On the Use of Visual Conceptual Information for the Indexing and Retrieval of Image Regions

by

Radi Jarrar

Thesis submitted

for fulfillment of the requirements for the degree of

Doctor of Philosophy

School of Information Technology

Monash University

2012

Copyright Notices

Notice 1

Notice 2

Contents

List of Tables

List of Figures

Abstract

To address the semantic gap, state-of-the-art automatic image annotation frameworks concatenate the low-level image features (such as color, texture, etc.) in high-dimensional spaces to learn a set of high-level semantic categories.

This research work investigates a multimedia indexing framework establishing a correspondence between visual conceptual information representing image regions and a set of high-level semantic concepts. The computational models for extracting the visual color, texture, and shape concepts utilized in this thesis aim to capture aspects related to human perception and understanding of the color, texture, and shape features of image regions.

Through the development of models that map the low-level features into a set of high-level symbolic descriptors, image regions are described in transparent and readable form. These symbolic descriptors, named in this thesis as the visual concepts, are then used to learn a set of high-level semantic categories to classify un-annotated image regions. A main contribution here is presenting a framework for characterizing visual shape concepts to describe the shapes of image regions.

The efficiency of using the visual conceptual information is demonstrated by conducting a comparison with a baseline framework operating on a set of low-level image features to learn the same set of semantic categories.

Using the visual concepts to describe image regions is shown to achieve promising results and outperform the baseline model.

The experimental results show that using the visual conceptual information to describe image contents achieves the goal of this thesis of narrowing the semantic gap between the low-level image features and high-level semantic categories.

Acknowledgements

As my PhD journey is coming to an end, I would like to thank the Almighty God for blessing me with the opportunity, power, strength, and health to conduct this PhD research and put together this dissertation. Al-hamdolellah.

During my PhD, I was extremely delighted of having great supervisors whom without this thesis would not have come to an end. I am immensely thankful to my supervisors, Prof Chris Messom, Dr. Thomas O'Daniel, and Dr. Mohammad Belkhatir.

Prof Chris, without his support, dedication, wisdom, and guidance, I would not have reached this stage. I have always appreciated his advices on the academic, professional, and personal levels as well. I am also grateful for him for giving me many opportunities that really benefited my academic experience.

Dr. Thomas for his kindness, guidance, advices, and tireless and endless support. I have learned a lot from him and enjoyed all discussions I had with him. I appreciate his time and comments on this thesis.

Dr. Mohammad for his guidance and support throughout the years, for starting out this project, and for giving me this chance at Monash before everything. And here, I would like to thank Bashar for introducing me to Dr. Belkhatir.

I am also grateful to have had a great administrative staff at our school who have been always great support to us. Many thanks to Lee Sock Wee, Siriyaten Ali, Jessica Yik, Theresa Wong, Jowey Lim, and Serena Liew. Thank you all for your great help in the previous years.

It was my real pleasure to have had great colleagues at Monash. Many thanks to all of you. I am specially grateful to Fariza Fauzi, Anushia Inthiran, and Bhawani

Selvaretnam for being such great friends. I'll always remember the supply of doughnuts during my confirmation. Also many thanks to Afkar Hellis for being always there a great friend and for all the fruitful coffee breaks. Many thanks for the amazing Habibi(s): Tan Choon Ling and Marc Cheong for their support, all the happy days, and all the laughter! I would like to thank Aminul Haque for his support and company during these years. And special thanks go to Mohammed Maree for being always there and for his appreciated support all over the years.

I could not be more thankful for having such amazing house-mates, very dear friends indeed, who have been incredibly supportive and understanding from moments of despair to moments of happiness: Muath Radwan, Yazeed Sleet, Khaled Majed, and Malik Hneif. You made it much easier to feel like home despite being so distant. Guys, those days are engraved in our memories forever. Happy ones. Definitely!

I could not have reached here without the support of my wonderful family and close friends. I am grateful to all my dear aunts, my noble uncles, and my amazing cousins for their endless support.

Last but not least, for those who I wouldn't be able to do anything without, my Mum, my sister, and my brother. I can't express in words how thankful I am for you. Without your love, support, and prayers, I wouldn't have progressed a step. I owe it all to you. And for Mohammed, I am sorry I couldn't attend your graduation. I will definitely make it to your higher degrees! :-)

To my Mother,

for endless love and sacrifice

&

in memories of my Father.

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

[redacted]

Radi Jarrar
June 29, 2012

Preface

Part of the material in Chapter 4 has appeared in the *IEEE International Conference on Image Processing (ICIP 2010)*:

Radi Jarrar and Mohammed Belkhatir, (2010). Towards automated conceptual shape-based characterization an application to symbolic image retrieval, Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, pp. 2673-2676, September 26-29, 2010, Hong-Kong, China

The contribution in Chapter 3 and part of the experiments in Chapter 6 of this thesis have been presented in the *IEEE International Conference on Image Processing (ICIP 2011)*:

Radi Jarrar, Mohammed Belkhatir, and Chris Messom (2011). On the use of conceptual information in a concept-based image indexing and retrieval framework, Proceedings of the IEEE International Conference on Image Processing (ICIP), IEEE, pp. 2441-2444, September 11-14, 2011, Brussels, Belgium

Key terms and definitions

Visual conceptual information[1]: a set of high-level symbolic descriptors that are used to describe the visual properties of images/image regions. It can be viewed as a natural way to represent image features that is derived from the human understanding of the visual properties of image objects.

Image region[2]: clusters of pixels in images that form entities that can be described or interpreted in terms.

Content-based image retrieval: a class of early image retrieval systems that rely on retrieving images from an image database using the image content (i.e., low-level image features) without using textual annotations to describe the content of images.

Low-level image features: the features that are extracted directly from digital representations of images and described in numeric representations without characterizing their semantic meaning.

Concept-based retrieval[3]: a class of image retrieval systems that aim at narrowing the semantic gap by capturing the semantic meanings conveyed in images to describe their contents using high-level descriptors based on the human perception.

[1] Visual conceptual information, Symbolic perceptual descriptors, and Visual concepts are used interchangeably in this thesis

[2] Image region and image object are used interchangeably in this thesis

[3] Concept-based retrieval and Semantic-based retrieval are used interchangeably in this thesis

Image retrieval systems in this class aim to annotate images with high-level descriptors. Images are then classified into semantic categories based on their associated annotations.

Semantic concepts: a set of text-based descriptors describing objects in an image or describing the entire scene. These descriptors, also known as image annotations, are assigned to images to facilitate further search for images from image database using natural language interaction rather than using the abstract low-level image features. The annotation might be at the region level of image objects or a set of keywords that describe the entire semantic scene of an image.

Semantic gap: the gap between the low-level features composing images and how human users understand the image contents. Images are indeed composed and matrices of numbers that represent the color of image pixels. These numbers reflect how images are stored in image datasets. Nonetheless, these representations do not reflect how human users comprehend image content. For example, human users translate image content into words that represent their contents such as an image of a beach, images of red flowers, etc.

Chapter 1

Introduction

Advances in technology such as digital cameras, camera-phones, webcams, scanners, storage media, and large online picture archives have led to a huge amount of professional and personal collections of digital images. Moreover, the advances on the web and photo sharing websites, such as Picasa[1] and Flickr[2], has significantly contributed to this leap. There is a need for effective and robust methods to index and categorize the huge amount of available visual data to facilitate and ease retrieval. To this end, many general-purpose image indexing and retrieval systems have been developed under two main categories: text-based and visual-based image retrieval systems.

Text-based image retrieval requires manual annotation of images by textual descriptors (Rui et al.; 1999). The associated textual annotations of images are used by the traditional database management systems to store image labels and facilitate later retrieval (Chang and Fu; 1980; Chang et al.; 1988). Inconsistency based on the variation of human interpretation among annotators and the time it consumes to annotate are the main disadvantages related to text-based image indexing.

[1] http://picasa.google.com/

[2] http://www.flickr.com/

The inherited problems of the text-based approach are addressed by introducing content-based image retrieval. Content-based indexing and retrieval of images refers to utilizing the visual contents of images to find similarities in patterns to classify/retrieve images. Advantages of the content-based approaches are being automated and generally fast. However, these approaches are still not able to characterize the semantic meanings conveyed in images. This problem is referred to as the 'semantic gap' (Smeulders et al.; 2000).

The semantic-based indexing and retrieval, also referred to as automatic image annotation, was introduced to tackle this issue by indexing and retrieving images with semantic concepts which characterize the image semantic meaning (e.g., an image composed of "houses", "trees", . . .). This approach makes it possible for human users to use text-based queries to retrieve images based on their visual content.

One of the issues related to automatic image annotation approaches is that they do not take into account the visual conceptual information of image content as humans comprehend and understand it. This is related to the fact that these systems tend to extract low-level image features to describe images for further classification. The extracted low-level features are far from the human understanding as they are represented in non-perceptual numeric values.

This thesis aims at addressing the semantic gap between the low-level image features and the humans' understanding of their visual properties by describing image regions using the visual conceptual information. The visual conceptual information refers to human readable symbolic descriptors that reflect the perceptual meaning of image objects. In this research work, the visual color, texture, and shape concepts are being used to describe image regions for the task of region-based classification. This method aims to map the low-level image features from their numeric representations to predefined sets of high-level color, texture, and shape categories that conform to the human perception. It can also be considered a quantization technique to quantize the low-level image features into a set of predefined categories.

In this thesis, a framework for the characterization and extraction of the visual shape concepts, where the shapes of image regions are described with high-level perceptual descriptors is presented. This framework is then integrated with models characterizing the visual color and texture concepts to describe and represent image regions. A set of machine learning algorithms is then evaluated on the visual concepts feature space to learn a set of high-level semantic categories to annotate un-labeled image regions. Experimentally, the theoretical proposition is evaluated on $15,000$ image regions from the IAPR TC-12 benchmark with an empirical comparison of six supervised machine learning algorithms.

This thesis answers the following research questions:

1. Can shapes of segmented image regions be represented using high-level perceptual descriptors rather than low-level shape features for the task of image description and classification?
2. Would describing image regions using the visual color, texture, and shape concepts outperform using a set of low-level image features and thus narrow the semantic gap?

This thesis presents the following contributions in the domain of region-based image indexing and automatic image annotation:

- Addressing the semantic gap between the low-level image features and the high-level human interpretation of their visual properties in describing image regions (Chapters 3 and 4 and evaluated in Chapter 6);
- Presenting a framework for the extraction and characterization of visual shape concepts (Chapter 4 with its evaluation in Chapter 6); and
- Proposing an automatic indexing framework that is based on the visual conceptual information to learn a set of high-level semantic categories to classify image regions (Chapters 3 and 5 and evaluated in Chapter 6).

1.1 Thesis structure

The rest of this thesis is organized as follows:

Chapter 2 - Literature review. This chapter reviews and presents previous work and research aspects in the areas of content-based image retrieval, computer vision, and automatic image annotation. It also reviews the main problem in content-based image retrieval and automatic image annotation,'the semantic gap,' and reviews several approaches to address this gap. The work presented in this thesis utilizes some techniques from this literature. This chapter gives the reader an overview of this field so as to highlight the main context of this thesis.

Chapter 3 - The visual conceptual information illustrates a main contribution of this thesis, which is using the visual conceptual information to describe and classify image regions. This chapter also reviews the frameworks adopted from the literature to characterize the visual color and texture concepts extracted from image regions.

Chapter 4 - A framework for the extraction and characterization of the conceptual information of shapes. This chapter presents another main contribution of this thesis, which is a framework for the characterization and extraction of visual shape concepts of image regions. Characterizing visual shape concepts has not been widely studied as has visual color and texture concepts. A framework for the extraction of visual shape concepts is detailed in this chapter.

Chapter 5 - Supervised learning algorithms for learning semantic categories. This chapter presents a theoretical overview of supervised machine learning algorithms that will be used in this thesis to learn image semantics using the visual conceptual information extracted from image regions. It also presents a notation and problem formulation necessary for the task of classifying image regions into high-level semantic categories using the visual conceptual information.

Chapter 6 - Experimental validation. This chapter presents all experiments carried out to illustrate the concepts and prove the assertions presented in the earlier chapters of this thesis. It starts with an overview of the implementation and experimental validation of the visual color and texture concepts' characterization framework. It then details the implementation of the framework for extracting the visual shape concepts of image regions. The semantic-based region classification using the visual conceptual information is then presented. It entails conducting empirical comparison between the supervised learning algorithms presented in Chapter 5 to classify image regions. Comparison with a baseline method is also presented for each learner. Lastly, an experiment on retrieval using visual concepts to reflect the perceptual properties of image regions is demonstrated.

Chapter 7 - Conclusion and future directions discusses the main findings of this thesis. It summarizes the main research contributions presented in this thesis and some suggestions for potential research work that could be carried out to expand this research work.

Chapter 2

Literature Review

This chapter reviews approaches, systems, and techniques in image indexing and retrieval. State-of-the-art research on image retrieval frameworks is discussed starting from the first class of image retrieval frameworks using manual text annotation, moving to semantic-based image indexing. The main obstacle for image retrieval systems, known as "the semantic gap," is illustrated and research work that aims to narrow this gap is reviewed.

2.1 Text-based image retrieval

The first class of image retrieval approaches is based on textual annotation of images. The textual annotations are keywords provided by human users to describe the image content. The annotations provided with images are then used by database management systems to allow text-based retrieval of images (Chang and Fu; 1980; Chang et al.; 1988).

This class of systems relies fully on human understanding of image content. Descriptions of image content (scenes, objects, etc.) may vary from one annotator to another, and this inconsistency among human annotators is the first drawback of this class of systems. Another key issue related to this class of systems is the labor factor and the time it consumes to annotate images. Annotating images manually is a cumbersome and expensive task especially for large-scale image collections (Siu and Zhang; 2003).

From the user side, these systems have been labelled Query-By-Keywords (QBK). QBK users use high-level semantic concepts to query images from the database. The queries are represented using textual keywords and terms. Again, the time consumed by manual annotation and subjectivity in choosing a set of "meaningful" keywords to be used for retrieval are limitations. Despite the fact that this method lacks an effective, consistent, and scalable indexing scheme, it remains in use because it is a relatively easy way to add image search to relational database management systems that were designed solely to handle text, and retrieval systems that are primarily document-based.

Current Web-based image retrieval systems using text-based search for multimedia data (such as Google[1] and Bing[2]) use the surrounding text extracted from HTML pages to index and retrieve images. The surrounding text includes the URL of the image, the image file name, the ALT tag of the image, the title of the webpage, and the text contextual to the image. While this technique addresses cost and scalability to some extent, the surrounding text does not necessarily provide relevant information about the visual content of the image, so depending on it solely is insufficient for successful indexing and retrieval.

2.2 Content-based image retrieval

At the most elemental level, a digital image can be described by the color and location of the pixels it is composed of. Text annotation relies on the human eye to detect patterns of color and contrast to make sense of an image, and natural language to

[1] http://image.google.com
[2] http://image.bing.com

describe these patterns along with whatever additional contextual information is available to the annotator.

Content-based image retrieval systems were introduced in the mid 1990s to overcome the problems associated with the text-based image retrieval mentioned above. In the last decade, a number of content-based image retrieval systems have been proposed such as IBMs QBIC (Flickner et al.; 1995), Visualseek (Smith and Chang; 1997), Photobook (Pentland et al.; 1996), FourEyes (Picard et al.; 1995), Chabot (Ogle and Stonebraker; 1995), MARS (Mehrotra et al.; 1997), Blobworld (Carson et al.; 2002), and Virage (Bach et al.; 1996). Extensive surveys for content-based image retrieval systems can be found in Lew et al. (2006); Veltkamp and Tanase (2002); Smeulders et al. (2000).

The indexing in content-based approaches relies on extracting a set of appropriate features describing the low-level content of images. These features are called low-level since they are extracted directly from digital representations of images and described in numeric representations without characterizing their semantic meaning. Feature extraction is a necessary pre-processing step for image retrieval. A more detailed discussion of the extraction process is deferred to Section 2.3 of this chapter and Chapter 3; for now it is sufficient to think of low-level features as typified by color, texture, and shape.

Content-based image retrieval approaches do not rely on symbolic representation or textual annotation of images. A human user can query through query-by-example (QBE) in which the user provides an image as a query to retrieve similar images, and query-by-canvas (QBC) in which the user draws an example that approximates the image he/she would like to retrieve.

In QBE approaches, users use images as queries so that the retrieval framework finds similarities with the images in the database. Approaches using the QBE technique include QBIC (Flickner et al.; 1995), Photobook (Pentland et al.; 1996), and PicToSeek (Gevers and Smeulders; 2000). And in the QBC technique, users draw a visual query using color, texture, and shape to approximate their queries. Image retrieval frameworks then find similarity between the image drawn and images in the database.

2.3 Visual descriptors and feature extraction

QBE and QBC depend solely on comparing low-level image features of the query image provided (or drawn) by users and the images in the database. The selection of image features and a measure of similarity is crucial in these approaches, since these will determine the query results.

Feature extraction and application of a vocabulary of visual descriptors is a pre-processing step for image retrieval. Image features aim to capture and describe the visual properties of images either globally for the entire image (such as color histogram) or locally for a small cluster of pixels (such as extracting color, texture, or shape features from image regions) (Datta; 2010).

2.3.1 Similarity measures

The images in the database are indexed as feature vectors in an N dimensional Euclidean space. To retrieve images from the indexed set, the retrieval system computes the distance between a feature vector of a query image and the feature vectors in the feature space. The smaller the distance, the higher the similarity to the query image.

The Euclidean distance is one of the most commonly used measures to calculate the distance between feature vectors. Minkowski distance is considered a generalization of the Euclidean, Manhattan, and Chebyshev distances (Hu et al.; 2008).

The Minkowski distance between two vectors $q = (x_1, x_2, ..., x_n) \in \mathbb{R}^n$ and $f = (y_1, y_2, ..., y_n) \in \mathbb{R}^n$ is defined as

$$d = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p} \tag{2.3.1}$$

The Manhattan distance is defined when $p = 1$ while the Euclidean distance is when $p = 2$. Chebyshev distance is obtained when p is approaching infinity.

Other distance measures that have been used to measure the distance between images in content-based image retrieval include Chi-square (χ^2), Earth Mover's Distance (EMD) (Rubner et al.; 2000), and cosine distance. Zhang and Lu (2003b), Eidenberger (2006), and Hu et al. (2008) review and present experimental evaluation of the most common distance measures used in image retrieval frameworks.

2.3.2 Color

Color is an important feature in image presentation and description. Color representation has been studied in many areas such as psychology, arts, and science. In computer vision and image retrieval, colors are used to describe image content by assigning the image pixels on a color space. Using the extracted color features, a query image can be then retrieved based on similar compositions of the color space.

A number of color spaces exist in the literature such as such RGB (Red, Green, Blue), HSV (Hue, Saturation, Value), CMYK (Cyan, Magenta, Yellow, Black), and YCbCr. Other color spaces that are close to human perception include HSV (Hue, Saturation, Value) and HMMD (González and Woods; 2008; Manjunath et al.; 2001).

The color descriptors are used to extract colors from images and represent them in their corresponding color spaces. A number of color features have been used to represent the low-level color features such color histogram (Swain and Ballard; 1991; Gong et al.; 1996), color correlogram (Huang et al.; 1997), color moments

(Stricker and Orengo; 1995), and the MPEG-7 standards color descriptors such as the dominant color descriptors, color layout descriptors, scalable color descriptors, and color structure descriptors (Manjunath et al.; 2002).

A color histogram is a simple and efficient way to represent the distribution of colors in an image region. It represents the number of pixels having colors in each fixed list of color ranges of a color space. The color correlogram was presented to tackle the issue of discarding the spatial information of color location and pixel relation to the pixels as in color histogram and color moments. Color correlogram records the distances of each pair of colors in an image to represent their spatial correlation (Huang et al.; 1997). It is represented as a table of color pairs where the $n-$th entry for a color pair (x, y) specifies the probabilistic estimation of having a pixel of color x at a distance n form a pixel y as specified by Huang et al. (1997).

These color descriptors represent the mechanics of colors rather than representing them in a conceptual manner as the human users comprehend and understand.

2.3.3 Texture

Another important feature to describe image content is the texture features. Texture can be viewed as capturing and describing the repeating patterns in images (Petrou and Sevilla; 2006). The study of texture in the field of computer vision has led to the identification of several descriptors (Nixon and Aguado; 2008).

Texture features can be broadly classified into statistical and spectral approaches. Statistical approaches aim at characterizing the local statistical properties to describe textures such as the Co-occurrences matrix (Haralick et al.; 1973) and Tamura features (Tamura et al.; 1978). The statistical approaches are considered to be compact and robust to describe texture features. For example, Tamura features are scale and orientation invariant.

The other category, spectral feature extraction techniques, involves transforming images into frequency domain and then calculating the features from the transformed images. These include Fourier transform (Petrou and Sevilla; 2006), Gabor filters (Manjunath and Ma; 1996), and Wavelet transform (Wang et al.; 2001).

Texture descriptors such as Gabor and Wavelet are widely used in image retrieval systems. Experiments studying human vision have shown that these descriptors match with human vision and capture aspects related to human perception (Manjunath et al.; 2002; Wang et al.; 2001; Manjunath and Ma; 1996; Liu, Zhang, Lu and Ma; 2007). Moreover, Manjunath and Ma (1996) have shown that Gabor filters outperform other texture feature extraction techniques such as the wavelet transform. Gabor filters have the advantage that texture features are captured in multiple orientations and scales.

The MPEG-7 adopted three texture descriptors, namely: homogeneous texture descriptor (HTD), texture browsing descriptors (TBD), and edge histogram descriptor (EHD) (Manjunath et al.; 2002).

Representing images using only texture features is not discriminative enough since texture descriptors are computed over gray-scaled images and visual color information is not taken into account. Thus, images of similar textures but different colors, which properly belong to different semantic classes, would be considered similar by considering the texture descriptors alone. Furthermore, these texture feature extraction techniques concentrate at the low-level texture feature computation and extraction from images and do not consider the symbolic representation and the human understanding of the texture features.

2.3.4 Shape

Various low-level shape feature extraction techniques exist in the literature (Nixon and Aguado; 2008). Basically, they are classified under two main categories: contour-based and region-based shape descriptors.

Dealing with shapes of image objects is not a straightforward task. To identify shapes of objects that reside in images, region identification processes are required. This includes applying edge detection tools or segmentation algorithms to extract the regions of interest in images. Having segmented an image, shape features can be extracted from segmented regions. However, the task of detecting shapes is difficult to completely automate.

Shape descriptors aim to extract and describe shapes in a way that is consistent with human perception. For this, human intervention to accurately characterize shapes in images is required, in particular, in fields where accurate detection of shapes is critical such as medical imaging.

Shape feature extraction techniques are typically divided into two categories; contour-based and region-based shape descriptors. The contour-based shape descriptors use only the boundary information to represent shapes and ignore the shape interior content. Examples of contour-based shape descriptors include Fourier descriptors (Zahn and Roskies; 1972), Polygonal approximation (Gu; 1995), and Curvature scale space descriptors (Abbasi et al.; 1999). On the other hand, region-based shape descriptors exploit both boundary and interior information of the shape. All pixels within the region are taken into account to obtain a shape representation. They are robust to noise and shape distortion. Common region-based descriptors exist in the literature such as Zernike moments (Teague; 1980), Generic Fourier Descriptors (Zhang and Lu; 2002b), and Grid representation (Lu and Sajjanhar; 1999). Zhang and Lu (2004, 2001); Yang et al. (2008) provide recent in-depth reviews for low-level shape feature extraction algorithms and techniques.

2.4 Image segmentation

Image segmentation aims to extract image regions of objects that reside in images. It is an important pre-processing step for image content analysis, object detection, and object tracking systems. The main aim of image segmentation is to allow access at the region level to extract local image features rather than extracting global features from the whole image. It can be considered the first pre-processing step towards extracting image features in region-based image retrieval systems.

Having an accurate and semantically meaningful image segmentation is still a challenging task. There is a gap between how humans characterize objects in images and how segmentation algorithms characterize and extract them from images.

A number of segmentation techniques exist in the literature that can be roughly categorized as clustering based techniques (Yang et al.; 2009), grid based techniques (Lim et al.; 2003), edge detection techniques (Chan and Vese; 2001), graph based techniques (Felzenszwalb and Huttenlocher; 2004), and region based techniques such region shrinking, growing, splitting, and merging (Tremeau and Borel; 1997; Deng and Manjunath; 2001).

The grid based technique is considered a simple method that divides images into blocks for image feature extraction (Zhang et al.; 2012). Vailaya et al. (2001); Lim et al. (2003); Qi and Han (2007) are examples of systems that use grid based segmentation.

Clustering based technique can be considered a relatively simple segmentation technique. The main idea of clustering based segmentation is to cluster image pixels into separate groups. The simplest implementation of clustering based segmentation is to cluster image pixels using their extracted color values in some color space. This works well on homogeneous color images, which is not the case of natural images that are rich in colors and texture patterns (Deng and Manjunath; 2001). A segmentation technique to address this problem is presented in Deng and Manjunath (2001). This segmentation algorithm aims to segment images and videos into homogeneous color-texture regions. They separate the segmentation process into two independent process: color quantization and spatial segmentation of images.

Graph-based segmentation represents the segmentation problem in terms of a graph in which the image pixels are nodes of a graph and the segmentation algorithm is to find the most accurate edges between nodes in the graph. Weights on edges are computed based on some property of the pixels it connects such as their intensities (Felzenszwalb and Huttenlocher; 2004).

Region-based techniques aim to find homogeneous characteristics of pixels in images (Zhang et al.; 2008). Pixels with similar patterns are then grouped to from regions in images. Region-based segmentation algorithms are broadly categorized as region growing/shrinking methods and split-and-merge methods (Zhang et al.; 2008). While region-based methods depend on the similarities between neighbor pixels, edge-based methods, also known as boundary-based methods, are based on some discontinuity property of pixel values (Freixenet et al.; 2002). This category is based on boundary detection algorithms, such as Canny edge detection (González and Woods; 2008), to characterize boundaries of objects which can be viewed as boundaries of image regions. A detailed review and experimental comparison on region and edge-based segmentation techniques can be found in Freixenet et al. (2002).

Some of the prominent segmentation algorithms in the literature include the normalized cuts (Shi and Malik; 2000), statistical region merging (Nock and Nielsen; 2004), local variation algorithm (Felzenszwalb and Huttenlocher; 1998), JSEG (Deng and Manjunath; 2001), Blobworld (Carson et al.; 2002), and the mean-shift algorithm (Comaniciu and Meer; 2002).

Evaluating image segmentation algorithms has attracted a considerable amount of research effort in the recent years due to the importance of image segmentation.

Estrada and Jepson (2009) conduct experimental comparison between four edge-based segmentation algorithms. They use human segmentations from the Berkeley segmentation database (Martin et al.; 2001) as ground truth to evaluate the segmentation quality of the compared algorithms. They have shown that the Spectral Embedding MinCut (Estrada and Jepson; 2005) outperforms other segmentation algorithms such as mean-shift (Comaniciu and Meer; 2002), normalized-cut (Shi and Malik; 2000), and local variation algorithms (Felzenszwalb and Huttenlocher; 1998).

Ge et al. (2006) present a benchmark for evaluating image segmentation. They define image segmentation as the process of extracting the single most perceptually salient structure from an image. This benchmark is experimentally evaluated on six image segmentation methods. Their study concludes by stating that there is no particular good segmentation algorithm to segment individual objects from images.

Detailed reviews on image segmentation and its evaluation can be found in Zhang et al. (2008); Estrada and Jepson (2009); Ge et al. (2006); Freixenet et al. (2002); Unnikrishnan et al. (2007).

In the context of this thesis, a region-based image indexing framework is presented using the visual conceptual information. The problem of image segmentation and the impact of segmentation accuracy on the performance of image retrieval are out of the scope of this work. The benchmark used to evaluate this framework, which is the segmented and annotated IAPR TC-12 (Escalante et al.; 2010), provides all images with their regions segmented manually rather than using state-of-the-art image segmentation techniques. Since the main contribution of this thesis is to use the visual conceptual information to describe and classify image regions, this dataset was selected as images are segmented manually by human users. The manual segmentation of image regions reflects the understanding of the human user of the boundaries of image objects (regions) reside in images. Accordingly, this dataset was selected to be used in the experiments conducted in this thesis especially a framework for characterizing the visual shape concepts of the segmented image regions is presented in this thesis.

2.5 The semantic gap

Retrieving images based on their image features has indeed overcome the problems of the manual annotation of images in the early text-based approaches. Nonetheless, content-based image retrieval systems that utilize low-level features for retrieval do not take into account the semantic characteristics of an image region or the semantic scene conveyed in the image. Evaluation of systems using techniques such as confusion matrix, precision and recall, and F-scores (cf. Section 6.1.1) have shown that there is a significant difference between the high-level concepts that users normally use for their query through natural language interaction and the image content described by the low-level features. In other words, there is a gap between content description based on low-level features and the human understanding of image content. This gap is known as the semantic gap.

Smeulders et al. (2000) describe the semantic gap as follows:

> *The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation*

This gap describes the lack of correspondence between the low-level features extracted from images and the semantics conveyed in images. The semantic gap implies that it is difficult to automatically characterize semantic content of an image through the extracted low-level features. For example, if the similarity between two images is measured based on their color distributions, then entities such as an orange and a basketball could be considered the same since they have a similar color distribution. This explains the unsatisfactory results between human users expectations from an image retrieval systems and the results returned. There is a huge gap between the image description techniques and the richness of human interpretation of image content (Zhang et al.; 2001).

The semantic gap is one of the biggest hurdles faced by the content-based retrieval and computer vision research communities. A tremendous amount of research has been carried out to narrow this gap but the issue is still far from being resolved.

2.6 Semantic-based image retrieval

In order to address the semantic gap, and overcome the aforementioned problems associated with the manual annotation, a new class of image retrieval systems has been introduced. Semantic-based retrieval, also called concept-based retrieval, aims to capture the semantic meanings conveyed in images and describe them using high-level descriptors based on human perception (e.g., describing an image with "sky", "trees", . . .).

The goal of these systems is to automatically annotate image(s) with high-level semantic concepts, and classify images into semantic classes (Smeulders et al.; 2000). It is also referred as Automatic Image Annotation (AIA), in which text-based descriptors characterizing objects in an image are assigned to that image to facilitate further search for images from the image database. The annotation might be at the region level of image objects or a set of keywords that describe the semantic scene of the whole image.

The aim of automatic image annotation is to find a correlation between the pixels creating an image with a relevant set of keywords reflecting the semantic meaning of objects/images as humans comprehend and understand them. Thus, the process of annotating an image can be seen as a process of translating image content from a numerical representation into high-level descriptors reflecting its visual content

(Duygulu et al.; 2002). As a result of the annotation, users can query images using the symbolic descriptors.

Semantic-based indexing approaches entail analyzing the image/region content to find the most accurate high-level descriptor to describe the image or image region. In this manner, the low-level image contents in their numeric representation are mapped to symbolic descriptions understood by users. Human users can then use symbolic concepts to query images. In this manner, the gap between human understanding of images and their representation is narrowed. As mentioned previously, feature extraction is a backbone and the main pre-processing step of any image retrieval system. Selecting an adequate set of image features is as necessary as selecting appropriate machine learning algorithm that fits well with a certain problem. Section 2.3 of this chapter has presented low-level color, texture, and shape features used in image classification and retrieval.

Semantic-based retrieval is still a challenging task, which has been approached using a number of methods and techniques. Liu, Zhang, Lu and Ma (2007) present in their extensive survey five main methods used in semantic-based retrieval to overcome the semantic gap: (1) using relevance feedback in which users are involved in deciding the relevant/irrelevant images in the retrieval process; (2) using some domain knowledge to define ontologies of high-level concepts for image annotation; (3) making use of machine learning tools to correlate a set of extracted features with high-level descriptors; (4) using semantic templates to support high-level image retrieval; and (5) using surrounding text extracted from web-pages as well as the visual content of images on the web for retrieval. The following sections provide an overview of each of these methods.

2.6.1 Relevance feedback

Relevance feedback has been widely studied to narrow down the semantic gap in image retrieval frameworks (Zhou and Huang; 2003). It is an on-line process that aims to learn the intention of users by taking into account their feedback on the results returned by the retrieval framework. It was first introduced in text document retrieval frameworks (Zhou and Huang; 2003). This idea then attracted researchers to apply it to image retrieval, in particular early content-based retrieval approaches in the 1990s.

The main idea is to narrow the semantic gap between the query image and the images retrieved by taking into account the users thinking and understanding. In the relevance feedback scheme, users annotate images retrieved as relevant or irrelevant to their query. Accordingly, retrieval precision is refined based on the human perception.

The aggregated information from end-users is used for continuous learning to enhance the results for following queries. It was shown that incorporating relevance feedback in image retrieval helps to dramatically boost retrieval performance (Zhou and Huang; 2003).

Zhou and Huang (2003) provide a typical scenario for relevance feedback in content-based image retrieval systems:

1. The initial retrieved images are shown by the system in response to a user's query using query-by-example, query-by-sketch, keyword, . . . , etc.;
2. User annotate retrieved images as relevant or irrelevant (positive or negative) to their queries; and
3. Machine learning algorithm learns the feedback provided by users. Return to step 2. The system keeps repeating steps 2 and 3 to enhance the retrieval results of the framework.

Various machine learning algorithms have been employed for learning users feedback such as support vector machines (Zhang et al.; 2001; Tong and Chang; 2001; Jing et al.; 2004), decision trees (MacArthur et al.; 2000), and Bayesian learning (Vasconcelos and Lippman; 2000a; Su et al.; 2003).

To illustrate the research interest, Rui et al. (1997) convert the image feature vectors into weighted term vectors so that they can apply the relevance feedback technique. Zhang et al. (2001) use support vector machines classifier to learn from training data of relevant/irrelevant images marked by users. Han et al. (2004) present a framework in which they learn explicit and implicit semantics through a feedback log that records users feedback information over time. Chatzichristofis et al. (2010) improve the retrieval scores by presenting an automatic relevance feedback method. Zhou and Huang (2003); Crucianu et al. (2004) present extensive reviews for relevance feedback and its use in content-based image retrieval.

Though relevance feedback has proven to narrow the semantic gap by taking into account the human users' opinion of the retrieved documents, the processes of relevance feedback itself is considered to be time consuming as human users have to annotate the set of returned images as relevant/irrelevant overtime.

2.6.2 Incorporating semantic hierarchies of concepts organized in ontological knowledge base

The use of ontologies comprising semantic hierarchies of objects has been used to improve the image annotation process (Hyvönen et al.; 2003; Mezaris et al.; 2003; Styrman; 2005; Liu, Zhang, Lu and Ma; 2007).

Developers of many frameworks have studied the effectiveness of incorporating ontologies as explicit representations of background knowledge. These frameworks aim at modeling the contextual knowledge for image understanding and analysis. The contextual knowledge helps, in particular, in the task of object recognition within scenes by providing predictions about objects that are most likely to appear in a specific context (Bannour and Hudelot; 2011), such as object co-occurrences (i.e., topological information) and the spatial information of object locations in an image.

Ontologies take advantage of the maturity and advances in text retrieval such as query expansion, index access, etc. to enhance the retrieval accuracy. Bannour and

Hudelot (2011) list the targets of using ontologies within image retrieval frameworks as:

1. A standardized unified description of the low-level image features;
2. Visual description ontology to represent the relations among image features;
3. Knowledge-base to represent the relations between high-level semantic concepts; and
4. Semantic mapping between the visual level and the high-level semantic level.

2.6.3 Using machine learning tools to infer semantics

Machine learning techniques have proven powerful when employed to help improve the performance of various multimedia applications (Liu, Zhang, Lu and Ma; 2007; Datta et al.; 2008; Zhang et al.; 2012).

Machine learning algorithms such as support vector machines, decision tree, nearest neighbor, etc. have been widely used for classifying images into semantic categories. The main task of the learner is to establish correspondence between the image features and the high-level semantic categories.

Machine learning tools used might be supervised such as support vector machines or unsupervised such as clustering techniques. Comprehensive surveys in this field can be found in Liu, Zhang, Lu and Ma (2007); Datta et al. (2008); Zhang et al. (2012). Moreover, experimental comparison of various machine learning algorithms on classifying image regions is carried out in (Escalante et al.; 2010). Chapter 5 details the theoretical formalism of the machine learning algorithms used in the experimental chapter (i.e., Chapter 6) as well as review of semantic-based frameworks and image classification systems that use these algorithms.

2.6.4 Semantic template

Semantic template is another approach to narrow the semantic gap be defining representative features for semantic categories. The representative features are defined by calculating them from sets of images in each semantic category (Liu, Zhang, Lu and Ma; 2007). Each semantic category is then represented by its semantic template, and classification models are built using the obtained semantic templates to classify new input images into high-level semantic categories.

Liu et al. (2008) discretize color and texture features from decision trees by converting their low-level features into semantic templates. The semantic templates are constructed by computing the mean of the color and texture features extracted from sample image regions for each semantic category. The semantic templates are then used to construct the decision tree to classify new input image regions.

In Zhuang et al. (1999) a relevance feedback process is used to generate semantic templates. A user submits a query image associated with a keyword describing that image. The system returns the most relevant images and asks for the user's feedback. After some iterations, the centroid of the most relevant images is computed to represent the class associated with the keyword the user queried.

The issue with semantic templates is that they are not perceptually indicative as they do not record the perceptual features of image content as they are computed from the low-level features, such as the centroid of features, to establish template features of semantic categories.

2.6.5 Surrounding text for web-based image retrieval

Using the contextual information associated with images is mainly used in web-based image retrieval. As mentioned in Section 2.1, web-based image search engines such as Google[3] and Bing[4] use the surrounding contextual information extracted from web pages such as image file names and image captions to retrieve images in response to a user's query.

The research area of using the surrounding text to enhance retrieval of images has attracted many researchers to enhance the web-retrieval by extracting a rich semantic representation from the surrounding text and incorporating text-based and visual-based techniques to empower the classification and retrieval of images from the web. Shen et al. (2000) use a set of basic information associated with images: image file name, image caption, the HTML ALT tag, and the page title. They classify images into semantic categories using the surrounding text and a relevance feedback process.

Kherfi et al. (2004) survey techniques and methods used in web-based image retrieval systems. They state that the presence of noisy information in web-pages is a major obstacle in using the surrounding contextual information. Zhou and Dai (2007) present a web-based image search engine that makes use of the surrounding text. First, candidate images are returned based on their associated text information. Images relevant to the user query are then identified by analyzing the visual contents of the returned images. The final search result is generated by combining an image-based rank of the web pages with a keyword-based rank of the web pages. A model for integrating surrounding text information with the visual features of images is presented in Chen et al. (2001).

Clinchant et al. (2011) present a new information fusion technique, namely semantic combination, for efficiently fusing text and visual image information. More recently, Fauzi (2012) presents a multifaceted concept-based indexing framework that extracts the surrounding image information, analyzes its semantics, and classifies it into five facets. The relationships between the faceted concepts are modeled. This faceted

[3] http://image.google.com

[4] http://image.bing.com

indexing model relates to the users' levels of image descriptions and meets user needs for specific/compound queries. Further reading on this subject can be found in Kherfi et al. (2004) and Datta et al. (2008).

2.6.6 Summary of semantic-based image retrieval

While the categorization of methods used to address the semantic gap presented by Liu, Zhang, Lu and Ma (2007) is broad and covers many approaches and techniques in semantic-based image retrieval, there is an extremely important generation of methods that aim to address the semantic gap, which they did not consider in their categorization: the conceptual information of image features. This thesis argues that visual conceptual information is an important method to narrow the semantic gap by describing image content using perceptual descriptors that are derived from human perception. The following section reviews existing systems that characterize the visual conceptual information of image features.

2.7 Visual conceptual information

Semantic-based image retrieval systems, that operate on conceptual information, are an important subset of the general semantic-based image retrieval systems. These systems utilize the perceptual information extracted from images to facilitate indexing and retrieval. The perceptual properties of image objects as humans understand and interpret them can be described as the visual conceptual information. The visual conceptual information can be viewed as a natural way to represent image features that is derived from the human understanding of the visual properties of image objects.

These descriptors involve vocabulary to describe the visual features extracted from images. It is important to distinguish between the vocabulary used in the context of the visual conceptual information and the vocabulary used in the bag-of-words

(BOW) representation. In the bag-of-words representation, the term ‘vocabulary’ refers to the clusters generated by clustering the extracted features (typically Scale-invariant feature transform (SIFT) features: an algorithm that extracts local features that are invariant to changes in illumination, rotation, scaling, and slight changes in viewpoint (Lowe; 1999)). As a result, this vocabulary is not derived in accordance with the human perception as it merely represents the results of a clustering algorithm. However, in the context of visual conceptual information, the obtained ‘vocabulary’ represents a set of high-level concepts abstracting the image features from their numerical representation into a set of transparent symbolic descriptors readable by human users. For example, the conceptual information of the color features differentiate ‘a red apple’ and ‘a green apple’.

Few studies have considered the conceptual features of image objects. Nakagawa et al. (2004) suggest mapping image segments to words describing the color, texture, and shape features as a first step in an image classification framework. Their implementation of the color names is inspired by the work presented in Mojsilovic (2002). As for the texture features, they assigned texture-related adjectives adapted from human perception-based clustering scheme described in Rao and Lohse (1993); Bhushan et al. (1997). However, the taxonomy used for the shape features is not explicit. As presented in their work, there is no pre-defined set of shapes used: they present simple shape-related words like rectangular, circular, angular, flat, jaggy, protruded, elongated, etc. Even though they utilize the conceptual information to narrow the semantic gap, they do not use it to query image objects such that the query reflects the human understanding of these features.

Using visual conceptual information has not gained as much interest as the relevance feedback method. Most of the solutions to address the conceptualization of the low-level image features have been mainly proposed for color features, neglecting shapes and textures.

The following sections review frameworks dealing with the conceptualization of the low-level visual features, which is the foundation for a major contribution of this work.

2.7.1 Conceptualizing the low-level color features

The extraction of the conceptual color information is guided by the research carried out in color naming and categorization by establishing a correspondence between color stimuli and 'basic color terms' (Berlin and Kay; 1991). Further work by Gong et al. (1996) presents a framework for characterizing eleven visual color concepts in the Munsell HVC (Hue, Value, Chroma) perceptive color space. Conway (1992) presents a color naming model in which particular points in HSL (Hue, Saturation, Lightness) color space correspond to 179 predefined color names.

Berk et al. (1982) propose a semantic color classification system named CNS (Color Naming System) in which the HSL (Hue, Saturation, Lightness) space is quantized into 627 predefined colors.

Belpaeme (2001) presents a color categorization system through the highlighting of a multi-agent framework. Agents perceive and categorize color stimuli and communicate among each other to learn color names.

Liu et al. (2004) introduce a region-based image retrieval system with high-level semantic color names in which the user is able to query with keywords specifying the semantic color names with the semantic concepts. Query by regions is supported as well. The low-level features of colors are extracted in the HSV space and mapped to one of 35 semantic color names defined in this framework. Liu et al. (2005) introduce a region-based image retrieval system with 93 high-level semantic color names.

Continuing the evolution of research work that proposes to use color names for the purpose of image description is van de Weijer and Schmid (2007). Their color naming is based on the 'basic color terms' of Berlin and Kay (1991). Using Bayes law, eleven color categories are learned from manually segmented and labelled regions from a dataset containing images from the E-bay auction website[5]. Experimentally, they combined the color names with shape features using SIFT-descriptors in a bag-of-words representation. Linear SVM classifiers are then trained using the obtained frequency histograms of the visual words. Their experiments involve measuring the classification accuracy without including the retrieval. Accordingly, the color names are not utilized in the retrieval phase.

van de Weijer et al. (2009) use a probabilistic latent semantic analysis (pLSA) model to learn the eleven color categories of Berlin and Kay (1991). Color categories are learned on a set of images collected from Google[6]. This implementation has been also used by Rusinol et al. (2010) in a framework that combines shape context descriptors with the visual color concepts to classify image objects.

Hou and Zhang (2007) propose a color conceptualization method that represents the expectations of the distributions of certain semantic themes. Their method does not involve mapping the colors in an image into a set of basic color terms; rather it concerns describing the high-level semantic theme of the image such as “warm” or “cold” without characterizing the perceptual nature of image features.

2.7.2 Conceptualizing the low-level texture and shape features

Although several computational models and techniques have been proposed for texture analysis and understanding used in content-based image retrieval systems, they often fail to capture aspects related to human perception. Belkhatir (2005) proposes to use a lexicon consisting of eleven texture clusters highlighted in Bhushan et al. (1997) as a basis for mapping low-level texture features to symbolic categories. The latter are described by text-based descriptors: bumpy, cracked, disordered, interlaced, lined, marbled, netlike, smeared, spotted, uniform, and whirly. Nakagawa

[5] http://www.ebay.com

[6] http://www.google.com

et al. (2004) also use this categorization as the basis to represent texture categories in an image classification system.

As for shape features, many studies involving their extraction and representation of their low-level features have been proposed (cf. Section 2.3.4). The aim of these methods and techniques is to capture the low-level shape features of objects depicted in images (Zhang and Lu; 2004).

For the symbolic shape characterization, two categories of shape taxonomies exist in the literature. The first proposes to classify shapes according to an application domain (such as archaeology) and the second describes shapes according to their geometrical features for the aim of generic shape categorization.

In the first category, most of the work considers proposing domain specific shape taxonomies and ontologies. For example, Liu, Zhang, Tjondronegoro and Geve (2007) present a framework for classifying birds based on their shape features. They introduce an ontology of bird names as the basis of their framework. Another specific field of study which has developed vocabularies for categorizing and naming shapes is the archaeology. Black and Weer (1936) use a set of main shape categories describing well-known archaeological artifact objects.

As far as the generic characterization of shape concepts is concerned, a framework that involves mapping the image segments to words as the first step of image classification is proposed in Nakagawa et al. (2004). They presented simple shape-related words like rectangular, circular, angular, flat, jaggy, protruded, etc. without showing a finite set of shape names used in their framework. More recently, Jarrar and Belkhatir (2010) propose a framework for the mapping and symbolic extraction of the shape features which is discussed further in this thesis in Chapter 4.

Chapter 3 presents the approach of describing images using the visual conceptual information as well as the theoretical foundation of the state-of-the-art methods adopted in this thesis to map the low-level color and texture features into visual concepts. Subsequently, Chapter 4 presents a main contribution of this thesis: a framework for characterizing the visual concepts of the shapes of image regions.

2.8 Summary

This chapter has presented the foundation related to the research area of this thesis. The evolution of image retrieval starting with the manual annotation of images using text descriptors, then utilizing the visual features from images to overcome the manual annotation of images forming the content-based image retrieval, then using the visual image features to infer textual annotations describing the semantic meanings conveyed in images have been presented. The semantic gap, which is the main bottleneck in the semantic-based image retrieval has been illustrated along with a review of approaches and techniques used in the literature to overcome this gap.

One of the methods to narrow the semantic gap is using the conceptual information of the visual image features to enrich the description of images in accordance to human perception. This is the main interest of the work presented in this thesis, which aims to describe image regions using the visual conceptual information of the color, texture, and shape descriptors to classify them into high-level semantic categories. This approach has been presented in Section 2.7 and is explained in Chapter 3. Chapter 4 also presents a framework for the extraction and characterization of the visual shape concepts of image regions.

Chapter 3

The Visual Conceptual Information

3.1 Introduction

Chapter 2 has shown that existing image indexing and retrieval frameworks tend to use the low-level visual features such as color, texture, shapes, spatial information, etc. to represent and index image content. However, these features are not descriptive in the way humans perceive and understand the visual properties of objects depicted in images. Moreover, these extracted features ignore the conceptual properties of the object or the scene they represent. There is still a, 'semantic gap,' between the low-level visual features and the richness of the human interpretation of their visual properties.

This chapter aims to address the semantic gap by using the visual conceptual information for the task of image description and indexing. In doing so, human users can use these visual conceptual features to specify the perceptual properties of the semantic objects in their queries. For example, instead of direct involvement of users in the time-consuming relevance feedback process, users can specify the visual properties of image objects in their queries. The results are then filtered according to the visual properties of color, texture or shape they provided.

To illustrate this, an example using the query "apple" on the Google image search engine[1] is shown in Figure 3.1. We can see that the result contains images of red colored apples, some of green colored apples, and some image of the Apple Inc. logo. The user, however, might be interested in retrieving only red colored apples. We can see that among the top 10 images retrieved, 5 of them contain the logo of Apple Inc. which is irrelevant to what the user is looking for. However, after modifying the query to include a simple conceptual visual property, e.g., "red apple", it can be seen that the results obtained are different as shown in Figure 3.2. Clearly, all the results obtained represent "red apple" since the images are filtered to those of a red color. The same applies for Figure 3.3 which shows results for the query "green apple".

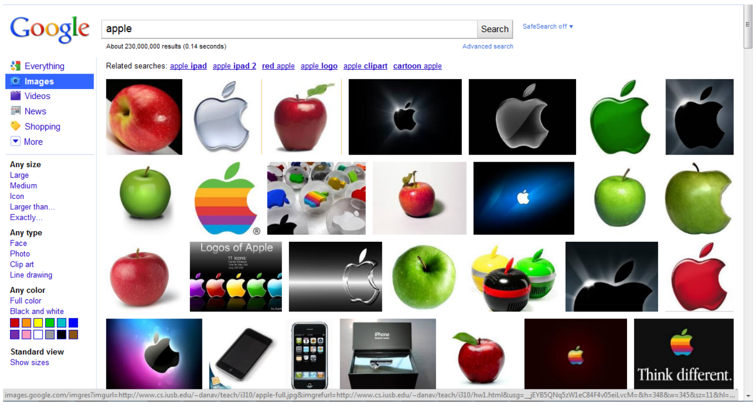


Figure 3.1: *The results of the Google image search for the word "apple"*

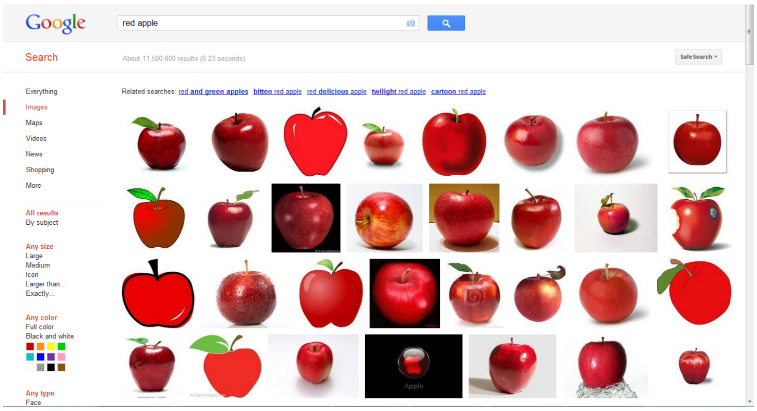
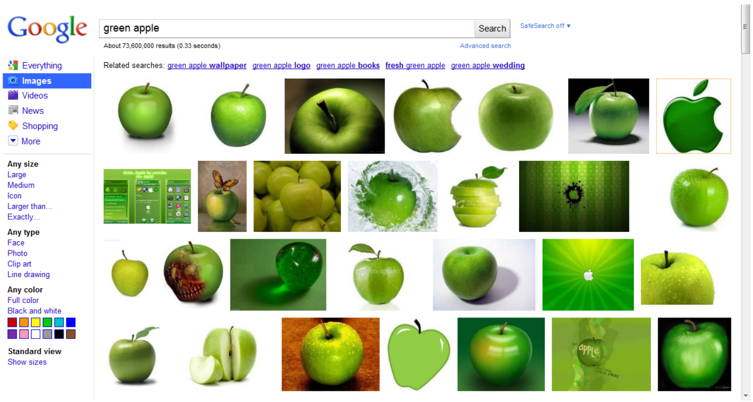


Figure 3.2: *The results of the Google image search for the query "red apple" specifying the color term* red

[1] http://image.google.com



About 73,600,000 results (0.33 seconds)

Related searches: green apple wallpaper green apple logo green apple books fresh green apple green apple wedding

Figure 3.3: *The results of the Google image search for the query "green apple" specifying the color term* green

This clearly has an impact on the precision of the results obtained. Querying using one keyword (such as apple) will retrieve all objects annotated with the term "apple" in the dataset. As far as the semantic-based image retrieval is concerned, the system will retrieve a set that maximizes the probability of the queried object based on the examples used in training the system.

The previous example is based on Google's image search approach in which the surrounding text, image caption, page title, etc. are used to do the retrieval (cf. Chapter 2, Section 2.6.5). The aim of this thesis is to use the visual information extracted from images to classify classify them into semantic categories. This is achieved using supervised machine learning algorithms that learn semantic categories using the visual features extracted from image regions. Each image region is labeled with a class label and the supervised learning algorithm establishes correspondence between image regions and their class labels.

Classifying images from the web using image captions and the surrounding contextual information (cf. Section 2.6.5) is out of the scope of this project.

Let us consider the "red apple" and "green apple" examples. If the indexing framework is trained with the semantic concept "apple", then there might be red as well as green images labelled with the semantic concept "apple" in the training set. Retrieval is based on finding the result that maximizes the probability. To solve such a problem, the easiest and simplest implementation is to filter the training examples of the category apple into two separate categories: red apple and green apple. Then for a new input instance it will be annotated with “red apple” or “green apple”. However, this solution is not practical since both green apples and red apples are apples. Moreover, this solution will require human subjects to manually label and categorize the images of each class into subclasses which consumes substantial time and effort. Besides that, for some other semantic categories, such as “sky”, there is a diversity of sub-classes, for example, red sky, light-blue sky, dark sky, blue sky, night-view sky, etc. In this case, the annotation and categorization will consume even more time and effort.

Motivated by this problem, this thesis considers the “visual conceptual information” of the image objects and scenes as a robust method that is based on the human interpretation of the visual properties of the image features. In doing so, images and image objects are automatically indexed using their visual conceptual properties instead of low-level features that are represented by numbers.

The visual conceptual information involves representing the low-level visual features using concepts that conform with human perception. It can be viewed as a natural way, based on the human understanding of the image visual features, to represent the image features. The visual conceptual information is also referred to as visual concepts in the context of this work.

The visual concepts can also be seen as a method to quantize the low-level image features into human-derived representations that will be used with some learning algorithms, especially those that are close to the way humans think, namely rule-based classifiers which are most commonly represented in decision trees.

Advantages of using the visual conceptual information include

- Addressing the semantic gap by allowing human users to leverage these visual concepts in refining the queries as opposed to relevance feedback systems.

- Tackling issues related to indexing images in high dimensional spaces, as these high-level concepts are a compact representation that is mapped from the low-level features which are usually of high-dimensions (10^2 and 10^3).
- Tackling the issue of the wide ranges in the feature spaces for a given semantic category such as "sky" which might be light-blue sky, cloudy sky, dark sky, or red sky (at sunset or sunrise) as illustrated previously. This is addressed by indexing the semantic concepts taking into account the semantic categories and distribution of the conceptual information of the image features.

This thesis aims to characterize the conceptual features in accordance to the human understanding. Thus, the color, texture, and shape features are used as human users can describe them using words contrarily to using other set of features such as Scale invariant feature transform (SIFT) (Lowe; 1999) or Speeded up robust features (SURF) (Bay et al.; 2008) that are extracted from interest points in images.

The following section reviews frameworks for mapping the low-level color and texture features into their corresponding visual conceptual information. Then the theoretical formation of the framework to characterize the visual color and texture concepts adopted in this thesis are presented in detail. Chapter 4 introduces a framework for the extraction and characterization of the visual shape concepts.

3.2 Theoretical models to characterize the color and texture concepts

To use the conceptual information of the visual properties of images, we must describe images using a pre-defined set of visual concepts. These categories describe the nature of the color, texture, and shape information extracted from image objects.

The main aim is to use the visual concepts to describe and index image regions. This involves describing image objects in symbolic words reflecting the perceptual nature of image objects as humans perceive and understand them. The main idea is to use the perceptual visual information of the image properties (such as color, texture, and shape features) to describe images rather than using the low-level features in their numeric representation. Each image region is rather described using symbolic perceptual descriptors that conform to the human understanding and interpretation of their visual properties. The set of characterized symbolic descriptors are then used to establish correspondence with a set of high-level semantic categories that describe the nature of image objects. These symbolic descriptors can be viewed as mid-level features between the low-level image features and the high-level semantic categories. For example, a football would be described as circular in shape, white and black in color, with a netlike texture. These visual conceptual properties of the 'football' object are utilized to represent image objects rather than the numeric low-level image features.

The representation of image regions consists of a histogram describing the probability distributions of the predefined symbolic color, texture, and shape categories. The obtained histograms comprise perceptual information and are derived from the human perception and understanding of the visual properties of image objects. So, users can use the set of characterized symbolic descriptors to specify the properties of objects they would like to retrieve. For example, users can specify the perceptual color information of objects to retrieve such as "people wearing red clothes" or "people wearing purple clothes". Figure 3.4 shows an example of using the visual concept representation to describe image regions. The low-level color, texture, and shape features are mapped to a set of perceptual categories. The first step is to extract the low-level color, texture, and shape features to extract the visual concepts. These are characterized by models mapping their low-level features into symbolic visual concepts. The output of each model is a histogram of the corresponding visual concepts over an image region as illustrated in Figure 3.4. The set of characterized visual concepts are concatenated in a unified representation that is used by a machine learning algorithm to learn semantic meanings of image regions.

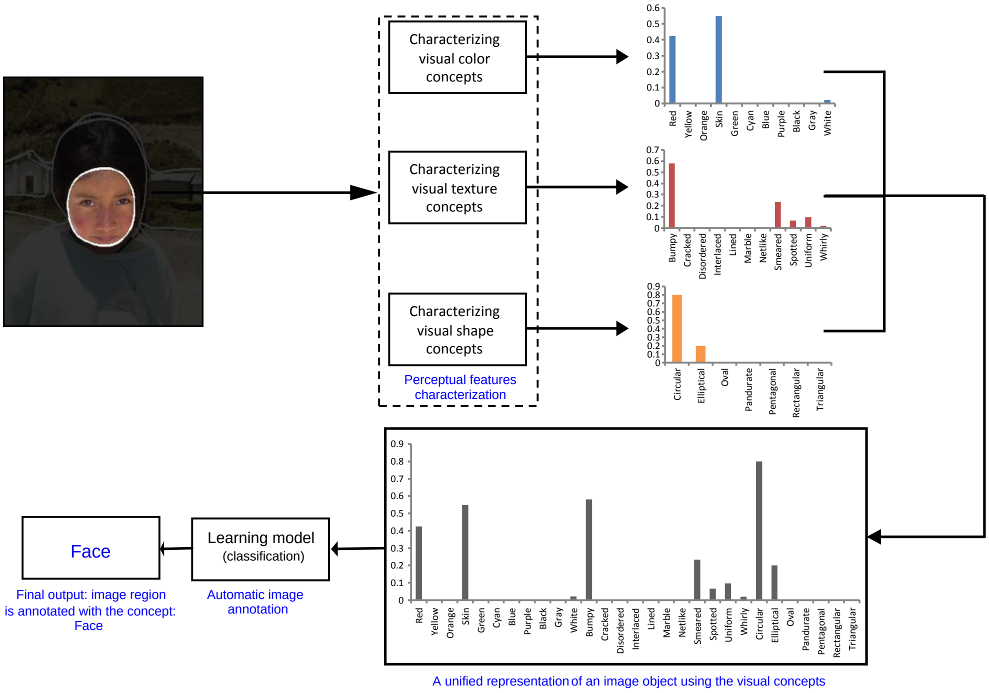

Characterizing visual color concepts
Characterizing visual texture concepts
Characterizing visual shape concepts
Perceptual features characterization
Red
Yellow
Orange
Skin
Green
Cyan
Blue
Purple
Black
Gray
White
Bumpy
Cracked
Disordered
Interlaced
Lined
Marble
Netlike
Smeared
Spotted
Uniform
Whirly
Circular
Elliptical
Oval
Pandurate
Pentagonal
Rectangular
Triangular
Face
Final output: image region is annotated with the concept: Face
Learning model (classification)
Automatic image annotation
A unified representation of an image object using the visual concepts

Figure 3.4: *An example of describing an image region using the visual conceptual information*

3.2.1 Characterizing conceptual color information

Research following Berlin and Kay (1991) have revolved around stressing a step of correspondence between color stimuli and “basic color terms” which they characterize by the following properties: (a) their application is not restricted to a given object class, for example, the color characterized by the term “olive color” is not valid; (b) they cannot be interpreted conjointly with object parts, e.g., “maple leaf color” is not a valid color; (c) their interpretation does not overlap with the interpretation of other color terms; and finally (d) they are psychologically meaningful.

Further work proposed in Gong et al. (1996) involves characterizing perceptual colors in the HVC (Hue, Value, Chroma) perceptive color space. The latter belongs to the category of user-oriented color spaces, as opposed to hardware-oriented spaces such as RGB (Red, Green, Blue). User-oriented color spaces define color as being perceived by a human through tonality (describing the color wavelength), saturation (characterizing the quantity of white light in the color spectral composition) and brightness (related to color intensity).

The model of characterizing the conceptual color information adopted in this thesis is based on the model proposed by Gong et al. (1996). Eleven perceptual color concepts are defined that are: **green** (c_1=Gn); **cyan** (c_2=C); **skin** (c_2=Sk); **gray** (c_4=G); **Red** (c_4=R); **orange** (c_4=O); **yellow** (c_4=Y); **purple** (c_4=P); **black** (c_4=Bl); **white** (c_4=W); and **blue** (c_2=Bu). A color descriptor C that contains the probability distribution of each of the eleven color concepts in a an image region r is obtained as

$$C = \{p(c_1|r), p(c_2|r), \ldots, p(c_{11}|r)\} \tag{3.2.1}$$

where $p(c_i|r)$ is the i-th color concept of the 11 aforementioned color concepts.

The process of characterizing these conceptual color categories involves algorithmically transforming the extracted low-level color features in the RGB color space to tonality, brightness, and saturation values in the perceptually uniform HVC color space. The mapping between the low-level features and the color categories is detailed in Chapter 6, Section 6.2.1.

3.2.2 Characterizing visual texture concepts

Several studies have proposed algorithms and techniques concerning the low-level texture features computation and extraction from images (cf. Section 2.3.3). As far as the symbolic representation and the human understanding of the texture features is concerned, few attempts have been made to propose ontologies and taxonomies for symbolic texture characterization and naming. Bhushan et al. (1997) conduct a psychological study of human perception of textures. They present a texture lexicon consisting of eleven texture categories. The computational model for the characterization of the symbolic texture features presented in this thesis involves mapping the low-level texture features into the eleven texture concepts presented in their study.

The computational model for texture extraction adopted in this thesis captures aspects related to human perception. Therefore, it is inspired by the work presented in Leow and Lai (1999) where a computational framework for texture extraction, which is the closest approximation of the human visual system, is proposed. The action of the visual cortex, where an object is decomposed into several primitives by the filtering of cortical neurons sensitive to several frequencies and orientations of the stimuli, is simulated by a bank of Gabor filters (cf. Section 2.3.3). However, as opposed to Leow and Lai (1999) which operates at the global level of an image, this work focuses on computational texture extraction at the image region level. Therefore, each region is characterized by its Gabor energy distribution within five spatial frequencies covering the whole spectral domain and six angular orientations. The values of these parameters are not reported in the original work in Belkhatir (2005).

Table 3.1 shows the eleven texture categories and their descriptions.

The process of mapping the low-level texture features into texture concepts is detailed in Section 6.2.2 in Chapter 6. Again, this framework is adopted from the literature to be integrated with the visual color and shape concepts to index and represent image regions.

These eleven textures define a feature vector T which contains the probabilistic estimation of the aforementioned texture categories over an image region r such that

$$T = \{p(t_1|r), p(t_2|r), \ldots, p(t_{11}|r)\} \tag{3.2.2}$$

where $p(t_i|r)$ is the i-th texture concept in the set of defined 11 texture concepts.

Texture category	Description
Bumpy	this texture cluster gathers textures with random 3-dimensional imperfections
Cracked	comprises textures exhibiting random linear orientation
Disordered	gathers textures that do not present any structure nor any dominant orientation
Interlaced	gathers structured textures with a weave-like structure
Lined	consists of linearly oriented textures (the orientation is along a straight line)
Marbled	consists of veined or striated textures
Netlike	consists of texture with two-directional characteristic features combined to form a weave. Textures in the category ‘disordered’ differ from textures in this category as they present a certain amount of variation and randomness
Smeared	groups textures presenting some disfigurement. It denotes negative aesthetics
Spotted	consists of textures with representative features being small, blob-like, and scattered over a plane
Uniform	refers to uniform textures in which the nature of the repetition is not specified
Whirly	consists of circularly oriented textures

Table 3.1: *The eleven texture categories and their descriptions*

3.3 Summary

This chapter has presented an approach to describe image content using the visual conceptual information that aims to address the semantic gap between the numerical representation of the low-level image descriptors and the richness of human interpretation of their content. The frameworks for characterizing the visual color and texture concepts adopted in the body of this thesis have been presented in detail and their implementation detail is further demonstrated in Chapter 6, Section 6.2.

In Chapter 4 a framework for the characterization and the extraction of the visual shape concepts is presented in detail. The processes of extracting the visual color and texture concepts used in the body of this framework and the integration of the visual color, texture, and shape concepts for the task of image description and indexing are presented in Section 6.4 of Chapter 6.

Chapter 4

A Framework for the Extraction and Characterization of the Conceptual Information of Shapes

4.1 Introduction

Section 2.7 in the literature review showed that solutions aiming at conceptualizing the low-level visual features have been proposed for color information (Conway; 1992; Gong et al.; 1996; Belpaeme; 2001; Liu et al.; 2005; van de Weijer et al.; 2009), texture information (Belkhatir; 2005), and spatial information (Belkhatir; 2009).

As for shape features, many studies exist in the literature concerning the extraction and representation of the low-level features of shapes (Zhang and Lu; 2004; Yang et al.; 2008). However, few studies and taxonomies aiming at the generic naming and symbolic description of shapes exist. Black and Weer (1936) propose a set of main shape categories describing archaeological artifact objects. They classify the shapes of the archaeological artifacts into the following categories: rectanguloid, trianguloid, elliptical, oval, ovate, pentagonal, panduriform, miscellaneous, and compound shapes.

Inspired by this taxonomy and as an extension to the research aiming at the conceptualization of the low-level signal features, this chapter illustrates a framework that aims to bridge the gap between the extracted shape features and their conceptual characterization. The conceptual shape knowledge base is represented by a lattice structure organizing these shape concepts through a partial order. This shape framework is then used to map the low-level features to high-level shape concepts using a support vector machine. It automatically maps extracted 64-dimensional Fourier descriptors characterizing the low-level shape features to shape concepts, with an implementation discussed in detail in Section 6.3 of Chapter 6.

This framework is a component of the whole system presented in the body of this thesis. This module is integrated with the frameworks to characterize the visual color and texture concepts adopted from the literature for the task of image description and semantic image annotation.

The remainder of this chapter is organized as follows. Section 4.2 presents the fundamental shape concepts which are the basis of the shape lattice. The transformation processes which develop new shapes are presented in Section 4.3, and the organization of shapes within a lattice-based structure is discussed in Section 4.4.

4.2 Fundamental shape concepts

Starting from seven basic shapes characterizing geometrical properties of shapes, a set of transformation functions is used to derive new shapes. The latter inherit the attributes of the basic shapes, which highlights an organization within a hierarchical structure defined by a partial order called the shape lattice. The seven basic shapes which form the foundation of the shape lattice are characterized by the shape concepts **circular**, **rectangular**, **triangular**, **pandurate**, **elliptical**, **pentagonal**, and **oval**.

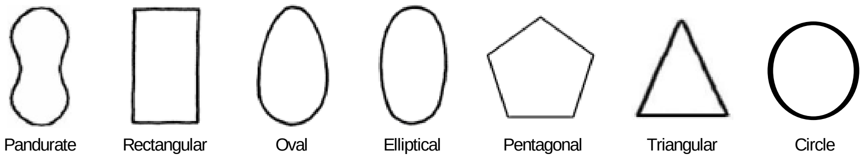


Figure 4.1: *The fundamental shapes*

It is taken as an axiom that shapes are characterized by geometric attributes rather than the semantics of the corresponding object. For example, traffic signs could be categorized based on their conceptual shape characteristics. Thus, “no parking” and “no entry” signs would be classified as *circular*, “T-junction” and “give way” would be classified as *triangular*, “parking” and “hospital” traffic signs would be characterized as *rectangular* and so on. Then, objects with similar shapes might be further classified by integrating additional features such as colors and textures.

This framework aims at the generic naming of extracted image regions using their geometrical properties not their semantic nature.

A decision was made to bound this framework with seven fundamental shapes, adopted from the Black and Weer (1936) taxonomy, because including more shapes results in a larger, wider, and more complex shape lattice. The experiments in Chapter 6 show that bounding this module with seven fundamental shapes provides sufficient coverage of variable shapes for the experiments conducted. From the aforementioned fundamental shapes, a number of new shapes are derived by applying a set of transformations as discussed in the following section.

4.3 Transformations to develop new shapes

A set of geometrical transformation processes are applied on the basic geometrical shapes in order to obtain novel shapes derived from the basic shapes and therefore inheriting their properties.

The transformation processes are applied successively on shapes to derive new shapes at lower levels in the shape lattice. By applying a single transformation process on a shape, a new level in the hierarchy is generated. So the first level comprises shapes that are modified by a single transformation process, shapes on the second level are modified with two transformation processes and so on until reaching the leaf-node level, in which, all the possible combinations of transformation processes have been applied on shapes.

The transformation processes are divided into two categories: *edge transformations* and *head/base transformations* as shown in Figure 4.2.

Four edge transformations can be applied to modify the edges of shapes: excurvation, incurvation, edge sharpening, and constriction. *Excurvation* curves the side of a geometrical shape outward, where *incurvation* is the inverse operation. *Constriction* takes an excurvated or incurvated side and curves into two lines connected by a wide angle (illustrated by shape (a) in Figure 4.3, which is a rectangle modified by two constrictions-outward and one constriction-inward). *Edge sharpening* adds a nail-like shape to the edge inward and outward.

Base/head transformations are three: sharpening, widening, and truncation. *Sharpening* is the process of narrowing the base/head of a shape, *widening* is the inverse operation. *Truncation* is the process of cutting the shape at the level of the base or the head.

The transformation processes that are applied in specific order are the constriction inward and constriction outward as they are subsequent to the incurvation and excurvation processes. Constriction outward is applied on an excurvated side, hence it is applied to an edge after applying the excurvation process; similarly constriction inward succeeds the incurvation process on an edge.

Order is not important for the other transformation processes shown in Figure 4.2.

The degree to which shape edges are modified are the same as Black and Weer (1936), the mathematical variation of the transformation and graphical modifications are out of the scope of this work.

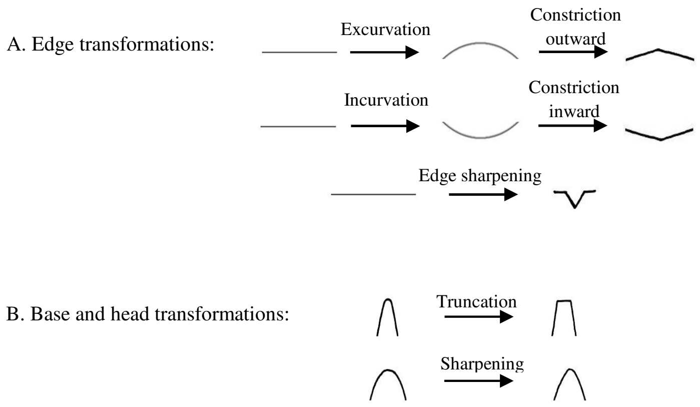


Figure 4.2: *The transformation processes used to develop new shapes in the shape lattice*

4.4 Lattice-based organization

The shape lattice consists of seven sub-lattice structures originating at each of the basic shapes. All possible compositions of shape transformation processes are considered to generate the lattice structure, yielding a total of 898 shapes in the shape lattice.

To generate the first level in each sub-lattice, a single transformation process is applied on a side, base or head. For example, the first level in the rectangle sub-lattice will comprise rectangle with an incurvated edge, rectangle with an excurvated edge, rectangle with a sharpened-inward edge, and rectangle with a sharpened-outward edge. The order of applying transformations is not important except for constrictions as noted. Another transformation is applied on the shapes at the first level to generate the second level and so forth. In other words, by applying a transformation on a shape, a new level of descendants is obtained, the height of the sub-lattice is increased by one, and the width of the lattice increases by the number of the new added shapes.

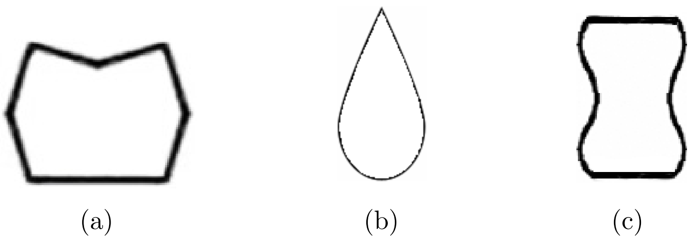
The transformation processes are applied to every generated shape at every level of descendants in the lattice until the leaf level is reached. The leaf level contains shapes with all the possible combinations of transformation after which no novel shapes can be generated.

The symbolic text-based descriptor (i.e., the visual shape concept) of a generated shape is constructed by using the basic geometrical name along with the name of the transformations applied. For a given shape, the equivalent shape concept is obtained by naming the corresponding basic geometrical form and successively concatenating the applied geometrical transformations which generate the shape. For example, "rectangular with three constricted-inward sides" or "triangular with two incurvated and one sharpened-inward sides".

Figure 4.3 shows three shapes generated by the geometrical transformations in the shape lattice.

The shape 4.3(b) in Figure 4.3, corresponds to the oval shape modified by one transformation, which is "head sharpening". This shape is located at the first level of descendants since only one transformation process is applied. The shape 4.3(c) represents a shape derived from the basic pandurate shape located at the second level of the lattice in the pandurate sub-lattice. This shape is generated by modifying the basic pandurate shape with a head truncation transformation resulting in a shape described by the concept "pandurate with truncated head". This shape is further modified by truncating its base resulting in the shape described by the shape concept "pandurate with truncated head and truncated base".

The shape 4.3(a) in Figure 4.3 is described by the shape concept "rectangular with two constricted-outward and one constricted-inward sides" at the sixth level of descendants in the rectangular sub-lattice. It is generated as follows



(a) (b) (c)

Figure 4.3: *Three shapes from the shape lattice: (a) Rectangular with two constricted-outward and one constricted-inward sides; (b) Oval with sharpened head; and, (c) Pandurate with truncated head and truncated base*

- first, one incurvation is applied on a single side of the basic rectangular shape resulting in a shape described by the concept "rectangle with one incurvated side" and located at the first level of descendants in the rectangular sub-lattice;
- then, the inward constriction transformation is applied on the incurvated edge generating the shape described by the shape concept "rectangle with a constricted-inward side" at the second level of the lattice;
- two excurvations are applied on two sides successively; and
- two outward constrictions are then applied on the two excurvated sides generated in the previous step, generating the final shape.

The following section illustrates the general organization of the seven sub-lattices in the main shape lattice. Further details of the organization of the oval sub-lattice highlighting the transformation processes applied to generate new shapes inheriting the basic oval shape are presented in Section 4.4.1.6 and illustrated in Figures 4.4 and 4.5.

It might be argued that the elliptical and oval shapes could be derived from the circular shape by modifying it with the transformation process. However, merging these shapes with the circular sub-lattice results in adding levels to the hierarchy, causing that sub-lattice to be very large, wide, and complex. Thus, the elliptical and oval shapes are considered fundamental shapes with their own sub-lattice hierarchies.

4.4.1 Sub-lattice organization

This section explains the transformation processes applied on each basic shape to generate its sub-lattice structure, its corresponding height, and the number of shapes generated in each sub-lattice. Figure 4.4 (I) shows the general organization of the shape lattice highlighting the first level of descendants in the lattice, which comprises the seven fundamental geometric shapes.

4.4.1.1 The circular sub-lattice

The set of successive transformations applied to create the circular sub-lattice are as follows. First, the truncation transformation process is applied to obtain the half-circular shape. Further transformations are applied to the base of the half circle which are the edge transformations incurvation, excurvation, inward and outward constriction, and inward and outward edge sharpening. As a result, the circular sub-lattice consists of four levels of descendants and contains a total of 8 shapes.

It can be noticed that only a few shapes exist in the circular sub-lattice. As illustrated, the first transformation applied was the truncation to transform the circular shape into half-circular. Accordingly, the subsequent transformations are constricted by the allowed transformations on edges as mentioned in Section 4.3. Moreover, adding more transformations results in repeating shapes from other sub-lattices. For example, if the half circle is truncated again, it will repeat the shape triangular with an excurvated edge that appears in the triangular sub-lattice.

4.4.1.2 The elliptical sub-lattice

The transformations applied to the elliptical shapes are head and base truncation, and the edge transformations incurvation, inward/outward edge sharpening, and inward constriction. The excurvation process is excluded since applying it on a truncated base or head of the elliptical shape will re-create the basic elliptical shape. The outward constriction is therefore not considered as it is applied after the excurvation transformation. In this sub-lattice there are seven levels of descendants and a total of 36 shapes.

4.4.1.3 The triangular and rectangular sub-lattices

The transformations allowed on the triangular and rectangular shapes are excurvation, incurvation, inward/outward constriction, and inward/outward edge sharpening. These transformations are applied to each side of the triangle and the rectangle by starting with one transformation process on one side until the leaf level of the structure is reached. The leaf level consists of shapes generated by all compositions of the transformation processes allowed on the triangular and rectangular shapes.

The sub-lattice originating at the triangular shape has four levels of descendants and contains a total of 48 shapes.

The difference between the triangular and rectangular sub-lattices is the height and the number of elements in each one, as the transformations are applied to the four sides of the rectangular shape. In the sub-lattice of the rectangular shapes, there are five levels of descendants with a total of 419 shapes.

4.4.1.4 The pentagonal sub-lattice

The transformations applied to the pentagonal shapes are incurvation, inward constriction and inward/outward edge sharpening. All edges of the basic pentagonal shape are of the same length. The excurvation transformation process is excluded because applying the excurvation transformation on the edges of the pentagonal shape will re-create a circular shape. Accordingly, the outward constriction is therefore not considered as it is applied after the excurvation transformation. The pentagonal shapes sub-lattice consists of six levels of descendants with a total of 300 shapes.

4.4.1.5 The pandurate sub-lattice

The pandurate sub-lattice is itself complex even though it can be treated as a derivative of the elliptical shape with two-sides incurvated. Thus, the pandurate shape is considered a fundamental shape with its own sub-lattice hierarchy.

The transformations applied to the edges of pandurate shapes are inward and outward constrictions, since each side of a pandurate shape consists of two excurvations and one incurvation. Then, the truncation process is applied to the base and/or head of each of the transformed shapes, and consequently, further processes are eligible to apply to the truncated sides which are: incurvation, inward constriction, and both inward and outward edge sharpening. Again, the excurvation process is applied neither on the truncated head nor the base, since it will reconstruct the original pandurate shape. In this sub-lattice structure, eleven levels of descendants are generated with a total of 412 shapes.

4.4.1.6 The oval sub-lattice

To generate the oval sub-lattice, the allowed transformations are head sharpening, head and base truncation and edge transformations such as incurvation, inward constriction and inward/outward edge-sharpening. The excurvation transformation is excluded since applying it on a truncated base or head of the oval shape will re-create the basic oval shape. Therefore, the outward constriction is not considered either as it is a subsequent process of the excurvation as described in Section 4.3.

The detail of the oval sub-lattice is presented in Figures 4.4 and 4.5.

At the first level of the oval sub-lattice, the transformation processes head sharpening, head truncation, and base truncation are applied on the basic oval shape producing three shapes: “oval with truncated head”, “oval with truncated base” and “oval with sharpened head” as shown in Figure 4.4.

From the shape described by the concept "oval with sharpened head" (Figure 4.4, (A)), a new shape is generated at the second level of descendants named "oval with sharpened head and truncated base" (Figure 4.4, (B)) on which the edge transformations are applied. These shapes constitute the third level of the lattice. The shape "oval with sharpened head and incurvated base" (Figure 4.4, (C)) is extended with the inward constriction at the fourth level of the sub-lattice structure.

Second, for the two shapes conceptually described as "oval with a truncated head (and base)" at the first level of descendants (shapes (D) & (E) in Figure 4.4), the allowed edge transformations on the head or base are applied. Then each of the generated shapes can be modified through truncation of the head or base.

The shape named "oval with incurvated head" is further modified by truncating its base (as shown in Figure 4.4,(F)), and the shape named "oval with sharpened outward base" (shape (G) in Figure 4.4) is modified by truncating its head and so on.

Then on the new truncated side, the allowed transformations are applied successively until no novel shapes can be generated, which corresponds to reaching the leaf level of the sub-lattice structure. The oval sub-lattice has a height of 7 levels and comprises a total of 41 shapes.

Figure 4.4 (II) shows the oval sub-lattice highlighting all the generated shapes at each level of descendants in the lattice, and Figure 4.5 shows a part of the oval sub-lattice, highlighting only one path originating from the basic oval shape with the concepts corresponding to each shape and the transformations applied until the leaf level is reached.

4.5 Conclusion

This chapter has presented the framework for the mapping and characterization of the conceptual shape information. The fundamental geometric shapes, the transformation processes applied on shapes to derive new novel shapes, and the final organization of the shape concepts within a lattice based organization from which the learning framework is constructed have been illustrated.

This framework is one component of the complete semantic-based image indexing framework presented in this thesis. Dealing with shape modelling is not an easy task especially when dealing with a generic description of the shape nature of image objects rather than their high-level semantic description. Thus, the presented framework has been bounded to describe the geometrical nature of the shape of extracted image regions using a set of fundamental shape concepts. Again, high-level semantically driven shape descriptors have been excluded as there will be a very large number of concepts describing shapes of natural objects.

This presented framework is integrated with the frameworks to characterize the visual color and texture concepts (cf. Chapter 3) adopted from the literature for the task of image description and semantic classification of image regions as illustrated in Section 6.4 of Chapter 6.

The implementation and experimental validation of this framework are presented in Chapter 6, Section 6.3. The experimental validation includes presenting the dataset containing all shapes within the shape lattice, which constitutes the knowledge base for the learning framework to map the low-level shape features to high-level shape concepts.

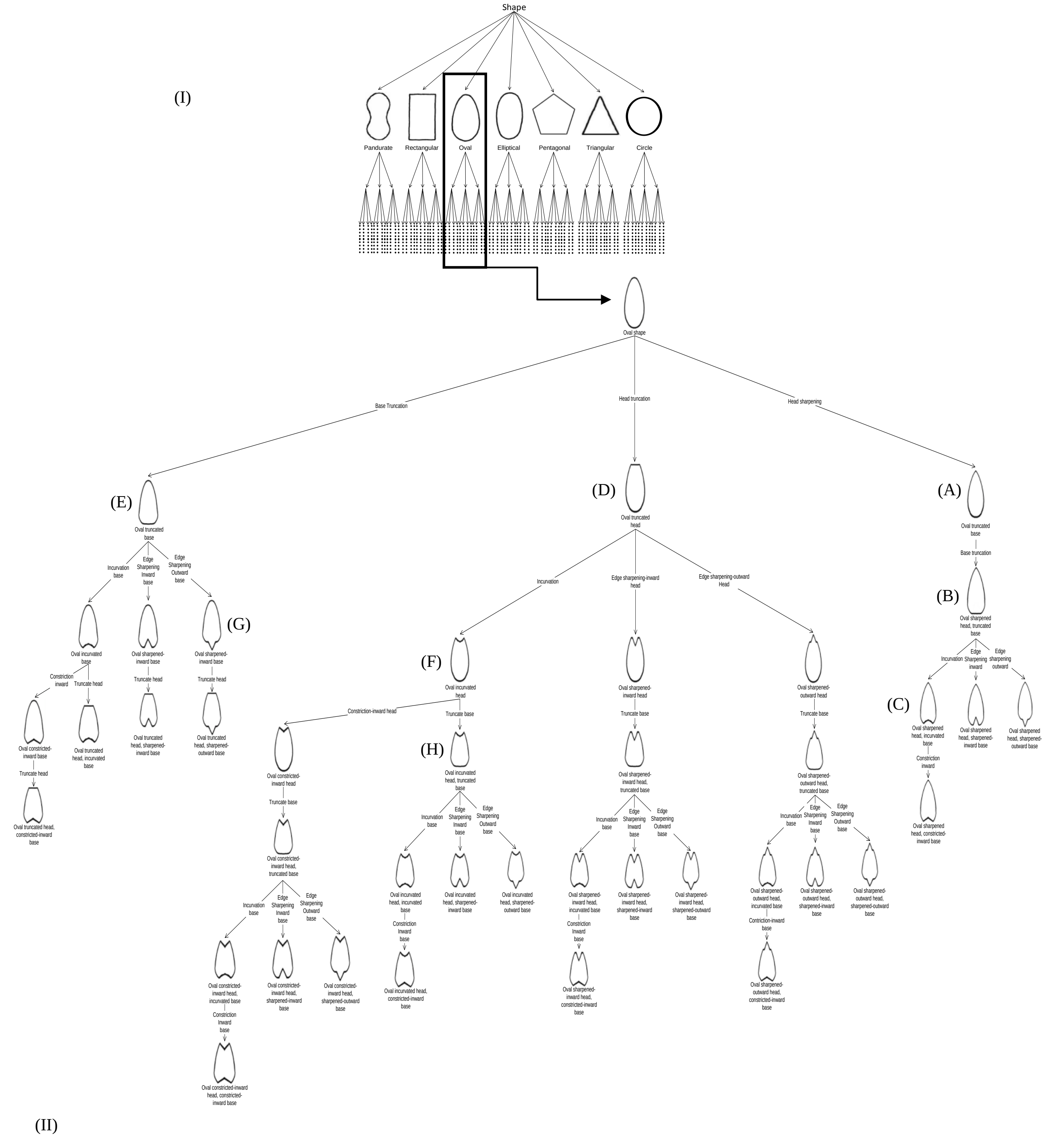

(I)
Shape
Pandurate
Rectangular
Oval
Elliptical
Pentagonal
Triangular
Circle
Oval shape
Base Truncation
Head truncation
Head sharpening
(E)
Oval truncated base
Incurvation base
Edge Sharpening Inward base
Edge Sharpening Outward base
Oval incurvated base
Oval sharpened-inward base
(G)
Oval sharpened-inward base
Constriction inward
Truncate head
Truncate head
Truncate head
Oval constricted-inward base
Oval truncated head, incurvated base
Oval truncated head, sharpened-inward base
Oval truncated head, sharpened-outward base
Truncate head
Oval truncated head, constricted-inward base
(D)
Oval truncated head
Incurvation
Edge sharpening-inward head
Edge sharpening-outward Head
(F)
Oval incurvated head
Constriction-inward head
Truncate base
Oval constricted-inward head
(H)
Oval incurvated head, truncated base
Truncate base
Oval constricted-inward head, truncated base
Incurvation base
Edge Sharpening Inward base
Edge Sharpening Outward base
Oval constricted-inward head, incurvated base
Oval constricted-inward head, sharpened-inward base
Oval constricted-inward head, sharpened-outward base
Constriction Inward base
Oval constricted-inward head, constricted-inward base
Incurvation base
Edge Sharpening Inward base
Edge Sharpening Outward base
Oval incurvated head, incurvated base
Oval incurvated head, sharpened-inward base
Oval incurvated head, sharpened-outward base
Constriction Inward base
Oval incurvated head, constricted-inward base
Oval sharpened-inward head
Truncate base
Oval sharpened-inward head, truncated base
Incurvation base
Edge Sharpening Inward base
Edge Sharpening Outward base
Oval sharpened-inward head, incurvated base
Oval sharpened-inward head, sharpened-inward base
Oval sharpened-inward head, sharpened-outward base
Constriction Inward base
Oval sharpened-inward head, constricted-inward base
Oval sharpened-outward head
Truncate base
Oval sharpened-outward head, truncated base
Incurvation base
Edge Sharpening Inward base
Edge Sharpening Outward base
Oval sharpened-outward head, incurvated base
Oval sharpened-outward head, sharpened-inward base
Oval sharpened-outward head, sharpened-outward base
Contriction-inward base
Oval sharpened-outward head, constricted-inward base
(A)
Oval truncated base
Base truncation
(B)
Oval sharpened head, truncated base
Incurvation
Edge Sharpening inward
Edge sharpening outward
(C)
Oval sharpened head, incurvated base
Oval sharpened head, sharpened-inward base
Oval sharpened head, sharpened-outward base
Constriction inward
Oval sharpened head, constricted-inward base
(II)

Figure 4.4: *(I) The main shape lattice; and (II) The oval shape sub-lattice originating at the basic oval shape*

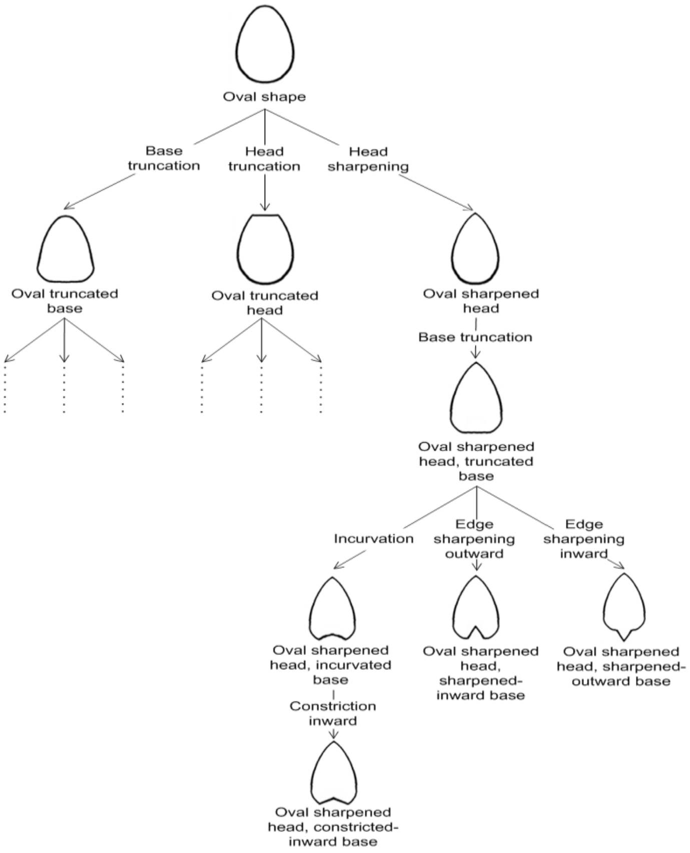


Figure 4.5: *Part of the Oval sub-lattice*

Chapter 5

Supervised Learning Algorithms for Learning Semantic Categories

5.1 Introduction

The goal of automatic image annotation systems is to find a correlation between the pixels creating an image and a set of keywords reflecting the semantic meaning of image objects, regions or semantic scenes as humans comprehend and understand them. Thus, the process of annotating images can be seen as a process of translating image content from a numerical representation into high-level semantic descriptors reflecting their visual content (Duygulu et al.; 2002). As a result of the annotation, users can query the database using symbolic descriptors to describe the perceptual content of images.

Several machine learning algorithms have been used for the task of establishing the aforementioned correlation using different low-level image features (Smeulders et al.; 2000; Zhang et al.; 2012). Which machine learning technique to use with which set of image features is still an open issue.

This thesis uses the visual conceptual information to describe and classify image regions into high-level semantic categories. This means a new set of features is being used for the learning and classification processes. Accordingly, it is unknown which supervised learner results in the highest classification accuracy. Thus, empirical comparison between several machine learning algorithms using the visual concepts is conducted to discover which learning algorithm performs better than the others using the visual concepts' feature space.

This chapter reviews the machine learning algorithms used to establish correspondence between image features and high-level semantic categories in the literature. The reviewed supervised learning algorithms are the algorithms that are subsequently used in the experiments to learn high-level semantic categories from the visual conceptual information.

This chapter is organized as follows. First, the notation and problem formulation that will be used in the rest of this chapter and in the experiments in Chapter 6 is presented. A brief theoretical introduction of each algorithm is presented in the following sections. A review of systems and frameworks utilizing each of the supervised learning algorithms is also included within each section.

5.2 Definitions and notations

This section aims to provide a unified notation and problem formulation that will be used in the rest of this chapter as well as in Chapter 6.

Let $r_i \in \Re^I$ be an image region in the set of all image regions in $\Re^I$ extracted from images in the dataset $\mathbb{D}$. The dataset $\mathbb{D}$ is split into disjoint training D_{train} and test D_{test} subsets. Each image region is annotated with a semantic class label $y_i \in \mathbb{S}$ that belongs to the set of semantic concepts $\mathbb{S}$ in the knowledge base $\mathbb{K}$. This knowledge base contains annotated training image regions (i.e., they correspond to a unique semantic concept with probability 1).

The visual shape concepts are characterized from the global shape of an image region and represented by the feature vector $s_r = \{p(s_1|r), \ldots, p(s_l|r)\}$ (where $l \leq 7$) comprising the probabilistic estimation of the seven fundamental shapes.

An image region r_i is then divided into a set of image patches $b_k \in \mathbb{B}$ where $\mathbb{B}$ is the set of all image patches extracted from image regions in $\Re^I$. The visual color and texture concepts are characterized from image patches by the feature vectors $c_{b_k} = \{p(c_1|b_k), \ldots, p(c_i|b_k)\}$ and $t_{b_k} = \{p(t_1|b_k), \ldots, p(t_j|b_k)\}$ (where $i, j \leq 11$) containing the distribution of each of the eleven color concepts and the probabilistic estimation of the eleven texture categories.

Each image region is then represented by a set of feature vectors characterizing its visual conceptual information. The set of extracted feature vectors from an image region are concatenated in a late fusion fashion (Snoek et al.; 2005). In late fusion the scores of the visual color, texture, and shape concepts are concatenated to learn high-level semantic categories in contrast to early fusion scheme (Snoek et al.; 2005) in which the low-level features are concatenated before any learning process.

These feature vectors are concatenated such that for each image patch b_k of an image region r_i, its corresponding visual color and texture concepts' feature vectors and the global shape of the image region r_i are concatenated generating the feature vector f_{b_k} as

$$f_{b_k} = \begin{bmatrix} c_{b_k} \\ t_{b_k} \\ s_r \end{bmatrix}$$

$f_{b_k} \in \mathbb{F}$ where $\mathbb{F}$ is the set of all feature vectors extracted from all image regions in $\Re^I$ that will be used as the input feature vectors to the supervised learning algorithm.

5.3 Machine learning techniques to address the semantic gap

In any classification problem, instances are represented as vectors of features describing classes in a dataset. The task of a classifier is to learn the correlation between the input features and classes so that to classify new input instances to one of the learned classes. The classification decision can be made based on a proximity measure such as k-Nearest Neighbor, the optimum separation hyperplane as in Support Vector Machine classification, or a classification rule-set using a rule based classifiers such as Decision tree learning algorithms.

Machine learning techniques have been widely used to learn semantic categories from a set of extracted low-level image features (Zhang et al.; 2012). Generally, each feature vector of a multimedia object is associated with a class label indicating the semantic meaning of that object. Machine learning algorithms use these feature vectors to find the correlation between the feature vectors in the feature spaces and their associated class labels. The process of learning the correlation results in a model responsible for classifying un-annotated new input instances.

A number of supervised learning methods have been introduced in the last decade. Which classifier fits best to a problem is an important aspect to consider before starting to generate classifiers for any classification problem. Some classification methods generalize well for a given problem and some, on the other hand, do not.

The main aim of this thesis is to use the visual conceptual information to describe image content and to learn high-level semantic categories. In doing so, the supervised machine learning algorithms will use the visual conceptual information as the input features. Which supervised learning algorithm to consider on this new set of features is crucial. Accordingly, an empirical comparison between a set of supervised learning algorithms is conducted to find which learning algorithm results in a higher classification accuracy in the conceptual feature space.

The following supervised learning algorithms are considered: Support Vector Machines (with different kernels); Decision Trees; Random Forests; Nearest Neighbor; Naive Bayes; and Logistic Regression classifiers. The following sections briefly introduce these learning algorithms and reviews frameworks from the literature using each technique for image annotation and semantic inference of image contents.

5.3.1 Support Vector Machines

Support Vector Machine (SVM) is currently one of the most promising machine learning algorithms. SVM is a strong discriminative classifier that was originally used for binary classification problems (Vapnik; 1998). However, many works have emerged recently proposing methods and techniques to solve the multiclass classification problem such as one-against-one and one-against-all techniques (Hsu and Lin; 2002).

Support vector machines can be considered the most widely used learning algorithm for the task of scene categorization, image and object classification (Csurka et al.; 2004; Zhang, Marszalek, Lazebnik and Schmid; 2006; van de Weijer and Schmid; 2007; Yang et al.; 2007; van de Sande, Gevers and Snoek; 2010; Albatal et al.; 2010; van Gemert et al.; 2010; van de Sande, Gevers and Smeulders; 2010; Escalante et al.; 2010; Sande and Gevers; 2010; Papadopoulos et al.; 2010)

A support vector machine classifier determines which class label is associated with a given N-dimensional feature vector in the feature space.

Given a set of training data $D_{train} = \{(f_1, y_1), (f_2, y_2), \ldots, (f_i, y_i))\}$ where $f_i \in R^n$ is an n-dimensional input feature vector and $y_i \in \{-1, 1\}$ is the associated class label with each input feature vector.

The support vector machine classifier constructs a separation hyperplane as the separation surface that maximizes the margin between the positive and negative examples. The margin denotes the distance between the data points belonging to the classes at each side of the separating hyperplane. The data points at the optimal sides of the margin are called support vectors. The maximum margin is called the *optimal separating hyperplane* and the generalization ability of the SVM classifier depends on the separating hyperplane (Abe; 2005).

In other words, SVM represents the input feature vectors as points in a feature space and it divides the points of different classes with the widest gap possible. New input instances are then mapped to the same feature space and predicted to belong to one of the learned categories.

The original form of support vector machine classification was developed to solve the binary classification problems, in which a linear optimal hyperplane suffices (Vapnik; 1998). However, for a non-linear classification problems, the support vector machine projects the original training data into a higher dimensional feature space using kernel functions to find the maximum-margin hyperplane.

In the case of non-linearly separable problems, projection kernels are used and support vector machines are then based on the resolution of the following optimization problem

$$\min_{w,b,\varphi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \phi_I \text{ subject to } y_i \left(w^T \psi \left(f_i \right) + b \right) \geq 1 - \phi_i \text{ and } \phi_i \geq 0 \qquad (5.3.1)$$

Here, the training vector f_i is set to be in a space of higher dimension (sometimes infinite) through the function ψ. The support vector machine then determines a separating linear hyperplane in this space. $K(f_i, f_j) = \psi(f_i)^T \psi(f_j)$ is the projection kernel and C is the penalty parameter of the error term.

SVM has been extensively used in image retrieval and annotation frameworks. An early use of SVM for semantic classification of images appeared in Chapelle et al. (1999). The authors classify images into 14 semantic categories at the global image level. A classifier is generated for each semantic concept using the one-against-all technique. They use HSV (Hue, Saturation, Value) color histogram as input features to SVM classifiers and a new image is classified by using its feature vector as an input to each of the 14 classifiers. The classifier with the highest probability confidence is selected as the concept to describe the image.

Shi et al. (2004) introduce a region-based classification of image regions into semantic categories using 23 SVM classifiers. An SVM classifier was generated for each semantic class. Csurka et al. (2004) generate SVM classifiers using a bags-of-keypoints representation for the task of image categorization. They point out that the linear kernel gives better results in comparison to quadratic and cubic kernels. Moreover, the overall performance of SVM outperforms naive Bayes classifier using the same features and dataset.

In van de Weijer and Schmid (2007) linear SVM classifiers are trained using the frequency histograms of the visual words of bags-of-words representation using color names and SIFT-descriptors. Zhang, Marszalek, Lazebnik and Schmid (2006) incorporate the Earth Mover's Distance (EMD) and the Chi-square (χ^2) distance into an SVM framework. The generated kernels (i.e., EMD and χ^2 kernels) are used in their evaluation of using the bags-of-features approach with different keypoint detectors and descriptors for the tasks of texture and object categorization. van Gemert et al. (2010) use a non-linear χ^2 kernel for a codebook-based categorization framework. In (van de Sande, Gevers and Smeulders; 2010; Sande and Gevers; 2010) SVM classifiers were utilized for the tasks of visual concepts and annotation. Using Chi-square (χ^2) kernel and the bags-of-words (BOW) representation, they could achieve top ranking for the large-scale visual concept detection tasks.

Escalante et al. (2010) conduct experimental comparison between different learning algorithms to classify image regions into semantic categories. Support vector machine classifier with Polynomial kernel was included. Their results reveal that SVM classifier was outperformed by other classifiers such as Random Forest, Kernel logistic regression, and Naive Bayes.

In the experiments in Chapter 6, different kernels are used and evaluated on the task of classifying image regions using the visual conceptual information. These include Linear, RBF, Polynomials of 2 and 3 degrees, and Sigmoid kernels.

5.3.2 Decision tree learning

Decision tree learning is a rule based learning technique in which a learning system selects only a subset of the instance attributes when forming the classification hypothesis. It is a method for approximating discrete-valued functions. The main advantages of decision tree classifiers are the hierarchical clarity, implementation simplicity, robustness to incomplete and noisy data, and that it can learn from a small number of training examples (Quinlan; 1986). The tree inference is close to the way humans think as the tree is mapped into a set of if-else rules understood by humans (Mitchell; 1997).

Decision trees use a set of labelled training instances to form the classification hypothesis. The learned rules are then formed in a tree structure starting with a single root node to some leaf nodes, each of which represents a classification result of an input instance. To classify new input instances, the instances are sorted down the tree from the root node to the leaf nodes in the generated tree. Each non-leaf node in the tree specifies a test of some attributes of the input instances, and each leaf node specifies a target class for the input instance. Based on the value of the attribute, the instance is classified down to the corresponding branch descending from that node.

Generally, the training instances are represented by many features; and each feature will be tested individually in the tree in the non-leaf nodes. The set of features form the hypothesis by which the tree classifies the input instances. However, an important question to ask here is which attribute should be tested first? The attribute selected first at the root node should be the most significant and discriminant for classifying the input instances. Selecting the most suitable attribute is achieved by statistical methods such as information gain and gain ratio (Han et al.; 2011). These methods measure how well a given attribute separates the training examples according to their target classification. Attributes are then sorted according to their values based on the measure selected.

ID3 (Quinlan; 1986) and C4.5 (Quinlan; 1993) are the most widely used algorithms to construct decision trees. The ID3 algorithm works only with discrete attributes. C4.5 works well with both discrete and continuous attributes. Another possibility is CART (Breiman; 1984) which is used for constructing classification and regression trees. It deals with both discrete and continuous attributes.

The task of the decision tree classifier is to sort the set of all feature vectors in $\mathbb{F}$ to establish the classification rules. Given a labelled training set D_{train} that contains the feature vectors of the high-level conceptual information of all image regions $r \in \Re$, the decision tree should establish the classification rules based on the values of the given attributes. Constructing a decision tree can be viewed as recursive process that starts by placing an attribute at the root node of the tree and then making a branch for each possible value. The split is made by testing the attribute at the root node against some constant. The construction proceeds recursively on each of the child nodes. A test for an attribute is made at each of the child nodes. These nodes are called the 'partitioning nodes' as they are simply testing the attributes and are responsible for splitting. The test and split is performed recursively until the node is reached and the input instance is classified.

As stated earlier in this section, statistical tests are used to select the most significant attributes in the dataset to generate the classification tree. The *information gain* measures the effectiveness of an attribute to classify data. It is based on the *entropy* measure which quantifies the impurity of a training subset. It is calculated as

$$E\left(D_{train}\right) = -\sum_{i=1}^{c} p_i \log_2\left(p_i\right) \tag{5.3.2}$$

where p_i is the proportion of D_{train} belonging to class i of all possible classes c. Thus, the information gain is calculated as

$$Gain\left(D_{train},\ f_k\right) = E\left(D_{train}\right)\ - \sum_{j=1}^{n} \frac{\left|D_{train_{(j)}}\right|}{\left|D_{train}\right|} E\left(D_{train_{(j)}}\right) \tag{5.3.3}$$

where $j \in values(f_k)$ is a value of attribute f_k, and $|D_{train_{(j)}}|$ is a subset of D_{train} for which f_k has the value j, and $|D_{train}|$ is the total number of instances in D_{train}. In the learning algorithm C4.5, the *gain ratio* is also used to select the attributes. The gain ratio divides the information gain by the information provided by the test outcomes. It takes into account the number and the size of the generated sub-trees into which an attribute partitions the dataset. It is measured by incorporating the split information which is actually the entropy of D_{train} with respect to the values of f_k but disregarding all information about the target class. It is calculated as

$$SplitInfo\left(D_{train},\ f_k\right)\ =\ -\sum_{j=1}^{n} \frac{\left|D_{train_{(j)}}\right|}{\left|D_{train}\right|} \log_2 \frac{\left|D_{train_{(j)}}\right|}{\left|D_{train}\right|} \tag{5.3.4}$$

and the gain ratio is defined in terms of the information gain (Eq. 5.3.3) and the split information (Eq. 5.3.4) as follows

$$GainRatio\left(D_{train}, f_k\right) = \frac{Gain\left(D_{train},\ f_k\right)}{SplitInfo\left(D_{train},\ f_k\right)} \tag{5.3.5}$$

In image retrieval, many works have explored the decision tree classification algorithms to classify images and infer their semantics. Huang et al. (1998) construct a classification tree for a hierarchical 11-category classification. They demonstrated that their approach outperforms the traditional nearest neighbor classifier. Sethi et al. (2001) classify outdoor images into four semantic classes using the CART algorithm. A semantic scene classification framework for web-images using the C4.5 decision tree algorithm was introduced by Wong and Leung (2008).

Another system that uses decision tree learning for classifying image regions is the Decision Tree-Semantic Template (DT-ST) (Liu et al.; 2008). This technique avoids complex discretization of attributes - as required for the ID3 decision tree - as it uses predefined values representing the centroid of both color and texture low-level features extracted from sets of regions representing each concept in the training set which they call *Semantic Templates*. These semantic templates are used by the tree as the decision criteria. Zhang et al. (2009) use the semantic template decision tree introduced by Liu et al. (2008). In this work, they use vector quantization to discretize the image features instead of the semantic template decision tree.

In the experiments in Chapter 6, the C4.5 decision tree algorithm is used since the features used are continuous valued features (i.e., the probabilistic estimations of the visual concepts). This has shown to achieve better results in comparison with other decision tree algorithms such as ID3 (Liu et al.; 2008). The experiments using the decision trees for the task of semantic-based inference of image regions are explained in Section 6.4.4.

5.3.3 Random Forest

A Random forest is an ensemble of random decision trees. The decision trees are generated by randomly selecting subsets of features. The final classification output of the random forest is done by voting for the most popular class among the generated decision trees (Breiman; 2001).

As in decision trees, the internal nodes of the randomly generated trees contain some test on the input data to split the data to be classified. An image region is classified by passing through all internal test nodes to some leaf nodes that contain the probabilistic estimation over image classes. The class with most votes is selected as the final classification.

The training set D_{train} is randomly split into smaller subsets. Each subset is used to train a new random tree. The learning in random forests is recursive in which each node splits the subset D into left and right partitions (two partitions) according to a given test.

The split criterion is evaluated by optimizing the information gain

$$\triangle G = -\frac{|D_l|}{|D|} E\left(D_l\right) \frac{|D_r|}{|D|} E\left(D_r\right) \tag{5.3.6}$$

in which D_l is the left partition and D_r is the right partition, the number of samples in a set is denoted by $|.|$, and $E\left(D\right)$ is the entropy measure (cf. Eq. 5.3.2).

The subset with the highest information gain is then used first to perform the split. This process is repeated using the features in the subset from that node. The process stops when the number of features is very small and cannot be further split or when it reaches a given depth (Bosch et al.; 2007).

Sharp (2008) present an implementation of decision trees and random forests on a GPU (Graphics Processing Unit). Their implementation is evaluated on the task of object recognition. A random forest is used to classify color features in a skin segmentation and detection framework in Khan et al. (2010). Uijlings et al. (2009) use random forests in a bags-of-words approach to project the image descriptors into visual vocabulary rather than k-means clustering to generate the codebook frequency histogram. k-means algorithm classifies the input data into a user specified number of k clusters (Wu et al.; 2008). Each cluster is identified by its centroid and new data points are then classified to the nearest centroid. The centroid of each k cluster is re-calculated as the mean of data points belonging to that cluster.

Bosch et al. (2007) classify images into object categories using random forests. In Moosmann et al. (2008), the authors introduce 'Extremely Randomized Clustering Forests' (ERC-Forests) which are ensembles of clustering trees generated at random to cluster image descriptors for the bags-of-words representation. They show that their method obtains better results than the k-means clustering algorithm. Shotton et al. (2008) introduce the Semantic texton forests, which are randomized decision forests, for the task of image categorization and semantic segmentation. Leistner et al. (2010) present a multiple-instance learning method based on randomized trees. The efficiency of this model is demonstrated on Corel Image Datasets for region-based image classification and also for the task of visual object tracking.

In (Escalante et al.; 2010) experimental comparison between different learning algorithms was conducted to classify image regions into semantic categories. The results show that Random Forest classifier has achieved the highest scores in means of correct classification in comparison to other classifiers such as Support vector machine, Kernel logistic regression, and Naive Bayes.

Forests of different sizes are generated to classify image regions using the visual concepts as presented in Chapter 6, Section 6.4.4. Experimental settings of the Random Forest classifiers are presented in Section 6.4.3.

5.3.4 Nearest Neighbor

Nearest Neighbor is one of the early classification solutions and can be considered the easiest to implement. It is a non-parametric supervised learning technique in which the classification decision is made based on the data itself without the need of a training stage, as in Support Vector Machines or Neural Networks. Nearest Neighbor classifies a feature vector f to the most common class in the set of k training feature vectors that are the nearest to f. This classification technique relies on the nearest neighbor distance estimation and it uses all the training data as templates for classification, hence it is also known as instance-based learning or "lazy learners" (Mitchell; 1997).

For a given new input vector, the nearest neighbor classifier searches for the nearest template in the feature space that minimizes the distance to the new input vector as

$$y^* = \arg\min_j d\left(f^{new}, f_j^{train}\right) \tag{5.3.7}$$

where the distance $d(,)$ is computed by similarity/dissimilarity measures such as χ^2, L_2, and Minkowski derivatives (cf. Eq. 2.3.1) (e.g., Euclidean distance, Manhattan, fractional, etc.) (Hu et al.; 2008; Zhang and Lu; 2003b).

Equation 5.3.7 represents the simplest implementation in which a new instance is classified to the nearest template in the training dataset. A more complicated form considers k nearest neighbors. For a given input vector, the k-Nearest neighbor classifier searches for k templates in the training instances and classifies the input vector into the class with the maximum number of occurrences.

Ladret and Guérin-Dugué (2001) apply kNN for image classification by using DCT (Discrete Cosinus Transform) as the distance measure. Another use of the kNN was proposed by Wu and Manjunath (2001). They present an adaptive search scheme for computing nearest neighbors of a given query in the context of relevance feedback.

Iqbal and Aggarwal (2002) use kNN to identify images that contain man-made objects such as buildings, bridges, and other architectural objects. In Cheng and Chen (2003), images are segmented based on color and texture features and the kNN classifier is used for classifying image regions into 20 categories.

In Zhang, Berg, Maire and Malik (2006) a hybrid method combining both Support Vector Machines and nearest neighbor classifiers to deal with multiclass classification problems is presented. This method computes the distances of a query image to all training examples, and picks the nearest k neighbors. If all the k neighbors have the same label, the query image is given that label; if not, the pairwise distances between k neighbors are computed and the distance matrix is converted to a kernel matrix and multiclass SVM is applied. Xiao et al. (2006) use kNN in a two-stage classification approach using features extracted globally from an image and then features extracted from image regions within that image. kNN classifiers are used at both stages and final annotations are obtained by fusing both the region and image level annotations.

k-Nearest Neighbor classifiers are widely used in image classification and object recognition systems. Nonetheless, it has been argued that its performance is not competitive with parametric techniques such as Support Vector Machines and Artificial Neural Networks. Boiman et al. (2008) argue that the capabilities of non-parametric image classification methods, in particular the nearest neighbor classifier, have been undervalued. They argue that two practices have contributed to the degradation of the performance of this non-parametric image classifier, which are:

- descriptor quantization (i.e., reducing the dimensionality of vectors to produce compact image presentation). Even though this process is essential for some classifiers, it is harmful for the non-parametric classifications as there is a training phase that may compensate for the loss of data due to the quantization process; and
- the use of image-to-image distance instead of image-to-class distance. Basically, nearest neighbor classifiers employ the descriptor distribution of each individual image separately. In their implementation they compute the distance between the query image and the entire class C (using all images). They claim in doing so, they get better generalization capabilities than employing the individual "image-to-image" measurements.

In Nilsback and Zisserman (2006) a nearest neighbor classifier is employed for the task of flower classification using bags-of-visual-words descriptors. In Battiato et al.

(2008), a nearest neighbor classifier is used on the task of image categorization using the bags-of-words representation.

In Chapter 6, kNN classifiers are generated with various numbers of the nearest neighbors k. The Euclidean and Manhattan distances are used to generate the kNN classifiers. Experimental setting are detailed in Section 6.4.3.

5.3.5 Naive Bayes

Among the machine learning algorithms used on the task of image indexing and annotation, Bayesian methodologies have been widely used and shown good results in comparison to other machine learning algorithms. A number of approaches to classify images use probabilistic Bayesian classifiers (Cox et al.; 2000; Vasconcelos and Lippman; 2000b; Vailaya et al.; 2001; Luo and Savakis; 2001; Fergus et al.; 2003; Csurka et al.; 2004; Jin et al.; 2004; Lim and Jin; 2005; Fei-Fei et al.; 2007; Behmo et al.; 2010; Liu et al.; 2010; Papadopoulos et al.; 2010).

Luo and Savakis (2001) classify images into indoor/outdoor using Bayesian networks. In Vailaya et al. (2001), images are classified as broad indoor/outdoor. The outdoor images are further classified into landscape, city, etc. Csurka et al. (2004) use a naive Bayes classifier using a bags-of-keypoints representation for the task of image categorization.

Naive Bayes classifiers are simple, efficient, and robust to noise and irrelevant attributes (Yang and Webb; 2001). In some classification problems this technique has outperformed other more complex classifiers such as decision trees (Domingos and Pazzani; 1997; Kononenko; 1990).

This classification algorithm is based on Bayes' rule that assumes that all features are conditionally independent of one another given the class. Also, the naive Bayes classifier assumes that all assumptions are explicitly built using a set of input feature vectors.

The naive Bayes classifier is constructed using Bayes' theorem which calculates the probability of each semantic concept $y \in \mathbb{S}$ given an instance (i.e., a feature vector) $f \in \mathbb{F}$:

$$p(y|f) = \frac{p(y)\, p(f|y)}{p(f)} \tag{5.3.8}$$

where the class prior $p(y)$ is estimated from the training set and $p(y|f)$ is the unknown probability estimation of the joint distribution of features f and the semantic classes y.

$p(f|y)$ is estimated by following Bayes theorem, which assumes that the attributes are independent given the class label

$$p\,(f_1,\; f_2,\; \cdots\;,\; f_n \mid y\;) \;=\; p\,(f_1|y)\;\; p\,(f_2|y)\;\; \ldots\;\; p\,(f_n|y) \tag{5.3.9}$$

where f_d is the d^{th} dimension of the feature vector containing the probabilistic estimations of the color, texture, and shape concepts in f. Even though this is a naive independence assumption, naive Bayes has shown comparable results with other classifiers (Domingos and Pazzani; 1997; Kononenko; 1990). Moreover, this assumption is feasible with the feature set used for the experiments as each attribute describes a specific distribution of a color, texture, or shape concept. Thus, the naive Bayes classifier can be written as

$$p\,(y \mid f) = p\,(y) \prod_{m=1}^{N} p\,(f_m \mid y) \tag{5.3.10}$$

and the classification rule of a new input instance f_{new} becomes

$$y^{*} = \underset{y \in \mathbb{S}}{\arg\max}\;\; p\,(y) \prod_{m=1}^{N} p\,(f_m^{new} \mid y) \tag{5.3.11}$$

By using this classifier, the image classification task can be stated as: given a set of visual concepts of an image region r, classify (i.e., annotate) r with the semantic class of the highest probability.

The classification task at hand involves quantitative (i.e., continuous) attributes, which are the distribution of the color concepts, the probability estimation of the texture concepts, and the probability distributions of the fundamental shape concepts of an input image object. In naive Bayes classifiers, quantitative attributes should be discretized so that the Bayesian classifier can calculate the prior-probability of classes (Yang and Webb; 2009).

A number of discretization methods have been proposed in the literature (Yang and Webb; 2002; Kotsiantis and Kanellopoulos; 2006; Liu et al.; 2002). In the experiments in Chapter 6, the entropy minimization discretization (EMD) algorithm (Fayyad and Irani; 1993) is used to discretize the continuous attributes as it has shown good results in comparison to other discretization methods (Liu et al.; 2002).

5.3.6 Logistic Regression

Logistic regression is a well-known classifier and widely used for binary classification problems. Logistic regression is a linear classifier and has been used in image classification and annotation problems such as in Magalhães and Rüger (2006); Hoi et al. (2009); He and Jia (2010). It is used for predicting the probability that a feature vector f belongs to class y_i.

Given a new image feature vector f from the test set D_{test}, logistic regression models the conditional probability that feature vector f belongs to class y as

$$p(y|f) = \frac{1}{1 + e^{-y\theta^T f}} \qquad (5.3.12)$$

where $y \in \{-1, 1\}$ is the class label, $\theta = (\theta_1, \theta_2, \ldots, \theta_n)$ are the classifiers weights assigned to input features.

Magalhães and Rüger (2006) learn image semantics from codebooks and keywords using a logistic regression classifier. They generate a hierarchy of cluster models corresponding to a hierarchy of vocabularies by clustering the dataset using a hierarchical expectation maximization (EM) algorithm based on a Gaussian mixture model. A batch mode active learning algorithm based on kernel based logistic regression is proposed in Hoi et al. (2009). In He and Jia (2010), a regularized logistic regression, using the L_1 norm regularization, is used to learn the correlation between image features and high-level semantic concepts. The L_1 norm is applied to allow conducting feature selection and classification simultaneously.

Regularized logistic regression classifiers are generated with several ridge values as presented in Section 6.4.3. Experimental results are reported in Section 6.4.4 of Chapter 6.

5.4 Summary

This Chapter has presented the theoretical formalization and background of the supervised learning algorithms used in the body of this thesis to classify image regions into high-level semantic categories. Section 6.4 of Chapter 6 presents the experiments conducted to compare these machine learning algorithms and it shows the performance of these learning algorithms.

Chapter 6

Experimental Validation

6.1 Introduction

This chapter presents the experiments carried out in validating the work presented in the preceding chapters of this thesis. First, Section 6.2 presents and reviews the implementation and experimental validation of frameworks aiming at conceptualizing the low-level color and texture features into visual concepts presented in Chapter 3. Section 6.3 presents the implementation and the experiments performed on the framework for characterizing and extracting the visual shape concepts presented in Chapter 4. Integration of the visual color, texture, and shape concepts to describe and classify image regions into high-level semantic categories is presented in Section 6.4. Finally, experiments on retrieval of image regions incorporating their perceptual properties is demonstrated in Section 6.4.6

6.1.1 Performance evaluation policy

There are several common approaches to measuring the relative effectiveness of query strategies based on *a priori* knowledge of the content of the database and the user's intent; these metrics are useful for evaluating experimental scenarios and refining the process of selecting query parameters. This section summarizes the performance metrics used in the experimental evaluation in this chapter. The evaluation metrics used are: precision, recall, and F-score measures. This section starts by illustrating the confusion matrix from which these measures are derived.

Confusion Matrix

The confusion matrix is a well-known method for classification systems. It contains all the information about the actual (the original class label) and predicted classification assigned by the classification method. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The performance of a classification system is then evaluated using the information obtained by the confusion matrix. Table 6.1 shows an example of a two-class classification problem.

		The predicted class	
		C1	***C2***
The actual class	***C1***	tp (true positive)	fp (false positive)
	C2	fn (false negative)	tn (true negative)

Table 6.1: *An example of confusion matrix of a two-class classification problem*

The contents of the confusion matrix are interpreted as follows

- ***tp (true positive)*** is the number of correct predictions that an instance is positive;
- ***fp (false positive)*** is the number of incorrect predictions that an instance is negative;
- ***fn (false negative)*** is the number of incorrect predictions that an instance is positive; and
- ***tn (true negative)*** is the number of correct predictions that an instance is negative.

The confusion matrix is used to derive up to six standard performance metrics. The most straight forward metric is the **accuracy**, which measures the proportion of the total number of predictions that were correctly classified. It is measured as follows

$$Accuracy = \frac{tp + fp}{tp + fp + fn + tn} \tag{6.1.1}$$

Precision and Recall

The precision and recall metrics are widely used evaluation metrics in information retrieval. They have been applied to the content-based image retrieval to measure the retrieval accuracy as well.

In the classification context, the precision and recall are measured using the confusion matrix (cf. Table 6.1). In classification, *precision* measures the accuracy such that a class has been predicted correctly. It is measured as

$$Precision = \frac{tp}{(tp + fp)} \tag{6.1.2}$$

where tp and fp are true positive and the false positive predictions for a specific class. These are extracted from the confusion matrix presented in Table 6.1.

The *recall* measures the ability of a classifier to select instances of a specific class from a dataset. It is commonly referred as the sensitivity of a classifier and corresponds to the true positive rate. It is measured as

$$Recall = \frac{tp}{(tp + fn)} \tag{6.1.3}$$

where tp and fn are the true positive and false negative predictions for a specific class. The denominator, $tp + fn$ is the total number of the test instances of a specific class.

F-score

The F-score measure is a harmonic mean of the precision and recall metrics, and is a popular evaluation measure in the research area of information retrieval. The advantage of the F-score is that it combines the precision and recall scores in a single value, and hence it helps in conducting statistical tests by considering a single F-score value rather than processing two values.

Formally, given the count of the true positives (tp), true negatives (tn), false positives (fp), and false negatives (fn), F-score is computed as follows:

$$F - score = \frac{2 \times tp}{2 \times tp + fp + fn} \tag{6.1.4}$$

6.2 Automatic characterization of the visual conceptual information

This section presents the processes of extracting the visual color and texture concepts. The implementations of the adopted frameworks are illustrated in detail in Sections 6.2.1 and 6.2.2.

6.2.1 Extracting the visual color concepts

The model for characterizing the conceptual color information is based on the work presented in (Gong et al.; 1996). Given a series of perceptive evaluations and observations, eleven color categories are defined, each described by a tonality (angular orientation in the HVC color space), a brightness value (in the $[1, 10]$ interval) and a decimal saturation value (in the $[0, 30]$ interval).

For example, the color **green** is represented by an angular orientation in the interval $[112, 196]$, a brightness value in the interval $[4, 10]$, and a decimal saturation value in the intervals $[1.5, 30]$. The color **cyan** takes an orientation value in $[196, 256]$, a brightness value in $[6, 8]$, and a decimal saturation value in $[1.5, 30]$. The color **gray** is defined either by a brightness value in the $[3, 4]$ interval, or a decimal saturation value strictly lower than 1.5 and a brightness value in the $[4, 8]$ interval. **Red**, **orange**, **skin**, **yellow**, **purple**, **black**, **white**, and **blue** are the other color categories.

The ranges of each color zone are listed in table 6.2

Color name	Hue (degree)	Value	Chroma
Red	$0 - 36$	$4 - 9$	$1.5 - 30$
	$36 - 64$	$4 - 9$	$15 - 30$
Orange	$64 - 112$	$4 - 8$	$9 - 30$
Yellow	$80 - 112$	$9 - 10$	$1.5 - 30$
Skin color	$36 - 64$	$4 - 9$	$1.5 - 15$
	$64 - 112$	$4 - 8$	$1.5 - 9$
Green	$112 - 196$	$4 - 10$	$1.5 - 30$
Cyan	$196 - 256$	$6 - 8$	$1.5 - 30$
Blue	$256 - 312$	$4 - 8$	$1.5 - 30$
Purple	$312 - 359$	$4 - 8$	$1.5 - 30$
Black	—	< 3	—
Gray	—	$4 - 8$	< 1.5
	—	$3 - 8$	—
White	—	< 9	< 1.5

Table 6.2: *Ranges of each color zone*

The interest of characterizing color in a human perceptual space is justified by the fact that efficient human interaction when specifying image retrieval frameworks is crucial and therefore aspects related to human perception are to be taken into account.

Characterizing the aforementioned visual color concepts (i.e., symbolic color categories) involves transforming the extracted low-level color features specified in the RGB space (primary step for low-level color extraction) to tonality, brightness, and saturation values in the perceptually uniform HVC space.

The first step in characterizing the visual color concepts involves transforming the RGB values extracted from an image region into the HVC (Hue, Value, Chroma) color space. The transformation process from the RGB triples to coordinates in the HVC space is adapted from the algorithm described in Gong et al. (1996):

1. The first step involves transforming the coordinates in the RGB color space into (X, Y, Z) components such that:

$$X = 0.607R + 0.174G + 0.201B \tag{6.2.1}$$

$$Y = 0.299R + 0.587G + 0.114B \tag{6.2.2}$$

$$Z = 0.066G + 1.117B \tag{6.2.3}$$

2. The second step transforms the (X, Y, Z) to $(M_1,\ M_2,\ M_3)$ such that:

$$M_1 = 11.6\left[\left(\frac{X}{X_0}\right)^{\frac{1}{3}} - \left(\frac{Y}{Y_0}\right)^{\frac{1}{3}}\right] \tag{6.2.4}$$

$$M_2 = 0.4 \times 11.6\left[\left(\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}}\right) - \left(\left(\frac{Z}{Z_0}\right)^{\frac{1}{3}}\right)\right] \tag{6.2.5}$$

$$M_3 = 0.23\left[\left(11.6\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}}\right) - 1.6\right] \tag{6.2.6}$$

where X_0, Y_0, and Z_0 represent the values of $X, Y,$ and Z for the color reference white.

3. The components in the (H, V, C) space are then determined from $(M_1,\ M_2,\ M_3)$

$$H^{'} = \arctan\left(\frac{M_2}{M_1}\right) \tag{6.2.7}$$

$$S_1 = M_1\left(8.88 + 0.966 \times \cos(H^{'})\right) \tag{6.2.8}$$

$$S_2 = M_2\left(8.025 + 2.558 \times \sin(H^{'})\right) \tag{6.2.9}$$

$$H = \arctan\left(\frac{S_2}{S_1}\right) \tag{6.2.10}$$

$$V = 11.6\left[\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}}\right] - 1.6 \tag{6.2.11}$$

$$C = \sqrt{S_1^2 + S_2^2} \tag{6.2.12}$$

Components H, V, and C correspond to the values of tonality, luminosity, and saturation. They are then mapped to the eleven color concepts introduced in Section 3.2.1. This process is iterated over all pixels in an image region to obtain the pixel percentage of each color concept, which constitutes the 11-dimensional feature vectors representing the distributions of the visual color concepts.

6.2.2 Characterizing the visual texture concepts

Texture analysis has been widely studied in the fields of computer vision and image processing. This has resulted in the identification of several algorithms aiming at extracting the low-level texture features in several content-based image retrieval architectures. However, these texture extraction algorithms do not capture aspects related to human perception.

The visual texture concepts are characterized from image regions using the work presented in Belkhatir (2005). This work aims at capturing aspects related to human perception of texture patterns and is inspired by Leow and Lai (1999), in which a computational model that is the closest approximation to human visual systems is presented.

The first step in characterizing the visual texture concepts involves extracting the low-level texture features. Here, Gabor features are extracted at different scales and orientations. Each image region is characterized by its Gabor energy distribution within five scales covering the whole spectral domain and six orientations.

Gabor texture features are extracted from image regions by applying Gabor filters such that

$$g(x,y) = \frac{1}{2\pi\lambda\sigma^2} \exp\left[-\frac{(\frac{x'}{\lambda})^2 + y'^2}{2\sigma^2} \exp\left(2\pi f j x'\right)\right] \tag{6.2.13}$$

where $(x', y') = (x\cos\theta + y\sin\theta, -x\sin\theta + y\cos\theta)$ are rotated coordinates at the angle θ from the x-axis, λ is the aspect ratio, and σ is the scale parameter.

Gabor filters are applied on an image region $I(x,y)$ with different frequencies f and orientations θ

$$e_{c,f\theta}(x,y) = I(x,y) * g_{c,f\theta}(x,y) \tag{6.2.14}$$

$$e_{s,f\theta}(x,y) = I(x,y) * g_{s,f\theta}(x,y) \tag{6.2.15}$$

where $g_{c,f\theta}(x,y)$ and $g_{s,f\theta}(x,y)$ are the real and imaginary components of Gabor filters (Leow and Lai; 1999). Gabor channels output energy is computed as

$$E_{f\theta}(x,y) = e^2_{c,f\theta}(x,y) + e^2_{s,f\theta}(x,y) \tag{6.2.16}$$

Gaussian filters are then used to smooth the channels' output to remove the local variation created by the sinusoidal terms in the Gabor functions after applying Gabor filters. Further normalization is achieved by dividing the energy value of each channel by the largest value at that location (i.e., the channels' output at each pixel location)

$$G_{f\theta}(x, y) = \frac{E_{f\theta}(x, y)}{\max\limits_{f,\theta} E_{f\theta}(x, y)} \tag{6.2.17}$$

The normalized Gabor outputs form the a multi-dimensional texture feature vectors.

The eleven visual texture concepts presented in Section 3.2.2 are characterized through a learning framework that is based on Support Vector Machine (SVM) classification (described in Section 5.3.1) mapping Gabor feature vectors to visual texture concepts. Eleven SVM classifiers were generated using the one-against-all technique to solve the multi-class classification problem. The SVM classifiers were generated using the Radial Basis Function (RBF) kernel. A posterior recognition probability for the classification is assigned by applying a Sigmoid function to the output of SVM classifiers as Platt (1999) suggested.

A dataset of $10,000$ texture images used in Leow and Lai (1999) was used to train the classifiers. As demonstrate in Belkhatir (2005), each of the eleven texture categories was trained with a distribution of $900 - 1000$ texture images. The performance of the classifiers is measured by using cross validation technique in which the dataset is split into n subsets of equal size. The number of folds (n) used in the experiment is not reported in the original work (Belkhatir; 2005). An SVM classifier is trained with $n - 1$ subsets and tested with the remaining one. Thus, each instance in the training set is predicted once so that the cross-validation accuracy is the percentage of instances that are correctly classified. Table 6.3 lists the cross-validation accuracies of the generated classifier for each of the 11 texture concepts.

Finally, a grid-search is applied to find the optimal parameters (i.e., the cost C and γ) to train the SVM classifiers. The parameters achieving the best cross-validation accuracy were used to generate the classification models.

The framework for characterizing the visual texture concepts was trained using a texture-based dataset (Leow and Lai; 1999) and not using a set of images from the dataset used in context of this thesis (i.e., SAIAPR TC-12 dataset). A comparison between using texture-based dataset and a natural images dataset to characterize the visual texture concepts is out of the scope of this work.

Texture category	Cross-Validation (%)
Bumpy	83.7
Cracked	85.2
Disordered	88.9
Interlaced	91.9
Lined	94.5
Marbled	98
Netlike	86.8
Smeared	83.4
Spotted	90
Uniform	97.3
Whirly	81.4

Table 6.3: *SVM cross-validation rates of the models for characterizing the visual texture concepts*

6.3 Experimental validation of the framework for the extraction and characterization of the conceptual shape information

6.3.1 Overview

This section presents the experimental validation of the framework for the extraction and characterization of the conceptual shape information presented in Chapter 4. The experiments presented in this section answers the first research question of this thesis that is *Can shapes of segmented image regions be represented using high-level perceptual descriptors rather than low-level shape features for the task of image description and classification?*.

The accuracy of the conceptual shape characterization is computed through 5-fold cross validation using the SVM generated classifiers as explained in Section 6.3.3. Then, the effectiveness of the proposed framework is demonstrated by instantiating the shape concepts within a conceptual image retrieval architecture. This architecture is compared to a content-based image retrieval architecture based on a query-by-example process operating on the low-level shape features. Subsequently precision and recall evaluation metrics are presented that show just how much this approach outperforms the query-by-example architecture.

The rest of this section is organized as follows. Section 6.3.4.1 presents the accuracy of mapping the low-level shape features to shape concepts. The process of expanding the dataset of all shapes in the shape lattice to increase the generalization ability of the classifiers is illustrated. Then, the cross-validation rates of the generated classifiers trained with the expanded dataset are presented. Section 6.3.4.2 illustrates the framework for symbolic shape retrieval that allows users to retrieve shapes using shape concepts. Experiments show that the precision and recall values obtained from the symbolic shape retrieval framework outperform those from the content-based image retrieval architecture.

6.3.2 Low-level shape features extraction

To characterize shapes and to map them to the corresponding shape concepts, the contour information is needed. In this framework, Fourier descriptors are utilized to extract the low-level shape features. Fourier descriptors exploit the region boundary information.

Fourier descriptors were selected as it is used to describe the boundary information (i.e., contours) of image regions (Nixon and Aguado; 2008). There are other algorithms that have proven to be strong in characterizing shapes (as well as texture features) in images. For example, Scale invariant feature transform (SIFT) (Lowe; 1999) and Speeded up robust features (SURF) (Bay et al.; 2008). However, SIFT and SURF would be more useful to be used in image classification systems in which entire non-segmented images are processed and SIFT or SURF are characterized over a set of interest points extracted from images. These approaches are mainly utilized in image classification systems and for detection of visual concepts in images and videos. Since the aim of this work is to conceptually describe the contours of segmented image regions, then using boundary-based shape algorithms is more suitable.

The main idea of Fourier descriptors is to characterize a contour by a set of numbers that represent the frequency content of the whole shape (Nixon and Aguado; 2008). Fourier descriptors have many advantages over other contour descriptors such as their simple derivation and normalization, straightforward matching and robustness to noise. They are, moreover, compact and perceptually meaningful descriptors. These properties make it a popular shape descriptor which, furthermore, outperforms other contour shape descriptors (Zhang and Lu; 2004).

The Fourier descriptors method provides the image region representation in the frequency domain by applying a discrete Fourier transform on its shape signature function as given in equation 6.3.1

$$F(d) = \frac{1}{N} \sum_{k=0}^{N-1} f(k) \exp^{\left(\frac{-j\pi\, d\, k}{N}\right)} \qquad d = 0,\ 1,\ 2,\ \ldots,\ N-1 \tag{6.3.1}$$

where $f(k)$ is the shape signature function, and N is the number of samples of signature function $f(k)$.

In general, the shape signature is any 1-dimensional function that represents 2-dimensional areas or boundaries to uniquely describe shape. Many shape signature functions exist in the literature such as the complex coordinates function, centroid distance function, chord length signature, cumulative angular function, and curvature signature (Zhang and Lu; 2002a). Here, the centroid distance function is used to extract the shape signatures as it has been shown to achieve the best and the most accurate results among other shape signatures in terms of computational complexity, robustness, convergence speed of its Fourier series, and the retrieval performance of its Fourier descriptors (Zhang and Lu; 2002a).

The Centroid distance function computes the distance between the boundary points, $(b_0, b_1, \ldots, b_i)$ and the centroid of shape as illustrated in equation 6.3.2.

$$f(b_i) = \sqrt{[x(b_i) - x_c]^2 + [y(b_i) - y_c]^2}, \quad b_i = 0,\ 1,\ 2,\ \ldots,\ N-1 \tag{6.3.2}$$

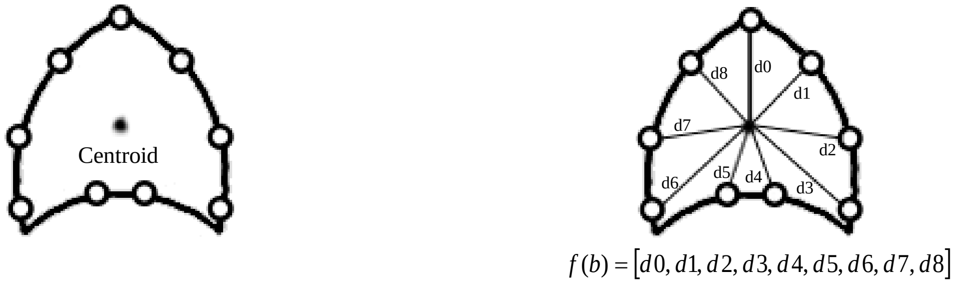
Here, (x_c, y_c) are the coordinates of the centroid of the image region. They are computed as the average of the shape boundary coordinates as equations 6.3.3 and 6.3.4 show

$$x_c = \frac{1}{N} \sum_{b_i=0}^{N-1} x(b_i) \tag{6.3.3}$$

$$y_c = \frac{1}{N} \sum_{b_i=0}^{N-1} y(b_i) \tag{6.3.4}$$

where N is the number of boundary points.

An example of the applications of the centroid distance function is shown in Figure 6.1. Figure 6.1-(a) shows the centroid and the sampled boundary points of the given shape, and Figure 6.1-(b) illustrates the distance between the centroid and the sampled boundary points of the shape.



(a) Shape with 9 sample points on the boundary.

(b) The centroid distance function of this shape.

Figure 6.1: *Centroid distance function*

In practice, the same object may have different sizes which means that the number of boundary points of the object may differ from one size to another. For matching purposes, the shape boundary or the shape signature of both the queried objects and models in the database should be sampled so that they have the same number of data points (Zhang and Lu; 2003a). To reach this objective, shapes should be sampled in a manner that yields a fixed number of boundary points. In this framework, the *equal point sampling* method is used to sample the data points of shapes.

The equal point sampling method selects candidate points spaced at an equal number of boundary points along the shape boundary. The space between two consecutive candidate points is given by $\frac{L}{N}$, where L is the boundary length and N is the total number of candidate points to be sampled. Here, shapes are sampled to 64 candidate points since it is a power-of-two integer which facilitates the use of the fast Fourier transform (FFT).

The extracted Fourier descriptors are shift and translation invariant since the signatures of shapes are captured using the centroid distance function that is translation invariant. Indeed, the centroid distance function is translation invariant as it does not record the exact coordinates of the boundary points but rather it records the distances between the center and the boundary points of a shape (Zhang and Lu; 2003a).

However, further normalization procedures are required to normalize the object so that it becomes rotation and scaling invariant. In other words, to achieve rotation and scaling invariance, normalization procedures are required to normalize the object such that its boundary has a standard size and orientation. Here, rotation normalization is achieved by taking only the magnitudes of the Fourier coefficients (i.e., $|F(d)|$) and ignoring the phase information. Furthermore, $|F(0)|$ reflects the energy of the centroid function. Thus the scaling normalization of the object's Fourier descriptors is achieved by dividing the magnitude of all $F(d)s$ by the magnitude of $F(0)$.

6.3.3 Mapping the low-level shape features to shape concepts

In order to determine which shape concept is associated with a given image region represented in a 64-dimensional Fourier descriptor, there exist a set of points $\{x_1, \ldots, x_{64}\}$ in the 64-dimensional input space of shape concepts and a set of labels $\{y_1, \ldots, y_i, \ldots, y_{898}\}$ such that the y_i value equals 1 if x_i corresponds to shape concept $c_{shape}[i]$ and -1 otherwise. The goal is to determine a function $f : S^N \rightarrow \{\pm 1\}$ that associates each point with its corresponding label. This function shall provide good results on the training set and be capable of generalizing on shapes which are not characterized with shape concepts.

Support Vector Machines learning is adopted for this framework to classify image regions into visual shape concepts. An overview of the theoretical foundation of SVM learning is presented in Section 5.3.1.

Among the possible kernels (linear, polynomial, radial basis function, sigmoid, . . .), the radial basis function (RBF) is chosen:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2),\ \gamma > 0$$

where γ is a kernel parameter. It is traditionally used in the case of non-linearity between the class labels and the input attributes. It holds several advantages with respect to other kernels, in particular it does not require fine-tuning of hyper-parameters and its computational complexity is reduced (Platt; 1999).

Support vector machines in their initial formulation are used in discrimination problems involving two classes. To solve the multi-class learning problem in this framework, the "one-against-rest" approach is implemented to generate the classifiers for reasons of optimized inter-class separation (Rifkin and Klautau; 2004).

However, this approach results in the specification of classifiers which generate a binary output. The aim is to associate a confidence value for the proposed classification. For this, the probabilistic estimation of the generated classifiers is considered (Platt; 1999). It uses a logistic function of the form $P(y_i = 1|f) = \frac{1}{1+\exp(Af+B)}$ where f is the support vector machines for input x_i and $y_i = \pm 1$ represents the class label. This doubly parametric function allows linking outputs of support vector machines to corresponding posterior probabilities. This method implies solving a non-linear optimization problem involving the pair of parameters (A, B) such that $\sum\limits_i t_i log(p_i) + (1-t_i) log(1-p_i)$ is minimized; where $p_i = \frac{1}{1+\exp(Af_i+B)}$ is the inferred posterior probability and $t_i = \frac{y_i+1}{2}$ is the target binary coding for the pair (x_i, y_i)

6.3.4 Framework evaluation

6.3.4.1 Accuracy of mapping the low-level shape features to shape concepts

In order to maximize the accuracy and generalization ability of the generated classifiers, a considerable amount of training data is needed. To have such a dataset, the shape lattice is expanded by applying a set of affine transformations to all the shapes in the lattice generating a total of 145,466 shapes. The affine transformations applied to the shapes are rotation, scaling, horizontal and vertical flipping with different parameters and scales. The resulting experimental dataset is a publicly available contribution of the work since there is no equivalent collection in the literature.

The dataset is split with 20% (35,151 shapes) randomly selected for testing and the remainder (110,315 shapes) for training the framework of shape-based image retrieval. Figure 6.2 shows some transformed shapes generated by applying affine transformations to the shape "rectangular with three constricted inward and one sharpened inward edges".

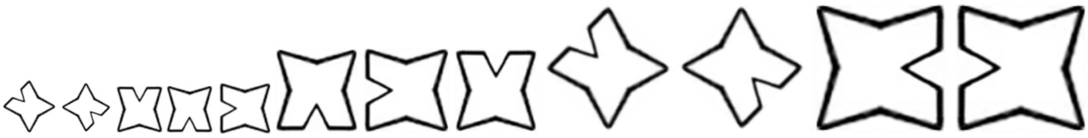


Figure 6.2: *The shape "rectangular with three constricted inward edges and one sharpened inward edge" modified with different rotation, scaling, and flipping transformations*

To determine the performance and generalization ability of the mapping between low-level features and shape concepts through support vector machines, n-fold (here $n = 5$) cross validation is applied to the testing set. In cross validation the training set is divided into n subsets of equal size. Out of these, one subset is tested using the classifier trained by the remaining $n - 1$ subsets. Accordingly, each instance in the training set is predicted once, and consequently the cross validation accuracy is the percentage of the correctly classified data. The cross validation procedure prevents the overfitting problem by evaluating the fitting provided by each parameter value set in the search process for the parameters C and γ. The best parameters are achieved with the best cross-validation accuracy. Table 6.4 lists the cross validation rates for the shapes generated from each of the basic shape concepts. The obtained values are all higher than 94%.

Shape category	Cross-Validation (%)
Circular	98.4
Oval	97.9
Elliptical	98.9
Pandurate	95.1
Rectangular	97
Pentagonal	94.4
Triangular	98.9

Table 6.4: *SVM cross-validation rates of the models for characterizing the visual shape concepts*

6.3.4.2 Shape-based symbolic image retrieval

To carry out this experiment, the shape concepts are instantiated within a conceptual image retrieval architecture. Its evaluation is based on the notion of image relevance, which consists of quantifying the correspondence between index and query shape image representations.

For comparison purposes, a content-based image retrieval architecture using query-by-example scheme was developed (cf. Section 2.2). This framework operates on the low-level shape features. Index and query representations are compared based on the Squared Chord distance which has shown competitive performance over other distance measure techniques (Hu et al.; 2008).

This experiment is evaluated using 35 queries, selected randomly at equal distribution between shape categories, which correspond to shapes from the shape lattice. As far as the conceptual image retrieval framework is concerned, these are translated as text-based descriptors based on the highlighted shape concepts such as “circle”, “oval with sharpened head and truncated base”, “rectangle with incurvated sides”. For the baseline QBE architecture, three example shape images were selected randomly for each query as an input to the query-by-example module.

The precision and recall indicators are used to evaluate the performance of the compared architectures (cf. Section 6.1.1).

However, in order to measure the overall performance of the system, the precision and recall values are averaged for queries corresponding to shapes inheriting from the same basic shape. Here the macro averaged precision and recall are used. These indicators are computed by calculating the precision and recall values for each shape concept and then taking their average based on the corresponding basic shape. The macro-averaged precision (Eq. 6.3.5) and recall (Eq. 6.3.6) are computed as

$$P_{macro} = \frac{\sum_{i=1}^{M} P_i}{M} \tag{6.3.5}$$

$$R_{macro} = \frac{\sum_{i=1}^{M} R_i}{M} \tag{6.3.6}$$

where M is the number of sub-classes in each of the seven fundamental categories.

Figures 6.3 and 6.4 respectively show the macro averaged precision and macro averaged recall for all of the 35 shape queries, grouped with respect to the corresponding fundamental shape. The results of the conceptual image retrieval architecture outperform the results of the QBE architecture for all shape queries.

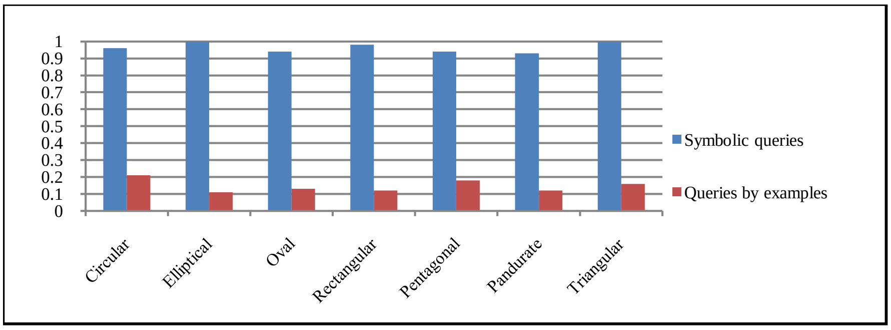


Figure 6.3: *The macro-averaged precision*

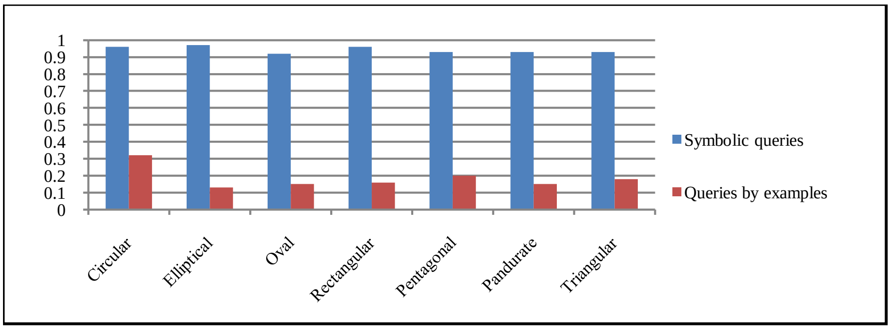


Figure 6.4: *The macro-averaged recall*

6.3.5 Characterizing visual shape concepts: summary

This section presented the experimental evaluation of the conceptual shape characterization framework. The effectiveness of using shape concepts was demonstrated by instantiating shape concepts within a conceptual image retrieval architecture. The latter outperforms a content-based retrieval architecture using query-by-example in a precision and recall based evaluation setting using a novel shape dataset consisting of more than 145, 000 shapes by a factor of 4.5 better for precision and a factor of 3 for recall.

6.4 Experimental validation of the semantic-based region indexing

This section demonstrates the experimental validation of classifying image regions using the visual conceptual information. It is compared to a baseline model operating at the low-level image features. This section answers the second research question in Chapter 1 that is: would describing image regions using the visual color, texture, and shape concepts outperform using a set of low-level image features and thus narrow the semantic gap?

6.4.1 Dataset

The semantic-based region classification is performed using the segmented and annotated IAPR TC-12 benchmark[1] (Escalante et al.; 2010). This benchmark, referred to as SAIAPR TC-12, is an extension of the large scale image collection IAPR TC-12[2] that was introduced for the evaluation of automatic image annotation methods and for studying their impact on multimedia information retrieval (Grubinger et al.; 2006). The IAPR TC-12 collection consists of $20,000$ still natural images taken from locations around the world and includes pictures of different sports and actions, photographs of people, animals, cities, landscapes, and many other aspects of contemporary life. Annotations of these images are provided in English, German, and Spanish.

The extended version, SAIAPR TC-12, is introduced as a benchmark for evaluating image classification and retrieval at the image region level. It consists of all images in the IAPR TC-12 dataset segmented manually and annotated according to a predefined set of labels.

[1] http://imageclef.org/SIAPRdata

[2] http://imageclef.org/photodata

The experiments in this section are carried out on a subset of the SAIAPR TC-12 dataset. This subset consists of 5, 000 randomly selected images. Fifty semantic categories from this benchmark are considered in the classification process. Out of these semantic concepts, 45 categories fall within the 50 most common categories in the whole dataset. WordNet[3] (Miller; 1995) is used to re-organize the selected categories in an ontology of semantic concepts highlighting the hypernymy (i.e., Is-A) relations among these concepts. In doing so, the semantic concepts are organized within a multi-layered ontology ordered by specific/partial order as shown in Figure 6.5. The total number of concepts included in the semantics ontology is 77 high-level semantic categories. Figure 6.5 shows the visual ontology organizing the semantic concepts. In the context of this work, the classification models are trained using the semantic concepts at the leaf level. However, the semantic categories are organized in a visual ontology to facilitate future expansion to add new semantic concepts and for easier readability of the semantic concepts used in this experiments. Incorporating semantic similarity measures between the semantic concepts in the visual ontology is outside the scope of this research work.

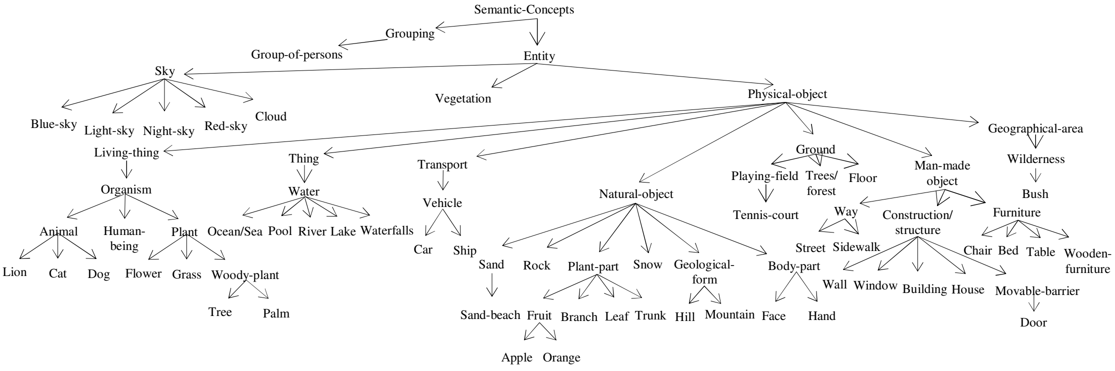


Figure 6.5: *The ontology organizing the high-level semantic categories*

Regions annotated with the semantic categories included in the semantic ontology are extracted. This has resulted in a total number of 15, 304 regions used for training and testing the semantic-based region classification. The regions in each category

[3] http://wordnet.princeton.edu/

are randomly split into disjoint training (80%) and testing (20%) subsets. The total number of the training regions is $12,276$ with $3,028$ regions for testing.

In (Escalante et al.; 2010) an experimental comparison between different machine learning algorithms on classifying image regions was presented. Their methodology is based on conducting a set of comparisons between subsets of their dataset. The subsets were selected based on the top common categories in the entire dataset. For example, top 2, top 3, top 4, ..., and top 100 categories. In this work, 50 categories were selected based on the devised semantic hierarchy depicted in Figure 6.5. This has resulted in not having a one to one match with the most common semantic categories in (Escalante et al.; 2010). Accordingly, a fair comparison with their work would require generating all classification models using the machine learning algorithms presented in Chapter 5 using their features. This experiment is not conducted in this thesis as the main aim of the experiments presented in this chapter is to compare the classification performance of classifying image regions, using a set of low-level image features and their corresponding visual concepts, into semantic categories.

6.4.2 Feature extraction

As discussed in Chapter 3, the visual color, texture, and shape concepts are used to represent and describe image regions in this framework. Each of these features is extracted through a model mapping its low-level features to the corresponding high-level features (i.e., visual concepts) implementing the theoretical frameworks described in Chapter 3 for the visual color and texture concepts and Chapter 4 for the visual shape concepts.

There is no need to apply a global segmentation algorithm on the images to extract image regions since the SAIAPR TC-12 benchmark provides the segmentation masks with their images (cf. Section 6.4.1) along with their high-level annotations.

Accordingly, the first step towards extracting the visual concepts from image regions is extracting their low-level features.

First, the low-level shape features are extracted from image regions characterizing their global boundaries. The global shape of the image region, represented by 64-dimensional Fourier descriptors, is mapped to its corresponding high-level shape concept through the framework for characterizing the visual shape concepts as explained in Section 6.3.

Having extracted the visual shape concepts, each image region is then divided into a set of overlapping image patches of a fixed size (35×35) pixels. The low-level color features in the HVC space and texture features represented as Gabor filters are extracted from the image patches to be further processed by the visual color and texture concepts characterization frameworks.

In this framework, a subset of shapes from the shape lattice; i.e., the seven basic shapes at the first level of descendants, is considered to describe image regions (cf. Section 4.2). Basically, an average user is interested in specifying the shape properties of objects using the basic shape characteristics. For example, users can differentiate between rectangular or square table and circular or oval one. This is an analogy to the use of basic color terms (Berlin and Kay; 1991). A basic color term of a language is defined as being not subsumable to other basic color terms. For example, the "teal" color is described as a medium blue-green, hence it is not considered as a basic color term. Similarly, normal non-expert users can differentiate between an oval and a rectangular table rather than describing a table as being "triangular with one constricted inward and one excurvated edges" table. Accordingly, the seven fundamental shape concepts of the shape lattice shown in Chapter 4 are utilized for describing image regions for the region-based semantic classification.

The characterized visual color, texture, and shape concepts are then combined together in a "late fusion" fashion (Snoek et al.; 2005) in which the distribution of each color concept and the confidence scores of the texture and shape concepts (cf. Section 5.2) are integrated to form the feature vectors that will be used to learn the set of high-level semantic categories. The concatenation of these vectors results in a redundancy of the shape concepts in the set of vectors extracted from an image, as the shape of an image region is represented in a single vector describing the global shape of a region while the color and texture features are extracted from multiple image patches extracted from the same image region. The reason for this redundancy is that a semantic object (i.e., an image object) may have many different shapes and might be somewhat occluded or deformed due to different viewpoints. This is mainly when images are captured by non-expert users as in personal photo collections. Moreover, the shape of one object may differ through time, such as the moon, which starts in a "crescent" shape, then it becomes "half circular", then "circular", and then it returns back as crescent and so on. In deference to this, the shape representation is implemented to be redundant in which the global shape of an image region is concatenated with each vector containing the color and texture representations extracted from each image patch in that image region. This representation aims to have training vectors with many possible combinations of color, texture, and shape features for the same object as it may appear with different visual properties in different images. This is similar to the approach presented in Nilsback and Zisserman (2006) in which shape features are redundant to give immunity to occluded flowers in a flower classification system.

To deal with the issue of imbalanced data between different classes, the training and test sets were sampled at random into three different sets to generate and test the classification models. The training/test sets were generated with different sizes. The ratio between the training/test sets is kept for sampled sets so that for each training set there exist a corresponding test set.

The final performance scores reported in Section 6.4.4 are computed as the mean classification value per class. The classification process is repeated three times with each of the training/test sets, and final average scores are reported. The detailed classification scores in precision, recall, and F-score measures are reported in Appendix A.

6.4.3 Experimental settings of the supervised learning algorithms

This section summarizes the parameter settings and variations for the learning algorithms used in the semantic-based region classification.

Support Vector Machines *(cf. Section 5.3.1)* Several kernels are used and tested: the linear, sigmoid, polynomial of 2 and 3 degrees, and Radial Basis Function (RBF) kernels. The LibSVM[4] library (Chang and Lin; 2011) is used in this experiment. The cost and gamma parameters are estimated through cross validation. The cross validation uses grid search to find the best values of C and γ. The grid search was modified to run on 4-core machines in parallel environment. The search for C is varied in $\{10^{-3}, 10^{-1}, 10^{1}, 10^{3}, 10^{5}, 10^{7}\}$ and γ in $\{10^{-7}, 10^{-5}, 10^{-3}, 10^{-1}, 10^{1}\}$.

The multiclass classification problem is solved using one-against-one (OAO) technique. The OAO technique has shown to yield better results than the one-against-all (OAA) when the number of classes is relatively high (Rifkin and Klautau; 2004; Hsu and Lin; 2002), which is the case of the number of semantic categories in this experiments.

Decision Tree *(cf. Section 5.3.2)* The C4.5 and Naive Bayesian learning algorithms are used. For the C4.5 algorithm, different splitting criterion and pruning options are used in the training process: the splitting criterion is varied $\{1, 2\}$ and the pruning confidence factors applied are $\{0.15, 0.1, 0.05, 0.25, 0.5\}$.

Random Forests *(cf. Section 5.3.3)* Forests are grown with sizes of $50, 100, 500$, and 1000 trees. The size of the feature set at each split is as suggested by Breiman

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

(2001): $n/2, n, 2n, 4n$, and $8n$ where n is the square root of the number of features. The Weka[5] (Hall et al.; 2009) implementation is used for Random Forest algorithm.

Nearest Neighbor *(cf. Section 5.3.4)* The Nearest Neighbor (kNN) classifiers are generated with the Euclidean and Manhattan distances. The features are sorted in KD-Tree (Friedman et al.; 1977) and Cover Tree (Beygelzimer et al.; 2006) to conduct the search. Their implementations in Weka are used. The search was also conducted linearly using all features. The number of the nearest neighbors k is varied to $\{1, 2, 4, 8, 16, 32, 64\}$ neighbors.

Naive Bayes *(cf. Section 5.3.5)* The continuous features for both the visual conceptual information and the baseline model are handled using Entropy Minimization Discretization (EMD), kernel estimation, and using features as single normal. The implementation of Weka is used as well as the options to handle the continuous attributes.

Logistic Regression *(cf. Section 5.3.6)* Logistic regression classifiers are generated using L_2 regularization. The LibLinear[6] library (Fan et al.; 2008) is used to generate the logistic regression classifiers.

Regularized models are trained varying the ridge parameter by factor of ten from 10^{-8} to 10^{5} following the approach of Caruana and Niculescu-Mizil (2006).

6.4.4 Performance evaluation

This section presents the performance evaluation of the region-based image description and classification using the visual conceptual information. The accuracy is measured by how well the classification models perform on a set of un-annotated image regions. The classification process results in annotating image regions as belonging to high-level semantic categories as given in the ground truth set.

[5] http://www.cs.waikato.ac.nz/ml/weka/

[6] http://www.csie.ntu.edu.tw/~cjlin/liblinear/

The annotation accuracy of image regions, described by the visual conceptual information as their features, is compared with a baseline framework that operates at the low-level image features to infer the semantics conveyed in image regions. The low-level features used in the baseline model are the average color in the HVC color space, Gabor filters with 5 scales and 6 orientations representing the texture features, and Fourier descriptors to represent the shape features of the extracted image region. Indeed, these are the low-level visual features extracted from images before the mapping to the high-level visual color, texture, and shape concepts. This experiment is conducted so that to compare semantic-based classification of image regions using low-level image features and classifying image regions after mapping the low-level features to their corresponding visual concepts.

The experiments in this section are performed as follows. First, the machine learning algorithms presented in Chapter 5 are used to establish correspondence between the visual conceptual information and the set of high-level semantic categories (cf. Section 6.4.1) and their performances are empirically compared. The evaluation metrics used are the precision, recall, and the F-score measures. Further, the difference in the classification performance between the visual concepts and the baseline method using each learner is shown in precision/recall figures. The experimental findings are supported with statistical tests.

The aim of these experiments is to measure the classification performance of using the visual color, texture, and shape concepts to describe and classify image regions. The classifier with the highest classification score is selected to conduct the experiments on region-based image retrieval incorporating visual conceptual characteristics of image objects in Section 6.4.6.

Several models are generated for each learning algorithm depending on the number of parameter settings. For example, different kernels are used for support vector machines and different discretization methods are used to discretize the continuous features for naive Bayes classification as shown in Section 6.4.3. Each learner is also used to generate a baseline model to be compared with the model generated using the visual concepts.

Result Discussion

Table 6.5 shows the mean normalized performance scores of each learning algorithm. Each score shown in the table represents the average of that performance metric over all its generated models. For example, the precision score of the decision tree represents all precision scores of all the models generated using the decision tree algorithms with all possible parameter variations as shown in Section 6.4.3. This is also applied to the other learning algorithms. The detailed classification scores for classifier are reported in Appendix A.

	Classification algorithm											
	NB		*kNN*		*LR*		*DTree*		*RF*		*SVM*	
	VC	*BL*	*VC*	*BL*	*VC*	*BL*	*VC*	*BL*	*VC*	*BL*	*VC*	*BL*
Average Precision (%)	27.7	17.6	38.7	26.8	39.7	20.7	35.6	19.6	54.8	32.9	47.6	32.1
CI	(17.3–38)	(-1.9–37.1)	(37.6–39.9)	(26.4–27.3)	(36.6–42.7)	(12.8–31.6)	(33.3–37.9)	(18.7–22)	(53.7–55.9)	(31.6–34.2)	(45–50.2)	(30.9–33.3)
Average Recall (%)	18.9	9	27.5	30.9	29.8	23.7	23.5	15.7	43.6	38	35	27
CI	(11.4–26.5)	(-7–25.8)	(26.6–28.4)	(29.8–31.9)	(27.7–32)	(19.3–29.8)	(21.3–25.8)	(15.2–16.6)	(41.8–45.4)	(36.3–39.7)	(32.3–37.7)	(20.9–33.2)
Average F-score (%)	17.4	8.8	29.7	26.0	28.6	16.6	26.7	16.4	46.7	30.6	37.8	24
CI	(8.5–26.3)	(-9.8–27.5)	(28.7–30.6)	(25.5–26.5)	(25.7–31.4)	(11.1–24.1)	(24–29.4)	(15.9–17.3)	(45–48.5)	(28.8–32.5)	(34.8–40.7	(18.8–29.1)

Table 6.5: *The classification performance of using the visual conceptual information against using the baseline model. The classification scores (precision, recall, and F-score) are averaged using all models generated by classifiers with the possible parameter settings. CI indicates the 95% confidence interval of the obtained results*

Random Forest classification results

It can be seen that the classification accuracy using the visual conceptual information (VC), in all performance metrics, outperforms the baseline model (BL). Random forest learning algorithm (RF) has the highest scores among the other classifiers. The high accuracy of Random Forest is consistent using both the visual concepts and the baseline model operating at the low-level image features. This is in line with the studies in Caruana and Niculescu-Mizil (2006) and Escalante et al. (2010) in which Random Forest classifier showed high classification accuracy in comparison to other classifiers. This is because aggregating the results of many decision trees for the final classification decision. Nonetheless, the classification difference between the visual concepts and the baseline using the Random Forest classifier is found to be statistically significant based in Mann-Whitney U test ($p < 0.05$). Mann-Whitney U test is used to find statistical significance between two groups. It is used instead of student's t-test when the assumption of data normality is violated. The data was found to be not normally distributed and Mann-Whitney U test is used to test any statistical significance between the two groups of results.

Appendix A shows the detailed precision, recall, and F-score values of the Random Forest classifiers.

Nearest Neighbor classification results

In the case of nearest neighbor classifiers (kNN), it can be observed that the classification difference is minimal but in favor of the visual concepts. The difference is not statistically significant based on Mann-Whitney U test ($p < 0.05$). Interestingly, this also is in line with one of the reasons Boiman et al. (2008) highlight that causes the degradation in nearest neighbor classification: feature quantization. However, using the visual concepts still outperforms the baseline method. It is also noteworthy that the difference in using KDTree or Cover Tree to sort the data instances is minimal. They have very similar classification accuracy, but there is a trade-off in the average training time as KDTree algorithm was found to be faster than the Cover Tree as shown in Section 6.4.5.

Increasing the number of nearest neighbors (i.e., 32 and 64) results in higher classification accuracy. Regarding the distance function used, Manhattan distance slightly outperforms the Euclidean distance. The difference between them was found to be minimal. The detailed classification scores of the kNN classifiers are listed in Appendix A.

Naive Bayes classification results

Naive Bayes (NB) does not perform well in comparison to other classifiers. The classifiers generated with continuous features discretized using Entropy Minimization Discretization (EMD) perform much better than the classifiers generated with the features discretized using kernel estimation or even using features as single normal. The single normal in fact performs quite poorly. This explains the wide CI error bar of the averaged F-score metrics as shown in Figure 6.6 and Figure 6.7. The difference is not statistically significant based on Mann-Whitney U test ($p < 0.05$). Appendix A shows the classification scores of all Naive Bayes classifiers. Figure 6.6 depicts the average F-score measures of the visual concepts and the baseline model of all learning algorithms.

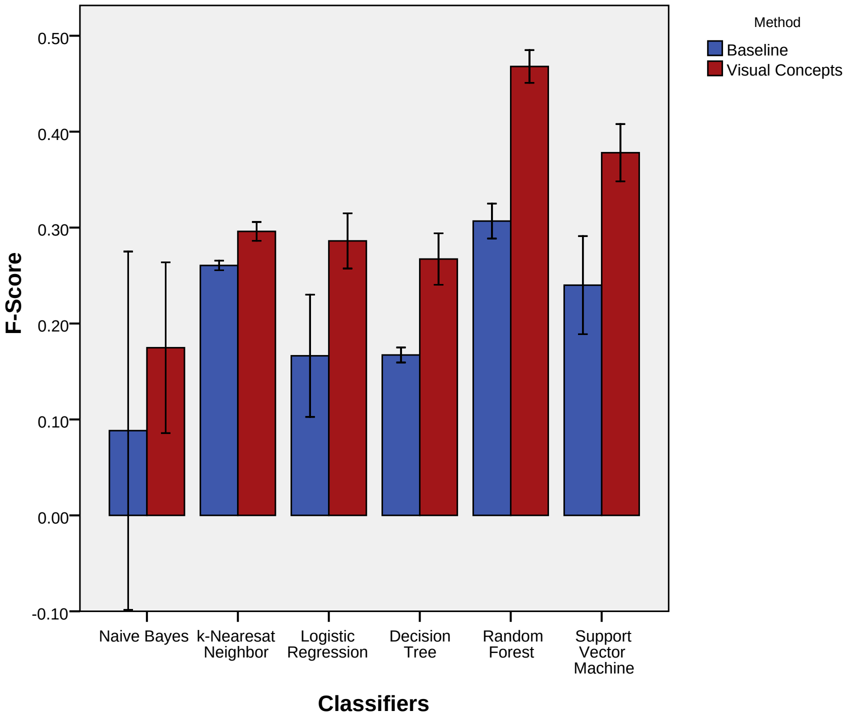


Figure 6.6: *The mean F-score metric of the classification results using the visual conceptual information in comparison with the baseline method with 95% Confidence Interval (CI) error bars*

Support Vector Machines classification results

As expected, Support Vector Machine (SVM) learning performs well. As far as the classification accuracy at hand is concerned, the performance of SVM classifiers is consistent on both the visual concepts and the baseline model. It is noteworthy that the Polynomial of 3 degrees kernel outperforms the other kernel methods; very close to it is the Polynomial of 2 degrees, then the linear kernel, the Sigmoid kernel is also very close; lastly the RBF kernel. However, the range of average score of all of these kernels is tight. The difference in the classification accuracies between the visual concepts and the baseline method using Support Vector Machine classifiers was found to be statistically significant according to Mann-Whitney U test ($p < 0.05$). The classification scores of all SVM classifiers are shown in Appendix A.

In comparison to other classifiers, SVM performs better except for Random Forests. Figures 6.6 and 6.7 show that SVM is in the second rank after RF for the visual concepts. However, its baseline is in the third rank after RF and kNN baseline models.

Decision Tree classification results

The performance of the Decision Tree learners is poor in comparison to the other classifiers. In the baseline models it performs even poorer. The Naive Bayes Decision Tree (NBDT) performs poorer than the C4.5 algorithm. The difference in the classification accuracies between the visual concepts and the baseline method using the decision tree was found to be statistically significant according to Mann-Whitney U test ($p < 0.05$). The detailed classification scores of Decision Tree classifiers are listed in Appendix A.

Logistic Regression classification results

As far as the Logistic Regression (LR) is concerned, it was noticed that classifiers generated with higher ridge value have achieved higher precision, recall, and F-score scores (cf. Appendix A). This is mainly the case with the visual concepts in which the classifiers generated with higher ridge values have resulted in the higher performance metric values. In the baseline, the classifier with the ridge value 10^0 yielded the highest scores.

The overall performance of LR using the visual concept is higher than the baseline method (classification scores are shown in Appendix A). Its performance was found to be statistically significant using Mann-Whitney U test ($p < 0.05$).

Performance comparison of learning algorithms

It can be seen that the F-score values of the classification using the visual concepts consistently outperforms the classification scores using the low-level features in the baseline method for each learner. Figure 6.7 shows the same results as Figure 6.6 but the F-score bars are grouped by the classifiers generated using visual concepts and the ones generated using the baseline method. It can be noticed that the performance of the classifiers is relatively consistent. For example, the Random Forest learner outperforms all the other classifiers. Naive Bayes classifier has the lowest classification accuracy in comparison to the other classifiers. Decision tree is also consistent in the fifth rank. The second most accurate classifier is the SVM. Nearest neighbor and logistic regression are ranked in the third and the fourth places.

Moreover, the behavior of the different parameter settings is to some extent consistent. In Random Forest, increasing the number of trees at each split results in a higher performance for both experiments as well. In naive Bayes, models generated using features discretized with EMD have achieved higher classification performance values than the kernel estimation for both the visual concepts and the baseline. In nearest neighbor, the linear search strategy for the nearest neighbor was also consistent in comparison to KDTree and Cover Tree algorithms. The overall difference in the F-score values is minimal. However, it was noticed that the number of nearest neighbors that yielded the highest F-score value using the visual concepts was with

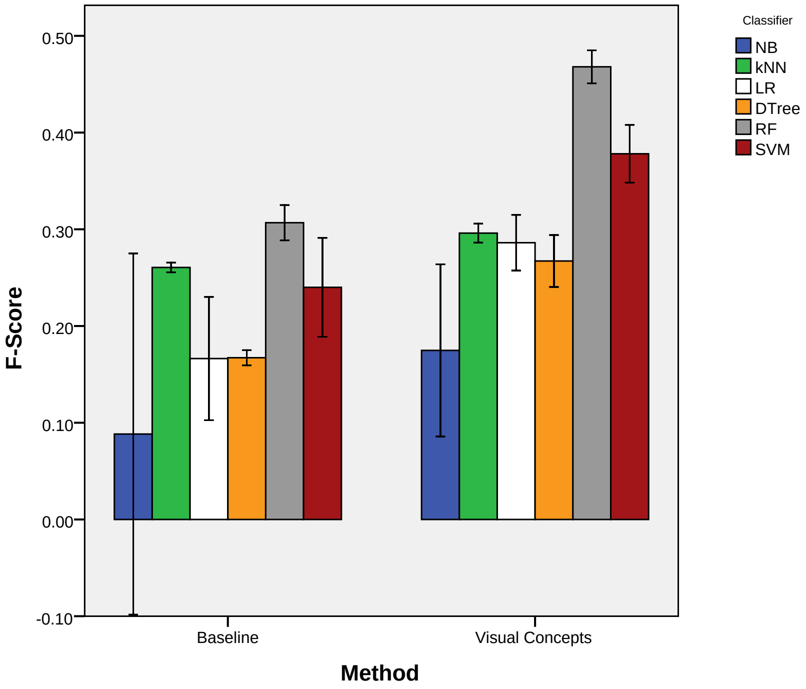


Figure 6.7: *The mean F-score metric of the classification results grouped by the visual conceptual information and the baseline method with 95% Confidence Interval (CI) error bars*

32 neighbors, whereas 8 neighbors in the case of the baseline. In Decision Tree learning, on average the splitting criterion value of 2 has achieved slightly higher F-score than the value 1. The pruning confidence factors of lower values (i.e., 0.15 and 0.1) has also slightly resulted in higher F-score values.

We can see that the difference in classification performance between the visual concepts and the baseline has noticeably improved with the Decision Tree, Naive Bayes, and Random Forest classifiers. This is related to how these algorithms work. As mentioned in Chapter 3, the visual concepts can be viewed as a perceptual-based quantization technique of the image features. Having the image features quantized, it plays an important role in generating the decision trees' splitting criterion. The impact of having the features quantized into a smaller number of features results in reducing the number of split nodes and a less scattered and more concise decision tree and thus, better classification accuracy. Moreover, having more features may cause the decision tree to over-fit.

The same applies to Random Forest classifiers since it is an ensemble classifier and consists of many generated decision trees.

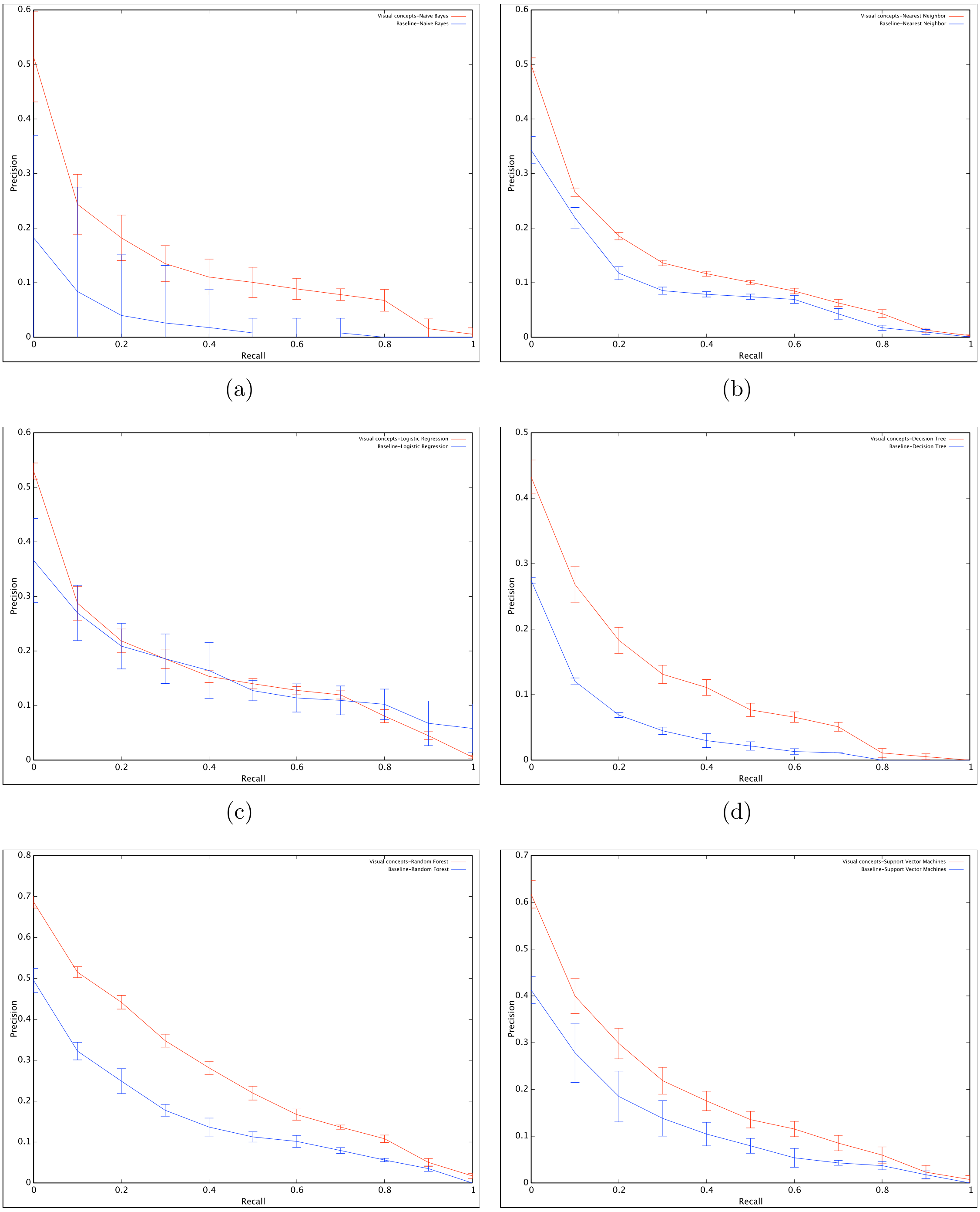
The performance of the Bayesian learner using the visual concepts is much higher than using the low-level features as the class-density estimations are computed using the visual concepts. Again, this results in using quantized feature vectors of dimensionality less than low-level visual features in the baseline method. Jakulin (2005) has shown that increasing the number of features usually decreases the classification accuracy. He explains that Naive Bayes classifiers always benefit from feature selection which reduces the number of features used in the classification process; in other words, having more uninformative features will cause the classification accuracy to degrade. In the case of the baseline method, as it uses low-level visual features without applying feature selection algorithms, the baseline is prone to having uninformative features which will cause the classification accuracy to further degrade.

The difference in the classification performance between the visual concepts and the baseline model for each learner is shown in precision/recall figures in Figure 6.8. It can be seen that the classification for each classifier has been enhanced by using the visual concepts over using the low-level image features.

Statistical validation

Using all F-score measures of all classifiers using the visual concepts and the baseline method, a statistical test is conducted to show that the improvement in classification accuracy using the visual conceptual information is statistically significant compared to using the low-level visual features.

The statistical test of choice is the "Kruskal-Wallis H test" as the test aims at reporting more than two sets of results. Kruskal-Wallis test is a nonparametric test



(a) (b) (c) (d) (e) (f)

Figure 6.8: *The precision and recall figures of: (a) naive Bayes; (b) nearest neighbor; (c) logistic regression; (d) decision tree; (e) random forests; and (f) support vector machine*

that is used to compare means of more than two groups. It is equivalent to one-way ANOVA and an extension to Mann-Whitney test. It is used when the assumption of data normality is violated unlike the one-way ANOVA that assumes the data is normally distributed. This test is performed on the "F-score" values presented in Figures 6.6 and 6.7 and it was selected as it was found that the F-score values are not normally distributed.

Kruskal-Wallis test show that there is a significant difference in the medians $\chi^2(11, N = 345) = 182.34$, $p < 0.05$. Because of this test significance, a post-hoc test to conduct pairwise comparisons among all groups should be considered. The post-hoc test, Dunn's analysis, has revealed that out of all classifiers, Random Forest using visual concepts has achieved the highest statistical significance. However, it was found that Random Forest using visual concepts is statistically significant against all classifiers except Support Vector Machines using visual concepts and the baseline model using Random Forest classifiers. As for the SVM classifiers it still has achieved higher classification accuracies as shown in Table 6.5. And as far as the baseline model using Random Forests is concerned, it is noticed that Mann-Whitney's test has shown that Random Forest using visual concepts is statistically significant over its baseline model. The difference here is that Dunn's analysis uses Bonferroni correction procedure to adjust the p values (i.e., the critical value for significance alpha) (Olejnik et al.; 1997). So the p-value is adjusted for each pairwise comparison and that caused the pairwise comparison of Dunn's post-hoc test to differ than Mann-Whitney's result when comparing the Random Forest's classification accuracy using the visual concepts and the baseline model.

In general, the overall statistical tests (i.e., the Kruskal-Wallis and the classifiers' individual Mann-Whitney tests) show that using the visual concepts to describe and classify image regions outperform using the low-level image features.

6.4.5 Computational time and cost

This section reviews some of the computational cost aspects used in performing the experiments.

Due to the high computational requirements needed to accomplish the training and testing for the machine learning algorithms, the experiments presented in Section 6.4 are performed on the Monash High Performance Computing (HPC) facilities[7]. The high computation requirements are mainly the memory and storage. Where applicable, some of the processes such as SVM grid search were also run on multi-core machines.

In Random Forest training, the main drawback found, especially when growing large number of trees, is the memory requirements. For generating Random Forest classifiers this is, on average, of 40 GB on Monash HPC. The average training time is 868.5 minutes.

In kNN classification, the difference in classification scores using KDTree or Cover Tree to sort the data instances was minimal. They have resulted in very similar classification accuracy. However, there is a trade-off in the average training time of 5.5 minutes for KDTree to 31.2 minutes for Cover Tree for the same model.

Linear search for the nearest neighbors has, on one hand, the highest scores in comparison to using KDTree and Cover Tree algorithms to search for the nearest neighbors. On the other hand, the training time for the linear search models was the highest of all nearest neighbor models with an average of 138.6 minutes.

Naive Bayes learning is considered to be a fast learner with an average training time of 112 minutes.

One of the problems encountered while generating the SVM models is the computation resources needed, especially during running the cross validation to get the best C

[7] http://www.monash.edu/eresearch/services/mcg/

and γ parameters. The search for the parameters is performed using the grid search provided with the LibSVM library. The code was modified to run the search in a parallel environment using 4-core machines. The longest average time was for linear kernel with $6,510.5$ minutes. The shortest time was for the sigmoid kernel with 258 minutes. The parameter search for the RBF kernel was $1,349.7$ minutes, Polynomial with 2 degrees was $1,425.3$ minutes, and Polynomial with 3 degrees took 916.6 minutes to find the best parameters.

Training SVM models is less expensive than the grid search. The best parameters found using the grid search through cross validation are used to generate the final SVM models. On average, the shortest training time was for the sigmoid kernel with 245 minutes, then the RBF with 266 and very close the linear kernel with 275 minutes, next to it polynomial of 2 degrees with 284 minutes, and finally the polynomial of 3 degrees with 388 minutes.

In Logistic Regression, the training time was noticed to be longer when the ridge value becomes higher. The average shortest training time was for the ridge value 10^{-8} with 1.5 minutes and the longest training time was 172 minutes when the ridge value was 10^5.

6.4.6 Retrieval using high-level concepts

This section represents an experimental application of using the visual concepts that is retrieval of image regions. This experiment is a prototype that simulates a region-based image retrieval framework within a conceptual image retrieval architecture coupling both the semantic categories and the visual conceptual information of image region. Its evaluation is based on the notion of image relevance which consists in quantifying the correspondence between index and query image representations. It should be noticed that this experiment is conducted on the feature level of image regions and the final score metrics are computed on the correct classification of image regions. This simulates a retrieval of image regions.

The classifier used for the semantic-based retrieval is a Random Forest classifier of 100 trees and each tree is constructed while considering 5 random features. The retrieval of semantic image regions is compared to a content-based image retrieval architecture based on a query-by-example process operating on low-level color texture, and shape features. Index and query representations are compared based on the Euclidean distance.

A set of 23 non-trivial textual queries of different semantic concepts were selected for the experiment. These queries were selected with the criterion of having rich perceptual representation of the high-level semantic categories they represent rather than selecting queries with the high-level semantic categories solely. Each query contains perceptual characteristics that reflect some visual properties of the image region, corresponding to the visual color, texture, and shape concepts. Examples of such queries are “whirly water”, “sunset”, “clear blue sky”, and “cracked wall”. Each query is mapped to relevant query terms that match the semantic color, texture or shape concepts of the original query. For instance, “triangle-like mountain” is mapped to “mountain with the probability of the triangular shape as being the highest”. This is achieved by filtering the specified semantic category based on the specified visual property. Table 6.6 shows the mapping of queries into query terms that match the semantic color, texture or shape concepts of the original query. These queries represent the semantic categories (simulating a user given query) (user gives) to the system in addition to some perceptual visual properties of the queried categories. In this way, the user’s interpretation of the visual properties is represented by using the visual conceptual information that reduced the semantic gap between the low-level representation and the humans’ interpretation of those features. For the baseline architecture, three image regions were randomly selected for each query as an input to the query-by-example module. (Three image regions represent a good threshold without further degrading the classification accuracy of the QBE model.)

Out of 3000 image regions used in this experiment, Figure 6.9 shows 100 randomly selected image regions.

The F-score indicator is used to evaluate the performance of the compared models. Figure 6.10 shows the F-score values of all the 23 non-trivial semantic queries. By analyzing the obtained results in Figure 6.10 , it is seen that the results of simulating textual queries containing the semantic concepts and their visual concepts in the simulation of the semantic-based region retrieval outperforms the content-based image retrieval model for most queries.

A statistical test is performed using the F-score values of both the semantic-based region retrieval using the visual concepts and the content-based retrieval. The statistical test is the Wilcoxon Signed Rank test. The Wilcoxon Signed test is a nonparametric statistical test and is alternative of the paired t-test. It is used instead of paired t-test when the normality assumption of the data is violated, which is the case with the F-score of both the retrieval using the visual concepts and the QBE module. This statistical test is applied to make statistical decision as to whether or not using the visual concepts in the retrieval process outperforms using the query-by-example model. A Wilcoxon Signed Rank test showed that there is a significant difference between using the visual concepts and the QBE, $z = -3.74$, $p < 0.05$. By conventional criteria, this difference is considered to be statistically significant.

6.5 Summary

This chapter has presented the experiments carried out in validating the work presented in this thesis. This chapter has given answers in the affirmative to the research questions of this thesis presented in Chapter 1: can shapes of segmented image regions be represented using high-level perceptual descriptors rather than low-level shape features for the task of image description and classification? and

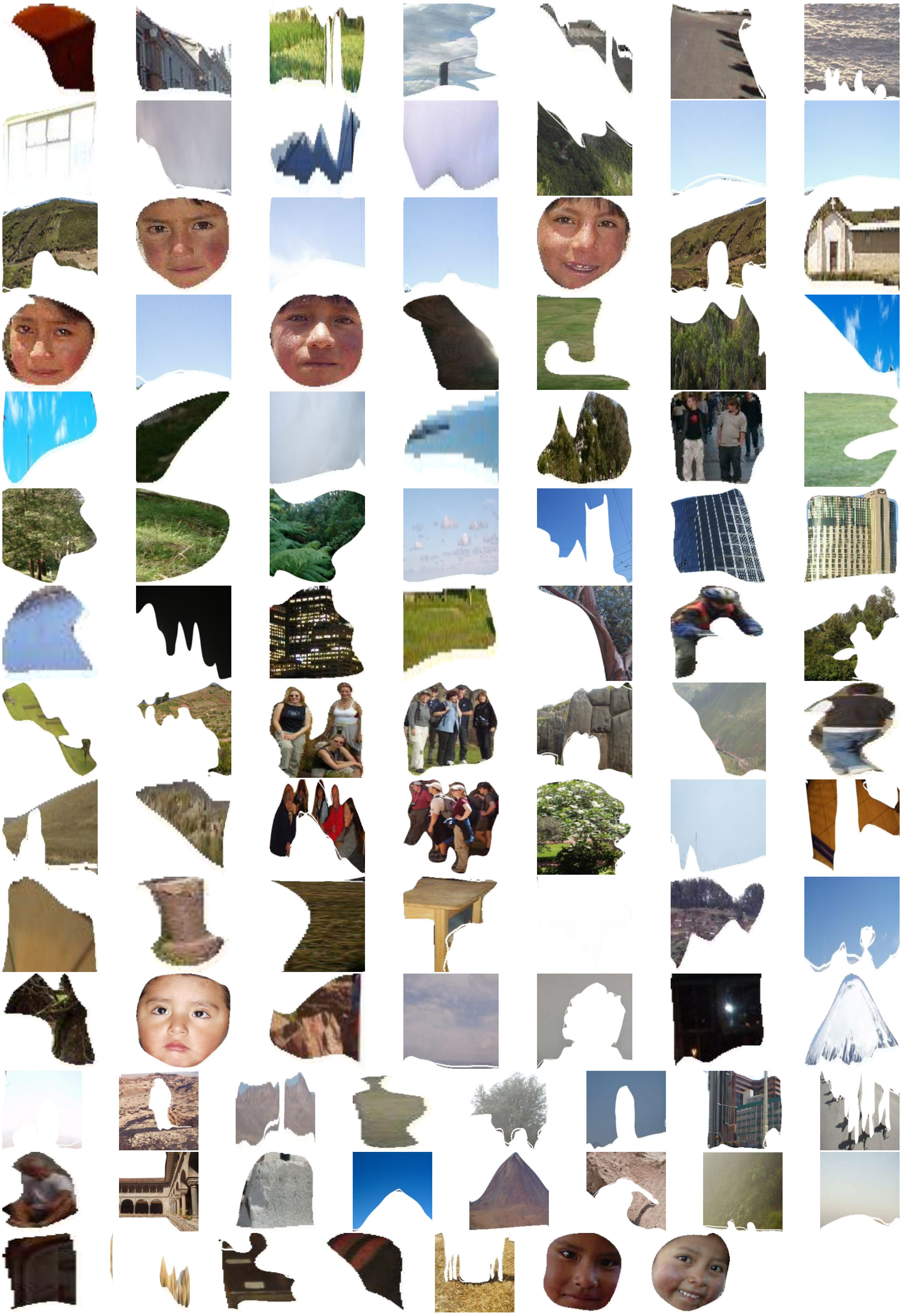


Figure 6.9: *Examples of image regions used in the retrieval experiment*

Query	Query mapping into semantics and visual concepts
Whirly ocean	ocean with the probability of the texture whirly as being the highest
Triangle-like mountain	mountain with the probability of the triangular shape as being the highest
Trees with green leaves	tree image regions with the color green having the highest distribution
Lined trees	tree image regions with the texture lined as being the highest
Sunset	sun with color orange as being the highest
Rocky mountain	mountain with the colors black and gray (dark colors) have high distributions
Rectangle-like mountain	mountain with the probability of the rectangle shape as being the highest
People wearing purple clothes	people image regions with the color purple as being the highest
People in red clothes	people image regions with the color red having the highest distribution
Lined people	people image regions with the texture lined as being the highest
Lined wall	wall with the texture lined having the highest probabilistic estimation
Lined vegetations	vegetations with a high score of the texture lined
Interlaced vegetations	vegetations with a high score of the texture interlaced
Lined door	door image region with the texture lined as being the highest
Gray wall	wall image regions with the color gray as being the highest
Dark-green vegetations	vegetations image regions with a high distribution of the black color
Dark skin face	face with the color black as being the highest
Circular face	face image regions with the shape circular as being the highest
Cracked wall	wall image regions with the a high probabilistic estimation of the texture cracked
Clear blue sky	sky image regions with the colors white and gray are very low and the color blue or cyan as being the highest
Green grass	grass image regions with the color green as being the highest
A night view	sky image regions with the color black has the highest distribution
A bed with a red cover	bed with a high distribution of the color red

Table 6.6: *The proposed queries and their mapping into semantic concepts and visual concepts describing their properties*

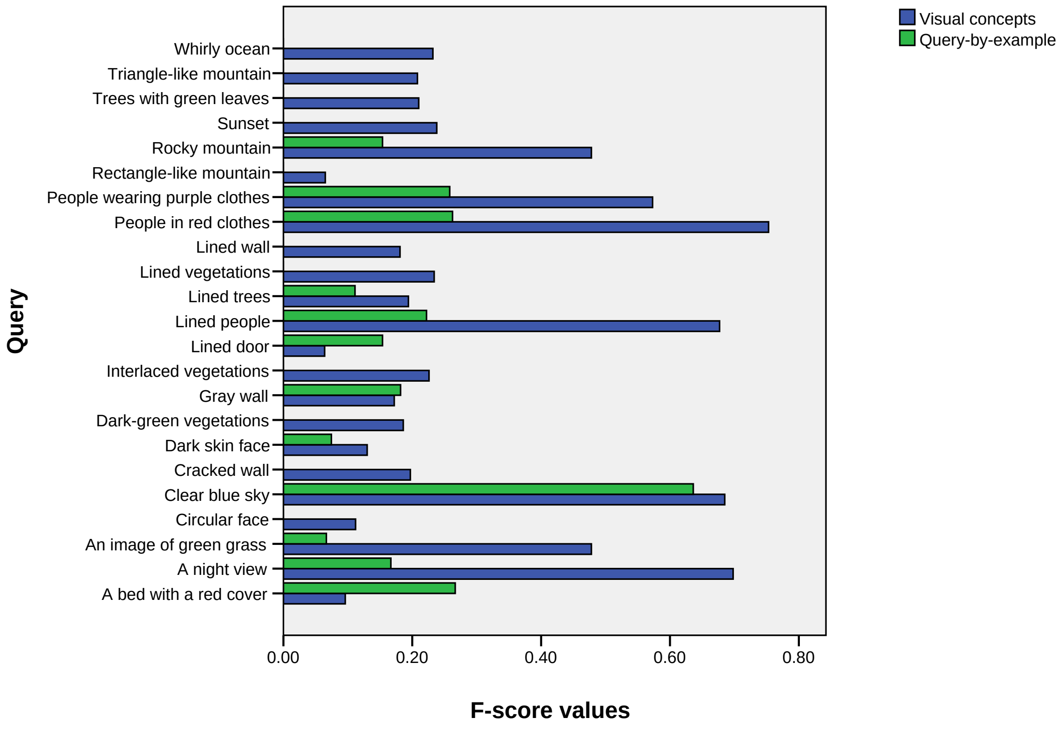


Figure 6.10: *The retrieval accuracy using the F-score metric between queries incorporating the visual concepts and the baseline method using query-by-example*

would describing image regions using the visual color, texture, and shape concepts outperform using a set of low-level image features and thus narrow the semantic gap?.

At first, an overview of the implementation of the frameworks for extracting the visual color and texture concepts was presented. Then the implementation detail of the framework for characterizing the visual shape concepts has been presented and has answered the first research question of this thesis. The integration of the visual color, texture, and shape concepts for the task of image description of image regions was then illustrated. This was tested against a baseline method operating at the low-level image features to classify image regions. Statistical tests supported the findings of the experiments. Lastly, a preliminary retrieval experiment that incorporate using the visual concepts to retrieve high-level semantic categories was presented and showed that using the visual concepts outperforms using a query-by-example architecture operating at the low-level image features.

Chapter 7

Conclusion and Future Directions

7.1 Summary of thesis contributions

Aiming at narrowing the semantic gap between the low-level image features and the humans' rich understanding of their visual properties, this thesis presented an approach to describe and classify image regions using their visual conceptual information. Through the development of models that map the low-level features into a set of high-level symbolic descriptors, such as red or green for color features; lined or marbled for texture features; and elliptical or rectangular for shape features, image regions are described in transparent and readable descriptors rather than their low-level image features. These symbolic descriptors, named in this thesis as the visual concepts, are then used to learn a set of high-level semantic categories to classify un-annotated image regions. These aims and motivations were discussed in Chapter 1.

The literature review in Chapter 2 highlighted approaches, systems, and techniques in image indexing and retrieval. The state-of-the-art research of image retrieval frameworks was discussed starting from the first class of image retrieval frameworks using manual text annotation to semantic-based image indexing. Methods and techniques used to address the semantic gap were also presented.

The theoretical formalization of using the visual concepts to represent image regions was presented in Chapter 3. Frameworks used to extract the visual color and texture concepts were demonstrated. In particular, the framework for characterizing the visual color concepts transforms the pixel values from their representation in the RGB color space to the HVC color space and then characterizes the image colors using a set of visual color concepts presented in Section 3.2.1. Section 3.2.2 described the framework for characterizing the visual texture concepts that maps the extracted Gabor features into a set of pre-defined texture concepts through Support Vector Machine classifiers.

Chapter 4 presented a framework that aims to describe image regions using generic visual shape concepts that describe their boundary information. A set of shape concepts were introduced to describe image regions. The development of shapes from the fundamental shape concepts and the transformations applied to derive new novel shapes were illustrated. All the shapes were organized in a shape lattice that consists of seven sub-lattices originated from the each of the seven fundamental shape concepts.

Chapter 5 reviewed the theoretical foundations of the supervised machine learning algorithms used in the experiments to learn a set of high-level semantic categories using the visual conceptual information. First, problem formulation and notations were introduced. Then a brief theoretical description of each learner is presented including an overview of automatic image annotation and image classification systems use each learning algorithm.

The experimental validation presented in Chapter 6 started by briefly describing the implementations and experimental validation of the frameworks characterizing the visual color and texture concepts in Section 6.2. The experiments of the framework for extracting the visual shape concepts from image regions were described in Section 6.3 and answered the first research question of this thesis that is if shapes of segmented image regions be represented using high-level perceptual descriptors rather than low-level shape features for the task of image description and classification.

The visual color, texture, and shape concepts are then used for the task of image description and classification in Section 6.4. The dataset used in this experiments, feature extraction, and pre-processing steps were described in Sections 6.4.1 and 6.4.2. Applying the supervised machine learning algorithms for classifying image regions using the visual conceptual information was presented in Section 6.4.4. The efficiency of using the visual conceptual information was demonstrated by conducting a comparison with a baseline framework operating at a set of low-level image features to learn the same set of semantic categories. The baseline model operates at the low-level image features extracted from image regions before the mapping to their corresponding visual concepts. This is conducted to show enhancements in the performance of classifying image regions using the visual conceptual information. The results of this section have answered the second research question of this thesis that is if describing image regions using the visual color, texture, and shape concepts outperforms using a set of low-level image features to annotate image regions with semantic categories and thus narrow the semantic gap. Statistical tests were also used to support the findings of these experiments.

7.2 Concluding remarks

This thesis presented an approach to describe and classify image contents using their visual conceptual information. This entails describing image regions using their visual color, texture, and shape concepts. The problem of image segmentation and the effects of image segmentation accuracy on the performance of classifying and retrieving image regions is out of the scope of this study. One of the main contributions is presenting a framework for characterizing visual shape concepts to describe the shapes of image regions. The main aim of this thesis has shown to achieve promising results and outperformed the baseline model.

Experiments on shape based conceptual characterization demonstrated that the visual shape concepts outperformed the low-level shape features in classifying shape objects.

As far as the semantic-based classification of image regions is concerned, using the visual concepts to describe image regions has shown to outperform using a set of low-level image features in the same task. Statistical tests were used to support these findings. Mann-Whitney U test was used to find statistical significance between models generated using the visual concepts for each classifier and their counterparts generated for the baseline model. This statistical significance between the pairs of models generated for each classification algorithm were also depicted by the confidence interval error bars on the mean F-score metrics in Figure 6.6 and Figure 6.7. In addition, Kruksal-Wallis H test on all the average F-score values was conducted to find the overall statistical significance of all classifiers. It was shown that the Random Forest classifier outperformed all the generated classifiers using the visual conceptual information and the baseline models. Random Forest has shown to be a strong classifier when dealing with image contents. The overall discussion of the results obtained was presented in Section 6.4.4.

As for retrieval, Section 6.4.6 demonstrated the effectiveness of using the visual concepts in the retrieval process. This has allowed users to specify the visual perceptual cues of the semantic categories they are looking for. Statistical tests also support the findings when comparing to retrieval of semantic categories operating on the low-level image features.

The experimental results showed that using the visual conceptual information to describe image contents achieved the goal of this thesis of narrowing the semantic gap between the low-level image features and high-level semantic categories. However, this approach still merits further exploration.

7.3 Future work

Future directions include

1. Expanding the framework for the extraction and characterization of the visual shape concepts. This may include expanding the shape lattice by adding additional shapes. It can be also further developed and expanded to accommodate some domain specific fields in computer vision and object recognition. Moreover, domain specific fields can utilize specific shape sub-lattice(s) from the presented shape lattice for specific tasks related to describing the detailed shapes of objects.

2. Extending the work on shapes for more complicated retrieval schemes. For example, a query expansion framework that maps the terms in queries into their perceptual properties for retrieval. For example, “a photo of a sun”, could be mapped to the corresponding visual concepts such as circular, along with perceptual color and texture descriptors, to facilitate the retrieval process.

3. Investigating more multimedia features that can be mapped to high-level perceptual features. For example, describing the spatial relations between image objects using adequate perceptual descriptors and motion features representation for video classification and retrieval.

4. Tousch et al. (2012) have shown that using semantic ontologies and hierarchies achieves better results than using a set of keywords to label images. Section 6.4.1 of Chapter 6 showed that the semantic categories were re-organized in a visual ontology. The classification models were trained using only the semantic concepts at the leaf level. Future work may include incorporating a technique to

measure the semantic similarities in the concepts in ontologies to be integrated with semantic-based image retrieval which may enhance the retrieval results. With such a system, all semantic concepts in the lattice might be incorporated in the retrieval process.

5. Integrating the presentation of the visual conceptual information in other multimedia representations that have proven to be strong in representing multimedia contents should be investigated. This includes presenting a perceptual representation of the bags-of-visual-words (BOW) representation to describe multimedia content. The BOW representation has been widely used and has been shown to be a strong technique for representing multimedia content for several tasks such as image classification, object classification, image and video retrieval, etc. (Csurka et al.; 2004; Yang et al.; 2007; Deselaers; 2008; Uijlings et al.; 2009). However, several researchers have pointed out that the main issue with the BOW representation is that it does not take into account the spatial information of image content (Lazebnik et al.; 2006; Yang et al.; 2007; Ozdemir and Aksoy; 2010). As far as the perceptual meaning of image features is concerned, the BOW representation does not take it into account which might be worth further investigation.

6. Investigating the integration of the visual conceptual information with powerful local features detection algorithms such as Scale-invariant feature transform (SIFT) (Lowe; 1999) and Speeded up robust features (SURF) (Bay et al.; 2008). Moreover, possible conceptual representation of SIFT and SURF is worth more exploration.

7. Employing the visual conceptual information for the task of semantic scene analysis rather than region-based classification only. This requires obtaining more visual features in order to be able to capture more information from images. In such a case, the visual shape concepts might be eliminated as the semantic scene analysis involves analyzing images without applying segmentation to

images. However, incorporating shape features might be still included in such a way that different image regions in images are extracted and analysed individually. Then the classified image regions in the same image contribute through another framework to infer the semantic meaning of the image scenery.

8. The experimental chapter presented an empirical comparison between six learning algorithms. The experiments presented can be extended to include more learning algorithms to classify images, possibly using different sets of image features on different learning algorithms and compare their performances using different evaluation metrics. For example, several texture feature extraction algorithms as well as several color features in different color spaces could be compared. Using different image datasets and a wider set of accuracy measures might be considered as well. It was noticed that there is a lack of a formal work comparing multiclass classification using supervised machine learning algorithms to classify image datasets. Caruana and Niculescu-Mizil (2006) present a large scale empirical comparison using binary datasets. However, this study does not concentrate on multimedia related datasets. Recently, Zhang et al. (2012) present an extensive review of supervised learning algorithms used for the task of automatic image annotation. No experimental comparisons are conducted to compare the reviewed learning algorithms. Deselaers et al. (2008) present a comparison on a wide variety of image features. Similarity distance between query and database images was utilized only to classify/retrieve images. A large scale comprehensive study on of classifying multimedia contents using a wide set of multimedia features with different learning algorithms, a different dataset, and different classification metrics that stresses the point of classifying multimedia visual contents might be an interesting study.

7.4 Summary

The fields of image classification, automatic image annotation, and image retrieval are expanding research areas. A significant effort and a very strong research community all over the world have been developing methods and techniques to improve these fields. As these research areas develop constantly, the ideas and techniques presented in this thesis can be deployed in areas of computer vision and multimedia content retrieval. In particular, having answered the research questions of this thesis affirmatively, this approach is worth more investigation and expansion to potentially improve the accuracy of multimedia content classification or retrieval. The future directions presented in Section 7.3 aim to provide some potential ideas for further investigation that build on the work presented in this thesis.

Bibliography

Abbasi, S., Mokhtarian, F. and Kittler, J. (1999). Curvature scale space image in shape similarity retrieval, *Multimedia Systems* **7**(6): 467–476.

Abe, S. (2005). *Support vector machines for pattern classification*, Advances in Pattern Recognition, Springer.

Albatal, R., Mulhem, P. and Chiaramella, Y. (2010). Visual Phrases for automatic images annotation, *Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI)*, IEEE, pp. 1–6.

Bach, J., Fuller, C., Gupta, A., Hampapur, A., Horowitz, B., Humphrey, R., Jain, R. and Shu, C. (1996). Virage image search engine: an open framework for image management, *Proceedings of SPIE Conference on Storage and retrieval of Image and Video Databases*, pp. 76–87.

Bannour, H. and Hudelot, C. (2011). Towards ontologies for image interpretation and annotation, *Proceedings of the International Workshop on Content-Based Multimedia Indexing (CBMI)*, IEEE, pp. 211–216.

Battiato, S., Farinella, G., Gallo, G. and Ravi, D. (2008). Scene categorization using bag of textons on spatial hierarchy, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 2536–2539.

Bay, H., Ess, A., Tuytelaars, T. and Van Gool, L. (2008). Speeded-up robust features (surf), *Computer Vision and Image Understanding* **110**(3): 346–359.

Behmo, R., Marcombes, P., Dalalyan, A. and Prinet, V. (2010). Towards optimal naive bayes nearest neighbor, *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, pp. 171–184.

Belkhatir, M. (2005). Combining visual semantics and texture characterizations for precision-oriented automatic image retrieval, *Proceedings of the European Conference on Information Retrieval (ECIR)*, Springer, pp. 457–474.

Belkhatir, M. (2009). An operational model based on knowledge representation for querying the image content with concepts and relations, *Multimedia Tools and Applications (MTAP)* **43**(1): 1–23.

Belpaeme, T. (2001). Simulating the formation of color categories, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 393–400.

Berk, T., Brownston, L. and Kaufman, A. (1982). New color-naming system for graphics languages, *IEEE Computer Graphics & Applications* **2**(3): 37–44.

Berlin, B. and Kay, P. (1991). *Basic color terms: Their universality and evolution*, University of California Press.

Beygelzimer, A., Kakade, S. and Langford, J. (2006). Cover trees for nearest neighbor, *Proceedings of the International Conference on Machine Learning (ICML)*, ACM, pp. 97–104.

Bhushan, N., Ravishankar Rao, B. and Lohse, G. (1997). The texture lexicon: Understanding the categorization of visual texture terms and their relationship to texture images, *Cognitive Science* **21**(2): 219–246.

Black, G. and Weer, P. (1936). A proposed terminology for shape classifications of artifacts, *American Antiquity* **1**(4): 280–294.

Boiman, O., Shechtman, E. and Irani, M. (2008). In defense of nearest-neighbor based image classification, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 1–8.

Bosch, A., Zisserman, A. and Muoz, X. (2007). Image classification using random forests and ferns, *Proceedings of the International Conference on Computer Vision (ICCV)*, IEEE, pp. 1–8.

Breiman, L. (1984). *Classification and regression trees*, Chapman & Hall/CRC.

Breiman, L. (2001). Random forests, *Machine Learning* **45**(1): 5–32.

Carson, C., Belongie, S., Greenspan, H. and Malik, J. (2002). Blobworld: Image segmentation using expectation-maximization and its application to image querying, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **24**(8): 1026–1038.

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms, *Proceedings of the International Conference on Machine Learning (ICML)*, ACM, pp. 161–168.

Chan, T. and Vese, L. (2001). Active contours without edges, *IEEE Transactions on Image Processing* **10**(2): 266–277.

Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A library for support vector machines, *ACM Transactions on Intelligent Systems and Technology* **2**(3): 1–27.

Chang, N. and Fu, K. (1980). Query-by-pictorial-example, *IEEE Transactions on Software Engineering* **SE-6**(6): 519–524.

Chang, S., Yan, C., Dimitroff, D. and Arndt, T. (1988). An intelligent image database system, *IEEE Transactions on Software Engineering* **14**(5): 681–688.

Chapelle, O., Haffner, P. and Vapnik, V. (1999). Support vector machines for histogram-based image classification, *IEEE Transactions on Neural Networks* **10**(5): 1055–1064.

Chatzichristofis, S., Zagoris, K., Boutalis, Y. and Papamarkos, N. (2010). Accurate image retrieval based on compact composite descriptors and relevance feedback information, *International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI)* **24**(2): 207–244.

Chen, Z., Wenyin, L., Zhang, F., Li, M. and Zhang, H. (2001). Web mining for web image retrieval, *Journal of the American Society for Information Science and Technology* **52**(10): 831–839.

Cheng, Y. and Chen, S. (2003). Image classification using color, texture and regions, *Image and Vision Computing* **21**(9): 759–776.

Clinchant, S., Ah-Pine, J. and Csurka, G. (2011). Semantic combination of textual and visual information in multimedia retrieval, *Proceedings of the ACM International Conference on Multimedia Retrieval (ICMR)*, ACM, pp. 1–8.

Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **24**(5): 603–619.

Conway, D. (1992). An experimental comparison of three natural language colour naming models, *Proceedings of the East-West International Conference on Human-Computer Interaction*, pp. 328–339.

Cox, I., Miller, M., Minka, T., Papathomas, T. and Yianilos, P. (2000). The bayesian image retrieval system, pichunter: theory, implementation, and psychophysical experiments, *IEEE Transactions on Image Processing* **9**(1): 20–37.

Crucianu, M., Ferecatu, M. and Boujemaa, N. (2004). Relevance feedback for image retrieval: a short survey, *State of the Art in Audiovisual Content-Based Retrieval, Information Universal Access and Interaction, Including Data models and Languages* pp. 1–20.

Csurka, G., Dance, C., Fan, L., Willamowski, J. and Bray, C. (2004). Visual categorization with bags of keypoints, *Workshop on statistical learning in computer vision-The European Conference on Computer Vision (ECCV)*, pp. 1–22.

Datta, R. (2010). *Semantics and aesthetics inference for image search: Statistical learning approaches*, PhD thesis, The Pennsylvania State University.

Datta, R., Joshi, D., Li, J. and Wang, J. (2008). Image retrieval: Ideas, influences, and trends of the new age, *ACM Computing Surveys (CSUR)* **40**(2): 1–60.

Deng, Y. and Manjunath, B. (2001). Unsupervised segmentation of color-texture regions in images and video, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **23**(8): 800–810.

Deselaers, T. (2008). *Image retrieval, object recognition, and discriminative models*, PhD thesis, Universitätsbibliothek.

Deselaers, T., Keysers, D. and Ney, H. (2008). Features for image retrieval: an experimental comparison, *Information Retrieval* **11**(2): 77–107.

Domingos, P. and Pazzani, M. (1997). On the optimality of the simple bayesian classifier under zero-one loss, *Machine Learning* **29**(2): 103–130.

Duygulu, P., Barnard, K., De Freitas, J. and Forsyth, D. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, pp. 349–354.

Eidenberger, H. (2006). Evaluation and analysis of similarity measures for content-based visual information retrieval, *Multimedia Systems* **12**(2): 71–87.

Escalante, H., Hernández, C., Gonzalez, J., López-López, A., Montes, M., Morales, E., Enrique Sucar, L., Villaseñor, L. and Grubinger, M. (2010). The segmented and annotated IAPR TC-12 benchmark, *Computer Vision and Image Understanding* **114**(4): 419–428.

Estrada, F. and Jepson, A. (2005). Quantitative evaluation of a novel image segmentation algorithm, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 1132–1139.

Estrada, F. and Jepson, A. (2009). Benchmarking image segmentation algorithms, *International Journal of Computer Vision* **85**(2): 167–181.

Fan, R., Chang, K., Hsieh, C., Wang, X. and Lin, C. (2008). LIBLINEAR: A library for large linear classification, *The Journal of Machine Learning Research* **9**: 1871–1874.

Fauzi, F. (2012). *Integrating the surrounding image information within a high-level conceptual framework for symbolic image indexing and retrieval on the WWW*, PhD thesis, Faculty of Information Technology, Monash University.

Fayyad, U. and Irani, K. (1993). Multi-interval discretization of continuous-valued attributes for classification learning, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1022–1027.

Fei-Fei, L., Fergus, R. and Perona, P. (2007). Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories, *Computer Vision and Image Understanding* **106**(1): 59–70.

Felzenszwalb, P. and Huttenlocher, D. (1998). Image segmentation using local variation, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 98–104.

Felzenszwalb, P. and Huttenlocher, D. (2004). Efficient graph-based image segmentation, *International Journal of Computer Vision* **59**(2): 167–181.

Fergus, R., Perona, P. and Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 264–271.

Flickner, M., Sawhney, H., Niblack, W., Ashley, J., Huang, Q., Dom, B., Gorkani, M., Hafner, J., Lee, D., Petkovic, D. et al. (1995). Query by image and video content: The QBIC system, *Computer* **28**(9): 23–32.

Freixenet, J., Muñoz, X., Raba, D., Martí, J. and Cufí, X. (2002). Yet another survey on image segmentation: Region and boundary information integration, pp. 21–25.

Friedman, J., Bentley, J. and Finkel, R. (1977). An algorithm for finding best matches in logarithmic expected time, *ACM Transactions on Mathematical Software (TOMS)* **3**(3): 209–226.

Ge, F., Wang, S. and Liu, T. (2006). Image-segmentation evaluation from the perspective of salient object extraction, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 1146–1153.

Gevers, T. and Smeulders, A. (2000). Pictoseek: Combining color and shape invariant features for image retrieval, *IEEE Transactions on Image Processing* **9**(1): 102–119.

Gong, Y., Chuan, C. and Xiaoyi, G. (1996). Image indexing and retrieval based on color histograms, *Multimedia Tools and Applications (MTAP)* **2**(2): 133–156.

González, R. and Woods, R. (2008). *Digital image processing*, Pearson/Prentice Hall.

Grubinger, M., Clough, P., Muller, H. and Deselaers, T. (2006). The IAPR TC-12 benchmark: A new evaluation resource for visual information systems, *International Workshop OntoImage*, pp. 13–23.

Gu, C. (1995). *Multivalued morphology and segmentation-based coding*, PhD thesis, PhD thesis, Signal Processing Lab of Swiss Federal Institute of Technology at Lausanne.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I. (2009). The weka data mining software: an update, *ACM SIGKDD Explorations Newsletter* **11**(1): 10–18.

Han, J., Kamber, M. and Pei, J. (2011). *Data mining: concepts and techniques*, Morgan Kaufmann.

Han, J., Ngan, K., Li, M. and Zhang, H. (2004). Learning semantic concepts from user feedback log for image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, pp. 995–998.

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

He, S. and Jia, J. (2010). Image annotation by sparse logistic regression, *Proceedings of the Pacific Rim Conference on Advances in Multimedia Information Processing (PCM)*, Springer, pp. 22–30.

Hoi, S., Jin, R. and Lyu, M. (2009). Batch mode active learning with applications to text categorization and image retrieval, *IEEE Transactions on Knowledge and Data Engineering* **21**(9): 1233–1248.

Hou, X. and Zhang, L. (2007). Color conceptualization, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 265–268.

Hsu, C. and Lin, C. (2002). A comparison of methods for multiclass support vector machines, *IEEE Transactions on Neural Networks* **13**(2): 415–425.

Hu, R., Ruger, S., Song, D., Liu, H. and Huang, Z. (2008). Dissimilarity measures for content-based image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, pp. 1365–1368.

Huang, J., Kumar, S., Mitra, M., Zhu, W. and Zabih, R. (1997). Image indexing using color correlograms, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 762–765.

Huang, J., Kumar, S. and Zabih, R. (1998). An automatic hierarchical image classification scheme, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 219–228.

Hyvönen, E., Saarela, S., Styrman, A. and Viljanen, K. (2003). Ontology-based image retrieval, *Proceedings of World Wide Web (Poster)*.

Iqbal, Q. and Aggarwal, J. (2002). Retrieval by classification of images containing large manmade objects using perceptual grouping, *Pattern Recognition* **35**(7): 1463–1479.

Jakulin, A. (2005). *Machine learning based on attribute interactions*, PhD thesis, Ljubljana, Slovenia: University of Ljubljana.

Jarrar, R. and Belkhatir, M. (2010). Towards automated conceptual shape-based characterization an application to symbolic image retrieval, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 2673–2676.

Jin, W., Shi, R. and Chua, T. (2004). A semi-naïve bayesian method incorporating clustering with pair-wise constraints for auto image annotation, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 336–339.

Jing, F., Li, M., Zhang, H. and Zhang, B. (2004). Relevance feedback in region-based image retrieval, *IEEE Transactions on Circuits and Systems for Video Technology* **14**(5): 672–681.

Khan, R., Hanbury, A. and Stoettinger, J. (2010). Skin detection: A random forest approach, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 4613–4616.

Kherfi, M., Ziou, D. and Bernardi, A. (2004). Image retrieval from the world wide web: Issues, techniques, and systems, *ACM Computing Surveys (CSUR)* **36**(1): 35–67.

Kononenko, I. (1990). Comparison of inductive and naive bayesian learning approaches to automatic knowledge acquisition, *Current trends in knowledge acquisition* pp. 190–197.

Kotsiantis, S. and Kanellopoulos, D. (2006). Discretization techniques: A recent survey, *GESTS International Transactions on Computer Science and Engineering* **32**(1): 47–58.

Ladret, P. and Guérin-Dugué, A. (2001). Categorisation and retrieval of scene photographs from jpeg compressed database, *Pattern Analysis & Applications* **4**(2): 185–199.

Lazebnik, S., Schmid, C. and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 2, IEEE, pp. 2169–2178.

Leistner, C., Saffari, A. and Bischof, H. (2010). MIForests: multiple-instance learning with randomized trees, *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, pp. 29–42.

Leow, W. and Lai, S. (1999). Invariant matching of texture for content-based image retrieval, *Workshop on Texture Analysis in Machine Vision*, pp. 1–7.

Lew, M., Sebe, N., Djeraba, C. and Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges, *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)* **2**(1): 1–19.

Lim, J. and Jin, J. (2005). Combining intra-image and inter-class semantics for consumer image retrieval, *Pattern Recognition* **38**(6): 847–864.

Lim, J., Tian, Q. and Mulhem, P. (2003). Home photo content modeling for personalized event-based retrieval, *IEEE Multimedia* **10**(4): 28–37.

Liu, H., Hussain, F., Tan, C. and Dash, M. (2002). Discretization: An enabling technique, *Data Mining and Knowledge Discovery* **6**(4): 393–423.

Liu, Y., Zhang, D. and Lu, G. (2008). Region-based image retrieval with high-level semantics using decision tree learning, *Pattern Recognition* **41**(8): 2554–2570.

Liu, Y., Zhang, D., Lu, G. and Ma, W. (2004). Region-based image retrieval with perceptual colors, *Proceedings of the Pacific Rim Conference on Advances in Multimedia Information Processing (PCM)*, Springer, pp. 931–938.

Liu, Y., Zhang, D., Lu, G. and Ma, W. (2005). Region-based image retrieval with high-level semantic color names, *Proceedings of the International Multimedia Modelling Conference (MMM)*, IEEE, pp. 180–187.

Liu, Y., Zhang, D., Lu, G. and Ma, W. (2007). A survey of content-based image retrieval with high-level semantics, *Pattern Recognition* **40**(1): 262–282.

Liu, Y., Zhang, J., Tjondronegoro, D., Geva, S. and Li, Z. (2010). Mid-level concept learning with visual contextual ontologies and probabilistic inference for image annotation, *Proceedings of the International Multimedia Modelling Conference (MMM)*, Springer, pp. 229–239.

Liu, Y., Zhang, J., Tjondronegoro, D. and Geve, S. (2007). A shape ontology framework for bird classification, *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, pp. 478–484.

Lowe, D. (1999). Object recognition from local scale-invariant features, *Proceedings of the International Conference on Computer Vision (ICCV)*, Vol. 2, IEEE, pp. 1150–1157.

Lu, G. and Sajjanhar, A. (1999). Region-based shape representation and similarity measure suitable for content-based image retrieval, *Multimedia Systems* **7**(2): 165–174.

Luo, J. and Savakis, A. (2001). Indoor vs outdoor classification of consumer photographs using low-level and semantic features, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 745–748.

MacArthur, S., Brodley, C. and Shyu, C. (2000). Relevance feedback decision trees in content-based image retrieval, *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries*, IEEE, pp. 68–72.

Magalhães, J. and Rüger, S. (2006). Logistic regression of generic codebooks for semantic image retrieval, *Proceedings of the International Conference on Image and Video Retrieval (CIVR)*, Springer, pp. 41–50.

Manjunath, B. and Ma, W. (1996). Texture features for browsing and retrieval of image data, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **18**(8): 837–842.

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

Manjunath, B., Salembier, P. and Sikora, T. (2002). *Introduction to MPEG-7: multimedia content description interface*, Vol. 1, John Wiley & Sons Inc.

Martin, D., Fowlkes, C., Tal, D. and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, *Proceedings of the International Conference on Computer Vision (ICCV)*, IEEE, pp. 416–423.

Mehrotra, S., Rui, Y., Ortega-Binderberger, M. and Huang, T. (1997). Supporting content-based queries over images in mars, *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, IEEE, pp. 632–633.

Mezaris, V., Kompatsiaris, I. and Strintzis, M. (2003). An ontology approach to object-based image retrieval, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 511–514.

Miller, G. (1995). Wordnet: a lexical database for english, *Communications of the ACM* **38**(11): 39–41.

Mitchell, T. (1997). *Machine learning.*, Burr Ridge, IL: McGraw Hill.

Mojsilovic, A. (2002). A method for color naming and description of color composition in images, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 789–792.

Moosmann, F., Nowak, E. and Jurie, F. (2008). Randomized clustering forests for image classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **30**(9): 1632–1646.

Nakagawa, a., Arai, S., Kutics, A. and Tanaka, H. (2004). Mapping image segments to words for image retrieval, *Proceedings of the IEEE Workshop on Multimedia Signal Processing (MMSP)*, IEEE, pp. 438–441.

Nilsback, M.-E. and Zisserman, A. (2006). A Visual Vocabulary for Flower Classification, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 1447–1454.

Nixon, M. and Aguado, A. (2008). *Feature extraction and image processing*, Academic Press.

Nock, R. and Nielsen, F. (2004). Statistical region merging, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **26**(11): 1452–1458.

Ogle, V. and Stonebraker, M. (1995). Chabot: Retrieval from a relational database of images, *IEEE Computer* **28**(9): 40–48.

Olejnik, S., Li, J., Supattathum, S. and Huberty, C. (1997). Multiple testing and statistical power with modified bonferroni procedures, *Journal of Educational and Behavioral Statistics* **22**(4): 389–406.

Ozdemir, B. and Aksoy, S. (2010). Image classification using subgraph histogram representation, *Proceedings of the International Conference on Pattern Recognition (ICPR)*, Vol. 10, IEEE, pp. 1112–1115.

Papadopoulos, G. T., Mezaris, V., Kompatsiaris, I. and Strintzis, M. G. (2010). Probabilistic combination of spatial context with visual and co-occurrences information for semantic image analysis, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 3205–3208.

Pentland, A., Picard, R. and Sclaroff, S. (1996). Photobook: Content-based manipulation of image databases, *International Journal of Computer Vision* **18**(3): 233–254.

Petrou, M. and Sevilla, P. (2006). *Image processing: dealing with texture*, John Wiley & Sons Inc.

Picard, R., Minka, T., of Technology. Media Laboratory. Vision, M. I. and Group, M. (1995). Vision texture for annotation, *Multimedia Systems* **3**(1): 3–14.

Platt, J. (1999). Probabilistic outputs for support vector machines, *Advances in Large Margin Classifiers* pp. 61–74.

Qi, X. and Han, Y. (2007). Incorporating multiple SVMs for automatic image annotation, *Pattern Recognition* **40**(2): 728–741.

Quinlan, J. (1986). Induction of decision trees, *Machine learning* **1**(1): 81–106.

Quinlan, J. (1993). *C4.5: programs for machine learning*, Morgan kaufmann.

Rao, A. and Lohse, G. (1993). Identifying high level features of texture perception, *CVGIP: Graphical Models and Image Processing* **55**(3): 218–233.

Rifkin, R. and Klautau, A. (2004). In defense of one-vs-all classification, *The Journal of Machine Learning Research* **5**: 101–141.

Rubner, Y., Tomasi, C. and Guibas, L. (2000). The earth mover's distance as a metric for image retrieval, *International Journal of Computer Vision* **40**(2): 99–121.

Rui, Y., Huang, T. and Chang, S. (1999). Image retrieval: Current techniques, promising directions, and open issues, *Journal of Visual Communication and Image Representation* **10**(1): 39–62.

Rui, Y., Huang, T. and Mehrotra, S. (1997). Content-based image retrieval with relevance feedback in MARS, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 815–818.

Rusinol, M., Noorbakhsh, F., Karatzas, D., Valveny, E. and Lladós, J. (2010). Perceptual image retrieval by adding color information to the shape context descriptor, *Proceedings of the International Conference on Pattern Recognition (ICPR)*, IEEE, pp. 1594–1597.

Sande, K. and Gevers, T. (2010). University of amsterdam at the visual concept detection and annotation tasks, *ImageCLEF* pp. 343–358.

Sethi, I., Coman, I. and Stan, D. (2001). Mining association rules between low-level image features and high-level concepts, *Proceedings of the SPIE Data Mining and Knowledge Discovery*, pp. 279–290.

Sharp, T. (2008). Implementing decision trees and forests on a gpu, *Proceedings of the European Conference in Computer Vision (ECCV)*, Springer, pp. 595–608.

Shen, H., Ooi, B. and Tan, K. (2000). Giving meanings to www images, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 39–47.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **22**(8): 888–905.

Shi, R., Feng, H., Chua, T. and Lee, C. (2004). An adaptive image content representation and segmentation approach to automatic image annotation, *Proceedings of the International Conference on Image and Video Retrieval (CIVR)*, Springer, pp. 545–554.

Shotton, J., Johnson, M. and Cipolla, R. (2008). Semantic texton forests for image categorization and segmentation, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 1–8.

Siu, W. and Zhang, H. (2003). *Multimedia information retrieval and management: Technological fundamentals and applications*, Springer Verlag.

Smeulders, A., Worring, M., Santini, S., Gupta, A. and Jain, R. (2000). Content-based image retrieval at the end of the early years, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **22**(12): 1349–1380.

Smith, J. and Chang, S. (1997). Visualseek: a fully automated content-based image query system, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 87–98.

Snoek, C., Worring, M. and Smeulders, A. (2005). Early versus late fusion in semantic video analysis, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 399–402.

Stricker, M. and Orengo, M. (1995). Similarity of color images, *Proceedings of SPIE Storage and Retrieval for Image and Video Databases*, pp. 381–392.

Styrman, A. (2005). *Ontology-based image annotation and retrieval*, Master's thesis, University of Helsinki.

Su, Z., Zhang, H., Li, S. and Ma, S. (2003). Relevance feedback in content-based image retrieval: Bayesian framework, feature subspaces, and progressive learning, *IEEE Transactions on Image Processing* **12**(8): 924–937.

Swain, M. and Ballard, D. (1991). Color indexing, *International Journal of Computer Vision* **7**(1): 11–32.

Tamura, H., Mori, S. and Yamawaki, T. (1978). Textural features corresponding to visual perception, *IEEE Transactions on Systems, Man, and Cybernetics* **8**(6): 460–473.

Teague, M. (1980). Image analysis via the general theory of moments, *Journal of the Optical Society of America (JOSA)* **70**(8): 920–930.

Tong, S. and Chang, E. (2001). Support vector machine active learning for image retrieval, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 107–118.

Tousch, A.-M., Herbin, S. and Audibert, J.-Y. (2012). Semantic hierarchies for image annotation: A survey, *Pattern Recognition* **45**(1): 333–345.

Tremeau, A. and Borel, N. (1997). A region growing and merging algorithm to color segmentation, *Pattern Recognition* **30**(7): 1191–1203.

Uijlings, J., Smeulders, A. and Scha, R. (2009). Real-time bag of words, approximately, *Proceeding of the ACM International Conference on Image and Video Retrieval (CIVR)*, ACM, pp. 1–8.

Unnikrishnan, R., Pantofaru, C. and Hebert, M. (2007). Toward objective evaluation of image segmentation algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **29**(6): 929–944.

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

van de Sande, K., Gevers, T. and Smeulders, A. (2010). The university of amsterdam's concept detection system at imageclef 2009, *Multilingual Information Access Evaluation II. Multimedia Experiments* pp. 261–268.

van de Sande, K., Gevers, T. and Snoek, C. (2010). Evaluating color descriptors for object and scene recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **32**(9): 1582–1596.

van de Weijer, J. and Schmid, C. (2007). Applying Color Names to Image Description, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 493–496.

van de Weijer, J., Schmid, C., Verbeek, J. and Larlus, D. (2009). Learning color names for real-world applications, *IEEE Transactions on Image Processing* **18**(7): 1512–1523.

van Gemert, J., Snoek, C., Veenman, C., Smeulders, A. and Geusebroek, J. (2010). Comparing compact codebooks for visual categorization, *Computer Vision and Image Understanding* **114**(4): 450–462.

Vapnik, V. (1998). *Statistical learning theory*, Wiley-Interscience.

Vasconcelos, N. and Lippman, A. (2000a). Learning over multiple temporal scales in image databases, *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, pp. 33–47.

Vasconcelos, N. and Lippman, A. (2000b). A probabilistic architecture for content-based image retrieval, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 216–221.

Veltkamp, R. and Tanase, M. (2002). Content-based image retrieval systems: A survey, *Technical report*, Department of Computing Science, Utrecht University.

Wang, J., Li, J. and Wiederhold, G. (2001). SIMPLIcity: Semantics-sensitive integrated matching for picture libraries, *IEEE Transactions on pattern analysis and machine intelligence (TPAMI)* **23**: 947–963.

Wong, R. and Leung, C. (2008). Automatic semantic annotation of real-world web images, *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* **30**(11): 1933–1944.

Wu, P. and Manjunath, B. (2001). Adaptive nearest neighbor search for relevance feedback in large image databases, *Proceedings of the ACM International Conference on Multimedia (ACM MM)*, ACM, pp. 89–97.

Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G., Ng, A., Liu, B., Yu, P. et al. (2008). Top 10 algorithms in data mining, *Knowledge and Information Systems* **14**(1): 1–37.

Xiao, Y., Chua, T. and Lee, C. (2006). Fusion of region and image-based techniques for automatic image annotation, *Proceedings of the International Multimedia Modelling Conference (MMM)*, Springer, pp. 247–258.

Yang, J., Jiang, Y., Hauptmann, A. and Ngo, C. (2007). Evaluating bag-of-visual-words representations in scene classification, *Proceedings of the International on Workshop on Multimedia Information Retrieval (MIR)*, ACM, pp. 197–206.

Yang, M., Kpalma, K. and Ronsin, J. (2008). A survey of shape feature extraction techniques, *Pattern Recognition Techniques, Technology and Applications, Peng-Yeng Yin (Ed.)* .

Yang, Y. and Webb, G. (2001). Proportional k-interval discretization for naive-bayes classifiers, *Proceedings of the European Conference on Machine Learning (ECML)*, Springer, pp. 564–575.

Yang, Y. and Webb, G. (2002). A comparative study of discretization methods for naive-bayes classifiers, *Proceedings of the Pacific Rim Knowledge Acquisition Workshop (PKAW)*, pp. 159–173.

Yang, Y. and Webb, G. (2009). Discretization for naive-bayes learning: managing discretization bias and variance, *Machine Learning* **74**(1): 39–74.

Yang, Z., Chung, F.-L. and Shitong, W. (2009). Robust fuzzy clustering-based image segmentation, *Applied Soft Compututing* **9**(1): 80–84.

Zahn, C. and Roskies, R. (1972). Fourier descriptors for plane closed curves, *IEEE Transactions on Computers* **100**(3): 269–281.

Zhang, D., Islam, M., Lu, G. and Hou, J. (2009). Semantic image retrieval using region based inverted file, *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, pp. 242–249.

Zhang, D. and Lu, G. (2001). Content-based shape retrieval using different shape descriptors: A comparative study, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1139–1142.

Zhang, D. and Lu, G. (2002a). A comparative study of Fourier descriptors for shape representation and retrieval, *Proceedings of the 5th Asian Conference on Computer Vision (ACCV)*, pp. 646–651.

Zhang, D. and Lu, G. (2002b). Shape-based image retrieval using generic fourier descriptor, *Signal Processing: Image Communication* **17**(10): 825–848.

Zhang, D. and Lu, G. (2003a). Content-based shape retrieval using different shape descriptors: A comparative study, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, IEEE, pp. 1139–1142.

Zhang, D. and Lu, G. (2003b). Evaluation of similarity measurement for image retrieval, *Proceedings of the International Conference on Neural Networks and Signal Processing*, IEEE, pp. 928–931.

Zhang, D. and Lu, G. (2004). Review of shape representation and description techniques, *Pattern recognition* **37**(1): 1–19.

Zhang, D., Monirul Islam, M. and Lu, G. (2012). A review on automatic image annotation techniques, *Pattern Recognition* **45**(1): 346–362.

Zhang, H., Berg, A., Maire, M. and Malik, J. (2006). SVM-KNN: Discriminative nearest neighbor classification for visual category recognition, *Proceedings of the IEEE International Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 2126–2136.

Zhang, H., Fritts, J. and Goldman, S. (2008). Image segmentation evaluation: A survey of unsupervised methods, *Computer Vision and Image Understanding* **110**(2): 260–280.

Zhang, J., Marszalek, M., Lazebnik, S. and Schmid, C. (2006). Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study, *International Journal of Computer Vision* **73**(2): 213–238.

Zhang, L., Lin, F. and Zhang, B. (2001). Support vector machine learning for image retrieval, *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, IEEE, pp. 721–724.

Zhou, X. and Huang, T. (2003). Relevance feedback in image retrieval: A comprehensive review, *Multimedia Systems* **8**(6): 536–544.

Zhou, Z. and Dai, H. (2007). Exploiting image contents in web search, *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 2922–2927.

Zhuang, Y., Liu, X. and Pan, Y. (1999). Apply semantic template to support content-based image retrieval, *Proceedings of the SPIE, Storage and Retrieval for Media Databases*, pp. 442–449.

Appendix A

Detailed results of the semantic-based region classification

This appendix presents the detailed classification scores of the semantic-based region classification presented in Section 6.4.

Table A.1: *Naive Bayes - Visual concepts*

Subset	Discretization method	Precision	Recall	F-score
subset1	kernel estimation	13.5	10.8	7.6
—	single normal	19.9	11.8	9.1
—	EMD	41.6	0.21	21.4
subset2	kernel estimation	22.4	20.2	13.9
—	single normal	20.3	13.4	11.7
—	EMD	44.6	34.4	35.9
subset3	kernel estimation	20.8	11.7	10.3
—	single normal	17.2	12.1	10
—	EMD	49.1	35.5	37.4

Table A.2: *Support Vector Machines - Visual concepts*

Subset	Kernel	Precision	Recall	F-score
subset1	Linear	47.9	31.1	33.6
—	Polynomial 2 degrees	46.9	32.8	35.5
—	Polynomial 3 degrees	46.7	32.8	35.6
—	RBF	38.9	28.7	31
—	Sigmoid	45.3	30	32.3
subset2	Linear	44.3	35.8	36.7
—	Polynomial 2 degrees	51.9	41.3	44.7
—	Polynomial 3 degrees	52.7	41.9	45.6
—	RBF	40.5	28.9	32.5
—	Sigmoid	48	35.9	37.7
subset3	Linear	47.8	34	34.8
—	Polynomial 2 degrees	54.7	41.7	46
—	Polynomial 3 degrees	55.3	42.4	46.8
—	RBF	44.8	30.9	35.3
—	Sigmoid	48.7	37.3	39

Table A.3: *Decision tree - Visual concepts*

Subset	Algorithm	Split criterion	Pruning factor	Precision	Recall	F-score
subset1	C4.5	1	0.05	23.2	12.7	13.8
—	—	—	0.10	21.1	12.5	13
—	—	—	0.15	30.4	12.2	13.3
—	—	—	0.25	30.4	13.2	14.7
—	—	—	0.5	30.4	13.2	14.7
—	—	2	0.05	35	24.1	25.2
—	—	—	0.10	33.4	23.8	24.6
—	—	—	0.15	33.6	23.7	24.7
—	—	—	0.25	33.3	23.5	24.6
—	—	—	0.5	33.2	22.9	24.3
—	NBDT			18.4	11.1	12.4
subset2	C4.5	1	0.05	42.1	31	35
—	—	—	0.10	41.1	29.8	34

Table A.3 – continued from previous page

Subset	Algorithm	Split criterion	Pruning factor	Precision	Recall	F-score
—	—	—	0.15	41.6	28.8	33.5
—	—	—	0.25	41.4	28.7	33.4
—	—	—	0.5	41.5	28.7	33.4
—	—	2	0.05	42.3	30.8	34.8
—	—	—	0.10	41.3	29	33.5
—	—	—	0.15	41.4	29	33.6
—	—	—	0.25	41.5	29	33.6
—	—	—	0.5	41.4	28.9	33.5
—	NBDT			24.7	14.5	17.5
subset3	C4.5	1	0.05	37.2	25.9	29.6
—	—	—	0.10	38.2	25.5	29.6
—	—	—	0.15	38.1	25.1	29.2
—	—	—	0.25	38	24.9	29
—	—	—	0.5	37.7	24.5	28.7
—	—	2	0.05	38.1	26.1	30
—	—	—	0.10	38.1	26	29.9
—	—	—	0.15	38	25.7	29.7
—	—	—	0.25	37.3	25.2	29.2
—	—	—	0.5	37.2	25.1	29.1

Table A.4: *Nearest Neighbor - Visual concepts*

Subset	Search technique	Distance measure	k	Precision	Recall	F-score
subset1	Cover tree	Euclidean	1	29.3	20.5	23.3
—	—	—	2	30.3	21.1	22.9
—	—	—	4	29.3	21.4	23.5
—	—	—	8	32.5	24.4	25.7
—	—	—	16	32.4	24.6	24.6
—	—	—	32	41.8	25.4	25.2
—	—	—	64	35.3	24.3	22.4
—	KDTree	Euclidean	1	29.3	20.5	23.3
—	—	—	2	30.3	21.1	22.9
—	—	—	4	29.3	21.4	23.5

Table A.4 – continued from previous page

Subset	Search technique	Distance measure	k	Precision	Recall	F-score
—	—	—	8	32.5	24.4	25.7
—	—	—	16	32.4	24.6	24.6
—	—	—	32	41.8	25.4	25.2
—	—	—	64	35.3	24.3	22.4
—	Linear	Euclidean	1	29.3	20.5	23.3
—	—	—	2	30.3	21.1	22.9
—	—	—	4	29.3	21.4	23.5
—	—	—	8	32.5	24.4	25.7
—	—	—	16	32.4	24.6	24.6
—	—	—	32	41.8	25.4	25.2
—	—	—	64	35.3	24.3	22.4
—	—	Manhattan	1	32.6	22.3	25
—	—	—	2	33.1	23.1	24.9
—	—	—	4	34.7	25.7	27.4
—	—	—	8	37.3	27.1	28.3
—	—	—	16	39.5	28	28.6
—	—	—	32	36	26.5	25.9
—	—	—	64	41.1	25.4	23.5
subset2	Cover tree	Euclidean	1	35.8	24	28.2
—	—	—	2	34.6	23.6	27.1
—	—	—	4	36.8	26.6	30
—	—	—	8	39.3	29	31.9
—	—	—	16	41.5	30.3	32.6
—	—	—	32	41.8	30.6	31.9
—	—	—	64	42.5	30.2	31
—	KDTree	Euclidean	1	35.8	24	28.2
—	—	—	2	34.6	23.6	27.1
—	—	—	4	36.8	26.6	30
—	—	—	8	39.4	29	31.9
—	—	—	16	41.5	30.3	32.6
—	—	—	32	41.8	30.6	31.9
—	—	—	64	42.5	30.2	31
—	Linear	Euclidean	1	35.8	24	28.2
—	—	—	2	34.6	23.6	27.1
—	—	—	4	36.8	26.6	30
—	—	—	8	39.4	29	31.9
—	—	—	16	41.5	30.3	32.6
—	—	—	32	41.8	30.6	31.9

Table A.4 – continued from previous page

Subset	Search technique	Distance measure	k	Precision	Recall	F-score
—	—	—	64	42.5	30.2	31
—	—	Manhattan	1	38.1	26	30.2
—	—	—	2	38.4	26.8	30.3
—	—	—	4	39.6	29.3	32.6
—	—	—	8	41.1	30.9	33.5
—	—	—	16	43.7	32.8	34.5
—	—	—	32	45.7	33.4	34.5
—	—	—	64	46.5	32.7	33.2
subset3	Cover tree	Euclidean	1	38.1	25.3	29.9
—	—	—	2	38.1	26.6	30.6
—	—	—	4	40.8	29.6	33.5
—	—	—	8	42.5	30.7	34.2
—	—	—	16	44.9	32.4	35.4
—	—	—	32	46	32.5	34.9
—	—	—	64	45.5	32.7	33.9
—	KDTree	Euclidean	1	38.1	25.3	29.9
—	—	—	2	38.1	26.6	30.6
—	—	—	4	40.9	29.6	33.6
—	—	—	8	42.5	30.7	34.2
—	—	—	16	44.9	32.4	35.4
—	—	—	32	46	32.5	34.9
—	—	—	64	45.5	32.7	33.9
—	Linear	Euclidean	1	38.1	25.3	29.9
—	—	—	2	38.1	26.6	30.6
—	—	—	4	40.9	29.6	33.6
—	—	—	8	42.5	30.7	34.2
—	—	—	16	44.9	32.4	35.4
—	—	—	32	46	32.5	34.9
—	—	—	64	45.5	32.7	33.9
—	—	Manhattan	1	40.3	27.9	32.2
—	—	—	2	40.6	29.2	33
—	—	—	4	42.6	31.6	35.2
—	—	—	8	44.9	33.4	36.6
—	—	—	16	46.3	34.6	37.2
—	—	—	32	48.6	35.8	37.6
—	—	—	64	49.1	35.2	36.3

Table A.5: *Logistic Regression - Visual concepts*

Subset	Ridge	Precision	Recall	F-score
subset1	10^{-8}	19.9	19.5	15.7
—	10^{-7}	19.9	19.5	15.7
—	10^{-6}	19.9	19.5	15.7
—	10^{-5}	19.6	19.8	15.8
—	10^{-4}	23.8	22.1	17.2
—	10^{-3}	21	22.8	16.7
—	10^{-2}	36.2	23.6	19.2
—	10^{-1}	43.4	27.2	27
—	10^{0}	49.5	30	31.7
—	10^{1}	46.9	31.1	33
—	10^{2}	48.5	32	34.1
—	10^{3}	47.6	32.1	34.1
—	10^{4}	47.2	32.8	34.6
—	10^{5}	46.7	32	33.7
subset2	10^{-8}	33.6	21.9	18.9
—	10^{-7}	33.6	21.9	18.9
—	10^{-6}	36.8	22.2	19.1
—	10^{-5}	36.3	24.2	20.6
—	10^{-4}	33.2	25.6	20.8
—	10^{-3}	35.8	26.9	22.2
—	10^{-2}	44	31.1	29.9
—	10^{-1}	46.6	33.3	34.6
—	10^{0}	48.3	35	37.1
—	10^{1}	47.8	36.6	38.8
—	10^{2}	47.5	38.2	39.3
—	10^{3}	47.5	38	39.3
—	10^{4}	48.2	38.7	39.8
—	10^{5}	48	38.6	39.4
subset3	10^{-8}	32.9	23.3	20.1
—	10^{-7}	33	23.4	20.2
—	10^{-6}	31.8	23.9	20.5
—	10^{-5}	35.1	25.9	21.5
—	10^{-4}	36.8	27	21.9

Table A.5 – continued from previous page

Subset	Ridge	Precision	Recall	F-score
—	10^{-3}	38.4	29.3	25.1
—	10^{-2}	43.6	33.5	32.5
—	10^{-1}	46.8	35.8	36.5
—	10^{0}	48	37.3	38.4
—	10^{1}	48.7	39.5	39.9
—	10^{2}	48.8	39.9	40.4
—	10^{3}	49	40.1	40.7
—	10^{4}	48.8	40	40.5
—	10^{5}	48.9	40.1	40.6

Table A.6: *Random Forest - Visual concepts*

Subset	Size of Forest	Size of features at each split	Precision	Recall	F-score
subset1	50	3	47.1	36.4	38.8
—	—	5	49.7	39	41.9
—	—	10	54	41.7	45.3
—	—	20	51.7	38.8	42.4
—	—	40	52.2	38.1	42.2
—	100	3	52.2	39	42.3
—	—	5	54.6	41.8	45.2
—	—	10	52	41.2	44.1
—	—	20	50.9	39.1	42.3
—	—	40	52.2	38	41.7
—	500	3	57.6	43.5	45.9
—	—	5	56.7	43	46.2
—	—	10	55.9	43.2	46.5
—	—	20	54.1	40.8	44.5
—	—	40	52.9	39	42.6
—	1000	3	55.8	42.5	45.2
—	—	5	58.3	44.7	47.5
—	—	10	55.8	43.5	46.7
—	—	20	54.3	41.7	44.9

Table A.6 – continued from previous page

Subset	Size of Forest	Size of features at each split	Precision	Recall	F-score
—	—	40	54.8	39.8	43.5
subset2	50	3	56.3	48.8	51.5
—	—	5	56.9	49.6	52.3
—	—	10	58.2	50.8	53.7
—	—	20	56.3	48.3	51.6
—	—	40	54.3	44.9	48.6
—	100	3	58.7	51.5	54
—	—	5	59.9	52.7	55.2
—	—	10	59.4	52.2	55
—	—	20	57.2	49.3	52.5
—	—	40	55.4	46.1	49.8

Table A.7: *Naive Bayes - Baseline*

Discretization method	Precision	Recall	F-score
kernel estimation	13.4	4.4	4.9
single normal	12.7	6.9	4.1
EMD	26.7	16.9	17.5

Table A.8: *Support Vector Machines - Baseline*

Kernel	Precision	Recall	F-score
Linear	32.8	27.2	23.4
Polynomial 2 degrees	32.4	22.3	19.8
Polynomial 3 degrees	32.5	22.8	20.7
RBF	32.7	28.5	26.3
Sigmoid	30.5	34.5	29.8

Table A.9: *Decision tree - Baseline*

Algorithm	Split criterion	Pruning factor	Precision	Recall	F-score
C4.5	1	0.05	20.2	17.3	17.5
—	—	0.10	19.2	15.4	16.1
—	—	0.15	19.3	15.1	16
—	—	0.25	19.3	15.1	16
—	—	0.5	19.3	15.1	16
—	2	0.05	19.9	16.3	16.8
—	—	0.10	20.2	16.1	16.7
—	—	0.15	20	16	16.6
—	—	0.25	19.5	15.4	16
—	—	0.5	19.5	15.3	16
NBDT			27.8	18.4	19.5

Table A.10: *Nearest Neighbor - Baseline*

Search technique	Distance measure	k	Precision	Recall	F-score
Cover tree	Euclidean	1	26.2	26.4	25.5
—	—	2	25.7	27.3	24
—	—	4	27.7	30.1	26.1
—	—	8	27.6	32.5	27.5
—	—	16	27.1	32.7	26.5
—	—	32	26.1	32.8	25.6
—	—	64	25.7	32.2	24.8
KDTree	Euclidean	1	26.2	26.4	25.5
—	—	2	25.7	27.3	24
—	—	4	27.7	30.1	26.1
—	—	8	27.6	32.5	27.5
—	—	16	27.1	32.7	26.5

Table A.10 – continued from previous page

Search technique	Distance measure	k	Precision	Recall	F-score
—	—	32	26.1	32.8	25.6
—	—	64	25.7	32.2	24.8
Linear	Euclidean	1	26.2	26.4	25.5
—	—	2	25.7	27.3	24
—	—	4	27.7	30.1	26.1
—	—	8	27.6	32.5	27.5
—	—	16	27.1	32.7	26.5
—	—	32	26.1	32.8	25.6
—	—	64	25.7	32.2	24.8
—	Manhattan	1	26.9	28.5	27.1
—	—	2	25.6	28.1	24.5
—	—	4	27.5	30.9	26.6
—	—	8	30.2	34	29.1
—	—	16	28.8	34.3	28.1
—	—	32	27.8	33.9	27.2
—	—	64	27.5	34.1	26.9

Table A.11: *Logistic Regression - Baseline*

Ridge value	Precision	Recall	F-score
10^{-8}	1.7	12.9	3
10^{-7}	1.7	12.9	3
10^{-6}	1.7	12.9	3
10^{-5}	1.7	12.9	3
10^{-4}	6.7	16.6	7.4
10^{-3}	9	22.9	10
10^{-2}	22.7	30.6	20.2
10^{-1}	29.3	36.1	29.2
10^{0}	32.9	37.2	30.6
10^{1}	37	34.6	28.5
10^{2}	40.5	27.5	24.9

Table A.11 – continued from previous page

Ridge value	Precision	Recall	F-score
10^3	36.3	25.3	23.5
10^4	35.9	25.1	23.1
10^5	34	25.1	23.5

Table A.12: *Random Forest - Baseline*

Size of Forest	Size of features at each split	Precision	Recall	F-score
100	6	30.6	33.8	26.6
—	11	31.2	36.3	28.7
—	22	33.5	39	31.6
—	44	34.5	40.1	32.9
—	88	33.2	38.7	32.2
500	6	31.3	35.2	27.2
—	11	30.7	36.9	29
—	22	33.8	40	32.4
—	44	34.9	40.7	33.3
—	88	35.5	40	32.9