# UNDERSTANDING USER INTENTIONS IN VERTICAL IMAGE SEARCH

BY

**Yuxin Chen**

Submitted to the Graduate Degree Program in
Computer Science and the Graduate Faculty of the
University of Kansas in Partial Fulfillment of the
Requirements for the Degree of Master of Science

---

Chairperson Dr. Bo Luo

---

Dr. Xue-wen Chen

---

Dr. Brian Potetz

Date Defended: July 7, 2011

The Thesis Committee for Yuxin Chen
certifies that this is the approved version of the following thesis:

# UNDERSTANDING USER INTENTIONS IN VERTICAL IMAGE SEARCH

Chairperson Dr. Bo Luo

Date Approved: July 7, 2011

# Abstract

With the development of Internet and Web 2.0, large volume of multimedia contents have been made online. It is highly desired to provide easy accessibility to such contents, i.e. efficient and precise retrieval of images that satisfies users' needs. Towards this goal, content-based image retrieval (CBIR) has been intensively studied in the research community, while text-based search is better adopted in the industry. Both approaches have inherent disadvantages and limitations. Therefore, unlike the great success of text search, Web image search engines are still premature.

In this thesis, we present iLike, a vertical image search engine which integrates both textual and visual features to improve retrieval performance. We bridge the semantic gap by capturing the meaning of each text term in the visual feature space, and re-weight visual features according to their significance to the query terms. We also bridge the user intention gap since we are able to infer the "visual meanings" behind the textual queries. Last but not least, we provide a visual thesaurus, which is generated from the statistical similarity between the visual space representation of textual terms. Experimental results show that our approach improves both precision and recall, compared with content-based or text-based image retrieval techniques. More importantly, search results from iLike are more consistent with users' perception of the query terms.

# Acknowledgements

I would like to express my deep-felt gratitude to my advisor, Dr. Bo Luo of the EECS Department at The University of Kansas, for his advice, inspiration, enduring patience and constant support. He has been guiding me, both consciously and unconsciously, through the time of my graduate research. I appreciate all his contributions of efforts, ideas, and funding to make my Master's experience productive and stimulating. It has been such an honor to be his graduate student.

I would also like to gratefully acknowledge the support and guidance of my committee member Dr. Xue-wen Chen, for his valuable advice and continuous encouragement. He has been supportive since my senior year at college, and contributed immensely to my professional time at KU. I would like to thank my committee member Dr. Brian Potetz, for his time, helpful comments and insightful questions for my dissertation. I'm also thankful for the great examples he has provided as an excellent professor.

In regards to the professional and academical life in KU, I have had the pleasure to work with or alongside the great colleagues and office mates in ITTC. I am grateful for the fun time during my research with the InfoSec Group members. Thanks to Hariprasad Sampathkumar for his contributions and enthusiasm for the iLike project; thanks to Yuhao Yang, Hongliang Fei, and all other amazing colleagues and friends who has stuck it out with me in graduate research and study.

My time at KU has been made enjoyable in large part due to the friends and groups

that became an important part of my life. I am specially thankful for time spent with the International Friends in KU, for the unlimited kindness they have shown to me. Special thanks to Len Andyshak, Morgan Scott, Joshua Shireman and Patrick Nadvornik for accompanying me along my spiritual journey and witnessing my personal growth. Meanwhile, I'm grateful for the joyful time I spent with my conversation parter Kyle Whitaker and his family. I would also like to express my deep appreciation to my friends Weizhe Zhang, Zhaojun Yang, Pan Pan, He Shen, Xiaoming Lu, Ningyu Shi, Anthony Effertz and Jiajia Liu, for both physical and moral supports during the past two years.

At last, I would always like to thank my family for all their endless love and encouragement - for my parents and grandparents who raised me with special love and supported me in all my pursuits, and for my brother and sister who are always standing by my side for any of my decisions. Thank you.

NOTE: This thesis was submitted to my committee on July 7, 2011.

# Table of Contents

# Chapter 1. Introduction

Due to the explosive growth of multimedia information on the Internet, there arise enormous demands for effective semantic retrieval over large scale visual databases. Typical visual retrieval methods rely heavily on keywords-based query over meta-data, i.e., retrieving and ranking images based on surrounding text or user-generated annotations. However, text-only search methods may fall short for a variety of reasons, such as when visual databases have little or no textual metadata, or if tags are inaccurate or ambiguous. In addition, there are times and situations when we imagine what we desire, but are unable to express this desire in precise wording. Take, for instance, a desire to find the perfect wedding dress from a bridal shop. Any attempt to depict what makes a dress "perfect" for you may end up undervaluing the beauty of imagination. To some extent, it may be easier to find such a dress by looking through the shop's dress collection and making unconscious "matches" with the one conjured by imagination, than to use textual descriptions that fail to capture the very essence of perfection. Therefore, when it comes to mining multimedia data, visual interpretation of image/video content for indexing is of great importance in the research filed.

Content-based image retrieval (CBIR), as we see it today, is any technology that in principle helps to organize digital picture archives by their visual content [Datta et al., 2006a]. The current state-of-the-art in CBIR holds some promise and maturity to be useful for real-world applications. For example, Google and Yahoo! are household names nowadays

primarily due to the benefits reaped through their use; online photo-sharing has become extremely popular with Flickr, which hosts hundreds of millions of pictures with diverse content. While we witness continued effort in solving the fundamental open problem of robust image understanding in the last decades, the research community has paved the way of this emerging area, and triggered stronger association of weakly related fields, such as database systems, computer vision, machine learning, information theory and and psychology [Wang et al., 2006a]. What we see today in image retrieval literatures as a few cross-field publications may very well spring into new fields of study in the foreseeable future.

Despite the fact that image retrieval has been enjoying a sustainable development in the previous years, there are still intrinsic difficulties in solving the core problems. As a real-world technology, one problem with all current CBIR approaches is the reliance on visual similarity for judging semantic similarity, which may be problematic for developing efficient search algorithms. On the other hand, What the average end-user can hope to gain from such applications, under what circumstances a typical user feels the need from a CBIR system, and how a particular user expects the system to aid in this process are some key questions to be answered in a successful system design. To be brief, the need of the hour is to establish how Web image retrieval techniques can reach out to the common population, in the way that both visual contents and user intentions are well interpreted. In our current research, we restrict the discussion to a specific subject - vertical image search. Our efforts in addressing such problems will be elaborated throughout this thesis.

## 1.1 Problem description

Amidst the recent progresses of the CBIR-associated fields, it is important to recognize the shortcomings of context-based indexing and searching of multimedia information over the Internet. Unlike the great success of text-based web search, the research community is still struggling on such area. In particular, major breakthroughs are expected to overcome some key challenges: first and foremost, visual feature similarities are not necessarily correlated with content similarities. There exists a *semantic gap*, which is the gap between low-level visual features and high-level semantic concepts, i.e. the gap between vision and perception. Second, there also exists a *sensory gap* between the object in the world and the information in a (computational) description derived from a recording of that scene, which makes recognition from image content challenging due to limitations in recording. Third, it is difficult to handle the excessive computation caused by high dimensional data, i.e., visual features extracted from images. Further more, most of the existing purely CBIR prototypes still use offline image databases that are not comparable with the scale of the Web, and the algorithm complexity turns to be unendurable once scaled up. Meanwhile, advance in indexing high dimensional data is far less promising than indexing techniques in the text domain. Last but not least, it is also difficult for users to provide or sketch a good query in the query-by-example scenario.

Very large-scale multimedia repositories (e.g. the Library of Congress Prints and Photographs Catalog) are only indexed and retrieved by manually annotated metadata - textural features. Recently, some simple CBIR methods have been incorporated into com-

mercial web image search engines. However, they still mostly rely on text-based methods, i.e. indexing, retrieving and ranking images based on surrounding texts or user-generated annotations. With the advances of text-based indexing, such systems demonstrate superior efficiency so that they are capable to handle very large amount of image data collected from the Web. However, the search performance of TBIR approaches is not always reliable, according to the following reasons:

- It is not always easy to accurately identify "surrounding texts". Textual description of Web images could be embedded at random positions in a webpage, and may get even more complicated owing to various formatting schemes from different network server provider.

- Surrounding texts do not necessarily describe the image content. Normally the Internet is flooded with massive abundant information, which makes it very difficult to extract "useful " features without any specific knowledge about the object, such as textual content and Website structure.

- Perceptions and descriptions of visual contents are very subjective and inconsistent. Search engine users and content creators (narrators) may use different terms. Besides, user-generated tags are usually short, hence there is more likely to be discrepancies between annotators' and users' vocabularies.

## 1.2 Proposed solution

To remedy the problems of text-only or visual-content-only image retrieval systems, some recent approaches have proposed alternative routes to utilize both textual and visual features in Web image search, e.g. [Luo et al., 2003, Cui et al., 2008b, Cui et al., 2008a, Jing et al., 2006, Wang et al., 2007]. They are mostly two-phase hybrid approaches, which first use text retrieval to obtain a candidate result set, and employ CBIR methods to further process (e.g cluster or rank) the candidates. In this way, image and text contents are not semantically associated - image (visual) features and textual features are used separately.

In this thesis, we present a vertical search engine, namely iLike, that truly integrates both text and visual features to improve image retrieval performance. As mentioned in [Smeulders et al., 2000a], narrow image domains usually have limited variability and better-defined visual characteristics, which makes content-based image search a tad bit easier to formulate (correspondingly, the high variability and unpredictability of the broad domains in generalized image search makes it more challenging). In the scenario of vertical search, we have a better chance to integrate visual features from images and textual features from surrounding text contents. First, text contexts are better organized, hence focused crawlers/parsers are able to generate data patterns and structured data. Second, we are able to associate text content with images with higher confidence, e.g. product images and product descriptions, paintings, and introductions, etc. Thrid, with the knowledge of the focused domain, we are able to select image features and similarity measures that are more effective for the domain. Finally, computation issue becomes less critical for a smaller data

set.

In addition to the merits of convenient system design, vertical search makes it easier for image retrieval evaluation as well, and appropriate modifications must be made to baseline evaluation metrics for consistency [Huijsmans and Sebe, 2005]. iLike retrieves relevant images by their association with search queries, where there is no clear intent of a picture, but instead the search proceeds by iteratively refined browsing. For example, a search query with keyword "floral" may initiate a group of product images with floral prints, and the search results will then be refined according to the user intentions interpreted by the proposed algorithm (i.e., what user want by providing keyword "floral"). The baseline approach follows the same procedure as iLike does, except for skipping the core algorithm of user intention interpretation.

Compared with existing research, we take a different approach that focuses on learning the association between textual and visual features from a very-large scale data set. With the extreme popularity of social media, we are able to collect a large image database with reasonable-quality labels. We first extract both textual features and low-level image features from the collected data set to identify associations between visual features and textual features at the level of image vs. text. However, due to the existence of the semantic gap, such associations make little sense at object/region vs. term level. We propose a statistical learning model to discover the inherent connections between the concepts behind textual terms with regions and low-level visual features from images, i.e. to map textual entities into visual feature space. Particularly, we will build a computational model to

extract concepts from textual terms, and link them with regions and features extracted from images. The model will be trained and tested on a large scale image dataset crawled from social media sites.

We have implemented iLike as a vertical product search engine for apparels and accessories, but the technique could be easily adopted in many domains where textual and visual contents co-exist. In iLike, we discover the relationships between textual features extracted from product descriptions and image features extracted from product pictures. Notable among the concepts introduced in this thesis are subspace transformation and "visual thesaurus". The overall goal therefore remains to bridge the semantic gap using the available visual features, associated textual descriptions and relevant domain knowledge to support varied search categories, ultimately to satiate the user.

## 1.3 Contributions

Our technical contributions are three-fold: (1) we bridge the *semantic gap* by integrating textual and visual features and hence significantly improve the precision of content-based image retrieval. We also improve the overall recall by yielding items that would otherwise be missed by searching with either type of the features. (2) We bridge the *user intention gap* between users' cognitive intentions (information needs) and their textual forms (queries) received by the IR systems. Our system is able to perceive users' "visual intentions" behind search terms, and apply such intention to leverage on relevance assessment and ranking. (3) By assessing representations of keywords in the visual feature space, we are able to

discover the semantic relationships of the terms and automatically generate a thesaurus based on the "visual semantics" of words.

Success of the proposed project will bring major impacts to research communities such as information retrieval, computer vision, and multimedia. We not only discover and explore a new viable path to the open problem of content-based multimedia information retrieval, but also introduce novel ideas and methods to all the related areas, and further stimulate research discoveries and industrial applications. The methods will be used for web image search, mobile search, information retrieval from very large offline multimedia repositories, automatic image tagging, synthetic image generation from text, image understanding, and robotics, etc. With the massive volume of multimedia contents that are produced and accessed in our daily life, the expected social impacts and industrial interests will be significant.

## 1.4 Thesis organization

The thesis is organized into the following chapters:

- **Chapter 1: Introduction** - An introduction to the area, the challenge and problem description, the proposed solution and evaluation criteria, and the contributions of this research.

- **Chapter 2: Background** - An overview of related image retrieval techniques like CBIR along with automated and folksonomic tagging approaches.

- **Chapter 3: System Overview** - A description of the system architecture, along with details of data acquistion and feature extraction methods.

- **Chapter 4: The Method** - A detailed elaboration on our method of image retrieval, which integrates both textual and visual features to build a weight vector for retrieving relevant images for a given query.

- **Chapter 5: Experimental Results** - A description of the evaluation strategy and results.

- **Chapter 6: Conclusion and Discussions** - A conclusion with discussions on the merits of our approach in comparison to other approaches.

# Chapter 2. Background

Early information retrieval systems like the Catalog records for Prints and Photographs in the Library of Congress involved manual annotation of images with textual meta-data which were used by the text based retrieval methods for providing access to those images. Manual annotation of data would however, be an extremely time consuming and expensive task when applying for large scale image databases. Also it is not possible to describe images accurately and completely just using a set of keywords. Most often discrepancies between the query words and keywords used to tag the images lead to poor image retrieval results. Content Based Image Retrieval (CBIR) systems were developed with a view to overcome the drawbacks of the meta-data based searches. CBIR techniques involve using visual features such as color, texture and shape information of the image to index and retrieve the image. Comprehensive surveys on CBIR can be found at [Smeulders et al., 2000b, Lew et al., 2006, Datta et al., 2006b].

In the context of CBIR, search has been described as a specification of minimal invariant conditions that *model the user intentions*, geared at bridging the semantic gap between high level image content and the low level visual features, while reducing the sensory gap due to accidental distortions, clutter, occlusion, etc. In this chapter, we provide an overview of the key theoretical and empirical contributions in the past years related to image retrieval and automatic image annotation. We also discuss significant challenges involved in the adaption of existing image indexing techniques for image retrieval research.

## 2.1 Image processing in CBIR

Feature selection is a primary step of any CBIR technique and involves selecting low level image features that can be used to suitably capture the image content. Most CBIR systems perform feature extraction as a preprocessing step: once obtained, visual features act as inputs to subsequent image analysis tasks, such as similarity estimation, concept detection, or annotation. So, the purpose of image processing in image retrieval must be to enhance aspects in the image data relevant to the query and to reduce the remaining aspects [Ma and Zhang, 1998]. We survey the key contributions over image feature extraction methods over color, the local texture, or local geometry/shape that are related to our proposed iLike prototype in this section. Considerations for indexing effectiveness are reflected during the discussion.

### 2.1.1 Color image processing

Color has been an active area of research in image retrieval, as for its superior discriminating potentiality of a three-dimensional domain compared to the single dimensional domain of gray-level images. Since the human perception of color is an intricate topic, and the recorded color varies considerably with the environment (i.e., the viewpoint of the camera, the orientation of the surface, the intensity and position of the illumination, etc ), capturing perceptual similarity turns out to be a challenging task.

RGB representations are in wide-spread use for image representation. However, as RGB representations describe the image in its literal color properties, and thus it makes most

sense when recording in the absence of variance, as is the case, where two-dimensional images are recorded in frontal view under standard conditions. To overcome the sensory discrepancy, the HSV space representation of an image is often selected for its invariant properties. The hue is invariant under the orientation of the object with respect to the illumination and camera direction and hence more suited for object retrieval.

One common approach to cope with the inequalities in observation due to surface reflection is to search for clusters in a color histogram of the image. The earliest use of color histograms for image indexing was that in [Swain and Ballard, 1991], demonstrating that histograms of multicolored objects provide a robust, efficient cue for indexing into a large database of models. Two indexing techniques, namely Histogram Intersection and Histogram Backprojection, were proposed for solving the image matching problem and object location problem in crowded scenes. Such method was further developed in [Stricker and Orengo, 1995] with improved indexing techniques to color information in digital images. In [Huang et al., 1998], color correlograms were proposed as enhancements to histograms, that took into consideration spatial distribution of colors as well.

Rather than histogram approaches for color processing, innovations in color constancy were made by taking specular reflection and shape into consideration [Finlayson, 1996]. Color constancy is the capability of humans to perceive the same apparent color in the presence of variations in illumination which change the physical spectrum of the perceived light. In this work, an illumination invariant color representation was employed to extract color features. Color constant indexing leads to some loss in discriminating power among

objects, but yields illumination independent retrieval instead.

### 2.1.2 Image texture processing

Many common textures (the structure and randomness of an image) are composed of small textons usually too great in number to be perceived as isolated objects. The elements can be placed more or less regularly or randomly. They can be almost identical or subject to large variations in their appearance and pose. In the context of image retrieval, research is mostly directed toward statistical or generative methods for the characterization of patches. Statistical features of grey levels were one of the earliest methods used to classify textures. [Haralick et al., 1973] suggested the use of grey level co-occurrence matrices (GLCM) to extract second order statistics from an image. Haralick defined the GLCM as a matrix of frequencies at which two pixels, separated by a certain vector, occur in the image. The distribution in the matrix will depend on the angular and distance relationship between pixels. Varying the vector used allows the capturing of different texture characteristics. Once the GLCM has been created, various features can be computed from it. The texture features, then, have been classified into four groups: visual texture characteristics, statistics, information theory and information measures of correlation.

Tamura et al. took the approach of devising texture features that correspond to human visual perception [Tamura et al., 1978]. They defined six textural features (coarseness, contrast, directionality, line-likeness, regularity and roughness) and compared them with psychological measurements for human subjects. The first three attained very successful

results and are used in our evaluation, both separately and as joint values. Coarseness has a direct relationship to scale and repetition rates and was seen by Tamura et al. as the most fundamental texture feature. Contrast aims to capture the dynamic range of grey levels in an image, together with the polarisation of the distribution of black and white. Directionality is a global property over a region. In [Tamura et al., 1978], the directionality strength of an image was calculated with a statistical measure from its directional histogram.

Besides the statistical analysis of image texture from its spacial distributions, wavelets have received wide attention. Many wavelet transforms are generated by groups of dilations or dilations and rotations that have been said to have some semantic correspondent. They have often been considered for their locality and their compression efficiency. One of the most popular signal processing based approaches for texture feature extraction has been the use of Gabor filters. These enable filtering in the frequency and spatial domain. It has been proposed that Gabor filters can be used to model the responses of the human visual system. [Turner, 1986] first implemented this by using a bank of Gabor filters to analyse texture. A bank of filters at different scales and orientations allows multichannel filtering of an image to extract frequency and orientation information. This can then be used to decompose the image into texture features. Classifying images based on the above features has been shown to be sucessful in literatures [Manjunath et al., 2001, Raimondo et al., 2009].

### 2.1.3 Shape and geometrical features

Image shape features refer to all properties that capture conspicuous geometric details in the image. Although searching for images using shape features has attracted much attention in past years, shape representation and description remains to be a difficult task. This is because when a three dimensional real world object is projected onto a two dimensional image plane, one dimension of object information is lost. It comes to the result that the shape extracted from the image only partially represents the projected object. In addition, shape is often corrupted with noise, defects, arbitrary distortion and occlusion, making the problem even more complex.

For retrieval, we need shape descriptors that allow a robust measurement of distances in the presence of considerable deformations. Shape descriptors can be divided into two main categories: region-based and contour-based methods. Region-based methods use the whole area of an object for shape description (i.e., [Khotanzad and Hong, 1988] utilized a set of Zernike moments calculated within a disk centered at the center of the image as shape descriptor), while contour-based methods use only the information present in the contour of an object, such as circularity, aspect ratio, discontinuity angle irregularity, length irregularity, complexity, right-angleness, sharpness, directedness, etc.

Local shape characteristics stem from directional color derivatives (or texture properties). In highly textured patches of diverse materials, such features were used to derive perceptually conspicuous details [Mojsilovic et al., 2000]. Scale space theory, which provides the theoretical basis for the detection of conspicuous details on any scale, was devised

as the complete and unique primary step in preattentive vision, capturing all conspicuous information. Contours of images represented in terms of geometric invariant moments [Dudani et al., 1977] have also been used to capture shape information. Besides conspicuous shape geometric invariants [Rivlin and Weiss, 1995], a method employing local shape and intensity information for viewpoint and occlusion invariant object retrieval was given in [Schmid and Mohr, 1997]. The method relies on voting among a complete family of differential geometric invariants, allowing efficient image retrieval from large database of images.

Global shape (object shape) analysis is a dense image data field different from local shape evaluation. To extract object-specific information contained in images, the theoretically best way is by segmenting the object in the image. However, in many cases, it is not necessary to know exactly where an object is in the image as long as one can identify the presence of the object by its unique characteristics. With a proper feature accumulating algorithm (i.e., [Swain and Ballard, 1991]), the object internal features are largely identical to the accumulative features computed over the object area. [Mehtre et al., 1997] provided abundant comparison of shape for retrieval, evaluating many features on a 500-element trademark data set. Straightforward features of general applicability include Fourier features and moment invariants [Vijay and Bhattacharya, 2009], sets of consecutive boundary segments, or encoding of contour shapes [Esperanca and Samet, 1997].

### 2.1.4 Storage of image features

Practically, the most interesting applications of retrieval are on large data sets, where there is statistically sufficient coverage of the image spectrum and learning general knowledge from the data sets makes sense. When storing the feature vectors in a database, linear file with one record of each feature vector, we have to scan through all feature vectors. In that case, we are bound to perform $N$ fetches of a record plus subsequent operations to find the data vector most similar to the query feature vector. The response time of a image indexing and retrieval system will be possibly out of reach for large volume data sets (say, half a million or more).

In addition to the number of images, the dimension of the image vector can also be considerable for the performance systems. One of the primary challenges of CBIR techniques is the intensive computational cost due to the need for indexing high dimensional visual features thereby preventing the wide spread adoption of CBIR for Web image search. In the example of a wavelet histogram for texture-based retrieval [Smith and fu Chang, 1996], an image had a nine-dimensional vector for each pixel compressed to a 512-bin histogram to a total of 5122 histograms of 512 bins per image. The shape indexing technique [Sharvit et al., 1998] represented an image vector by a hierarchically ordered set of six types of nodes and three types of links, each encoding a number of image descriptors.

Moreover, indexing in high dimensional spaces is difficult by the curse of dimensionality, a phenomenon by which indexing techniques become inefficient as the dimensionality of the feature space grows [Hughes, 1968]. A lot of traditional multidimensional indexing

techniques, such as k-d tree, quad-tree, R-tree and its variant $R^+$-tree and $R^*$-tree, are usually not scalable to dimensions higher than 20 [White and Jain, 1996].

## 2.2 Image annotation

Image annotation or image tagging is an area closely related to image retrieval. Image annotation techniques were primarily developed to address the semantic gap of CBIR techniques and to help improve image search quality. The task of image tagging involves assigning a set of text labels that can be used to describe an image or what it contains. In the thesis, effective image tagging is treated as a means of satisfactory image search. We divide them according to their user-dependence into two types of approaches, i.e., automatic image tagging and folksonomic tagging, and discuss in the following section some realistic scenarios that arise in image annotation.

### 2.2.1 Automatic image tagging

Automated image tagging techniques were developed to address the semantic gap of the CBIR systems and also to overcome the tedium of manual annotation. These systems help in automatically adding tags or meta-data for the images with words that can be used to bridge the semantic gap. The task of automated image tagging has primarily been treated as a pattern classification problem and hence several supervised machine learning techniques have been attempted for it. In general, the learning task is to build a classifier or

model that identifies the mapping between the low-level image features and the high-level concepts of keywords that have used to classify the images as a part of a training set. Once the classifier or model is built, it calculates the similarity of all the trained classes and assigns the unlabeled instance to a class with the highest similarity measure.

We have witnessed a wealth of promise in automatic image tagging as an emerging technology. In [Vailaya et al., 2001], a hierarchical three-stage classification using Bayes classifiers was proposed. The images were first classified as outdoor or indoor, then the outdoor images were further classified as city or landscape and finally the landscape images were classified into sunset, forest, and mountain classes. [Jeon et al., 2003] proposed an automatic image annotation model based on cross-media relevance models. They assumed that regions in the image can be represented by using a small vocabulary of blobs. Blobs were generated from image features using clustering. Given a training set with annotations, using probabilistic models they were able to predict the probability of generating a word given the blobs in an image. On the other hand, [Barnard et al., 2003] treated image annotation as a machine translation problem. Other means of text-image interaction, which make use of visual information to help annotate images have also been proposed [Li et al., 2006, Zhou and Dai, 2007]. [Wang et al., 2006b, Kennedy et al., 2006] made use of image search results as a means to improve the annotation quality. [Li and Wang, 2008] developed ALIPR - "Automatic Linguistic Indexing of Pictures - Real time", which automatically generated tags in real time based on only the pixel information in the image. It was based on a generative modeling technique, where a model for producing image seg-

ments and words was built from the training set. For a testing image, text terms were ranked by the posterior probabilities, and top results were selected as the tags for the image. This method was built upon the ALIP, which used a 2-D Multi-resolution Hidden Markov Model (MHMM) [Li and Wang, 2005]. Earlier systems were not fast enough to consider performing image annotation in real time. By exploiting statistical relationships between words and images and without having to identify individual objects in the images, [Li and Wang, 2008] demonstrated that it was possible to provide more than 98% of the words with at least one correct annotation out of the top 15 selected words, making it possible to annotate images in real time.

### 2.2.2 Folksonomic tagging

Automatic image tagging approaches were shown to be most effective when the keywords have frequent occurrence and strong visual similarity. However, it is still a challenge for these techniques to annotate images with more specific or visually less similar keywords. It has been shown that manual annotation of images with user generated labels can be used to improve the quality of the image search results [Lieberman et al., 2001, von Ahn and Dabbish, 2004]. Manual annotation of images can be categorized into tagging and browsing. Tagging allows users to annotate an image with a chosen set of keyword set or vocabulary. Google Image Labeler (http://images.google.com/imagelabler/) and Flickr image tags (http://www.flickr.com/photos/tags/) are some examples of such efforts. On the other hand, browsing requires users to sequentially browse a set of images and judge

their relevance to a predefined keyword. [Yan et al., 2007] introduced a hybrid approach called frequency based tagging that combines both tagging and browsing into a unified framework.

The practice of creating and managing tags is referred to as folksonomic tagging in the context of Web 2.0, which aims at facilitating sharing of user generated content. Folksonomic tagging of images by users who are not trained in image annotation usually tends to be subjective and often leads to ambiguous annotations, especially when there is no fixed vocabulary for the annotation. In order to overcome the tedium of manual tagging and to improve the quality of the image tags, automated tag recommendation systems like [Wu et al., 2009] have been developed. With a growing number of social network sites which allow sharing and tagging of photos, methods like [Sawant et al., 2010] have been used to develop fully automated and folksonomically scalable tag recommendation systems. Such systems leverage the collective vocabulary of a group of users, which is less susceptible to noise from an individual's subjective annotation, resulting in high quality image tags.

## 2.3 Image search on the web

### 2.3.1 Text-based image retrieval

Current web based image search engines like Google Image Search(http://images.google.com), Yahoo image search(http://images.search.yahoo.com) and Bing(http://www.bing.com) primarily rely on textual meta data for image retrieval. They take the keywords specified

as apart of the search query and match them with the meta-data associated with the image which may include the image file name, the image URL, any alternate text provided for the image and any other surrounding text present in the web page containing the image. Then again, since the textual information surrounding an image may not necessarily correctly describe the image, the retrieval performance of the meta-data based searches can still be poor. There are also more aggressive text based methods [Aslandogan et al., 1997, Shen et al., 2000] which were employed on the text surrounding the images to better associate semantic information with the images. Link analysis techniques [Lempel and Soffer, 2001, Cai et al., 2004] have also been employed to improve the search performance.

### 2.3.2 Image retrieval with visual features

As we have discussed at the beginning of chapter 2, content based image retrieval techniques incorporates various visual features for image indexing and search. When the information from images is captured in a feature set, there are two possibilities for endowing them with meaning: One derives an unilateral interpretation from the feature set, while the other one compares the feature set with the elements in a given data set on the basis of a similarity function. With sufficient data and computational power, it is possible to learn the semantics of objects from their appearance. Along this route, latent semantic indexing [Deerwester et al., 1990] was proposed to practical use in image retrieval. First, a corpus was formed of documents (in this case, images with a caption) from which features were

computed. Then, by singular value decomposition, the dictionary covering the captions was correlated with the features derived from the images. Such approach, together with the majority of semantic retrieval techniques, essentially boil down to a classification problem. For a review on statistical pattern recognition, see [Jain et al., 2000].

Some recent works relies on a Bag of Visual Words approach to categorize, index and retrieve images [Lazebnik et al., 2006, Tirilly et al., 2008, Yang et al., 2007, Jiang et al., 2007]. Based on local descriptors for the images, this approach is similar to the bag of words representation for text documents in terms of forms and semantics. It describes and detects interesting regions in the images, builds the visual vocabulary and indexes the images based on this vocabulary. Existing weighting schemes for indexing which are mostly migrated from the text retrieval domain do not take into account the difference between textual and visual words. [Bouachir et al., 2009] proposed an improvement to indexing for the Bag of Visual Words approach. In the approach, they make use of Scale Invariant Features Transform (SIFT) to extract the local features and make use of a new weighting scheme based on a Fuzzy model to index the images. Sequently, [Kogler and Lux, 2010] applied a fuzzy clustering technique for visual words creation and visual words assignment and showed that fuzzy clustering led to more robust results in terms of retrieval performance.

### 2.3.3 Hybrid methods for image search

Several prototypes for content-based image search on the web are available in literatures [Frankel et al., 1996, Sclaroff et al., 1997, Mukherjea et al., 1999, Chen et al., 2001,

Kompatsiaris et al., 2001]. However, our approach to image retrieval is significantly different from these existing approaches in the way that we integrate both textual and visual features to improve retrieval performance. In a web search context there are often both images and surrounding text available as a part of the web page which when used together can help to bridge the semantic gap and also provide for better indexing by integrating both textual and visual features. Luo et al. [Luo et al., 2003] introduced a two-stage hybrid approach where a text-based search is first used to generate an intermediate result set with high recall and low precision which is then refined at the second step by applying CBIR to cluster or re-rank the results. Although this approach suffers from over simplified image features and clustering methods, the idea of applying CBIR after text search seems to be a viable alternative. More recently, Bing image search (http://www.bing.com/images/) has started to employ CBIR techniques to re-rank search results [Cui et al., 2008b, Cui et al., 2008a], when users select the "show similar images option". More complicated re-ranking algorithms [Jing et al., 2006, Wang et al., 2007] have been proposed to improve search performance and user experience.

### 2.3.4 Domain-specific image search

Applying CBIR for the general web is hard problem. Some research efforts have proposed to apply CBIR to vertical search, which cater only to specific sub-domains of the web. These vertical search engines employ focused crawlers to crawl constrained subsets of the general web, and evaluate user queries against such domain specific collections of docu-

ments. Besides leveraging the benefits of a smaller data set, these engines can also employ domain knowledge to help with relevance assessment and result ranking. Some example of vertical image search include: photo album search [Zhang et al., 2006], product search (http://www.like.com/, http://www.riya.com/), airplane image search (http://www.airliners.net/), etc. There are also off-line image retrieval systems that work on domain specific collections of images, such as personal albums [Zhang et al., 2004, Cui et al., 2007], leaf image search [Wang et al., 2002, Dua et al., 2007], fine arts images search [Yee et al., 2003], etc. These approaches made use of domain specific knowledge in image preprocessing, feature selection and similarity measurements. For example, leaf image searches may have emphasis on shape and texture features while personal album searches may employ face recognition methods to improve search performance.

Thus far, we have introduced the background knowledge and most of the related works of our research. In our approach to improve image retrieval performance, we integrate both text and visual features and apply it to the search of images in the vertical domain of clothing and apparels. Our efforts to build an image search system based on feature from cross domains varies from the previous works. We'll present and evaluate our iLike solution for vertical image search in the following chapters, as an aggressive attempt to overcome the core problem of image retrieval, which is to access the visual similarity at semantic level.

# Chapter 3. System Overview

## 3.1 System architecture

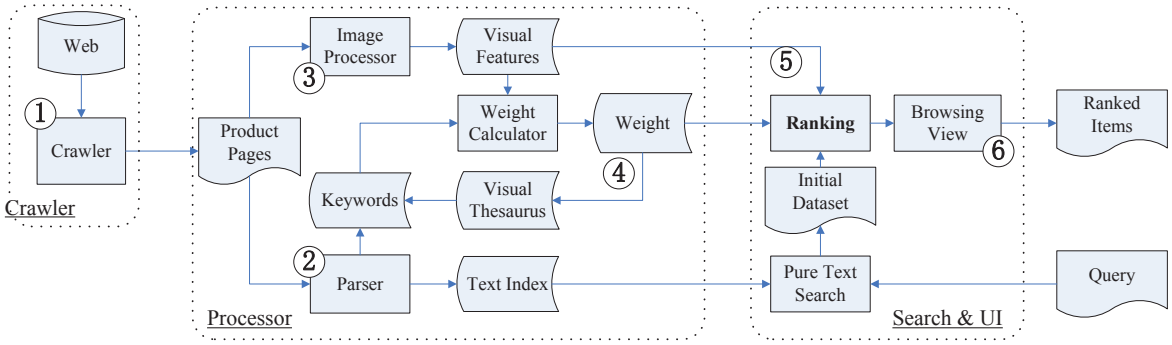



Figure 3.1: System overview of iLike.

Since our goal is to integrate textual and visual features in vertical search, it is of our interest to select a domain where text content is directly associated with image content. Online shopping, especially clothing shopping, is a good example of such domains. In shopping websites, text descriptions are always available with item images, and are usually faithful descriptions of the image contents. Moreover, we believe that both text descriptions and product images are equally important since: (1) from users' perspectives, they can only issue keyword queries for product search; on the other hand, while browsing the results, users focus more on visual presentations than the text specifications. (2) Due to different personal tastes, the descriptions of fashionable items are very subjective, hence traditional text-based search on such descriptions may not yield satisfactory results. Especially, the recall can be very low when there is a discrepancy between user's and narrator's tastes or

vocabularies. (3) In many cases, two items may have similar style in human perception, but we see huge difference in the visual features. Hence, pure content-based image search will not yield high recall either. Therefore, this is an ideal case in which we can demonstrate the power of combining visual and textual features in vertical search. Note that our arguments are based on fashion shopping, but they are also true in many other shopping categories. Therefore, our system could be migrated to other categories with minimum modification.

The iLike system is comprised of three major components: the Crawler, the (Pre-)Processor, and the Search and UI component. As shown in Figure 3.1: (1) the Crawler fetches web pages from retailer websites, where structured text descriptions and item images are both available. (2) The text parser preprocesses pages using a customized parser, and fits item information (e.g. title, description) into a pre-defined universal schema. Using classic text retrieval methods, text processor generates term dictionary and text index. (3) Simultaneously, the image processor segments product images and calculates low level visual features. (4) Next, we integrate textual and visual features by calculating a "centroid" and a weight vector in the visual feature space for each text term. With those text term weight vectors, we then construct a visual thesaurus for each text term, which in turn can be used to improve the quality of the weight vectors. Such vectors are further utilized in item ranking. (5) Finally, the User Interface provides query interface, as well as browsing views of search results.

In iLike , a user starts with a traditional text query (since query-by-example is not really practical in this scenario), and the system returns a ranked list of relevant items (namely

the *initial result set*) using classic text retrieval algorithms. For each result in the initial result set, we construct a new query by integrating textual and visual features from item images. Each expanded query is evaluated to find more "similar" items. More importantly, a weight vector which represents the "visual perception" behind the text query is enforced during evaluation of the expanded queries. For instance, with a query "silky blouse", the weight factor will increase the significance of some texture features, and fade out irrelevant features, hence correctly interpret the visual meaning behind search term "silky". The overall philosophy of our approach is to infer user intention from the query and enhance the high level features that are implicitly favored, while diversify (if possible) on other features.

## 3.2 Crawling and feature extraction

**Data Acquisition.** In the prototype, we have initially crawled a total of 42292 product items from eight online retailers: Banana Republic, Old Navy, Gap, Athleta, Piperlime, Macy's, Bluefly and Nordstrom. They all provide mid-sized hi-quality images and well structured textual description. We use focused crawlers to harvest both text and images. Please note that the system is easily expandable by implementing more customized crawlers and parsers.

For each product, we record the name, category, class, online product ID, local path of the main image, original URL, detailed textual description, color tags, and size information, if available. We use an unique id for each product item, to identify both the database record

and the image file. Text information is stored in a MySQL database (Version 5.0)[1], and all the customized software (e.g. focused crawlers) are written in C# programming language.

**Visual Features.**

In order to make a sufficient coverage of an image's semantic meaning, we attempt to diversify the part of feature selection. In iLike, a set of 401 commonly used texture, shape, intensity and color features are extracted to represent the low-level visual features of images.

We use gray-level co-occurrence matrix (GLCM)[Haralick et al., 1973] to capture the basic texture information: contrast, correlation, energy, and homogeneity of the grayscale images are calculated, each of which generating a 4-scale feature vector. To summarize the relative frequency distribution (which describes how often one gray tone will appear in a specified spatial relationship to another gray tone on the image), a vector of 13 Haralick texture features are extracted from the grey level co-occurrence matrices. Image coarseness and direction are obtained by calculating 3 dimensions of Tamura texture features [Tamura et al., 1978]. To capture texture patterns in frequency domain, we apply Gabor wavelet filters in 8 directions and 5 scales, acquiring a vector of 40 texture features. Besides, fourier descriptors[Vijay and Bhattacharya, 2009] are also employed, contributing 9-dimensional feature vector to our feature set. To extract the shape information, we represent the contour of an image in terms of 7 geometric invariant moments[Dudani et al., 1977], which are invariant under rotation, scale, translation and reflection of images. We capture

[1]http://www.mysql.com/

the spatial distribution of edge with 5 edge strengths generated from an edge histogram descriptor[Manjunath et al., 2001] in the MPEG-7 standard. The distribution of edges is a good texture signature that is useful for image to image matching even when the underlying texture is not homogeneous. As part of shape features, the edge orientation is represented by phase congruency features (PC) [Kovesi, 1999] and high-order moments of characteristic function (CF) [Teague, 1980]: A three-level Daubechies wavelet decomposition of the test image is performed before edge detection. At each level, the first four moments of phases, which are generated by Sobel edge detector, are obtained, together with the first three moments of the characteristic function, yielding a 106-dimensional feature vector. To capture the color distribution [Stricker and Orengo, 1995], we first divide an image into several blocks (we use 1 by 1, 2 by 2, 3 by 3 blocks in iLike), and then extract the first three moments of all blocks in each of the YCbCr channel, i.e., for a color image we store 90 floating point numbers as color moments (CM) features. The color histogram features are generated by color quantization approach. We map the original image into the HSV color space, and implement color quantization using 72 colors(8 levels for H channel, 3 levels for S channel and 3 levels for V channel).

The chosen features have been proved to work well for image classification in literature [Ma and Zhang, 1998, Stricker and Orengo, 1995, Manjunath et al., 2001], etc. On the other hand, we do not want our search performance to be overwhelmed by very complicate and computationally intensive visual features. Meanwhile, a comparative study [Deselaers et al., 2004] has shown that the effectiveness of visual features is dependent on

the particular task. However, such a specifically optimized system cannot be easily migrated to other domains, due to the labor-intensive manual feature selection process. Instead, in iLike features are automatically weighted based on their significance to the user intent (implicitly carried by the query). Less important features are faded out, while more important features are enhanced. Therefore, unlike other CBIR approaches, the "quality" of low-level visual features is not the key factor in our system. As a side effect, our method is robust: the ranking quality is less sensitive to the selection of low-level image features. We will further discuss on feature quality and correlation in chapter 4.

**Segmentation.** Our database contains images of products in all shapes and sizes. Various retailers have different specifications of their product demo, some of which have introduced non-ignorable errors to feature extraction. For instance, the presence of a lingerie model could significantly influence the feature distribution. To simplify and clean the representation of product images and minimize the error of features, we perform an "YCbCr Skin-color Model"[Kakumanu et al., 2007]-based image segmentation on selected domains(i.e, categories and shopping sites that usually have models in) to remove the skin area and highlight product items.

As a specific segment of online images, the content of online product images usually aligns in the center with a decent margin around, while the background color varies due to various styles of different retailers. When comparing the similarities of products across different shopping sites, it is important to focus only on the subjects and leave out the rest. In our framework, we enhance the local features of certain region by performing a multiple

size partitioning on the target image, and the weighting scheme of iLike will pick out the block(s) of interest.

**Normalization.** Our system uses diverse types of image features. However, features from different categories are not comparable with each other, since they take values from different domains. Without any normalization, search results will be dominated by those features taking larger values. To reduce the interferences brought by different feature types and scopes, we map the range of each feature $\vec{x}$ to $(0, 1)$:

$$y_i = \frac{x_i - min(\vec{x})}{max(\vec{x}) - min(\vec{x})} \tag{3.1}$$

in which $i$ indicates the $i$-th item. After normalizing, all the features are mapped into $\vec{y}$ with the same scale, and thus become comparable.

# Chapter 4. The Method

In multimedia information retrieval, the roles of textual feature space and visual feature space are complementary. Textual information better represents the semantic meaning, while visual knowledge plays a dominant role at the physical level. They are separated by the semantic gap, which is the major obstacle in content-based image retrieval. In this chapter, we present an innovative approach to bridge the semantic gap and allow easy transformation from one space to another.

## 4.1 Representing keywords

For online images and their descriptions, the textual description is a projection of the narrator's perception of the image content. However, there are difficulties using only text features to retrieve mixtures of image/textual contents: perception is a subjective matter, the same impression could be described through different words. Moreover, calculating text similarity (or distance) is difficult - distance measurements (such as cosine distance in TF/IDF space) do NOT perfectly represent the distances in human perception. For instance, from a customer's perspective, 'relaxed-cut' is similar to 'regular-cut' and quite different from 'slim-cut'. However, they are equally different in terms of textual representation (e.g. in vector space model).

To make up for the deficiency of pure text search or pure CBIR approaches, we ex-

Figure 4.1: some items that has the keyword "dotted" in their descriptions.

plore the connections between textual and visual feature subspaces. The text description represents the narrator's perception of the visual features. Therefore, *items share similar descriptions may also share some consistency in* ***selected*** *visual features*. Moreover, *if the consistency is observed over a significant number of items described by the same keyword, such a set of features and their values may represent the human "visual" perception of the keyword*. In addition, if items with different descriptions demonstrate a different value distribution on these selected visual features, we can further confirm the correlation between the terms in the textual description and these visual features.

For instance, let us look at the items with the keyword "dotted" in their descriptions (some examples are shown in Figure 4.1). Although they come from different categories and different vendors, they all share very unique texture features. On the other hand, they all differ a lot in other features, such as color and shape. It indicates that the term "dotted"

is particularly used to describe certain texture features. When a user searches with this term, her intention is to find such texture features, not about color or shape. In this way, many terms could be connected with such a "visual meaning". In iLike , the first step is to discover such "visual meanings" automatically.

**Base representation.** Suppose there are $N$ items sharing the same keyword, and each item is represented by a $M$-dimensional visual feature vector: $\vec{X}_k = (x_{k_1}, x_{k_2}, \ldots, x_{k_M})^T$, where $k \in [1, N]$. The mean vector of the $N$ feature vectors could be utilized as a *base representation* of the keyword in the visual feature space:

$$\vec{\mu} = (\frac{1}{N}\sum_{k=1}^{N} x_{k_1}, \frac{1}{N}\sum_{k=1}^{N} x_{k_2}, \ldots, \frac{1}{N}\sum_{k=1}^{N} x_{k_M})^T$$

When $N$ is large enough, $\vec{\mu}$ will preserve the common characteristics in the image features and smooth over the various sections. In such a manner, the mean vector is rendered as a good representation of the keyword. However, those $N$ feature vectors may not share consistency over *all* visual features, hence, not all dimensions of the mean vector make sense. As shown in the "dotted" example, those items are only similar in some texture features, while they differ a lot in color and shape features. Such consistency/inconsistency on the feature is a better indicator of the significance of the feature towards human perception of the keyword. Therefore, a more important task is to quantify such consistency or inconsistency.

## 4.2 Weighting visual features

As shown in the "dotted" example, features coherent with the human perception of the keyword tends to have consistent values; while other features are more likely to be diverse. To put it another way, suppose that we have two groups of samples: (a) *positive*: $N_1$ items that have the keyword in their descriptions, and (b) *negative:* $N_2$ items that do not contain the keyword. In this way, if the meaning of a keyword is coherent with a visual feature, its $N_1$ values in the positive group should demonstrate a different distribution than the $N_2$ values in the negative group. Moreover, the feature values in the positive group tend to demonstrate a small variance, while values in the negative group are usually diversified.

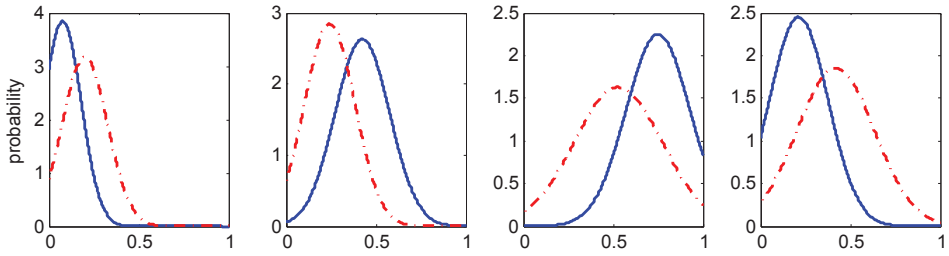



Figure 4.2: Examples of "good" feature distributions.

Figure 4.2 demonstrates the value distribution of eight different texture features for the keyword "dotted". In the figure, blue (solid) lines represent distributions of the positive samples, while red (dashed) lines represent the distributions of negative samples. Note that sample sets are fitted to normal distributions for better presentation in the figure. However, when we quantitatively compare both distributions, we do not make such assumption. For the selected textures features, distributions of the positive samples are significantly different

from negative samples (e.g. items described by the keyword is statistically different from other items in these features).

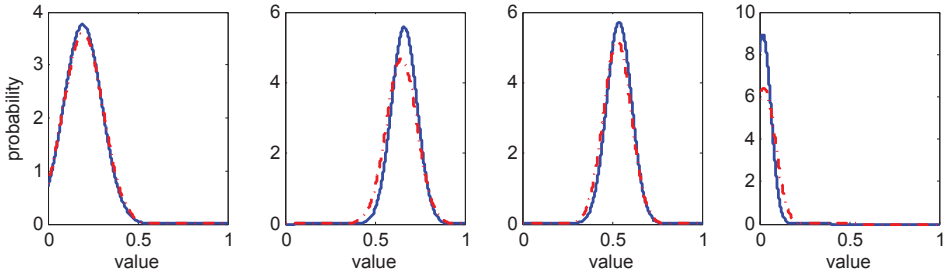



Figure 4.3: Examples of "bad" feature distributions.

On the contrary, the two distributions are indistinguishable for some selected color and shape features (Figure 4.3). As the "dotted" features vary in color in shape, products in a variety of categories of products have such pattern (i.e., there are "dotted" ties, skirts, bags, and pajamas, which are demonstrated in Figure 4.1). The diversity and randomness of these features makes them less representative for the "dotted" features. In other words, the selection of the color and shape features based on keyword "dotted", is similar with random sampling over these features across the whole population.

As we can see from Figure 4.2 and Figure 4.3, there are still overlaps between the distributions of positive and negative samples. This indicates that there are items visually similar to the positive items on those "good" features, but they do not have the particular keyword (e.g. "dotted") in their descriptions. In the experimental results in chapter 5, we will show that iLike is able to retrieve such items without getting false hits (e.g. items with similar colors to the positive samples, but not the "dotted" texture).

The difference between two distributions could be quantitatively captured by running Kolmogorov-Smirnov test (K-S test) [Conover, 1998] across each dimension of feature vectors. The two sample K-S test is commonly used for comparing two data sets because it is nonparametric and does not make any assumption on the distribution. The null hypothesis for this test is that the two samples are drawn from the same distribution. For $n$ i.i.d samples $X_1, X_2, ...X_n$ with unknown distribution, an empirical distribution function can be defined as follows:

$$S_n(x) = \begin{cases} 0, \text{ if } x < X_{(1)} \\ \frac{k}{n}, \text{ if } X_{(k)} \leq x < X_{(k+1)}, k \in \{1, 2, ..., n-1\} \\ 1, \text{ if } x \geq X_{(n)} \end{cases}$$

where $X_{(1)}, X_{(2)}, \cdots X_{(n)}$ are ascending values. The K-S statistic for a given function $S(x)$ is

$$D_n = \max_x |S_n(x) - S(x)|$$

The cumulative distribution function $K$ of Kolmogorov distribution is

$$K(x) = 1 - 2\sum_{i=1}^{\infty}(-1)^{i-1}e^{-2i^2x^2} = \frac{2\pi}{x}\sum_{i=1}^{\infty}e^{-(2i-1)^2\pi^2/(8x^2)}.$$

It can be proved that $\sqrt{n}D_n = \sqrt{n}\max_x |S_n(x) - S(x)|$ will converge to the Kolmogorov distribution [Conover, 1998]. Therefore if $\sqrt{n}D_n > K_\alpha = Pr(K \leq K_\alpha) = 1 - \alpha$, the null hypothesis for the K-S test will be rejected at confidence level $\alpha$.

Similarly, to determine whether the distributions of two data sets differ significantly, the K-S statistic is

$$D_{n,m} = \max_x |S_n(x) - S_m(x)|$$

and the null hypothesis will be rejected at level $\alpha$ if

$$\sqrt{\frac{nm}{n+m}}D_{n,m} > K_\alpha \tag{4.1}$$

The P-value from the K-S test is used to measure the confidence of the comparison results against the null hypothesis. Back to our scenario, for each keyword, a P-value is calculated at each dimension of the feature vector. Features with lower P-values demonstrate statistically significant difference between positive and negative groups. For instance, The P-values for the features shown in Figure 4.2 are: 0, $3.901 \times 10^{-319}$, $2.611 \times 10^{-255}$, $5.281 \times 10^{-250}$; and for Figure 4.3 are: $2.103 \times 10^{-1}$, $1.539 \times 10^{-5}$, $8.693 \times 10^{-4}$, $1.882 \times 10^{-5}$. As we can see, items described by the keywords have significantly different values in those features, compared with items that are not described by the keyword. Therefore, such features are more likely to be coherent with visual meaning of the keyword, and hence more important to the human perception of the keyword. On the contrary, items with and without the keyword have statistically indistinguishable values on other visual features, showing that such features are irrelevant with the keyword.

In this way, we can use the inverted P-value of the K-S test as the weight of each visual feature for each keyword. Note that P-values are usually extremely small, so it is necessary to map the value to a reasonable scale before using it as weight. Ideally, the mapping function should satisfy the following requirements: (1) it should be a monotone decreasing function: lower P-values should give higher weight; (2) when the variable decreases under a threshold (conceptually, small enough to be determined as "statistically significant"), the

function value decreases slower. Therefore, we apply two steps of normalization. First, we designed a mapping function:

$$f(x) = \frac{\arctan(-log(x) - C) + \arctan(C)}{\pi}$$

where $C = (max(x) - min(x))/2$. It is then followed by a linear scaling to map the data range from to (0, 1), rendering itself as the weight vector of the keyword.

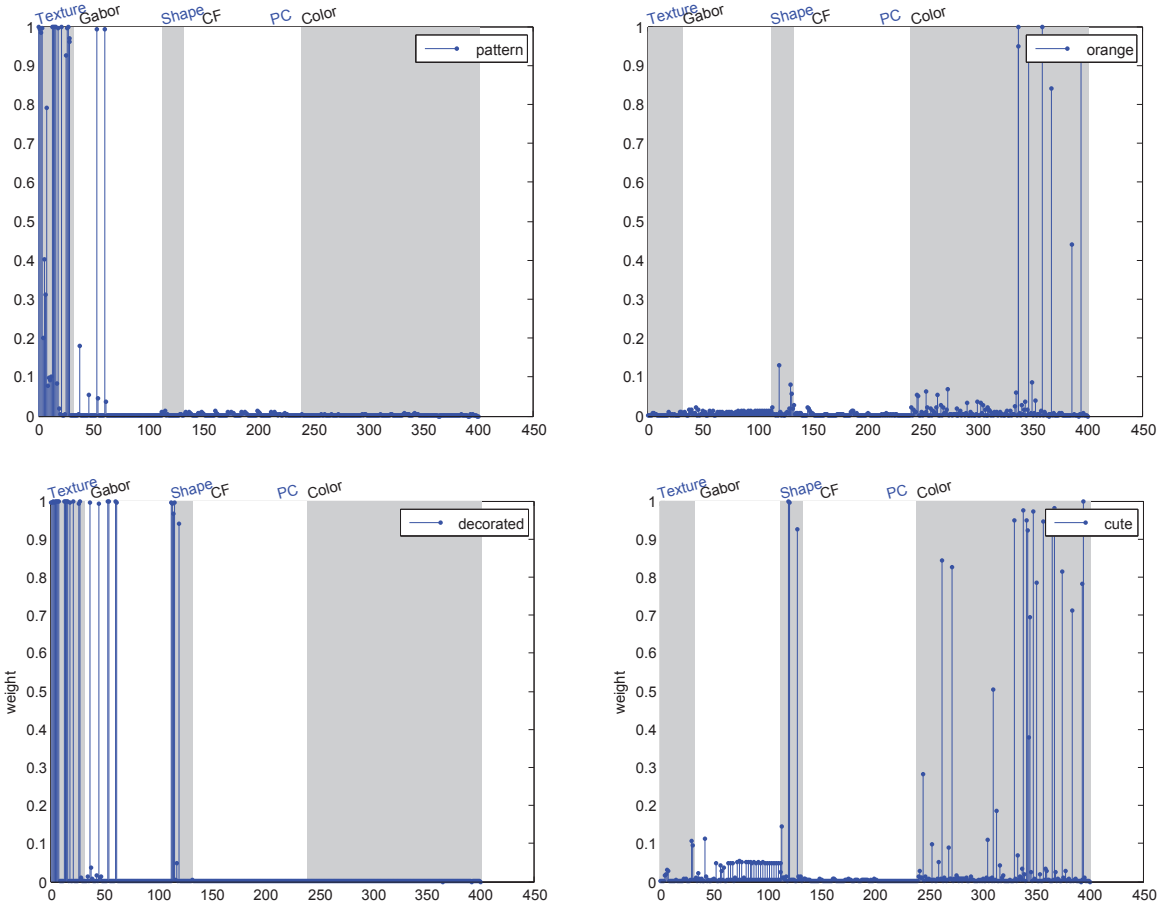



Figure 4.4: Weight vectors for terms "pattern" (upper left), "orange" (upper right), "decorated" (lower left), and "cute". (lower right)

By re-weighting visual features for each keyword, we amplified the features that are

significant for the keyword, while faded out the others. As an example, Figure 4.4 shows the normalized weight vectors computed from keywords "pattern", "orange", "decorated", and "cute". In the figure, the X axis represents visual features (as introduced in chapter 3): dimensions (1-32) are texture features: contrast, correlation, homogeneity, coarseness, direction, moment invariant etc.; (33-112) are texture features from the frequency domain: Gabor texture, Fourier descriptors, etc.; (113-239) are shape features: shape invariant moments, edge directions, moments of characteristic function and phase congruency; and (240-401) are color features: color moments and color histogram. Note that we group the visual features as above just for the convenience of discussion, and those groups of features might be overlapping with each other. In the figure, a large value (higher weight, lower P-value) are generated by statistically different positive and negative samples, indicating that the feature is more likely to have some kind of association with the human perception of the term. From the figures, we can see that some texture features show more significance in representing the keyword "pattern", while the visual features of keyword "orange" is primarily captured by color features. In this way, when user queries with term "pattern", we can infer that she is more interested in texture features, while local color and shape features are of less importance. Most importantly, we can further retrieve items with similar visual presentation in such features, but do not have the particular term ("pattern") in their descriptions.

On the other hand, it is difficult to imagine or describe the human visual perception for some keywords. Fortunately, our approach is still capable of assessing such perceptions.

For instance, Figure 4.4 also shows the weight vectors for terms "decorated" and "cute". It is not easy for a user to summarize the characteristics of "cute" items. However, when we look at the figure, the visual meaning is obvious. "Cute" items share some distinctive distributions in the color and shape features, while they are diversified in intensity and high frequency textual features.

## 4.3 Visual thesaurus

Thesauri are widely used in information retrieval, especially in linguistic preprocessing and query expansion. Although manually generated thesauri have higher quality, the developing process is very labor-intensive. Meanwhile, we can automatically generate thesauri using statistical analysis of textual corpora, based on co-occurrence or grammatical relations of terms. In iLike, we generate a different type of thesaurus – a *visual thesaurus*, based on the term distributions in the visual space, i.e., the statistical similarities of the visual representations of the terms.

In iLike, two terms are similar in terms of "visual semantics" if they are used to describe visually similar items. Since each term is used to describe many items, the similarity is assessed statistically across all the items described by both terms. In particular, the visual representation (mean vector) and weight vector for two terms $t_1$ and $t_2$ are denoted as $M$-dimensional vectors: $\vec{\mu}_{t_1}$, $\vec{\mu}_{t_2}$, $\vec{\omega}_{t1}$, $\vec{\omega}_{t_2}$, respectively. The similarity between $t_1$ and $t_2$ is

defined as the cosine similarity of two weighted mean vectors:

$$sim(t_1, t_2) = \frac{\sum_{i=1}^{M}(\mu_{t_1,i} \times \omega_{t_1,i}) \times (\mu_{t_2,i} \times \omega_{t_2,i})}{(\sum_{i=1}^{M} \mu_{t_1,i} \times \omega_{t_1,i}) \times (\sum_{i=1}^{M} \mu_{t_2,i}. \times \omega_{t_2,i})} \tag{4.2}$$

In this formula, each term vector (in visual feature space) is weighted by its weight vector, so that only values of statistically meaningful components are preserved. In this way, we are able to compute the semantic similarities between text terms, and such semantic similarities are coherent with human visual perception in this particular application domain. We also observe that some non-adjective terms demonstrate moderate similarity with many other terms. We eliminate the high frequency terms through post-processing. We are also able to compute antonyms, which are terms having a similar set of significant feature components but carrying consistently opposite values on such features, i.e. their weight vectors are similar, but weighted mean vectors are different.

Examples of synonyms and antonyms are shown in Figure 4.5. As we can see, weight vectors of terms "pale", "white" and "grey" are quite similar, indicating that they are related to a similar set of visual features in human perceptions (in this case, mostly color features). Meanwhile, the weighted mean vectors of "pale" and "white" are similar, while that of "grey" is very different.

We calculate the term-wise similarity across the dictionary, to generate a domain-specific "visual thesaurus" or a "visual WordNet". Some examples are shown in Table 4.1. This thesaurus could be used for query expansion for existing text-based product search engines, or in many other information retrieval applications.

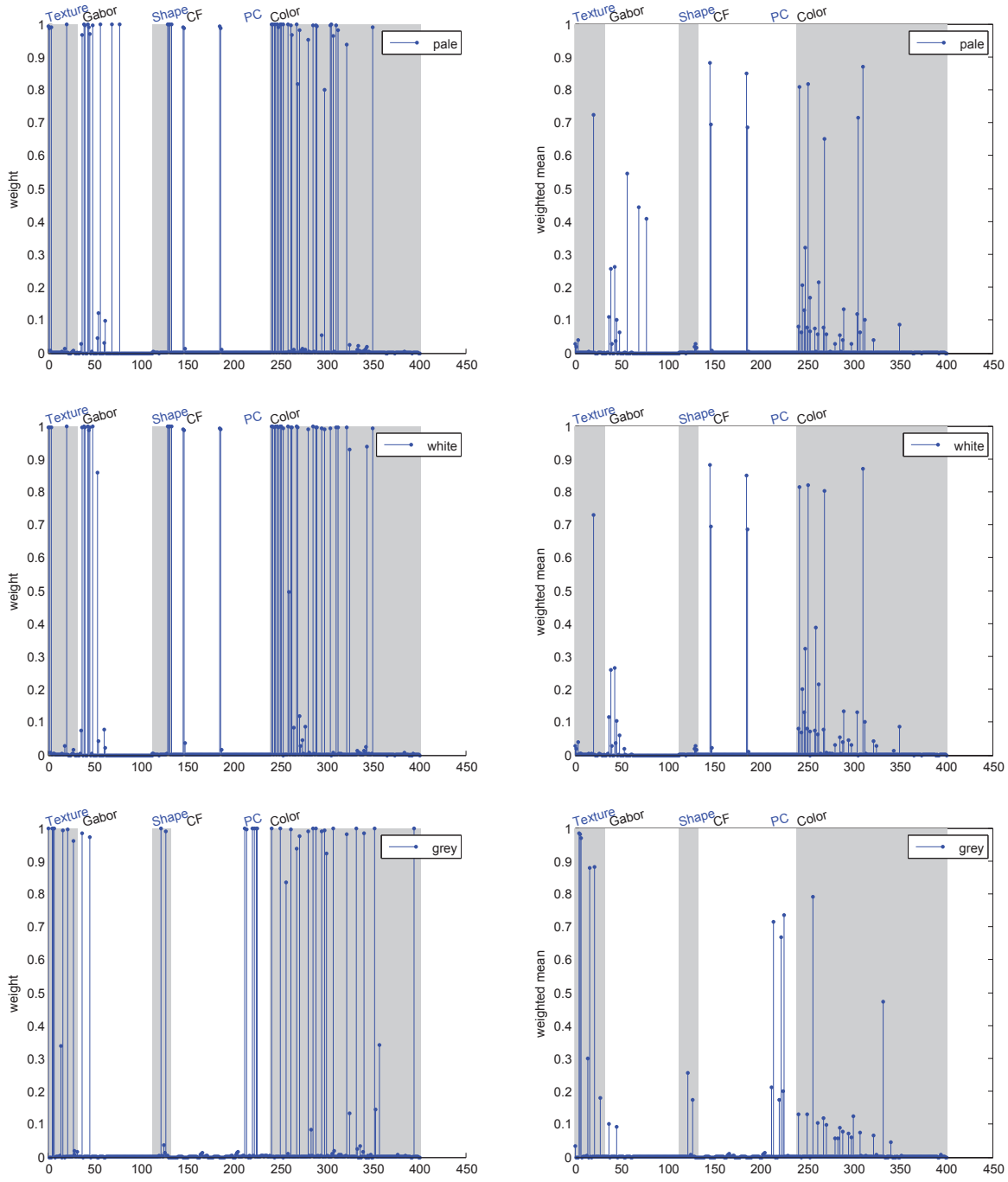



Figure 4.5: Weight vectors (left) and weighted mean vectors (right) for terms "pale", "white" and "grey" (from top to bottom).

Table 4.1: Visual thesaurus

| Words | First Few Words in Visual Thesaurus |
|---|---|
| swirl | whimsical, dots, button-down, motif, geometric, tree, mosaic, tropical |
| pattern | border, tropical, tree, geometric, lively, print, abstract, patchwork |
| designed | spirit, accented, texture, finish, frame, good, fashion, takes, ideal, framed |
| silky | cinched, scooped, softly, wraps, draping, elastane, fitted, fashioned, finely |
| necklace | sassy, impeccably, star, garter |
| lingerie | boyshort, garter, trunk, slimmer, canvas, ankles, dance, prom, impeccably |
| swimwear | beaches, rings, halter, beach, ring, sexier, created, glass, ocean, bottoms |
| stylishly | topstitching, grosgrain, level, tan, t-shirts, brown, lurex, sublimely |
| fits | detailing, adds, fabulous, clothing, wearing, designer, sister, nape, posh |
| shoes | shoe, outsole, kicks, slide, foot, strappy, sneakers, footbed, loafer |

## 4.4 Weight vector optimization

As we have introduced, product descriptions could be very subjective due to personal tastes. Different narrators/retailers may use different words to tag similar objects. Due to the existence of synonyms, we observe *false negatives* in the negative sets. A false negative is an item that: (1) is actually relevant to the term, (2) demonstrates similar visual features with the positive items, (3) is described by a synonym of the term, not the term itself, and hence is categorized in the negative set of the term. As shown in the "good" features in Figure **??**, we still observe overlaps in the feature value distribution of negative and positive samples. Such overlaps will reduce the weight of the corresponding feature towards any of the synonym terms, and possibly decrease search performance.

The domain-specific visual thesaurus can help us find both synonyms and antonyms. By merging items described by synonyms, we can decrease the number of false positive items caused by those synonyms, hence, we can observe higher consistency on significant features, and get higher weights out of it. In iLike, we first generating an initial visual thesaurus

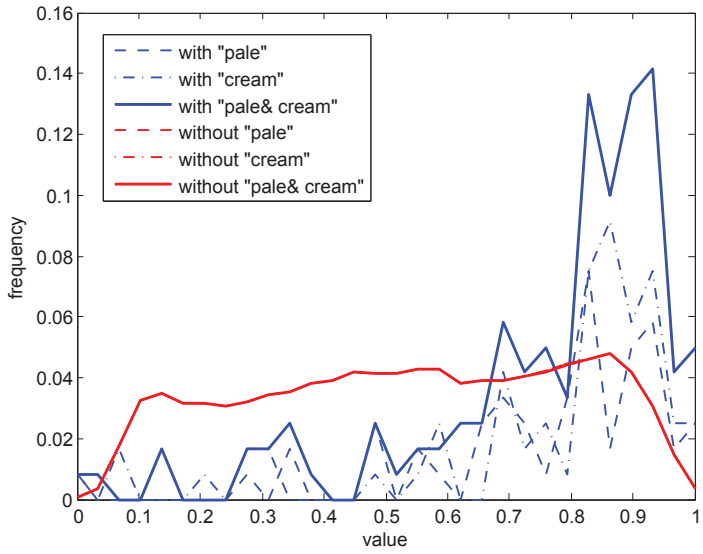



Figure 4.6: Sample feature distributions of antonyms.

for all the terms in the dictionary. Next, for each term, we add the items described by its top synonyms into its positive set. A high threshold is enforced in determining the top synonyms, so that we do not introduce false positives into the positive set. We re-calculate the new weight vector according to the updated positive/negative sets. An example of the value distribution (normalized) of a color feature of the positive and negative sets identified by terms "pale" and "cream" are shown in Figure 4.6 (dashed lines). The distribution of the positive and negative sets from the combined set are also shown. For demonstration, we normalized the distributions that the areas under each curve is 1. We can see that the feature distribution of the combined positive set is cleaner and narrower. By iteratively combining similar keywords in the visual thesaurus, we can improve the quality of the weight vectors. Our experiments have shown that the number of synonyms to be merged decreases significantly after each iteration.

## 4.5 Feature quality and correlation

In CBIR, the entropy of low-level visual features is widely used for feature selection and image annotation. While they are effective in some scenarios, such algorithms share one common disadvantage: the semantic gap. In iLike, we reemploy this problem by utilizing the entropy of feature weights across all keywords.

In section 4.2, we have generated a weight vector for each keyword, measuring the significance of each image feature dimension towards the keyword. Intuitively, a visual feature that is significant for a number of keywords is a "good" feature, while a visual feature that is in-significant for all keywords is a "bad" feature. Practically, we do not find any feature that is significant for (almost) all keywords. If such a feature existed, it would not be a good feature since it would not represent any distinctive visual meaning.

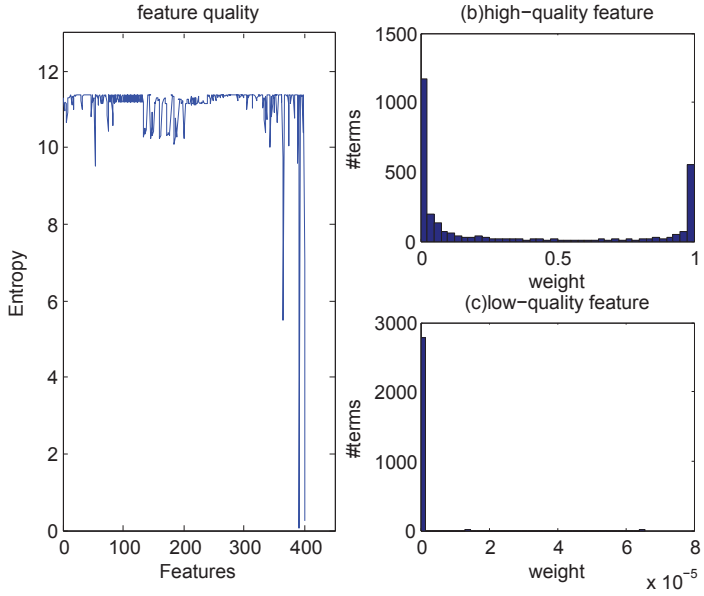



Figure 4.7: Feature quality.

In this way, for each feature, we collect weight values across all keywords (i.e. the $i$th component of all weight vectors). The entropy of each collection of weights is used as a quality assessment of the particular feature. The feature-quality curve is shown as Figure 4.7 (a). On the other hand, Figure 4.7(b) and (c) demonstrates the weight histogram for two difference features. As we can see, the feature shown in Fig 4.7(b) has higher weights for some terms, while the feature in Fig 4.7(c) has low weights for all terms. That is to say, the first feature is able to distinguish the positive and negative sets for some terms, while the other feature does not work well for any term. The first feature is certainly better than the other one. Figure 4.7 also shows that most of the selected features demonstrate good quality, except for a few color features (e.g. those with much lower entropy in Figure 4.7 (a)). This is consistent with the CBIR literature.

On the other hand, features may be correlated. In iLike, if two features are significant for a similar set of keywords, and insignificant for the others, they are somewhat correlated. To quantitatively study the correlations among the selected visual features, we calculated the pair-wise Pearson product-moment correlation coefficient (PMCC) for all the features, and the results are shown in Figure 4.8, in which black denotes maximum correlation, and white denotes no correlation. We can see that features are mostly independent, with moderate correlations among same type of features. We observe stronger correlations among CF and PC features. Such correlations introduce some computational overhead in iLike, but the impact on search precision is very limited.

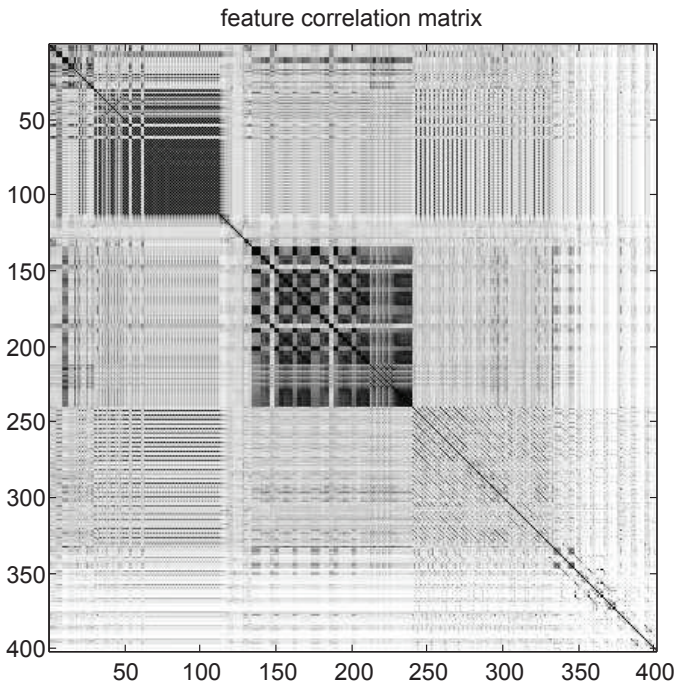



Figure 4.8: Feature correlation.

## 4.6 Query expansion and search

As we have introduced, in iLike, we first employ classic text-based search to obtain an initial set (since users could only provide text queries). For each keyword in the user query, the system loads its corresponding weight vector, which is generated off-line. Weight vectors from query terms are combined to construct the query weight vector $\vec{\omega}_Q$, which represents user intention in the visual feature space. For each item in the initial set, we use its visual features to construct a base query $\vec{q}_i$. We also obtain an expanded weight vector $\vec{\omega}_E$ from its textual description. Therefore, given a query q, the new query corresponding to the $i$-th item in the initial set is:

$$\vec{q'}(Item_i, Query) = \vec{q_i}. \times (\alpha \cdot \vec{\omega}_Q + \beta \cdot \vec{\omega}_E) \tag{4.3}$$

where $.\times$ indicates component-wise multiplication. Practically, $\beta$ is set to a much smaller value than $\alpha$, to highlight the intention from users. In the new query, features that are insignificant to the search terms carry very small values. Hence the new query could be used to search the item database on the basis of their Euclidean distances, without further enforcing the weights.

# Chapter 5. Experimental Results

## 5.1 Settings

We have implemented iLike on a database crawled from eight fashion shopping sites. We obtain a 401-dimensional visual feature vector from the main product image for each item. Both the visual and textual feature pre-processing are carried out on an off-line basis. For each user query, we calculate the initial result set based on text-based retrieval, and display in the title row of output. For each item in the initial set, we expand the user query with the textual and visual features from the item, and enforce the weight vector which infers user intention. The query expansion parameters $\alpha$, $\beta$ in Equation 4.3 are set to 0.9, 0.1, respectively. The search results using expanded and weighted query is displayed in columns, with the original item (from initial result set) in the title row.

To evaluate iLike, we use traditional Content-Based Image Retrieval approach as a baseline. The baseline approach employs the same visual features and product database as iLike does, with the only difference in that CBIR skips query expansion and feature weighting. Once the original image feature vector has been obtained, the baseline algorithm ranks the retrieved items according to the Euclidean distance between two vectors in the visual space.

## 5.2 Search examples

(a) (b) (c)

Figure 5.1: Search results for query "printed": (a) user selection from the initial set; (b) iLike query vector and the top 2 results; (c) Baseline (CBIR) query vector and the top 2 results;

Figure 5.1 shows an example of iLike and baseline results of query "printed". As shown in Figure 5.1(b), the iLike query highlights the features that are coherent with the search term "printed", and fades out features that are insignificant to the search term. We can see that the items retrieved by iLike share some local texture features (i.e., "printed" patterns). Meanwhile, although items in the CBIR result set (Figure 5.1(c)) are visually similar with the initial selection, they do not exhibit any relevance with the query term ("printed"), instead, local color and shape features dominates visual similarities. We can see that iLike successfully captures the user intention behind the search term, picks up a smaller subset of visual features that are significant to such intention, and yields better search results. Figure 5.2 shows more examples with different queries. For queries like "pattern" and "swirl", iLike identifies the local texture features in the frequency domain, as the relevant features, while for queries like "yellow" and "orange", the color features are identified as the more relevant ones. Thus it can be seen that iLike is capable of under-

standing the intention behind the query terms and is able to select relevant features that yield search results consistent with human perceptions.


Figure 5.2: Illustrative examples of search results.

Table 5.1: Name of similar items returned by iLike with keyword "printed"

| *Q* | **Black printed jersey caftan dress** | **Short sleeve printed tunic top** |
|---|---|---|
| *1* | Lemongrass floral silk v-neck dress | Flutter short sleeve striped tee |
| *2* | Navy printed silk racerback dress | Crochet scoopneck burnout tee |
| *3* | Black geometric printed sateen dress | Short sleeve plaid snap front |
| *4* | Lagoon sequined mesh racer dress | Printed mesh drop-waist bubble dress |
| *5* | Charcoal studded jersey panel dress | Short sleeve embroidered scoopneck top |
| *6* | Grey plaid belted ruffle shirt dress | Sleeveless Braided Scoopneck Top |

On the other hand, compared with text-based search, iLike significantly increases recall by yielding items that do not contain query terms in their textual descriptions. Table 5.1 shows two groups of item names returned by iLike with query "printed". Except for the initial results set (retrieved by text-based search), there are only 1 items that contains the query term in title or description fields. All other items are retrieved by content-based image search. Figure 5.3 provides a comparison of the search results returned by both

iLike and the baseline search methods for the query "ruffle shirt".

To sum up, most of the results demonstrate patterns that fits our perception of the query terms. Especially, (1) not all the returned items have the term in the descriptions; they are retrieved by visual features. (2) if we only use the visual features from initial result set (row 1) as the query, the results will drift away from user intention. Many other items has higher overall visual similarity with the items in the initial set. Thanks to the weighting approach, we are able to infer the implicit user intention behind the query term, pick up a smaller subset of visual features that are significant to such intention, and yield better results.

## 5.3 User study

To further evaluate iLike, we design and implement a user evaluation system to gather feedbacks of the performance of iLike and the baseline approach. The effectiveness of iLike by combining textual and visual features in product search is the focus of this study. We conduct iLike and baseline searches on the same data resources, and record user inputs as ground truth for each search.

First, 50 distinct adjective and noun keywords, which are commonly used to describe certain features of apparels, are employed to initiate different searches. These initiating keywords are evenly distributed in the visual feature space, i.e., describing texture, shape or color features. Next, 5 items from the initial result set (items from the title row) of each keyword are randomly selected as query images to be evaluated. The top 10 results from


Figure 5.3: (a) iLike search results for query "ruffle shirt"; (b) Baseline search results for query "ruffle shirt"

iLike and baseline for each query image, together with 20 randomly selected items from the same category, are saved for user evaluation. Therefore, for each query and seed item, we prepare up to 40 item images for evaluation.

Twenty participants from the University of Kansas are invited to evaluate the system. All participants have had experiences of using web search engines for several years, including online shopping experiences. Each of them is asked to evaluate at least five distinct queries. Specifically, for each query and seed image set, a participant is provided with the prepared item images (displayed in random order), and he/she is asked to mark items that he/she determines to be relevant to the query. Data from both server logs and user behavioral logs (time stamps) are recorded and analyzed for system evaluation.

Table 5.2: Overall performance of iLike and Baseline

| | **TP** | **TN** | **FP** | **Precision** | **Recall** |
|---|---|---|---|---|---|
| iLike | 574 | 406 | 353 | 0.59 | 0.62 |
| *baseline* | 476 | 504 | 451 | 0.48 | 0.51 |

Table 5.2 shows the statistics of comparing iLike with traditional CBIR. In the table, a true positive (TP) is a retrieved item (from iLike or CBIR) that is marked as relevant by the evaluators; a false positive is a retrieved item that is marked as irrelevant by the user. Figure 5.4 shows the average Precision-Recall Curve of iLike and CBIR. In the evaluation, 98 distinct queries are evaluated by 20 users. 927 of all the retrieved items have been identified by users as relevant, in which 574 are captured by iLike and 406 by the baseline approach. The overall precision and recall of iLike outperform CBIR by 21%, indicating that iLike is better at perceiving user's "visual" intentions behind search terms.

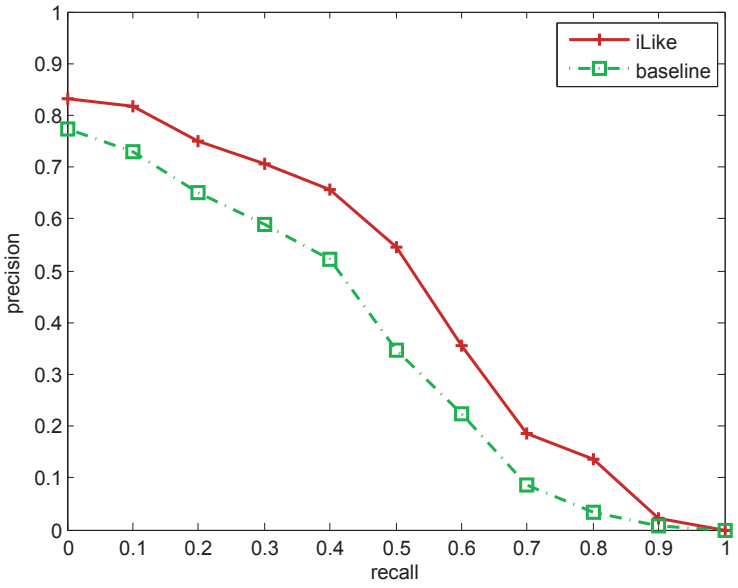
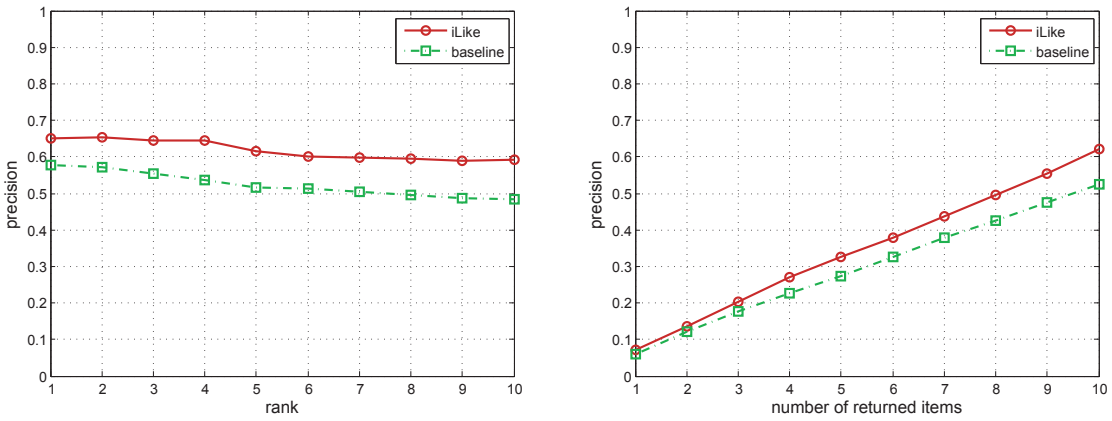



Figure 5.4: Average Precision-Recall Curve.

Figure 5.5 (a) and (b) shows the average precision and recall rate at different retrieval indexes. The statistical results illustrate a significant and stable difference between iLike and the baseline approach in both the number and coverage of the retrieved relevant items, which is in full support of the qualitative analysis in section 5.2.

To evaluate the robustness of iLike, we then compare iLike with the baseline approach across different queries. Figure 5.6 shows the $R$-Precision histograms for all the 98 distinct queries. An $R$-Precision histogram presents the differences between the precision of iLike and baseline at recall point $R$. A positive bar means that iLike outperforms the baseline approach on this query. From the 5-precision histogram (upper) we can see that iLike achieves better precision for the majority of queries; with $R$ increasing (lower), the precision of iLike still keeps on top of CBIR. These results are in agreement with Figure 5.5 (a). However, there are some queries where iLike performs worse than CBIR in both




(a) (b)

Figure 5.5: (a) precision-rank curve, (b) recall-rank curve.

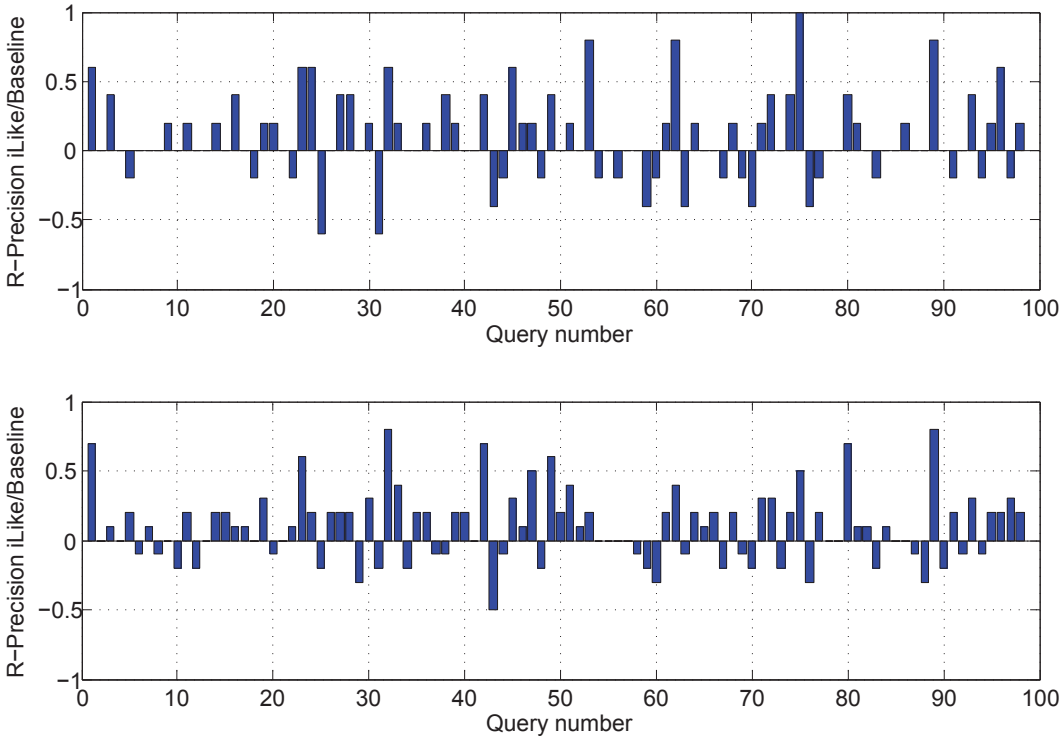



Figure 5.6: Precision Histogram $RP_{A/B}(i) = RP_A(i) - RP_B(i)$; Upper: R=5, Lower: R=10.

5-precision and 10-precision, i.e., query 25 ("voile"), 31 ("crinkle"), 43 ("metallic") and 76 ("polka"), etc. One common feature of those terms could be the complicated "visual" meaning, which is difficult for users to interpret. It is equally possible that users who participate in the evaluation do not fully understand such terms, and therefore choose "relevant" items without considering the keywords, making iLike fail in capturing these intentions.

# Chapter 6. Conclusion and Discussions

Content-based image retrieval, as a field, has grown tremendously in the last decade. Unfortunately, the research community still struggle to develop and implement scalable CBIR systems, which means that the core problems remain unsolved. In this thesis, we comprehensively survey, analyze and compare current progress of image retrieval in terms of the indexing techniques involved. We overview the image processing approaches for feature extraction. We also discuss significant challenges involved in the adaptation of existing image retrieval techniques that can be useful for bettering understanding user intentions.

This thesis proposed iLike, a vertical search engine for apparel shopping. The goal of the research, as discussed, is to explore the possibilities of bridging the semantic gap between the high-level semantic content and low-level visual features. Compared with existing research, we take a different approach that focuses on learning the association between textual and visual features from a very-large scale data set. With the extreme popularity of social media, we are able to collect a large image database with reasonable-quality labels.

We first extract both textual features and low-level image features from the collected data set to identify associations between visual features and textual features at the level of image vs. text. However, due to the existence of the semantic gap, such associations make little sense at object/region vs. term level. We then propose a statistical learning model to discover the inherent connections between the concepts behind textual terms with regions and low-level visual features from images, i.e. to map textual entities into visual

feature space. Particularly, we build a computational model with a text-guided weighting scheme to extract concepts from textual terms, and link them with features extracted from images. Such weighting scheme infers user intention from query terms, and enhances the visual features that are significant towards such intention.

Experimental results show that iLike is effective and capable of bridging the semantic gap. Through the comprehensive user study, iLike has demonstrated outstanding performance for a large number of descriptive terms. In some cases, it does not work well for some keywords (mostly non-adjectives). Many of such words have abstract meaning and are very unlikely to be included in user queries (e.g. zip, logo). To sum up, by combining textual and visual features, iLike manage to pick "good" features that reflect users' perception, and therefore is effective for vertical search.

# References


[Aslandogan et al., 1997] Aslandogan, Y. A., Thier, C., Yu, C. T., Zou, J., and Rishe, N. (1997). Using semantic contents and wordnet in image retrieval. In *SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 286–295.

[Barnard et al., 2003] Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D. M., and Jordan, M. I. (2003). Matching words and pictures. *J. Mach. Learn. Res.*, 3:1107–1135.

[Bouachir et al., 2009] Bouachir, W., Kardouchi, M., and Belacel, N. (2009). Improving bag of visual words image retrieval: A fuzzy weighting scheme for efficient indexation. In *Signal-Image Technology Internet-Based Systems (SITIS), 2009 Fifth International Conference on*, pages 215 –220.

[Cai et al., 2004] Cai, D., He, X., Li, Z., Ma, W.-Y., and Wen, J.-R. (2004). Hierarchical clustering of www image search results using visual, textual and link information. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 952–959.

[Chen et al., 2001] Chen, Z., Wenyin, L., Hu, C., Li, M., and Zhang, H.-J. (2001). ifind: a web image search engine. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, page 450.

[Conover, 1998] Conover, W. J. (1998). *Practical Nonparametric Statistics*. John Wiley & Sons.

[Cui et al., 2008a] Cui, J., Wen, F., and Tang, X. (2008a). Intentsearch: interactive online image search re-ranking. In *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, pages 997–998.

[Cui et al., 2008b] Cui, J., Wen, F., and Tang, X. (2008b). Real time google and live image search re-ranking. In *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, pages 729–732.

[Cui et al., 2007] Cui, J., Wen, F., Xiao, R., Tian, Y., and Tang, X. (2007). Easyalbum: an interactive photo annotation system based on face clustering and re-ranking. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 367–376.

[Datta et al., 2006a] Datta, R., Joshi, D., Li, J., James, and Wang, Z. (2006a). Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 39:2007.

[Datta et al., 2006b] Datta, R., Joshi, D., Li, J., James, and Wang, Z. (2006b). Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 39:2007.

[Deerwester et al., 1990] Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6):391–407.

[Deselaers et al., 2004] Deselaers, T., Keysers, D., and Ney, H. (2004). Features for image retrieval – a quantitative comparison. In *DAGM 2004*.

[Dua et al., 2007] Dua, J.-X., Wang, X.-F., and Zhang, G.-J. (2007). Leaf shape based plant species recognition. *Applied Mathematics and Computation*, 185(2):883–893.

[Dudani et al., 1977] Dudani, S. A., Breeding, K. J., and McGhee, R. B. (1977). Aircraft identification by moment invariants. *IEEE Trans. Computers*, 26(1):39–46.

[Esperanca and Samet, 1997] Esperanca, C. and Samet, H. (1997). A differential code for shape representation in image database applications. In *Image Processing, 1997. Proceedings., International Conference on*, volume 1, pages 556 –559 vol.1.

[Finlayson, 1996] Finlayson, G. (1996). Color in perspective. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(10):1034 –1038.

[Frankel et al., 1996] Frankel, C., Swain, M. J., and Athitsos, V. (1996). Webseer: An image search engine for the world wide web. Technical report, Chicago, IL, USA.

[Haralick et al., 1973] Haralick, R. M., Shanmugam, K., and Dinstein, I. (1973). Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, 3(6):610 –621.

[Huang et al., 1998] Huang, J., Kumar, S., Mitra, M., and Zhu, W.-J. (1998). Spatial color indexing and applications. In *Computer Vision, 1998. Sixth International Conference on*, pages 602 –607.

[Hughes, 1968] Hughes, G. (1968). On the mean accuracy of statistical pattern recognizers. *Information Theory, IEEE Transactions on*, 14(1):55 – 63.

[Huijsmans and Sebe, 2005] Huijsmans, D. P. and Sebe, N. (2005). How to complete performance graphs in content-based image retrieval: Add generality and normalize scope. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27:245–251.

[Jain et al., 2000] Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: A review. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 22(1):4–37.

[Jeon et al., 2003] Jeon, J., Lavrenko, V., and Manmatha, R. (2003). Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 119–126.

[Jiang et al., 2007] Jiang, Y.-G., Ngo, C.-W., and Yang, J. (2007). Towards optimal bag-of-features for object categorization and semantic video retrieval. In *Proceedings of ACM International Conference on Image and Video Retrieval*.

[Jing et al., 2006] Jing, F., Wang, C., Yao, Y., Deng, K., Zhang, L., and Ma, W.-Y. (2006). Igroup: web image search results clustering. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 377–384.

[Kakumanu et al., 2007] Kakumanu, P., Makrogiannis, S., and Bourbakis, N. (2007). A survey of skin-color modeling and detection methods. *Pattern Recogn.*, 40(3):1106–1122.

[Kennedy et al., 2006] Kennedy, L. S., Chang, S.-F., and Kozintsev, I. V. (2006). To search or to label?: predicting the performance of search-based automatic image classifiers. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, MIR '06, pages 249–258.

[Khotanzad and Hong, 1988] Khotanzad, A. and Hong, Y. H. (1988). Rotation invariant pattern recognition using zernike moments. In *Pattern Recognition, 1988., 9th International Conference on*, pages 326 –328 vol.1.

[Kogler and Lux, 2010] Kogler, M. and Lux, M. (2010). Bag of visual words revisited: an exploratory study on robust image retrieval exploiting fuzzy codebooks. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*, MDMKDD '10, pages 3:1–3:6.

[Kompatsiaris et al., 2001] Kompatsiaris, I., Triantafyllou, E., and Strintzis, M. (2001). A world wide web region-based image search engine. *Image Analysis and Processing, International Conference on*, 0:0392.

[Kovesi, 1999] Kovesi, P. (1999). Image features from phase congruency. *Journal of Computer Vision Research*, 1(3).

[Lazebnik et al., 2006] Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169 – 2178.

[Lempel and Soffer, 2001] Lempel, R. and Soffer, A. (2001). Picashow: pictorial authority search by hyperlinks on the web. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pages 438–448.

[Lew et al., 2006] Lew, M. S., Sebe, N., Djeraba, C., and Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19.

[Li and Wang, 2005] Li, J. and Wang, J. (2005). Alip: the automatic linguistic indexing of pictures system. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 1208 – 1209 vol. 2.

[Li and Wang, 2008] Li, J. and Wang, J. Z. (2008). Real-time computerized annotation of pictures. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30:985–1002.

[Li et al., 2006] Li, X., Chen, L., Zhang, L., Lin, F., and Ma, W.-Y. (2006). Image annotation by large-scale content-based image retrieval. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 607–610.

[Lieberman et al., 2001] Lieberman, H., Rozenweig, E., and Singh, P. (2001). Aria: An agent for annotating and retrieving images. *Computer*, 34(7):57–62.

[Luo et al., 2003] Luo, B., Wang, X., and Tang, X. (2003). A world wide web based image search engine using text and image content features. In *IS&T/SPIE Electronic Imaging, Internet Imaging IV*.

[Ma and Zhang, 1998] Ma, W.-Y. and Zhang, H.-J. (1998). Content-based image indexing and retrieval. *Handbook of multimedia computing*, pages 227–253.

[Manjunath et al., 2001] Manjunath, B., Ohm, J.-R., Vasudevan, V., and Yamada, A. (2001). Color and texture descriptors. *Circuits and Systems for Video Technology, IEEE Transactions on*, 11(6):703 –715.

[Mehtre et al., 1997] Mehtre, B. M., Kankanhalli, M. S., and Lee, W. F. (1997). Shape measures for content based image retrieval: a comparison. *Inf. Process. Manage.*, 33:319–337.

[Mojsilovic et al., 2000] Mojsilovic, A., Mojsilovi?, R., Kovacevic, J., Hu, J., Safranek, R. J., Member, S., Member, S., and Ganapathy, S. K. (2000). Matching and retrieval based on the vocabulary and grammar of color patterns. *IEEE Trans. Image Processing*, 9:38–54.

[Mukherjea et al., 1999] Mukherjea, S., Hirata, K., and Hara, Y. (1999). Amore: A world wide web image retrieval engine. *World Wide Web*, 2(3):115–132.

[Raimondo et al., 2009] Raimondo, S., Simone, S., Claudio, C., and Gianluigi, C. (2009). Prosemantic features for content-based image retrieval. In *7th International Workshop on Adaptive Multimedia Retrieval*.

[Rivlin and Weiss, 1995] Rivlin, E. and Weiss, I. (1995). Local invariants for recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 17(3):226 –238.

[Sawant et al., 2010] Sawant, N., Datta, R., Li, J., and Wang, J. Z. (2010). Quest for relevant tags using local interaction networks and visual content. In *Proceedings of the international conference on Multimedia information retrieval*, MIR '10, pages 231–240.

[Schmid and Mohr, 1997] Schmid, C. and Mohr, R. (1997). Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:530–535.

[Sclaroff et al., 1997] Sclaroff, S., Taycher, L., and Cascia, M. L. (1997). Imagerover: A content-based image browser for the world wide web. In *CAIVL '97: Proceedings of the 1997 Workshop on Content-Based Access of Image and Video Libraries (CBAIVL '97)*, page 2, Washington, DC, USA. IEEE Computer Society.

[Sharvit et al., 1998] Sharvit, D., Chan, J., Tek, H., and Kimia, B. B. (1998). Symmetry-based indexing of image databases. *J. Visual Communication and Image Representation*, 9:366–380.

[Shen et al., 2000] Shen, H. T., Ooi, B. C., and Tan, K.-L. (2000). Giving meanings to www images. In *Proceedings of the eighth ACM international conference on Multimedia*, MULTIMEDIA '00, pages 39–47.

[Smeulders et al., 2000a] Smeulders, A. W. M., Member, S., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000a). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1349–1380.

[Smeulders et al., 2000b] Smeulders, A. W. M., Member, S., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000b). Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1349–1380.

[Smith and fu Chang, 1996] Smith, J. R. and fu Chang, S. (1996). Automated binary texture feature sets for image retrieval. In *In Proc ICASSP-96*, pages 2239–2242.

[Stricker and Orengo, 1995] Stricker, M. and Orengo, M. (1995). *Similarity of Color Images*, volume 2, pages 381–392.

[Swain and Ballard, 1991] Swain, M. J. and Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision*, 7:11–32.

[Tamura et al., 1978] Tamura, H., Mori, S., and Yamawaki, T. (1978). Textural features corresponding to visual perception. *IEEE Trans. Systems, Man, and Cybernetics*, 8(6).

[Teague, 1980] Teague, M. R. (1980). Image analysis via the general theory of moments∗. *J. Opt. Soc. Am.*, 70(8):920–930.

[Tirilly et al., 2008] Tirilly, P., Claveau, V., and Gros, P. (2008). Language modeling for bag-of-visual words image categorization. In *CIVR'08*, pages 249–258.

[Turner, 1986] Turner, M. R. (1986). Texture discrimination by gabor functions. *Biol. Cybern.*, 55:71–82.

[Vailaya et al., 2001] Vailaya, A., Figueiredo, M., Jain, A., and Zhang, H.-J. (2001). Image classification for content-based indexing. *Image Processing, IEEE Transactions on*, 10(1):117 –130.

[Vijay and Bhattacharya, 2009] Vijay, A. and Bhattacharya, M. (2009). Content-based medical image retrieval using the generic fourier descriptor with brightness. *Machine Vision, International Conference on*, 0:330–332.

[von Ahn and Dabbish, 2004] von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326.

[Wang et al., 2006a] Wang, J. Z., Boujemaa, N., Del Bimbo, A., Geman, D., Hauptmann, A. G., and Tesić, J. (2006a). Diversity in multimedia information retrieval research. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, MIR '06, pages 5–12.

[Wang et al., 2007] Wang, S., Jing, F., He, J., Du, Q., and Zhang, L. (2007). Igroup: presenting web image search results in semantic clusters. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 587–596.

[Wang et al., 2006b] Wang, X.-J., Zhang, L., Jing, F., and Ma, W.-Y. (2006b). Annosearch: Image auto-annotation by search. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 1483–1490, Washington, DC, USA. IEEE Computer Society.

[Wang et al., 2002] Wang, Z., Chi, Z., and Feng, D. (2002). Fuzzy integral for leaf image retrieval. In *Fuzzy Systems, 2002. FUZZ-IEEE'02. Proceedings of the 2002 IEEE International Conference on*, volume 1, pages 372–377.

[White and Jain, 1996] White, D. A. and Jain, R. (1996). Similarity indexing: Algorithms and performance. In *In Storage and Retrieval for Image and Video Databases (SPIE*, pages 62–73.

[Wu et al., 2009] Wu, L., Yang, L., Yu, N., and Hua, X.-S. (2009). Learning to tag. In *18th International World Wide Web Conference*, pages 361–361.

[Yan et al., 2007] Yan, R., Natsev, A., and Campbell, M. (2007). An efficient manual image annotation approach based on tagging and browsing. In *Workshop on multimedia information retrieval on The many faces of multimedia semantics*, MS '07, pages 13–20.

[Yang et al., 2007] Yang, J., Jiang, Y.-G., Hauptmann, A. G., and Ngo, C.-W. (2007). Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the international workshop on Workshop on multimedia information retrieval*, MIR '07, pages 197–206.

[Yee et al., 2003] Yee, K.-P., Swearingen, K., Li, K., and Hearst, M. (2003). Faceted metadata for image search and browsing. In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408.

[Zhang et al., 2006] Zhang, L., Chen, L., Jing, F., Deng, K., and Ma, W.-Y. (2006). Enjoyphoto: a vertical image search engine for enjoying high-quality photos. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 367–376.

[Zhang et al., 2004] Zhang, L., Hu, Y., Li, M., Ma, W., and Zhang, H. (2004). Efficient propagation for face annotation in family albums. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 716–723.

[Zhou and Dai, 2007] Zhou, Z.-H. and Dai, H.-B. (2007). Exploiting image contents in web search. In *IJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence*, pages 2928–2933, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.