CHAPTER 2

REVIEW OF LITERATURE

The last decade has witnessed lots of interest in research on content based image retrieval. This has paved the way for many new techniques and systems, and a growing interest in related fields to support such systems. The text-based searching approaches can effectively retrieve images without understanding the content. But it is very difficult for the users to give a low-level description of desired image. Even if the user provides an example image with similar content, most current algorithms fail to precisely relate its high level concept, or the semantics of the image, to its lower level content. The problem with these algorithms is their dependence on visual similarity in deciding the semantic similarity. Moreover, semantic similarity is a subjective measure.

CBIR uses visual contents of an image to search the desired images. It deals with the fundamental problem of mathematically describe the image content (image signature) and then, assessing the similarity between a pair of images based on their signatures. In spite of the apparent simplicity of this, there are significant obstacles that need to be overcome in order to design an efficient CBIR system.

It is important to find a good image signature using suitable image features for the design of an efficient CBIR system. Most of the CBIR approaches rely on a preprocessing step of feature extraction, which aims to extract suitable image features (descriptors) such as color, texture, shape and spatial layout, that carries enough information to allow successful retrieval of relevant images from a database containing thousands or millions of images.

It is observed that, the problem of searching images according to these visual signatures is very challenging since there are many factors affecting the retrieval performance, like resolution, illumination variations and occluded objects. This motivates the need for an intermediate representation such as Visual Word and Vocabulary. Visual Word is a small part of an image which carries some information related to these features.
Several techniques have been proposed which make use of visual vocabulary for
the creation of model vector and for finding content similarity, such as a technique to
map each visual region to a weighted set of words is explored in [141]. In this technique
the set of visual words is obtained by selecting words based on proximity in descriptor
space. P Duygul has proposed an approach in which the images are first segmented into
regions; regions are classified into visual words, using variety of features. Then a
mapping between visual words and keywords is learned using an EM method [143]. In
2003, J. Lee and J. Wang has proposed an approach for the linguistic indexing of images
that uses Wavelets to extract image features and Hidden Markov Models (HMMs) to
learn the association of those features to the keywords describing the images [146].

When the retrieval process considers only the model vectors that represent the
visual content of images, it sometimes fails to produce accurate results. This is because
the approach based on visual vocabulary, totally ignores the geometry of the extracted
visual words. Two images can contain similar visual words, but in a totally different
spatial layout from the other [140]. Thus, the inclusion of a geometry consistence check
would be very useful. The RANSAC (Random Sample Consensus) algorithm is popularly
used to preserve the geometric consistency during image retrieval [140].

The RANSAC (Random Sample Consensus) algorithm is a simple, yet powerful,
technique that is usually applied to the task of estimating the parameters of a model,
using data which is contaminated by outliers. RANSAC estimates a global relation that
fits the data, while simultaneously classifying the data into inliers (points consistent with
the relation) and outliers (points not consistent with the relation). Due to its ability to
tolerate a large fraction of outliers, the algorithm is a popular choice for a several robust
estimation problems.

Even after considering these factors in the design of CBIR system, it sometimes
fails to precisely retrieve desired images. Then it is useful to refine the results using the
feedback given by the user for the retrieved results in multiple iterations. Relevance
feedback is a process in which user feedback is important. An image retrieval system
presents a ranked set of images which are relevant to the user’s initial query and then
iteratively solicits the user for feedback on the relevance of images and uses the feedback to compose and improve the retrieval result.

In recent years many commercial systems such as WebSeek [91], Netra [80] and experimental system such as MIT’s photo book [85], WBIIS [92] have been proposed and used for visual search. These systems are based on the centralized computing model. One of the limitations of the centralized CBIR system is that, the feature extraction, indexing, and querying are done in a centralized manner, which can be computationally expensive, and is difficult to scale up.

The worldwide infrastructure of computers and networks created an exciting opportunity for collecting vast amounts of data and for sharing computers and resources on an unprecedented scale. In the last few years, the emerging Peer-to Peer (P2P) model has become a very powerful and attractive paradigm for developing Internet-scale file systems [74, 76] and sharing resources. This motivates the design of CBIR in P2P network.

This chapter focuses on major research issues addressed in this work. The phases involved in the design process of an efficient P2P-CBIR system include feature extraction and representation, preservation of geometric consistency, query modification (relevance feedback), peer clustering and query routing in P2P network. The detailed literature review of all these phases is discussed in this chapter. Current research directions are also studied and their shortcomings are highlighted. A Summary of research contribution is given with their justification and implication.

2.1. Text Based Image Retrieval

Early techniques for image retrieval were not based on visual features but on the textual annotation of images. Images were first annotated with text and then searching is carried out using a text-based approach of conventional database management systems. Comprehensive surveys of early text-based image retrieval approaches can be found in [161, 162]. Through text descriptions, images can be organized by semantic hierarchies for easy navigation and browsing based on standard Boolean queries. Since automatically generating descriptive texts for a widespread spectrum of images is not feasible, most
text-based image retrieval systems relies on manual annotation of images. Annotating the images manually is an expensive task for large image databases, and it is often subjective, context-sensitive and incomplete.

As the difficulties faced by text-based retrieval became more and more severe. The efficient management of the rapidly growing visual information became an urgent problem. This motivated the need for content-based image retrieval techniques. It was observed that a more efficient and natural way to represent and index visual information would be based on properties that are inherent in the images themselves (visual features). Researchers in the area of computer vision, database management, human-computer interface, and information retrieval were attracted to this field. Since then, research on content-based image retrieval has developed speedily [11, 23, 24, 35, 49, 50]. Since 1997, the number of research publications on the techniques of visual information extraction, organization, indexing, user query and interaction, and database management has immensely increased. Similarly, a large number of academic and commercial retrieval systems have been developed by universities, government organizations, companies, and hospitals. A brief survey of these techniques is given in the following sections.

2.2 Visual Information Extraction and Representation

Content-based image retrieval uses the visual contents of an image to represent and index the images. The visual contents of the images in the database are extracted and described by visual content descriptor. The feature vectors of the images in the database forms a feature database. The system then retrieves the desired images based on the similarity of these extracted feature vectors.

A good visual feature should be invariant to the accidental variance introduced by the imaging process. However, there is an inverse relation between the invariance and the discriminative ability of visual features, since a very wide class of invariance loses the ability to discriminate between important differences.

A visual content descriptor can be either global or local. A global descriptor uses the visual features of the whole image, whereas a local descriptor uses the visual features of regions or objects to describe the image content. To obtain the local visual descriptors, an image is usually divided into parts and then features are extracted from each part.
locally. The easiest way of dividing an image is to partition it, which cuts the image into tiles of equal size and shape. A simple partition does not generate perceptually meaningful regions, but it is a way of representing the global features of the image at a finer resolution. A better method is to divide the image into homogenous regions according to some criterion, using region segmentation algorithms that have been rigorously investigated in computer vision. A more complex way of dividing an image, is an object segmentation that obtain semantically meaningful objects (like ball, car, horse).

Most Content based image retrieval systems perform feature extraction as a preprocessing step, generating global image features like color histogram or local descriptors like shape and texture. A region based color descriptor indexed in 3-D space along with their percentage coverage within the regions is proposed in [6], and observed to be computationally more efficient in similarity based retrieval than conventional color histograms. The authors state that this compact representation is more efficient in terms of search and retrieval, than the high dimensional histograms, and it has also overcome the limitations associated with earlier schemes such as dimensionality reduction and color moment descriptors. In [7], a multi-resolution histogram approach is proposed which captures the spatial image information efficiently and observed to be an effective technique to retrieve images. S. Jeong has proposed a technique in which Gaussian mixture vector quantization (GMVQ) is used to extract color histograms that yield better retrieval than uniform quantization and vector quantization with squared error. A set of color and texture descriptors which are thoroughly tested for inclusion in the MPEG-7 standard, and suitable for natural images and video, is described in [25]. These include histogram-based descriptors, dominant color descriptors, spatial color descriptors and texture descriptors suited for browsing and retrieval.

Shape is a key attribute of segmented image regions, and its efficient and robust representation plays an important role in the retrieval. A shape similarity measure based on discrete curve evolution to simplify contours is discussed in [22]. Contour simplification helps to remove noise and irrelevant shape features from consideration. C. Schmid has proposed a new shape descriptor for shape matching, referred to as shape context [34] which is a compact and robust to a number of geometric transformations.
A Dynamic Programming (DP) approach to shape matching has been proposed in [37]. The limitation of this approach is that the computation of Fourier descriptors and moments is slow, although pre-computation may help to produce real-time results. Use of Dynamic Time Warping (DTW) distance instead of Euclidean distance is observed to be an accurate shape matching technique which presented by I. Bartolini in [17]. For characterizing shape within images, reliable segmentation is important, without which the shape estimates will not be meaningful. Even though the general problem of segmentation in the context of human perception is difficult to solve, there are some interesting new directions, one of the most important approaches is segmentation based on the Normalized Cuts criteria [27]. This approach is based on the theory of spectral clustering and it has been extended to texture image segmentation by using cues of contour and texture differences [25].

Segmentation based on the mean shift procedure [40], multi-resolution segmentation of low depth of field images [28], a Bayesian framework based segmentation involving the Markov chain Monte Carlo technique [30], and an EM algorithm based segmentation using a Gaussian mixture model [21] are the recently proposed important approaches for segmentation.

Features based on local invariance such as interest points approach is used extensively in image retrieval. Scale and affine invariant interest points that can deal with significant affine transformations and illumination changes is observed to be an effective features for image retrieval [34]. Similarly, wavelet-based salient points have been popularly used for retrieval [26]. The significance of such special points lie in their compact representation of important image regions, leading to efficient indexing and good discriminative power, especially in object-based retrieval. A comparative study of different types of color interest points used in image retrieval is given in [33], while a comparative performance evaluation of the various proposed interest point detectors is reported in [34]. When a huge number of image features are available, one way to improve generalization and efficiency in classification and indexing is to work with a feature subset. An automatic feature subset selection for SVMs is one of the most
popularly used algorithms for feature subset selection. A summary of various feature extraction techniques used in CBIR is given in Table 2.1

<table>
<thead>
<tr>
<th><strong>Low Level Descriptor</strong></th>
<th><strong>Features</strong></th>
<th><strong>Approach</strong></th>
<th><strong>Researcher (year)</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduced SIFT</td>
<td>Distance function</td>
<td>Ledwich et. al., (2004)</td>
<td></td>
</tr>
<tr>
<td>SIFT</td>
<td>Bag of Words</td>
<td>Zhou et. al., (2011)</td>
<td></td>
</tr>
<tr>
<td>Color Descriptors</td>
<td>Interactive Genetic Algorithm</td>
<td>Lai et. al., (2009)</td>
<td></td>
</tr>
<tr>
<td>Color Descriptors</td>
<td>Fuzzy Logic</td>
<td>Xiaoling et. al., (2005)</td>
<td></td>
</tr>
<tr>
<td>New Color SIFT Descriptor</td>
<td>Euclidean Distance</td>
<td>Verma et. al., 2010</td>
<td></td>
</tr>
<tr>
<td>Visual Concept and Local Features</td>
<td>SIFT</td>
<td>Bag of Words</td>
<td>Menglin wu et. al., (2012)</td>
</tr>
<tr>
<td>Compact Color Descriptors</td>
<td>Euclidean Distance</td>
<td>Deng et. al., (2001)</td>
<td></td>
</tr>
<tr>
<td>Binary Color Histogram</td>
<td>Similarity Based Retrieval</td>
<td>Utenpattanant et. al., (2006)</td>
<td></td>
</tr>
<tr>
<td>Color Constancy</td>
<td>Bag of Words</td>
<td>Joze et. al., (2010)</td>
<td></td>
</tr>
<tr>
<td>Color Descriptor</td>
<td>DCT (Discrete Cosine Transformation)</td>
<td>Malik et. al., (2012)</td>
<td></td>
</tr>
<tr>
<td>Color Histogram and Global Descriptor</td>
<td>Similarity Based Retrieval</td>
<td>Sharma et. al., (2011)</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1 (a) Summary of Feature Extraction Technique (Color Feature)
<table>
<thead>
<tr>
<th>Low Level Descriptor</th>
<th>Features</th>
<th>Approach</th>
<th>Researcher (year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Statistical and structural texture features</td>
<td>Tamura Texture Feature Extraction Techniques</td>
<td>Fuzzy logic to reduce semantic gap</td>
<td>Kulkarni et. al., [69] (2010)</td>
</tr>
<tr>
<td>Statistical and structural texture features</td>
<td>Statistical and structural texture features</td>
<td>Weighted Feature Identification</td>
<td>Selvarajah et. al., [70] (2011)</td>
</tr>
<tr>
<td>Texture Descriptor</td>
<td>Texture Descriptor</td>
<td>Texture Features Extracted using Co-Occurrence Matrices</td>
<td>Idrissi et. al., [71]</td>
</tr>
<tr>
<td>Texture Descriptor</td>
<td>Texture Descriptor</td>
<td>Edge Statistic</td>
<td>S. Michel et. al., [73] (1994)</td>
</tr>
</tbody>
</table>

Table 2.1 (b) Summary of Feature Extraction Technique (Texture Feature)
<table>
<thead>
<tr>
<th>Low Level Descriptor</th>
<th>Features</th>
<th>Approach</th>
<th>Researcher (year)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>Curvature Scale Space</td>
<td>F. Mokhtarian et. al., (1997)</td>
</tr>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>Template Matching</td>
<td>Sougata et al., (1997)</td>
</tr>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>Shape Matrix and Snake Model</td>
<td>Sheng et. al., (2005)</td>
</tr>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>Fourier Transforms</td>
<td>Michaelortega et. al., (1997)</td>
</tr>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>PIC- SOM</td>
<td>Brandt et. al., (2000)</td>
</tr>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>Canny Edge Detection</td>
<td>Ramamurthy et. al., (2011)</td>
</tr>
<tr>
<td>Contour Region</td>
<td>Shape Descriptor</td>
<td>Comparative Analysis</td>
<td>Zhang et. al., (2002)</td>
</tr>
<tr>
<td></td>
<td>Shape Descriptor</td>
<td>Convex Part</td>
<td>Fumikazu et. al., (1995)</td>
</tr>
<tr>
<td>Edge Co-occurrence Matrix (ECM)</td>
<td></td>
<td>Euclidean Distance.</td>
<td>Rautkorpi et. al., 2004</td>
</tr>
<tr>
<td>Contour</td>
<td></td>
<td>Euclidean Distance.</td>
<td>Arivazhagan et. al., (2007)</td>
</tr>
</tbody>
</table>

Table 2.1 (c) Summary of Feature Extraction Technique (Shape Feature)
<table>
<thead>
<tr>
<th>Low Level Descriptor</th>
<th>Features</th>
<th>Approach</th>
<th>Researcher (year)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Color and Texture Descriptors</td>
<td>Color and Texture Descriptors</td>
<td>Self-Organizing Map, Learning Vector Quantization, PCA</td>
<td>Chang et. al., (2010)</td>
</tr>
<tr>
<td>Color and Texture Descriptors</td>
<td>Color and Texture Descriptors</td>
<td>Semantic Color Names to Same Regions</td>
<td>Liu et. al. (2005)</td>
</tr>
<tr>
<td>Color and Texture Descriptors</td>
<td>Color and Texture Descriptors</td>
<td>Region Saliency</td>
<td>Wang et. al., (2002)</td>
</tr>
<tr>
<td>Color-Texture Descriptor</td>
<td>Color-Texture Descriptor</td>
<td>Decision Tree-Based Learning Algorithm</td>
<td>Liu et. al., (2008)</td>
</tr>
<tr>
<td>Color and Texture Descriptors</td>
<td>Color and Texture Descriptors</td>
<td>Multi-Objective Genetic Algorithm</td>
<td>Tran et. al., 2005</td>
</tr>
<tr>
<td>Color and Texture Descriptors</td>
<td>Color and Texture Descriptors</td>
<td>Multilayer Perceptron</td>
<td>Sajjad et. al., [96] (2012)</td>
</tr>
</tbody>
</table>

Table 2.1 (d) Summary of Feature Extraction Technique (Color-Texture Feature)
<table>
<thead>
<tr>
<th>Low Level Descriptor</th>
<th>Features</th>
<th>Approach</th>
<th>Researcher (year)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Color-Texture-Shape Descriptors</td>
<td>A unified graph graph and a Novel Random Walk Model</td>
<td>Ma et. al., (2010)</td>
</tr>
<tr>
<td></td>
<td>Color Histogram, Gabor and Wavelet</td>
<td>Comparative Analysis of Feature Extraction Technique</td>
<td>Ashok kumar et. al., (2011)</td>
</tr>
<tr>
<td>Color-Texture-Shape Feature</td>
<td>Color-Texture-Shape Descriptors</td>
<td>Cluster-Based Retrieval of Images by Unsupervised Learning (CLUE),</td>
<td>Chen et. al., (2003)</td>
</tr>
<tr>
<td></td>
<td>Color-Texture-Shape Descriptors</td>
<td>Modular Neural Network for Object Detection</td>
<td>Gallo et. al., (2011)</td>
</tr>
<tr>
<td></td>
<td>Color-Texture-Shape Descriptors</td>
<td>Modular Neural Network Based Class Recognition for Robots</td>
<td>Zheng et. al., [109] (2011)</td>
</tr>
</tbody>
</table>

Table 2.1 (e) Summary of Feature Extraction Technique  
(Color-Texture-Shape Feature)
2.3 Object Based Image Retrieval

In real world images, the same object often has different appearances, due to the effects like partial occlusion and deformation. Factors, such as illumination or 3D viewpoint change, make the retrieval task even more challenging. Thus, it is useful to retrieve images by producing local feature correspondences between two images based on the assumption that more feature correspondences can be obtained between two images that contain the same object than that of the two irrelevant ones. Feature correspondence is considered as an important property in computer vision. It is widely used as a graph matching task, which tries to optimize a complex objective function based on the feature vectors and spatial constraints [133], [134]. This approach attains better accuracy, but it is computationally expensive, which makes these algorithms inappropriate for image retrieval task.

While the global extraction of features and local from regions presents good results in certain retrieval problems, in the case of "object"-based applications, they are observed to be less efficient. Thus, most modern algorithms begin with the determination of some interest points within an image. The interest points are the regions that carry some properties such as invariance to various image transformations, illumination, etc. After finding the interest points, the regions are defined in the neighborhood of these points and descriptors are formed.

O. Chum has proposed an approach in which the representation of local image structure and a matching scheme is presented which is insensitive to many appearance changes [132]. This method is applied to two-view matching problems of images from different modalities. A technique to learn and recognize object class models from unlabeled and unsegmented cluttered scenes is presented in [133]. In this work, objects are modeled as flexible constellations of parts. A probabilistic representation is used for all aspects of the object: shape, appearance, occlusion and relative scale. An entropy-based feature detector then applied for region selection within an image. R. Sukthankar has proposed a technique to tackle the problem of near-duplicate image retrieval with a part-based representation of images using distinctive local descriptors extracted from points of interest, which are invariant to several transformations.
2.4 Geometric Consistency Preserving Techniques for CBIR

When the retrieval process considers only the model vectors that represent the visual content of images, sometimes fails to retrieve accurate results. This is because the object based approaches completely ignores the geometry of the extracted interest points. Two images can contain similar visual words, but in a totally different spatial layout from each other. Thus, the inclusion of a geometry consistence check would be very useful. RANSAC is the most popular approach used for preservation of geometric consistency. It is a simple, yet powerful, technique that is usually applied to the task of estimating the parameters of a model, using data that may be contaminated by outliers. RANSAC estimates a global relation that fits the data, while simultaneously classifying the data into inliers (points consistent with the relation) and outliers (points not consistent with the relation). Due to its ability to tolerate a large fraction of outliers, the algorithm is a popular choice for several of robust estimation problems.

The working of RANSAC is based on the ‘hypothesis and verify’ framework: a minimal subset of the input data points is randomly chosen and model parameters are estimated from this subset. The model is then evaluated on the entire dataset and its support (the number of data points consistent with the model) is determined. This ‘hypothesis and verify’ loop is repeated until the probability of finding a model with better support than the current model falls below a certain predefined threshold. Usually RANSAC find the correct solution even with high degrees of contamination; however, the number of samples required to do so increase exponentially, and the associated computational cost is very large. There have been a number of efforts aims to increase the efficiency of the fundamental RANSAC algorithm. Some of these strategies [22, 23, 24] aim to optimize the process of model verification, while others [25, 26, 27] seek to modify the sampling process in order to generate more useful hypotheses. While these efforts have shown considerable potential, none of them are directly applicable in situations where real-time performance is important. Relatively fewer efforts have been directed towards the aim of formulating RANSAC in a manner that is suitable for real-time implementations.
Nister describes the preemptive RANSAC framework, where a fixed number of hypotheses are evaluated in a parallel, multi-stage setting [28]. In this case the aim is to find, the best possible solution from a restricted set of hypotheses within a fixed time constraint.

The preemptive RANSAC framework is useful for real-time implementation but there exist few limitations in this scheme. One of the crucial limitations of preemptive RANSAC is its inherent non-adaptiveness to the data. The selection of a fixed number of hypotheses implies that a good prior estimate of the inlier ratio is available; in practice, this is often not the case. For low contamination problems, preemptive RANSAC is often slower than standard RANSAC, since it evaluates many more hypotheses than necessary. On the other hand, when the inlier ratio is too low, preemptive RANSAC is unlikely to find a good solution, since it does not test enough hypotheses. Fundamental RASNAC and its evolution is studied and described in the following subsections.

2.4.1 Standard RANSAC

RANSAC is widely applied to assess the parameters of a model using data which are contaminated by outliers. The data that are consistent with the desired model are called inliers and the rest are outliers.

The input of the RANSAC algorithm is a set $\Omega$ of data points. The objective is to find the optimal parameters $\theta^*$ of a model that maximize the number of inliers. RANSAC performs two steps, generating a hypothesis of the parameters and verifying it with the data.

(i) In the hypothesis generation step, a sample $S_k$ of data points are randomly chosen from $\Omega$ and a hypothesis $\theta^k$ of the parameters is computed from $S_k$.

(ii) In the hypothesis verification step, the support set $I_k$, i.e., the set of inliers consistent with $\theta^k$ is calculated and the quality of $\theta^k$ is specified by the cardinality of $I_k$, $|I_k|$. The two steps are repeated until the probability $\eta$ of finding a better model falls under a threshold $\eta_0$. The termination criteria can be expressed as: after $k$ samples, the probability $\eta$ of all these $k$ samples being contaminated by at least one outlier is

$$\eta = (1 - \varepsilon^m)^k \leq \eta \quad (2.1)$$
where $\varepsilon$ is the fraction of inliers in $\Omega$. Since $\varepsilon$ is unknown earlier, it is updated once a new maximum is reached in the iteration. Algorithm for standard RANSAC is given in Fig. 2.1

**Input:** $\Omega$, $\eta_0 (=0.01)$, $\Delta (=3)$

**Repeat** until $\eta \leq \eta_0$

1. **Hypothesis generation**
   a. Randomly choose a sample $S_k$ of $m=7$ from $\Omega$
   b. Compute the fundamental matrix $F_k$ using the 7-point algorithm [14]. There will be one or three real solutions.
2. **Hypothesis verification**
   a. Calculate the support set $I_k$ of $F_k$. If there are three solutions, retain the solution with the largest support set.
   b. If $I_k$ is larger than any previous set, i.e., a new maximum is reached, store $I_k$, $I^*=I_k$, and update the fraction of inliers $\varepsilon = |I^*|/n$.

**Output:** If the non-randomness requirement [17] is satisfied, compute $F^*$ from $I^*$ using the normalized 8-point algorithm [14], [18] and output the cardinality of the support set of $F^*$. Otherwise, output zero.

**Fig. 2.1 Algorithm for Standard RANSAC**

As discussed in [66], using 7-point algorithm in the iteration has an advantage that using 7 correspondences instead of 8 can decrease the probability of selecting contaminated samples in the hypothesis generation step. Normalized 8 point algorithm is applied in the final step since it performs better than the 7-point algorithm when nine or more correspondences are available, which is shown in [67]. The non-randomness requirement [69] prevents the RANSAC algorithm from choosing an incorrect model which by chance gains large support.
2.4.2 Randomized RANSAC

Randomized RANSAC with $T_{dd}$ test (R-RANSAC) [71] is a RANSAC variation which aims to accelerate the standard RANSAC. The running time $T$ of the standard RANSAC can be expressed as

$$T = k (T_G + n T_v)$$

(2.2)

where $k$ is the number of samples, $T_G$ is the time for generating a hypothesis from the sample, $T_v$ is the average time needed to compute the hypothesis, and $n$ is the number of data points in the whole set $\Omega$

There are usually two ways to make RANSAC faster:

(i) Reducing the necessary number of samples drawn.

(ii) Reducing the number of data points that are estimated in the hypothesis verification step.

The scheme of R-RANSAC belongs to the latter one. At the beginning of the hypothesis verification step, R-RANSAC performs the $T_{dd}$ test, and full evaluation of the whole data is carried out only for those promising hypotheses that pass the $T_{dd}$ test. As defined in [71], $T_{dd}$ test is passed if all $d$ randomly selected points are consistent with the hypothesized model. In this manner, those inconsistent hypotheses generated from contaminated samples are probably rejected by the $T_{dd}$ test. The average number of data evaluated for each hypothesis is smaller than $n$.

R-RANSAC is very suitable for the image retrieval task, because most of the generated hypotheses are very large, especially between two irrelevant images. By rejecting the incorrect models in the next stage, R-RANSAC can achieve a speedup. In our implementation, $d$ is set to be 1 as suggested in [71]. Algorithm for R-RANSAC with $T_{1,1}$ test for estimating the fundamental matrix is given in Fig.2.2
**Input:** \( \Omega, \eta_0 (=0.01), \Delta(=3) \)

**Repeat** until \( \eta \leq \eta_0 \)

1. **Hypothesis generation**
   a. Randomly select a sample \( S_k \) of \( m=7 \) from \( \Omega \)
   b. Compute the fundamental matrix \( F_k \) using the 7-point algorithm [14], [16].
      There will be one or three real solutions.

1. **Hypothesis verification**
   a) Perform the \( T_{1,1} \) test. If no solution passes the test, jump to the hypothesis generation step. Otherwise, compute the support set \( I_k \) of \( F_k \). If two or three solutions are available, keep the one with the largest support set.
   b) If \( I_k \) is larger than any previous set, i.e. A new maxima is reached store \( I_k, I^* = I_k \), and update the fraction of inliers \( \epsilon = |I^*|/n \).

**Output:** If the non-randomness requirement is satisfied, compute \( F^* \) from \( I^* \) using the normalized 8-point algorithm and output the cardinality of the support set of \( F^* \). Otherwise, output zero.

**Fig.2.2 Algorithm for R-RANSAC with \( T_{1,1} \) Test**

**2.4.3 Locally Optimized RANSAC**

The locally optimized RANSAC (LO-RANSAC) [67] improves the standard RANSAC by adding a local optimization step (LO step), which is carried out only if a new maximum in the number of inliers is reached. The brief structure of LO-RANSAC for computing the fundamental matrix is presented in Fig.2.3
Input: $\Omega, \eta_0 (= 0.01), \Delta (= 3)$

Repeat until $\eta \leq \eta_0$

1. Hypothesis generation
   a. Randomly select a sample $S_k$ of $m=7$ from $\Omega$
   b. Calculate the fundamental matrix $F_k$ using the 7-point algorithm [14], [16].
      There will be one or three real solutions.

2. Hypothesis verification
   a. Calculate the support set $I_k$ of $F_k$, i.e., the correspondences with error smaller than $\Delta$. If there are three solutions, keep the solution with the largest support set.
   b. If $I_k$ is larger than any previous set i.e., $|I_k| > |I_j|$ for all $j < k$ then run LO step. If the model $F_{LO}$ obtained in the LO step is better than $F_k$, then $I^* = I_{LO}$, otherwise $I^* = I_k$, Update the fraction of inliers. $\epsilon = |I^*|/n$

Output: If the non-randomness requirement is satisfied, calculate $F^*$ from $I^*$ using the normalized 8-point algorithm and output the cardinality of the support set of $F^*$. Otherwise, output zero.

Fig.2.3 Algorithm for LO-RANSAC for Estimating the Fundamental Matrix

The termination criterion of the algorithm is decided on the basis of the optimal support set $I^*$ while the execution of the local optimization step depends on $I_k$. The LO step applied is the inner RANSAC with iteration [67].

Due to LO step, more inliers are detected, and consequently the number of samples drawn decreases. Comparing with the standard RANSAC, LO-RANSAC speeds up the hypothesize-and-verify iteration by generating fewer samples.
2.4.4 Progressive Sample Consensus

In fundamental RANSAC, samples are chosen randomly from the whole data. The progressive sample consensus (PRO-SAC) [69] uses a new sampling strategy. Under the assumption that tentative correspondences with higher similarity are more likely to be inliers, PROSAC first sorts the correspondences in the descending order of their similarity and then chooses a sample among the top-ranked data. The size of the subset from which samples are generated is increased gradually.

There is a termination length optimization step in PROSAC. The optimal termination length \( n^* \) is defined as

\[
n^* = \arg\max_{m \leq n' < n} \left( \frac{|I_{n'}|}{n'} \right)
\]  \hspace{1cm} (2.3)

Where \( |I_{n'}| \) is the number of inliers in the top \( n' \) data points. The fraction of inliers \( \varepsilon \) in (Eq. 2.3) is then updated if \( |I_{n^*}| / n^* \) is a new maximum. PROSAC algorithm for estimating the fundamental matrix is given in Fig.2.4.

**Input:** \( \Omega, \eta_0 (= 0.01), \Delta (= 3) \)

Sort the tentative correspondences in ascending order of the Euclidean distance of their SIFT descriptors.

Repeat until \( \eta \leq \eta_0 \)

1. Hypothesis generation
   a) Choose a sample \( S_k \) of size \( m (=7) \) from \( \Omega \) using the guided sampling strategy.
   b) Compute the fundamental matrix \( F_k \) using the 7-point algorithm. There will be one or three real solutions

2. Hypothesis verification
   a) Compute the support set \( I_k \) of \( F_k \). If there are three solutions, retain the solution with the largest support set.
b) If $I_k$ is larger than any previous set, i.e., a new maximum is reached, then $I^* = I_k$ and run the termination length optimization step to get the optimal length $n^*$. If $\left| \frac{I_{n^*}}{n^*} \right|$ is a new maximum, then update the fraction of inliers, i.e., $\varepsilon = \left| \frac{I_{n^*}}{n^*} \right|$.

Output: If the non-randomness requirement is satisfied, compute $F^*$ from $I^*$ using the normalized 8-point algorithm and output the cardinality of the support set of $F^*$. Otherwise, output zero.

Fig. 2.4 Algorithm for PROSAC

The guided sampling strategy enables the most promising samples being inspected earlier, which means a relatively large $\varepsilon$ can be obtained in the early stages and the algorithm terminates early. The termination length optimization is further accelerate the algorithm by finding an optimal fraction of inliers $\left| \frac{I_{n^*}}{n^*} \right|$. The property of early termination of PROSAC only holds for relevant images, since the fundamental assumption of PROSAC is valid only for two views of the same scene.

2.5 Relevance Feedback

CBIR system sometimes fails to precisely retrieve desired images. Then it is useful to refine the results using the iterative feedback given by the user. Relevance feedback is a process in which user feedback is important. An image retrieval system presents a ranked set of images which are relevant to the user’s initial query and then iteratively solicits the user for feedback on the relevance of images and uses the feedback to compose and improve the retrieval result. Relevance feedback can be considered as a learning problem, a user provides feedback examples from the retrieved results of a query and the system learns from such examples to refine retrieval results.
According to Mitchell’s [113] definition, machine learning deals with the question of how to design a computer program that automatically improve with experience. In this view, any task that could be improved with respect to certain performance measure based on some experience can be considered as the machine-learning task. In CBIR, relevance feedback is a task to improve the retrieval performance and the experience here is feedback examples provided by the users. Hence, classical machine-learning methods, such as decision tree learning [119], artificial neural networks [118], Bayesian learning [126,123], and kernel-based learning [122] can be used for relevance feedbacks in CBIR.

Users are usually reluctant to provide a large number of feedback examples; here the number of training samples is very small, typically less than ten in each round of the feedback session. On the contrary, feature dimensions in CBIR systems are usually high. Hence, the crucial issue in performance of relevance feedback in CBIR systems is how to learn from small training samples in a very high dimension feature space. This fact makes many learning methods, such as decision tree learning and artificial neural networks, not suitable for CBIR [120]. The key issues in addressing relevance feedback in CBIR as a small sample learning problem include: how to learn fast from small sets of feedback samples to improve retrieval accuracy effectively; how to accumulate knowledge learned from feedback; and How to integrate low-level visual and high-level semantic features in the query. However, most of the published works have been focused on the first issue.

Compared with other learning methods, Bayesian learning is observed to be more efficient [26]. Vasconcelos and Lippman treated feature distribution as a Gaussian mixture and used Bayesian inference for learning during feedback iterations in a query session [123]. Richer information captured by the mixture model also makes image regional matching possible.

The approach proposed in [126] used Monte Carlo sampling to search the set of sample that will minimize the expected number of future iterations. In estimating the expected number of future iterations, entropy is used as an estimate of the number of future iterations under the ambiguity specified by the current probability distribution of the target image over the all test images. Tong and Chang [122] proposed an SVM active learning algorithm to select a sample to maximally reduce the size of vector space in
which the class boundary lies. It is observed that, selecting the points near the SVM boundary can achieve this goal, and it is more efficient than other schemes, which require exhaustive trials on all the test items. Therefore, in their work, the points near the SVM boundary are used to approximate the most-informative points; and the most-positive images are chosen as the ones farthest from the boundary on the positive side in the feature space.

Some researchers consider a relevance feedback process in CBIR as a pattern recognition or classification problem. Under such a consideration, the positive and negative examples provided by user can be treated as training examples and a classifier could be trained. Then, such classifier can separate all data set into relevant and irrelevant groups. It seemed that many existing pattern recognition tools could be adopted for this task and many kinds of classifiers have been experimented, such as a linear classifier [125], nearest-neighbor classifier [124], Bayesian classifier [120], support vector machines (SVM) [122], and so on. In this category, the most popular algorithm is represented by [122] where the SVM classifier is trained to divide the positive and negative examples. Then such SVM classifier will classify all images in database into two groups: relevant and irrelevant groups to a given query.

2.6 Content Based Image Retrieval in P2P Network

Since mid-1990s, many CBIR systems have been proposed and developed, including QBIC [77], WebSEEK [91], WBIIS [92], SIMPLIcity [96], MARS, [81], NeTra [80] Photobook [85] and other systems for domain-specific applications. The images are represented in the form of a feature vector with their similarity which is based on the distance between the feature vectors. One of the drawbacks of these systems is that the feature extraction, indexing, and querying are carried out in a centralized way, which can be computationally intensive, and it is difficult to scale up.

One of the promising future trends in CBIR includes distributed computing on data collection, data processing, and information retrieval. By extending the centralized system model, we not only can increase the size of image collections easily, but we can
also overcome the scalability bottleneck problem by distributing the task of image retrieval. P2P network is a recently evolved paradigm for distributed computing.

The P2P file sharing applications can accomplish tasks that are difficult for the conventional centralized computing models to achieve. For example, by distributing data storage over networked computers, one can have virtual data storage that is much more than what can be stored in a local computer. In addition, one may also foresee data security by distributing pieces of an encrypted file over many computers. By doing so, one imposes a difficult barrier for an intruder to overcome because one needs to break into several computers before getting the file [97]. Likewise, one may also distribute the computation among different computers to achieve a high performance throughput [94]. Peer-to-Peer (P2P) network offers a completely decentralized and distributed paradigm on top of the physical network, which avoids the coordinator bottleneck problem.

Emerging P2P networks or the implementations such as Gnutella [Gnutella], Napster [101], Freenet [99], LimeWire [100], and eDonkey [98] offer the following advantages:

- **Distributed Resource**: The storage, information and computational cost can be distributed among the peers, allowing many individual computers to achieve a higher throughput [36].

- **Increased Reliability**: The P2P network improves reliability by eliminating dependence on centralized coordinators. In other words, the P2P network can still be operational even after a certain portion of peers is down [96].

- **Comprehensiveness of Information**: The P2P network has the potential to reach every computer on the Internet.

P2P networks, which are formed by equally privileged nodes connecting to each other in a self-organizing way, have been one of the most important architectures for data sharing. Popular P2P file-sharing networks such as eDonkey counts millions of users [98] and tens of millions of files. The ever-growing amount of multimedia data and computational power of P2P networks motivates the need and potential for large scale multimedia retrieval applications such as content-based image sharing. While P2P networks are well known for their efficiency, scalability and robustness on file sharing,
providing extended search functionality such as content-based image retrieval (CBIR) faces the following challenges:

- In contrast to centralized environments, data in P2P network is distributed among different nodes, thus a CBIR algorithm needs to index and search for images in a distributed manner.
- Unlike distributed servers/clouds, nodes in P2P networks have limited network bandwidth and computational power, thus the algorithm should keep the network cost low and the workload among nodes balanced.
- As P2P networks are under constant churn, where nodes join and leave the network frequently, the index needs to be updated dynamically to adapt to such changes.

To support content indexing and to avoid message flooding, structured overlay networks such as Distributed Hash Tables (DHTs) [86, 93] are often implemented on the top of a physical network. By organizing the nodes in a structured way, messages can be efficiently routed between any pair of nodes, and the index integrity can be maintained during network churn.

For the CBIR, most of the existing systems adopt a global feature approach: an image is represented as a high dimensional feature vector (e.g., color histogram), and the similarity between files is measured using the distance between two feature vectors [3, 4, 5]. Usually, the feature vectors are indexed by a distributed high-dimensional index or Locality Sensitive Hashing (LSH) over the DHT overlay. However, due to the limitation known as “curse of dimensionality”, the majority of these solutions have high network costs or serious workload balance issue among nodes when the dimensionality of feature vectors is high.

In a P2P-CBIR, when a peer initiates a search for an image, it broadcasts a query request to its connected peers. Peers then forward the request to their connected peers and this process continues. Unlike the client server architecture of the web, the P2P network allows individual computers that join and leave the network frequently to share information directly with each other without the help of dedicated servers. Each peer acts as a server and as a client simultaneously on these networks, a peer can become a
member of the network by connecting with one or more peers in the current network. Messages are sent over multiple hops from one peer to another while each peer looks up its locally shared collection and responds to queries. Basically, this model of query broadcasting is wasteful because peers are forced to handle irrelevant query messages. This type of search is called a Brute Force Search (BFS).

There are several solutions proposed to solve the query broadcasting problem. Chord [93], CAN [86], Pastry [88] and Tapestry [88] tackle it by distributing the index storage into different peers, thus sharing the workload of a centralized index server. Distributed infrastructure of both CAN and Chord uses Distributed Hash Table (DHT) technique to map a filename to a key; each peer is responsible for storing a certain range of (key, value) pairs. When a peer looks for a file, it hashes the filename to a key and asks the peers responsible for this key for the actual storage location of that file. Chord approach models the key as a m-bit identifier and arranges the peers into a logical ring topology to determine which peer is responsible for storing which pair (key, value). CAN approach model the key as a point on a D-dimensional Cartesian coordinate space, while each peer is responsible for the pairs (key, value) inside its specific region. Such systems take a balance between the centralized index and totally decentralized index approaches. They speed up and reduce message passing for the process of key lookup (data location).

Some extensions of DHTs to perform content-based retrieval and textual similarity matches are proposed in Tang et al.[94] and Harrenetal.[91]. Although DHT’s are scalable, their performance under the dynamic conditions of prevalent P2P systems is still unknown due to the penalty incurred in joining and leaving [90]. As DHTs mandate a specific network structure and incur a certain penalty on joining and leaving the network, some researchers propose methods that operate under the prevalent dynamic P2P environment, for example, Gnutella. Crespo proposed a routing indices approach for retrieving text documents in P2P systems [76]. Under this scheme, each peer maintains a routing index, which is used for forwarding queries to peers that are supposed to contain more documents of the same category as the queries. This method requires all peers to agree upon a set of document categories.
2.7 Research Opportunities from Literature Review

The detailed study of literature in the area of feature extraction, object recognition, geometric consistency preservation, relevance feedback and visual query routing in P2P network revealed following basic aspects:

1. Image data exist with extremely diverse visual content, and it is rapidly growing in size. Conventional text based image retrieval techniques have many limitations. This motivates the need to work in the area of content based image retrieval.

2. Retrieval performance of CBIR system is highly influenced by the choice of features used for representation of images.

3. It is important to make use of an object based recognition techniques to meet the requirement of real world images which are affected by the factors such as illumination or 3D viewpoint change.

4. Inclusion of geometric consistency preservation can improve the accuracy of retrieval.

5. The crucial issue in performing relevance feedback in CBIR systems is how to learn from small training samples in a very high dimension feature space.

6. P2P-CBIR can overcome the scalability problem of centralized CBIR system by using a decentralized retrieval algorithm.

With these aspects the current research is focused on following areas:

1. Presenting the new feature extraction technique which can improve the retrieval performance

2. Proposing an efficient method for object based image retrieval.

3. To investigate the distinctive properties of several RANSAC variants and to utilize it to preserve the geometric consistency and to speed up the image retrieval procedure.

4. To present a novel probabilistic approach for relevance feedback which uses uncertainty based sampling to minimize the users responsibility of query formation and to improve the retrieval accuracy

5. To design a peer clustering and an intelligent query routing strategy to search images efficiently over the P2P network.
2.8 Thesis Contribution

The efforts in meeting the objectives of the current research discussed in chapter 1, resulted in different contributions in the major steps of this research, which are summarized below:

1. Feature Extraction

Images are represented by their visual content, such as color, texture and shape. Among them, color is the most common feature extracted for various image search applications. These features represent image color content and are often presented as color histograms. The major drawback of these traditional color histogram methods is that they do not take the image color distribution into consideration. In this research, we have presented an efficient algorithm for adaptive color feature extraction according to the image color distribution that reduces the distortion incurred in the feature extraction process. The proposed method for feature extraction is discussed in chapter 3.

2. Object Based Image Retrieval and Preservation of Geometric Consistency

We have proposed an object based image retrieval system using local feature correspondences. This system is based on the assumption that more correspondences can be determined between two images that contain the same object. Initially, tentative key point correspondences are generated by comparing the SURF descriptors, then the two-view geometric relations are computed using RANSAC algorithm. The similarity of two images is computed by the number of tentative matches and the true correspondences. We have further improved the conventional RANSAC algorithm by limiting the number of samples drawn in the ‘hypothesis and verify’ phase of it. This speeds up the retrieval process significantly. It is observed that the searching accuracy is negligibly affected due to the important property of our proposed algorithm to early terminate on relevant images. The proposed technique is presented in chapter 4.
3. **Design of P2P- CBIR System**

We have proposed a novel architecture for CBIR that works seamlessly in a P2P environment. When users need to search for an image, all peers in the network will look up their own collection of images and respond to the requesters. Such architecture completely utilizes the storage and computation capability of computers on the network. However, absence of a centralized index requires a query to be broadcasted throughout the network in order to achieve good results. To solve this query broadcasting problem, a peer clustering and efficient query routing strategy is proposed to search images efficiently over the P2P network. Proposed Design of an efficient CBIR system which works in P2P network is also presented in chapter 4.

4. **Relevance Feedback**

In CBIR, the search is initiated using a query as an example. The top rank similar images are then presented to the user. Then, the interactive process allows the user to give information (feedback) to refine his request as much as necessary in a relevance feedback loop. Most of the time, user information consists of binary labels indicating whether or not the image belongs to the desired concept. Positive labels indicate relevant images for the current concept, and the negative labels indicate irrelevant images. The query is then modified according to this labeling. We have considered this query modification process as a two class classification problem: the relevant class and the irrelevant class. We have used the uncertainty based sampling approach for relevance feedback. Uncertainty-based sampling strategy aims at selecting images that the user is the most uncertain about. We have presented an approach in which system computes a probabilistic output for each top ranked image and selects the samples with the probabilities closest to 0.5. The proposed approach for relevance feedback is discussed in chapter 5.

2.10 **Conclusion**

A Literature survey of the work is presented in this chapter. The focus is on the design of an efficient content based image system which works in P2P network for decentralized sharing of distributed computing resources.
The study of literature in the area of feature extraction, Object based retrieval, relevance feedback and peer clustering and query routing in P2P network is described in detail. The implicit research opportunities in this area are enlisted at the end of the literature survey.