© 2018 IJSRSET | Volume 4 | Issue 6 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section : Engineering and Technology

An Enhanced Feature Extraction Model for the Information Retrieval In Crop Disease Detection

Sampathkumar S[1], Rajeswari R[2]

[1]Assistant Professor, Department of Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore, India

[2]Assistant Professor (Senior Grade), Department of Electrical and Electronics Engineering, Government college of Technology, Coimbatore, India

ABSTRACT

Fast similarity search has more importance in a large-scale dataset while using for image indexing and retrieval in many applications. Semantic hashing is an efficient way to accelerate similarity search, which designs compact binary codes for a large number of images so that semantically similar images are mapped to close codes. Among various hashing approaches, spectral hashing has shown better performance by learning the binary codes with a spectral graph. And Color is one of the most important and widely used in content analysis and retrieval. However, most promising color descriptors consume massive amounts of computation and storage, which is a serious drawback. One of these promising color techniques in image retrieval is the Color Correlogram (CC), but the technique also suffers from the aforementioned drawbacks. Compact and conceptual Correlogram descriptor can be used to represent color Correlogram. Compact Correlogram uses compact-generalized Correlogram, which compresses color and generalizes the distances of the actual Correlogram descriptor and conceptual Correlogram uses spatial correlations for the dominant color of few images instead of a large number of quantized color used by the original descriptor. These two representations are integrated with multiple instance learning method to indexing and retrieval of images in both text application and in multimedia applications. Multiple-instance learning has been widely used in image indexing for its capability of exploring region-level visual information of images. Semantic hashing with multiple instance learning leads better results compared to other methodology in terms of indexed precision with respect to image pixels and image bits.

Keywords: Color Correlogram, Compact Correlogram, Content-Based Image Retrieval (CBIR).

I. INTRODUCTION

Image Indexing is an application for processing document images which is a part of a business workflow at where roles are assigned to perform specific functions. In a typical workflow, document images progress from role to role based on information specified in data entry fields. In response to the data entry, Image Indexing updates the status of the image and, if specific criteria have been met, transitions the image from role to role. The problems of image retrieval are becoming widely recognized, and the search for solutions an increasingly active area for research and development. Problems with traditional methods of image indexing have led to the rise of interest in techniques for retrieving images on the basis of automatically-derived features such as color, texture and shape – a technology now generally referred to as Content-Based Image Retrieval (CBIR). Content-Based Image Retrieval (CBIR), a technology that realizes information retrieval by using the image content directly, is currently a popular method in information retrieval. Google, Yahoo, Bing and Baidu have all launched image search engines taking the content of images as input. Since people focus more on the color, image retrieval methods based on color feature have been developing fast during the past decade. Feature extraction is a preprocessing step for image indexing and retrieval in the CBIR system and several color description technologies have been proposed to be used as the feature vectors.

Color Histogram is the earliest method to express the feature, which is invariant to rotational changes, distance changes and partial occlusion of the target object around the view. However, the technology still lacks maturity, and is not yet being used on a significant scale. In the absence of hard evidence on the effectiveness of CBIR techniques in practice, opinion is still sharply divided about their usefulness in handling real-life queries in large and diverse image collections. In such a situation, it is difficult for managers and users of image collections to make informed decisions about the value of CBIR techniques to their own work. The use of auto-Correlogram as feature vectors in content based image retrieval (CBIR) systems outperformed color histograms and other types of feature vectors. The objective of image indexing is to retrieve similar images from an image database for a given query image. Each and every image has its unique feature. And hence it can be implemented by comparing their features that are extracted from the images. The similarity criteria among images may be based on the features such as color, intensity, shape, location and texture. There are two types of techniques used for image indexing.

1. Textual indexing
2. Content based indexing

Textual indexing

It is very simple techniques; keeping in mind the user approach keywords are given for a particular image. This includes Caption indexing, Keyword additions Standard subject headings, Classification.

Content-based indexing

It is also known as automated indexing. In this technique images are indexed based on their content like color, shape, direction, texture, spatial relation etc. This kind of indexing is taken care by software itself, algorithms are developed which can differentiate the color, shape, textures etc. The image retrieved through this technique is known as Content Based Image Retrieval (CBIR).

Similarity search is defined as the task of finding close samples for a given query. It is of great importance to many multimedia applications, such as content-based multimedia retrieval. Recently, with the rapid evolution of the Internet and the explosive growing of visual contents on the Web, large-scale image search has attracted considerable attention. Exhaustively comparing the query image with each sample in the database is infeasible because the linear complexity is not scalable in practical situations. Hashing-based methods are promising in accelerating similarity search for their capability of generating compact binary codes for a large number of images in the dataset so that similar images will have close binary codes. Retrieving similar neighbors is then accomplished simply by finding the images that have codes within a small Hamming distance of the code of the query. It is extremely fast to perform similarity search over such binary codes, because

1) The encoded data are highly compressed and thus can be loaded into the main memory;
2) The hamming distance between two binary codes can be computed efficiently by using bit XOR operation and computing the number of set bits (an ordinary PC today would be able to do millions of Hamming distance computation in just a few milliseconds).

Many hashing algorithms have been developed in recent years and the hashing methods can mainly be divided into two categories: unsupervised methods and supervised methods. Such hashing-based methods for fast similarity search can be considered as a means for embedding high dimensional feature vectors to a low dimensional Hamming space, while retaining as much as possible the semantic similarity structure of data. There are several weaknesses of such approaches. First, the distance metric learning and hashing can be regarded as two isolated steps in this approach, and the objective optimized in distance metric learning may not be appropriate for hashing. Second, for many hashing algorithms that rely on similarity rather than distance, such as SH, it needs a further step to convert distance to similarity and the involved radius parameter is usually sensitive. Unsupervised methods use just the unlabeled data to generate binary codes for given points, while supervised methods, which incorporate the label information, are able to preserve the semantic similarity and thus facilitate semantic retrieval and classification. It improves spectral hashing, a state-of-the-art hashing algorithm that has shown superior performance than many other methods, by optimizing the graph Laplacian that is built based on the pairwise similarities of images in the hash function learning process.

II. GROWTH OF DIGITAL IMAGING

The use of images in human communication is hardly painted pictures on the walls of their caves, and the use of maps and building plans to convey information almost certainly dates back to pre-Roman times. But the twentieth century has witnessed unparalleled growth in the number, availability and importance of images in all walks of life. Images now play a crucial role in fields as diverse as medicine, journalism, advertising, design, education and entertainment.

Technology, in the form of inventions such as photography and television, has played a major role in facilitating the capture and communication of image data. But the real engine of the imaging revolution has been the computer, bringing with it a range of techniques for digital image capture, processing, storage and transmission which would surely have startled even pioneers like John Logie Baird. The involvement of computers in imaging can be dated back to 1965, with Ivan Sutherland's Sketchpad project, which demonstrated the feasibility of computerised creation, manipulation and storage of images, though the high cost of hardware limited their use until the mid-1980s. Once computerised imaging became affordable, it soon penetrated into areas traditionally depending heavily on images for communication, such as engineering, architecture and medicine. Photograph libraries, art galleries and museums, too, began to see the advantages of making their collections available in electronic form.

III. THE NEED FOR IMAGE DATA MANAGEMENT

The process of digitisation does not in itself make image collections easier to manage. Some form of cataloguing and indexing is still necessary – the only difference being that much of the required information can now potentially be derived automatically from the images themselves. The extent to which this potential is currently being realized is discussed below.

The need for efficient storage and retrieval of images – recognized by managers of large image collections such as picture libraries and design archives for many years. After examining the issues involved in managing visual information in some depth, the participants concluded that images were indeed likely to play an increasingly important role in electronically-mediated communication. However, significant research advances, involving collaboration between a number of disciplines, would be needed before image providers could take full advantage of the opportunities offered. They identified a number of critical areas where research was needed, including data representation, feature extractions and indexing, image query matching and user interfacing.

One of the main problems they highlighted was the difficulty of locating a desired image in a large and varied collection. While it is perfectly feasible to identify a desired image from a small collection simply by browsing, more effective techniques are needed with collections containing thousands of items. Journalists requesting photographs of a particular type of event, designers looking for materials with a particular colour or texture, and engineers looking for drawings of a particular type of part, all need some form of access by image content.

IV. CHARACTERISTICS OF IMAGE QUERIES

Based on query of a user, Access to a desired image from a repository might thus involve a search for images depicting specific types of object or scene, evoking a particular mood, or simply containing a specific texture or pattern. Potentially, images have many types of attribute which could be used for retrieval, including:

- the presence of a particular combination of colour, texture or shape features (e.g. green stars);
- the presence or arrangement of specific types of object (e.g. chairs around a table);
- the depiction of a particular type of event (e.g. a football match);
- the presence of named individuals, locations, or events (e.g. the Queen greeting a crowd);
- subjective emotions one might associate with the image (e.g. happiness);
- Metadata such as who created the image, where and when.

Each listed query type (with the exception of the last) represents a higher level of abstraction than its predecessor, and each is more difficult to answer without reference to some body of external knowledge. This leads naturally on to a classification of query types into three levels of increasing complexity

Level 1 comprises retrieval by primitive features such as colour, texture, shape or the spatial location of image elements. Its use is largely limited to specialist applications such as trademark registration, identification of drawings in a design archive, or colour matching of fashion accessories.

Level 2 comprises retrieval by logical features, involving some degree of logical inference about the identity of the objects depicted in the image.

Level 3 comprises retrieval by abstract attributes, involving a significant amount of high-level reasoning about the meaning and purpose of the objects or scenes depicted.

V. CBIR

Hard information on the effectiveness of automatic CBIR techniques is difficult to come by. Few of the early systems developers made serious attempts to evaluate their retrieval effectiveness, simply providing examples of retrieval output to demonstrate system capabilities. System developers do now generally report effectiveness measures such as precision and recall with a test database, though few discuss subjective measures of user satisfaction. In the absence of comparative retrieval effectiveness scores measuring the effectiveness of two different systems on the same set of data and queries, it is difficult to draw many firm conclusions. All that can be said is that retrieval effectiveness scores reported on image retrieval systems are in the same ball park as those commonly reported for text retrieval.

However, the main drawback of current CBIR systems is more fundamental. It is that the only retrieval cues they can exploit are primitive features such as colour, texture and shape. Hence current CBIR systems are likely to be of significant use only for applications at level 1. This restricts their prime usefulness to specialist application areas such as fingerprint matching, trademark retrieval or fabric selection. IBM's QBIC system has been applied to a variety of tasks, but seems to have been most successful in specialist areas such as colour matching of items in electronic mail-order catalogues, and classification of geological samples on the basis of texture.

Region-Based Automatic Image indexing

Automatic image indexing is usually regarded as a classification problem. Given a keyword, each image is predicted to be positive or negative according to whether it is associated with the concept. Therefore, almost all classification methods can be applied here. Several works treat annotation as translation from image instances to keywords. The translation paradigm is typically based on some model of image and text co-occurrences. The translation approach is extended to models that ascertain associations indirectly through latent topic/aspect/context spaces. Despite its appealing structure, this class of models remains sensitive to the choice of topic model, initial parameters, prior image segmentation, and more importantly the inference and learning approximations to handle the typically intractable exact analysis. Cross media relevance models, continuous relevance model, and multiple Bernoulli relevance model assume different, nonparametric density representations of the joint word-image space. In particular, MBRM achieves a robust annotation performance using simple image and text representations: a mixture density model of image appearance that relies on regions extracted from a regular grid, thus avoiding potentially noisy segmentation, and the ability to naturally incorporate complex word annotations using multiple Bernoulli models. However, the complexity of the kernel density representations can be an obstacle for large-scale application.

VI. SEMANTIC HASHING

Semantic hashing is as effective technique in generating compact binary codes, there is a problem that receives little attention and few efforts have been paid to exploit it. In SH, as well as other hashing methods, the Euclidean distance is usually used to measure the distance between two samples, and then the Euclidean distance is converted to similarity via a Gaussian function to construct the graph Laplacian. However, the Euclidean distance may not accurately reflect the inherent distribution of the images. In addition, when converting the distance to similarity, the radius parameter is sensitive and it is often hard to tune it to its optimal value. A single layer of binary features may not the best way to capture the structure in the count data. It can be described by efficient way to learn additional layers of binary features. After learning the first layer of hidden features, an undirected model that defines p(v,h). It is a complicated, non-factorial prior on h that is defined implicitly by the weights. This peculiar decomposition into p(h) and p(vjh) suggests a recursive algorithm: keep the learned p(vjh) but replace p(h) by a better prior over h, i.e. a prior that is closer to the average, over all the data vectors, of the conditional posterior over h

It can sample from this average conditional posterior by simply applying p(hjv) to the training data. The sampled h vectors are then the "data" that is used for training a higher-level RBM that learns the next layer of features. It could be initialize the higher-level RBM model by using the same parameters as the lower-level RBM but with the roles of the hidden and visible units reversed. This ensures that p(v) for the higher-level RBM starts out being exactly the same as p(h) for the lower-level one. Provided the number of features per layer does not decrease, show that each extra layer increases a variational lower bound on the log probability of the data.

The directed connections from the first hidden layer to the visible units in the final, composite graphical model are a consequence of the fact that we keep the p(vjh) but throw away the p(h) defined by the first level RBM. In the final composite model, the only undirected connections are between the top two layers, because we do not throw away the p(h) for the highest-level RBM.

The first layer of hidden features is learned using a constrained Poisson RBM in which the visible units represent word counts and the hidden units are binary. All the higher-level RBM's use binary units for both their hidden and their visible layers. Each layer is updated by update rules. This greedy, layer-by-layer training can be repeated several times to learn a deep, hierarchical model in which each layer of features captures strong high-order correlations between the activities of features in the layer below. To suppress noise in the learning signal, we use the real-valued activation probabilities for the visible units of all the higher-level RBM's, but to prevent hidden units from transmitting more than one bit of information from the data to its reconstruction, the pretraining always uses stochastic binary values for the hidden units.

The variational bound does not apply if the layers get smaller, as they do in an auto encoder, but, as we shall see, the pretraining algorithm still works very well as a way to initialize a subsequent stage of fine-tuning. The pretraining finds a point that lies in a good region of parameter space and the myopic fine-tuning then performs a local gradient search that finds a nearby point that is considerably better.

Recursive greedy learning of the deep generative model:

– Learn the parameters h1 = (W1,k1,b1) of the Constrained Poisson Model.

– Freeze the parameters of the Constrained Poisson Model and use the activation probabilities of the binary features, when they are being driven by training data, as the data for training the next layer of binary features.

– Freeze the parameters h2 that define the 2nd layer of features and use the activation probabilities of those features as data for training the 3rd layer of binary features.

– Proceed recursively for as many layers as desired.

After pretraining, the individual RBM's at each level are "unrolled" to create a deep autoencoder. If the stochastic activities of the binary features are replaced by deterministic, real-valued probabilities, we can then backpropagate through the entire network to fine-tune the weights for optimal reconstruction of the count data. For the fine tuning, we divide the count vector by the number of words so that it represents a probability distribution across words. Then we use the cross-entropy error function with a "softmax" at the output layer. The fine-tuning makes the codes in the central layer of the autoencoder work much better for information retrieval.

During the fine-tuning, we want backpropagation to find codes that are good at reconstructing the count data but are as close to binary as possible. To make the codes binary, we add Gaussian noise to the bottom–up input received by each code unit.

Assuming that the decoder network is insensitive to very small differences in the output of a code unit, the best way to communicate information in the presence of added noise is to make the bottom–up input received by a code unit be large and negative for some training cases and large and positive for others.

To prevent the added Gaussian noise from messing up the conjugate gradient fine-tuning, we used "deterministic noise" with mean zero and variance 16, chosen by cross-validation. For each training case, the sampled noise values are fixed in advance and do not change during training. With a limited number of training cases, the optimization could tailor the parameters to the fixed noise values, but this is not possible when the total number of sampled noise values is much larger than the number of parameters.

To speed-up the pretraining, we subdivided both datasets into small mini-batches, each containing 100 cases,2 and updated the weights after each mini-batch. For both datasets each layer was greedily pretrained for 50 passes (epochs) through the entire training dataset. The weights were updated using a learning rate of 0.1, momentum of 0.9, and a weight decay of 0.0002 _ weight _ learning rate. The weights were initialized with small random values sampled from a zero-mean normal distribution with variance 0.01.

For fine-tuning we used the method of conjugate gradients3 on larger mini-batches of 1000 data vectors, with three line searches performed for each mini-batch in each epoch. To determine an adequate number of epochs and to avoid overfitting, we fine-tuned on a fraction of the training data and tested performance on the remaining validation data. We then repeated fine-tuning on the entire training dataset for 50 epochs. Slight overfitting was observed on the 20-newsgroups corpus but not on the Reuters corpus. After fine-tuning, the codes were thresholded to produce binary code vectors. The asymmetry between 0 and 1 in the energy function of an RBM causes the unthresholded codes to have many more values near 0 than near 1, so we used a threshold of 0.1.