Computer Vision and Image Understanding 135 (2015) 1–15

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu

Visual landmark recognition from Internet photo collections: A large-scale evaluation ☆

Tobias Weyand *, Bastian Leibe

Computer Vision Group, RWTH Aachen University, Germany

ARTICLE INFO

Article history:
Received 14 September 2014
Accepted 7 February 2015
Available online 23 February 2015

Keywords:
Landmark recognition
Image clustering
Image retrieval
Semantic annotation
Compact image retrieval indices

ABSTRACT

The task of a visual landmark recognition system is to identify photographed buildings or objects in query photos and to provide the user with relevant information on them. With their increasing coverage of the world's landmark buildings and objects, Internet photo collections are now being used as a source for building such systems in a fully automatic fashion. This process typically consists of three steps: clustering large amounts of images by the objects they depict; determining object names from user-provided tags; and building a robust, compact, and efficient recognition index. To this date, however, there is little empirical information on how well current approaches for those steps perform in a large-scale open-set mining and recognition task. Furthermore, there is little empirical information on how recognition performance varies for different types of landmark objects and where there is still potential for improvement. With this paper, we intend to fill these gaps. Using a dataset of 500 k images from Paris, we analyze each component of the landmark recognition pipeline in order to answer the following questions: How many and what kinds of objects can be discovered automatically? How can we best use the resulting image clusters to recognize the object in a query? How can the object be efficiently represented in memory for recognition? How reliably can semantic information be extracted? And finally: What are the limiting factors in the resulting pipeline from query to semantics? We evaluate how different choices of methods and parameters for the individual pipeline steps affect overall system performance and examine their effects for different query categories such as buildings, paintings or sculptures.

1. Introduction

Recognizing the object in a photo is one of the fundamental problems of computer vision. One generally distinguishes between object categorization and specific object recognition. Object categorization means recognizing the *class* that an object belongs to, *e.g.* painting or building, while specific object recognition means recognizing a specific object *instance*, such as the Mona Lisa or the Eiffel Tower. In this paper, we consider the latter task, *i.e.*, specific object recognition. In particular, we are interested in two applications, namely photo auto-annotation and mobile visual search. A photo auto-annotation system recognizes objects in a user's photo albums and labels them automatically, saving the user the effort of manually labeling them. A mobile visual search system provides a user with information on an object that they took a picture of with their smartphone. Because a large part of the photos in these applications are typically tourist photos, many of the objects that such systems need to recognize are landmarks. Therefore, the problem is typically referred to as *landmark recognition*. However, many other types of objects, such as paintings, sculptures or murals, can also be recognized by such systems.

The first step of building a landmark recognition system is to compile a database consisting of one or more photos of each object that shall be recognized. However, since the number of objects that can possibly appear in a user's photos is virtually infinite, it is impossible to construct and maintain such a database by hand. An elegant solution is to build the database from the data it is meant to be applied to, namely public photos from Internet photo collections such as Flickr, Picasa or Panoramio. This approach has several attractive properties: (i) Objects are discovered in an unsupervised, fully automatic way, making it unnecessary to manually create a list of objects and collecting photos for each of them. (ii) The resulting set of objects is likely to be much better adapted to the queries a photo auto-annotation or visual search system might receive than a hand-collected set of objects. (iii) The level of detail of object representation is automatically adapted

☆ This paper has been recommended for acceptance by Vincent Lepetit.

* Corresponding author at: Mies-van-der-Rohe-Strasse 15, D-52074 Aachen, Germany.

E-mail addresses: weyand@vision.rwth-aachen.de (T. Weyand), leibe@vision.rwth-aachen.de (B. Leibe).

http://dx.doi.org/10.1016/j.cviu.2015.02.002
1077-3142/

to the demand. The most popular objects will be represented by the most photos in the database, increasing their chance of successful recognition, while only little memory is used on less popular objects. This approach has gained popularity in the research community [1–5] and is also being used in applications such as Google Goggles [6].

Constructing such landmark recognition systems on a large scale involves three main research problems: (i) Finding interesting structures in internet image collections, (ii) automatically connecting them to associated semantic content by examining user-provided titles and tags, and (iii) compactly representing them for efficient retrieval. While each of those problems has already been studied in isolation, so far there has not been a systematic evaluation of all three aspects in the context of a fully automatic pipeline. Image retrieval approaches like [7,8] find matching images for a query, but have no notion of the semantics of the depicted content. Tag mining approaches [3,9,4,10] try to find a description of an image cluster, but so far the large effort to evaluate tag quality has prevented quantitative evaluations in a large-scale setting. Landmark object discovery approaches [1,11,4–6] aim at finding interesting buildings and other objects, but no systematic evaluation has been performed that analyzes what types of objects can be discovered and how recognition performance varies with these types. Furthermore, it is still largely unclear what is the best strategy to determine the identity of the recognized object based on the set of retrieved database images (which becomes a non-trivial problem whenever image clusters overlap and images may contain multiple landmark objects).

In this paper, we evaluate the whole process of constructing landmark recognition engines from Internet photo collections. To do this in a realistic large-scale setting we require a dataset containing thousands of objects. Moreover, in order to create a realistic application scenario, the target database objects should not be specified by hand (as in many other datasets), but should be mined automatically. Last but not least, the dataset should not be limited to buildings, but also contain smaller objects, such as paintings or statues.

Our evaluation is based on the Paris 500k dataset [12] containing 500 k photos from the inner city of Paris, which was mined from Flickr and Panoramio using a geographic bounding region rather than keyword queries to obtain a distribution unbiased towards specific landmarks. Thus, in contrast to other common datasets, there is no bias on tag annotations or content. In order to evaluate landmark recognition in a realistic setting, we additionally collected a query set of almost 3000 Flickr images from Paris that is disjoint from the original dataset. Our evaluation thus mimics the task of photo auto-annotation where a photo uploaded to a photo sharing website is automatically annotated with the object it depicts.

To evaluate the performance of landmark recognition, we use a recent landmark discovery algorithm [5] to discover landmarks in the dataset. We created an exhaustive ground truth for the relevance of each of the discovered landmarks with respect to each of the 3000 queries, which involved significant manual effort. This is the first ground truth for evaluating landmark recognition on an unbiased and realistic dataset. To enable the comparison of other approaches with the ones evaluated in this paper, the ground truth is publicly available.

To give a detailed performance analysis for different types of objects, we introduce a taxonomy for the objects landmark recognition systems are able to recognize. Throughout our evaluation, we report both summary performances over the entire database and detailed findings for different object categories that show how their recognition is affected by the different stages of the system. As our results show, the observed effects vary considerably between query categories, justifying this approach. We give detailed results for each category, and use the four use cases of *Landmark Buildings*, *Paintings*, *Building Details* and *Windows* as representatives for different challenges. The taxonomy is available along with the ground truth.

Note that our goal is not primarily to propose novel methods (although some of the methods evaluated in Section 6 and Section 7 are indeed novel), but to provide answers to the following questions:

- How many and what kinds of objects are present in Internet photo collections and what is the difficulty of discovering objects of different landmark types (Section 5)?
- How to decide which landmark was recognized given a list of retrieved images (Section 6)?
- How to efficiently represent the discovered objects in memory for recognition (Section 7)?
- Are the user-provided tags reliable enough for determining accurate object names (Section 8)?
- Given the entire retrieval, recognition and semantic labeling pipeline, what are the factors effectively limiting the recognition of different object categories (Section 9)?

Our analysis provides several interesting insights, for example:

- Semantic annotation is the main bottleneck for system performance. In many cases, the correct object is visually recognized, but the name of the object cannot be determined due to the sparsity and amount of noise of user-provided image titles and tags.
- Different bottlenecks exist for different object categories. For example, *Murals* are easy to recognize using the standard visual words pipeline, but reliable semantic information is often missing for them. For other objects like museum exhibits, the opposite is the case: While semantic information is readily available, they are hard to recognize visually due to their spatial structure and scarce visual examples.
- When the desired application is building recognition, a seeding-based clustering method can bring significant computational savings, since buildings are already discovered when using few seeds, while smaller objects require orders of magnitude more seeds.
- Different techniques for compactly representing object clusters are optimal for different object types.

As a result of this evaluation, we can identify several interesting directions, where progress can still be made.

2. Engine architecture

The architecture of a typical landmark recognition engine such as [1,13,4,6] is shown in Fig. 1. Large amounts of tourist photos are clustered, resulting in a set of *objects*. By *object*, we denote a cluster of images that show the same entity. We will refer to the images in each object cluster as its *representatives*. Since the clusters may overlap, a representative can belong to multiple objects. Each object is then associated with *semantics* (typically its name), *e.g.*, by mining frequently used image tags. The set of representatives for each cluster is often decimated by eliminating redundant images in order to save memory and computation time. To recognize the object in a query image, a visual search index [14,8,15] containing all *representatives* is queried, producing a ranked list of matches. Based on this list, *objects* are ranked w.r.t. their relevance to the query and the corresponding *semantics* are returned.

In this paper, we evaluate different choices for the components of this framework and demonstrate how they affect the system's