2. Literature Review

This chapter reviews other researches related to image retrieval, including content-based and region-based image retrieval, visual features and visual words for image representation, image annotation for extracting semantic information from images, and image matching and relevance feedbacks.

2.1. Image Retrieval

Content-based image retrieval has become a very active research area since the 1990's due to the rapid increase in the use of digital images. Many image retrieval systems have been designed in the past. This section describes several well-known works for content-based and region-based image retrieval.

2.1.1. Content-based image retrieval

The goal of content-based image retrieval (CBIR) is to retrieve the desired images for a user from a large image database, based on the image contents [Gudivada and Raghavan 95] [Zachary and Iyengar 99]. Many CBIR systems have been built, and surveys on these systems can be found in [Datta et al. 05] [Lew et al. 06] [Rui et al. 99] [Zachary and Iyengar 99] [Schettini et al. 01] [Castelli and Bergman 01]. Here we briefly describe some of the well-known CBIR systems.

QBIC

QBIC (Query by Image Content) [Flickner et al. 95] is the first commercial CBIR system, which allows the user to submit a query that contains color, texture,

shape, user sketches, and spatiotemporal features specified by an interface tool. The architecture of the QBIC system is organized into two main components: the database population and the database query. The database population extracts visual features from images and videos and from an index for database storage and retrieval. After the user has graphically composed the query, the database query component extracts features from the query and computes a distance metric on the indices in the database.

MARS

MARS (Multimedia Analysis and Retrieval System) [Rui et al. 97] provided an integrated multimedia information retrieval and database management infrastructure. MARS supports multimedia information as first-class objects suited for storage and retrieval based on their content. Research in the MARS project is categorized into the following four areas: multimedia content representation, multimedia information retrieval, multimedia feature indexing, and multimedia database management.

PicHunter

PicHunter [Cox et al. 96] [Cox et al. 00] is a prototype CBIR system developed at NEC Research Institute at Princeton. PicHunter represented a simple instance of a general Bayesian framework for using relevance feedback to direct a search. Also, PicHunter makes use of hidden annotation rather than a possibly inaccurate or inconsistent annotation structure that the user must learn and make queries in.

PicSOM

PicSOM [Laaksonen et al. 02] is a CBIR system which is based on a neural network algorithm called the Self-Organizing Map (SOM) to organize images into map units in a two-dimensional grid so that similar images are located near each other. A tree-structured version of the SOM algorithm (TS-SOM) is used to create a

hierarchical indexing of the image database.

iFind

iFind [Zhang et al. 00] is a web-based image retrieval system developed at Microsoft Research China. It provided the functions of text-based image search, querying by image example, and their combination. Images in the database are indexed by their low-level (visual) features, high-level (semantic) features collected from the web, and annotations if they are available.

NETRA

NETRA [Ma and Manjunath 99] used color, texture, shape and spatial location information in segmented image regions to search and retrieve similar regions from the database. A distinguishing aspect of this system is its incorporation of a robust automated image segmentation algorithm that allows object or region based search.

Photobook

Photobook [Pentland et al. 94] is a tool developed at MIT Media Laboratory for performing queries on image databases based on image content. It works by comparing features associated with images, not the images themselves. These features are in turn the parameter values of particular models fitted to each image. These models are commonly color, texture, and shape, though Photobook will work with features from any model. Features are compared using one out of a library of matching algorithms that Photobook provides, inclusive of Euclidean, Mahalanobis, divergence, vector space angle, histogram, Fourier peak, and wavelet tree distances, and their linear combinations.

VisualSEEK

VisualSEEk [Smith and Chang 96] is a web tool for searching for images and

videos developed at Columbia University. VisualSEEk supports queries based on both visual features and their spatial relationships. Main research features are spatial relationship query of image regions and visual feature extraction from compressed domain. The visual features used in their systems are color set and wavelet transform based texture feature. To speed up the retrieval process, they also developed binary tree based indexing algorithms.

SIMPLIcity

SIMPLIcity (Semanticssensitive Integrated Matching for Picture LIbraries) [Wang et al. 01] was an image retrieval system using the region-based approach. This retrieval system designed semantics classification methods, a wavelet-based approach for feature extraction, and integrated region matching (IRM) based upon image segmentation. The IRM was developed using a region-matching scheme that integrates properties of all the regions in the images in order to reduce the effect of the badly image segmentation.

Blobworld

Blobworld [Carson et al. 02] is also a region-based image retrieval system. It designed a region-based image representation, called Blob, which provides a transformation from the raw pixel data to a small set of image regions. Blob was created by clustering pixels in a joint color-texture-position feature space. Blobworld provided the function that allows the user to view the internal representation of the submitted image and the query results.

2.1.2. Region-based image retrieval

Region-based image retrieval (RBIR) is a special type of CBIR. In an RBIR

system, image regions, parts of an image with relatively homogeneous subjects or features, play the roles of kernels, and are used to index images in most RBIR systems. RBIR systems can be categorized as two types according to the chosen query format: whole-image-as-query (WIQ) and image-region-as-query (IRQ). [Chiang et al. 04]

In a WIQ RBIR, the user provides the example image, and the system extracts information from the whole image for performing the query. NETRA [Ma and Manjunath 99] used color, texture, shape and spatial location information in segmented image regions to search and retrieve similar regions from the database. J. Z. Wang et al. developed SIMPLIcity [Wang et al. 01] that employs Integrated Region Matching (IRM) to measure the distance between two images. IRM aims on reducing the influence of image segmentation. Smith and Li decomposed an image into regions and represented it as a region string [Smith and Li 99]. The string was then converted to the composite region template (CRT) descriptor matrices that provide the relative ordering of symbols. K. Barnard and N. V. Shirahatti proposed a CBIR system for modeling the joint probability of image region features and associated texts [Barnard and Shirahatti 03].

In an IRQ RBIR, the user performs a query by choosing regions from the example image according to their requirements, and then the RBIR system returns images having regions that are similar to the query regions. In Blobworld [Carson et al. 02], each region of an image is considered a blob associated with color and texture descriptors. Users can specify the attributes of some specific regions as the query, rather than providing a description of the entire image. K. Vu et al. proposed a similarity model for noise-free queries where the noise represents the irrelevant regions of an image [Vu et al. 01]. They also discussed how their approach handles the translation and scaling matching.

2.2. Visual Features

Many types of visual features have been proposed for characterizing image contents, including color, texture, and shape features. Most image retrieval systems perform feature extraction as a preprocessing step. This section describes visual features that are used to implement our tasks, containing color histogram and moments, Gabor texture, and SIFT descriptor.

2.2.1. Color histogram and color moments

Color histogram [Swain and Ballard 91] shows the distribution of colors in an image, where each histogram bin represents a color in the adopted color space. The use of the color histogram first requires the color space to be chosen (e.g. HSV, LAB, LUV, etc). Let K_1, K_2, and K_3 be the number of bins used to quantize the three color channels. The *color histogram* (*CH*) is defined as an $K_1 \times K_2 \times K_3$ -dimensional feature vector,

$$CSH = \{h_{ijk} \mid 1 \le i \le K_1, 1 \le j \le K_2, 1 \le k \le K_3\}, \tag{2.1}$$

where each h_{ijk} value in the histogram corresponds to the number of pixels having the values in color channels.

Stricker and Orengo proposed using the *color moments* (*CM*) approach to overcome the quantization effects in the color histogram [Stricker and Orengo 95]. Let x_i be the value of pixel x in the i-th color component, and N be the number of pixels in the image. The first- and second-order color moments of an image can be defined as:

$$CM = (\mu_1, \mu_2, \mu_3, \sigma_1, \sigma_2, \sigma_3), \tag{2.2}$$

where $\mu_i = \frac{1}{N}\sum_{x=1}^{N} x_i$ and $\sigma_i = \frac{1}{N}\sum_{x=1}^{N}(x_i - \mu_i)^2$, i = 1, 2, or 3.

2.2.2. Gabor texture

To extract the Gabor texture feature of an image I, the image is first filtered with a bank of scale and orientation quantization Gabor filters, and then the means and standard deviations of the outputs of the filters are computed. Formally, filtering an image $I(x, y)$ with Gabor filter g_{mn}, in [Manjunath et al. 01], is

$$W_{mn}(x, y) = \int\int I(x, y) \cdot g^{*}_{mn}(x - x_1, y - y_1) dx_1 dy_1, \tag{2.3}$$

where $0 \le m \le \kappa$, $0 \le n \le \tau$, and there are κ+1 and τ+1 scale and orientation quantizations, respectively. All means and standard derivations of the magnitude $|W_{mn}(x, y)|$,

$$\begin{aligned} \mu_{mn} &= \int\int |W_{mn}(x, y)| dxdy \text{ and} \\ \sigma_{mn} &= (\int\int (|W_{mn}(x, y)| - \mu_{mn}(x, y))^2 dxdy)^{1/2}, \end{aligned} \tag{2.4}$$

are computed to form the Gabor texture feature, which is denoted as

$$G(I) = \{\mu_{00}, \sigma_{00}, ..., \mu_{\kappa\tau}, \sigma_{\kappa\tau}\}. \tag{2.5}$$

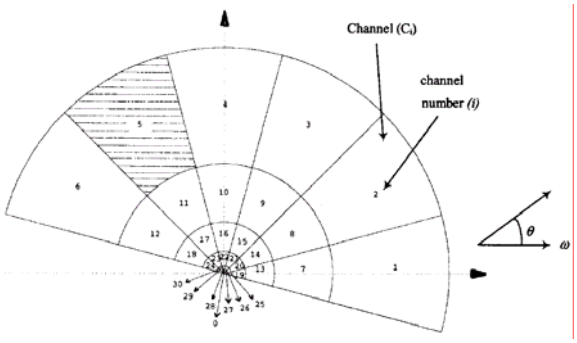

Figure 2-1. Illustration of Gabor texture with 5 scales and 6 orientations.

Figure 2-1 illustrates and example of Gabor texture, which is taken from [Manjunath et al. 01] by quantizing scale and orientation with 5 and 6, respectively. Thus, this example yields the Gabor texture with 5x6x2=60 dimensions. In our implementation, we set κ=5 and τ=3, and hence there are 6×4×2=48 dimensions in the Gabor texture space.

2.2.3. SIFT descriptor

A Scale Invariant Feature Transform (SIFT) descriptor [Lowe 99] [Lowe 04] is a set of distinctive invariant features extracted from images that can be used to perform a reliable matching between different views of an object or scene. In the extraction of the SIFT descriptor, four steps are mainly included: (i) detection of scale-space extreme, (ii) accurate keypoint localization, (ii) orientation assignment, and (iv) local image descriptor. Figure 2-2 shows an example taken from [Lowe 04]. The example shows a 2 × 2 descriptor array computed from an 8 × 8 set of samples.

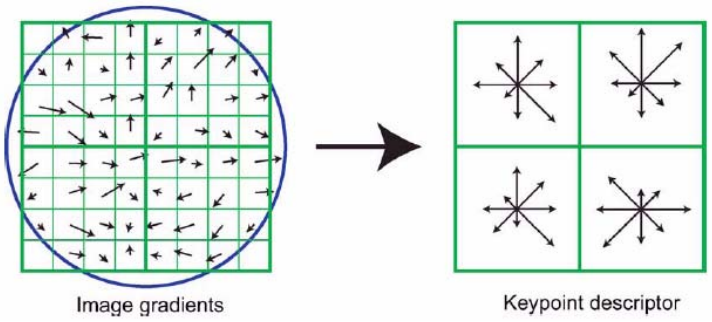

Figure 2-2. An example for the SIFT descriptor. The left part is the gradients of the image, and then these gradients are accumulated over 4x4 subregions which are shown as the right part.

2.3. Region and Visual Words

While visual features are extracted from an image, how to represent an image by use of the extracted features is also important, special for region-based image retrieval. Our proposed image representation is based on the model of visual words, which consider an image consists of the bag of visual words, and visual words are built based on the region features in the feature space. Thus, this section provides the reviews of image segmentation and visual words.

2.3.1. Image segmentation

The goal of image segmentation is to partition an image into a set of region according to some criteria. Different methods for image segmentation have been applied to region-based tasks, e.g., retrieval, image annotation, object recognition, for different goals. The most intuitive method for image segmentation is to segment objects (or foreground subjects) from an image for region-based image matching [Barnard and Forsyth 01] [Carson et al. 02] [Jeon03] [Jing et al. 04] [Wang et al. 01], even though this is very difficult. Figure 2-3 shows an example of segmentation that is used in Blobworld [Carson et al. 02]. The segmentation results greatly affect the performances of region-based tasks, but, in general, it is still an open problem to automatically segment images. Hence, some researchers divided an image into rectangular girds [Feng et al. 04] [Maree et al. 05] or a large number of overlapping circular regions [Fergus et al. 05] [Sivic et al. 05].

Figure 2-3. An example of image segmentation in Blobworld system.

In general, we cannot perform a perfect segmentation for an image because we do not know what the perfect is. Hence, the selection of segmentation methods must be dependent of the characteristics of the application. Moreover, it is also important to design a good representation and matching scheme for images in order to cushion the effect of the imperfect segmentation.

2.3.2. Visual words

The original concept of visual words is derived from the text analysis of documents. The text terms (i.e., words) are appropriate for indexing and analyzing documents in the traditional information retrieval. However, it is difficult to find a proper units for better representing images. Thus, some researchers viewed visual features that are extracted from an image as a type of “word” in the compound of the image. The role of visual words in images is similar to that of text items in documents. Figure 2-4 depict the application of the visual word, including image understanding and retrieval, including object recognition [Duygulu et al. 02] [Fergus et al. 05] [Sivic et al. 05], image annotation [Jeon03] [Feng et al. 04] [Barnard and Forsyth 01], image retrieval [Carson et al. 02] [Jing et al. 04], and video retrieval [Sivic and Zisserman 04] [Feng et al. 04].

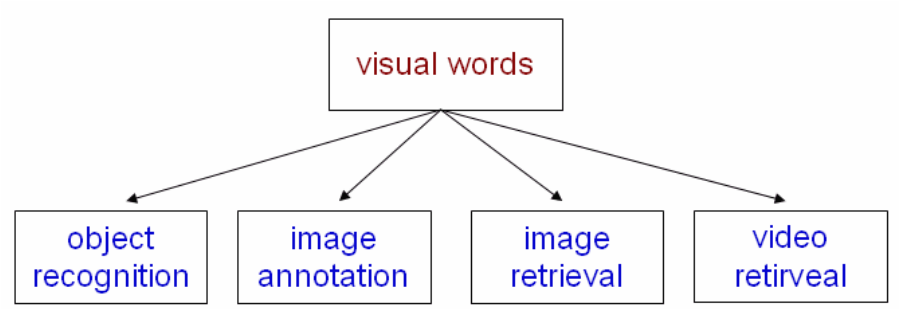

Figure 2-4. Applications of the visual word.

Carson et al., in Blobworld system, defined a blob which is associated with a set of region features [Carson et al. 02]. Blob can be regarded as a type of visual words. An image was first segmented into several object regions, and each region was considered to be a blob. The blob involved two components: (i) the region that is a part of the image and (ii) the visual features that are extracted from the region. Therefore, an image can be regarded as the compound of the blobs. Barnard and Forsyth proposed a statistical model for learning the relationships between textual keywords of images and visual features of blobs [Barnard and Forsyth 01].

The most common approach to generate visual words is to extract a set of region features from the images and quantize these feature vectors into a pre-built vocabulary of visual words, e.g., in [Csurka et al. 04], [Fei-Fei and Perona 05], [Fergus et al. 05], and [Sivic et al. 05]. Simply speaking, the construction of visual words is to quantize or cluster (most using *K*-means clustering) region features in the feature space. Wang et al. designed SIMPLIcity system, which is a famous region-based image retrieval system described in Section 2.1, by performing *K*-means clustering for region features [Wang et al. 01]. Jing et al. employed the Generalized Lloyd Algorithm (GLA) to quantize feature vectors into a set of codewords [Jing et al. 04], which is similar to visual words. Ye et al. applied Minimum-Entropy Based Discretization, which was

first proposed in [Fayyad and Irani 93] to discrete the real-valued visual features, to extract the visual words [Ye et al. 05]. Lin et al. designed a model of “color naming” that divided the color space (in HSV) into 35 colors [Lin et al. 04]. These 35 colors represented images more semantically than the original color space did.

2.4. Image Annotation

The goal of image annotation is to assign several labels with an image. In computer vision, only visual features can be directly extracted from an image. How to model the relationship between semantic labels and visual features for an image is the main issue in image annotation. Many state-of-the-art works for image annotation and concept detection were reviewed in [Datta et al. 05]. In this section, we introduce several important works of image annotation.

Duygulu et al. considered the annotation task as a translation problem [Duygulu et al. 02]. Their work designed a translation model, between blobs segmented and extracted from images and semantic labels, to translate images to labels. This work also provided a benchmarking dataset that is widely used in many researches for evaluating the performance of image annotation.

Relevance model [Lavrenko and Croft 01] is another type of approaches to the problem of image annotation. The basic idea of relevance model is to learn a statistical model between feature vectors (or visual words) and annotation labels. Based on the idea, there have been several famous works to design different statistical models for annotation, e.g., Cross-Media Relevance Model (CMRM) [Jeon et al. 03], Multiple Bernoulli Relevance Model (MBRM) [Feng et al. 04], and Image-Keyword Document Model (IKDM) [Zhou et al. 05].

Most of the above approaches use the co-occurrence of image regions and labels. Blei and M. Jordan proposed a graphical probabilistic model of the dependence of annotation words on image regions [Blei and Jordan 03]. Carneiro and Vasconcelos formulated the image annotation as a supervised learning problem [Carneiro and Vasconcelos 05]. Srikanth et al. proposed methods to use a hierarchy defined on the annotation labels derived from a textual ontology to improve automatic image annotation and retrieval [Srikanth et al. 05]. Chang et al. designed an approach called soft annotation to give images a confidence level for each trained semantic label [Chang et al. 03].

Most of these approaches are based on supervised classification, that is to say, a large amount of training images is necessary to avoid overfitting. Unfortunately, it is often difficult to collect much enough data for training, for example, it is a hard task to manually annotate thousands of images. Hence we design a semi-supervised learning approach by integrating labeled and unlabeled images to overcome the problem.

2.5. Image Matching

The basic idea to rank and match image in image retrieval is to define a distance or a similarity between two images based on the representing features of images. There have been a large number of fundamentally different approaches proposed in the past. In comparing two histograms that have the same number of bins, Minkowski- and quadratic-form distances [Smith 97] are widely used in many researches. On the other hand, Earth Mover's Distance (EMD) [Rubner et al. 00] that measures the minimal cost that must be paid to transform one distribution to another

is appropriated for comparing two variable-length distributions.

Besides, many researches are also worth discussing. A number of image retrieval tasks were proposed based on Bayesian probabilistic framework, e.g., in [Cox et al. 00] Vasconcelos 00] [Zhang et al. 06]. J. Z. Wang et al. proposed integrated region matching (IRM) approach that aims to reduce the influence of segmentation variation because current segmentation approaches are less than perfect [Wang et al. 01]. The use of the MPEG-7 descriptors to train self-organizing maps (SOM) for image retrieval has been developed [Laaksonen et al. 02]. Support vector machine (SVM) is also applied to design the image retrieval in [Jing et al. 04] [Tong and Chang 01].

2.6. Learning in Relevance Feedback

Relevance feedback is a query modification technique that attempts to capture the user's precise needs through iterative feedbacks and query refinement [Buck 95] [Rui et al. 98] [Datta et al. 05] [Zhou and Huang 03]. In the typical information retrieval of documents, relevance feedback is the strategy of automatically altering the existing query using information supplied by users about the relevance of previously retrieved documents. It has been shown good improvements in precision for test document collections when relevance feedback is used.

Recently, more researches for the content-based image retrieval have paid increased attention to study the relevance feedback techniques [Cox 96] [Rui 97] [Jing et al. 04] [Vasconcelos 00]. The performance of content-based image retrieval being unsatisfactory for many practical applications is mainly due to the gap between the high-level semantic concepts and the low-level visual features. Unfortunately, the contents of images for general-purpose retrieval are much subjective and personal.

Research on relevance feedback has focused on adaptively refining a user's initial query to more accurately select the desired data, trying to bridge the gap and improve the retrieval performance.

The standard process in relevance feedback can be divided into two steps. The first step is that the system retrieves the most similar images for the query images of the previous stage, and the second one is that the user assigns relevant/irrelevant images from the retrieved results to be the query of the next stage. Then, the user feedbacks, either relevance or irrelevance, allow that the retrieval system interactively learns what the user requests are in order to refine the retrieval results in image retrieval.

An intuitive approach to relevance feedback is to warp or tune the feature spaces according to the user's relevant and irrelevant examples [Atmosukarto et al. 05] [Rui et al. 98] [Zhang and Chen 02]. This approach, that is to say, performs the scheme of query movement to tune the query by use of the user feedbacks in the retrieval. How to warp the feature spaces, or how to weight the query movement, is the key point for this approach. However, this approach does not consider the semantic information. Hence, the warping of feature spaces cannot fully reduce the semantic gap between human's perceptions embedded in the query and the visual features extracted in images.

Regarding the interactive process of relevance feedback, the user provides the positive and negative feedbacks, and then the system try to learn what the user wants. Hence, it is reasonable to construct a learning system for relevance feedback of image retrieval. Many types of learning models have been applied in relevance feedback for image retrieval, such as Bayesian framework [Cox et al. 00] [Su et al. 03] [Vasconcelos 00], SVM [Jing et al. 04], and active learning [Tong and Chang 01]

[Zhang and Chen 02]. Also, Goh et al. described some characteristics of the learning problem for relevance feedback [Goh et al. 04].