CBGIR: Content-Based Geographic Image Retrieval (Demo Paper)

Shawn Newsam, Daniel Leung, Oscar Caballero, Justin Floreza, and Jesus Pulido
Electrical Engineering & Computer Science
University of California at Merced
snewsam,cleung3,ocaballero,jfloreza,jpulido@ucmerced.edu

ABSTRACT

We describe and demonstrate CBGIR, a web-based system for performing content-based image retrieval in large sets of high-resolution overhead images. The system provides a familiar Google Maps interface to navigate the images and select regions of interest. A query-by-example paradigm is used to retrieve the most visually similar images to this region from a large target set of image tiles. Similarity can be computed with respect to a number of visual features including color, texture, and local invariant descriptors. We describe the salient components of the system and provide sample retrieval results.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*spatial databases and GIS*; I.5.4 [**Pattern Recognition**]: Applications; I.4.8 [**Image Processing and Computer Vision**]: Scene Analysis

Keywords

content-based image retrieval

1. INTRODUCTION

This paper describes a web-based system[1] for performing content-based image retrieval against a target set of geographic images. In particular, a query-by-example paradigm is used in which a user navigates overhead images using a Google Maps interface; selects a region of interest as the query; submits the query to a remote server; and is presented with the set of image tiles which are most similar to the query with respect to a visual feature such as color or texture.

Such a system provides novel ways for users to interact with large collections of geographic images without requiring the manually expensive task of semantic-level annotation. However, the demo itself serves a number of useful purposes including: 1) it allows a quick visual evaluation of candidate features for other image analysis tasks such as land-use/cover classification or object recognition; 2) it provides a visual method for conveying what is meant by content-based image analysis–this is useful both in research settings when interacting with geographic domain experts as well as educational settings when teaching computer science courses on image processing and computer vision; and 3) the system itself represents a sand-box of sorts for a variety of undergraduate and graduate research projects ranging from image feature extraction, efficient similarity search, web client-server programming, and user interface design and implementation.

[1]http://vision.ucmerced.edu/demos/GIR/

ACM GIS '10 , November 2-5, 2010. San Jose, CA, USA

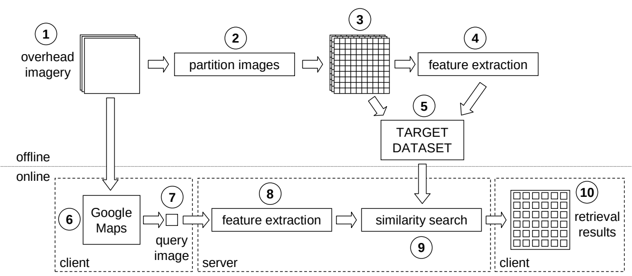

Figure 1: CBGIR system architecture.

2. SYSTEM OVERVIEW

The CBGIR system architecture, shown in figure 1, can be broadly decomposed into offline and online components.

Offline The offline components represent a preprocessing step. The large set of overhead images (1) are partitioned (2) into smaller tiles (3). We currently use 256x256 pixel non-overlapping tiles but other configurations are possible. Feature extraction (4) is performed on each of these tiles. Finally, the image tiles and their corresponding features are stored as the target dataset (5). The offline processing need only be performed when new images become available for ingestion into the system.

Online The online components perform the similarity search. A client-side (browser) Google Maps visual interface (6) is used to navigate the large overhead images and select a region of interest as the query image (7). Feature extraction is performed (8) on the query image on the server (webserver). The extracted features are used to perform a similarity search (9) against the target dataset. Finally, the most similar images are sent to the client for display (10). The online datapath is executed once for each user query.

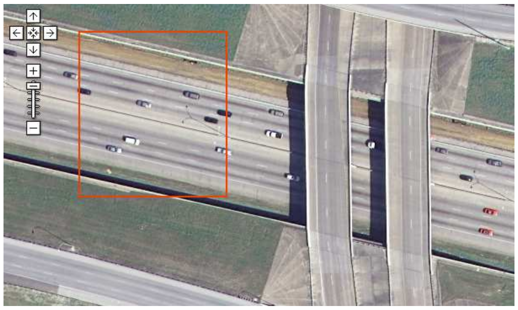

Figure 2: A Google Maps interface is used to navigate overhead images and select a query image (shown by the red bounding box). This image is a one foot pixel resolution aerial photo of the Dallas region.

3. SYSTEM DETAILS

3.1 Google Maps Interface

The Google Maps Javascript API is used to implement the visual interface for navigating the collection of overhead images and selecting the region of interest to serve as the query image. The familiarity of the Google Maps interface for zooming and panning makes the system accessible to new users. The region of interest is selected by using the right mouse button to specify the two opposing corners of a bounding box. The non-Google Maps components of the interface include GUI components for selecting which overhead image is displayed, which visual feature is used to perform the search, how many images should be retrieved, and for initiating the search.

3.2 Visual Features

The system architecture is general enough to support a range of visual features. In particular, any feature which can be represented as a vector and compared using a pairwise similarity measure can be integrated into the system. The current implementation includes two-dimension simple statistics features composed of the mean and standard deviation of the luminance channel of an image (this simple feature serves as a baseline); 512-dimension color histogram features in three colorspaces, RGB, HLS, and CIE Lab; 60-dimension homogeneous texture features compliant with the MPEG-7 Multimedia Content Description Interface [2] standard; and 100-dimension histograms of quantized local invariant descriptors extracted at salient points (a so called bag-of-visual-words representation [3] based on scale invariant feature transform descriptors [1]).

3.3 Similarity Search

Once the visual feature has been extracted from the query image, it is used to perform a similarity search against the target dataset. Specifically, a similarity measure is used to compare the query vector with each of the target vectors. The resulting similarity scores are used to identify the most similar target tiles. A straightforward linear search is performed for the most similar tiles since multi-dimensional indexing techniques provide no benefit past a dozen or so dimensions. The current implementation uses a histogram intersection similarity measure to compare histograms (both color and bag-of-visual-words) and the Euclidean distance to compare the homogeneous texture features.

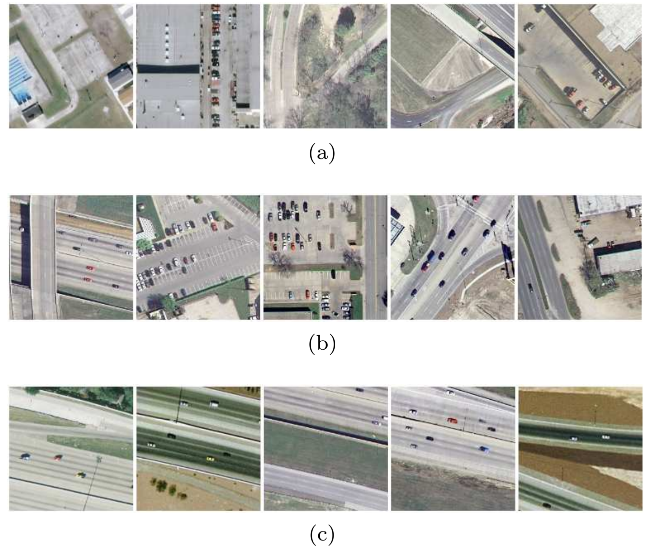

Figure 3: The five most similar image tiles to the query in figure 2 with respect to (a) the simple statistics features (the mean and standard deviation of the luminance channel); (b) color histogram features computed in the HLS colorspace; and (c) homogeneous texture features. The target set contains approximately 50,000 tiles.

4. SAMPLE RETRIEVALS

The CBGIR system currently contains approximately 30 large one foot pixel resolution aerial images downloaded from the USGS National Map. This results in a target dataset of over 50,000 256x256 tiles. Figure 2 shows a screenshot of a user navigating an image of the Dallas region and selecting a query image. Figures 3(a), 3(b), and 3(c) show the five most similar images to this query with respect to simple statistics features, color histogram features computed in the HLS colorspace, and homogeneous texture features.

5. ACKNOWLEDGEMENTS

This work was funded in part by NSF grant IIS-0917069 and a Department of Energy Early Career Scientist and Engineer/PECASE award. Oscar Caballero was supported by an NSF Alliances for Graduate Education and the Professoriate (AGEP) scholarship. Any opinions, findings, and conclusions or recommendations expressed in this work are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

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

[2] B. S. Manjunath, P. Salembier, and T. Sikora, editors. *Introduction to MPEG7: Multimedia Content Description Interface*. John Wiley & Sons, 2002.

[3] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *IEEE International Conference on Computer Vision*, 2003.