Expert Systems with Applications 36 (2009) 4451–4459

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Ensemble one-class support vector machines for content-based image retrieval

Roung-Shiunn Wu [a,*], Wen-Hsin Chung [b]

[a] Department of Information Management, National Chung Cheng University, 168 University Road, Min-Hsiung, Chia Yi 62102, Taiwan
[b] Graduate Institute of Information Management, National Chung Cheng University, Taiwan

ARTICLE INFO

Keywords:
Content-based image retrieval
Relevance feedback
Ensemble
Support vector machines

ABSTRACT

In order to narrow semantic gap between user query concept and low-level features in content-based image retrieval, SVM-based relevance feedback techniques are developed to learn user's query concept by labeling some samples. The major difficulty in relevance feedback is to estimate the support of target image in high-dimensional feature space with small number of training samples. To overcome this limitation, we propose an ensemble method to boost image retrieval accuracy and to improve its generalization performance. Images are segmented into multiple instances. A set of moderate accurate one-class support vector machine classifiers are trained separately by using different sub-features extracted from instances. The ensemble method results in a highly accurate by combining moderately accurate weak classifiers. Our propose ensemble scheme not only provides a robust mechanism in selecting strong query concept related images for relevant feedback, but also achieves a generalization performance in image retrieval.

1. Introduction

In many applications, digital images have been processed, organized, and stored in image repositories. This results in a subsequent shift to management of information content for image retrieval. In general, image retrieval has been primarily concerned with two approaches. One approach is to use keywords or text annotations to describe image semantics for retrieving images from a database (Guo, Jain, Ma, & Zhang, 2002). The limitation of this applicability is due to time consuming and expense of human annotation. The other approach is to use content-based image retrieval (CBIR) (Hong, Tian, & Huang, 2000; Rui, Huang, & Mehrotra, 1997). The main idea of CBIR relies on low-level features representation of images in terms of their visual content such as color, texture, and shape in order to compare images.

To answer a query, the image search engine scans all of the images in image database by comparing their feature vectors with that of the query image. The similarity measure is to take consideration distances of low-level features between the query image and retrieval images. However, only the use of low-level features for CBIR cannot achieve a satisfactory measurement performance, since user's high-level query concept cannot be easily expressed by low-level features.

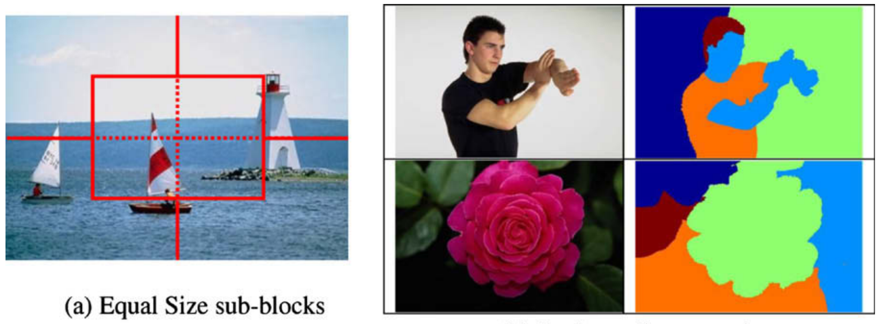
For images, high-level semantics may be somewhat correlated with low-level features. Different users at different times have different interpretation for the same image. Images in database cannot adopt a fixed clustering structure for retrieval, since image retrieval is user dependent and time varying. To narrow the gap between low-level features representation of images and user's high-level semantic concepts, relevance feedback is employed to learn user's query concepts (He, King, Ma, Li, & Zhang, 2003; Yang, Dong, & Fotouhi, 2005). The image retrieval system iteratively provides a small number of images for the user to label as "relevant" or "irrelevant". Based on the query image and labeled images, the retrieval system then dynamically adjusts its learning toward relevant samples and away from irrelevant samples. After several rounds of relevance feedback, the similarity measurement of retrieval images may be in satisfaction with user's interest.

If an image is represented by a point in a feature space, relevance feedback with relevant and irrelevant images, i.e. positive and negative training samples, becomes a classification problem. Learning algorithm such as binary class SVM may not provide a stable result for small training samples (Zhou & Huang, 2001). The desired output of image retrieval is not a binary decision on each sample as given by a classifier, but rather a rank of top k samples return. The rank of irrelevant samples is of no concern. Hence, a more generalization classification or learning algorithm is needed to resolve this issue.

In order to overcome the limitation associated with binary class SVM with relevance feedback, one-class SVM is employed to form relevance feedback for content-based image retrieval. By exploiting one-class SVM information, it is able to predict accurately the local relevance of feature dimensions and to capture user perceived similarity. We propose a framework that ensembles one-class SVMs (Kim, Pang, Je, Kim, & Yang Bang, 2003). The final decision

* Corresponding author. Tel.: +886 5 2720411x34621; fax: +886 5 2721501.
E-mail address: roungwu@ccu.edu.tw (R.-S. Wu).

0957-4174/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2008.05.037



(a) Equal Size sub-blocks Segmentation

(b) Region Segmentation

Fig. 1. Block and region segmentations.

of the ensemble for a given image samples is due to majority votes after individual one-class SVM classifiers are trained by using different sub-feature vectors. Our aims are to provide images which are strong related to query image for users to label, to improve retrieval efficiency by using sub-features for training weak learners, and to boost retrieval accuracy by using ensemble method.

The remainder of this paper is organized as follows. In Section 2, we provide some prior works of relevant feedback, multiple instances learning, and visual features extraction. One-class SVM is described in Section 3. The proposed ensemble framework is presented in Section 4. In Section 5, we provide the experiment design and report the results. Conclusion is given in Section 6.

2. Related works

2.1. Relevance feedback

Relevance feedback is an effective way to refine queries by capturing query concepts from the user. The feedback loop is interactive with the user iteratively until the user satisfies with the result. According to Yang et al. (2005) and Lu et al. (2003), the previous relevance feedback research can be categorized into query point movement and re-weighting method. The idea of query point movement is to compute the new optimal query point that is closer to relevant samples and far away from irrelevant samples. The most used formulation of the query point movement is Rocchio's formula. The technique has been implemented in several image retrieval systems, such as MARS (Rui et al., 1997) and Mindreader (Ishikawa, Subramanya, & Faloutsos, 1998). The re-weighting method (Peng, Bhanu, & Qing, 1999) is used to improve relative importance of feature dimensions for the similarity metric. It analyzes from the relevant objects in order to capture the degree of importance of each dimension and then give the different weights on them.

In relevance feedback retrieval system, range query and K-nearest-neighbors (KNN) query are two common similarity query types for determining the numbers of retrieving samples to be fetched for users to label. Range query is to find the images in the collection that are within predefined distance from the query image. There are two common problems in range query. One is no result is fetched; the other is to fetch too many results. KNN returns K images which are closest to the query. The Euclidean distance and the Mahalanobis distance are the most used common measure of similarity between two images. Hong et al. (2000) used the Mahalanobis distance for the fundamental similarity measure. They applied SVM and relevance feedback to learn weights and integrated weights into the similarity metric.

The major difficulty in relevance feedback is to estimate the support of target image in high-dimensional feature space with small number of training samples (Schölkopf, Platt, Shawe-Taylor, Smola, & Williamson, 2001). Images which are interested to users are small portion of large images in database, in which most of them remains unlabeled. The strategy is to get some feedbacks from users by providing users the result from the previous query and to refine the search strategy.

2.2. Multiple instances segmentation

In this research work, we employ block-based and region-based segmentation techniques to segment an image into five segments. Each segment represents an instance. Visual features are extracted from each instance. Block-based segmentation is to segment each image into sub-blocks. The size and shape of sub-blocks can be either the same or different (Meghini, Sebastiani, & Straccia, 2001). We divide each image into five equal size quadrants as illustrated in Fig. 1a. Four of them are segmented from the upper, bottom, left, and right halves of the image. The other one is segmented in the central area of the image.

Region-based or object-based segmentation is to segment an image into several segments-based on homogenous objects as illustrated in Fig. 1b. The use of region-based segmentation algorithms is not as efficient as block-based segmentation. However, the objects may be close to user's query interest. Cour, Benezit, and Shi (2005) proposed a multiscale spectral image segmentation algorithm. We employ Multiscale Ncut[1] which is a toolbox in Matlab to segment images. We also segment an image into five regions.

2.3. Visual features extraction

There are numerous low-level visual features to represent an image, such as color, texture, and shape. Color information is not only the most essential but also the intensively used feature for image retrieval. Texture is another type of primitive visual cues for image retrieval. Texture contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment (Haralick, Shanmugam, & Dinstein, 1973). We use three texture feature extraction techniques Gabor filters, wavelet transforms, and co-occurrence matrix to extract texture features from each instance. We construct a color sub-feature vector and three texture sub-feature vectors for each instance. The instance sub-feature vectors are stored in image database and are indexed to the original image for retrieval processing. Sub-feature vectors constructions are described in the following subsections.

[1] Download from http://www.seas.upenn.edu/~timothee/.