A survey on feature extraction techniques for color images

Dr. Sanjay Kumar , Er. Ankur Chauhan

ABSTRACT: The applications of machine learning and data analysis are rapidly increases. In such kind of applications multimedia data is compared for storage. In addition of that during retrieval and search a significant amount of resources are consumed by these methods. Nowadays various applications are available that claim to extract the accurate information from such colored image databases with less computational cost. In addition of that each image of data includes different information, thus for preserving the query relevant information from search content based methods are appropriate for image data search. During information extraction based on the content of images various kinds of feature extraction techniques are available. These techniques lead to provide the information with fewer amounts of storage and less computational cost. This presented work focus on the information reflected by various feature extraction techniques and where they can be easily adaptable. The given technique is further utilized in different application such as object detection using multimedia data base.

Keywords: content based filtering, face recognition, feature extraction, survey.

1. INTRODUCTION

In machine learning process data is recognized using their meaningful patterns. These patterns are extracted using the patterns pattern analysis of data. Thus machine learning is a task of data pattern analysis and pattern extraction. To find the recognizable patterns among data different mathematical models are available. These methods reduce the amount of data with actual relationship or difference between two data instances. Therefore that is computationally complex domains where uncertainty and randomness nature of the data is used to extract data pattern.

In this paper an evaluation of different techniques of image based pattern analysis and feature extraction techniques are provided, by which the optimal properties between data can be distinguished. In addition of that the image data feature extraction methodologies are investigated by which using less computational most appropriate and informative attributes are recovered from image.

For appropriate understanding of the relation between the data processing and image processing first we illustrate an example, suppose we have a set of random documents, for classifying or appropriate arrangement of these documents according to their domain, needed to find some knowledge about the document contents, therefore first needed to read a document and then estimate the domains and topic reside in the given document. In the same way for finding the proper patterns over image data, pre-processing, data model construction and implementation is required.

Image is a different kind of data source which comprises a huge amount of information, such as color information, objects, edges, pixel definition, dimensions and others. Therefore the treatment of image data is a sensitive concern to conserve the complete information. This paper address the different key features and properties of image data by which the information from the image is removed and consumed for different applications of face recognition, image retrieval and others.

2. BACKGROUND STUDY

Feature is an image pattern which differs from its immediate neighborhood. It is usually associated with a change of an image property or several properties, though it is not localized exactly on this change. The key image properties are intensity, color, and texture. Good features should have the following properties:

Repeatability: If two images of the same objector scene are taken under different conditions, a high percentage of similar features visible on both the images.

Distinctiveness/in formativeness: The intensity patterns underlying the detected features should show a lot of variations, such that features can be distinguished and matched.

Locality: The features should be local, to reduce the probability of occlusion and to allow simple model approximations of the geometric and photometric deformations between two images taken under different viewing conditions.

Quantity: The number of detected features should be large enough, so that a reasonable number of features are detected even on small objects. However, the optimal number of features depends on the application. Ideally, the number of detected features should be controllable over a large range by a easy and intuitive threshold. The density of features should reflect the information content of the image to provide a compact image representation.

Accuracy: The detected features should be localized accurately, with respect to scale and shape in both image locations.

Efficiency: Preferably, the detection of features in a new image should allow for time-critical applications. Repeatability, the most important property of all, can be achieved in two different ways: either by invariance or by toughness.

Invariance: When huge deformations are to be expected, the preferred approach is to model these mathematically if possible, and then develop methods for feature detection that are unaffected by these mathematical transformations.

Robustness: In case of relatively small deformations, it often suffices to make feature detection methods less sensitive to such deformations, i.e., the accuracy of the detection may decrease, but not radically. Typical deformations that are undertaken using robustness are image noise, discretization effects, compression artefacts, blur, etc. Also geometric and photometric deviations from the mathematical model used to attain invariance are often overcome by counting more robustness.

3. CONTENT BASED IMAGE RETRIEVAL

An image retrieval system can be defined as searching, browsing, and retrieving images from massive digital image databases. Although Conventional and common techniques of retrieving images usage annotation of words for ease in searching. However image search can be illustrated by dedicated technique of search which is generally used to find images. For searching images user offers the query image and the system returns the image parallel to query image [2].

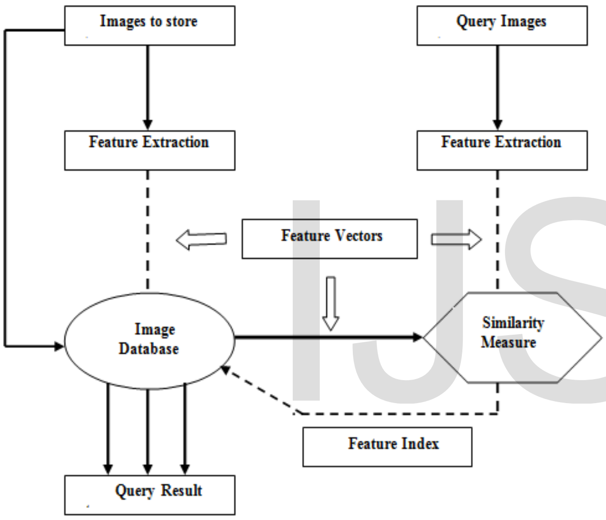


Figure 1 General Image Retrieval System

Thus there is a need for development of efficient and effective techniques to manage large image databases. So many image retrieval systems have been developed in recent years. There are three main kinds of image retrieval techniques available: text-based method, content-based method and hybrid method.

Text based Image retrieval system: Text Based Image Retrieval is currently used in almost all web image retrieval systems. This approach uses the text associated with image to determine what the image contains. This text can be text nearby the image, the image's filename, a hyperlink leading to the image, an explanation to the image, or any other piece of text that can be associated with the image.

Issues:

1. There are many irrelevant words nearby textual descriptions, which results in low image search precision rate.
2. The surrounding text does not fully describe the semantic content of images, which results low image search recall rate.
3. The third problem is same word can be used to refer more than one object. Due to the query polysemy, the result searcher will not succeed to find images tagged other languages.

Content Based Image Retrieval: Content Based Image Retrieval is a set of techniques for retrieving semantically-relevant images from database based on automatically-derived image features [3]. This plans at avoiding the use of textual explanations and instead retrieves images based on their visual resemblance.

The main goal of CBIR is maintaining efficiency during image indexing and retrieval [4]. The computer must be capable to retrieve images from a database without any human assumption on particular domain. One of the major tasks for CBIR systems is similarity comparison, take out characteristic of every image based on its pixel values and defining rules for comparing images. These characteristics become the image representation for calculating similarity with other images in the database. Images are compared by computing the difference of its feature components to other image descriptors [5]. These descriptors are attained globally by extorting information on the means of colour histograms for colour features; global texture information on coarseness, contrast, and direction; and shape features about the curvature, moments invariants, circularity, and eccentricity. Similarly, the Photo book system features to symbolize image semantics [6]. These global advances are not sufficient to support the queries looking for images where particular objects in an image having specific colours and/or texture are present, and shift/scale invariant queries, where the position and/or the dimension of the query objects may not appropriate [7].

Most of the presented CBIR systems think each image as a whole; however, a single image can contain various regions/objects with entirely different semantic meanings. Therefore, rather than viewing each image as a whole, it is more reasonable to view it as a set of regions. The features employed by the majority of Image Retrieval systems include colour, texture, shape and spatial layout. Such features are actually not effective for CBIR, if they are removed from a whole image, because they suffer from the different backgrounds, overlaps, occlusion and cluttering in different images and do not have adequate ability to capture important properties of objects, as a result most popular approaches in recent years are to change the focus from the global content clarification of images into the local content explanation by regions or even the objects in images. The content-based approach can be summarized as follows:

1. Computer vision and image processing techniques are used to extract content features from the image.
2. Images are represented as collections of their prominent features. For a given image, an appropriate representation of the feature and a notion of similarity are determined.
3. Image retrieval is executed based on calculating similarity or Dissimilarity in the feature space, and results are ordered based on the similarity measure.

Region based image retrieval system: rather than deploying global features over the entire content, RBIR systems divide an image into number of homogenous regions

ISSN 2229-5518

and take out local features for each region then features of various regions are used to symbolize and index images in RBIR. For RBIR, The user supplies a query object by choosing a region of a query image and then the consequent similarity measure is calculated between features of region in the query and a set of features of divided regions in the features database and the system returns a ordered list of images that contain the same object.

4. LOW LEVEL FEATURE EXTRACTION TECHNIQUES

This section comprises the different feature vector calculation methods that are consumed to design algorithm for image retrieval system.

A. Grid Color Moment

Color feature is one of the most commonly used features in low level feature. Compared with shape and texture feature, color feature shows superior stability and is more insensible to the rotation and zoom of image. Color also adds more information, which is used as powerful tool in content-based image retrieval. In color indexing, the aim is to retrieve all the images whose color and texture compositions are analogous to query image. In color image retrieval there are numerous methods, but here we will discuss some noticeable methods. For color feature vector we will use "Grid-based Color Moment" analysis. How to calculate this feature vector for a given image: [8]

- Convert the image from RGB for HSV color space
- Equivalently divide the image into 3x3 blocks
- For each of these nine blocks
- Calculate its mean color (H/S/V)

$$x' = \frac{1}{N}\sum_{i=1}^{N} x_i$$

Where N is the number of pixels in a block, $x_i$is the pixel intensity in H/S/V channels.

- Compute its variance (H/S/V)

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - x')^2$$

- Compute its skewness (H/S/V)

$$\gamma = \frac{\frac{1}{n}\sum_{i=1}^{N}(x_i - x')^3}{\left(\frac{1}{n}\sum_{i=1}^{N}(x_i - x')^2\right)^{3/2}}$$

- Each block will have 3+3+3=9 features, and thus the entire image will have 9x9=81 features. Now first need to normalize the 81 features.

To do the normalization, for each of the 81 features:

- Compute the mean and standard deviation from the training dataset

$$\mu = \frac{1}{M}\sum_{i=1}^{M} f_i$$

$$\sigma = \sqrt{\frac{1}{M}\sum_{i=0}^{M}(f_i - \mu)^2}$$

- M is the number of images in the training dataset, and f_i is the feature of the i^{th} training sample.
- Perform the "whitening" transform for all the data (including training data and testing data), and get the normalized feature value:

$$f'_i = \frac{f_i - \mu}{\sigma}$$

B. Canny Edge Detection

The purpose of edge recognition in general is to significantly decrease the amount of data in an image, while conserving the structural properties to be used for further image processing. [9, 10]

The algorithm runs in 5 separate steps:

1. **Smoothing:** Blurring of the image to eliminate noise.
2. **Finding gradients:** The edges must be marked where the gradients of the image has huge magnitudes.
3. **Non-maximum suppression:** Only local maxima should be marked as edges.
4. **Double thresholding:** Potential edges are determined by thresholding.
5. **Edge tracking by hysteresis:** Final edges are resoluted by suppressing all edges that are not linked to a very strong edge.

Smoothing

It is predictable that all images taken from a camera will include some amount of noise. To stop that noise is mistaken for edges, noise must be condensed. Therefore the image is first smoothed by applying a Gaussian filter where the kernel of Gaussian filter with a standard deviation is σ = 1.4. The effect of smoothing the test image with this filter is shown in Figure 2.

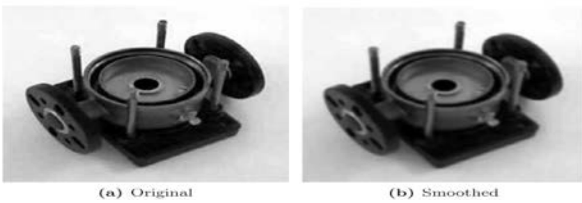


Figure 2 Smoothing effect on image

Finding gradients

The gradient magnitudes also known as edge strengths are determined using Euclidean distance.

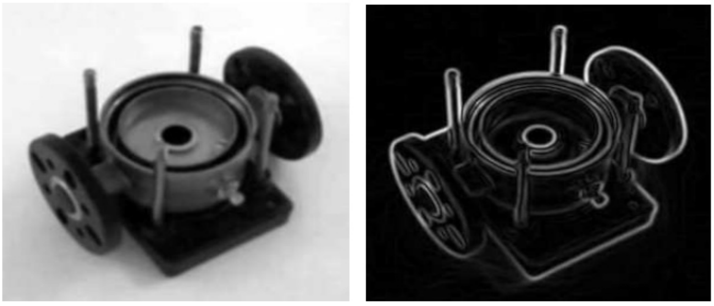
$$|G| = \sqrt{G_x^2 + G_y^2}$$

Or sometimes simplified by applying Manhattan distance measure to reduce the computational complexity.

$$|G| = |G_x| + |G_y|$$

G_x and G_y are the gradients in the x- and y-directions respectively.

The Euclidean distance compute has been applied to the test image. The calculated edge strengths are evaluated to the smoothed image in Figure (3).



(a) Smoothed (b) Gradient magnitudes

Figure 3 Gradient magnitudes of image

The image gradient magnitudes often indicate edges clearly. However, the edges are classically broad and thus smoothing do not specify exactly where the edges are. To make it possible the direction of the edges must be determined as.

$$\theta = arcTan\left(\frac{|G_y|}{|G_x|}\right)$$

Non-maximum suppression

The purpose of this step is to convert the "blurred" edges in image of the gradient magnitudes to "sharp" edges. Mostly this is done by conserving all local maxima in the gradient image, and erasing everything else. The algorithm is for each pixel in the gradient image:

1. Round the gradient direction θ to nearest 45◦, corresponding to the use of an 8-connected neighborhood.
2. Evaluate the edge power of the existing pixel with the edge strength of the pixel in the positive and negative gradient direction.
3. If the edge strength of the current pixel is largest; conserve the value of the edge strength. If not, remove the value.

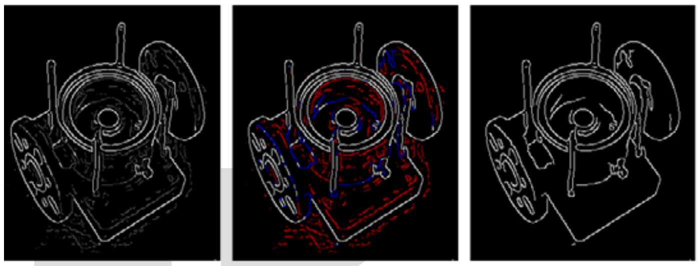
Double Thresholding

The edge-pixels remaining after non-maximum suppression step are marked with their strength pixel-by-pixel. Many of these will possibly be true edges in the image, but some maybe caused by noise or color differences for instance due to rough surfaces. The easiest way to discern between them would be to use a threshold, so that only edges stronger that a definite value would be preserved. The Canny edge detection algorithm utilizes double thresholding. Edge pixels stronger than the high threshold are marked as strong, whereas, edge pixels weaker than the low threshold are removed.

Edge tracking by hysteresis

Strong edges are taken as "certain edges", and can immediately be incorporated in the final edge image. Weak edges are integrated if and only if they are linked to strong edges. The judgment is of course that noise and other small differences are unlikely to result in a strong edge(with appropriate adjustment of the threshold levels). Thus strong edges will only be due to true edges in the original image. The weak edges can either be due to true edges or noise/color variations.

Edge tracking can be implemented by BLOB-analysis (Binary Large Object). The edge pixels are divided into connected BLOB's using 8-connected neighbor-hood. BLOB comprising at least one strong edge pixel is then conserved, while other BLOB's are suppressed. The effect of edge tracking on the test image is shown in Figure 4.



(a) Double thresholding (b) Edge tracking by hysteresis (c) Final output

Figure 4 Blob Analysis

Local Binary Pattern

Given a pixel in the image, an LBP [11] code is computed by comparing it with its neighbors:

$$LBP_{P,R} = \sum_{P=0}^{P=1} s(g_p - g_e)2^p$$

$$s(x) = \begin{cases} 0 & x \geq 0 \\ 1 & x < 0 \end{cases}$$

g_e is the gray value of the central pixel, g_p is the value of its neighbors, P is the total number of involved neighbors and R is the radius of the neighborhood. Suppose the coordinate of g_e is (0, 0), then the coordinates of g_p are

$$\left(Rcos\left(\frac{2\pi p}{P}\right), Psin\left(\frac{2\pi P}{P}\right)\right)$$

The gray values of neighbors that are not in the image grids can be estimated by interpolation. Suppose the image is of size I*J after the LBP pattern of each pixel is identified, a histogram is built to represent the texture image:

$$H(k) = \sum_{i=1}^{I}\sum_{j=1}^{j} f(LBP_{P,R}(i,j),k), k \in [0,k]$$

$$f(x,y) = \begin{cases} 1 & x = y \\ 0 & otherwise \end{cases}$$

K is the maximal LBP pattern value. The U value of an LBP pattern is defined as the number of spatial transitions (bitwise 0/1 changes) in that pattern

$$U(LBP_{P,R}) = |s(g_{p-1} - g_e) - s(g_0 - g_e)| + \sum_{p=1}^{P-1} |s(g_p - g_e) - s(g_{p-1} - g_e)|$$

The uniform LBP patterns refer to the patterns which have limited transition or discontinuities ($U \leq 2$) in the circular binary presentation. In practice, the mapping from $LBP_{P,R}$ to $LBP_{P,R}^{u2}$ which has P*(P-1)+3 distinct output values, is implemented with a lookup table of 2^P elements. To achieve rotation invariance, a locally rotation invariant pattern could be defined as:

$$LBP_{P,r}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_e) & if U(LBP_{P,R}) \leq 2 \\ P+1 & otherwise \end{cases}$$

The mapping from $LBP_{P,R}$ to $LBP_{P,R}^{u2}$ which has P+2 distinct output values.

GABOR filter

In the one-dimensional case, the Gabor function comprises of a typical exponential localized around x = 0 by the envelope with a Gaussian window shape [10].

$$g_{\alpha,\varepsilon}(x) = \sqrt{\alpha/\pi}\, e^{-\alpha x^2} e^{-i\varepsilon x}$$

$\alpha \in R^+$ and $\varepsilon, x \in R$, where $\alpha = (2\sigma^2)^{-1}$, σ^2 is variance and ε is frequency. Dilation of the complex exponential function and shift of the Gaussian window, when the dilation is fixed form kernel of a Gabor transform. The Gabor transform employs such kernel for time-frequency signal analysis. The declared Gaussian window is the best time frequency localization window in a sense of the Heisenberg uncertainty principle [12].

In a two-dimensional case, the supreme square of the correlation between an image and a two-dimensional Gabor function offers the spectral energy density concerted around a given position and frequency in a certain direction. Moreover, the two-dimensional convolution with a circular Gabor function is separable to series of one-dimensional ones

$$g_{\alpha,\varepsilon}(x) = g_{\alpha,\varepsilon 0}(x_0)\, g_{\alpha,\varepsilon 1}(x_1)$$

For $\varepsilon = (\varepsilon_0, \varepsilon_1)$ and $x = (x_0, x_1)$ Here, the actual frequency of the two-dimensional function is determined by $\varepsilon = (\varepsilon_0^2 + \varepsilon_1^2)^{1/2}$ Furthermore $\vartheta = \arctan\left(\frac{\varepsilon_1}{\varepsilon_0}\right)$ is an angle between x-axis and a linear perpendicular to the ridges of a wave.

Gabor Wavelet

Elements of a family of mutually similar Gabor functions are called wavelets when they are created by dilation and shift from one elementary Gabor function, i.e.

$$g_{\alpha,\varepsilon,a,b}(x) = |a|^{-1/2} g_{\alpha,\varepsilon}\left(\frac{x-b}{a}\right)$$

for $a \epsilon R^+$(scale) and $b \epsilon R$(shift). By convention, the mother wavelet has the energy localized around x = 0 as well as all of the wavelets are normalized $\|g\| = 1$. Although the Gabor wavelets do not form orthonormal bases, the discrete set of them form a frame.

The used notation is in accordance with [13]. The first order partial derivative of image I with respect to variable x is denoted by I_x. Analogously I_{XX} denotes the second order partial derivative with respect to x and I_{xy} is the second order mixed derivative. Furthermore $I_x(x, \sigma_D)$ denotes a partial derivative attained at the location of an point x and computed by using a Gabor wavelet with scale $a \propto \sigma_D$.

Edge Detection

For the edge detection, the complication in two perpendicular directions is performed with variously dilated wavelets. It is essential to use a wavelet which serves as the first order partial differential operator. Consequently, local maxima of module are found.

$$M(x, \sigma_D) = \sqrt{I_x^2(x, \sigma_D) + I_y^2(x, \sigma_D)}$$

Only the maxima above a given threshold are considered (due to noise and slight edges). As a result, the edges for each scale are obtained.

Corner Detection

The key idea is to attain the partial derivatives required for a construction of an autocorrelation matrix by using the convolution with the Gabor wavelets.

$$\mu(x, \sigma_I, \sigma_D) = \sigma_D^2 g(\sigma_I) * \begin{bmatrix} I_x^2(x, \sigma_D) & I_x I_y(x, \sigma_D) \\ I_x I_y(x, \sigma_D) & I_y^2(x, \sigma_D) \end{bmatrix}$$

A Gaussian window of SI scale is used to determine the average of the derivatives. On this matrix, detectors are based. Also here, it is essential to use such a Gabor wavelet which serves as the first order partial differential operator.

Blob Detection

Following the same principle, BLOB'S can be detected [14] from the second order partial derivatives using a Hessian matrix

$$H(x, \sigma_D) = \begin{bmatrix} I_{XX}(x, \sigma_D) & I_{xy}(x, \sigma_D) \\ I_{xy}(x, \sigma_D) & I_{yy}(x, \sigma_D) \end{bmatrix}$$

5. CONCLUSION AND FUTURE WORK

In search of the noble feature selection and extraction method different digital image database based data retrieval techniques are studied. These techniques are used with different kind of systems. The in our future work we will incorporate all the three feature extraction techniques for consuming with the object recognition task. In this purpose for colour feature analysis grid colour movement, for texture LBP and for finding the edge detection Gabor filter is utilized for discussion.

7. REFERENCE

ISSN 2229-5518

1. KRYSTIAN MIKOLAJCZYK, TINNE TUYTELAARS, “Local Image Features”, Universiteit Leuven, Kasteelpark Arenberg 10, Leuven, Belgium

2. Minsu Cho, Jungmin Lee, and Kyoung Mu Lee,“Reweighted Random Walks for Graph Matching”, Computer Vision–ECCV 2010, 2010 – Springer

3. Priti Maheswary, Dr. Namita Srivastava, “Retrieval of Remote Sensing Images Using Colour & Texture Attribute”,(IJCSIS) International Journal of Computer Science and Information Security,Vol. 4,No. 1&2, 2009

4. S. Oraintara, T. T. Nguyen, “Using Phase and Magnitude Information of the Complex directional Filter Bank for Texture Image Retrieval”, IEEE International Conference on Image Processing, Vol. 4, Pages 61-64, Oct. 2007

5. P. S. Hiremath and Jagadeesh Pujari. “Content Based Image Retrieval based on Color, Texture and Shape features using Image and its complement”, International Journal of Computer Science and Security, Volume (1) : Issue (4)

6. Philipp Sandhaus, Mohammad Rabbath, and Susanne Boll,“Employing Aesthetic Principles for Automatic PhotoBook Layout”,Advances in Multimedia Modeling, 2011 – Springer

7. Michele Saad, “Low-Level Color and Texture Feature Extraction for Content-Based Image Retrieval”, Final Project Report, EE 381K: Multi-Dimensional Digital Signal Processing, Pages: 20-28, May 2008

8. Yu-Gang Jiang, Jun Yang, Chong-Wah Ngo, Alexander G. Hauptmann, “Representations of Key point-Based Semantic Concept Detection: A Comprehensive Study”, IEEE, 2008

9. Canny Edge Detection, March 23, 2009

10. Masoud Nosrati, Ronak Karimi, Mehdi Hariri, “Detecting Circular Shapes From Areal Images Using Median Filter and CHT”, World Applied Programming, Vol (2), Issue (1), . 49-54, January 2012

11. Zhenhua Guo,Lei Zhang,David Zhang, “A Completed Modeling of Local Binary Pattern Operator for Texture Classification”, IEEE transaction on image processing, 2010

12. S. Mallat, “A Wavelet Tour of Signal Processing : The Sparse Way”, 3 edition, 2009