicks were required to mark the rectangular object boundaries. Each object was associated with a 10-word or, 70-characters description on an average. So, each image required typing 310 characters and 4 mouse clicks to associate a meaningful annotation.

Using our efficient batch annotation approach, each photo retrieved 20 verified similar photos on average and with a single click, the annotations for one image was propagated to 20 more images in a few seconds. So, at the end of the task, a total of 600 images were annotated. Thus, with the same efforts for annotating 30 photos, plus 30 extra clicks and a couple of minutes extra time, we annotated 600 photos

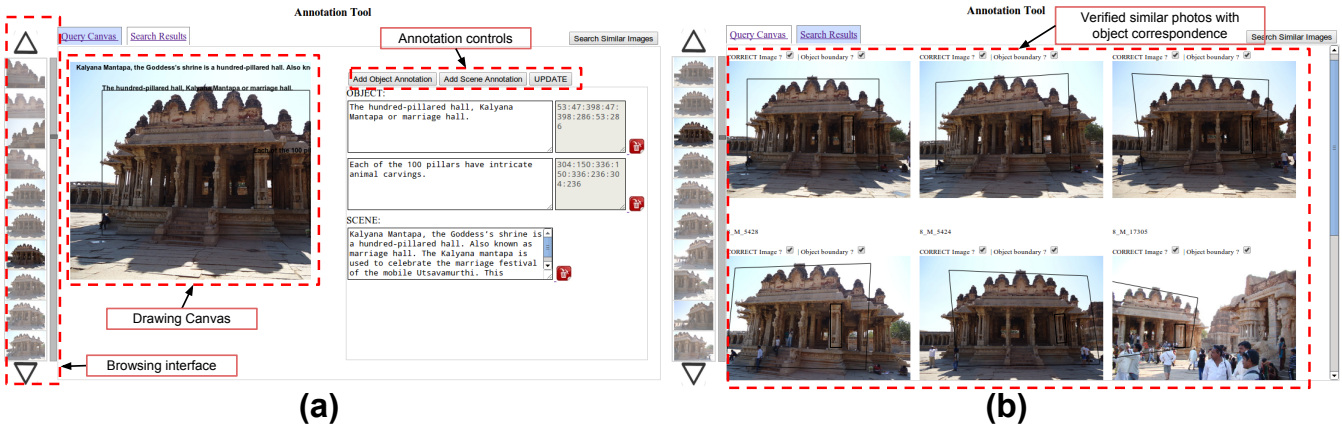

Figure 8. Snapshots of the web-based Annotation Tool allowing a user to select, annotate and propagate a suitable description for an image or an object. (a) displays the interface for browsing photos, the controls for adding annotations to a selected photo and the canvas where the image and object boundaries are drawn; (b) shows the list of similar photos, with corresponding object boundary regions, to which the source annotations can be propagated.

instead of 30. This process was 20 times efficient than the naive way of annotating images one-by-one.

The task was assigned to 5 different users, who had visited the site before and had access to online and textual resources. Each user was assigned 6 photos. The user who completed the task earliest took 15 minutes and the most time taken by any user was 20 minutes. Hence, we were able to richly annotate 600 images in a short span of 20 minutes, which would have otherwise consumed hours of a typical data-entry professional.

V. Conclusion

In this paper, we demonstrate a simple web-based scalable tool for annotating large image datasets, emphasizing the importance of rich content to be associated with every image. Target datasets are primarily, the community photo collections of tourist and heritage sites, where a lot of information with tiny interesting details could be built up for each distinct artifact or structure present. We design an easily navigable interface suitable for various types of users - historians, students, researchers, tour guides as well as casual tourists. The tool is developed to work in a client-server fashion. For efficiently annotating a large number of similar images, we make use of the image similarity graph, which is pre-computed from the dataset. Bag-of-Words (BoW) based image retrieval and SIFT-based matching and verification is used to build the similarity graph. We demonstrate the usage of the tool to annotate a large photo collection of $20K$ images from a heritage site.

VI. Acknowledgements

This work was partly supported by Indian Digital Heritage project of DST, India.

References

[1] J. Sivic and A. Zisserman, *Video Google: A Text Retrieval Approach to Object Matching in Videos*. ICCV, 2009.

[2] Martin A. Fischler and Robert C. Bolles, *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography*. Communications of the ACM, 1981.

[3] J. Philbin , O. Chum, M. Isard, J. Sivic and A. Zisserman, *Object retrieval with large vocabularies and fast spatial matching*. CVPR, 2007.

[4] Noah Snavely, Steven M. Seitz, Richard Szeliski, *Photo tourism: Exploring photo collections in 3D*. SIGGRAPH, 2006.

[5] Michael Goesele, Noah Snavely, Brian Curless, Hugues Hoppe, Steven M. Seitz, *Multi-View Stereo for Community Photo Collections*. ICCV, 2007.

[6] Ian Simon, Noah Snavely, and Steven M. Seitz, *Scene Summarization for Online Image Collections*. ICCV, 2007.

[7] Ian Simon and Steven M. Seitz, *Scene Segmentation Using the Wisdom of Crowds* . ECCV, 2008.

[8] James Hays and Alexei A. Efros, *Scene Completion Using Millions of Photographs*. SIGGRAPH, 2007.

[9] R. Fergus, L. Fei-Fei, P. Perona and A. Zisserman, *Learning Object Categories from Google's Image Search.*. CVPR, 2005.

[10] James Hays and Alexei A. Efros, *IM2GPS: estimating geographic information from a single image* . CVPR, 2008.

[11] Noah Snavely, Rahul Garg, Steven M. Seitz, and Richard Szeliski, *Finding Paths through the World's Photos*. SIGGRAPH, 2008.

[12] S. Gammeter, L. Bossard, T. Quack and L. Van Gool, *I know what you did last summer: object-level auto-annotation of holiday snaps.* ICCV, 2009.

[13] J. Panda, S. Sharma and C V Jawahar, *Heritage App: Annotating Images on Mobile Phones.* ICVGIP, 2012.

[14] D. Nistér and H. Stewénius, *Scalable Recognition with a Vocabulary Tree.* CVPR, 2006.