

Content-Based Product Image Retrieval and Classification

Guidance Professor: Zheng Tang

Graduate School of Science and Engineering

University of Toyama

He Zhang

目次

Abstract.....	4
1. Introduction.....	7
1.1 Text-based image retrieval	7
1.2 Image retrieval and classification based on content.....	9
1.3 Typical image retrieval systems	15
1.4 Similarity measure	16
1.5 Performance evaluation of retrieval methods.....	17
1.6 The key contributions of the thesis	18
2. Image Representation.....	20
2.1 Introduction.....	20
2.2 Low-level representation.....	21
2.2.1 Globle representation	21
2.2.2 Local feature	25
2.3 Intermediate semantic representation.....	30
2.4 Conclusion	36
3. Product Image Retrieval with Combination of Descriptors	37
3.1 Introduction.....	37
3.2 Feature choice and combination.....	37
3.2.1 Color.....	37
3.2.2 Texture	40
3.2.3 Shape.....	41
3.2.4 Descriptor combination.....	42
3.3 Experiment.....	43
3.3.1 Experiment set.....	43
3.3.2 Result and analysis.....	43
3.4 Conclusion	46
4. Image Retrieval System based on Visual Attention Model	47
4.1 Introduction.....	47
4.2 Saliency map algorithm.....	49
4.3 Saliency edge extraction algorithm	49
4.4 Image retrieval algorithm combining of saliency object and edge information.....	50
4.5 Experiment.....	52
4.5.1 Experiment set.....	52
4.5.2 Result and analysis.....	52
4.6 Conclusion	54
5. Product classification with Stacked Auto-Encoder Cclassifier	55
5.1 Introduction.....	55
5.2 Stacked auto-encoder	56
5.3 Experiment.....	58
5.4 Conclusion	61
6. Product Classification based on SVM and PHOG Descriptor	62
6.1 Introduction.....	62

6.2	Support vector machine.....	62
6.3	Image descriptor.....	72
6.3.1	Local shape	72
6.3.2	Spatial layout.....	73
6.4	Experiment and Result	73
6.4.1	Experiment set.....	73
6.4.2	Experimental results and analysis	75
6.5	Conclusion	76
7.	Conclusions.....	77
	Acknowledgements.....	80
	References.....	81

Abstract

Content-based image retrieval (CBIR) is an application of computer vision techniques to the image retrieval problem (searching for digital images in large databases). The "content-based" means that the search analyzes the contents of the image rather than the metadata such as keywords, tags, or descriptions which are associated with the image. The term "content" in this context might refer to colors, shapes, textures, or any other information that can be derived from the image itself. As humans manually enter keywords for images in a large database can be inefficient, expensive and may not capture every keyword that describes the image, a system that can filter images based on their content would provide better indexing and return more accurate results.

Electronic commerce emerged in the 1970s. With the development of internet, online shopping becomes more and more popular. Product images become the main form of commodity exhibition, and is also the decisive factor for consumers to understand and purchase goods. But so far, most of the shopping sites still rely on the establishment of keyword indexing. The retrieval methods based on the keywords can not retrieve accurately and unable to describe the visual content well, which can not meet the needs of the users. Consequently, content-based product image retrieval and classification has become a hot topic of current research.

In this thesis, we proposed a combination algorithm of color descriptor, (LBP) texture descriptor and (HOG) shape descriptor for product image retrieval. The color histogram has the advantages of simple, invariant of image rotation, scale and translation. however, the color histogram lost the space distribution information. The HOG is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization. The LBP analyzes the fix window features with structure methods and extract global feature with statistic methods. The LBP and HOG features provide texture and shape information of an object within an image, respectively. In addition, they are robust to noise, so it is beneficial to combine the three features

together. The product image retrieval experiments indicate the combination can boost the performance of retrieval system significantly.

(HVS) model is used by visual experts to deal with biological and psychological processes that are not yet fully understood. Such a model is used to simplify the behaviors of a very complex system. The study on HVS shows that, when people observed images, the brain based on visual attention mechanism can quickly respond to the area of interest and draw visual attention to the part of the image. Consequently, it's certainly reasonable to establish (VAM) through simulation of the human visual system to get the most attractive part and represent it with a gray scale image. This thesis analyzes the existing visual attention model features and apply visual attention model to the product image retrieval algorithm. Firstly, the saliency map is generated based on visual attention model, on which the saliency part is extracted with dynamic threshold method. Then, the edge information of saliency part is obtained through Canny operator. Image retrieval is implemented through combining the color histogram of saliency map and gradient direction histogram of saliency edge. The proposed algorithm highlights the perception of object areas, inhibits background effects and improves retrieval performance.

For the efficiency of product retrieve, it is of necessity to classify the numerous products into some categories automatically. Each category can be classified into many sub-categories. First, we implement product classification depending on the visual characteristics and stacked auto-encoder classifier. A stacked auto-encoder consists of multiple layers of sparse auto-encoders, the outputs of each layer is the inputs of the successive layer. An auto-encoder attempts to learn appropriate features to represent its raw input, and higher layers tend to learn higher-order features, which construct a hierarchical grouping of the input. Second, (SVMs) and (PHOG) descriptor are employed to implement product classification. The SVMs can efficiently perform a non-linear classification using the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. The PHOG is an excellent image global shape descriptor, which consists of HOG over sub-region of each image

at each resolution level. In this thesis, we adopt SVM classifier combined with PHOG descriptors to implement product-image classification. Experimental results showed the effectiveness of the proposed algorithms.

Keywords: Content-based image retrieval, product image retrieval and classification, Visual Attention Model, Support Vector Machine, Pyramid of Histograms of Orientation Gradients, stacked auto-encoder

1. Introduction

There are various ways for human to percept the outside world, such as vision, hearing, smell, touch, taste, etc. However, the proportion of information obtained with vision is above 70%, much more than the others. Just as the saying goes, ‘one picture worth a thousand words’, compared to other media, the visual information is much more intuitive, vivid, and rich.

Electronic commerce, commonly known as E-commerce, is a type of industry where the buying and selling of products or services is conducted over electronic systems such as the Internet and other computer networks. Electronic commerce emerged in the 1970s. Generalized e-commerce refers to all the commercial activities using the web (not limited to electronic trading), such as B2B, online retail, online group-buying, electronic payment. With the development of internet, online shopping with the internet becomes more and more popular. Product images become the main form of commodity exhibition, and also become the decisive factor for consumers to understand and purchase goods. It becomes much easier to obtain various product images, which not only brings people conveniences, but also brings a lot of problems. For example, how to quickly find the desired image on the massive image databases.

There are two main types of image retrieval techniques. One is based on text (text based image retrieval), and the other is based on content (Content based image retrieval). But so far, most of the shopping sites still rely on the establishment of keyword indexing. The retrieval methods based on the keywords can not retrieve accurately and unable to describe the visual content well, which can not meet the needs of users.

1.1 Text-based image retrieval

The relevant text-based image retrieval research began from the 1970s. If the text information can be obtained, the image retrieval can be implemented directly. So far, most of the shopping sites still rely on the establishment of keyword indexing, i.e. text-based image retrieval.

Since the text based search system can support the queries of large numbers of data with multiple ways and the text information can be relatively easily containing user intentions, the users can get the corresponding results through entering the query words. This convenient characteristic makes the annotation-based image retrieval become the most popular image retrieval methods. Most of the Internet search engine (such as Google, Yahoo, Baidu) have offered text-based image retrieval functions. Textual information about images can be easily searched using existing technology, however, it requires humans to manually describe each image in the database. This is impractical for very large databases or if the images are generated automatically. This may leave out images that use different synonyms in their descriptions.

Figure.1.1 illustrates the complexity of text-based retrieval in E-commerce setting. The product information may be transmitted by multiple merchants through online commerce engine; the provided product description text is often inadequate. As Figure 1.1 (a) shows, a description of the motherboard and the mouse is only shown "P43A" and "sansun04128". In addition, many e-commerce sites may accept the product information provided by thousands of businessmen every day, the description style may be various, for example, some like to use "notebook" while others may use "laptop". On the other hand, the descriptions of different commodity categories may be text overlay, which may affect the retrieval results. Figure 1.1 (b) illustrates two completely different products (laptop and battery), the description of laptop ("Acer Travel-Mate 40621LCI with battery" is almost identical to the battery ("Acer Travel-Mate 40621LCI battery") the minor difference is just the conjunction "with".



(a)



(b)

Figure 1.1 Complexity of text-based retrieval in E-commerce setting.

(a) Short under-descriptive text, (b) Overlapping text across categories

Due to these drawbacks, in recent years, many researchers have developed methods of image retrieval and classification based on contents. Consequently, content-based image retrieval has become a hot topic of current research.

However, due to various types of image sets, it is difficult to form a common model. There still exists a great gap between the research and the practical applications.

1.2 Image retrieval and classification based on content

The CBIR is an approximate matching technology, which integrates fields of

computer vision, image processing, image understanding, databases, and etc^[1-7]. The CBIR independently measure the content of the image similarity between the images to achieve image search.

The most common method to compare two images based on content-based is using an image distance to measure their distance. An image distance measure compares the similarity of two images in various dimensions such as color, texture, shape, and others. Search results then can be sorted based on their distance to the queried image. A distance of 0 means an exact match with the query image, a value greater than 0 indicates various degrees of similarities between the images.

The CBIR is divided into three layers: feature layer, object layer and semantic concept layers, each corresponding to an layer of the image semantic. The feature layer is the specific color, shape, texture and other visual characteristics and their combination; object layer is the objects that appear in the image and the spatial relationships between the objects, semantics layer refers to the concept of person-level image contents, which can be divided into 3 parts from low to high, they are scene semantics, behavioral and emotional semantics. Scene semantics refers to the scene of the image, such as beach, sky, etc. Behavioral semantics primarily refers to the behavior of objects in the image, such as a volleyball match emotional semantic refers to the subjective feelings of the image (e.g. happy, angry, etc.). All of them combine together to express meaning of an image.

The CBIR is build upon computer vision and image comprehension theory, and it's a combination of artificial intelligence, object oriented technology, cognitive psychology, database and other multidisciplinary knowledge. When describing image content, we extract visual features from image automatically rather than rely on manual annotation. And retrieval process is not about keywords matching but about similarity match. The CBIR has many advantages like impersonal, labor-saving, complex description, strong universality and broad prospect. The general framework of the CBIR is show in Figure 1.2.

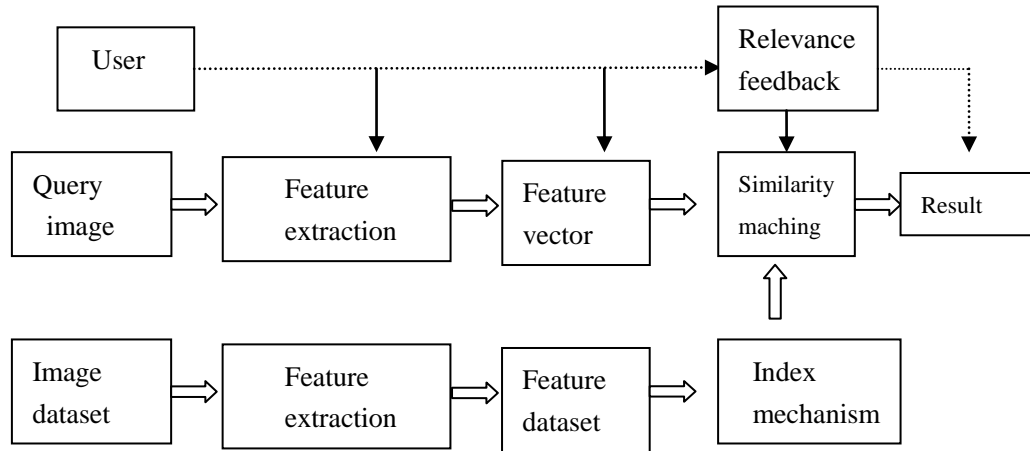


Figure 1.2 The general framework of CBIR

After several decades of developments, the image retrieval technology has made great progress, but some new problems are constantly emerging. The main questions under discussion are as follows:

(1) Semantic gap problem: the semantic gap is the gap that exists between the underlying visual feature space and the user's high-level semantic concept space, which has been a big problem that puzzled the CBIR and web image retrieval.

(2) Lack of labeled samples problem: The algorithms based on the machine learning need a large number of labeled samples to finish the training process, but the artificial labeling is costly and there are limited labeled samples to be obtained.

(3) Feature extraction problem: Image feature extraction is a key step to image retrieval, but there is a gap between the underlying features that the computer can describe and the high-level semantic features that humans can understand, which makes the computer unable to describe the image well. Some factors will affect the extraction of image feature, such as perspective differences, lighting effects, scale differences, occlusion problem and complicated background. Figure 1.3 and Figure 1.4 shows the in-class and the scale variation of product images.



Figure 1.3 In-class variation illustration of product images



Figure 1.4 Scale variation of product images

(4) Large scale problem: With the rapid growth of the scale of the image dataset, the data indexing technology need to be adjusted accordingly to ensure real-time data search, The CBIR always uses high dimensional image feature, the commonly used indexing technologies are kd-tree, R-tree, LSH (Local Sensitive Hashing) and so on. For the massive level of image retrieval, the distributed storage and parallel

computing are generally used. Large-scale internet data also has a cumulative characteristic, that is, the data are increasing every time, this requires image retrieval algorithm which can dynamically update, and has the characteristics of the incremental learning.

Currently, the applications of the CBIR in electronic commerce are only limited to several commodity categories^[8,9,10], such as electronic, clothing and textile. Image retrieval of single commodity categories has the characteristics of small image dataset and content outstanding. The process of image retrieval needs full consideration to the features of the product category and select the appropriate features to improve the retrieval accuracy. As the scale of the image retrieval is limited, there is little influence on feature selection to the retrieval speed.

Some shopping sites have applied CBIR technologies and proposed some new retrieval methods (Figure 1.5, 1.6, 1.7)^[40,41,42]. For example, like.com^[42] (Google has bought like.com to implement Google shopping) firstly applied CBIR technology to the electronic commerce shopping platform, in which consumers can choose commodities features (such as color, shape and style) for product retrieval. Furthermore, like.com provides a retrieval method based on product details, on which consumers can select a region of interest on the image to search. Google Labs released a similar image search service with one kind of two-layer image retrieval method. Although the CBIR technology has effectively used in many fields, it is still a relatively new research direction to study the applications of CBIR on e-commerce.

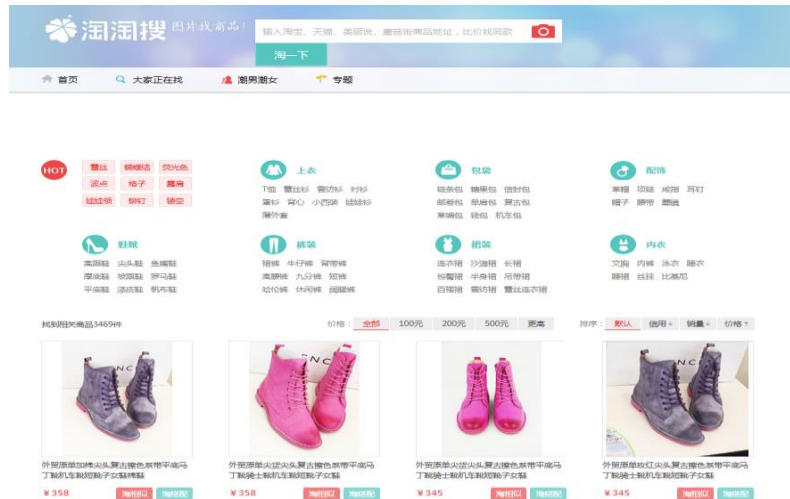


Figure 1.5 Taotaosou interface



Figure 1.6 Antusou interface

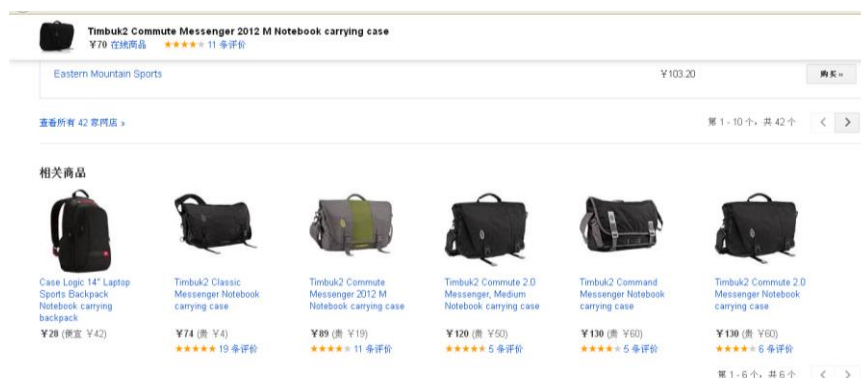


Figure 1.7 Google shopping

1.3 Typical image retrieval systems

(1) QBIC system^[1,5]

This is a retrieval system composed of content-based image and dynamic scene, developed by the research center of IBM Almadon and it is the first system software that has applied to business. QBIC system offers several kinds of retrieval mode such as example retrieval, sketch contours and so on. The system use color, texture and shape and other features to describe image content, and search image in combination with keyword, what is more, it use R^* as its high-dimensional index structure.

(2) Retrieval ware system^[2]

The system is a CBIR engine developed by Excalibur. The new version of Retrieval Ware system offers several joint-retrieval ways about content properties information, these properties include color, color structure, texture, aspect ratio, shape as well as brightness structure. The system accept multi-feature fusion, furthermore, the users can set weight factor for every vision feature when searching the image.

(3) Photobook system^[3]

The system is an interactive search engine that used for image retrieval and image browsing and was invented by the multimedia lab of MIT. Photoshop system made up of three sub-systems which are respectively used for shape, texture and facial feature extraction. USA police has put this technology into use and has done good job. FourEyes is the latest version of the system, in which Pichard came up with another idea, that is let the users to take in part in the process of retrieval and image interpretation. Meanwhile, owe to human's subjective perception, they put forward to image retrieval based on society of model, the method can better realize a kind of image retrieval based on human-computer interaction mode.

(4) MARS

MARS (Multimedia Analysis and Retrieval System) a novel visual feature-based machine-learning framework for large-scale semantic modeling and classification of

image and video content. The system has something different from other systems, that is the system involves computer vision, database management and information retrieval and other cross-disciplinary knowledge. It adopts a more comprehensive characteristics descriptors based on image visual content. In addition, it also supports multi-characteristic fusion retrieval based on multi-tree index structure.

(5) Mires system.

The system is invented by Key Laboratory of Intelligence Information Processing of The institute of Computing Technology. This system combines the high-level semantic features with low-level visual features and make a description. It uses machine learning to extract the semantic categories of images and to describe the high-level semantic contents of images. The low-level visual features that was used in the system include color, texture, edge and so on. Mires system realizes the positive and negative relevance feedback method based on kernel function and SVM.

(6) The ImgRetr system is a CBIR engine based on web invented by Tsinghua University. Visual content features used by this system mainly include color, texture, color and contour distribution. In addition, Microsoft Research Asia, institute of pattern recognition and control of CAS, Huazhong university of Science and Technology, National Defense University, Fudan University has put a lot of resources in the development of retrieval experiment system of their own, and has obtained certain achievements.

1.4 Similarity measure

It is undoubtedly important to define a suitable image feature similarity measure for the effect of the image retrieval. Since most of the features are represented by vectors, the vector space model is generally used to measure the similarity. The metric commonly used in image search are blocks distance and Euclidean distance, which are defined as follows:

$$\rho(x, y) = \sum_{i=1}^n x_i - y_i \quad (1.1)$$

$$\rho(x, y) = \sqrt{\sum_{i=1}^n |x_i - y_i|} \quad (1.2)$$

Another metric to measure the distance between two images is the so-called histogram intersection method, which computes the ratio of shared pixel levels, the formula is shown in the following Equation 1.3 :

$$\rho(x, y) = \sum_{i=1}^n \min \{x_i, y_i\} / \sum_{i=1}^n y_i \quad (1.3)$$

where x and y are the two histograms with n dimensions. The value is limited to $[0, 1]$, the bigger the value, the more similar between the two representations.

Another more effective measure is the quadratic distance based metric. Histogram between M and N quadratic distance can be defined as:

$$D = (N - M)^T A (N - M) \quad (1.4)$$

Where A is the color similarity weighting matrix, which can assign different weigh to different dimension.

1.5 Performance evaluation of retrieval methods

Recall also called recall rate, which refers to the ratio of the image number that the system has retrieved and relevant to the query image to the image number that relevant to the query image in the image gallery. Recall represents the accuracy of relevant images that a system has retrieved.

Suppose P is the test image, a is a set of image that the system has retrieved, b is the number of irrelevant image retrieved, c is the number of relevant image that has not retrieved yet. The d is the number of irrelevant excluded images. The Recall R_{recall} , Precision $P_{precision}$, False negative $FN_{false-negative}$, False positive $FD_{false-positive}$ are defined as follows:

Table1 Some notions for the performance evaluation of retrieval methods

	Relevance	Irrelevance
Detection	a	b
No detection	c	d

The calculation recall ratio formula is as follows:

$$R_{recall} = \frac{a}{a + c} \quad (1.5)$$

$$R_{precision} = \frac{a}{a + b} \quad (1.6)$$

$$FN_{False-negative} = \frac{c}{a + c} \quad (1.7)$$

$$FD_{False-positive} = \frac{b}{a + b} \quad (1.8)$$

1.6 The key contributions of the thesis

Content-based image representation is the basis of image retrieval and classification. Chapter 2 explores the image representation methods from the aspects of low-level representation and intermediate semantic representation.

In chapter 3, we proposed a combination algorithm of color descriptor, LBP texture descriptor and HOG shape descriptor for product image retrieval. The product image retrieval experiments indicate the combination can boost the performance of retrieval system significantly.

Chapter 4 analyzes the existing visual attention model features and apply visual attention model to the product image retrieval algorithm. Firstly, the saliency map is generated based on visual attention model, on which the saliency part is extracted with dynamic threshold method. Then, the edge information of saliency part is

obtained through Canny operator. Image retrieval is implemented through combining the color histogram of saliency map and gradient direction histogram of saliency edge.

Chapter 5 implements product classification depending on the visual characteristics and stacked auto-encoder classifier. An auto-encoder attempts to learn appropriate features to represent its raw input, and higher layers tend to learn higher-order features, which construct a hierarchical grouping of the input.

In chapter 6, SVM (Support Vector Machine) and PHOG (Pyramid of Histograms of Orientation Gradients) descriptor are employed to implement product classification. Support vector machines can efficiently perform a non-linear classification using the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. PHOG is an excellent image global shape descriptor, which consists of a histogram of orientation gradients over sub-region of each image at each resolution level.

In the end, the thesis summarizes the research.

2. Image Representation

2.1 Introduction

In the field of image processing, the concept of feature is used to denote a piece of information which is relevant for solving the computational task related to a certain application. The feature concept is very general and the choice of features in a particular computer vision system may be highly dependent on the specific problem at hand.

Feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (e.g. the same measurement in both feet and meters) then the input data will be transformed into a reduced representation set of features (also named features vector). feature extraction refers to transforming the input data into the set of features. If the features extracted are carefully chosen, it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

In this chapter, the so-called image representation indicates the description of an image in the feature space, which transforms from the original two-dimension space. The pixels are usually vast and susceptible to the lighting, noise and other factors, so they are not suitable for image retrieval and classification, and must be converted into a compact, robust representation which can reflect the image semantic. This chapter will explore the image representation methods, from the image low-level representation to intermediate semantic representation, which are the basis for further research of product image retrieval and classification.

2.2 Low-level representation

2.2.1 Globle representation

Color feature

Being widely used in image retrieval technology, color feature extraction is the most intuitive, expressive visual feature, and is one of the most important image features. Although image feature alone can not effectively distinguish the image content, color feature is often closely related to an object. For example, most of the basketballs are red, and most of the grassland are green. Besides, with respect to other features of image, color image have better robustness and is less dependent on change of image size, direction, and shift. There are many ways to describe color feature, the most commonly used are color histogram, cumulative color histogram, center moment method and so on. Among them, CLD recommended by MPEG-7 standard can be used in describing image's color spatial distribution. Compares to the traditional color histogram, CLD describe spatial information and is widely used in image matches, particularly for similarity retrieval based on space structure.

The CLD's extraction process includes the following steps: image block, representative color extraction, DCT, zigzag scanning.

The MPEG-7 recommends using $YCrCb$ color space, therefore, color space must be converted before extracting CLD of image. Detailed steps are as follows:

(1) Divide the image into 8×8 blocks, calculate the of pixel's color components for each block's and set average value as each block's representative color.

(2) Convert RGB image to $YCrCb$ image, we can get 64 blocks of image with different brightness, hue and saturation

$$\begin{aligned} Y &= 0.299 \times R + 0.587 \times G + 0.114 \times B - 128 \\ Cb &= 0.169 \times R - 0.331 \times G + 0.500 \times B \\ Cr &= 0.500 \times R - 0.419 \times G - 0.081 \times B \end{aligned} \quad (2.1)$$

(3) Compute DCT:

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}$$

$$0 \leq p \leq M-1, \quad 0 \leq q \leq N-1 \quad (2.2)$$

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}} & p=0 \\ \frac{2}{\sqrt{M}} & 1 \leq p \leq M-1 \end{cases} \quad \alpha_q = \begin{cases} \frac{1}{\sqrt{N}} & q=0 \\ \frac{2}{\sqrt{N}} & 1 \leq q \leq N-1 \end{cases} \quad (2.3)$$

(4) Make zigzag scanning for the 64 coefficient, we get coefficient of low frequency 8 by 8 matrix.

(5) Similarity measurement. For the two groups of CLDs based on query image and stored image, we can work out the similarity of the query image and the stored image.

$$D = \sqrt{\sum_i w_{yi} (DY_i - DY'_i)^2} + \sqrt{\sum_i w_{bi} (DCb_i - DCb'_i)^2} + \sqrt{\sum_i w_{ri} (DCr_i - DCr'_i)^2} \quad (2.4)$$

i show the scanning order of zigzag coefficient, w is the weight of coefficient.

The recommend weight coefficient is as below:

$$w_y = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Texture feature

Texture structure not only reflects the arrangement of surface, but also reflects its contact with the surrounding background^[13,14,18]. Texture is often seen as a measure of local features of an image. Although the distinction using image texture is less notable comparing with visual feature like shape, color, etc. It is one of the important properties of the image itself. Texture structure plays a supporting role in image retrieval. Different product image generally has notable texture feature of itself. The product of the same type may exhibit different textures feature due to different type of material. Therefore, texture feature extraction is conducive to commodity image retrieval.

Traditional texture feature extraction mainly includes gray-level co-occurrence matrix, which is a traditional method. Besides, spectrum can also be used for extracting texture feature of an image. Among them, Gabor Wavelet transform provides a very useful method for texture analysis, and was widely used in image retrieval. It is considered one of the best way for image recognition.

Using Gabor filter to extract texture features proved to be an ideal choice As Gabor can be used as edge detector whose direction and size can be adjusted. And texture is described by two kinds of local feature's statistic property of given area, that is edges and line.

Given an $P \times Q$ image $I(x, y)$, the discrete Gabor transformation to the image is as below:

$$G_{mn}(x, y) = \sum_s \sum_t I(x-s, y-t) \psi_{mn}^*(s, t) \quad (2.5)$$

$$\psi(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right] \cdot \exp(j2\pi Wx) \quad (2.6)$$

$$\begin{aligned} \psi_{mn}(x, y) &= a^{-m} \psi(u, v) \\ m &= 0, 1, \dots, M-1; \quad n = 0, 1, \dots, N-1 \end{aligned} \quad (2.7)$$

where the parameter s and t are the mask size of filter. m and n are the scale and orientation of the filter.

$$\begin{aligned} u &= a^{-m} (x \cos \theta + y \sin \theta), \quad v = a^{-m} (-x \sin \theta + y \cos \theta) \\ a &> 1, \quad \theta = n\pi / N \end{aligned} \quad (2.8)$$

$$a = \left(\frac{U_h}{U_l} \right)^{\frac{1}{M-1}}$$

W is the modulation frequency of Gaussian function.

$$\begin{aligned} W_{m,n} &= a^m U_l \\ \sigma_{x,m,n} &= \frac{(a+1)\sqrt{2\ln 2}}{2\pi a^m (a-1)U_l} \\ \sigma_{y,m,n} &= \frac{1}{2\pi \tan(\frac{\pi}{2N}) \sqrt{\frac{U_h^2}{2\ln 2} - (\frac{1}{2\pi\sigma_{x,m,n}})^2}} \end{aligned} \quad (2.9)$$

The energy information of the image I with difference scales and orientations is calculated as:

$$E(m,n) = \sum_x \sum_y |G_{mn}(x,y)|, \quad m = 0,1,\dots,M-1; \quad n = 0,1,\dots,N-1 \quad (2.10)$$

Shape feature

Shape is the key information for content-based image retrieval^[11]. Many shape description and similarity measurement techniques have been developed in the past. Shape features are presented in two ways: one is based on the shape of the contour description and the other is region-based shape description.

This is the most common and general classification and it is proposed by MPEG-7. It is based on the use of shape boundary points as opposed to shape interior points. Under each class, different methods are further divided into structural approaches and global approaches. This sub-class is based on whether the shape is represented as a whole or represented by segments/sections (primitives).

2.2.2 Local feature

Local features computed for interest regions have proved to be very successful in applications of image retrieval and classification. Various invariant detectors and descriptors have been proposed and evaluated in the context of viewpoint invariant matching^[43,44].

Local feature extraction

(1) Region Detectors

The concept of feature detection refers to methods to compute abstractions of image information and make local decisions at every image point whether there is an image feature of a given type at that point or not. The resulting features will be subsets of the image domain, often in the form of isolated points, continuous curves or connected regions^[45].

The desirable property for a feature detector is repeatability: whether or not the same feature will be detected in two or more different images of the same scene.

As a built-in pre-requisite to feature detection, the input image is usually smoothed by a Gaussian kernel in a scale-space representation and one or several feature images are computed, often expressed in terms of local derivative operations.

Region detectors use different image measurements and can be invariant to various transformations. There is a number of detectors invariant to affine transformations which provide elliptical regions. However, the region locations and scales are the same in their scale invariant versions, only the shape of regions varies.

(2) Corner detector

A corner can be defined as the intersection of two edges. Without loss of generality, we will assume a gray-scale 2-dimensional image is used. Let this image be given by I . Consider taking an image patch over the area (u, v) and shifting it by (x, y) . The weighted sum of squared differences (SSD) between these two patches, denoted S , is given by:

$$S(x, y) = \sum_u \sum_v w(u, v) (I(u+x, v+y) - I(u, v))^2 \quad (2.11)$$

$I(u+x, v+y)$ can be approximated by a Taylor expansion. Let I_x and I_y be the partial derivatives of I , such that

$$I(u+x, v+y) \approx I(u, v) + I_x(u, v)x + I_y(u, v)y \quad (2.12)$$

$$S(x, y) \approx \sum_u \sum_v w(u, v) (I_x(u, v)x + I_y(u, v)y)^2 \quad (2.13)$$

$$S(x, y) \approx (x \ y) A \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.14)$$

where A is the structure tensor,

$$A = \sum_u \sum_v w(u, v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} = \begin{bmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{bmatrix} \quad (2.14)$$

This matrix is a Harris matrix. By analyzing the eigenvalues of A , this characterization can be expressed in the following way:

If $\lambda_1 \approx 0$ and $\lambda_2 \approx 0$ then this pixel (x, y) has no features of interest.

If $\lambda_1 \approx 0$ and λ_2 has some large positive value, then an edge is found.

If λ_1 and λ_2 have large positive values, then a corner is found.

The computation of the eigenvalues is computationally expensive. Instead, the following function M_c is suggested, where K is a tunable sensitivity parameter:

$$M_c = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2 = \det(A) - k * \text{trace}^2(A) \quad (2.15)$$

(3) Blob detector

Informally, a blob is a region of a digital image in which all the points can be considered to be similar to each other in some sense^[46].

There are two main classes of blob detectors:

① Differential methods, which are based on derivatives of the function with respect to position;

② Methods based on local extrema, which are based on finding the local maximum and minimum of the function.

The Laplacian of the Gaussian (LoG) is the most common blob detector. Given an input image $f(x, y)$, this image is convolved by a Gaussian kernel to give a scale space representation.

$$L(x, y; t) = g(x, y, t) * f(x, y) \quad (2.16)$$

The Gaussian kernel at a certain scale t is as follows.

$$g(x, y; t) = \frac{1}{2\pi t^2} e^{-\frac{x^2 + y^2}{2t^2}} \quad (2.17)$$

Then, the Laplacian operator

$$\nabla^2 L = L_{xx} + L_{yy} \quad (2.18)$$

is computed. The operator response is strongly dependent on the size of the blob structures and the size of the Gaussian kernel used for pre-smoothing. In order to automatically capture blobs of different (unknown) size in the image domain, a multi-scale approach is therefore necessary. A straightforward way to obtain a multi-scale blob detector with automatic scale selection is to consider the scale-normalized Laplacian operator

$$\nabla_{norm}^2 L(x, y; t) = t(L_{xx} + L_{yy}) \quad (2.19)$$

and to detect scale-space maxima/minima, that are points that are simultaneously local maxima/minima of $\nabla_{norm}^2 L$ with respect to both space and scale ^(47,48). Thus, given a discrete two-dimensional input image $f(x, y)$ a three-dimensional discrete scale-space volume $L(x, y; t)$ is computed according to

$$(\hat{x}, \hat{y}, \hat{t}) = \arg \max \min_{local_{(x,y,t)}} (\nabla_{norm}^2 L(x, y; t)) \quad (2.20)$$

The difference of gaussians (DOG) approach

From the fact that the scale space representation $L(x, y; t)$ satisfies the diffusion equation

$$\partial_t L(x, y; t) = \frac{1}{2} \nabla^2 L \quad (2.21)$$

$\nabla^2 L(x, y, t)$ can also be computed as the limit case of the difference between two Gaussian smoothed images

$$\nabla_{norm}^2 L(x, y; t) \approx \frac{t}{\Delta t} (L(x, y; t + \Delta t) - L(x, y; t - \Delta t)) \quad (2.22)$$

DOG can be seen as an approximation of the Laplacian operator^[49] and used in the SIFT algorithm^[50].

Local descriptor

The local descriptors characterize the detected local invariant regions. There is a large number of descriptors have been developed. The simplest descriptor is a vector of image pixels. However, the high dimensionality of such a description results in a high computational complexity. The distribution-based descriptors use histograms to represent different characteristics of appearance or shape. A simple descriptor is the distribution of the pixel intensities represented by a histogram. D and G^[50] proposed a scale invariant feature transform (SIFT), which combines a scale invariant region detector and a gradient-based descriptor in the detected regions^[50]. The descriptor is represented by a histogram of gradient locations and orientations.

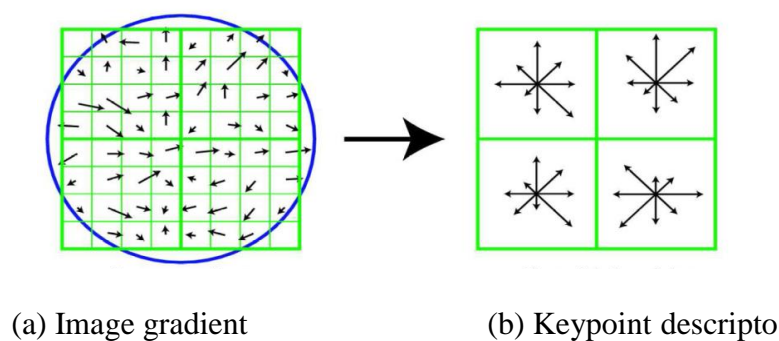


Figure 2.1 The process illustration of SIFT descriptors

As Figure 2.1 illustrates, the resulting descriptor is of 128 dimensions. Each orientation plane represents the gradient magnitude corresponding to a given orientation. The gradient locations and orientations are quantized and weighted by the gradient magnitudes. The descriptor is normalized by the square root of the sum of squared components, which makes the descriptor robust to illumination, small geometric distortions and small errors in the region detection.

Similar to the SIFT descriptor, geometric histogram^[51] and shape context^[52] compute a 3D histogram of location and orientation for edge points. Gradient

location-orientation histogram (GLOH) is also an extension of the SIFT descriptor, which computes for a log-polar location grid in radial direction. Shape context is a 3D histogram of edge point (extracted by the Canny detector) locations and orientations.

Mikolajczyk, etc.^[53] proposed an extension of the SIFT descriptor, and showed that it outperformed the original method. Furthermore, Mikolajczyk, etc.^[53] observed that the ranking of the descriptors is mostly independent of the interest region detector and that the SIFT based descriptors perform best. This shows the robustness and the distinctive character of the region-based SIFT descriptor. Shape context also shows a high performance. However, for textured scenes or when edges are not reliable its score is lower.

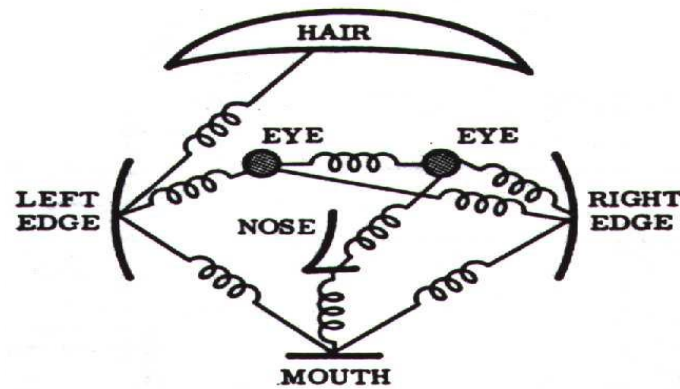
2.3 Intermediate semantic representation

Low-level features describe images with relative simple representations. but once there is a significant background interference or in-class variation, the performance of image classification and retrieval would degrade. The intermediate representation is referred to as semantic modeling, build a bridge of low-level representation and high-level semantic feature of the images.

Semantic Objects: describe the image with the detected objects, which mainly relies on an initial segmentation of the image into meaningful regions, and the segmented regions are labeled as semantic objects. Figure 2.2 illustrates the part-based model for Semantic object detection (a) and constellation model of human face (b) .



(a) Semantic object detection



(b) Constellation model of human face

Figure 2.2 Part-based model

Semantic Properties: semantic is described by a set of statistical properties of the image, such as naturalness, openness and roughness, which are shared by images of a same category. These methods require neither segmentation nor the processing of local regions or objects. The image is described by visual properties. Oliva and Torralba^[54,55,56] proposed a computational model based on a very low dimensional representation referred as the Spatial Envelope. It consists of five perceptual qualities: naturalness, openness, roughness, expansion and ruggedness. These dimensions may be reliably estimated using spectral and coarsely localized information. It is possible to assign a specific interpretation to each dimension: along the openness dimension, the image refers to an open or a closed environment, etc. Figure 2.3 illustrates the organization of man-made environments according to the degrees of openness and expansion^[54].

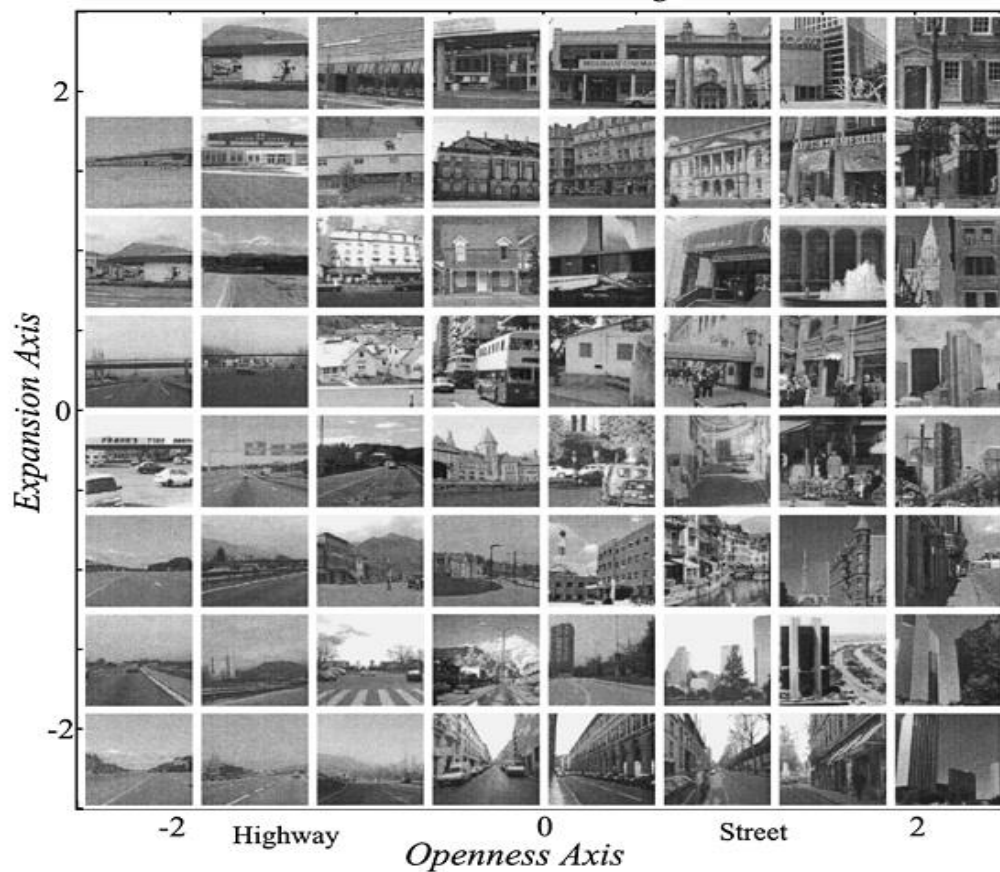


Figure 2.3 Organization of man-made environments according to the degrees of openness and expansion.

Local semantic concepts: represent the image with intermediate properties which are extracted from local descriptors around points.

The bag-of-words model is the representative of local semantic concepts, which was first proposed for document classification and further applied for computer vision applications. Constructing the bag-of-words from the images involves the following steps:

- ① Automatically detect regions,
- ② Compute local descriptors over those regions,
- ③ Quantize the descriptors into visual words to form the visual vocabulary,

④ Find the occurrences in the image of each specific word in the vocabulary in order to build the histogram of visual words.

Figure 2.4 schematically describes the four steps of the bag-of-words model.

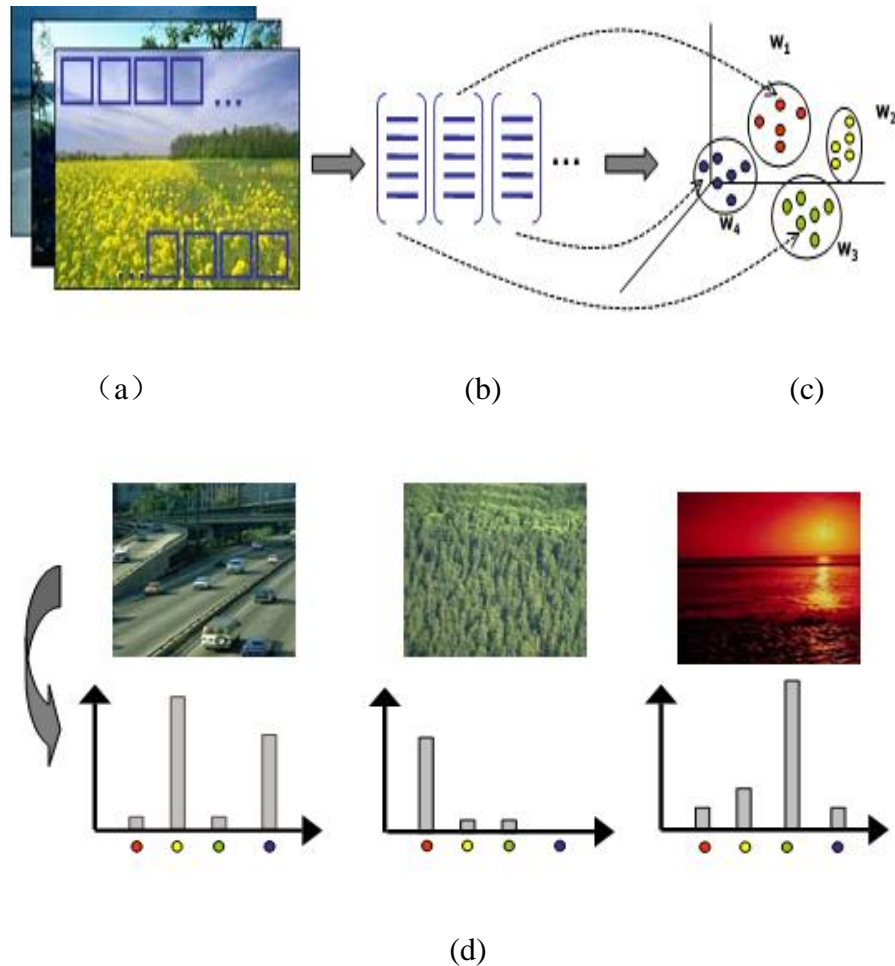


Figure 2.4 Classic bag of word model

(a)Region detection (b) Feature extraction (c) Vector quantization (d) Bag-of-words

Bag of word (BOW) model is one simple case of general bag of word (GBOW) model. Figure 2.5 schematically illustrates the rough steps of the GBOW model.

Let an image I be represented by a set of low-level descriptors (e.g., SIFT) $\{x_i\}, i = 1, \dots, N$, M indicates the number of interesting regions on the image, N_m is

the index of regions, let f and g denote some coding and pooling operators, respectively, z denotes the representation of the whole image by sequentially coding, pooling over all regions, then

(1) The coding process:

$$\alpha_i = f(x_i), \quad i = 1, \dots, N \quad (2.23)$$

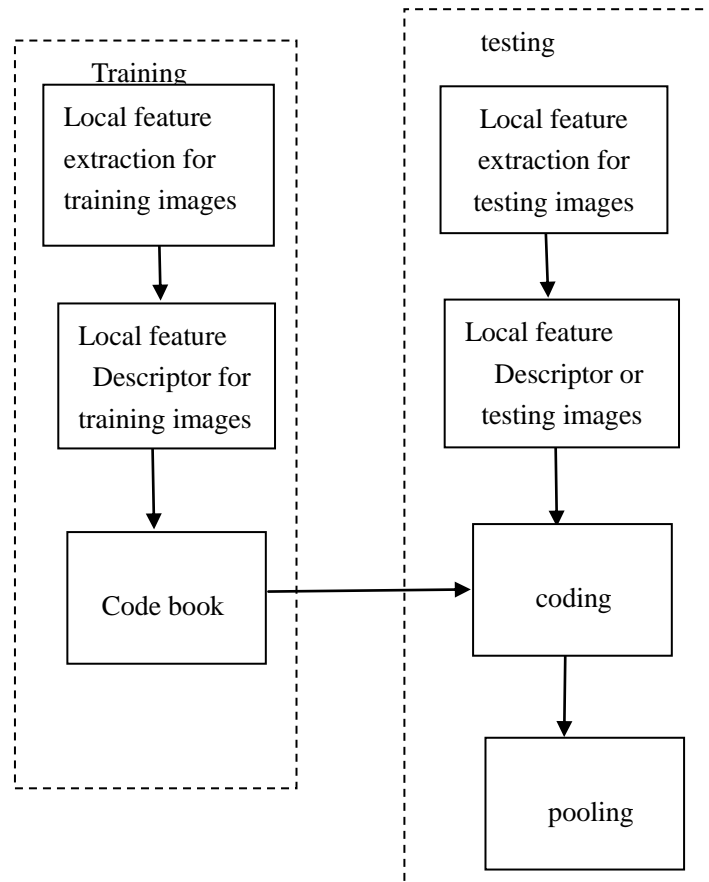


Figure 2.5 The mid-features built with the local features

In the classic bag-of-words framework, the code book usually is learned by an unsupervised algorithm (e.g., K-means), f minimizes the distance to the codebook, and g computes the average over the pooling region:

$$\alpha_i \in \{0,1\}^K, \alpha_{i,j} = 1, \quad \text{iff} \quad j = \underset{k \leq K}{\operatorname{argmin}} \|x_i - d_k\|_2^2 \quad (2.24)$$

Van Gemert J C, etc.^[21] replaces hard quantization by soft quantization:

$$\alpha_{i,j} = \frac{\exp(-\beta \|x_i - d_j\|_2^2)}{\sum_{k=1}^K \exp(-\beta \|x_i - d_k\|_2^2)} \quad (2.25)$$

Wright J, etc.^[58,59] and Yang, etc.^[60] employ sparse coding, which use a linear combination of a small number of codewords to approximate x_i .

$$\alpha_i \in \{0,1\}^K, \alpha_{i,j} = 1 \quad \text{iff} \quad j = \arg \min \|x_i - d_k\|_2^2$$

$$h_{m,j} = \max_{i \in N_m} \alpha_{i,j} \quad \text{for } j = 1, \dots, K,$$
(2.26)

(2) The pooling process:

$$h_m = g(\{\alpha_i\}_{i \in N_m}), \quad m = 1, \dots, M \quad (2.27)$$

① global average pooling

$$h_{m,j} = g(\{\alpha_i\}_{i \in N_m}) = \frac{1}{|N_m|} \sum_{i \in N_m} \alpha_{i,j}, \quad m = 1, \dots, M, j = 1, \dots, K \quad (2.28)$$

where K is the number of visual words.

② max pooling

$$h_{m,j} = \max_{i \in N_m} \alpha_{i,j} \quad m = 1, \dots, M, j = 1, \dots, K \quad (2.29)$$

(3) The image can be represented as:

$$z = [h_1^T \dots h_m^T]. \quad (2.30)$$

2.4 Conclusion

Content-based image representation is the basis of image classification. This chapter explores the image representation methods from the aspects of low-level representation and intermediate semantic representation. The low-level representation can be divided into two categories, global representation and local representation; the intermediate semantic representations include local semantic concept, in which the BOW model on the basis of local semantic concept model is one of the recent mainstream methods for image retrieval and image classification.

3. Product Image Retrieval with Combination of Descriptors

3.1 Introduction

It is a critical problem to find the commerce quickly and accuracy. For the content-based image retrieval, the image is submitted directly to the system, rather the description words for the product. The algorithm compares the features of submitted image with that of the images in the database, the most similar image will be returned to the users. The schematic process is illustrated as Figure 3.1.

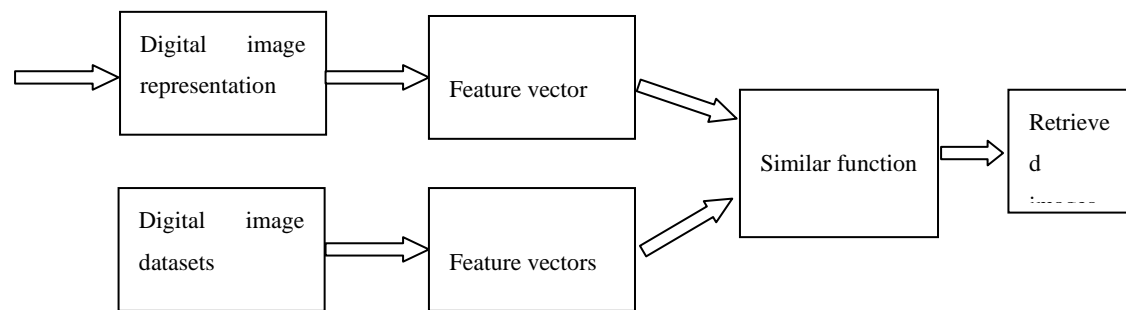


Figure 3.1 The structure of content based retrieval system

3.2 Feature choice and combination

Information from different sources can improve the retrieval performance. The combination of descriptors associated with different visual properties, such as color, texture and shape.

3.2.1 Color

The color signature is extracted from the whole images to produce a global descriptor. The color dictionary $\{C_1, C_2 \dots C_k\}$ is built by quantizing each components separately, and regularly in 4 or 8 bins, leading to 64 or 512 colors in total $r^{[21]}$. In order to reduce the impact of large uniform colored areas, we collect typical and unique colors to build the color dictionary.

- 1) Collect some random images.
- 2) Resize each image to 100*100 pixels and convert to HSV, then split it in some 8*8 blocks.
- 3) Find the most occurring color of each block.
- 4) Cluster the colors from all images with k-means algorithm, produce k color palette.

The HSV color space is employed to the system as its mode is closer to the vision of human than the RGB color space. The conversion from default the RGB space to the HSV space is as follows:

Assuming (r, g, b) are the three components (red, green and blue) of one specific color, the values of which range from 0 to 1. And assuming max as the maximum of the three real number of r, g and b , min is as the minimum of the three components. (h, s, v) are the three components in the HSV space, in which $h \in [0, 360]$ is the hue of the color. $s, v \in [0, 1]$ are the value and saturation of the color, respectively. The transformation formula are as follows ^[22]:

$$h = \begin{cases} 0^0 & \text{if } max = min \\ 60^0 \times \frac{g-b}{max-min} + 0^0, & \text{if } max = r \text{ and } g \geq b \\ 60^0 \times \frac{g-b}{max-min} + 360^0, & \text{if } max = r \text{ and } g < b \\ 60^0 \times \frac{g-b}{max-min} + 120^0, & \text{if } max = g; \\ 60^0 \times \frac{g-b}{max-min} + 240^0, & \text{if } max = b; \end{cases} \quad (3.1)$$

$$l = \frac{1}{2}(\max + \min) \quad (3.2)$$

The color space quantization.

$$H = \begin{cases} 0, h \in (345, 15] \\ 1, h \in (15, 25] \\ 2, h \in (25, 45] \\ 3, h \in (45, 55] \\ 4, h \in (55, 80] \\ 5, h \in (80, 108] \\ 6, h \in (108, 140] \\ 7, h \in (140, 165] \\ 8, h \in (165, 190] \\ 9, h \in (190, 220] \\ 10, h \in (220, 255] \\ 11, h \in (255, 275] \\ 12, h \in (275, 290] \\ 13, h \in (290, 316] \\ 14, h \in (316, 330] \\ 15, h \in (330, 345] \end{cases} \quad S = \begin{cases} 0, s = (0, 0.15] \\ 1, s = (0.15, 0.4] \\ 2, s = (0.4, 0.75] \\ 3, s = (0.75, 1] \end{cases} \quad V = \begin{cases} 0, v = (0, 0.15] \\ 1, v = (0.15, 0.4] \\ 2, v = (0.4, 0.75] \\ 3, v = (0.75, 1] \end{cases} \quad (3.3)$$

For the component hue of the HSV space, 0 represents for red color, 120 for green color, 240 for blue color. The visible spectrum covers the hue region from 0 to 240. In this thesis, the HSV space are quantized into 256 bins, which is computed as illustrated in (3.3).

Color signature computes the histogram of colors in the image for the fixed codebook, firstly, resize the image to 100*100, for each pixel, select the closest color in the color codebook according to the Euclidean distance. The k-dimension histogram is built with the distribution of the color in the codebook.

Inverse document frequency down weights the impact of the common visual words and increases the importance of the rare visual words. The histogram is updated

by applying if weighting terms. The power-law method regularizes the contribution of each color in the final descriptor.

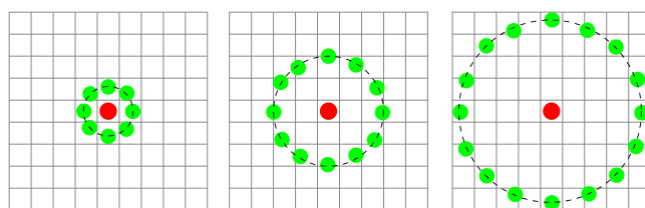
The L1 vector normalization is performed to make the vector comparable. Signatures are compared to find the nearest neighbor of the query images in a signature database, in which the choice of the metric is therefore critical.

3.2.2 Texture

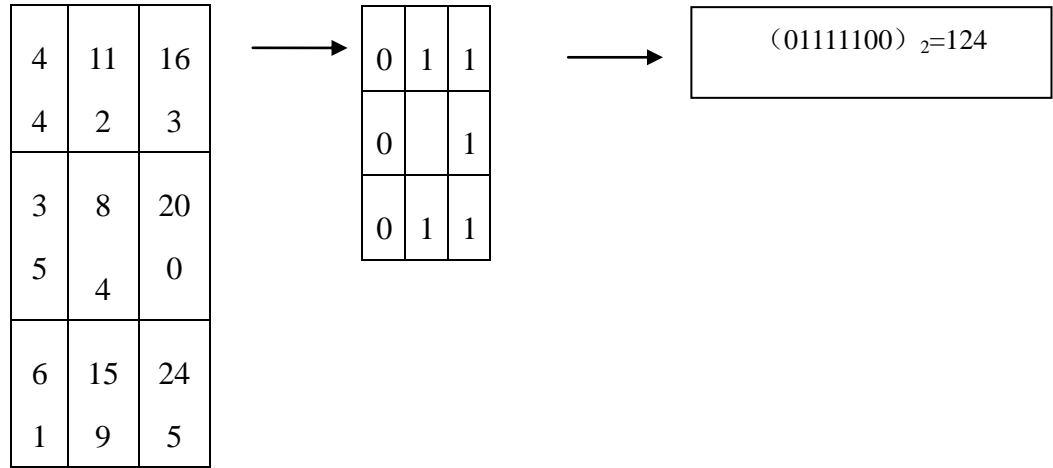
The text features extraction is based on either statistics or structure. Maenpaa T, Ojala T, Pietikainen M and Soriano M presented LBP (Local Binary Pattern) algorithm^[23,24], which analyzes the fix window features with structure methods and extracts global feature with statistic methods. For the RGB color images, convert the color space to the gray space:

$$G(i) = R(i) * 0.3 + G(i) * 0.59 + B(i) * 0.11 \quad (3.4)$$

The LBP operate on 3*3 windows, binary process for the pixels in the window according to the central pixel, the LBP value is obtained by the weighted sum of the pixels in the window.



(a) Three neighborhood examples



(b) Calculation of LBP at the 3*3 window

Figure 3.2 Basic LBP operator

3.2.3 Shape

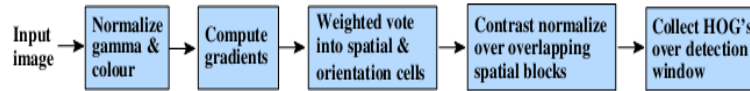


Figure 3.3 HOG operator

Dalal N and Triggs D firstly described the HOG (Histogram of Oriented Gradient) descriptors^[25]. HOG is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization to improved accuracy.

The first step of calculation is the computation of the gradient value, which simply apply the 1-D centered, point discrete derivative mask in one or both of the horizontal and vertical directions. Specifically, this method requires filtering the color or intensity data of the image with the following filter kernels:

$$[-1, 0, 1] \text{ and } [-1, 0, 1]^T$$

The second step of calculation involves creating the cell histograms. Each pixel

within the cell casts a weighted vote for an orientation-based histogram channel based on the values, which is found in the gradient computation. The cells themselves can either be rectangular or radial in shape, and the histogram channels are evenly spread over 0 to 180 degrees or 0 to 360 degrees, depending on whether the gradient is "unsigned" or "signed". The unsigned gradients which are used in conjunction with 9 histogram channels performed best in their human detection experiments. As for the vote weight, pixel contribution can be the gradient magnitude itself.

In order to account for changes in illumination and contrast, the gradient strengths must be locally normalized, which requires grouping the cells together into larger, spatially connected blocks. The HOG descriptor is then the vector of the components of the normalized cell histograms from all of the block regions.

These blocks typically overlap, meaning that each cell contributes more than once to the final descriptor.

The optimal parameters were found to be 3*3 cell blocks of 6*6 pixel cells with 9 histogram channels. Moreover, they found that some minor improvement in performance could be gained by applying a gaussian spatial window within each block before tabulating histogram votes in order to weight pixels around the edge of the blocks less.

Since the HOG descriptor operates on localized cells, the method upholds invariance to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions.

3.2.4 Descriptor combination

The color feature histogram has the advantages of simple, invariant of image rotation, scale and translation; however, the color feature histogram lost the space distribution information. The LBP and HOG features provide texture and shape information of an object within an image, respectively. In addition, they are robust to the noise, so it is beneficial to combine the three features.

In this chapter, the color feature, The LBP texture feature and HOG feature are normally weighted to get the retrieve result. For the query image Q and any image I in the database, the distance $d(Q, I)$ is computed as:

$$d(Q, I) = W_{\text{color}} d_{\text{color}} + W_{\text{texture}} d_{\text{texture}} + W_{\text{shape}} d_{\text{shape}} \quad (3.6)$$

The local feature is extracted from local regions using a scale-invariant detector. The descriptors are coded and pooled to a single vector.

3.3 Experiment

3.3.1 Experiment set

For the commerce retrieval tests, we employed the Microsoft product image set P1 100^[26] (<http://research.microsoft.com/en-us/people/xingx/pi100.aspx>), which contains 10000 product images from Amazon website, each image is adjusted to 100*100 resolution. The samples of PI 100 are illustrated in Figure 3.6.

3.3.2 Result and analysis

The recall and precision are two common methods to scale the performance of the retrieval system.

Table 3.1 The components for recall and precision

	related	No-related
Returned	A	B
No returned	C	D

A: The number of returned positive images.

B: The number of returned negative images.

C: The number of loss-returned positive images.

D: The number of loss-returned negative images.



Figure 3.4 The samples of PI 100 product image database



Figure 3.5 In-class variation samples of PI 100 product image dataset

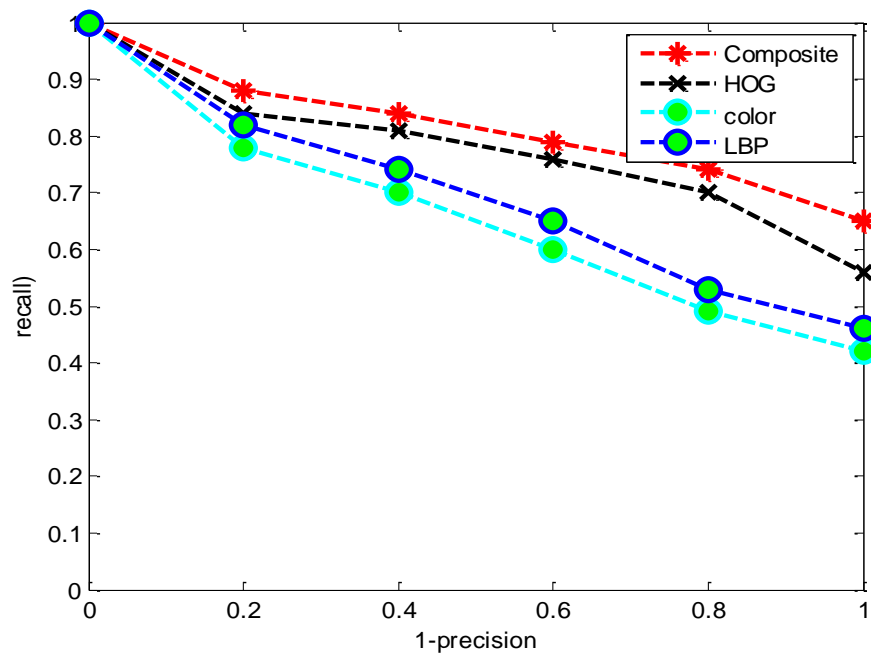


Figure 3.6 The experiment result

Recall is defined as "the number of returned positive images/The number of all the positive images", while "precision" is "the number of returned positive images/The number of all the returned images", that is:

$$\text{Recall} = A / (A + C);$$

$$\text{Precision} = A / (A + B) \quad (3.7)$$

From the result curve, we can conclude that the HOG performs best among the three descriptors, the second is LBP descriptor, the employed color descriptor is the worst of all, which indicates that, compared to the shape and the texture, the color feature is less important to discriminate product categories. The composite feature performs better than any single descriptor. Through the process of cross validation, the weigh parameters for W_{color} , W_{texture} , W_{shape} are set to 0.2, 0.3, 0.5, respectively. However, the experiments also indicate that the performance with uniform combination is very similar to the optimum parameter after validation, and is better than any single descriptor.

3.4 Conclusion

In this chapter, we proposed a combination algorithm of color descriptor, LBP texture descriptor and HOG shape descriptor for product image retrieval. The product image retrieval experiments indicate the combination can boost the performance of retrieval system significantly. Future study will focus on the selection more efficient descriptors set and design more effective combination algorithms to improve the overall performance of commerce retrieval system.

4. Image Retrieval System based on Visual Attention Model

4.1 Introduction

Region of interest (ROI), (also called significant areas), is the key area which can reflect the content and catch people's attention in the image. Existing ROI detection methods are mainly based on artificial designation or image segmentation algorithm. The former is hard to avoid subjectivity, and the latter (segmentation technique) still needs perfection. Consequently, it is hard to implement image representation and retrieval based on ROI, and it becomes a very important issue for CBIR to detect the ROI objectively in the image.

The study on HVS (Human Visual System, HVS)^[34-39] shows that, When people observed image, the brain based on visual attention mechanism can quickly respond to the area of interest^[39] and draw visual attention to the part of the image. Consequently, It's certainly reasonable to establish VAM (Visual Attention Model) through simulation of the human visual system to get the most attractive part and represent it with a gray-scale image, which provides a new idea and method to the field of image semantic understanding^[19].

This chapter analyzes the existing visual attention model features and apply visual attention model to the product image retrieval algorithm. The flow chart of the algorithm is shown in Figure 4.1. Firstly, the saliency map is generate based on visual attention model, on which the saliency part is extracted use dynamic threshold method. Then, the edge information of saliency part is obtained through Canny operator. Image retrieval is implemented by combining the color histogram of saliency map and gradient direction histogram of saliency edge.

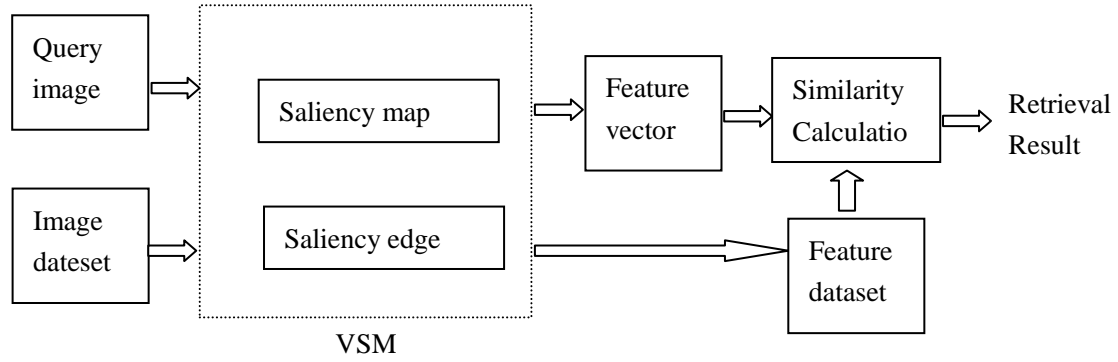


Figure 4.1 The image retrieval system based on VSM

According to the original causes of visual attention, visual attention model can be divided into two categories, One is Bottom-Up model and the other is Top-Down model^[34,35]. Bottom-Up model is the information processing which is driven by low-level visual features and irrelevant to the task. The typical Bottom-Up models include: Itti model, Reissfeld model and Sintoniford model. On the other hand, Top-Down model is driven by high-level visual features and is regarded as one kind of task-related information processing. The representative Top-Down models are: Markov model and Rybak model^[36] and so on. Visual attention model usually represent the visual areas of significance with a gray-scale image, the greater the gray values, the greater the saliency degree. The gray-scale image is usually referred to as saliency map. As Top-Down model requires a large-scale image database to learn, which is computationally intensive and lack of versatility, Bottom-Up model, which generates a saliency map with underlying visual features of a region and its surrounding areas, is much more popular.

According to different ways to get the saliency, such models can be divided into three: local contrast calculation, global contrast calculation, and both. The local contrast calculation methods are being dominated to get the saliency based on visual attention model. Itti^[39] proposed a calculation model to calculate the respective characteristic dimension of the concerns map under a variety of multi-scale characteristics (brightness, color, orientation, etc.) and the center-around difference (Center-Surround) has formed.

4.2 Saliency map algorithm

Saliency map algorithm specific steps are summarized as follows:

1. Color space conversion:

The image to be processed $I(x, y)$ are converted to HSV color space and the CIELab color space, the former is used to calculate a global saliency, the latter is used to calculate the local saliency.

2. Feature sub-maps calculation:

Implement Gaussian pyramid convolution $G(x, y, \sigma)$ with $H(x, y)$, $S(x, y)$, $V(x, y)$, respectively. Three scales layers are set to 3 and 9 feature sub-maps are obtained.

3. Sub-saliency map calculation:

The nine feature sub-maps are converted into the frequency domain, and then obtained through the inverse Fourier transform to get the nine sub-saliency map after redundant signals are removed.

4. Global saliency map calculation.

The Global saliency map is formed by combining the weighted nine sub-saliency map.

4.3 Saliency edge extraction algorithm

Edge is the most basic visual features of the image, which contains a wealth of information about the shape and structure and can be widely used in target detection, image processing, image retrieval, machine vision and many other fields. Edge information extraction has been important issue in the field of pattern recognition. Gray-scale based edge extraction methods including: threshold segmentation, segmentation and edge gradient operator, etc., in which Canny operator is a widely-used popular edge detection operator.

However, not all of the detected edge will help to understand the semantics of the image. In fact, only the edge information on the object is the target of attention, these

marginal information includes saliency object contour and texture on saliency edges. Saliency object contour provides important shape information, but the same contour possibly express radically different semantic. The texture edges on the target structure contain important information, which also helps people to further understand the semantics of the image.

The edge saliency map is defined as blow:

$$S_e(i, j) = S_I(i, j) * E_c(i, j) \quad (6.1)$$

Where $S_I(i, j)$ express the saliency map of the original image $I(i, j)$, $E_c(i, j)$ is the edge image with Canny edge detector, then $S_e(i, j)$ is the edge saliency map, in which the intensity of the pixels express the saliency value. The edge is essentially distributed in saliency contour and region and can well reflect the texture and contour semantic information for subsequent edge feature extraction.

4.4 Image retrieval algorithm combining of saliency object and edge information

Color histogram and edge gradient direction histogram of the saliency region are employed to represent as the image descriptors.

Color histogram is the most common color feature. As the number of colors in the saliency region is less than the color number in the entire image, we use the HSV color space 72 colors non-uniform quantization method, as follows:

$$H = \begin{cases} 0, & h \in (315, 20], \\ 1, & h \in (20, 40], \\ 2, & h \in (40, 75], \\ 3, & h \in (75, 155], \\ 4, & h \in (155, 190], \\ 5, & h \in (190, 270], \\ 6, & h \in (270, 295], \\ 7, & h \in (295, 315], \end{cases} \quad S = \begin{cases} 0, & s \in [0, 0.2] \\ 1, & s \in (0.2, 0.7] \\ 2, & s \in (0.7, 1] \end{cases} \quad V = \begin{cases} 0, & v \in [0, 0.2] \\ 1, & v \in (0.2, 0.7] \\ 2, & v \in (0.7, 1] \end{cases} \quad (6.2)$$

The quantized H, S, V components are combined into a one-dimensional feature vector according to formula:

$$L=9H+3S+V \quad (6.3)$$

Then the range of L is [0, 1, 2... 71]

According to the above quantization scheme, each image is represented as normalization color histogram of saliency region.

$$H_{sa} = (h[c_1], h[c_2], \dots, h[c_k], \dots, h[c_n]) \quad (6.4)$$

Where $h[c_l]$ is the frequency of color c_k , n is the quantization level.

Compute the gradient orientation for any pixel (i,j) on the saliency edge:

$$\theta(i, j) = \arctan\left\{\frac{[f(i, j+1) - f(i, j-1)]}{[f(x+1, j) - f(i-1, y)]}\right\} \quad (6.5)$$

Where $f(i, j)$ is the intensity of the pixel (i,j) . Divide the gradient orientation θ into M parts, calculate the frequency of the pixels assigned to each part with gradient orientations to get the edge orientation histogram:

$$H_{edge} = (h[\theta_1], h[\theta_2], \dots, h[\theta_k], \dots, h[\theta_m]) \quad (6.6)$$

Where $h(\theta_k)$ is the frequency of the edge pixels corresponding to θ_k

Then, any image I can be represented with the color feature and the saliency edge feature, such as:

$$(h[c_1], \dots, h[c_{k_1}], \dots, h[c_{72}], h[\theta_1], \dots, h[\theta_{k_2}], \dots, h[\theta_{36}]) \quad (6.7)$$

Where the parameter n and m are set to 72 and 36, respectively.

The distance of two image I_1 and I_2 is defined as:

$$d(I_1, I_2) = w_{sa} d_{sa} + w_{edge} d_{edge} = w_{sa} \cdot \sum_{k_1=1}^{72} \frac{|h_{I_1}(c_{k_1}) - h_{I_2}(c_{k_1})|}{1 + h_{I_1}(c_{k_1}) + h_{I_2}(c_{k_1})} + w_{edge} \cdot \sum_{k_2=1}^{36} \frac{|h_{I_1}(\theta_{k_2}) - h_{I_2}(\theta_{k_2})|}{1 + h_{I_1}(\theta_{k_2}) + h_{I_2}(\theta_{k_2})} \quad (6.8)$$

Where w_{sa} and w_{edge} are the weights for the features of color and edge, respectively.

$$w_{sa} + w_{edge} = 1 \quad (6.9)$$

4.5 Experiment

4.5.1 Experiment set

For the commerce retrieval tests, we employed the Microsoft product image set P1 100^[6] (<http://research.microsoft.com/en-us/people/xingx/pi100.aspx>), which contains 10000 product images from Amazon website, each image is adjusted to 100*100 resolution. The samples of PI 100 are illustrated as Figure 6.1. All the experiments in this thesis were implemented on the computer with Intel Pentium 2.00GHz CPU, 3GB RAM, Windows XP operation system and MATLAB2010a.

4.5.2 Result and analysis

The recall and precision are two common methods to scale the performance of the retrieval system. Recall is defined as the number of returned positive images /the number of all the positive images, while "precision" is the number of returned positive images/the number of all the returned images. Figure 4.2 illustrates the image retrieval result based on color, color (saliency), edge, edge (saliency), and combination of color saliency and edge saliency.

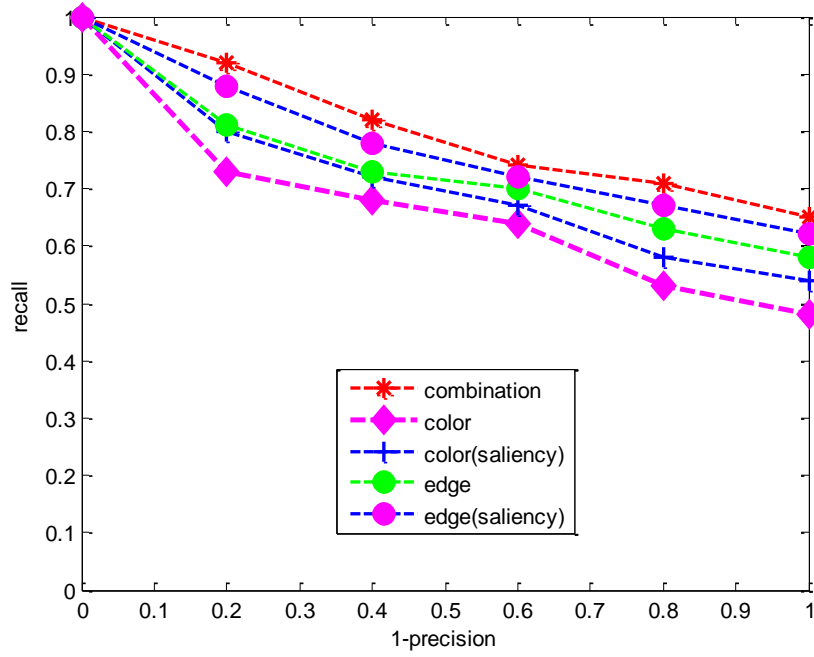


Figure 4.2 The image retrieval result based on color, color (saliency), edge, edge (saliency), and combination of color saliency and edge saliency.

From the result curve, we can conclude that: the combination of color saliency and edge saliency performs best. The edge with saliency performs better than the edge without saliency computation. The employed color descriptor with saliency also performs better than the color descriptor without saliency computation, which indicates that, applying visual attention model is conducive to the product image retrieval algorithm

The combination performs better than any single descriptor. Figure 4.3 illustrates the image retrieval result based on different weights of w_{sa} and w_{edge} . The weight parameters for w_{sa} and w_{edge} are set to [0.5 0.5], [0.4 0.6], [0.3 0.7], [0.6 0.4], [0.7 0.3], respectively. The experiment results indicate the performance of [0.6 0.4] is slightly higher than [0.7 0.3] and [0.5 0.5], the latter two are prior to [0.4 0.6] and [0.3 0.7].

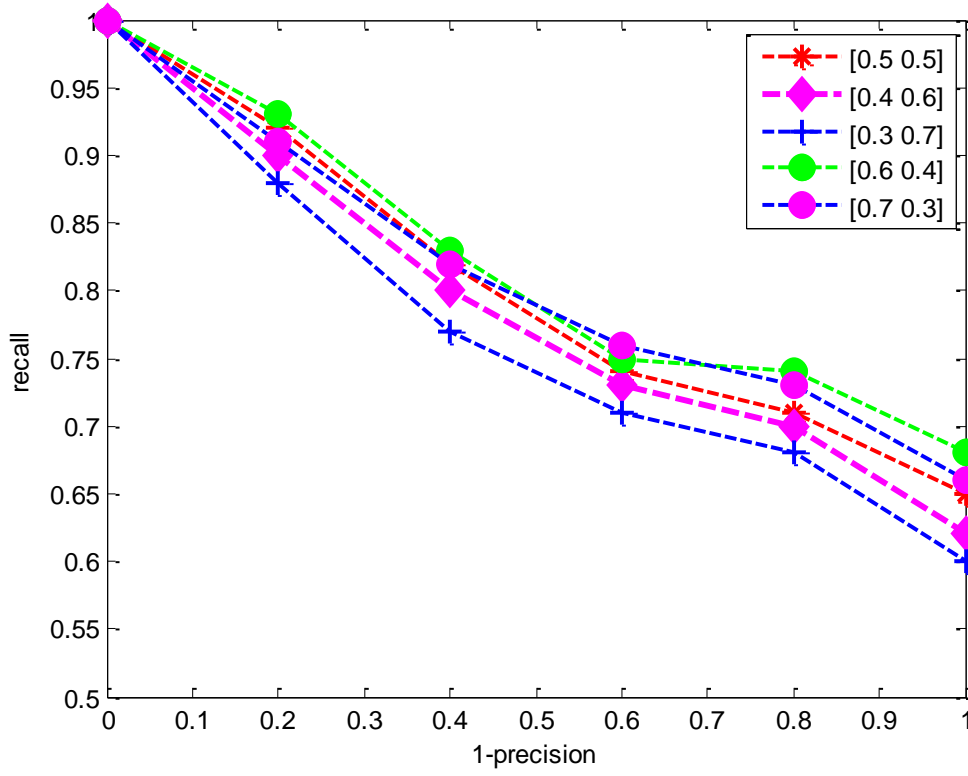


Figure 4.3 The image retrieval result based on different weights

of w_{sa} and w_{edge}

4.6 Conclusion

This chapter presents a product image retrieve algorithm combining with color histogram and edge histogram based on saliency analysis. The proposed algorithm highlights the perception of object areas, inhibits background effects and improves retrieval performance.

5. Product classification with Stacked Auto-Encoder Classifier

5.1 Introduction

In this chapter, we implement product classification depending on the visual characteristics and stacked auto-encoder classifier^[65-68]. A stacked auto-encoder consists of multiple layers of sparse auto-encoders, the outputs of each layer is the inputs of the successive layer, which will describe in detail as below.

An auto-encoder neural network is an unsupervised learning algorithm that applies back-propagation, setting the target values to be equal to the inputs. Figure 5.1 illustrates an auto-encoder neural network.

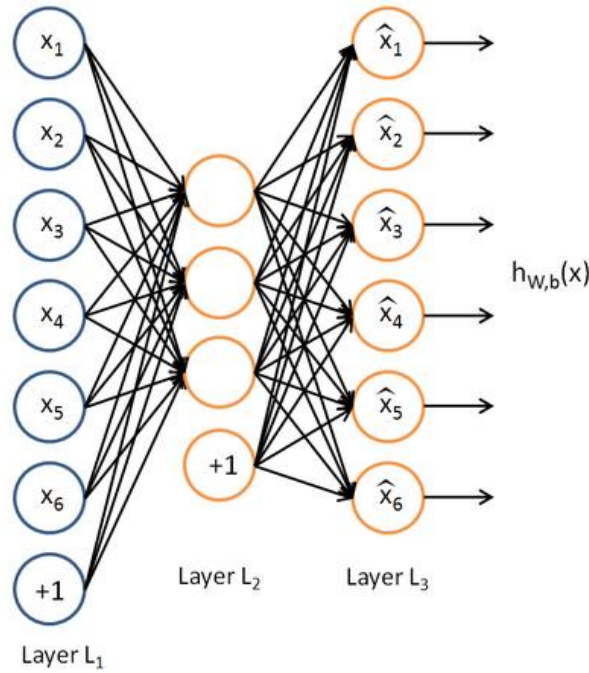


Figure 5.1 Autoencoder neural network

Suppose a set of unlabeled training examples $\{x^{(1)}, x^{(2)}, x^{(3)}, \dots\}$, $x^{(i)} \in R^n$. The auto-encoder neural network learns an approximation to the input x , i.e.

$$h_{w,b}(x) \approx x$$

The parameters as well as the code of the input are obtained by minimizing the reconstruction error.

If the features are correlated, then this algorithm will be able to discover some of those correlations and learn a low-dimensional representation.

We employ sigmoid as activation function, which 0 indicate a neuron as being inactive when it outputs values close to -1.

If we impose a sparsity constraint on the hidden units (Figure 5.2), i.e.

$$\text{-input: } h = W^T X$$

$$\text{-loss } L(X;W) \|Wh - X\|^2 + \lambda \sum_i |h_j| \quad (5.1)$$

then the auto-encoder will still discover interesting structure in the data, even if the number of hidden units is large.

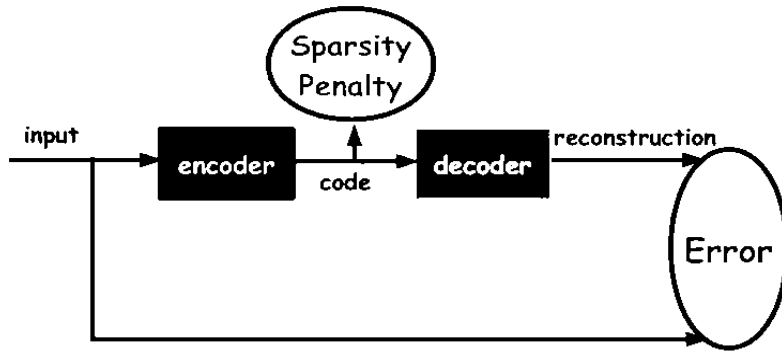


Figure 5.2 Sparsity penalty on the auto-encoder neural network

5.2 Stacked auto-encoder

A stacked auto-encoder consists of multiple layers of sparse auto-encoders, the outputs of each layer is the inputs of the successive layer.

Greedy layer-wise training is employed to get parameters for a stacked auto-encoder. The outputs of each layer are the inputs of the subsequent layer. Train each layer on this input vector to obtain parameters W, b . Transform the input into a vector, which consists of activation of the hidden units.

For k_{th} auto-encoder, Let $W^{(k,1)}, W^{(k,2)}, b^{(k,1)}, b^{(k,2)}$ denote the parameters $W^{(1)}, W^{(2)}, b^{(1)}, b^{(2)}$, respectively, the encoding step:

$$a^{(l)} = f(z^{(l)}) \quad (5.2)$$

$$z^{(l+1)} = W^{(l,1)} a^{(l)} + b^{(l,1)} \quad (5.3)$$

The decoding step is in the reverse order:

$$a^{(l+n)} = f(z^{(l+n)}) \quad (5.4)$$

$$z^{(l+n+1)} = W^{(n-l,2)} a^{(n+l)} + b^{(n-l,2)} \quad (5.5)$$

After this training phase, fine-tuning is employed to tune the parameters of all layers by back propagation at the same time, in which the gradients from the classification error is back propagated into the encoding layers.

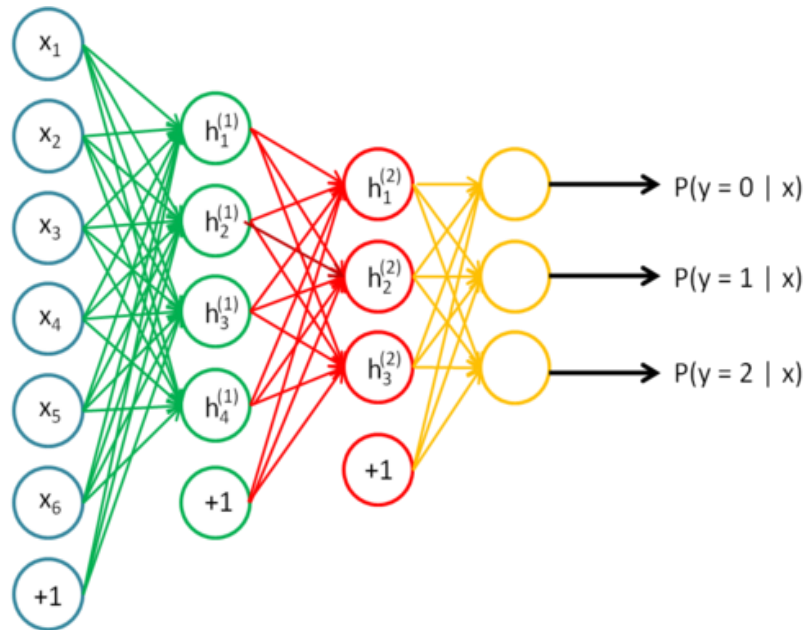
For the purpose of classification, a classifier is added after the last encode, and is trained with standard supervised algorithm (decreasing gradient algorithm).

As higher-order features, the learned features $a^{(n)}$ gives a representation of the input. which can be used for classification. The cooperation of the learned features and original features can boost the accuracy greatly.

Figure 5.3 shows a stacked auto-encoder with 2 hidden layers and a classifier layer. The processes are as follows:

- (1) Learn primary features $h^{(1)(k)}$ on the raw input $x^{(k)}$.
- (2) Learn secondary features $h^{(2)(k)}$ using the primary feature activations $h^{(1)(k)}$
- (3) Train a classifier with these secondary features $h^{(2)(k)}$ and classification labels.

(4) Combine all three layers together, including a stacked auto-encoder with 2 hidden layers and a classifier layer.



Input Features 1 Features 2 Soft-max classifier

Figure 5.3 A stacked auto-encoder with 2 hidden layers and a classifier layer

5.3 Experiment

All the experiments are implemented on the computer with Intel Core2 CPU 2.1GHz, 2GB RAM, running Windows XP operating system and MATLAB2010b software. The selected categories involve the most common product: Hiking backpack, Box glove, Boot, Cap, Jacket, Baby shoe. The samples of these six categories are illustrated in Table 4.1. In the experiments, we use a stacked auto-encoder for product classification. We employ two auto-encoder layers and perform layer-wise training and fine-tuning through back-propagation both layers. First we forward propagate the training set through the first auto-encoder. Next, continue to forward propagate the L1 features through the second auto-encoder to obtain the L2 hidden unit activations. These activations are then used to train the soft-max classifier. To implement fine

tuning, all three layers are considered as a single model. 10 images per category are used for fine-tuning^[68,69].

Table 5.2 shows the classification results (confusion matrix) before fine-tuning. The accuracies of box glove, boot, cap, jacket, baby shoe, hiking backpack are about 90%, 92%, 95%, 100%, 94%, 88%, respectively. Table 5.3 gives the experiment data after fine-tuning. The average accuracies are improved to 100%, 95%, 98%, 100%, 97% and 92% for the six categories, respectively, which indicates the fine-tuning plays an important role in the classification process.

Table 5.1 Product samples

Box glove						
Boot						
Cap						
Jacket						
Baby shoe						
Hiking backpack						

Table 5.2 Confusion matrix without fine-tuning

	Hiking backpack	Box glove	Boot	Cap	Jacket	Baby shoe
Hiking backpack	90%	0	5	0	5%	
Box glove	0	92%	2%	0	0	6%
Boot	5	0	95%	0	0	0
Cap	0	0	0	100%	0	0
Jacket	6%	0	0	0	94%	0
Baby shoe	0	5%	7%	10%	0	88%

Table 5.3 Confusion matrix with fine-tuning

	Hiking backpack	Box glove	Boot	Cap	Jacket	Baby shoe
Hiking backpack	100%	0	0	0	0	0
Box glove	0	95%	0	0	0	5%
Boot	2	0	98%	0	0	0
Cap	0	0	0	100%	0	0
Jacket	3%	0	0	0	97%	0
Baby shoe	0	2%	3%	3%	0	92%

5.4 Conclusion

In this chapter, we implement product classification depending on the visual characteristics and stacked auto-encoder classifier. A stacked auto-encoder consists of multiple layers of sparse auto-encoders, the outputs of each layer is the inputs of the successive layer. An auto-encoder attempts to learn appropriate features to represent its raw input, and higher layers tend to learn higher-order features, which construct a hierarchical grouping of the input. The samples in our experiments are generally restricted to six common categories. Future research should focus on more complicated learning algorithms to implement sub-category product classification.

6. Product Classification based on SVM and PHOG Descriptor

6.1 Introduction

For the efficiency of product retrieve, it is of necessity to classify the numerous product into some categories automatically, such as clothes, household appliances, office equipment. Each category can be classified into many sub-categories. For example, shoes, bags, T-shirts are different types of textile and clothing products. In this thesis, SVMs (Support Vector Machines) and PHOG (Pyramid of Histograms of Orientation Gradients) descriptor are employed to implement the product classification. As supervised learning models^[27], Support Vector Machines (SVMs) exhibits standout learning capability. Support vector machines can efficiently perform a non-linear classification using the kernel trick, implicitly map their inputs into high-dimensional feature spaces. PHOG is an excellent image global shape descriptor, which consists of a histogram of orientation gradients over each image subregion at each resolution level^[30]. The process will be described in detail as blow.

6.2 Support vector machine

The classification can be seen as one kind of machine learning task. In the recent last two decades, Support Vector Machine (SVMs) have become an popular supervised tool for machine learning community.

We have N training points $\{x_i, y_i\}$, $i=1, \dots, m$. where each input $x \in R^d$ is in one of two classes label $y = -1$ or $+1$. Assuming the data is linearly separable. The hyperplane can be described as:

$$w \cdot x + b = 0$$

Where w is normal to the hyperplane. $\frac{b}{\|w\|}$ is the perpendicular distance from the hyperplane to the origin. The so-called support vectors are the examples closest to the hyperplane.

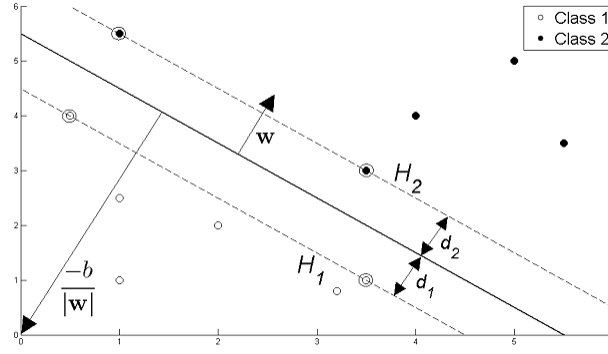


Figure 6.1 Hyperplane through two classes

As Figure 6.1 illustrates, support vector machine boils down to finding the appropriate w and b so that:

$$\begin{aligned} x_i \cdot w + b &\geq +1 \quad \text{for } y_i = +1 \\ x_i \cdot w + b &\leq -1 \quad \text{for } y_i = -1 \end{aligned} \quad (6.1)$$

The two can be combination into one equation:

$$y_i \cdot (x_i \cdot w + b) - 1 \geq 0 \quad (6.2)$$

The distance from H_1 to the hyperplane d_1 is required to be equal to d_2 , the distance from H_2 to the hyperplane. d_1 or d_2 is named the margin of SVM, which should be maximized. The maximizing is equivalent to solving:

$$\begin{aligned} \min \|w\| &\Rightarrow \min \frac{1}{2} \|w\|^2 \\ y_i \cdot (x_i \cdot w + b) - 1 &\geq 0 \end{aligned} \quad (6.3)$$

The question boils down to performing Quadratic Programming (QP) optimization.

Employing the Lagrange multipliers α to cater for the constrains:

$$\begin{aligned}
L_p &= \frac{1}{2} \|w\|^2 - \alpha[y_i \cdot (x_i \cdot w + b) - 1] \\
&= \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \alpha[y_i \cdot (x_i \cdot w + b) - 1] \\
&= \frac{1}{2} \|w\|^2 - \sum_{i=1}^m \alpha[y_i \cdot (x_i \cdot w + b)] + \sum_{i=1}^m \alpha_i
\end{aligned} \tag{6.4}$$

Differentiate L_p with respect to w and b and setting the derivatives to zero:

$$\begin{aligned}
\frac{\partial L_p}{\partial w} &= 0 \Rightarrow w = \sum_{i=1}^m \alpha_i y_i x_i \\
\frac{\partial L_p}{\partial b} &= 0 \Rightarrow \sum_{i=1}^m \alpha_i y_i = 0
\end{aligned} \tag{6.5}$$

$$\begin{aligned}
L_D &= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i x_j \\
&= \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i H \alpha_j \\
&= \sum_{i=1}^m \alpha_i - \frac{1}{2} \alpha^T H \alpha
\end{aligned} \tag{6.6}$$

Where

$$H = y_i y_j x_i x_j \tag{6.7}$$

L_D is the dual form of the primary L_p . The dual form requires only the dot product of each input vector x_i .

The maximizing L_D needs to find:

$$\max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \alpha^T H \alpha \tag{6.8}$$

Subject to: $\alpha_i \geq 0$

$$\sum_{i=1}^m \alpha_i y_i = 0 \tag{6.9}$$

The convex quadratic optimization can be solved with QP solver. Then

$$w = \sum_{i=1}^m \alpha_i y_i x_i \quad (6.10)$$

The support vector x_s have the form:

$$y_s \cdot (x_s \cdot w + b) = 1$$

$$y_s \cdot \left(\sum_{l \in S} \alpha_l y_l x_l \cdot x_s + b \right) = 1 \quad (6.11)$$

Where s denotes the set of the support vectors.

$$y_s^2 \cdot \left(\sum_{l \in S} \alpha_l y_l x_l \cdot x_s + b \right) = y_s \quad (6.12)$$

$$\text{As } y_s^2 = 1$$

$$b = y_s - \sum_{l \in S} \alpha_l y_l x_l \cdot x_s \quad (6.13)$$

So each new sample x' is classified by

$$y' = \text{sgn}(x' \cdot w + b) \quad (6.14)$$

Binary Classification for Data that is not Fully Linearly Separable

For the data set which is not fully linearly separable, the constrain can be relaxed to allow for some misclassified samples by introducing a positive variable

$$\xi_i, i = 1, 2, \dots, m.$$

$$\begin{aligned} x_i \cdot w + b &\geq +1 - \xi_i \quad \text{for } y_i = +1 \\ x_i \cdot w + b &\leq +1 + \xi_i \quad \text{for } y_i = -1 \end{aligned} \quad (6.15)$$

The two can be combination into one equation:

$$y_i \cdot (x_i \cdot w + b) - 1 + \xi_i \geq 0 \quad (6.16)$$

The generalized optimal hyperplane is determined by the minimizing the function:

$$\frac{1}{2}\|w\|^2 + C \sum_i \xi_i \quad (6.17)$$

Subject to:

$$y_i \cdot (x_i \cdot w + b) - 1 + \xi_i \geq 0 \quad (6.18)$$

Reformulating as a Lagrangian:

$$\Phi(w, b, \alpha, \beta, \xi) = \frac{1}{2}\|w\|^2 + C \sum_i \xi_i - \sum_{i=1}^m \alpha_i (y_i \cdot (x_i \cdot w + b) - 1 + \xi_i) - \sum_{j=1}^m \beta_j \xi_j \quad (6.19)$$

The parameter C controls the trade-off between the slack penalty and the margin. α and β the Lagrange multipliers. The Lagrangian has to be minimized with respect to w, b , and maximized with respect to α, β . The dual problem is given by:

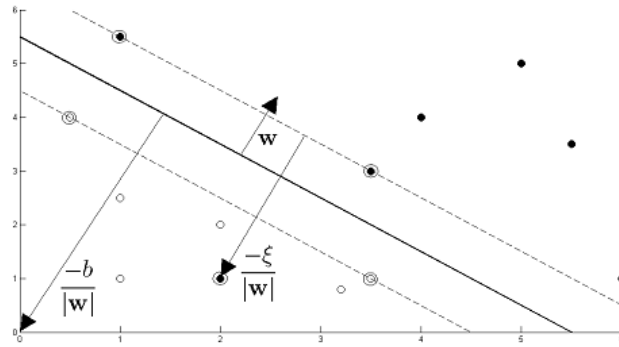


Figure 6.2 Non-linearly separable classes

$$\max_{\alpha, \beta} (\min_{w, b, \xi} \Phi(w, b, \alpha, \beta, \xi)) \quad (6.20)$$

The minimum with respect to w, b, ξ is given by:

$$\begin{aligned}
\frac{\partial \Phi}{\partial b} = 0 &\Rightarrow \sum_{i=1}^m \alpha_i y_i = 0; \\
\frac{\partial \Phi}{\partial w} = 0 &\Rightarrow w = \sum_{i=1}^m \alpha_i x_i y_i; \\
\frac{\partial \Phi}{\partial \xi} = 0 &\Rightarrow \alpha_i + \beta_i = C.
\end{aligned} \tag{6.21}$$

Hence:

$$\max_{\alpha, \beta} (\min_{w, b, \xi} \Phi(w, b, \alpha, \beta, \xi)) = \max_{\alpha} -\frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j x_i x_j + \sum_{i=1}^m \alpha_i \tag{6.22}$$

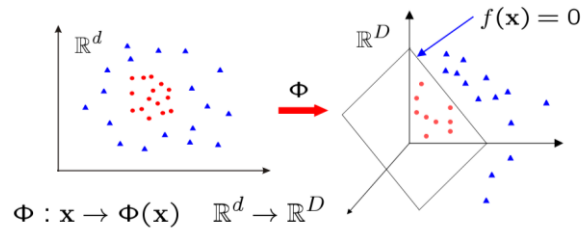
With constrains:

$$\begin{aligned}
\sum_{i=1}^m \alpha_i y_i &= 0, \\
0 \leq \alpha_i &\leq C, \quad i = 1, 2, \dots, m;
\end{aligned} \tag{6.23}$$

SVM classifier in a transformed feature space

Data may be linearly separable in the high dimensional space, but not linearly separable in the original space. Employing kernel trick, classifiers can be learned for high dimensional space, without actually having to map the data into the high dimensional space.

The support vector machine maps the input vectors into a high-dimension space, in which a max margin support hyper plane is set up to classify the samples. Figure 6.3 illustrates the process.



(a) Original space (b) feature space (RBF mapping)

Figure 6.3 Separable classification with RBF kernel functions in the original space and feature

The idea of the kernel function is to enable operations to be performed in the input space rather than the potentially high dimensional feature space. Hence the inner product does not need to be evaluated in the feature space.

A kernel is a function k that for the data $x, z \in X$, satisfies

$$k(x, z) = \langle \phi(x), \phi(y) \rangle \quad (6.24)$$

Where ϕ is a mapping from X to a feature space F [61]

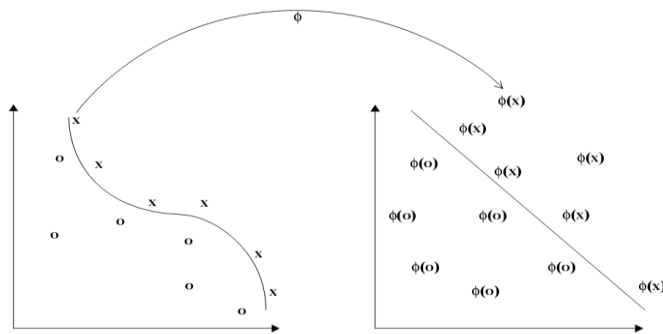


Figure 6.4 ϕ is a mapping from X to a feature space

Gram matrix:

Given a set of data $S = \{x_1, x_2, \dots, x_m\}$, the gram matrix is defined as the $m \times m$ matrix

G whose entries satisfies $G_{ij} = \langle \phi(x_i), \phi(x_j) \rangle = k(x_i, x_j)$

The matrix is also called kernel matrix.

Table 6.1 The kernel matrix K_l

	1	2	...	m
1	$k_l(x_1, x_1)$	$k_l(x_1, x_2)$...	$k_l(x_1, x_m)$
2	$k_l(x_2, x_1)$	$k_l(x_2, x_2)$...	$k_l(x_2, x_m)$
...
m	$k_l(x_m, x_1)$	$k_l(x_m, x_2)$...	$k_l(x_m, x_m)$

Figure 6.5 The stages of kernel methods

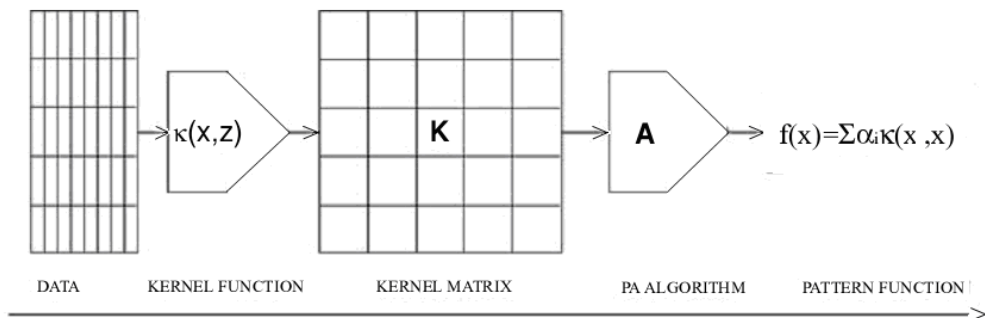


Figure 6.5 shows the stages involved in the kernel analysis. The data is processed with a kernel to create a gram matrix, and then a pattern analysis algorithm is employed to produce a pattern function, which is used to process unseen samples.

Given some arbitrary function k , if it corresponds to a scalar product in some feature space, the valid kernel is called Mercer kernel, which satisfies:

(1) Symmetric.

$$k(x_i, x_j) = k(x_j, x_i);$$

(2) Positive semi-definite for all training sets S . i.e. its kernel matrices are

definitely positive semi-definite.

$$\alpha^t K \alpha \geq 0,$$

Where K is $m \times m$ gram matrix. $\alpha \in R^d$

For the kernel-based SVM classifier, the dual problem is given by:

$$\begin{aligned} \max_{\alpha, \beta} (\min_{w, b, \xi} \Phi(w, b, \alpha, \beta, \xi)) &= \max_{\alpha} -\frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \phi(x_i) \phi(x_j) + \sum_{i=1}^m \alpha_i \\ &= \max_{\alpha} -\frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j k(x_i, x_j) + \sum_{i=1}^m \alpha_i \end{aligned} \quad (6.25)$$

The optimal hyperplane:

$$\begin{aligned} f(x) &= \sum_{i=1}^m \alpha_i y_i \phi(x_i) \phi(x) + b \\ &= \sum_{i=1}^m \alpha_i y_i k(x_i, x) + b \end{aligned} \quad (6.26)$$

Subject to

$$y_i f(x) \geq 1 - \xi_i \text{ for } i = 1, 2, \dots, m \quad (6.27)$$

The general processes of C-SVM^[27] are explained as below.

(1) Assuming the training dataset D is given as:

$$D = \{(x_i, y_i) \mid x_i \in R^d, y_i \in \{-1, 1\}\}, i = 1, 2, \dots, m \quad (6.28)$$

Where y_i is the label of training sample x_i , m is the total number of training samples.

(2) In the training phrase, select appropriate kernel function $k(x_i, x_j)$ and penal parameter $C > 0$.

(3) Construct and solve the convex programming problem:

$$\begin{aligned}
\min_{\alpha} \quad & -\sum_{i=1}^m \alpha_i + \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j k(x_i, x_j), \\
s.t. \quad & \sum_{i=1}^m \alpha_i y_i = 0, \\
& 0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, m;
\end{aligned} \tag{6.29}$$

Where $k(x_i, x_j) = \phi(x_i)^T \phi(x_j)$. The solution $\alpha^* = (\alpha_1^*, \dots, \alpha_m^*)^T$ are always sparse, i.e. there are small number of non-zero coefficients, the corresponding training samples are defined as the support vector machines.

(4) Choose the α_j^* with the value of $(0, C)$, calculate:

$$b^* = y_j - \sum_{i=1}^m y_i \alpha_i^* k(x_i, x_j); \tag{6.30}$$

(5) For the test sample x , the final classification decision function is

$$f(x) = \text{sgn}\left(\sum_{i=1}^m \alpha_i^* y_i k(x_i, x) + b^*\right) \tag{6.31}$$

Below are some common kernel functions for image classification in practice.

(1). Linear Kernel

If $\Phi(x) = x$, we get the linear kernel:

$$k_{\text{linear}}(h, h') = h^T h' \tag{6.32}$$

(2). Radial basis function kernel

Radial basis function kernels are built based on Euclidean distance, with one adjustable parameter, i.e. σ .

$$k_{\text{Gaussian}}(h, h') = \exp(-\|h - h'\|^2 / 2\sigma^2) \tag{6.33}$$

Small value of σ allow kernel classifiers to fit any labels, corresponding to the large value of d in the polynomial kernel. While large value of σ gradually reduce the kernel to a constant function, which make it impossible to learn any meaningful classifier.

(3). Histogram intersection kernel^[28]

$$k_{HI}(h, h') = \sum_i \min(h_i, h'_i) \quad (6.34)$$

(4). Chi-Square Kernel^[29]

The Chi-Square kernel comes from the Chi-Square distribution.

$$k(h, h') = \exp(-\rho \chi^2(h, h')) \quad (6.35)$$

Where:

$$\chi^2(h, h') = \sum_i \frac{(h_i - h'_i)^2}{h_i + h'_i}$$

6.3 Image descriptor

In this part, an image is represented with its local shape (distribution over edge orientations within a region) and its spatial layout (tiling the image into regions at multiple resolutions). The PHOG (Pyramid of Histograms of Orientation Gradients) descriptor^[30] consists of a histogram of orientation gradients over each image subregion at each resolution level.

6.3.1 Local shape

Local shape can be described by a histogram of edge orientations (quantized into K bins) within an image subregion. The edge orientations are quantized into K bins, each of which represents the number of edges which have a certain angular range orientations. The contribution of each edge is weighted by its magnitude.

6.3.2 Spatial layout

The PHOG image descriptor is a concatenation of all the HOG^[28] vectors, each of which is computed for each grid cell at each pyramid. Consequently, level 0 is represented by K-bin histogram, level 1 is represented by a 4*K-bin histogram, etc, and the final PHOG descriptor of the entire image is a vector with dimensionality $K \sum_{l \in L} 4^l$. For example, for levels up to L= 1 and K= 30 bins it will be a 150-vector. To prevent overfitting, we limit the number of levels to L= 3 in our implementation. More to the point, the PHOG is normalized to sum to unity to ensure that images with more edges are not weighted more strongly than others. The diagrams shown in Figure 6.6 depict the PHOG descriptor at each level^[62].

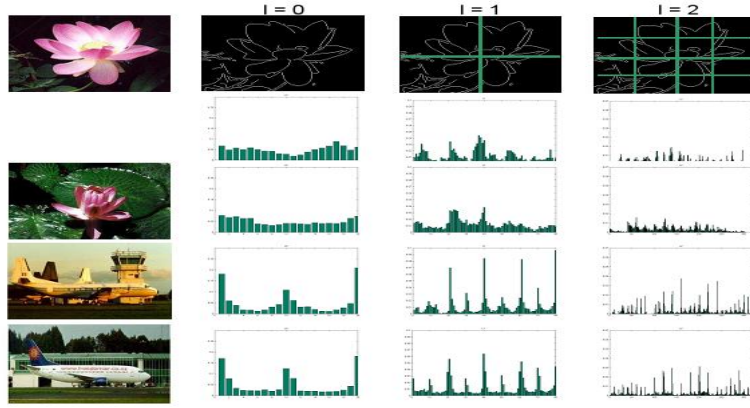


Figure 6.6 PHOG descriptor at each level (0~2)

6.4 Experiment and Result

6.4.1 Experiment set

I adopt PHOG descriptors combined with SVM to implement product–image classification. All the experiments in this thesis were implemented on the computer with Intel Pentium 2.00GHz CPU, 3GB RAM, Windows XP operation system and MATLAB2010a. The product images of ten categories were mainly collected from MSN shopping web site. Figure 6.3 illustrates some samples of the product images.



Box glove						
Boot						
Cap						
Jacket						
Baby shoe						
Neclace						
Briefcase						
Curtain						
Hiking backpack						
Helmet						

Figure 6.3 Product images of ten categories

I adopted LIBSVM^[27] package as SVM classifier implementation. and use the kernels defined in section 2. Multi-class classification is done with one-versus-one method. For the one-versus-one approach, classification is done by a max-wins voting strategy, in which every classifier assigns the sample to one of the two classes, then the vote for the assigned class is increased by one vote, and finally the class with the most votes determines the instance classification^[32].

For the shape implementation, we use Canny edge detector to extract edge contours and use a 5*5 Sobel mask to compute the orientation gradients (ranging from 0 to 180). The HOG descriptor is discretized into K (ranging between 5 and 50) orientation bins. The vote from each contour point depends on its gradient magnitude. The pyramid level is set to 3.

Classification accuracy is the common assessment of classification performances. In order to get objective classification results, all the experiments employed cross-validation methods, in which all the samples are divided into two parts, 70% for the training and 30% for the test. All the experiments are repeated N times and the average accuracy is calculated as the final result. Average classification accuracy rate use the following formula to calculate:

$$Average \ accuracy = \frac{1}{N} \sum_{i=1}^N \frac{R_i}{S_i} \quad (6.9)$$

Where N indicates the number of experiments, R_i is the the number of images which are correctly classified in the i_{th} experiment, S_i is the the total number of image in the i_{th} experiment.

6.4.2 Experimental results and analysis

Table 6.2 illustrated the classification accuracy variation with the number of training images per category and kernel functions. The experimental results indicate:

(1) The average accuracies increase with the training samples. However, the accuracy is moving towards stabilization as the training number reaches 30.

(2) Chi-square kernel performs best, followed by histogram intersection kernel, RBF kernel, linear kernel perform worst.

Table 6.2 Average classification accuracy variation with training samples and kernel functions (L=3) (%)

	5	15	30	50
Linear kernel (C=1500)	76.5	80.0	81.8	82.0
RBF kernel (C=1500, g=0.07)	82.4	88.3	91.3	91.5
Histogram intersection kernel	85.2	90.6	96.1	96.3
Chi-square kernel	85.5	90.8	96.2	96.6

6.5 Conclusion

In this thesis, we adopt SVM classifier combined with PHOG descriptors to implement product-image classification. Experimental results showed the effectiveness of the proposed algorithm. As an effective descriptor, PHOG can flexibly represent the spatial layout of local image shape. However, PHOG is not a kind of universal descriptor, and the product categories we test are far from the practical applications. It is a prosperous direction to combine PHOG and other complimentary descriptor (such as appearance descriptor, color descriptor and texture) with (multiple effective kernel^[7] function based) SVM classifier to implement large-category product image classification.

7. Conclusions

So far, most of the shopping sites still rely on the establishment of keyword indexing. However, "one picture worth thousands of words", the text-based methods can not retrieve accurately and unable to describe the visual content well, which cannot meet the needs of users.

The CBIR is build upon computer vision and image comprehension theory, and it's a combination of artificial intelligence, object oriented technology, cognitive psychology, data base and other multi-disciplinary knowledge. For the efficiency of product retrieve, it is of necessity to classify the numerous products into some categories automatically. Automatic product image classification can effectively improve the overall effectiveness of the E-commerce market, Consequently, it is a critical requirement of E-commerce intelligent

However, some adverse effects, such as the within-class variation, obscure, pose variation, and background interference, are hard to overcome. Therefore it is still a challenging problem in the field of computer vision. This dissertation focuses on the content-based product image retrieval and classification.

In this thesis, we proposed three algorithms to implement product retrieval and classification.

(1) Product image retrieval based on feature combination. The features include color descriptor, LBP texture descriptor and HOG shape descriptor. The color feature histogram cannot describe the space distribution information, but it has the advantages of simple, invariant of image rotation, scale and translation; HOG is computed on a dense grid of uniformly spaced cells and uses overlapping local contrast normalization. LBP analyzes the fix window features with structure methods and extract global feature with statistic methods. The LBP and HOG features provide texture and shape information of an object within an image, respectively. In addition, they are robust to noise, so it is beneficial to combine the three features. The product image retrieval experiments indicate the combination can boost the performance of

retrieval system significantly. Future study will focus on the selection more efficient descriptors set and design more effective combination algorithms to improve the overall performance of commerce retrieval system.

(2) The study on HVS shows that the brain based on visual attention mechanism can quickly respond to the area of interest and draw visual attention to the part of the image. It's certainly reasonable to establish VAM (Visual Attention Model) through simulation of the human visual system to get the most attractive part and represent it with a grayscale image, which provides a new idea and method to the study of image semantic understanding. The salient region detection method produces saliency value of each region in an image, following the visual saliency laws and utilizing a variety of basic image features comprehensively, such as color, intensity, local orientation, etc. This thesis analyzes the existing visual attention model features and apply visual attention model to the product image retrieval algorithms. Firstly, the saliency map is generated based on visual attention model, on which the saliency part is extracted with dynamic threshold method. Then, the edge information of saliency part is obtained through Canny operator. Image retrieval is implemented through combining the color histogram of saliency map and gradient direction histogram of saliency edges. The proposed algorithm highlights the perception of object areas, inhibits background effects and improves retrieval performance.

(3) The aim of content-based image classification is to implement semantic classification automatically based on the visual features. Each category can be classified into many sub-categories. In this thesis, we implement product classification depending on the visual characteristics and stacked auto-encoder classifier. A stacked auto-encoder consists of multiple layers of sparse auto-encoders, the outputs of each layer is the inputs of the successive layer. Experimental results showed the average accuracies are improved to 100%, 95%, 98%, 100%, 97% and 92% for the six categories, respectively, which indicates the fine-tuning plays an important role in the classification process. In addition, we adopt SVM classifier combined with PHOG descriptors to implement product-image classification. Support

vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, which can efficiently perform a non-linear classification using the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces. PHOG is an excellent image global shape descriptor, which consists of a histogram of orientation gradients over each image sub-region at each resolution level. Experimental results showed the effectiveness of the proposed algorithm. It is a prosperous direction to combine PHOG and other complimentary descriptors (such as appearance descriptor, color descriptor, texture) with (multiple effective kernel^[7] function based) SVM classifier to implement large-category product image classification.

Acknowledgements

My deepest gratitude goes first and foremost to Professor Zheng Tang, my supervisor, for his constant encouragement and guidance. He has walked me through all the stages of the writing of this thesis. Without his consistent and illuminating instruction, this thesis could not have reached its present form.

Secondly, I would like to thank Professor Masaaki Shimizu and professor Koji Kikushima who have given me much valuable advice and guidance in the process of the paper writing.

Last my thanks would go to my beloved family for their loving considerations and great confidence in me all through these years. I also owe my sincere gratitude to my friends and my fellow classmates who gave me their help and time in listening to me and helping me work out my problems during the difficult course of the thesis.

References

- [1] Nblack W, Barber R, et al: The QBIC project: querying images by content using color, texture and shape. In Proc. of SPIE: Storage and Retrieval for Image and Video Database Sanjoes, pp. 173-181. (1994).
- [2] Hirata K and Kato T: Query by Visual Example-Content based image Retrieval. Proceedings of the 3rd International Conference on Extending Database Technology: Advances in Database, Vienna, Austria, pp. 23-27. (1992).
- [3] Pentland A, Picard R W. Sclaroff S. Photobook: Content-Based manipulation of image databases [J]. International Journal of Computer Vision. Vol. 18, No. 3, pp.233-256. (1996).
- [4] Arnold W M, Marcel W, Simone S. Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 22, No. 12, pp. 1-22. (2000).
- [5] W. Niblack: The QBIC project: querying images by color, texture and shape. IBM Research Report, pp.5-16. (1993).
- [6] Bach J R, Fuller C, Gupta A: The Visage image search engine: an open framework for image management, Proc. SPIE Storage and Retrieval for Image and Video Database, pp. 12-33. (1996).
- [7] Kherfi M L, Ziou D, Bernardi A: Image retrieval from the World Wide Web: Issues, techniques, and systems, ACM Comput. Surv. Vol. 36, No. 1, pp. 35-67. (2004).
- [8] <http://www.like.com>.
- [9] <http://www.iresearch.cn/>.
- [10] <http://www.taotaosou.com>.

- [11] Fang Y, Yamada K, Ninomiya Y, et al: A Shape-independent Method for Pedestrian Detection with Far-infrared Images. *IEEE Transactions on Vehicular Technology*, Vol. 53, No. 6, pp. 1679-1697. (2004).
- [12] Jiang Y G, Ngo C W, Yang J: Towards optimal bag-of-features for object categorization and semantic video retrieval. *Proceedings of the 6th ACM international conference on Image and video retrieval*. Amsterdam, The Netherlands, pp. 494-501. (2007).
- [13] Yue J, Li Z B, Liu L, et al: Content-based image retrieval using color and texture fused features. *Mathematical and Computer Modelling*. Vol. 54, pp. 1121-1127. (2011).
- [14] Yang N C, Chang W H: A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval [J]. *Journal of Visual Communication & Image Representation*. Vol. 19, No. 2, pp. 92-105. (2008).
- [15] Wang X Y, Yu Y J, Yang H Y: An effective image retrieval scheme using color, texture and shape features [J]. *Computer Standards & Interfaces*, Vol. 33, pp. 59-68. (2011).
- [16] Middleton L: Edge detection in a hexagonal-image processing framework, *Image and Vision Computing*, Vol. 19, pp. 1071-1081. (2001).
- [17] Chun Y D, Kim N C, Jang I H: Content-based image retrieval using multiresolution color and texture features, *IEEE Transactions on Multimedia*, Vol. 10, No.6, pp. 1073-1086. (2008).
- [18] Wang H, Dai F, Zhang L, Lu S X: An image retrieval method based on texture features of object region [C]. *International Conference on Electronics and Optoelectronics*, Vol. 4, pp. 83-86. (2011).
- [19] Bamidele A, Stentiford F, Morphet J: An attention-based approach to content-based image retrieval. *BT Technology Journal*. Vol. 22, No.7, pp. 151-160. (2004).

- [20] Li H, Wang X. Tang J, etc al: iSearch: towards precise retrieval of item image. Proceedings of the Third International Conference on Internet Multimedia Computing and Service. Chengdu, China, pp. 5-8. (2011).
- [21] Wengert C, Douze M, Jégou H: Bag-of-colors for improved image search. Proceedings of the 19th ACM Multimedia. pp. 1437-1440. (2011).
- [22] http://en.wikipedia.org/wiki/HSL_and_HSV.
- [23] Maenpaa T, Ojala T, Pietikainen M, Soriano M: Robust texture classification by subsets of local binary patterns, 15th International Conference on Pattern Recognition. Vol 3, pp. 935-938. (2000).
- [24] Ahonen T, Hadid A, Pietikainen M: Face description with local binary patterns: Application to face recognition, IEEE Transaction on Pattern Analysis and Machine Intelligence. Vol. 28, No.12, pp. 2037-2041. (2006).
- [25] Dalal N and Triggs D: Histograms of oriented gradients for human detection. In: Anon. Proceedings of Conference on Computer Vision and Pattern Recognition. San Diego, California, USA. New York, pp. 556-893. (2005).
- [26] Xing Xie, Lie Lu, Menglei Jia, Hua Li, Frank Seide, Wei-Ying Ma, Mobile Search with Multimodal Queries, Proceedings of the IEEE, Vol. 96, No. 4, pp. 589-601. (2008).
- [27] Chang, C. and C. Lin, 2011: LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, Vol. 2, No. 3, pp. 1-39. (2011).
- [28] Barla A, Odone F and Verri A: Histogram intersection kernel for image classification. International Conference on Image Processing, ICIP. Italy, Vol. 3, pp. 513-516. (2003).
- [29] Jianguo, Z, et al: Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study. Conference on Computer Vision and Pattern Recognition Workshop, pp. 213-238. (2006).

- [30] Bosch A, Zisserman A and Munoz X: Representing shape with a spatial pyramid kernel [C]. Proceedings of the 6th ACM international conference on Image and video retrieval. NK, USA. pp. 401-408. (2007).
- [31] Dalal N and Triggs B: Histograms of oriented gradients for human detection. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, Vol. 1, pp. 886-893. (2005).
- [32] http://en.wikipedia.org/wiki/Support_vector_machine.
- [33] Siddiquie, B, Vitaladevuni S and Davis L: Combining Multiple Kernels for Efficient Image Classification. Workshop on Applications of Computer Vision (WACV), pp. 1-8. (2009).
- [34] Fu H, Chi Z, Feng D: Attention driven image interpretation with application to image retrieval. Pattern Recognition, Vol 39, No 7. pp. 1604-1621. (2006).
- [35] Salah A A, Alpandin E, Akarn L: A selective attention-based method for visual pattern recognition with application to handwritten digit recognition and face recognition. IEEE Trans on Pattern Analysis and Machine Intelligence. Vol. 24, No 3. pp. 420-422. (2002).
- [36] I. A Rybak, V I Gusakova, A V Golovan, et al: A model of attention-guided visual perception and recognition, Vision Research. Vol.38, pp. 2387-2400. (1998).
- [37] Ma Y F, Zhang H J: Contrast-based image attention analysis by using fuzzy growing [C]. Proceedings of the Eleventh ACM International Conference on Multimedia, New York: ACM Press, pp. 374-381. (2003).
- [38] Goferman S, L. Manor Z and Tal A: Context-aware saliency detection. pp. 2376-2383. (2011).
- [39] Itti L, Koch C and Niebur E: A model of saliency-based visual attention for rapid scene analysis [J]. IEEE TPAMI, Vol. 20, No 11. pp. 1254-1259. (1998).
- [40] 安图搜, <http://www.antuso.com/>.
- [41] Taotaosou, <http://www.taotaosou.com/>.

- [42] Riya's Like.com Is First True Visual Image Search.
<http://techcrunch.com/2006/11/08/riyas-likecom-is-first-true-visual-image-search/>.
- [43] K. Mikolajczyk and C. Schmid: Scale & affine invariant interest point detectors. International journal of computer vision. Vol. 60, No 1, pp. 63-86, (2006).
- [44] Mikolajczyk K, Tuytelaars T, Schmid: Zisserman A et. Al: A comparison of fine region detectors. International journal of computer vision. Vol. 65 No. 1, pp. 43-72. (2005).
- [45] http://en.wikipedia.org/wiki/Feature_detection_%28computer_vision%29.
- [46] http://en.wikipedia.org/wiki/Blob_detection.
- [47] Lindeberg T: Scale-Space Theory in Computer Vision. Springer. (1994).
- [48] Lindeberg T: Feature detection with automatic scale selection (abstract page). International Journal of Computer Vision. Vol 30, No 2. pp. 77-116. (1998).
- [49] Lindeberg T: Scale invariant feature transform. Scholarpedia, Vol 7, No 5, pp. 10491. (2012).
- [50] Lowe D G: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision. Vol 60, No 2. pp. 91-110. (2004).
- [51] Ashbrook A, Thacker N, Rockett P and Brown C: Robust recognition of scaled shapes using pairwise geometric histograms. Proceedings of the sixth British Machine Vision Conference, Birmingham, UK, pp. 503-512. (1995).
- [52] Belongie S, Malik J and Puzicha J: Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 24, No 4, pp. 509-522. (2002).
- [53] Krystian M, Cordelia S: A Performance Evaluation of Local Descriptors. IEEE Trans. Pattern Anal. Mach. Vol 27, No 10, pp. 1615-1630. (2005).

- [54] Oliva A, Torralba A: Modeling the shape of the scene: a holistic representation of the spatial envelope, *International Journal of Computer Vision*. No 42, No 3, pp. 145-175. (2001).
- [55] Torralba A, Oliva A: Semantic organization of scenes using discriminant structural templates, *International Conference on Computer Vision*, Korfu, Greece, pp. 1253-1258. (1999).
- [56] Oliva A, Torralba A: Scene-centered description from spatial envelope properties, in: *International Workshop on Biologically Motivated Computer Vision*, LNCS, Tuebingen, Germany, Vol 2525, pp. 263-272. (2002).
- [57] Van Gemert J C, Veenman C J, Smeulders A W M, et al: Visual word ambiguity. *Pattern Analysis and Machine Intelligence*, Vol 32, No 7, pp. 1271-1283. (2010).
- [58] Wright J, Yang A Y, Ganesh A, et al: Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 210-227. (2008).
- [59] Wright J, Ma Y, Mairal J, et al: Sparse representation for computer vision and pattern recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Champaign, Urbana, IL, USA, Vol 98, No 6. pp. 1031-1046. (2010).
- [60] Yang, J, Yu K, Gong Y, et al: Linear spatial pyramid matching using sparse coding for image classification [C], *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, Florida, USA, pp. 1794-1801. (2009).
- [61] Shawe-Taylor J: *Kernel Methods for Pattern Analysis*. Cambridge university press. (2004).
- [62] Bosch A, Zisserman A and Munoz X: Representing shape with a spatial pyramid kernel. *Proceedings of the 6th ACM international conference on image and video retrieval*. NK, USA, pp. 40-408. (2007).

- [63] Zhiyong L: Intelligent traffic control theory and its applications. Science Press. (2003).
- [64] Ning J, et al: Geomagnetic sensor models based classification algorithm. Application Research of Computers, Vol. 27. (2010).
- [65] Hinton, G. E. and Salakhutdinov, R. R: Reducing the dimensionality of data with neural networks. Science. (2006).
- [66] Bengio Y, Lamblin P, Popovici P: Larochelle, H. Greedy Layer-Wise Training of Deep Networks. NIPS. (2006).
- [67] Deep Learning and Unsupervised Feature Learning. <http://www.stanford.edu/class/cs294a/>.
- [68] Vincent P, Larochelle H, Bengio Y and Manzagol P: Extracting and Composing Robust Features with Denoising Autoencoders. Proceedings of the 25th international conference on Machine learning. New York, NY, USA. pp. 1096-1103. (2008).
- [69] http://deeplearning.stanford.edu/wiki/index.php/UFLDL_Tutorial.
- [70] Training a deep autoencoder or a classifier on MNIST digits. <http://www.cs.toronto.edu/~hinton/MatlabForSciencePaper.html>.