IMAGE RETRIEVAL USING VISUAL ATTENTION

by

Liam M. Mayron

A Dissertation Submitted to the Faculty of

The College of Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

Florida Atlantic University

Boca Raton, Florida

May 2008

IMAGE RETRIEVAL USING VISUAL ATTENTION

by

Liam M. Mayron

This dissertation was prepared under the direction of the candidate's dissertation advisor, Dr. Oge Marques, Department of Computer Science and Engineering, and has been approved by the members of his supervisory committee. It was submitted to the faculty of The College of Engineering and Computer Science and was accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

SUPERVISORY COMMITTEE:

Dissertation Advisor

Chairman, Department of Computer Science and Engineering

Dean, College of Engineering and Computer Science

Dean, Graduate College

4-7-08
Date

ACKNOWLEDGMENTS

> *Let the honor of your student be as dear to you as your own, the honor of your colleague as the reverence for your teacher, and the reverence for your teacher as the fear of Heaven.*
>
> Rabbi Elazar ben Shammua, Pirkei Avot

My mentor and dear friend Dr. Oge Marques deserves special thanks. His genuine dedication to learning had an impact on me from the moment this work began. Our many discussions were both academically challenging and enlightening. His advice and support were essential to the successful completion of this research.

The guidance of Dr. Borko Furht, not only during the course of this dissertation, but since the start of my undergraduate studies, has been invaluable. It was his encouragement that first motivated me to pursue this degree, and for that I will always be grateful.

Dr. Hari Kalva provided thoughtful insight as well as resources without which many of the results in this dissertation would not have been possible to obtain. I truly appreciate his help and support.

The feedback provided by Dr. Dragutin Petkovic was critical to the success of this work. I am greatly appreciative of the insights he provided, all of which enhanced my understanding and generated countless new ideas.

Without the company and collaborative efforts of my colleagues the creation of this dissertation would have been a far less enjoyable experience. In particular, I would like to thank Lakis Christodoulou, Chris Holder, and Dr. Daniel Socek at MLAB at Florida Atlantic University. A special thanks goes to Gustavo Borba and Dr. Humberto Gamba at the Federal University of Technology in Parana, Brazil – our research together was always illuminating.

My family and friends, both near and far, provided endless support, motivation, and inspiration. My parents' unconditional love was an unwavering source of strength throughout the writing of this dissertation.

ABSTRACT

Author:	Liam M. Mayron
Title:	Image retrieval using visual attention
Institution:	Florida Atlantic University
Dissertation Advisor:	Dr. Oge Marques
Degree:	Doctor of Philosophy
Year:	2008

The retrieval of digital images is hindered by the semantic gap. The semantic gap is the disparity between a user's high-level interpretation of an image and the information that can be extracted from an image's physical properties. Content-based image retrieval systems are particularly vulnerable to the semantic gap due to their reliance on low-level visual features for describing image content. The semantic gap can be narrowed by including high-level, user-generated information. High-level descriptions of images are more capable of capturing the semantic meaning of image content, but it is not always practical to collect this information. Thus, both content-based and human-generated information is considered in this work.

A content-based method of retrieving images using a computational model of visual attention was proposed, implemented, and evaluated. This work is based on a study of contemporary research in the field of vision science, particularly computational models of bottom-up visual attention. The use of computational models of visual attention to detect salient by design regions of interest in images is investigated. The method is then refined to detect objects of interest in broad image databases that are not necessarily salient by design.

An interface for image retrieval, organization, and annotation that is compatible with the attention-based retrieval method has also been implemented. It incorporates the ability to simultaneously execute querying by image content, keyword, and collaborative filtering. The user is central to the design and evaluation of the system. A game was developed to evaluate the entire system, which includes the user, the user interface, and retrieval methods.

This dissertation is dedicated to the memory of my grandfather

Harry Z. Klinger

His passion for knowledge will always be an inspiration

CONTENTS

Appendix

TABLES

FIGURES

Chapter 1

INTRODUCTION

All life is only a set of pictures in the brain, among which there is no difference betwixt those born of real things and those born of inward dreamings, and no cause to value the one above the other.

H.P. Lovecraft, author, 1890 – 1937

1.1 Overview and motivation

The foremost challenge when constructing an image retrieval system is the *semantic gap*. It is defined as "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" [132]. A content-based description of an image can differ considerably from a human user's description, just as two people can provide different descriptions of the same scene.

This work is concerned with *content-based image retrieval* (CBIR). CBIR systems create machine-interpretable descriptions of an image's physical characteristics. These descriptions, known as *extracted features*, can then be compared using a measure of *similarity*. The similarity between a given *query* image and every image in the *image archive* is then computed by CBIR system. *Results* are then displayed in order of decreasing similarity.

This research was inspired by emerging work in the computational modeling of bottom-up visual attention. The human visual system (HVS) is able to rapidly resolve many complicated image analysis issues, including the semantic gap. Central to the HVS is visual attention, consisting of bottom-up and top-down components. Visual attention is our mechanism for serially selecting the most relevant points to sample from a massively parallel input source (the eyes). It is necessary because the human brain does not have the bandwidth to process the entire visual stimulus at once. Bottom-up visual attention responds to the *saliency* of objects in the scene. For example, a bright, flashing light is more likely to attract bottom-up attention due physical characteristics such as its color, intensity, and motion. Bottom-up attention is involuntary and instinctual. It occurs during the initial phases of gaze allocation, before top-down attention can influence vision. In contrast with bottom-up attention, top-down attention relies on knowledge, memory, and interpretation to drive the vision process.

This work is among the first to claim that visual attention can be used for content-based image retrieval. To test the hypothesis a *proof of concept* was proposed and evaluated. The successful proof of concept results led to the development of a *new method* extending the proof of concept in its use of visual attention and the rigorousness of its evaluation. It has been shown in the literature that people search for *objects* in images, and that it is not only desirable, but necessary to include this functionality in image retrieval systems [39, 7]. This work uses computational models of visual attention to detect salient by design *regions of interest* in images. Image features for content-based retrieval are extracted solely from the detected regions. This method is then refined to detect *objects of interest* in broad image databases that are not necessarily salient by design.

The semantic gap can also be narrowed by using alternate descriptions of images, such as *keyword annotation* [132]. Searching by keywords is a popular, proven method of information retrieval (exemplified by the success of Web-based search engines). The pervasiveness of keyword searching in daily life results in (and is caused by) it being an intuitive method of specifying search parameters. When searching for images using keywords, the textual descriptions of images are generated by humans for the images in the archive. This is both costly and vulnerable to (often unintentional) bias – users may have different ways of describing the same image.

Another way to diminish the semantic gap is to employ *collaborative filtering*. Collaborative filtering relates images based on the past actions of multiple users. Collaborative filtering does not make any attempt to interpret the content of images. Instead, human users create relationships between images. When two images are associated by a user, the likelihood that they are related increases. Once a sufficient amount of associations have been collected, they can be used as the basis

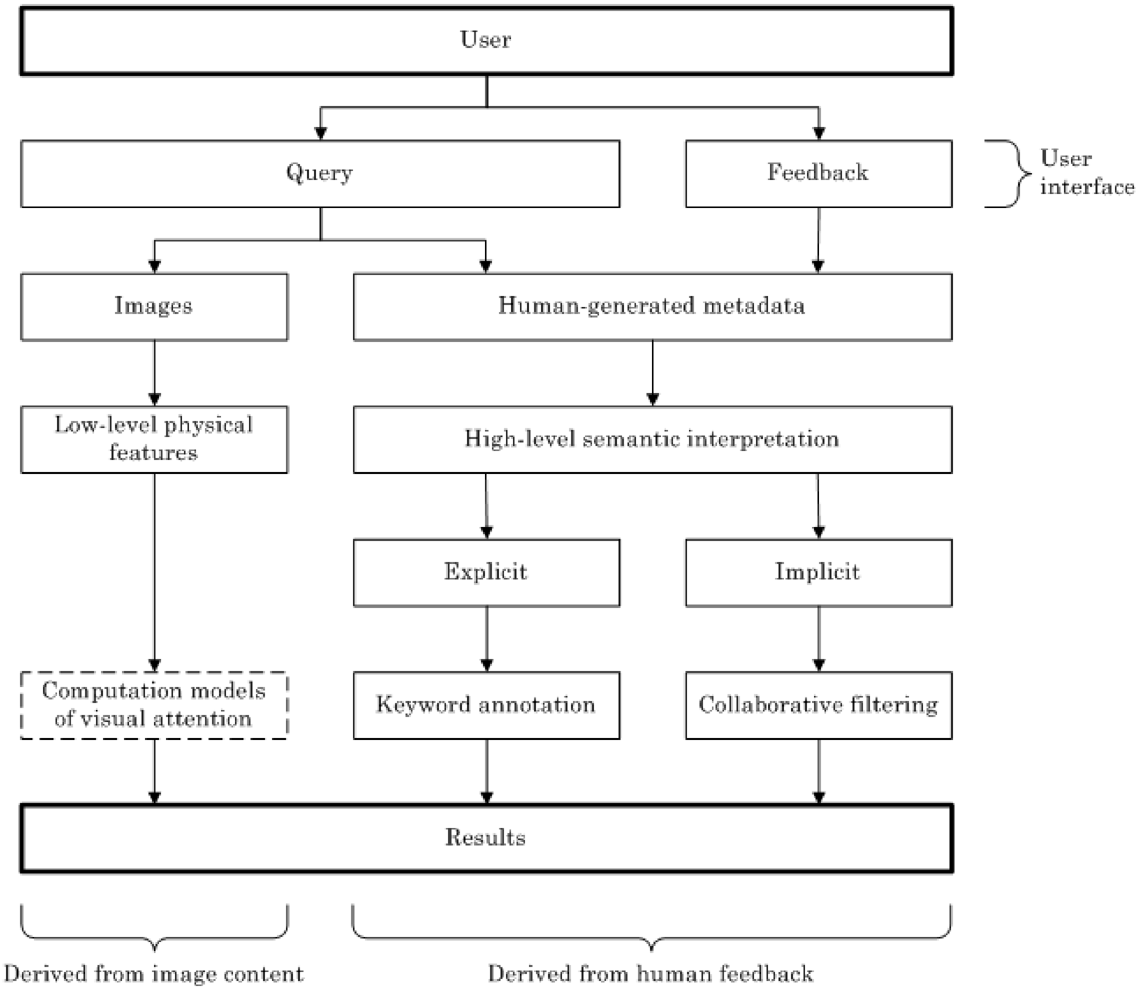

User
Query
Feedback
User interface
Images
Human-generated metadata
Low-level physical features
High-level semantic interpretation
Explicit
Implicit
Computation models of visual attention
Keyword annotation
Collaborative filtering
Results
Derived from image content
Derived from human feedback

Figure 1.1: High-level representation of the integration of retrieval by image content, keyword annotation, and collaborative filtering

of future retrieval. An example of collaborative filtering are recommendation systems employed by online stores. When someone purchases more than one item, each of those items is associated with every other item the user has purchased. When the history of many users is taken in aggregate, it is possible to infer which items are likely to be purchased together and make recommendations to future users.

The integration of content-based, keyword annotation, and collaborative filtering together in the image retrieval system described in this dissertation is illustrated in Figure 1.1. The user specifies the query and judges the quality of the results. The user composes their query and provides feedback through a user interface (consisting of the *Query* and *Feedback* blocks in Figure 1.1). The content-based track (the left side of Figure 1.1) relies solely on content-based data (i.e. metrics derived from low-level visual features). The other track is based on human-generated metadata. This may either be explicitly stated by the user (in the case of keyword annotation), or implicit (as with collaborative filtering). Searching based on keyword annotation and retrieving images using collaborative filters occurs entirely without knowledge of the image content itself, only relying on previous human actions.

The user interface is a critical, yet often overlooked, component of image retrieval systems [7]. This work proposes a new, human-centered user interface for image retrieval, organization, and annotation. This interface allows the simultaneous composition of content-based, keyword-based, and collaborative queries. The user is central to the design and evaluation of the system. A game was developed to evaluate the entire system, which includes the user, the user interface, and retrieval methods.

1.2 Literature review

The semantic gap – the disparity between a user's high-level interpretation of an image and the information that can be extracted from an image's low-level physical properties – has been extensively discussed in the literature [132, 28, 30, 15, 50, 145] and remains an open problem.

A review of vision science and its disciplines is presented in [105]. Among the most significant and relevant early work is that of Noton and Stark on saccades and scanpathts [99]. Oliva discussed the gist of a scene – the interpretation of an image without attention – in [100]. However, Pylyshyn showed that attention is a prerequisite for most vision tasks [116]. Styles presents a discussion on top-down and bottom-up visual attention in [139].

Computational models of human visual attention have proven useful in a variety of applications. These applications include object detection [95, 150, 120], task-motivated analysis [96], face detection [128], compression of multimedia data [26, 55, 56], gaze prediction [29, 107, 108, 112], and image retrieval [87, 86]. This work uses the Itti-Koch model of visual attention [61] and the Stentiford model of visual attention [134].

Smeulders et al. provide an overview of content-based image retrieval at the turn of the century [132]. The Query by Image and Video Content (QBIC) system developed by IBM [38] is an early, significant CBIR system. An object-based approach to image retrieval was presented by Tao and Grosky in [142] and refined in [141]. Carson et al. developed “Blobworld”, a system that searches image segments [7]. Ma et al. proposed “NeTra” in [79]. Like Blobworld, NeTra also extracts region-based features. Contemporary commercial applications of content-based image retrieval include shopping [118], content filtering [147] and content rights management [53].

The ESP Game [148] and Peekaboom [149] have demonstrated the potential of using games to motivate humans to manually annotate large amounts of images, bypassing content-based image retrieval.

Another alternative to content-based image retrieval is collaborative filtering. Kanade and Uchihashi [65, 146] proposed *content-free image retrieval.* Content-free image retrieval is the application of collaborative filtering to images. Users group images based on their perceived similarity. This approach was extended with a Bayesian framework in [75].

1.3 Contributions

The key contributions of this work are:

- **Design, implementation, and evaluation of an attention-driven method to extract regions of interest from images containing objects that are salient by design**: this objective was realized as a *proof of concept* design employing a database where each image specifically contains one or more *salient by design* regions of interest. This contribution appears in Chapter 3.

- **Design, implementation, and evaluation of an attention-driven method to detect objects of interest in broad image databases**: the proof of concept design was extended using a *new method* for detecting objects of interest. Instead of using only the saliency map, individual points of attention are considered. The size and scope of the image database used for evaluation was expanded. Chapter 4 presents this work.

- **Design and implementation of an image organization and retrieval system incorporating visual features, keywords, and collaborative filtering**: this system, referred to as the *Perceptually-Relevant Image Search Machine (PRISM)*, is presented in Chapter 5. PRISM includes image search, organization, and annotation capabilities. It incorporates the ability to compose queries by image content, keyword, and collaborative filtering simultaneously.

- **Development of a new method for evaluating image organization, annotation, and retrieval systems using a game metaphor**: a new method for evaluating image retrieval system using a *game* is proposed in Chapter 6. Presenting evaluation as a game allows the user to be included in the evaluation of image retrieval tasks.

- **Study of recent advances in image retrieval**: contemporary work in image retrieval, including recent CBIR implementations, Web-based image search engines, “Web 2.0”-style applications, and alternative hardware interfaces, is analyzed and summarized. This study appears in Section 2.4.

- **Study of established, relevant work in cognitive science, concentrating on visual attention, with applications for image retrieval**: the literature in the field of cognitive science is vast. The topic of visual attention was given particular focus, centering on the application of visual attention for computer vision in general and image retrieval in specific. Section 2.2 presents this research.

- **Survey of image databases for object-centered image retrieval**: the selection of an image database is central to the evaluation of an image retrieval system. A taxonomy for evaluating such databases was created. This taxonomy was used to evaluate a wide variety of image databases. This survey appears in Section 4.4.

1.4 Organization

This remaining chapters of this dissertation are structured as follows:

- **Chapter 2**: background information and related work on vision science, computational models of visual attention, content-based image retrieval, text retrieval, and collaborative filtering

- **Chapter 3**: a proof of concept method demonstrating the detection of regions of interest using saliency
- **Chapter 4**: a new method that extends the proof of concept with the objective of detecting objects of interest using the entire computational model of visual attention
- **Chapter 5**: a new interface for image retrieval, organization, and annotation
- **Chapter 6**: a method of evaluating image retrieval systems using a game
- **Chapter 7**: conclusion and future work
- **Appendix A**: implementation details of PRISM and the PRISM Game

Chapter 2

BACKGROUND

A photograph is a secret about a secret. The more it tells you the less you know.

Diane Arbus, photographer, 1923 – 1971

2.1 Introduction

This chapter presents background knowledge that is referred to throughout the remainder of this dissertation.

An overview of relevant topics from the field of vision science is presented in Section 2.2. Vision science is the study of the human visual system and the associated neurological processes. The focus of this discussion is on *attention* – how the human visual system selects what portion of a scene to look at.

Computational models of visual attention are discussed in Section 2.3, with emphasis on the Itti-Koch model (Section 2.3.1) and the Stentiford model (Section 2.3.2).

Section 2.4 surveys the most relevant CBIR implementations. The requirements and the design of a CBIR system are presented. Methods of feature extraction and similarity measurement are defined. Background information as well as design details of text retrieval systems are presented in Section 2.5. Section 2.6 presents collaborative filtering, its requirements, benefits, challenges, and design considerations. Content-based image retrieval, retrieval by keywords (text), and collaborative filtering are combined in the PRISM system described in Chapter 5.

2.2 Vision science

Vision science [105] is the study of how humans see and interpret the light that lands on the sensor known as the retina. The following are the key research topics in vision science that relate to CBIR [85]:

- **Attention**: attention is concerned with how the HVS prioritizes and selects what regions of a scene it attends to
- **Perception**: perception is the interpretation of sampled visual information
- **Memory**: the topic of memory encompasses access to past knowledge, rules, and intuition, as well as the recording of current imagery
- **Contextual effects**: the scene an object appears in may greatly affect our interpretation of both the object and the scene
- **Function, category, language, and semantic meaning**: the HVS is influenced by our language and surrounding culture (e.g. our ability to identify

certain colors correlates to a name for a particular color appearing in our language [67])

Attention and perception are the two key topics of particular relevance to this study.

It is not possible for the human visual system (HVS) to consider an entire image at once. Rather, we rapidly select several points-of-attention to direct our vision at when presented with a new scene. As a result, most of the light that radiates and falls upon our retinas is ignored. As with other senses, our brain acts as a filter that reduces the stimuli we perceive at any one time. We can tune in to a single voice in a crowd or ignore the sensation our clothes make against our skin. Similarly, unless we specifically pay attention to certain elements in a visual scene, only those areas of a scene that are salient or relative to the active visual search task will be attended to. In order to accomplish this, our eyes make a rapid series of movements known as *saccades*, the aggregate of which are known as *scanpaths* [99]. This ability to prioritize our attention is not only a matter of efficiency, but critical to survival.

Attention can either be bottom-up or top-down (Figure 2.1). While each is well-defined, there remains a gray area in that there are cases where we are not sure if top-down or bottom-up factors are responsible for attention, nor do we know with certainty how the two interact. Bottom-up attention is rapid and involuntary – it is an instinct. In general, bottom-up processing is motivated by the stimulus

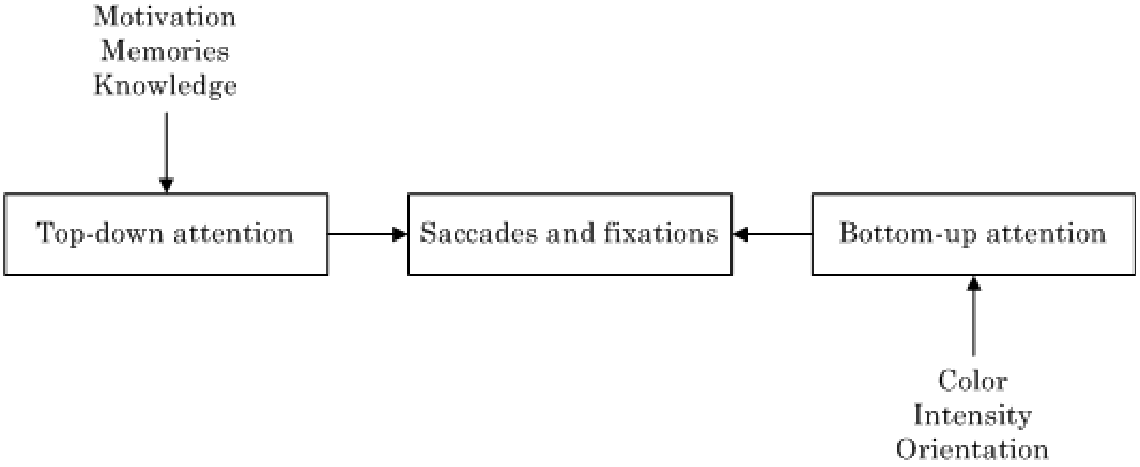


Figure 2.1: Factors that influence attention

presented to the HVS [139]. Our immediate reaction to a fast movement, bright color, or shiny surface is performed subconsciously and automatically. Features of a scene that influence where our bottom-up visual attention is directed are the first to be considered by the brain and include color, movement, and orientation, among others [61]. For example, we impulsively shift our attention to a bright, flashing light. This salient point, if it determined to be relevant to the task at hand, may have more sophisticated processing resources devoted to it – a top-down process. Top-down attention is influenced by knowledge – what we have learned and can recall. Top-down processing is initiated by memories and past experience [139]. Looking for a specific letter on a keyboard or the face of a friend in a crowd are tasks that rely on learned, top-down knowledge. Ultimately, both bottom-up and top-down factors contribute to how we choose to focus our attention, although salient components of a scene influence bottom-up attention *before* top-down knowledge does [14].

Once the HVS selects what merits further inspection, perceptual abilities must interpret the stimuli. Perception is the processing of these senses [139]. Perception occurs in a variety of specialized areas in the brain.

A key challenge of computer vision (and vision in general) is that a single stimulus may have multiple interpretations. In humans and in most computer systems the stimulus consists of one or more two-dimensional projections of a three-dimensional world. Naturally, there is considerable information that is lost in this translation.

In humans, rapid recognition and interpretation of a scene depends on context. Indeed, determining the context, also known as the *gist of a scene* can occur even without attention [100]. Still, in most visual processing tasks attention is needed prerequisite for perceptual processing. Once context is determined our memories and acquired rules (knowledge) lead to expectations of the visual environment [116]. These expectations can be extremely powerful in eliminating potential interpretations of a scene and can even be difficult to overcome despite overwhelming evidence indicating a different interpretation is valid. There are two notable ways this can occur:

- **Priming**: Palmer [105] demonstrates that humans are more successful in identifying an object if it is preceded by relevant information. In this particular example, a mailbox and loaf of bread both are drawn to appear very similar in appearance (perhaps even identically). When primed with an image of a kitchen a loaf of bread is identified. When primed with an outdoor scene the same figure is interpreted as a mailbox.

- **Expected spatial location**: Biederman's hydrant [3] is a classic experiment in which he demonstrates the difficulty people have in identifying objects if they do not occur at the expected position. In this case a fire hydrant is drawn floating in the air rather than fixed to the ground. He shows that participants take notably longer to identify the oddly located hydrant.

Vision science is a diverse field covering many topics. Attention is particularly relevant to this work due to its purpose of narrowing the amount of sampled information to the most salient regions in the field of view. The computational modeling of visual attention is described next (Section 2.3).

2.3 Computational models of visual attention

2.3.1 The Itti-Koch model

Itti, Koch, and Niebur [61] presented "a model of saliency-based visual attention for rapid scene analysis". It is a *biologically plausible* model – its creation was based on a study of actual neurological processes in primates (this can be contrasted with *biologically inspired* models which are only loosely based on natural processes).

The model is depicted in Figure 2.2. The input is a single image. There are many intermediate outputs along the way, but the key result is the generation of

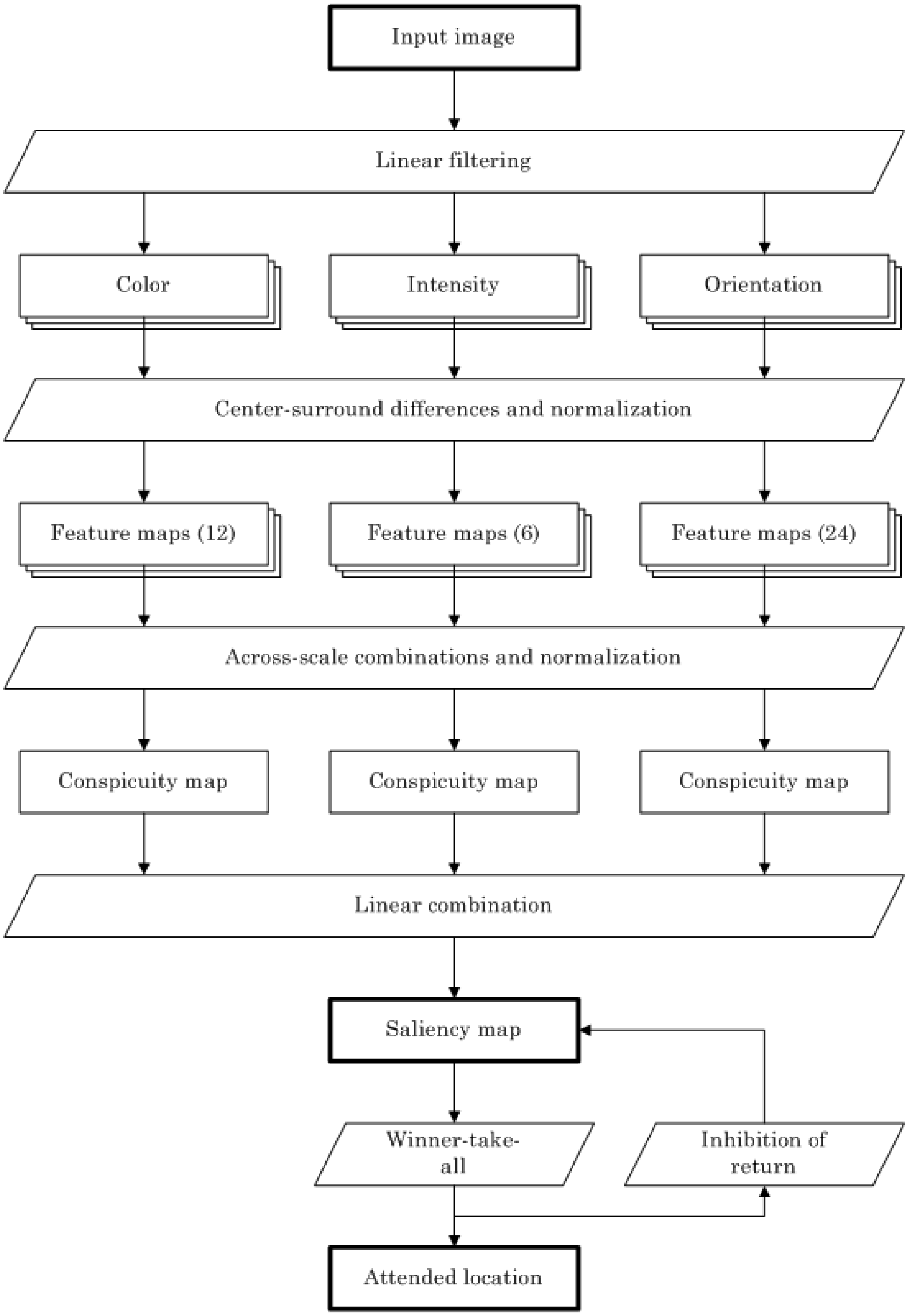


Figure 2.2: The Itti-Koch computational model of visual attention (adapted from [61])

a saliency map from which attended locations can be determined. The description that follows is based on [61].

First, linear filters are applied to the input image. The input image is sampled at multiple spatial scales using Gaussian pyramids which correspond to a variety of resolutions ranging from the original size to 1/256 of the original size.

While the model was subsequently extended to incorporate a variety of features (e.g. motion [152]), [61] uses three: color, intensity, and orientation. Feature extraction is performed by center-surround operations [64] akin to those of the neurons in the human vision system.

Color maps model color opponency in the human cortex. The are generated for red-green/green-red double opponency as well as blue-yellow/yellow-blue double opponency. In all, twelve center-surround feature maps are generated for color.

The intensity feature maps are the normalized average of red, green, and blue color components of the image. Six feature maps are generated by the center-surround differences.

Orientation maps are generated at each scale for four orientations (0, 45, 90, and 135 degrees). In total, 24 feature maps are generated for orientation.

The key feature of this model is its combination of all intermediate feature maps. This is done by giving more significance to maps which contain fewer pronounced peaks and vice versa. A normalization operator is used to accomplish this. This operator multiplies the map by the difference between its global maximum and

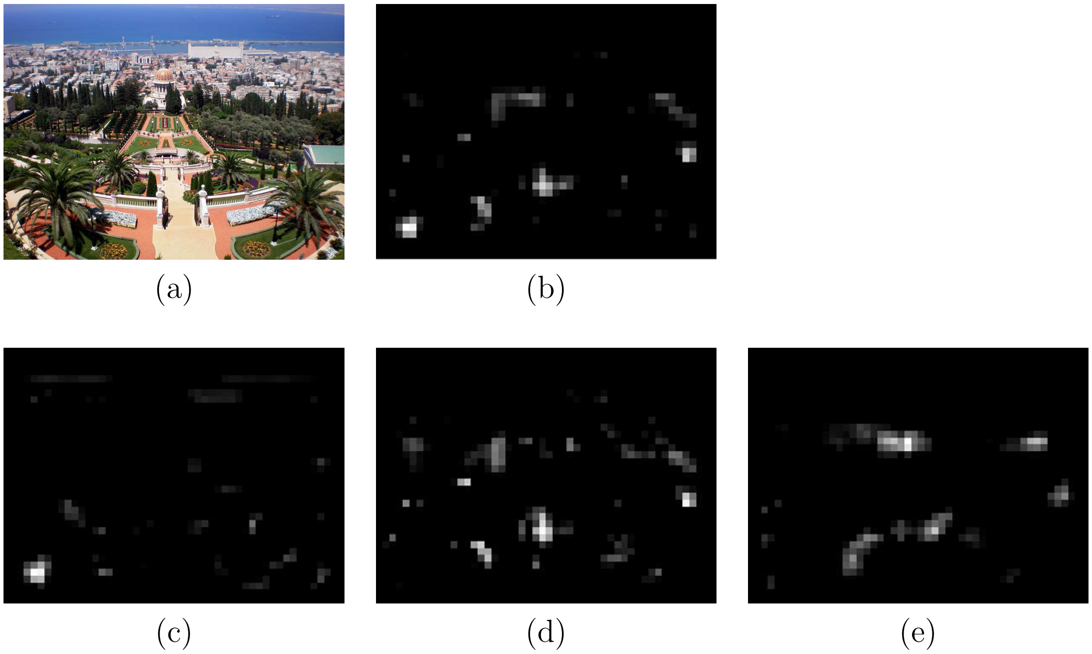


Figure 2.3: The conspicuity maps and saliency map generated by the Itti-Koch model of visual attention. The original image is shown in (a). The saliency map is (b). The color, intensity, and orientation conspicuity maps appear in (c), (d), and (e), respectively

the average of all local maxima.

The normalized feature maps are combined into three conspicuity maps, one each for color (Figure 2.3 (c)), intensity (Figure 2.3 (d)), and orientation (Figure 2.3 (e)). These conspicuity maps are generated at the medium scale of the Gaussian pyramids and are thus several times smaller than the original image. Each conspicuity map is created by resizing each feature map to the same medium scale and adding the feature maps together.

To create the saliency map (Figure 2.3 (b)) the three conspicuity maps are normalized and added together (each contributes an equal third to the saliency map). The remainder of the model interprets the grayscale saliency map in the following way: the greater a given pixel's value, the more salient the location is. The maximum value of the saliency map is the most salient location in the image. A winner-take-all (WTA) feed-forward neural network maps each pixel in the saliency map to a neuron. A voltage builds at each neuron to the point where a neuron in the network "fires" (selects a salient location). Three actions occur when the network selects a salient location: (1) attention shifts to this new point, (2) the neural network is reset (global), and (3), the local, selected area is suppressed.

Suppressing the selected point of attention and the surrounding area is referred to as the *inhibition of return* (IOR). IOR ensures that attention may shift from point to point in the image. If the attended location was not suppressed attention would not be able to shift to subsequent points. The suppressed area gradually recovers over time. IOR can be thought of as "stamping out" the attended location. IOR continues in a loop over time, decreasing saliency at attended areas of the saliency map with each iteration. Thus, there is an integral time-based component to the saliency map, just as there is with human visual attention. More details on the inner workings of this model of visual attention can be found in [54, 58, 59].

The model was applied to visual search in [58], where the authors tested the model's ability to search for specific salient targets within a scene. The calculated times to search for objects in these images (photographs of outdoor scenes with military vehicles) were compared to the actual recorded times of human subjects searching for the same target objects. The model outperformed the human in 75% of the cases [58].

Object detection and recognition is can also be achieved using this model. In [95] the saliency map is biased by the characteristics of the target object. This biasing is in the form of target masks of the desired objects which are translated into weighing coefficients. Each fixation is compared to a hierarchy of known objects for recognition. This approach is also employed in [150]. Similarly, attention can also be modulated by task [96]. Face detection has also been implemented [128]. Rutishauser et al. [120] employ the Itti-Koch model to extract regions and recognize objects by examining the area around the most salient patch of an image and then using region-growing techniques. Key points extracted from the detected object are used for object recognition. Repeating this process after the IOR has occurred enables the recognition of multiple objects in a single image.

Image and video compression is another application of this model. In [26] points of attention generated by the model are used to foveate multiple points in a video to improve MPEG compression. The points from the model are used to determine the regions of interest where the quality of the video must be preserved. Because compression occurs in portions of the video in which people are not likely to devote attention to, the artifacts were not noticed [26]. In [55] and [56] Itti et al. extend and validate this approach against data collected from human subjects using an eye tracker.

The ability of the Itti-Koch saliency model to predict human attention and gaze behavior has been analyzed [29, 107, 108, 112]. The Itti-Koch model is not free of criticism. It is not difficult to find cases where the Itti-Koch model does not produce results that are consistent with actual fixations. Henderson et al. [46] document one such instance where the Itti-Koch saliency map is not congruent with the human eye saccades. However, [46] adds the constraint that the visual task being measured is active search, not free viewing. The Itti-Koch model was not initially designed to include the top-down component that active search and similar tasks require.

The experiments in this dissertation apply the Itti-Koch computational model of visual attention to specific problems in image retrieval. In Chapter 3 an image retrieval system based on clustering salient regions of interest is implemented. At the center of this system is the saliency map. In this case the saliency map is used to generate cues of the most salient regions in images in combination with the model of visual attention described in Section 2.3.2. Features were extracted only from these regions and used to cluster the images in the image archive. This approach is then improved and expanded in Chapter 4.

2.3.2 The Stentiford model

Stentiford proposed a model of visual attention in [134]. This model was refined for content-based image retrieval in [135]. The model is referred to as *the Stentiford model of visual attention* in this dissertation. It is a biologically-inspired model, not biologically-plausible and, as such, does not adhere strictly to processes of the HVS. The Stentiford model produces a *visual attention* (VA) map.

The Stentiford model has been applied to a variety of applications. In [5] it was used to detect regions of interest for JPEG2000, preserving detail in the selected regions while applying greater compressing to the remainder of the image area. It was also applied to image compression in [136], similarly using the VA map to preserve certain parts of the image while aggressively compressing the rest. The model was also compared to the behavior of the human eye using separate eye tracker hardware in [103] (this work was extended toward image retrieval using an eye tracker without the model of visual attention in [104]). The model was used as a similarity measure, comparing images by the difference in their structure in [9] and expanded in [137].

The Stentiford model functions by suppressing areas of the image exhibiting patterns that are repeated elsewhere in the image. Thus, unique areas of the image will be the identified at the end of the VA map generation process (these areas will be the least-suppressed regions of the image). This results in flat, relatively homogeneous regions receiving low VA scores and active, varying regions receiving high scores. The VA map is generated by selecting a random neighborhood of pixels and comparing its similarity of other regions in the image. An example of matching neighborhoods is shown in Figure 2.4. In this figure a simple 16 × 16 pixel binary image is shown. Set pixels are gray, cleared pixels are white. The randomly-generated regions of pixels are outlined in black. The regions were initially generated around a pixel near the top-left of the image (the pixel filled with the diagonal lines in Figure 2.4). It is being compared to a region in the right side of the image in Figure 2.4. This comparison reveals an exact match, not affecting the VA score of the source pixel. Ultimately, each pixel in the image is assigned a VA score. First, a random pattern of pixels (the pixel neighborhood to be sampled) is generated around each pixel. The generated neighborhood is compared to others randomly-selected ones in the image. The degree of mismatch between the regions is computed. For example, identical neighborhoods will have no mismatch and will not modify the VA score of that pixel, while different neighborhoods will raise the VA score. In the end, the regions with the highest scores are those with the least similarity to the rest of the image. For a detailed explanation please refer to [2].

Although the Stentiford VA map is similar in purpose to the Itti-Koch saliency map, its use and interpretation are inherently different. Most significantly, there is no time component to the VA map, whereas time is essential to the understanding of the saliency map of Itti-Koch. This limits the VA map to spatial analysis, offering little discrimination between multiple highly-scored regions of an image (i.e. if two

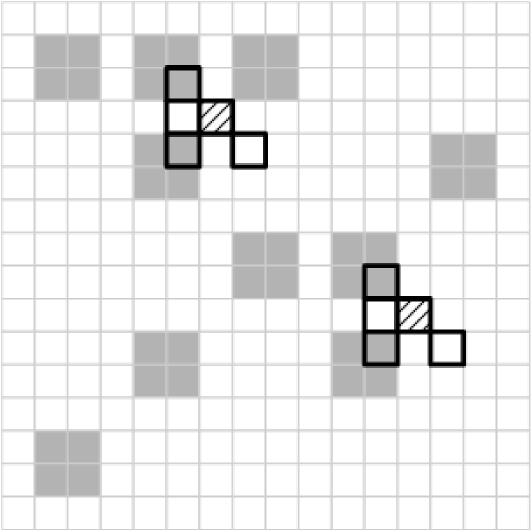
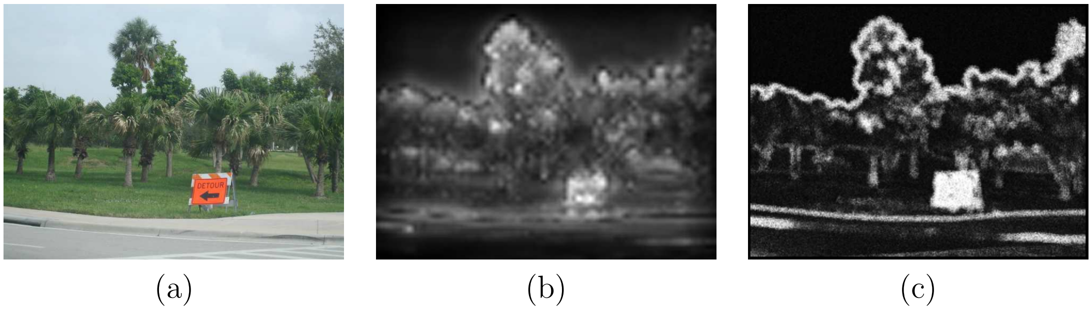


Figure 2.4: Matching pixel neighborhoods in the Stentiford Model (adapted from [135]). Each box indicates a pixel. The outlined pixels are those in the randomly-generated comparison region

regions in the VA map are both awarded high scores the model does not incorporate an mechanism such as Itti-Koch's winner-take-all neural network to absolutely select one location over another). Furthermore, the VA map can only be used on static images, not video (unlike the Itti-Koch model).

However, the Stentiford VA map is useful for segmenting regions. Unlike color-based segmentation algorithms, which tend to differentiate between heterogenous regions, the VA map excels at keeping closely-located heterogeneously-colored regions together. The Itti-Koch map does not detect regions, but single salient points of fixation, which leaves the determination of the attended region beyond the fixation to later processing. Figure 2.5 shows an image (Figure 2.5 (a)), its saliency



(a) (b) (c)

Figure 2.5: Comparison between the Itti-Koch saliency map and Stentiford visual attention map. The original image appears in (a). The saliency map is shown in (b). The visual attention map is (c).

map (Figure 2.5 (b)), and its visual attention map (Figure 2.5 (c)). Note that although the target object (the orange “detour” sign) is highlighted in each map, it appears more consistent in the visual attention map. However, the visual attention map also gives the same significance to many other areas of the image (including the border between the top of the trees and the sky as well as as the road) which are less prominent in the saliency map.

2.3.3 Other models

Rybak et al. [121] proposed a computational model of visual perception and recognition that is led by attention. In their work they represent images using the computed scanpaths. It is demonstrated that this scanpath can be used to recognize images invariantly.

Draper et al. have shown that even a simple implementation of visual attention (in this case, detecting corners) can yield useful results [27]. Their work models the expert object recognition pathway – the part of the brain that recognizes specific object. Attention is used to feed data points to this pathway. Ultimately, this results in hierarchical categories of objects.

2.4 Content-based image retrieval

2.4.1 Overview

Content-based image retrieval (CBIR) is a technique for retrieving digital images based on featured extracted from low-level physical properties. It is a field that has has begun to produce a wide variety of real-world applications. These applications range from shopping [118] to content filtering [147] and content rights management [53] (with many other novel CBIR implementations remaining in research labs). The prospect of an intuitive, effective CBIR system is a worthwhile objective which promises to become a useful tool in a variety of domains, from organizing personal photos to security applications.

A CBIR system is a specific type of visual information retrieval (VIR) system. Chang et al. [8] provide criteria for the classification of VIR systems. A subset of these criteria relevant to this work follows:

- **Automation**: typically, low-level features are extracted without human intervention, although semi-automatic systems also exist.

- **Adaptability**: an adaptable system will have the ability of modify the features extracted from archive content depending on user requirements.

- **Abstraction**: there are a variety of levels visual data can be indexed at (e.g. feature-, object-, syntax-, semantic-level, etc.). The higher the level of abstraction, the closer the system is to our own natural language.

- **Generality**: a system designed for a narrow domain may have improved retrieval efficiency within that domain but not necessarily similar performance if the database is made more generic.

- **Content**: the content of the multimedia archive used by a VIR system may either be static or dynamic. Handling dynamic multimedia databases is more complex (retrieval algorithms must be modified in order to account for the changing nature of the content).

- **Categorization**: several visual information retrieval systems rely solely on high-level, semantically-meaningful categories for retrieval. Others mix categorization with other retrieval methods (e.g. allowing the user to select a high-level category and then restricting results only to members of that category). Yet other systems to not use any form of categorization.

2.4.2 Related work

IBM's Query by Image and Video Content system, QBIC [38], is one of the earliest landmark CBIR systems. It allows the user to specify their desired query and retrieve similar images. Queries can be composed by directly specifying image features or by providing example images. Being among the first implementations, it is widely cited as an example CBIR system. CBIR has expanded in many directions since the introduction of QBIC. Advances have been made in the features used to describe the image content, similarity measures employed to compare images, and how users interact with CBIR systems, including user interfaces and feedback mechanisms. The reader is referred to [132] for a summary of the state of the art in CBIR prior to 2000.

In this work we emphasize CBIR systems that make use of image regions, rather than global descriptors, for retrieval. An object-based approach to image retrieval was presented in [142]. The system proposed in [142] uses a web-based interface in order to allow users to query using specific objects. First a user provides the URL of an image of their choosing. The system then allows the user to select regions within the image for querying. The system, one of the first published attempts to query by regions within images, admitted the challenge of effectively extracting features from individual objects within images. The method of feature extraction of this work was refined in [141].

In [7] a method to convert raw pixel data in images to "blobs", or color- and texture-consistent regions is presented. The authors argue that, as images themselves are composed of objects, segmenting the image into regions is more meaningful than a global representation and query. However, they reason that, due to the difficulty of obtaining good segmentation, many CBIR systems used global descriptors, preferring to have less representative descriptors instead of poorly-defined regions (which would also result in inaccurate image descriptions). Thus, there is a barrier that must be overcome to entry before a CBIR system can utilize regions. This barrier is an effective and reliable segmentation algorithm. The implementation presented in [7] uses Expectation-Maximization [22] to cluster pixels based on their color, texture, and position. The computed regions are presented to the user, who can then select regions to be the basis of the query. Carson et al. identify two limitations of image retrieval systems [7]:

- Users are looking for objects within images but image retrieval systems represent only low-level features and disregard spatial organization.
- Image retrieval systems are not intuitive. Results rarely provide a reason they were returned and query refinement is not always straightforward.

The work of Forsyth et al. [39] is cited by Carson et al. [7] in order to make the case that it is not only desirable, but necessary to identify objects within images. In [39] it is stated that in order to satisfy the wide variety of potential queries, it must be possible to define objects within images, as difficult as the task may be.

Interfaces employed by CBIR systems are critiqued in [7], stating that it is not sufficient to “set a few knobs”, view the results of the query, and adjust the parameters. More specifically, this interface paradigm restricts user control and obscures the features used by the system for retrieval. Blobworld overcomes these restrictions by allowing the user to select and weigh the image regions to be used in the query.

Another region-based CBIR system, NeTra, was proposed by Ma et al. in [79]. As in [7] a method of segmenting images is incorporated. Rather than global features, region-based features based on color, texture, and shape are used for indexing and retrieval. Furthermore, NeTra is web-based, enabling platform-independence and greater accessibility. In concurrence with [39], [79] states that “automated image segmentation is clearly a bottleneck for enhancing the retrieval performance”. In NeTra, segmentation is accomplished by following edge flow [78]. Edge flow is a segmentation method that is able to combine different features, each weighed differently (e.g. color, texture, and shape). Feature selection and weighting is done by the user through the interface in NeTra.

CLUE, proposed by Chen et al. [11], is an image retrieval system that takes a different approach than that of Blobworld or NeTra to providing more semantically-consistent results. It employs the region-based segmentation and feature extraction method described in [10]. The system is novel in that it displays results as clusters of related images rather than as a list of results ranked by decreasing similarity. In this way, potential results are not only similar to the query, but to each other as well. This leads to results that, at the very least, are more semantically consistent (i.e. outliers will be excluded as they do not conform to the rest of the results, despite their similarity to the query).

Zhao and Grosky [157] explicitly considered the meaning of the semantic gap in terms of the design of a CBIR system. The system proposed in [157] is refined in [156]. The work uses Latent Semantic Analysis (LSA), a processing technique usually associated with text, not images, to cluster frequently-occurring image features (please refer to Section 2.5 for more information). It is stated in [157] that metadata can be either content-dependent or content-independent. Content-dependent metadata refers to the traditional features of CBIR systems, building indices solely from the physical properties of the image (e.g. low-level features such as color, texture, and shape). Content-independent metadata is all other data that cannot be automatically derived from the image content, such as the names of objects within images, the relationship between the objects and the scene, or even the location the picture was taken at, or simply a text description of the image. However, four reasons for text descriptions of images alone being insufficient for effective retrieval are provided:

- Text cannot capture perceptual saliency
- Certain entities or events cannot be captured by text alone

- Text cannot correlate perceptual and conceptual features

- Text descriptions are subject to the annotator's own interpretation and, as a result, lead to inconsistencies and ambiguities

The combination of the ESP Game [148] and Peekaboom [149] dispute some of this criticism. The ESP Game relies solely on textual descriptions of images. Inconsistencies and ambiguities are mitigated by finding concurrency in the descriptions provided by a large number of users, which was not accounted for in [157]. Using the data collected by the ESP Game, Peekaboom adds spatial location metadata to the text descriptions. When the entirety of the collected data – text, locations of text labels, and the times of guesses – is taken into consideration it can be seen that perceptual information is expressed.

2.4.3 Components

The architecture of a generic CBIR system is shown in Figure 2.6. It is a refinement the CBIR architecture proposed in [83]. The major components are maintained, but the relationships between these components are modified and explicitly labeled. It consists of the following components:

- **User**: the user is the most important consideration when designing an image retrieval system. Ultimately, if the user is not satisfied, the design for the rest of the system is for naught. In the most general sense, the user is not

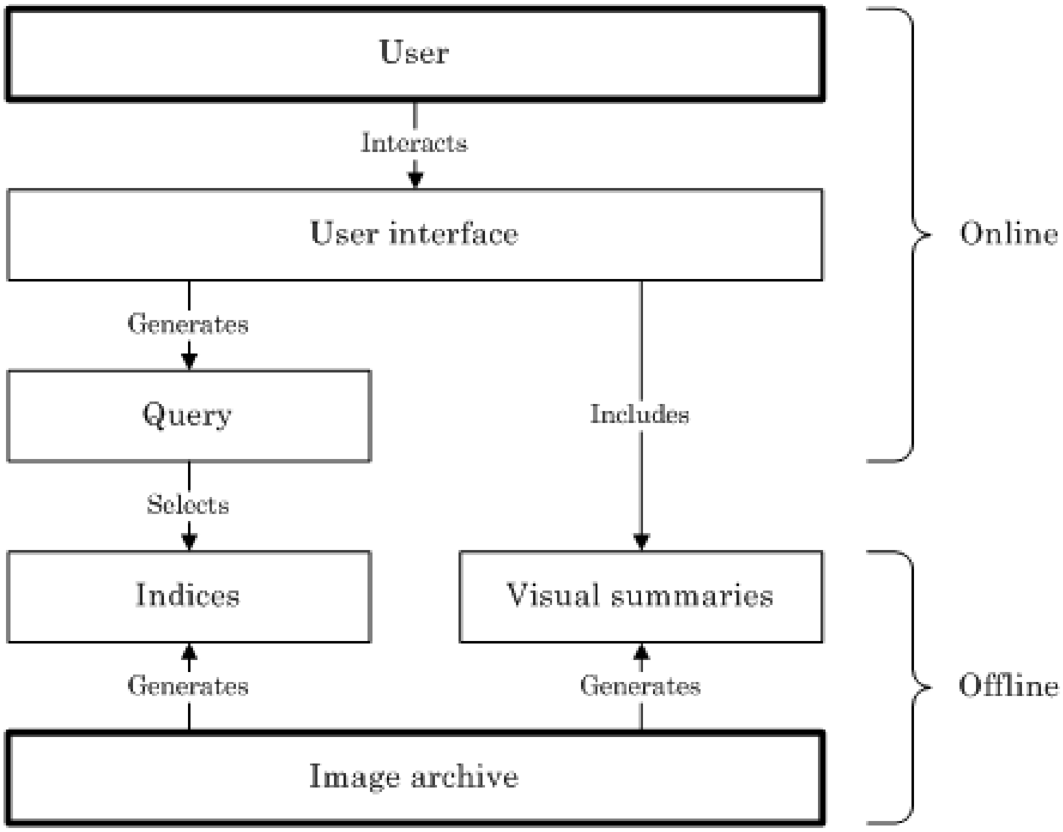


Figure 2.6: General architecture of a CBIR system

necessarily a human user, but may be another computer-based application (e.g. an intelligent agent that queries and makes decisions based on the returned results). However, generally, CBIR research does place a human as the end-user, including this work. The remainder of this dissertation concerns the design of a CBIR system specifically intended for a human user. In a CBIR system the user plays the critical role of composing the query by interacting with the user interface and evaluating the quality of the retrieval results.

- **User interface**: the user interface allows the composition of queries and evaluation of results. A variety of strategies for composing queries may be considered (they are described later in this section). The user interface includes visual summaries of the image archive content. The summaries may be used for the composition of a query, free browsing of the image archive, or representation of query results.

- **Query**: the query is generated by the user through interactions with the user interface in order to select a subset of images from the image archive. The query may be as narrow as asking for the return of a specific image from the archive, or as vague and challenging as "show me pictures of happy people" (see Section 2.4.4 for more details).

- **Indices**: indices are a searchable representation of the image archive. They are generated through a process known as *feature extraction.* Feature extraction algorithmically analyzes the physical content of images and results in a new representation that allows the similarity between images to be expressed.

- **Visual summaries**: traditionally, visual summaries of an image archive are thumbnails – low-resolution version of the images in the image archive. In certain cases visual summaries may include additional information, such as representations derived from feature extraction.

- **Image archive**: the image archive is a collection of digital signals sampled from projections of light onto two-dimensional sensors. The scope of the image archive greatly affects the design of the other components. For example, if the scope is restricted to a narrow field it may be possible to improve the quality of the indices by selecting more effective feature extraction algorithms and design the user interface for the narrower task. On the other hand, a large and diverse image database makes it more difficult to retrieve specific images due to the greater ambiguity and increased number of potential responses to queries.

Furthermore, the components of a CBIR system can exists as either online or offline processes (also indicated in Figure 2.6):

- **Online**: the user, user interface, and query are the minimum components of the online domain of a CBIR system. These three components are the minimum portions of a CBIR system that must be dynamic, changing with each use.

- **Offline**: these components, the indices, visual summaries, and image archive, are the only components of a CBIR system that can be offline and static. In this case, they are computed one time and do not change, regardless of the query or interactions with the user interface.

Note that components denoted as offline do not necessarily need to be so and may be implemented as online instead. For example, the image archive may grow over time, or the indices may change as in response to user actions. However, those components marked as online in Figure 2.6 are required to be implemented as such.

2.4.4 Query

A wide range of paradigms for the design of an interface to query a CBIR system have been proposed. The interface is the gateway to the image archive. The minimal user interface, simply the ability to browse all images in the archive technically satisfies the requirement that an image retrieval system allow the user to retrieve their intended image from the image archive, but is hardly optimal due to the amount of time required to manually browse the archive. Subsequent interfaces either organize images or provide querying capabilities. Interfaces can be characterized as follows:

- **Browsing**: this is the simplest way to access an image archive. As an improvement over a flat browsing structure, images may organized into groups (sometimes by clustering) for the user to peruse, as in CLUE [11].

- **Customized categories**: images are structured as hierarchical, domain-specific categories. One example of an ontology based system is described in [52]. Hierarchies may consists of multiple, semantically-meaningful levels (e.g. a top-level category may be “vehicles” and contain categories such as "airplanes" and "cars", which may themselves include further subclassifications).

- **Query by example image (QBE)**: this is the classical content-based image search paradigm. The user provides a sample image with the intention of having the system retrieve similar images. An example of a system that allows searching by this paradigm is QBIC [38]. There are shortcomings to this approach. Most significantly, query by example image requires that a user acquire a representative image before querying. This image may be from another collection, obtained through means beyond the particular CBIR system. This can be concerning, as the user must use other means to search for images. Alternatively, the example image may be contained within the same image archive used by the CBIR system. In this case, another querying paradigm, such as browsing, must be implemented, or random images may be requested until the users selects and appropriate example.

- **Query by image region**: queries can be based on a user or system-defined subset of an entire region. In order to accomplish this the user must be allowed to manually-define a region of the image, or an unsupervised method of segmentation must be incorporated into the system. Blobworld [7] and NeTra [79] are two systems that allow query by image region. The object of this image paradigm is to improve retrieval results by making the query based on the most relevant portion of the given image.

- **Query by multiple example images (QBME)**: the user can provide several example images to the system, as in [4]. Commonalities between all the query images can be used the basis of the query. Furthermore, each query image can be individually weighed, with the most representative image contributing the most to the query.

- **Query by visual sketch**: several implementations provide drawing tools for the user to create an arbitrary image, including that of Santini and Jain [123] and Retrievr [69]. This is useful in the absence of an example image. The challenge of this approach is that is relies on the artistic abilities of the user, resulting in one of the most demanding query interfaces.

- **Query by direct specification of visual features**: this is this most technical approach, as in Webseek [133]. The user must understand the characteristics and implications of each specified visual feature. This approach can be difficult to use for users unfamiliar with the design and inner workings of the system.

- **Query by keyword**: if images have previously been annotated or if textual contest is available it is possible to search them using text. Google Image Search is a successful example of this method [42]. Google Image Search automatically annotates images by using the surrounding text of the web page on which the image appears. Query by keyword systems can also rely on manual annotation of individual images by humans. This is an effective query method that may operate entirely in the absence of image content, although it is not always feasible to annotate images in a suitable fashion (i.e. large database may take too long to manually annotate).

- **Multimodal query**: multimodal queries are those that combine multiple modalities (e.g. touch, voice prompts, body movements, etc.). recent innovations, such as large-scale multi-touch displays [18] indicate the increasing feasibility of multimodal queries, or those that incorporate multiple forms of input. In [18] multiple users can use all their fingers in a natural way to manipulate and organize images. One example of academic work investigating multimodal interfaces [88], which demonstrates a dynamic, predictive user interface for visual search tasks.

2.4.5 Feature extraction

The selection of a feature or group of features to extract from digital images is one of the most critical decisions in the design of a CBIR system. The selection of an inappropriate feature will directly and adversely affect the results. For example, a color-based feature would not be appropriate for searching an image database where all objects are similarly colored.

While we perceive a particular expression color due to the physical characteristics of the light that radiates onto the rods and cones in our eyes, in digital technology color can be represented in a variety of ways, many of which have significantly different interpretations and applications. For example, certain color spaces may be more invariant to data corruption than others, while others may make changing the represented data simpler. Four different color spaces were inspected for this work: RGB, YCbCr, HSV, and HMMD. For each representation a histogram can be constructed and compared to determine the similarity between images. Color features can be extracted from the global image, or from a particular region of interest using a region mask.

- **RGB**: This is perhaps the most common representation of an image. RGB expresses images as three values – a combination of red, green, and blue values. In digital imaging these values usually range between 0 and 255, although fewer levels of discrimination may be used for compression. This color space forms the basis from which the others in this work are derived. Figure 2.7 shows an image decomposed into its R (red), G (green), and B (blue) components.

- **YCbCr**: The YCbCr represents color image as a combination of luminance (Y) and chrominance (C_b and C_r) components. It was developed to enable color television broadcasts. One benefit of this representation is that both chrominance components can be disregarded by older equipment incapable of displaying color images while still displaying a meaningful image based solely on luminance information. Furthermore, both chrominance components are frequently sampled at a lower rate and interpolated at the time of display, enable a compressed image to be transmitted. Perceptually, it has been demonstrated that these components are less significant in understanding the image than luminance. Figure 2.8 shows an image and its YCbCr components. The following formulas were used to convert from RGB to YCbCr [45]. The constants used to scale R, G, and B in the equations may vary slightly in different implementations.

$$Y = 0.299 \times R + 0.587 \times G + 0.114 \times B \tag{2.1}$$

$$C_b = -0.169 \times R - 0.331 \times G + 0.500 \times B \tag{2.2}$$

$$C_r = 0.500 \times R - 0.419 \times G - 0.081 \times B \tag{2.3}$$

- **HSV**: HSV stands for hue (H), saturation (S), and value (V). It is particularly suited for applications such as digital image editing because each component can be acted upon individually (e.g. an image can be brightened my modifying only the value component, leaving the others unchanged – the equivalent operation in the RGB color space would need to modify the red, green, and blue components). A color in HSV can be visualized as falling within a cone. Hue is a value in degrees, which represents the color as a continuous spectrum. Thus, a value at 0 degrees is near one at 355 degrees. Saturation and value are typically expressed as percentages. A saturation of 100% indicates that the pixel has the maximum amount of "colorfulness", whereas a saturation of 0% for a particular pixel indicates a subdued grayness. Value is similar to the intensity of a pixel, or the amount of light that is radiated from that point. A value of 0% indicates black, regardless of the hue or saturation. Figure 2.9 shows an image and representations of the hue, saturation, and value components. While the hue component does not map well to such a representation, the saturation component shows and value components are quickly understood. The equations used for translating from the RGB color space to the HSV color space are shown below. Note that Max is the maximum of either red, blue, or green color components of a pixel. Similarly, Min is the minimum of these values.

$$Max = maximum(red, green, blue) \tag{2.4}$$

$$Min = minimum(red, green, blue) \tag{2.5}$$

$$Delta = Max - Min \tag{2.6}$$

$$H = 60 \times \begin{cases} \frac{G-B}{Delta}, & \text{Max} = \text{R} \\ 2 + \frac{B-R}{Delta}, & \text{Max} = \text{G} \\ 4 + \frac{R-G}{Delta}, & \text{Max} = \text{B} \end{cases} \tag{2.7}$$

$$S = Delta/Max \tag{2.8}$$

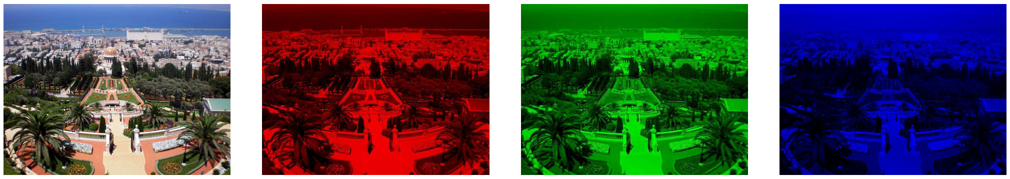
$$V = Max \tag{2.9}$$

- **HMMD**: The HMMD color space is similar to HSV, composed of the hue (H), sum (S), and difference (D). Difference is equivalent to *Delta* (Equation 2.6), while hue (Equation 2.7)is the same as its HSV counterpart. Figure 2.10 shows an image decomposed into the HMMD color space. S is calculated as shown in Equation 2.10.

$$S = \frac{Max + Mim}{2} \tag{2.10}$$

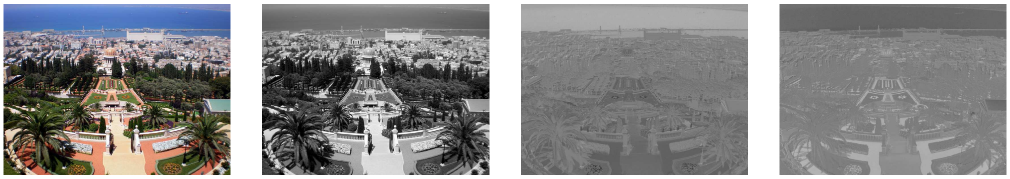
2.4.6 Similarity

A similarity measure is metric which expresses how close or far two n-dimensional feature vectors are. When a query is executed the similarity between



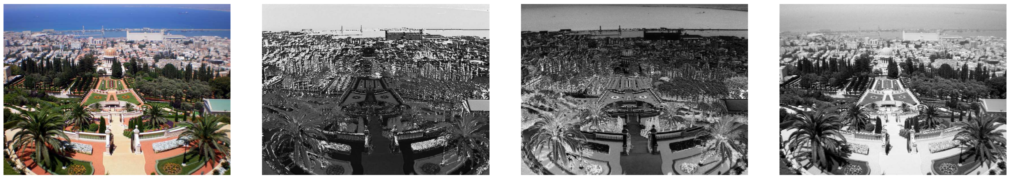
Original R G B

Figure 2.7: An image represented in the RGB color space



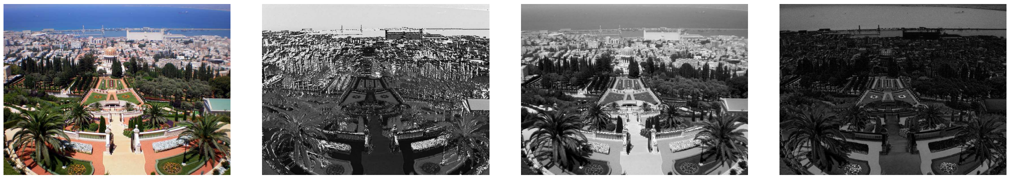
Original Y C_b C_r

Figure 2.8: An image represented in the YCbCr color space



Original H S V

Figure 2.9: An image represented in the HSV color space



Original H S D

Figure 2.10: An image represented in the HSD color space

the query image (or query images) and every image in the database is calculated. The images which are closest (those with the smallest distances from the query) are expected to be better results.

The following list presents several distance measures. The notation for each is the same. Each distance measure compares two vectores, u and v. Both vectors have a dimensionality of n.

- **L1 distance**: L1 distance (also known as *Manhattan distance* and *city block distance*) is the sum of the absolute difference of each point in the vector. The L1 distance between vectors u and v is

$$L1(u,v) = \sum_{i=0}^{n} |u_i - v_i| \tag{2.11}$$

- **L2 distance**: L2 distance is commonly referred to as *Euclidean distance*. In two dimensions the Euclidean distance is equivalent to placing a ruler between two points and recording the measurement. This can be extended to n-dimensions. It is the simplest distance measure. The Euclidean distance between vectors u and v can be expressed as

$$L2(u,v) = \sqrt{\sum_{i=0}^{n} (u_i - v_i)^2} \tag{2.12}$$

- **D1 distance**: The D1 distance measure was proposed in [51]. It is defined as

$$D1(u,v) = \sum_{i=0}^{n} \frac{|u_i - v_i|}{1 + u_i + v_i} \tag{2.13}$$

- **Cosine distance**: Cosine similarity is the angle between two n-dimensional vectors. The smaller the angle, the closer the vectors are. It is a common similarity measure in text document retrieval. Although it is not typically used to compare image features such as color histograms, it is a valid and applicable distance measure to use to compare vectors. Cosine similarity is calculated as follows

$$CosD(u,v) = \frac{u \cdot v}{||u|| * ||v||} \tag{2.14}$$

For clarity, this equivalent representation is also provided

$$CosD(u,v) = \frac{\sum_{i=0}^{n} u_i v_i}{\sqrt{\sum_{i=0}^{n} u_i^2}\sqrt{\sum_{i=0}^{n} v_i^2}} \tag{2.15}$$

- **Histogram intersection distance**: Histogram intersection is a distance measure specifically developed to distinguish between color histograms and, as such, of particular interest [140]. The equation for histogram intersection distance is

$$HistD(u,v) = \sum_{i=0}^{n} min(u_i, v_i) \tag{2.16}$$

The value may be normalized for the number of pixels in the image. It indicates "the number of pixels from the model that have corresponding pixels of the same color in the image" [140]. The objective of the method is to exclude distracting values and only count common colors.

2.5 Text retrieval

Text retrieval (also referred to as document retrieval or text analysis) is a subset of the broader field of information retrieval (of which CBIR also belongs to). It is the interpretation of a corpus – a collection of "documents" – in response to a given query. The query may be a small as a single term. Contemporary text retrieval systems have scaled to massive proportions. Vannevar Bush's classic article "As We May Think" [6] is often given credit for "the idea of access to large amounts of stored knowledge" [130] – the idea of an automated text retrieval system.

Just as it is not feasible to search through thousands of images directly in an image retrieval system, it is not practical to search through an entire corpus (collection of all documents in the database) to fulfill a text query. A text retrieval system must present an alternate representation for its searched documents in order to be an efficient solution. Text retrieval is a mature and deep field. It is covered here to the extent it is necessary to build a system to search for images by keyword annotation.

Text retrieval systems include the following three components:

- **Indices**: inverted indices (terms pointing to the documents containing the terms generated from the source documents themselves) are constructed
- **Text analysis**: due to the nature of text, techniques such as stemming (reducing words to their common root) and collecting synonyms are used
- **Similarity-based ranking**: documents and queries are be represented as vectors and a vector similarity measure such as the cosine coefficient is be used

Results are typically returned as a ranked (ordered) list of documents in decreasing similarity. Document similarity is a combination of certain statistical values that may include [158]:

- The frequency of a term within a document
- The frequency of a term within the query
- The number of documents containing a certain term
- The total number of occurrences of a certain term within all documents in the collection
- The number of documents in the collection
- The number of indexed terms in the collection

Similarly, Fang et al. [35] group retrieval heuristics into four categories: term frequency, term discrimination, length normalization, and term frequency-document length. In general, the objective is to find documents that are more relevant to the query term. A document that mentions the query term more frequently may not be sufficient if it is far longer than other documents. Thus, each term must be individually weighted using the following three heuristics [158]:

- The more a documents a term appears in, the less weight it is given
- The more frequently a term appears within a single document, the more weight it is given
- The more terms a document contains, the less weight it is given

Vector-based methods for text retrieval include *term frequency-inverse document frequency* (TF-IDF) [102]. TF-IDF first was introduced in [122]. IDF was shown as the optimal weighing in the case where each document retrieves itself [106]. Many variations of term weighing schemes exist, but they generally vary in the following three ways [76]:

- **Term frequency**: this is often modified by application-specific constant terms
- **Document frequency**: occasionally, this may also be modified by constant terms

- **Document length normalization**: to avoid biasing results towards long documents, document lengths are normalized

Latent Semantic Analysis (LSA) is often combined with with text retrieval to improve the performance of results by analyzing the semantic relationships between the frequencies of terms in a set of documents [68]. It was first proposed in [20].

A first step in the creation of a text retrieval system (once the system has been specified) is the generation of an inverted index. An inverted index [159] is a list of terms with pointers to the documents they occur in. The index is composed of all terms that occur in all documents. Each entry in the inverted index includes pointers to each document that contains the term, and possibly additional information, such as the page or paragraph number in which the term occurs. Figure 2.11 illustrates the process of searching for the term "building" using an inverted index. Instead of searching the content of each document for the term the system may simply hash to the entry in the list. Such a simple operation scales remarkably well, regardless of the number of documents. Additionally, the inverted index is a far smaller data structure than the corpus and can be stored in main memory [1]. Topics such as tokenization and parsing are not needed to implement such a system and, as such, are beyond the scope of this work.

To search for a single term using an inverted index simply hash to the entry and retrieve the documents that are pointed to by that entry. To search using

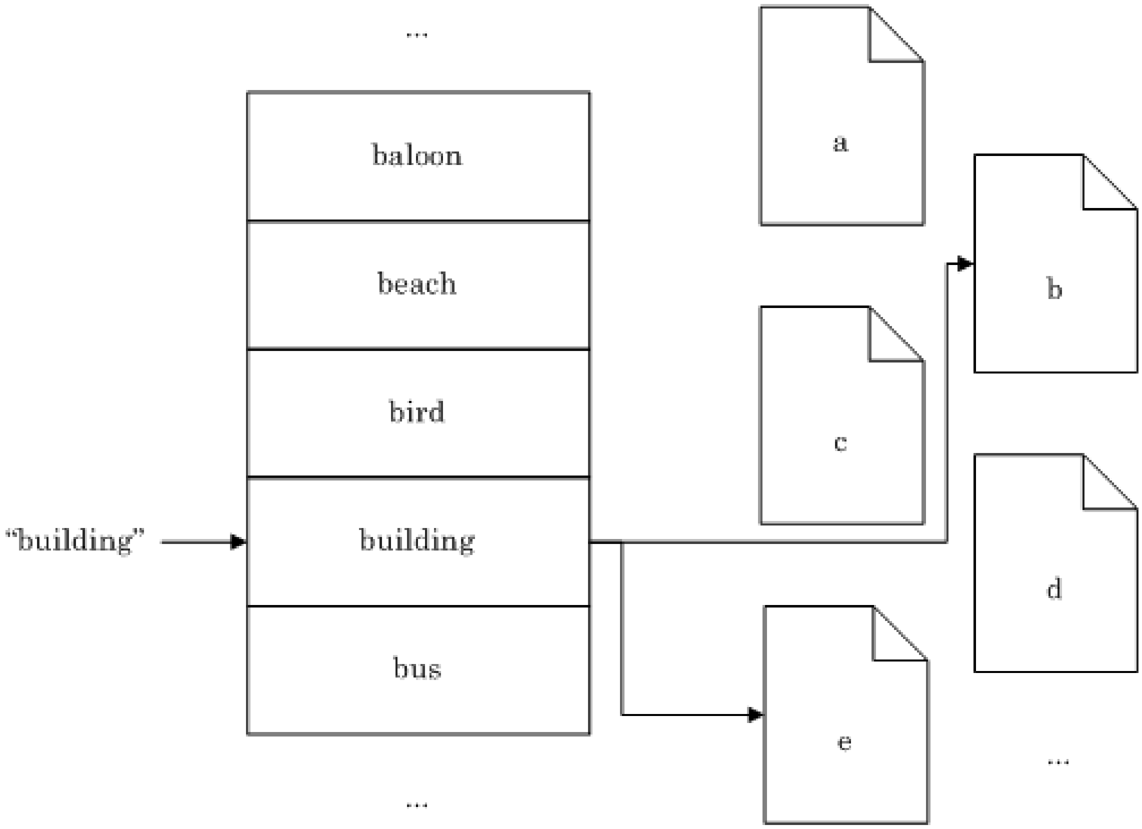


Figure 2.11: Querying using an inverted index

multiple query terms one must first consider how those query terms are associated, be it by a NOT, AND, or OR boolean operator. Each case is handled as follows:

- x **NOT** y: look up x and y in the inverted index. Return all results from the set returned by x except those which also occur in the set returned by querying for y.
- x **AND** y: look up x and y in the inverted index. Return only results which occur in both the set of results for each x and y.

- x **OR** y: look up x and y in the inverted index. Return all results from either set, excluding duplicates.

Documents can be conceived of as vectors in high-dimensional space, where each entry corresponds to a term in the document and a weight associated with that term. Note that this representation disregards the semantic ordering of words in a document. The weighing scheme has several options. Several weighing schemes include (but are not limited to):

- **Binary**: weights are either 0 or 1, i.e. indicating only the presence of a particular term in a document.

- **Number of occurrences of the term**: in this case, the weight is set to the number of times a particular term occurs in a given document. However, this scheme may favor long documents over short ones.

- **Term frequency-inverse document frequency (TF-IDF)**: TF-IDF was introduced in [122]. This weighing scheme takes into account the frequency of a term within a particular document, but also normalizes by the total number of times the term appears in the corpus. The result is a weighing method that gives "rare" or "novel" terms more prominence (as opposed to common terms such as "the", "as", "and", etc.). It is calculated as follows, where i refers to a particular document, j refers to the query term, n_{ij} represents the number of times term j occurs in document i. and $|D|$ is the total number of documents in the corpus:

$$TF_{ij} = \frac{n_{ij}}{\sum_0^k n_{kj}} \tag{2.17}$$

$$IDF_j = \log \frac{|D|}{|n_j|} \tag{2.18}$$

$$TFIDF_{ij} = TF_{ij} \cdot IDF_j \tag{2.19}$$

Documents can be compared in a number of ways, although cosine similarity (see Section 2.4.6) is a typical distance measure [1].

2.6 Collaborative filtering

Collaborative filtering (CF) systems collect information (feedback) from many users and then use this information to recommend new items based on a particular user's profile. Whereas content-based system analyze the data itself to determine relevance, CF inspects past user actions, searches for similar users, and bases recommendations on these results.

A collaborative filtering algorithm is used to recommend products a user may be interested in based on their past actions. These recommendations are determined by the interests of similar users [73]. It is important to note that at no point has any analysis of the content of the items sold (their textual descriptions, music, etc.) ever been done. Collaborative filtering is entirely content-free.

Collaborative filtering is far from a solved problem, despite the effective results that sites such as Amazon.com have demonstrated. For example, Netflix, a movie rental service, has offered a prize of $1 million to the party that can submit an improved collaborative filtering method [97]. As of the writing of this dissertation the Netflix prize remains unclaimed.

According to Herlocker et al. [47], CF systems have three main benefits over content-based systems:

- **The ability to analyze content that is not easily interpreted by automated processes**: examples of feelings, ideas, and politicians as examples of things that cannot yet be effectively analyzed by content-based approaches are provided in [47]. For example, in the domain of image retrieval it is be difficult to use content-based techniques to locate emphatic images (such as "happiness" or "excitement"), but this is a realistic task for a collaborative filtering system.

- **The ability to provide recommendations based on user taste**: CF systems are able to take advantage of uniquely human judgments (such as taste). For example, CF systems may be able to determine well-taken photographs by relying on the ratings of many users.

- **The ability to return serendipitous recommendations**: a shortcoming of content-based systems is that there are many cases where relevant results have minimal common content. In these instances CF systems are able to yield results that are sometimes surprising, given that the user would not have considered them otherwise.

Herlocker et al. [47] state “the potential for collaborative filtering to enhance information filtering tools is great. However, to reach the full potential it must be combined with existing content-based information filtering technology”. Melville et al. [91] concur with this statement.

There are three key issues facing collaborative filtering systems:

- **Sparsity [91]**: the vast majority of items in a dataset are only rated by a small subset of users. The odds of finding similar users are low, particularly in the early life of a system. Content-based retrieval techniques do not suffer from this issue. Generally, they provide complete similarity information for all images in the database.

- **First-rater problem [91]**: this is also known as the *cold-start* problem. No item can be recommended unless it has previously been recommended. It is a particular challenge to new and obscure items in the dataset. In content-based systems features are usually extracted prior to the system becoming operational – there is no equivalent of the first-rater problem in CBIR.

- **Gray sheep [12]**: without many users, individual ratings may vary significantly. There is no expectation of consistency between users. Users whose own preferences differ widely from most others have little hope of obtaining useful results from a CF system.

Schein et al. studied the cold-start (first-rater) problem in [126]. "Pure collaborative filtering cannot help in a cold-start setting, since no user preference is available to form any basis for recommendations" [126]. Thus, in the absence of CF information, the content itself must be inspected using content-based methods. Three modes of testing were employed:

- **Implicit rating prediction**: when no rating is available, such as when a user purchases an item, the action indicates need but not satisfaction.
- **Rating prediction**: if ratings are available (such as for movies) it is possible to predict future ratings.
- **Rating imputation**: imputation is the prediction of data for which a rating is implied, but not explicit. For example, if a user has seen a movie, the objective may be to determine how likely they are to give it a particular rating. Imputation is useful when datasets are incomplete. It is not always necessary to imputer missing data. In simple cases incomplete items can be deleted or ignored.

Melville et al. [91] augmented CF with content-based information in order to address the two issues (sparsity, first-rater problem) they identified. Their technique turns the sparse CF matrix into a dense one by filling it in with the content-based results. CF is then performed on the dense matrix. The technique is known as *content-boosted collaborative filtering*, or CBCF. As per the implementation, the sparsity problem is resolved. Users have ratings for all items. Because all users now have items in common with all other users, there are far more choices to rely on for potential ratings. In their experiments, this approach produced far more robust results when randomly dropping elements from the CBCF rating array. The cold-start problem is also mitigated by CBCF. In the absence of CF predictions, content is instead used to make a prediction. In addition, their results show that the CBCF prediction outperforms the baseline content-based prediction. Finally, if either the content-based or collaborative filtering components of the CBCF model are improved, the overall predictions will as well.

Claypool et al. considered the specific case of finding articles in online newspapers and also produced a system that combined content-based and collaborative filters [12]. "Both humans and computers need help filtering information" [12]. This point is well-taken. The professional movie critic is a human filter of information. These days, however, the amount of information is expanding faster than we can manually filter it. Thus, there is promise for approaches such as CF as well as the semantic web, which promises to make a much wider volume of information interpretable by automated agents [84]. “Collaborative filtering applies the speed of computers with the intelligence of humans” [12].

In the same work, the authors propose a hybrid approach that formulates predictions based on the weighted average of content-based and collaborative filters. The weighing method of Claypool et al. [12] is as follows: as user ratings are recorded the error between the rating and the content-based prediction is computed. Weights are then changed to minimize past error. In their words, the “approach fully realizes the strengths of content-based filters, mitigating the effects of the sparsity and the early rater problems” [12]. The *early rater problem* is another term for the cold-start and first-rater problem. It is particularly interesting that their implementation balances content-based and collaborative predictions uniquely for each user. Additionally, content-based and collaborative filters can be balanced for each item in the dataset. “As the number of users and ratings for the item increase, the collaborative filter is (usually) weighted more heavily, increasing the overall accuracy of the prediction” [12].

“Both the collaborative filtering and content-based scored are important but the extent of their importance towards the aggregate score (or prediction) is very user-specific” [12]. This statement highly correlates to the interaction of top-down and bottom-up processes in human visual attention. Indeed, content-based analysis techniques are based on predetermined rules that have been manually tuned over time. Collaborative filtering is based on knowledge, acquired knowledge applied to a particular situation. Content-based image retrieval suffers from the semantic gap and the sensory gap because it is an inherently bottom-up process. No matter how strong content-based analysis is, it will have a difficult (if not impossible) time bridging these gaps without additional top-down information. In certain limited domains the top-down knowledge may be learned, but this knowledge cannot be extended to generic situations. User- and situation-specific knowledge (collaborative filtering) is the missing top-down component that offers hope for overcoming the gaps of CBIR.

Paulson and Tzanavari concur that a hybrid content-based and collaborative model holds promise [110]. “Hybrid techniques seem to promise to combine the positive features of both content-based and social-filtering [collaborative filtering] methods, diminish their shortcomings, and thus produce a more robust system” [110]. The authors proceed to present their own hybrid approach which uses conceptual graphs which consist of concepts and relations. “There have been few other attempts to combine content information with collaborative filtering” [91].

There is a relatively limited amount of work that has applied collaborative filtering to image retrieval. Kanade and Uchihashi [65, 146] proposed an approach known as *content-free image retrieval* which takes advantages of the human user’s own perceptual abilities. Instead on trying to retrieve images based on analysis of their content they take a different approach. They ask the user to group similar images together. Over time many relationships can be collected. Retrieval relies solely on this feedback. If image A is often grouped with image B then the probability that image B is related to image A increases. Two issues are identified. First, there is the cold start problem. The system needs to be used to group images before it is useful in retrieving them. They suggest using content-based retrieval methods as a good alternative. As an alternative, they suggest inserting new images randomly into results and having the use sort out the results. Another issue is the amount of feedback that is needed before the approach will work. This approach was improved in [75] by using a Bayesian framework.

2.7 Summary

This chapter presented background information that will be referred to throughout the rest of this dissertation. The chapter began with an overview of vision science and its disciplines. Attention was then expanded on, due to its applicability to this work and the emergence of computational models of visual attention. The computational model developed by Itti et al. [61] and that of Stentiford [134] were discussed. An overview of CBIR, text retrieval, and collaborative filtering was presented, including related work and contemporary challenges.

Chapter 3

PROOF OF CONCEPT

When one is happy there is no time to be fatigued; being happy engrosses the whole attention.

E. F. Benson, author, 1867 – 1940

3.1 Introduction

A new model for grouping images based on the similarities between their salient regions of interest is proposed in this chapter. This work was published in [86, 87]. It incorporates two computational models of human visual attention in order to compute salient regions of interest (ROIs). Features in the RGB and HMMD color spaces are extracted from these regions. These extracted features can then be used in a variety of applications, such as clustering and CBIR.

Section 2.4 provided a discussion of CBIR, including the state of the art and current challenges. While some CBIR systems base retrieval results solely on global image characteristics, several recent implementation have considered retrieval based on individual image regions (e.g. [7, 79, 72]). This field of interest is sometimes referred to as object-based image retrieval (OBIR) [49, 72, 142]. Segmentation is a

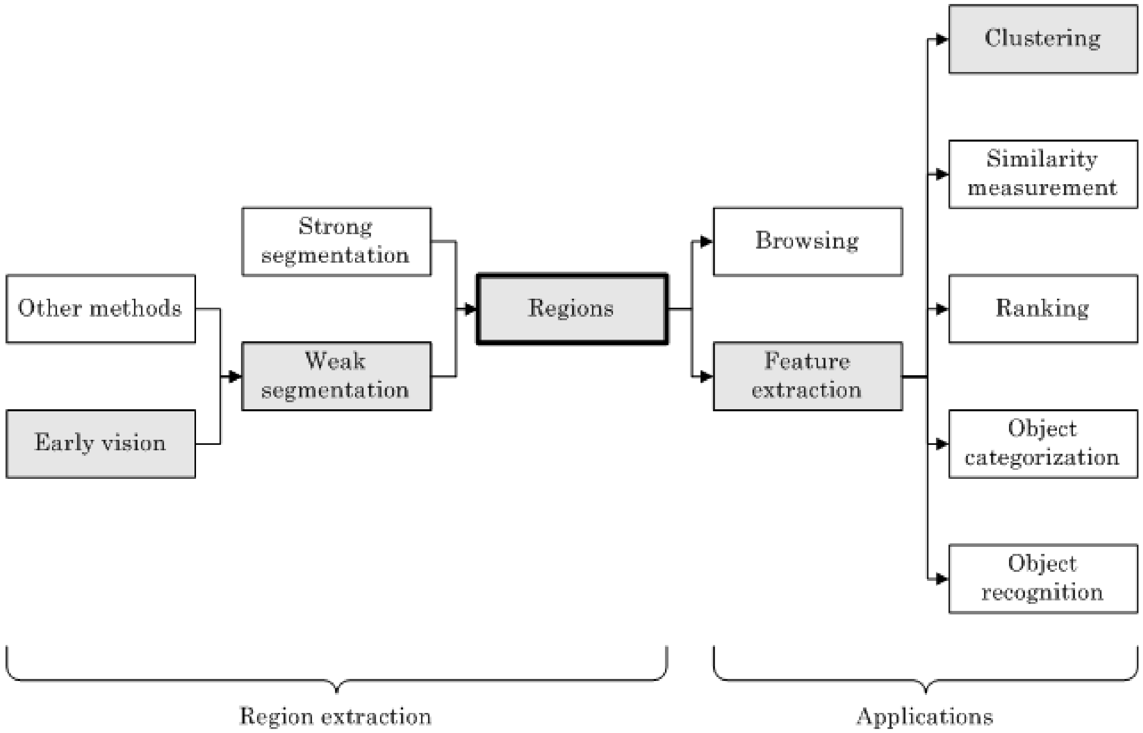


Figure 3.1: Scope of the proof of concept, where where shaded blocks indicate the focus of this work

prerequisite for this type of image retrieval. There are two broad classes of segmentation: strong segmentation (exact) and weak segmentation (approximate) [132].

Figure 3.1 outlines the scope of this work. In this figure the components which are part of this work are shaded, although other notable alternative implementations are also illustrated. The objective of this work is to extract regions (the center of the figure). Regions can be extracted using either weak or strong segmentation. Weak segmentation was selected for this work. Although weak segmentation may result in segments that are not ideally-formed, strong segmentation is difficult to successfully implement, particularly for broad domains [132]. This project uses a computational model of early visual processes (especially bottom-up visual attention) to generate weak image segments. The computational model of bottom-up visual attention was selected partially due to its novelty (this is among the first work to apply early vision to CBIR), and partially due to its potential to detect relevant areas of an image. Extracted regions can either be used for browsing or in further analysis. This work extracts RGB and HMMD color features from the regions and uses these features to cluster the regions, although other applications are possible (e.g. similarity measurement, ranking, object categorization, and object recognition).

It has been shown that promising results for the detection of salient regions can be achieved using even a simple model of human visual attention. For example, Draper et al. [27] find corners in images and uses these corners as cues of saliency. The work models the *expert object recognition pathway*, which is the part of the brain which is able to recognize specific objects. There are four components of their model: early vision (visual attention), the lateral occipital complex (extraction of edge-based properties), the fusiform gyrus (categorization), and the right inferior frontal gyrus (instance matching).

Computational visual attention was used for image exploration in [81]. The model in their work modulates the saliency map (encodes top-down information) by using past knowledge, aiding object recognition. This contrasts with the proposed model of this experiment, which is purely bottom-up. Additionally, our implementation is unsupervised, whereas in [81] user interaction is needed.

Research has shown that the performance of object recognition systems can be improved with the inclusion of a computational model of visual attention [120]. In the case of [120], the model of attention guides the system to recognize only the most salient objects within a scene. This dissertation extends a similar approach to CBIR. However, there are several differences between object recognition (as in [120]) and similarity-based image retrieval (this work). These differences include the degree of interactivity, the relative importance of recall and precision, the broader application domains and corresponding semantic ranges, and the application-dependent semantic knowledge associated with extracted regions [13].

The key hypothesis tested in this chapter is that image retrieval can be improved by using a computational model of visual attention. Section 3.2 presents the design and components of the *proof of concept* system. Experiments and results appear in Section 3.3. The proposed method's ability to detect salient regions of interest is evaluated in Section 3.3.2. The method is then applied to CBIR in Section 3.3.3. A discussion of the experiments and results is presented in Section 3.4.

3.2 Design

3.2.1 Overview

This section presents a biologically-inspired approach to image retrieval that extracts ROIs using two computational models of visual attention. The effectiveness of the method is empirically demonstrated using a 110-image database containing 184 regions of interest, with each region of interest being salient by design.

The method demonstrated here incorporates two models of visual attention. The first, the Itti-Koch model [61] produces a *saliency map* which targets the most salient locations within an image at a certain point in time. The other, the Stentiford model [135], produces a *visual attention map* which highlights the most unique areas of an image. The saliency map is used as a first cue to which points in an image are salient, and the visual attention map is used to extract salient regions around the aforementioned points. Image are clustered together based on the features extracted from these regions, not from global areas of an image. This results in images being associated with other images sharing common salient regions, rather than global characteristics (e.g. two images dominated by a blue sky are not necessarily similar in our model unless they also share a common ROI).

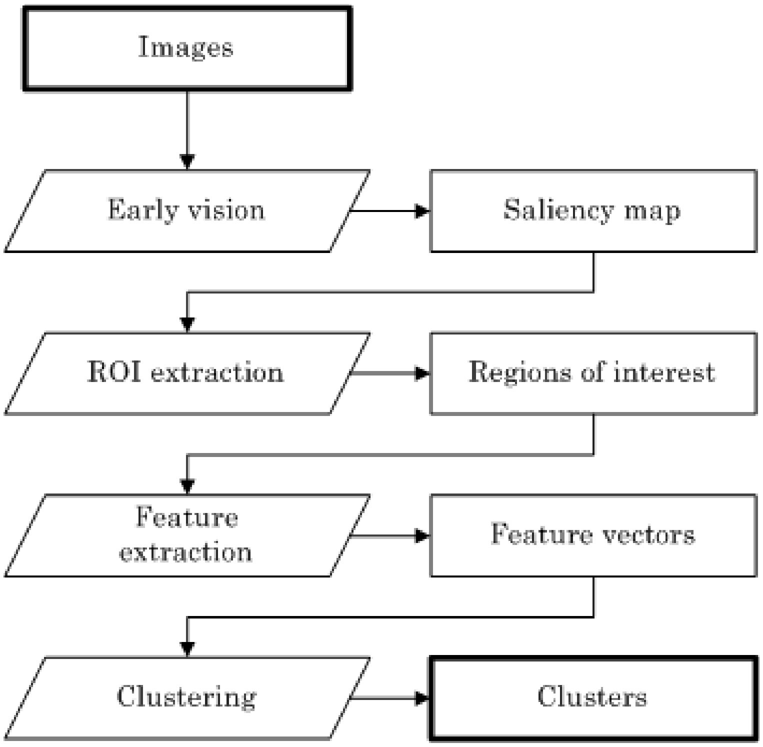


Figure 3.2: The proposed attention-driven model for grouping similar images

3.2.2 Components

The proposed design consists of four stages, as depicted in Figure 3.2: early vision (visual attention), region of interest extraction, feature extraction, and clustering. These stages are described in the following list:

- **Early vision**: The first stage models early vision, that is, what our visual attention system is able to perceive in the first milliseconds after initially perceiving a stimulus. The purpose of this stage is to indicate the most salient areas of an image. The input to this stage is a single source image. The output is the saliency map (derived from color, intensity, and orientation) and the visual attention map. We use the Itti-Koch model of visual attention to generate the saliency map. It has been successfully tested in a variety of applications [58]. Saliency maps were computed using a Java implementation of the Itti-Koch model of visual attention [98]. The visual attention maps proposed by Stentiford were generated by our research group's MATLAB implementation of the methods described in [135]. The proposed model is not domain-specific and does not impose limits on the variety of images that it applies to, provided that there is at least one salient ROI within the image.

- **Region of interest extraction**: This stage generates ROIs based on the saliency and visual attention maps. A detailed description of the process can be found in [87]. It is inspired by [120]. The model appreciates not only the magnitude of the peaks in the saliency map, but the size of the resulting salient regions as well. The extracted ROIs represent the areas of the image that are likely to be attended to first. These are the only regions of the image that are used for feature extraction, the next stage in our model. This algorithm combines the Itti-Koch saliency map with segmented results from Stentiford's visual attention map in a way which leverages the strengths of both methods while mitigating their shortcomings. Two major strengths of the Itti-Koch

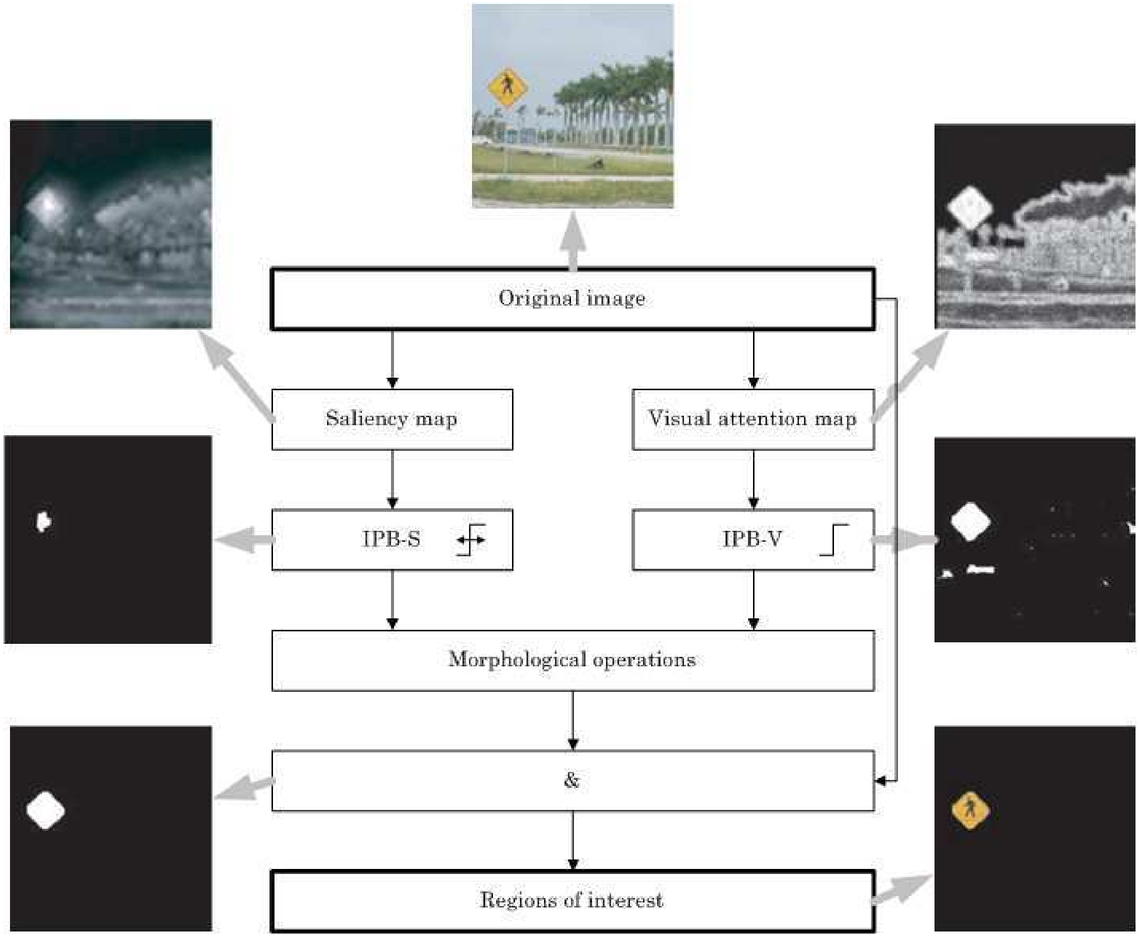


Figure 3.3: General block diagram of the region of interest extraction algorithm

saliency map are its ability to account for color, orientation, and intensity, as well as its ability to discriminate between salient regions. In contrast, Stentiford's visual attention map is based on color and shape alone and is less discriminative. Stentiford's method excels in bounding salient regions, whereas Itti-Koch emphasizes salient points. This enables the method proposed by Stentiford to handle relatively large regions.

The ROI extraction algorithm is illustrated in Figure 3.3. A binarizing threshold is applied to the grayscale Itti-Koch saliency map in order to extract the most salient points. This threshold is a critical parameter in our experiments. A lower threshold will result in more seeds being passed to the next phase and, ultimately, more predicted regions of interest (more true positives and false positives). A higher threshold results in fewer seeds for ROI extraction (fewer true positives and more false negatives). If the threshold is very low, the ability to discern between multiple ROIs will be reduced as multiple independent regions will appear to be a single large region (a low threshold will result in more of the image appearing to be salient). This is the worst case as the number of false positives and false negatives both increase. The IPB-S (Image Processing Block – Saliency Map) module is responsible for this function of binarizing the saliency map. The complementary block for the visual attention map is labeled as IPB-V (Image Processing Block - Visual Attention Map) in this figure. This block binarizes the visual attention map, although the binarizing threshold is far less sensitive due to the nature of the visual attention map. Regions in the Stentiford visual attention map that occur in the same location as salient points are preserved and used as a mask to extract regions of interest. Ideally, this method will produce a mask which corresponds to the most prominent objects in a scene and thus allows is to discard what is not salient. The mask generation block combines the outpue of IPB-S and

IPB-V into a mask of all ROIs in the image. Both maps are combined using the logical AND function.

- **Feature extraction**: The proposed system is flexible enough to allow a variety of feature extractions algorithms to be used (e.g. common CBIR features such as color histograms, color correlograms, texture descriptors, shape descriptors, etc.). The reader is referred to [80, 24] for a comparative analysis of such features. These features are calculated on a region-by-region basis, resulting in each ROI being assigned its own feature vector and an image potentially being associated with multiple feature vectors. In this work two color-based feature extraction descriptors have been been implemented. The first is a 27-bin RGB color histogram. The other is a 32-cell quantized HMMD descriptor. The HMMD descriptor operates in a color space that is closer to being perceptually-uniform than the RGB color space. Thus, we anticipated better results from HMMD than RGB.

- **Clustering**: Ultimately, this design groups the feature vectors together by employing a general-purpose clustering method. Because several regions of interest may be extracted from a single image, an image may also be assigned to multiple clusters as well. This is a distinction between this approach and other cluster-based approaches, which often restrict cluster membership to one cluster per image. The flexibility afforded by the existence of multiple

ROIs per image allows images to be associated based on the characteristics of the regions that are more likely to be perceptually-relevant (rather than the alternative, global information). Chen et al. [11] demonstrated that clustering and ranking relevant results is a viable alternative to the typical method of presenting results as a linear ranked list. An example of our results, illustrated in Figure 3.4, shows how 18 images have been divided into five clusters based on their salient regions of interest. The images are from the *FAU Salient* image database (Section 4.4.2). There are a total of 20 regions of interest. The final clusters group images based on whether they contain the following objects: a miniature basketball, tennis ball, blue plate, red box, and yellow road sign. This example illustrates how the proposed clustering approach groups related images together despite large variations their backgrounds.

3.3 Experiments and results

3.3.1 Methodology

A subset of the STIMautobahn, STIMcoke, and STIMtriangle image databases (available from `http://ilab.usc.edu/imgdbs/` [60]) was selected for these experiments . Please see Section 4.4 for a detailed discussion of image databases for CBIR. These image databases were chosen due to a particular requirement of this work: each image should contain at least one *salient by design* region. In the selected image databases, these regions may be road signs, soda cans, or road warning triangles,

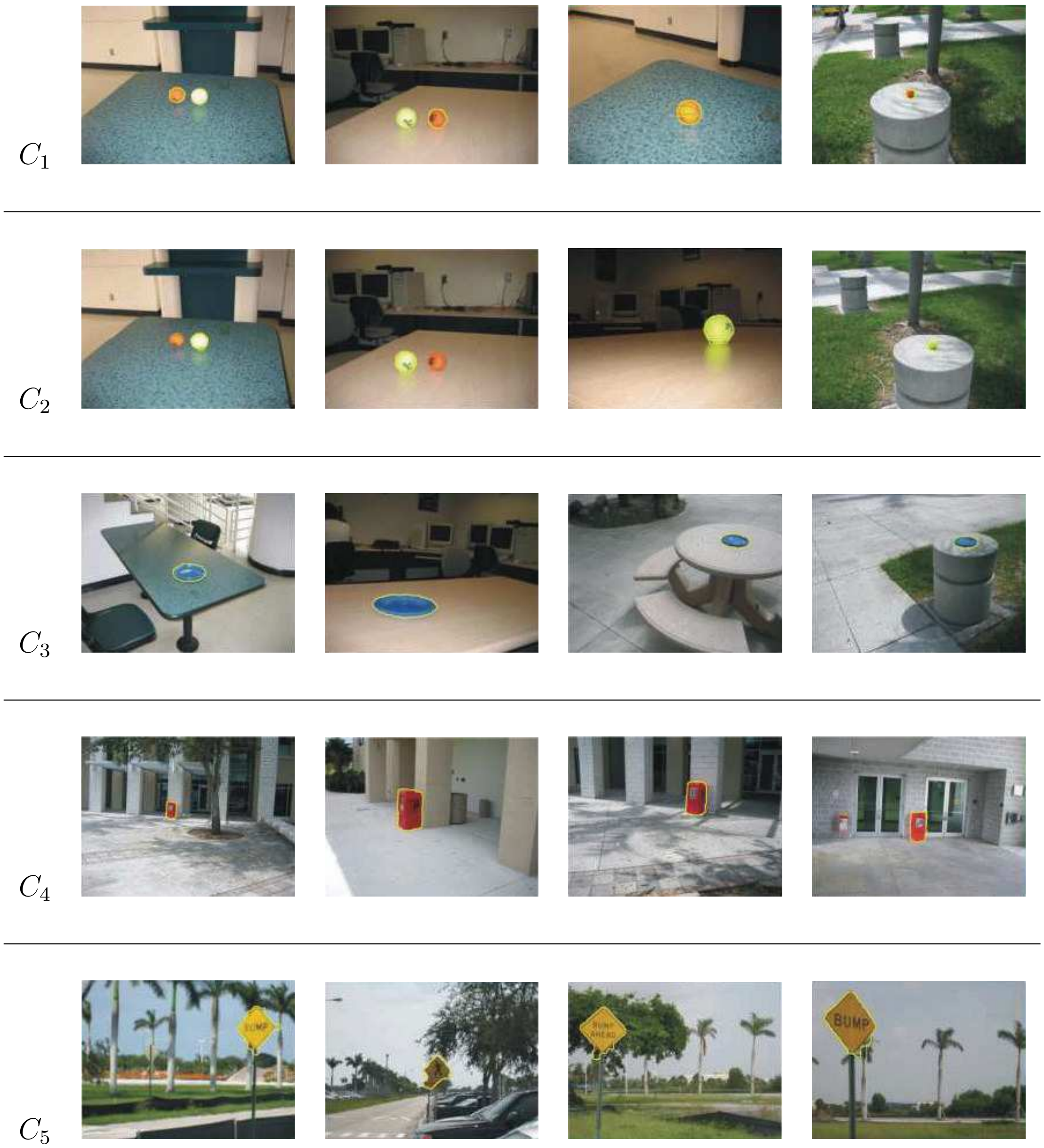
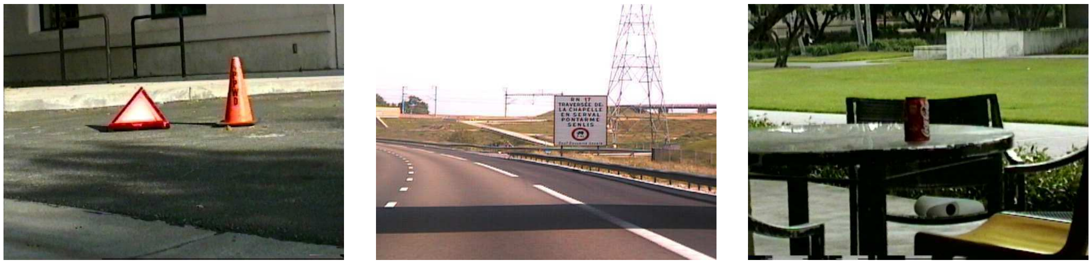
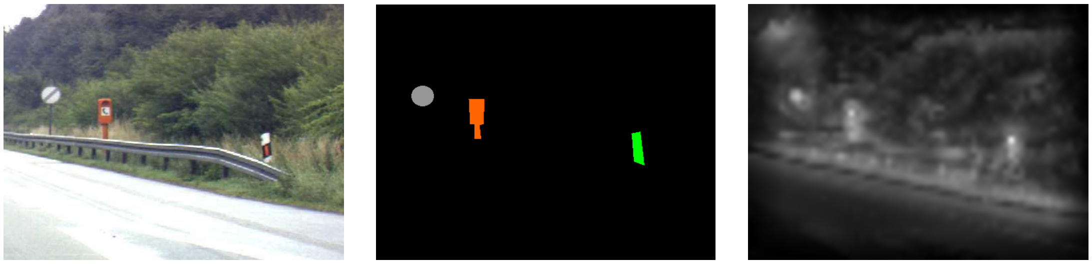


Figure 3.4: Examples of clustering based on ROIs for a small dataset. The extracted ROIs are outlined.



(a) (b) (c)

Figure 3.5: Examples of the three types of images in the resulting dataset. (a) consists of warning triangles, (b) road signs, and (c) soda cans



(a) (b) (c)

Figure 3.6: Sample ground truth and saliency information. (a) is the original image, (b) is the ground truth, and (c) is the resulting saliency map

respectively.

A dataset of 110 images was selected: 41 images from the STIMautobahn set (road signs, Figure 3.5 (a)), 41 from the STIMcoke dataset (red soda cans in a variety of settings, Figure 3.5 (b)), and the remaining 28 from STIMtriangle (road emergency triangles, Figure 3.5 (c)).

Ground truth object masks were manually created by the author of this dissertation and verified by three members of the research group. 184 regions were recorded. Several images contained multiple ROIs. Each image contained at least one ROI. These 184 regions were manually assigned one of 22 semantically-meaningful labels (listed under the *Category* column in Table 3.1).

An example of the ground truth is shown in Figure 3.6. In this figure (a) is the original image. It contains three salient objects (all road signs). The manually-generated ground truth is shown in (b). In this case, the three objects all belong to different clusters, as shown by the different colors assigned to the corresponding ROI mask. Figure 3.6 (c) shows the computed saliency map.

The saliency map was computed for each image in the database and used to extract the salient ROIs following the process described in Section 3.2.2 and illustrated in Figure 3.3. Each ROI was encoded as both a 27-bin RGB histogram and a 32-cell quantized HMMD descriptor. The result was 184 RGB feature vectors and 184 HMMD feature vectors. Each set of feature vectors were independently clustered using the k-means clustering algorithm [66].

The objective of any information retrieval system is to maximize the number of true positives (TP) and minimize both the number of false positives (FP) and false negatives (FN). The more images that are retrieved, the more true positives can be included in the results set. In the extreme case, when the size of the results set is equivalent to the size of the corpus the number of true positives will always be maximized, as every document will be returned. However, this is not a practical approach to information retrieval, as the number of false positives is also maximized.

Practitioners must balance the desire to display relevant results while being cautious of including irrelevant results.

Precision measures how many of the retrieved documents are relevant. It is defined in Equation 3.1. Note that $TP + FP$ is the total number of images or regions retrieved (the total number of results depending on the experiment). If all of the results are relevant then *Precision* is at its maximum value, 1. If none of the documents in the results set are relevant *Precision* is 0.

$$Precision(TP, FP) = \frac{TP}{TP + FP} \tag{3.1}$$

Recall differs from *Precision* in that it measures what part of all relevant results in the corpus was included in the retrieved set of results. Thus, if all relevant results have included, *Recall* is 1. If no relevant results are returned *Recall* is 0. *Recall* is defined in Equation 3.2.

$$Recall(TP, FN) = \frac{TP}{TP + FN} \tag{3.2}$$

Precision and *Recall* have an inverse relationship. It is possible for *Precision* to be 1 (ideal) and *Recall* to be very low, particularly if the corpus is large in relation to the size of the results set. Conversely, if the size of the results set is close to the size of the entire corpus *Recall* will approach 1 and *Precision* will approach 0. The tradeoff between *Precision* and *Recall* is usually plotted in a *P-R graph*.

3.3.2 Region of interest extraction

The objective of this experiment is to verify the quality of regions extracted using a computational model of human visual attention.

It was necessary to determine the ideal value of the binarizing threshold of the IPB-S block (see Figure 3.3). When this parameter is increased, the generated points of attention are reduced in number and the detected regions are fewer (thus, fewer false positives and more false negatives). When the value of this parameter is reduced, more points of attention proceed to the mask generation stage (more false positives but fewer false negatives). A balance must be empirically determined. The natural tool to use for this estimation is the *Receiver Operating Characteristic* (ROC) curve.

ROC curves [19] represent results in a way that allows parameters to be evaluated and tuned. ROC curves plot the $HitRate$ against the $FalseAlarmRate$. The $HitRate$ is equivalent to $Precision$ (Equation 3.3). The $FalseAlarmRate$ (Equation 3.4) is the ratio between the false positives in the result set and the total number of false positives that exist in the entire corpus. It represents wasted effort due to incorrect predictions. ROC curves are generated by varying parameters in a model, affecting the $HitRate$ and $FalseAlarmRate$. Visual inspection of the ROC curve can be used to set the varied parameter. Although the desire is to maximize the $HitRate$, the risk of a $FalseAlarmRate$ that is too high must be taken into account.

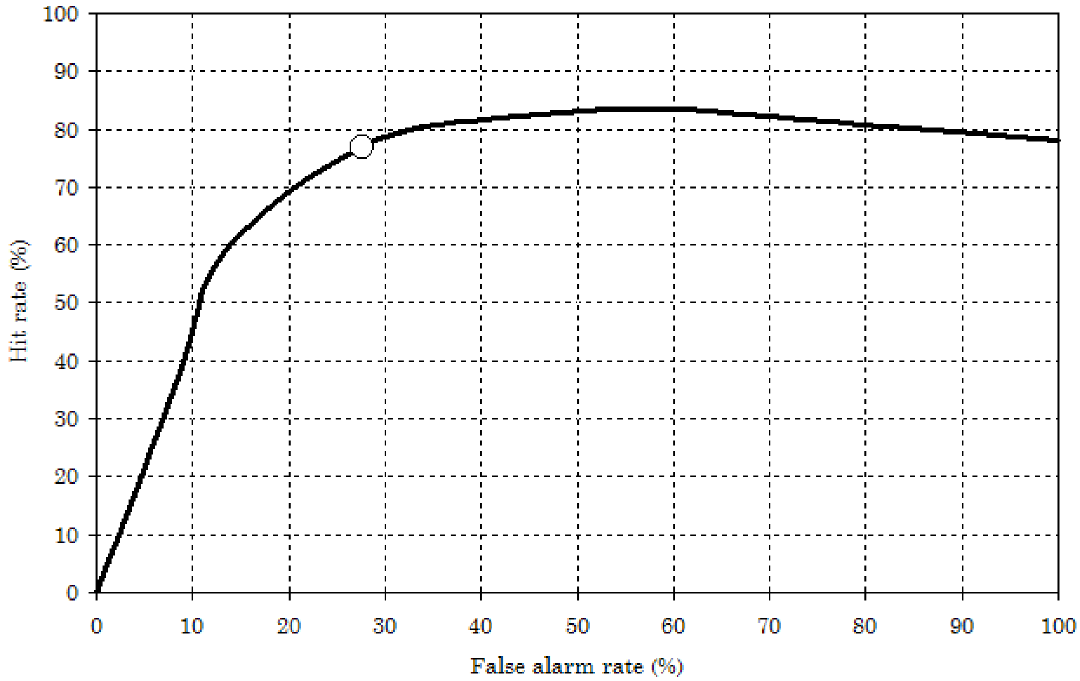


Figure 3.7: ROC curve generated to evaluate the ROI extraction algorithm as a function of the threshold used to binarize the saliency map in the IPS-S block.

$$HitRate(TP, FP) = Precision(TP, FP) = \frac{TP}{TP + FP} \tag{3.3}$$

$$FalseAlarmRate(TP, FP) = \frac{FP}{TP + FP} \tag{3.4}$$

The resulting ROC curve is shown in Figure 3.7. This figure plots the *false alarm rate* (vertical axis) against the *hit rate* (horizontal axis).

Regions obtained in this experiment are classified as follows:

- **True positive**: in this experiment, a true positive is a region in the ground truth that has been identified by the proposed ROI extraction algorithm. In Figure 3.10 (a), the warning triangle is a true positive.

- **False positive**: a false positive is a predicted ROI (a region that remains at the end of the ROI extraction method) but does not correspond to any ground truth ROI in the same image. In Figure 3.10 (a) the spurious regions are false positives.

- **False negative**: a false negative is an ROI from the ground truth that is not identified by a predicted ROI. In Figure 3.10 (b) the soda can is a false negative.

The other possible case, true negatives, does not apply to these experiments as this is not a binary classifier and we do not explicitly identify regions of the image that are not to be detected.

The marked point on the ROC curve is at a hit rate of 76.74% and a false alarm rate of 27.67%. The threshold value that generated this point, 190, was determined to be the best value of the binarizing parameter for this dataset. Other points on the ROC curve could have been selected. For example, one may be more cautious and select a point that yields a lower false alarm rate. Or, one could be more aggressive and select a threshold resulting in a higher hit rate.

3.3.3 Content-based image retrieval

The extracted regions (at the optimum threshold value – Section 3.3.2) are evaluated in a CBIR system in this section. The objective of this experiment is to compare the performance of the extracted regions to a baseline case (searching by global image features) and the optimal case (searching by perfectly-segmented regions of interest – the ground truth). In this section we test the hypothesis that CBIR will be improved by searching by region-based rather than global features.

The same 110-image database and ground truth (184 manually-defined regions, 22 categories) used in Section 3.3.2 was used for this experiment. The HSV color histogram was extracted. Thus, three sets of features were compared:

- The HSV color histogram for all pixels in each of the 110 images (global)
- The HSV color histogram for the 184 ground truth regions of interest
- The HSV color histogram for the 194 extracted regions of interest

Note that more regions were extracted from regions (194 regions) than exist in the ground truth (184 regions).

In order to evaluate the system, 184 queries were posed – each ground truth region was a separate query. Three cases were evaluated: global (using every pixel in the image), regions of interest (using the area of the image coinciding with the extracted regions), and the ground truth (the area of the image coinciding with the particular ground truth mask). The precision-recall graph for the HSV color

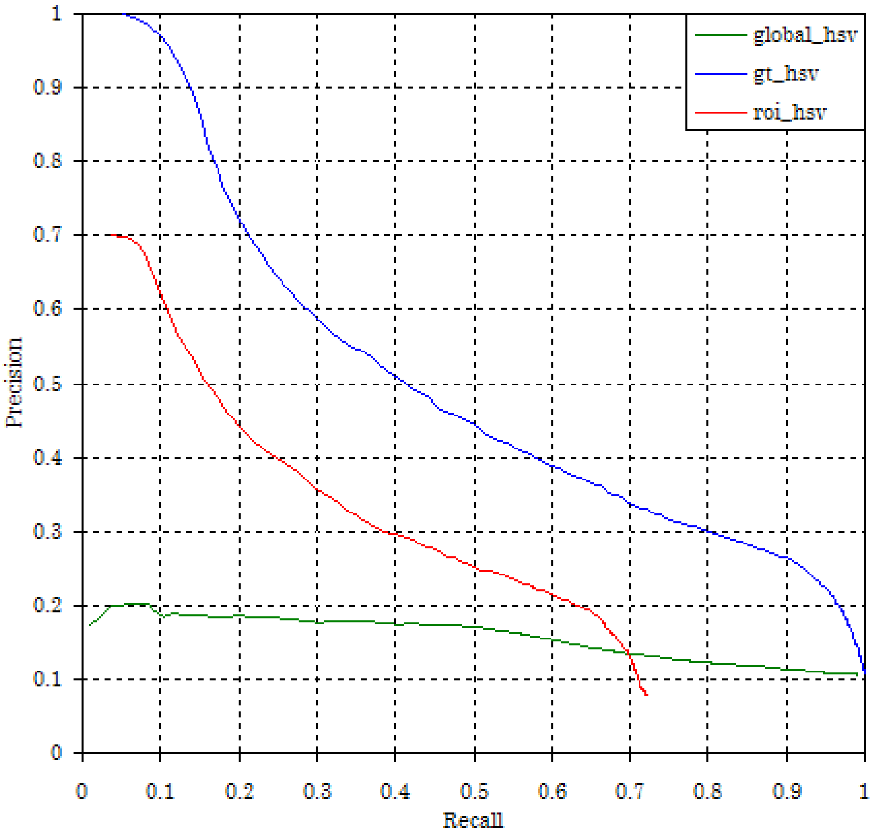


Figure 3.8: Precision-recall results for the HSV color space

histogram feature is shown in Figure 3.8. The D1 distance measure (defined in Section 2.4.6) was used to compare feature vectors and rank the results.

It can be observed from Figure 3.8 that the worst performance results from the *Global* feature, the best from the *GT* (ground truth) feature, with the *ROI* (extracted region of interest) feature exhibiting performance that is in between *Global* and *ROI*. The graph shows that global features that weight all parts of an image equally are not appropriate for region-based image retrieval. The ground truth results show the optimum potential performance of this approach, if one were able to extract strongly-segmented, perfectly-defined regions. The ROI results show that our performance performs better than global features at all but the highest recall rates, which confirms the hypothesis presented at the beginning of this section. The disparity between the performance of the ground truth and that of the extracted regions can be narrowed by improving the ROI extraction method. Please note that the ROI feature never reaches full recall (recall equal to 1) because it is not able to extract all regions (i.e. false negatives remain).

Table 3.1 shows the mean and weighted average precision for all categories for the HSV feature. The mean average precision is the average of all values. The weighted average precision is biased by the number of images in the category (shown in the *Count* column). Certain categories are clearly appropriate to use with this approach, such as *black_red_inverted_sign*, *orange_pylon*, and *yellow_fire_hydrant*. Others, such as the *yield* category are not appropriate for region-based image retrieval and exhibit better performance using global features. Certain categories are not detected, regardless of global- or ROI-based features, including *gray_utility_connection* and *green_lawn_ornament*.

The overall weighted mean of all categories shown in Table 3.1 for region-based retrieval is 0.39. The ground truth weighted mean is 0.59. Using global features results in a weighted mean of 0.19. Both region-based and ground truth

Table 3.1: Mean and weighted average precision for all categories using HSV

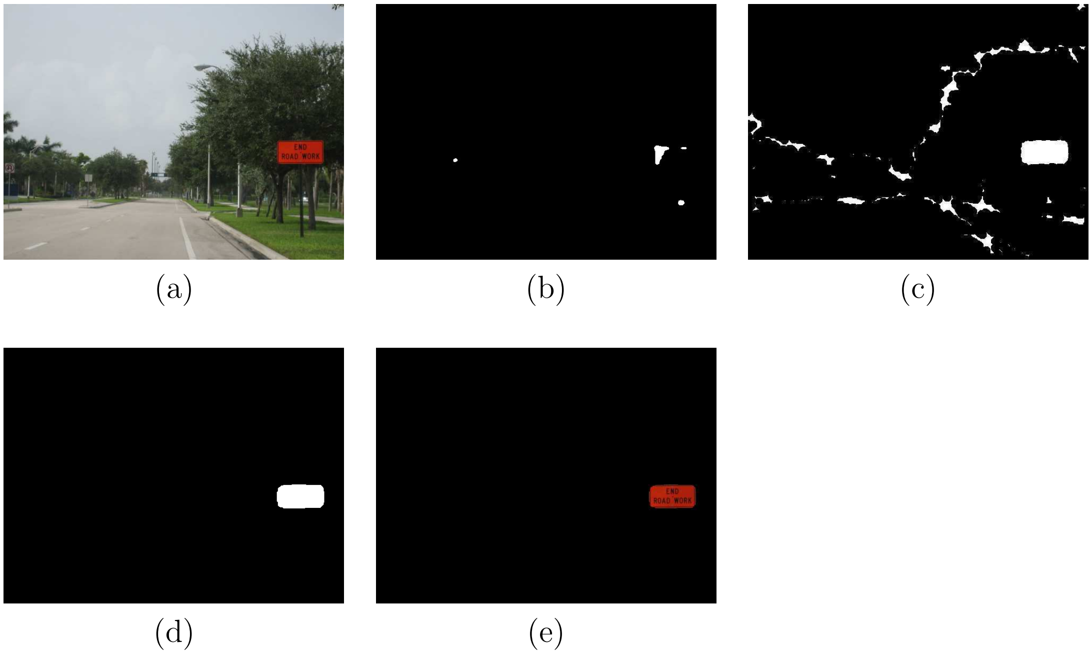
Category	Count	Global	ROI	GT
black_red_inverted_sign	3	0.00	1.00	1.00
blue_round_sign	6	0.17	0.50	0.50
coke_can	41	0.34	0.63	0.68
gray_utility_connection	3	0.00	0.00	1.00
green_lawn_ornament	2	0.00	0.00	1.00
handicap_sign	5	0.00	0.36	0.44
orange_black_marker	2	0.00	0.25	0.50
orange_phone	3	0.00	0.56	0.89
orange_pylon	3	0.00	1.00	1.00
red_white_circle	4	0.19	0.13	0.44
small_simple_white_sign	1	0.00	0.00	1.00
tennis_ball	13	0.12	0.12	0.46
triangle_warning_sign	28	0.26	0.48	0.55
turquoise_cup	9	0.09	0.15	0.56
white_flat_object	3	0.00	0.67	1.00
white_gray_sign	10	0.23	0.37	0.42
white_marker	20	0.24	0.06	0.56
white_red_turning_bank_arrow	3	0.00	0.67	1.00
white_square_red_circle	9	0.16	0.21	0.26
yellow_fire_hydrant	2	0.00	1.00	1.00
yellow_sign	11	0.02	0.24	0.36
yield	3	0.22	0.00	0.78
Arithmetic mean		0.09	0.38	0.70
Weighted mean		0.19	0.39	0.59

features outperform the global approach, confirming the hypothesis proposed in this section. The disparity between the ground truth and the ROI results can be reduces by improving the process of extracting regions. Even with the current method, the results compare favorably to the state of the art in CBIR. For example, Datta et al. combine human-generated ontologies with CBIR in order to automatically add semantic information to images [16]. Average precision for their implementation is between 35% and 45%, depending on the scenario. Our performance is in line with these results.

3.4 Discussion

This chapter presented a method to detect regions of interest using a computational model of visual attention. The experiments were performed on a 110-image dataset encompassing 184 ROIs with at least one ROI per image. The extracted ROIs are derived from the most salient regions of the image. The proposed method was also evaluated in a CBIR system. It was demonstrated that the extracted regions outperform retrieval using global features.

At the selected binarizing threshold for the saliency map 77% of ROIs are detected, with a false alarm rate of 28% (obtained in Section 3.3.2). The proportion of ROIs detected (77%) indicates that there are still cases where ROIs will never be found (23% false negatives). Other, methods, such as those that are not content-based, must be used in these cases. This limited the performance of the implemented



(a) (b) (c)

(d) (e)

Figure 3.9: An example of region of interest extraction. (a) shows the original image, (b) the processed saliency map, (c) the processed visual attention map, (d) the region mask, and (e) the extracted regions.

CBIR sytem (Section 3.3.3 as well).

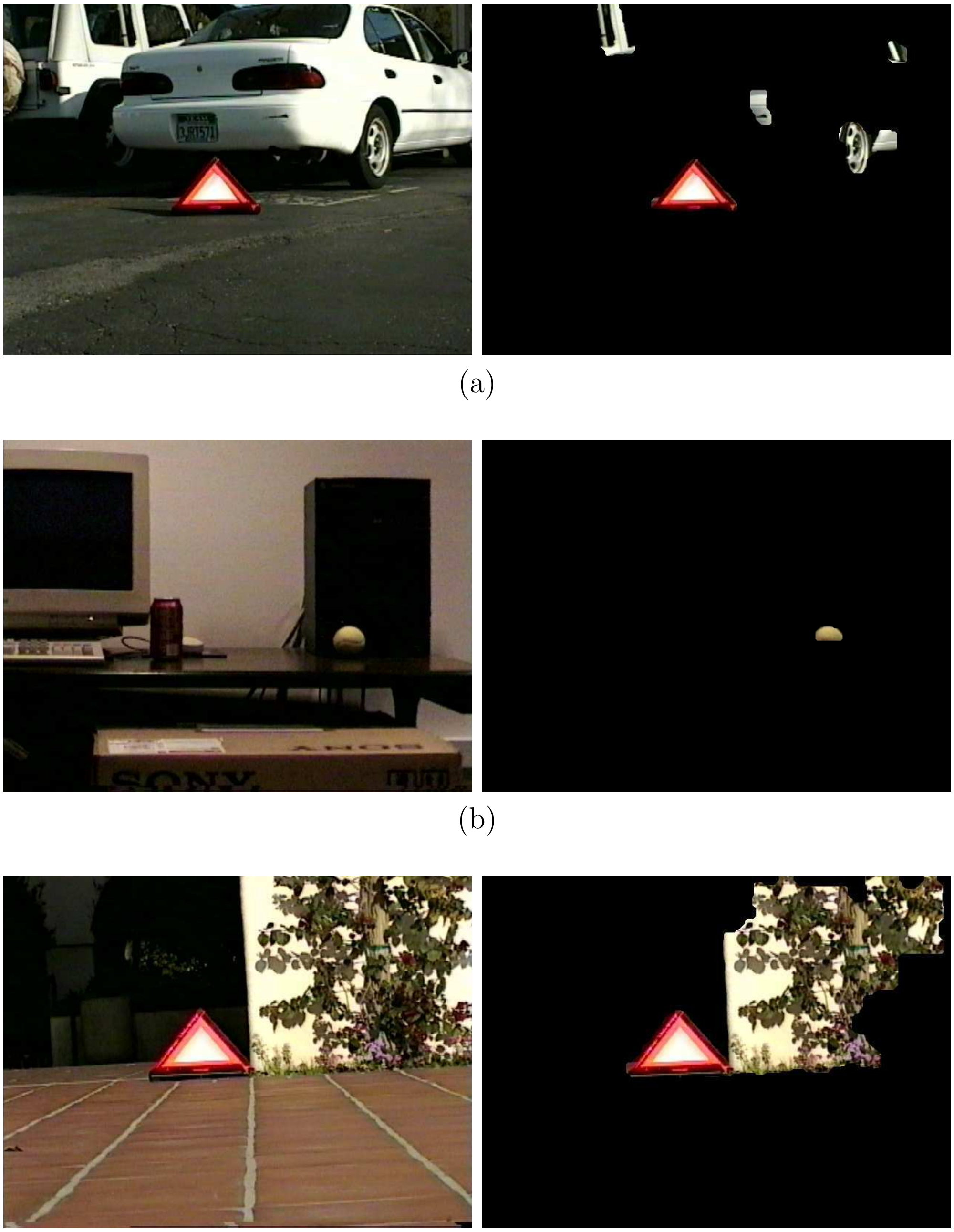
An example of a successfully extracted regions is shown in Figure 3.9. The original image is shown in Figure 3.9 (a). In this case the target image is the orange road sign. The saliency map (Figure 3.9 (b)) hits the road sign as well as several extraneous regions. The visual attention map (Figure 3.9 (c)) exhibits similar results. Note the ability of the visual attention map to extend the region to nearly the border of the road sign. The generated mask (Figure 3.9 (d)) eliminated the spurious regions and segments the target (Figure 3.9 (e)).

It is possible to improve the ROI extraction algorithm. There are three ways the extraction method can fail:

- **False positive**: a ground truth ROI does not correspond to a predicted ROI. This case is shown in Figure 3.10 (a). While the intended region, the orange and white warning triangle, is detected, many extraneous regions are also found. These are all labeled as false positives as they do not correspond to regions in the ground truth.

- **False negative**: the predicted ROI does not correspond to a ground truth ROI. Figure 3.10 (b) illustrates this case. In this figure the intended ROI, a red soda can, is not detected. Because the region is missed it is recognized as a false negative. Instead, a region was generated for the distractor, a lime-colored tennis ball. Since this region does not exist in the ground truth it is designated as a false positive.

- **False region**: the predicted ROI does correspond to a ground truth ROI, but is it an inadequate representation (either too large or too small). In Figure 3.10 (c) a single region was predicted by the proposed model. While this region correctly encompasses the intended object (the red warning sign), it includes a large amount of extraneous information – too much to consider this a useful region. The feature vector generated from this region would not be accurate. Note that this case includes the instance in which a region grows so large as to encompass two regions (label two independent regions as one).



(a)

(b)

(c)

Figure 3.10: Cases in which the method of predicting ROIs fails. (a) indicates several false positives. (b) is a false negative. (c) is a false region.

ROC curves are traditionally monotonically increasing, but the one presented in Figure 3.7 is not. While the ROC curve produced by these experiments initially corresponds to a traditional ROC curve, towards the end the hit rate actually decreases while the false alarm rate continues to increase. This is due to the nature of modifying the binarizing threshold in our experiment. Lower thresholds result in more seeds and larger regions of interest. When ROIs become too large and overlap each other, the ability to retrieve and distinguish between multiple ROIs is reduced. Missing an ROI will have repercussions throughout the remainder of the process, as it can never be recovered. An absent ROI can affect entire clusters of predicted regions.

Restricting the model solely to bottom-up visual attention limited its performance – no region-specific information could be included. For example, we could have gently introduced top-down heuristics by extracting signature features from each of the ground truth clusters and using this to modulate the sensitivity of the saliency map (making it more sensitive to these types of regions) or filter out outlier ROIs.

Lu et al. modified the model presented in this chapter [77]. Their work was based on results published by the author's research group in [86]. Lu et al. replace the Stentiford visual attention map with an *expectation-maximization* (EM) algorithm for segmenting images. They constructed a complete CBIR system based on the model. Their system was evaluated using 5000 images from the Corel image database divided into 50 categories. The approach was compared against seven different CBIR methods (fusion, UFM model, IRM model, two types of global HSV color histograms, color indexing, and EHD). In every case, their implementation (derived from the one presented in this chapter) exhibited superior performance.

Chapter 4

THE NEW METHOD

The only factor becoming scarce in a world of abundance is human attention.

Kevin Kelly, editor, b. 1952

4.1 Introduction

The previous chapter presented a *proof of concept* implementation of a method to extract and cluster salient regions of interest from images. This chapter presents a *new method* that extends that work in the directions presented in the following paragraphs.

Among the most significant extensions to the proof of concept work is the dataset used. Instead of being limited to *salient regions*, only *objects* are considered. Salient regions may include areas of an image that are salient, such as foliage, mountain peaks, or the sky, but are not semantically-meaningful objects. This work uses the more restrictive category. Furthermore, these objects are not necessarily salient by design. The size of the dataset has been expanded from 110 images containing 184 salient-by-design regions to 17,436 images containing 45,064 objects.

Finally, the proof of concept database used manually-generated, accurate ground truth. This new method uses the provided ground truth, with no guarantee as to its quality.

Whereas the proof of concept was specifically designed to cluster related regions, this portion of the work may be applied to broader applications, such as classical QBE CBIR.

The proof of concept used a Java implementation (`http://privatewww.essex.ac.uk/ ranewc/research/visualAttentionJava.html`) of the computational model of visual attention which was not the official implementation. The new method employs the official, more detailed C++ implementation known as Ezvision (`http://ilab.usc.edu/toolkit/`).

In the proof of concept work only the initial saliency map was used. In this work the entire model, including the inhibition of return is used to simulate points of attention.

The objectives of this portion of the dissertation are:

- Make use of the entire computational model of visual attention, including the IOR
- Analyze the performance of the Itti-Koch model of visual attention for detecting objects of interest
- Formalize criteria for evaluating and comparing image databases

  - Evaluate against a variety of publicly-available image databases (with ground truth)
  - Include image databases that were not necessarily constructed with salient regions in mind, but that identify semantically-meaningful objects within images

- Provide results that can be used by a seed-based region-growing method as starting points for growing regions of interest from which features can be extracted for a variety of applications (including clustering and CBIR).

4.2 Methodology

In order to fulfill the objectives of this portion of the dissertation, the following steps are performed:

- **Select dataset**: instead of using a proprietary, manually-generated dataset, a survey of publicly available datasets is performed (Section 4.4). A subset of these datasets were selected for use in experiments based on the developed evaluation criteria.

- **Compute points of attention**: the Itti-Koch computational model of visual attention was used to compute points of attention for all images

- **Evaluate the points of attention**: compare the locations of the points of attention to regions in the ground truth to evaluate the performance of the computational model of visual attention in detecting objects of interest

- **Cluster points of attention**: group multiple points of attention together resulting in seed points

- **Evaluate seed points**: compare the predicted seed points (the centroids of the post-processed clusters of points of attention) to regions in the ground truth

4.3 Design

In the overall project, a content-based method of retrieving images based on their salient regions of interest, not on their global properties, is proposed.

Figure 4.1 summarizes the proposed framework. There are three main phases. First, a computational model of visual attention is used to model early vision. Its output are points of fixation which are intended to correspond to predicted salient regions in the image. The combination of these points and the original images are provided to a seed-based region-growing module which produces masks for each computed region. Finally, features are extracted from the areas of the original images distinguished by region masks.

The most notable difference between this model and that of Figure 3.2 is the lack of a clustering block, even though one could be added, if desired. However,

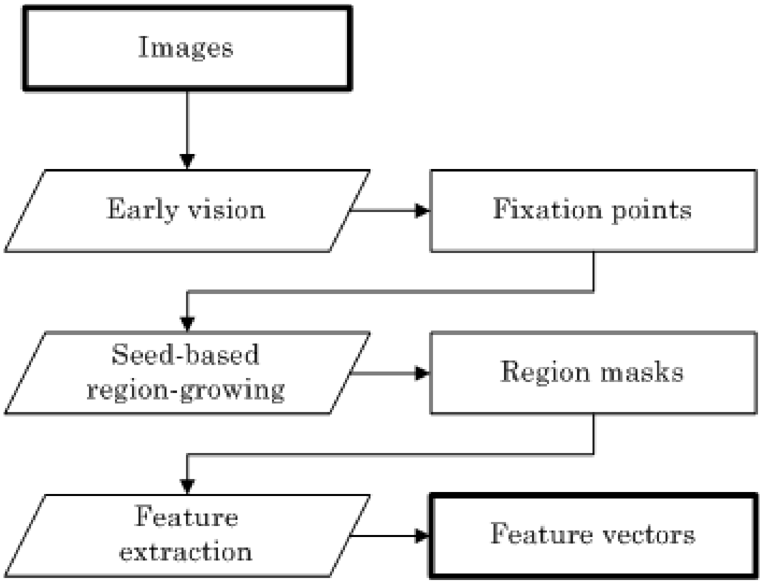


Figure 4.1: The proposed framework of the complete system

there are more subtle, but significant differences. This experiment uses the Itti-Koch computational model of visual attention in a different way. Instead of generating a single saliency map for each image, a new saliency map is generated for each sample (point of attention) until the predicted time occurrence of the next point of attention is later than the set threshold (please see Section 4.5 for details on these experiments). Instead of sampling several of the most salient points at the initial time (0 ms), as is the case when using only a single long-range saliency map, only the single most salient point at a given time is considered. The inhibition-of-return component of the Itti-Koch model, unused in Figure 3.2, becomes an integral component of the new model (Figure 4.1).

4.4 Dataset

This Section presents twenty-one image databases that are suitable for use in region-oriented CBIR.

There are several additional factors which motivate the creation of standard, readily available databases for image retrieval, beyond the need for a common database (although this is the most pressing need – researchers “often use completely different sets of images [...] making it hard to compare the performance of systems” [92]). The creation of ground truth is both time consuming and potentially prone to mistakes. Reusing an existing database makes results more readily comparable, saves time, and allows the use of a larger database than would be practical to manually create (if one is available).

4.4.1 Ground truth

CBIR holds the promise of making large-scale image retrieval fast and practical, but the creation of appropriately-sized databases with sufficient ground truth remains a daunting task requiring considerable time and effort. In this work an *image database* is a collection of images. In the worst case it is provided simply as images in a flat file structure. A database that lacks appropriate *ground truth* cannot be used to evaluate CBIR systems. There are several ways ground truth information can be provided:

- **File naming pattern**: It is possible to embed information within the file name, e.g. naming images “airplane01”, “airplane02”, ”car01”, etc. This is sufficient for classification applications.

- **Single-level directories**: Many image collections (such as Corel) divide images into folders, with each folder having a semantically-meaningful label. For example, a folder named “airplanes” will only contain images of airplanes. This is also appropriate for image classification and categorization.

- **Multi-level directory hierarchies**: Images can be organized into multiple, hierarchical, semantically-meaningful directories. In this case, a folder may be called “vehicles” and contain several subfolders such as “cars” and “airplanes” (which may contain further subfolders). This naturally lends itself to category-browsing CBIR interfaces.

- **Related images**: For each image a list of “related” images that should be retrieved in a query is provided. The definition of what constitutes a related image is left to the creator of the image dataset. An example of an image database that provides ground truth in this form are the UCID databases ([124, 125]). This is very useful for traditional, global CBIR, but not for region-oriented CBIR. The lack of information to classify images or identify specific objects makes databases that only provide this form of ground truth an inappropriate choice.

- **Object masks**: For each image one or more additional images are provided. These images (typically binary images) are masks for the objects within the original image. All pixels in the original image with the same coordinates as mask pixels in the mask image correspond to the target object.

- **Metadata**: This is the richest form of ground truth. A separate file (typically one per image) is provided. There is a wide variety of information this file may contain. It may contain the coordinates of polygons bounding objects within the image (a bounding box is a special case of a bounding polygon). Text annotation of the objects within the image may be provided instead or in addition to coordinates. Metadata may be provided in a custom format specific to the database or as XML. Additional information may be included within metadata (e.g. the GPS coordinates the picture was taken at, the names of people in the image, etc.).

Often, a combination of several types of ground truth is provided (e.g. a database may provide keywords describing images as metadata, organize the data into folders, and include object masks).

There have been several efforts to create standard image databases for CBIR (not specifically for region-based CBIR), but the adoption has not yet been widespread. Benchathlon [93] is not oriented towards retrieving images based on the objects contained within, although it was intended to make CBIR systems comparable with each other. Several workshops and conferences are also specifically interested in establishing and promoting benchmarks and evaluation criteria for CBIR, such as ImageCLEF [25], ImageEVAL [143], and TRECVID [131]. A discussion is available in [17].

4.4.2 Surveyed databases

A variety of databases were considered for this work. This evaluation is summarized in Table 4.1. Brief notes on each of the databases follow:

- **Caltech [37]**: consists of mostly vehicles such as cars, planes, and motorcycles. Some of the images do not contain objects of interest.
- **Caltech 101 [36]**: images are divided into 101 semantic categories, although individual objects are not considered. The categories are not evenly distributed (the largest refers to 800 images, whereas the smallest has only 31 images).
- **FAU Salient [87]**: a database of 1471 images consisting mostly of signs, sports balls, and other salient objects. Pictures were taken both indoors and outdoors at different times of day. Most of the photos were designed so that the image contains a single regions of interest, although a variety of distractors are present throughtout. Additionally, there is a subset of images with multiple target regions of interest.

- **LabelMe [119]**: a massive database of manually-labeled objects. The LabelMe database is an order or magnitude larger than any other database considered. Interactive tools allow human users to manually draw polygons around regions or objects in the image (e.g. both a road and a car may be annotated). The database accepts submitted annotation, and, as a result, continues to become more complete.

- **MIT-CSAIL [144]**: a large image database, but only a small fraction of the images are labeled. The object class sizes ranges from as small as 1 to as large as 693. In addition to the 107 object classes provided in the ground truth, 18 region classes, such as "floor" or "sky" are also distinguised.

- **MSRC OCV1 [127]**: a small, fully-annotated database of images with ground truth provided as manually-generated pixel-precise masks. Categories include bicycles, cars, cows, airplanes, people, and several outdoor scenes without objects of interest.

- **MSRC OCV2 [127]**: similar to MSRC OCV1, but over twice as large.

- **MSRC ORID [127]**: thousands of images are grouped into semantic categories, although no object-specific annotation is provided.

- **Renninger [117]**: images are grouped into semantic categories. The images are general scenes without consideration of the specific objects within them, making the database more suitable for CBIR based on global characteristics.

- **Simplicity1000 [151]**: images are grouped into semantic categories, although no object-specific annotation is provided.
- **Simplicity10000 [151]**: this database shares the same characteristics as Simplicity1000, except it is ten times as large and has ten times as many classes.
- **STIMautobahn [60]**: a small set of images with salient objects (road signs and markers). Ground truth is provided as manually-generated pixel-wise masks.
- **STIMcoke [60]**: a small set of images with a salient soda can in each one. Ground truth is as in STIMautobahn.
- **TU Darmstadt [70]**: the database provides three object categories: cars, cows, and motorcycles. The database is completely annotated except for one image.
- **TU Graz-02 [101]**: four object categories are provided, although one consists of images with no objects or regions of interest.
- **VOC2005 1 [34]**: the PASCAL Object Recognition Database Collection [109] is an effort to standardize the annotation (ground truth) of image databases. An annual competition is held to compare various image retrieval systems. Many of the considered databases are part of the collection. Because the

annotation is uniform across the databases, databases that are part of this collection may be preferred. The databases in this survey that are part of the PASCAL Object Recognition Database Collection are Caltech, Caltech 101, MIT-CSAIL, TU Darmstadt, TU Graz-02, and the VOC Challenge databases. In this particular database the images have been divided into four classes: bicycles, cars, motorcycles, and people. Images in the database have been compiled from other image databases cited in this list.

- **VOC2005 2 [34]**: a second database for the VOC2005 challenge. It has the same characteristics as the first VOC2005 database.

- **VOC2006 Trainval [33]**: the images in this database are from flickr [155] and Microsoft Research Cambridge [127]. Objects may fall into one of ten object classes: bicycles, buses, cats, cars, cows, dogs, horses, motorbikes, people, and sheep.

- **VOC2006 Test [33]**: a test database for the VOC2006 challenge. It has the same characteristics as the first VOC2006 database.

- **VOC2007 Trainval [32]**: the database for the 2007 VOC competition is similar to the one from the previous years, except that the number of images has increased, as has the number of classes making the database more challenging. Annotation data is now in XML rather than text files.

- **VOC2007 Test [32]**: a test database for the VOC2007 challenge. It has the same characteristics as the first VOC2007 database.

Table 4.1 shows a summary of the twenty-one databases considered. The truncated mean is the arithmetic mean except the two largest and two smallest values are removed from consideration (and thus the total number of values considered is reduced by four). This was done to stop extremely large (or small) outlying values from improperly skewing the results – those from the LabelMe [119], MIT-CSAIL [144], STIMautobahn [60], and STIMcoke [60] databases.

4.4.3 Evaluation criteria

There are a variety of criteria that must be taken into account when considering an image database. They are summarized in the following subsections.

- **Scope**: The selection of a dataset, particularly for content-based image retrieval, is critical. The dataset is an individual image database or an aggregation of several image databases selected to evaluate a system. It defines the scope of the retrieval algorithms to be developed. A database that is too narrow in scope may prevent the created methods from being extended to more general tasks, while a database that is too broad may make the retrieval problem intractable [132].

- **Database size**: There is a wide diversity of database sizes used in CBIR literature. Databases as small as dozens of images have been used in CBIR

Table 4.1: Comparison of image databases

Name	Images	Labeled	Annotation	Objects	Classes	Density
Caltech [37]	5775	4620	Box	1293	6	0.28
Caltech 101 [36]	9197	9197	None		101	
FAU Salient [87]	1471	1471	None		12	
LabelMe [119]	160569	41221	Polygon	Many	Many	
MIT-CSAIL [144]	72000	2873	Polygon	10358	Many	3.61
MSRC OCV1 [127]	240	240	Mask		9	
MSRC OCV2 [127]	591	591	Mask		23	
MSRC ORID [127]	4322	4322	None		33	
Renninger [117]	994	994	None		10	
Simplicity1000 [151]	1000	1000	None		10	
Simplicity10000 [151]	10000	10000	None		100	
STIMautobahn [60]	90	90	Mask	176	1	1.98
STIMcoke [60]	104	104	Mask	104	1	1.00
TU Darmstadt [70]	327	326	Box	336	3	1.03
TU Graz-02 [101]	1476	1280	Mask	1816	4	1.42
VOC2005 1 [34]	1316	1316	Box	1716	4	1.30
VOC2005 2 [34]	659	659	Box	1375	4	2.09
VOC2006 Trainval [33]	2618	2618	Box	5455	10	2.08
VOC2006 Test [33]	2686	2686	Box	5598	10	2.08
VOC2007 Trainval [32]	5011	5011	Box	15662	20	3.13
VOC2007 Test [32]	4952	4952	Box	14976	20	3.02
Arithmetic mean	13590.48	4551.00		4905.58	20.05	1.92
Truncated mean	6565.21	2855.79		4310.10	16.41	1.91

tests, while ones as large as half a million images have recently been experimented with [153]. The Internet is the largest collection of digital images ever assembled (publicly). The amount of images on the Internet is staggering. For example, the Picsearch image search engine claims to have indexed over 2 *billion* pictures from the Internet (as of January 2008) [114]. As previously stated, the image retrieval task must be well-bounded enough to be tractable, making extremely large databases ill-suited for experimentation. In other words, developing a CBIR system which uses a dataset as large, varied, and unbounded as the Internet is a task which is beyond the grasp of current techniques. Instead, one may either reduce the size of the database or narrow its scope. Databases may be classified as personal, domain-specific, enterprise, archive, or Web, with each classification generally increasing in size and scope [17].

Table 4.1 shows that the truncated mean (excluding the two largest and two smallest outliers) of the image databases is approximately 6565 images.

- **Number of labeled images**: Ideally, every image in the database will have been manually annotated, enabling a wide variety of experiments and evaluation to be performed on the entire database. Of course, the ideal case is not always encountered. There are several options in the absence of a fully-annotated database. One may choose to use a different database which is fully-annotated, if one exists. Alternatively, the experiments could be modified to not require full annotation. If full annotation is required, the annotated subset of the database could be used alone for experiments, although this may result in a test database which is too small.

Table 4.1 shows the number of images which have some sort of associated annotation (denoted by the “labeled” column). This annotation may be a general class for the entire image, labels given to one or more objects/regions of interest in the image, or a list or related images (correct responses in a content-based query). While the smaller databases tend to be fully-annotated, the largest ones only annotate a fraction of their images.

- **Annotation type**: There are a variety of ways an image database may be annotated, if at all. Annotation is one component of the entire ground truth the database may provide. The type of annotation a database provides affects the retrieval algorithms which can be tested using that particular database. Databases containing images classified only on a global level of granularity (e.g. a database divided into image categories) and without specific object-based annotation are given the label “None” in the *Annotation* column in 4.1. Databases which provide a bounding box associated with an object’s label (e.g. two sets of coordinates) are labeled as “Box” in the same column. An example of an image with bounding box annotation is shown in Figure 4.2.

More detailed than a bounding box is a database labeled with more than two coordinates per image, resulting in a bounding "Polygon" rather than a bounding box, as shown in Figure 4.3. The most precise annotation is a "Mask" of each image in the database. An example of an image and a mask for an object within the image is shown in Figure 4.4.

In the case that a CBIR application required a database providing annotation in the form of a bounding box. databases providing polygon or mask annotation can also be used by enclosing those regions within a bounding box.

- **Density**: Density, $D(d)$, is the average number of annotated objects in each image in the particular database. In other words, it is the total number of annotated objects in the database divided by the total number of annotated images in the database (Equation 4.1).

$$D(d) = \frac{\sum_{i=0}^{n} O(i)}{\sum_{i=0}^{n} A(i)} \tag{4.1}$$

$$O(i) = \texttt{The number of objects in image } i \tag{4.2}$$

$$A(i) = \begin{cases} 1, & O(i) > 0 \\ 0, & \text{else} \end{cases} \tag{4.3}$$

Equation 4.1 takes d, the database, as its only parameter. For each image i in the database (a database contains n images) it counts the number of objects in the image (Equation 4.2). This is divided by the total number of images that contain objects, as computed by the sum of all calls to Equation 4.3.

- **Number of annotated objects**: An image may have no annotated objects, one annotated object, or multiple annotated objects. If exactly one object in each image has been annotated the number of annotated objects will be the same as the number of images in the database. In this case the density of the database is exactly 1.00. However, if a database consisting of 100 images contains a single image with 100 annotated objects and the remaining 99 images lack any annotated objects the evaluation of this database will be skewed. While, in this case, the number of annotated objects is equivalent to that of a database with 100 images, each containing a single object, the density (100.00) would be considerably different.

  The number of annotated objects in the database can be higher than the number of images in the database if more than one object, on average, has been annotated in each image. It may be lower than the number of images in the database if annotation is incomplete (i.e. less than one object per image has been annotated).

  The truncated mean of the number of annotated objects in the considered image databases is 4310.10 (Table 4.1), indicating that most images have at least one object in them. In actuality, the percentage of images with objects in them is lower as many images contain more than one object.

- **Number of classes**: Images in a database are divided into distinct *classes*. These classes can be used to evaluate queries using metrics such as precision and recall. It is the number of different categories for images (if only global labeling is given) or the number of different objects in the database (if available). This is different than *objects*, in that an image may contain a lion, a tiger, and a bear (three objects), but still fall under a single class ("animals").

  Image or object classes may vary from extremely broad (e.g. indoor, outdoor), to general (e.g. people, sports, vehicles), to more narrow categories (e.g. boats, planes, cars). The greater the number of image/object classes, the less potential distance exists between classes, and the more difficult the retrieval task becomes. Additionally, one should note the relative size of each class within a database. Some database provide classes which are equally sized (e.g. *Simplicity1000* consists of 10 classes, each of which contains 100 images). Others have non-uniform class sizes (e.g. classes in *MIT-CSAIL* range from 1 to 693 images).

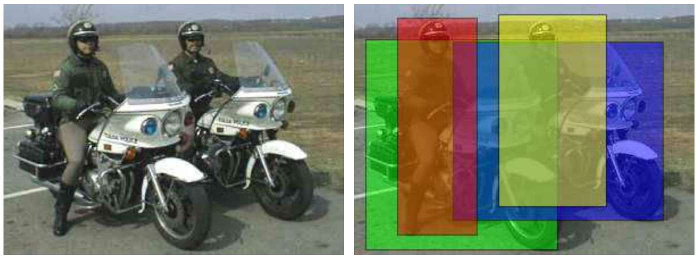


Figure 4.2: Ground truth bounding boxes (original image from *VOC 2005*)

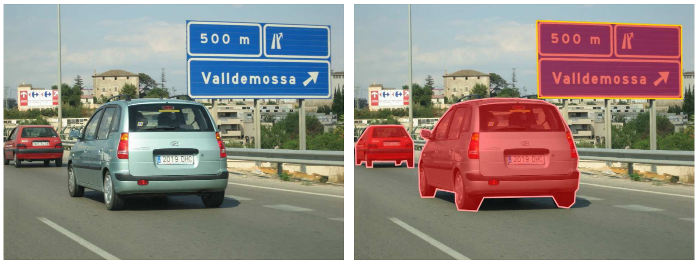


Figure 4.3: Ground truth polygons (original image from *LabelMe*)

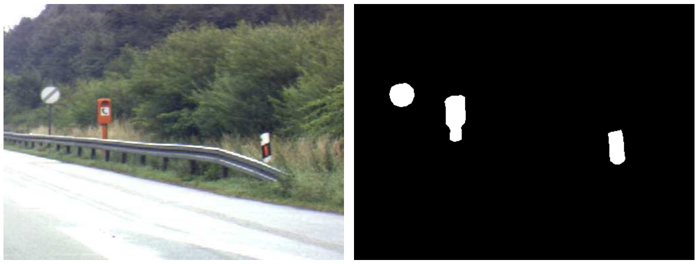


Figure 4.4: An image and its object masks (images from *STIM Autobahn*)

4.4.4 Discussion

This Section has defined several parameters for the selection of image databases in the context of region-based CBIR. In summary, the most important requirements for dataset selection are:

- **Database size**: is a small database sufficient to prove the validity of experiments? Will the proposed method be able to adequately scale to large databases (if necessary)?

- **Annotation**: does the database need to be fully-annotated? If so, are there restrictions on the type of annotation the database provides?

- **Density**: are there limits on the number of desired objects in each image in the database? For example, the proposed method may not account for multiple objects within a single image.

- **Classes**: are few or many semantic classes desired? Should these classes be broad or narrow?

Many CBIR applications must consider large databases, in this case leaving only LabelMe [119], MIT-CSAIL [144], and perhaps Simplicity10000 [151] for consideration. However, only LabelMe provides annotation for significantly more than 10000 images. LabelMe has been criticized for not providing more controls on the integrity of its data, which is user-contributed and is vulnerable to pollution [149].

For initial testing, the small databases may be more appropriate, as the quality of their ground truth (masks of the relevant objects) is of much higher quality than simply categories or even coarse bounding boxes. Still, one must be wary of distorted results when using small collections.

Researchers have more options than ever before with regards to using freely available (and thus, readily comparable) databases in their projects. Using the wrong dataset can be perilous to nascent research and has the potential to misrepresent results early on. Additionally, the database must be challenging enough to properly validate the research.

Projects which are ongoing (such as LabelMe [119]) improve with time, constantly adding new annotation from users. In this case, it would be desirable to have a fully annotated, non-changing subset of the LabelMe database to benchmark CBIR applications. Furthermore, it would be helpful for the maintainers of databases to standardize their formatting and ground truth in a format such as the one used by the participants of the PASCAL Object Recognition Database Collection [109].

4.4.5 Selected dataset

Eight different image databases comprise the dataset used in this experiment. The selected databases are summarized in Table 4.2. In this table the following metrics, a subset of those described in 4.4.3, are listed:

- **Images**: the number of images in the database.

Table 4.2: Properties of the selected image databases

Name	Images	Objects	Classes	Density
STIMcoke [60]	104	104	1	1.00
STIMautobahn [60]	90	178	1	1.98
VOC2005 1 [34]	1316	1716	4	1.30
VOC2005 2 [34]	659	1375	4	2.09
VOC2006 Trainval [33]	2618	5455	10	2.08
VOC2006 Test [33]	2686	5598	10	2.08
VOC2007 Trainval [32]	5011	15662	20	3.13
VOC2007 Test [32]	4952	14976	20	3.02
Arithmetic mean	2179.50	5633.00	8.75	2.09

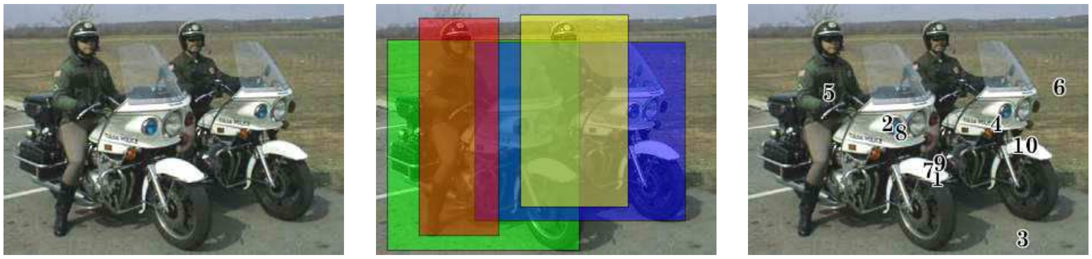
- **Objects**: the total number of annotated objects. Because each image has, on average, more than one annotated object, the number of annotated objects is greater than the number of annotated images.

- **Classes**: the number of different object categories in the dataset. The number of classes must be considered alongside the nature of those categories, not in isolation. Overlapping categories (e.g. “cows” and “farms”) are more difficult to distinguish between than disjoint categories (e.g. “dinosaurs” and “airplanes”). Several classes used in the databases in this work are “people”, “motorcycles”, and “cows”.

- **Density**: the total number of annotated objects in the database divided by the total number of annotated images in the database.

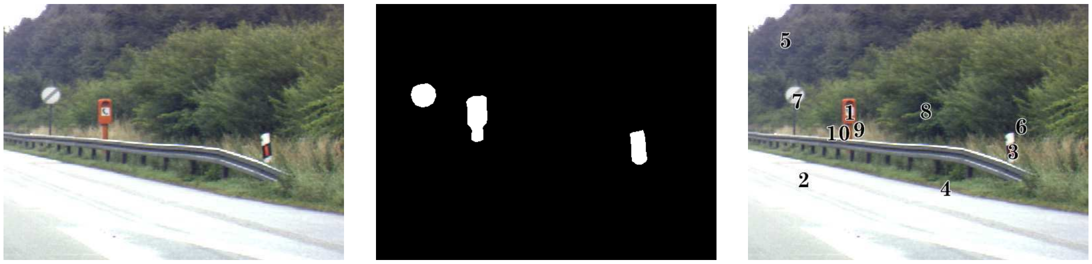
In order to be considered as a candidate for use in experiments a database must by fully-annotated. This annotation can be either bounding boxes surrounding the regions of interest (Figure 4.5 (b)), pixel-wise image masks (Figure 4.6 (b)), or polygons drawn around relevant objects in the image. In this work all ground truth annotation was either originally provided in bounding box form or converted to bounding boxes surrounding ground truth masks, for fairness in the results.

The STIMcoke database [60] was designed to have one salient region of interest (a soda can) within each image and was selected to provide a more controllable reference case (due to its small size, limited number of regions, and salient properties of the objects of interest). The PASCAL Visual Object Classes (VOC) [34] Challenge aims to classify objects in realistic scenes. An annual competition is held to compare various image retrieval systems. The selected databases contain images in categories such as people, cars, and bicycles. One benefit of using the VOC databases for experiments is that they are completely annotated. The VOC2005 databases are less complex (and challenging) than the ones used in the more recent challenges in every respect. The VOC2007 database is particular challenging given the number of objects and the variety of classes.



(a) (b) (c)

Figure 4.5: Ground truth bounding boxes and the calculated points of attention (original image from [34]). (a) is the original image, (b) the ground truth bounding boxes, and (c) the calculated points of attention.



(a) (b) (c)

Figure 4.6: Object masks and the calculated points of attention (original image from [60]). (a) is the original image, (b) the ground truth masks, and (c) the calculated points of attention.

4.5 Experiments and results

4.5.1 Points of attention

Points of attention were generated for each image using the Ezvision toolkit [61]. Ground truth was available for each image in the form of one or more bounding boxes for target objects and a label associated with each bounding box. Each generated point of attention has a set of coordinates, a predicted time of the fixation (in milliseconds), and an activation voltage (in millivolts). The higher this voltage is, the more salient is the attended location. There are two perspectives from which the results can be calculated, from the point of view of the points of attention of from the perspective of the ground truth regions. For the former (points of attention):

- **True positive**: this is the ideal case, when a predicted point of attention falls within a ground truth bounding box for an object within the image. It is denoted as TP_{poa}. In Figure 4.7 points 2, 3, 4, and 5 are all true positives. Point 5 is counted only as a single true positive.

- **False positive**: this occurs when a point falls outside of any of the bounding boxes in an image, incorrectly identifying a region of the image as being an object. It is denoted as FP_{poa}. Point 1 in Figure 4.7 is a false positive as it is a predicted point of attention that is not within any bounding box.

False negatives and true negatives do not apply to points of attention.

For the latter (regions of interest):

- **True positive**: a region that is hit by a point of attention is a true positive. A region can only count once (subsequent hits are ignored). It is denoted as TP_{roi}. In Figure 4.7 regions a, b, and d are true positives as they have all been hit be points of attention.

- **False negative**: this error occurs when an ground truth object eludes all of the predicted points of attention and is never identified as a region for further inspection, and thus can never be recovered at a later stage of processing. It is denoted as FN_{roi}. In Figure 4.7 bounding box c is a false negative as it is never hit by a predicted point.

False positives and true negatives do not apply to regions of interest.

Additionally, the maximum number of potential false positives ($MaxFP_{poa}$), defined as the sum of the number of true positives and false positives (Equation 4.4).

$$MaxFP_{poa} = TP_{poa} + FP_{poa} \tag{4.4}$$

Samples were taken ever 100 ms between 100 ms and 10000 ms. Note that 10000 ms is also the limit used in [58]. For each sample TP_{poa}, TP_{roi}, FP_{poa}, and FN_{roi} were recorded, based on where the points of attention landed relative to the ground truth. Two metrics were then calculated:

- **Hit Rate**: the proportion (in percent) of true positives from the maximum number of possible true positives (Equation 4.5)

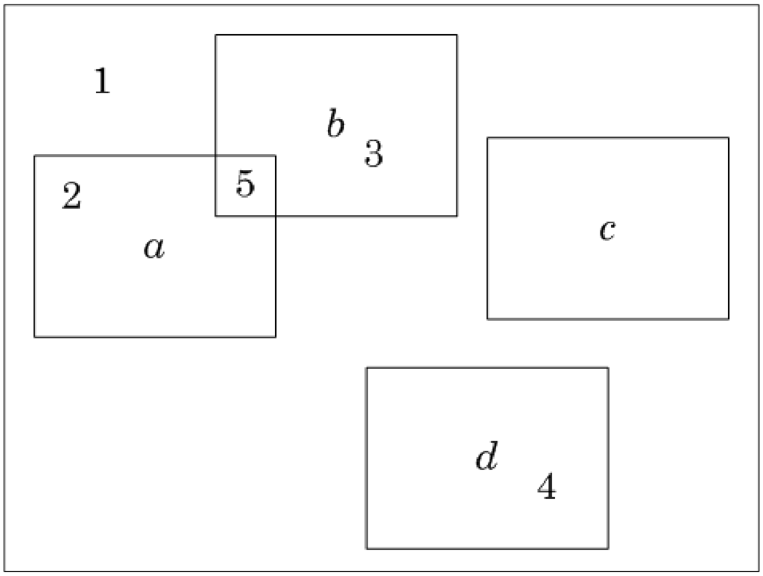


Figure 4.7: The evaluation methodology is illustrated in this figure. Numbers 1-5 represent predicted points of attention while letters *a* through *d* indicate ground truth regions of interest.

- **False Alarm Rate**: the proportion (in percent) of false positives from the maximum number of false positives (Equation 4.6)

The Hit Rate is defined as follows (it is equivalent to recall):

$$Recall = HitRate_{roi} = \frac{TP_{roi}}{TP_{roi} + FN_{roi}} \tag{4.5}$$

The False Alarm rate is defined as follows:

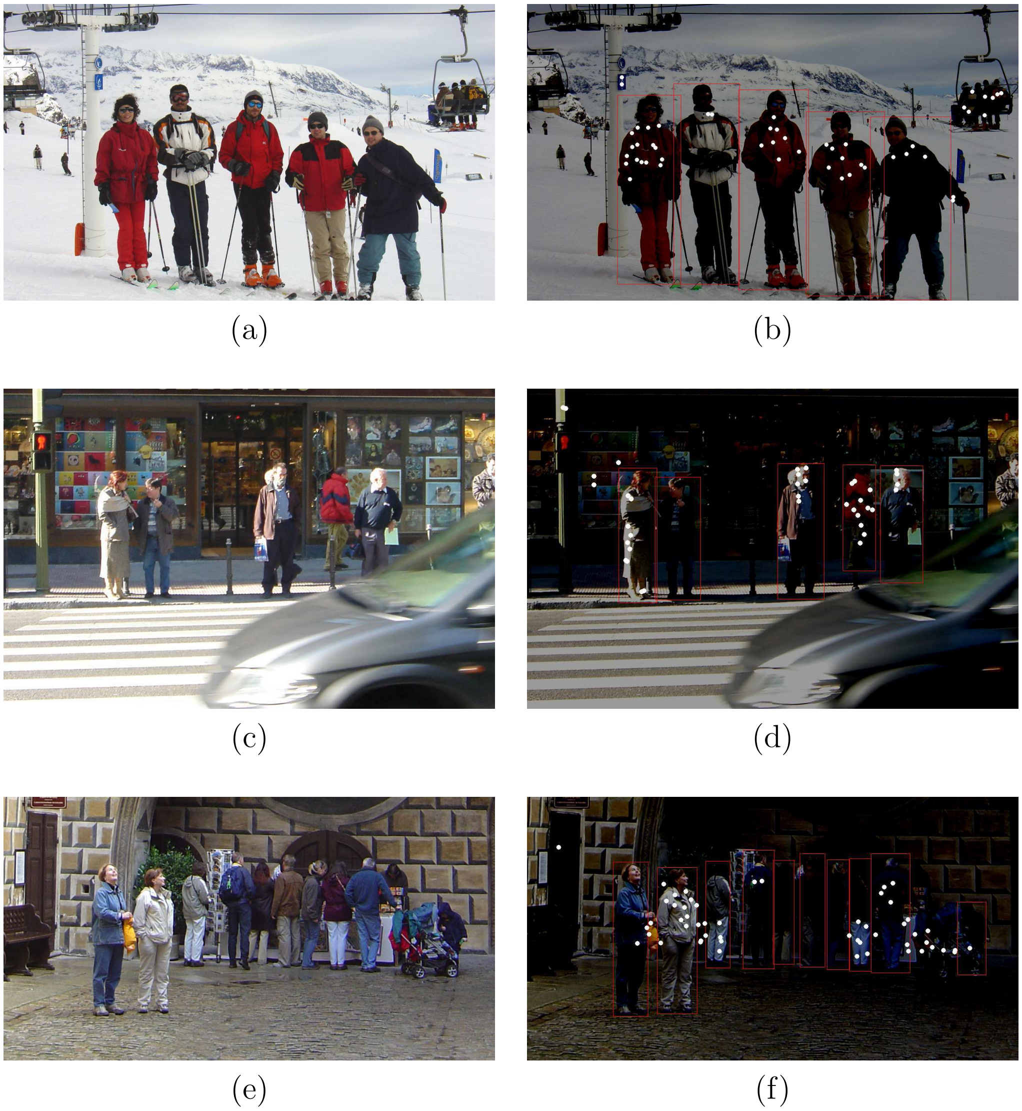
$$FalseAlarmRate_{poa} = \frac{FP_{poa}}{MaxFP_{poa}} \tag{4.6}$$

Additionally, Precision is defined as:

$$Precision = HitRate_{poa} = \frac{TP_{poa}}{MaxFP_{poa}} \tag{4.7}$$

Three illustrative sample results are provided in Figure 4.8, which shows three representative cases in ascending order of difficulty. Figure 4.8 (a) is a picture of people skiing. This is the easier sample case, with clearly-defined ground truth. However, note the imperfect ground truth in the rightmost person, or the missing annotation of the people in the ski lift. The white dots in Figure 4.8 (b) indicate the predicted points of attention. Figure 4.8 (c) is a more difficult case of people in front of a crosswalk. Note that the rightmost person is not included in the ground truth. Finally, Figure 4.8 (e) is the most difficult case. In this image it is not possible to identify all of the people using attention.

ROC curves for each of the 8 selected databases are shown in Figure 4.9. The corresponding Precision-Recall curves are shown in Figure 4.10. The results were obtained by varying the time parameter. Points predicted later than the varied time threshold were discarded. Thus, at low thresholds few points are considered, and at the maximum threshold all points are considered. In Figure 4.9 better performance is towards the top-left corner, whereas worse performance is towards the bottom-right. Figure 4.9 shows that certain objects are not salient and will *never* be detected by the plain computational model, no matter how long it is allowed to inspect the image. This is concerning, as objects that are not detected at this stage in our model cannot be recovered later on. For the six VOC databases between 17% and 25%



(a)

(b)

(c)

(d)

(e)

(f)

Figure 4.8: Sample images with ground truth regions of interest (red boxes) and points of attention (white dots) overlayed

of objects are not detected. Still, (for the VOC databases) this occurs at low False Alarm Rates. The figures also show diminishing returns. Performance improves quickly at first, then stalls as time passes. This indicates that minimal additional "value" is gained by running the model for long periods of time. Finally, there is a dramatic difference in performance between the six VOC datasets (which are harder as they do not contain salient-by-design objects) and the two STIM datasets. The false alarm rate is far higher for the STIM datasets. This is due to the nature of their ground truth. For example, STIMcoke only targets the soda can in each image, and performes exceptionally well at this, hitting over 96% of regions (the best of all datasets). However, other salient objects in the images are not included in the ground truth and count as misses. ROC and PR curves generated by varying voltage generally exhibit similar characteristics to the graphs generated by varying time, but are far less consistent, exhibiting large, unpredictable variations at high voltages. The activation voltage of the WTA neural network is not as reliable a parameter as time. As a result, time is used as the sole parameter in subsequent experiments.

Table 4.3 shows numerical results for the six VOC databases at the selected time cutoff (as determined by the ROC curves in Figure 4.9). Additionally, the density of the database and the average percentage of each image in the database occupied by ground truth regions is shown. The average time was 616.67ms. Points predicted after this time have less value than those before. In all cases, the $HitRate_{POA}$

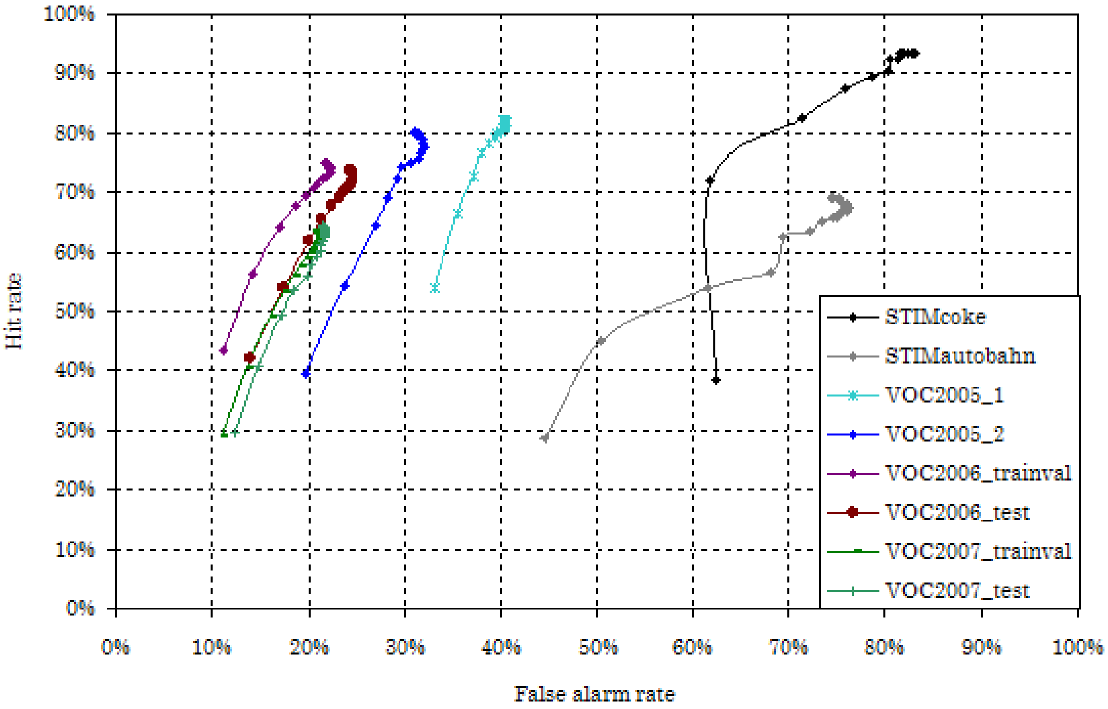


Figure 4.9: ROC curve generated by varying time

Table 4.3: Numerical results using points of attention

Database	Time	Hit_{POA}	$FalseAlarm_{POA}$	Hit_{ROI}	Density	Occupied
	(ms)	(%)	(%)	(%)		(%)
VOC2005 1	500	61.30	38.70	78.21	1.30	29.52
VOC2005 2	600	70.40	29.60	74.25	2.09	30.95
VOC2006 Trainval	700	79.13	20.87	71.51	2.08	37.47
VOC2006 Test	600	76.73	23.27	69.06	2.08	36.52
VOC2007 Trainval	700	79.09	20.91	59.31	3.13	36.71
VOC2007 Test	600	80.87	19.13	57.57	3.02	36.69
Mean	616.67	74.59	25.41	68.32	2.28	34.64

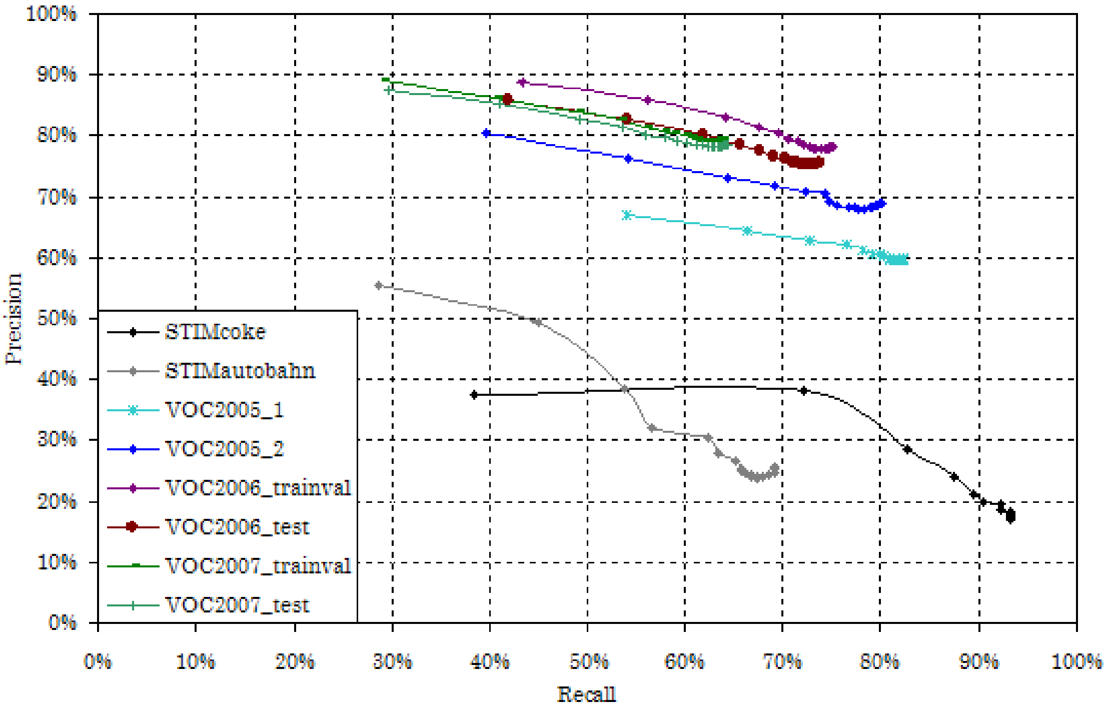


Figure 4.10: Precision-recall curve generated by varying time

is greater than the percentage of regions occupied by the ground truth. This shows that the model is working, as a randomly selected point would have a chance of hitting the ground truth equivalent to the amount of the image occupied by the ground truth (the rightmost column in Table 4.3. Finally, an inverse relationship between the $HitRate_{ROI}$ and density is seen. Density is a better indication of the difficult of detecting objects (and not the percentage of an image occupied by the ground truth). The greater the density, the lower the $HitRate_{ROI}$.

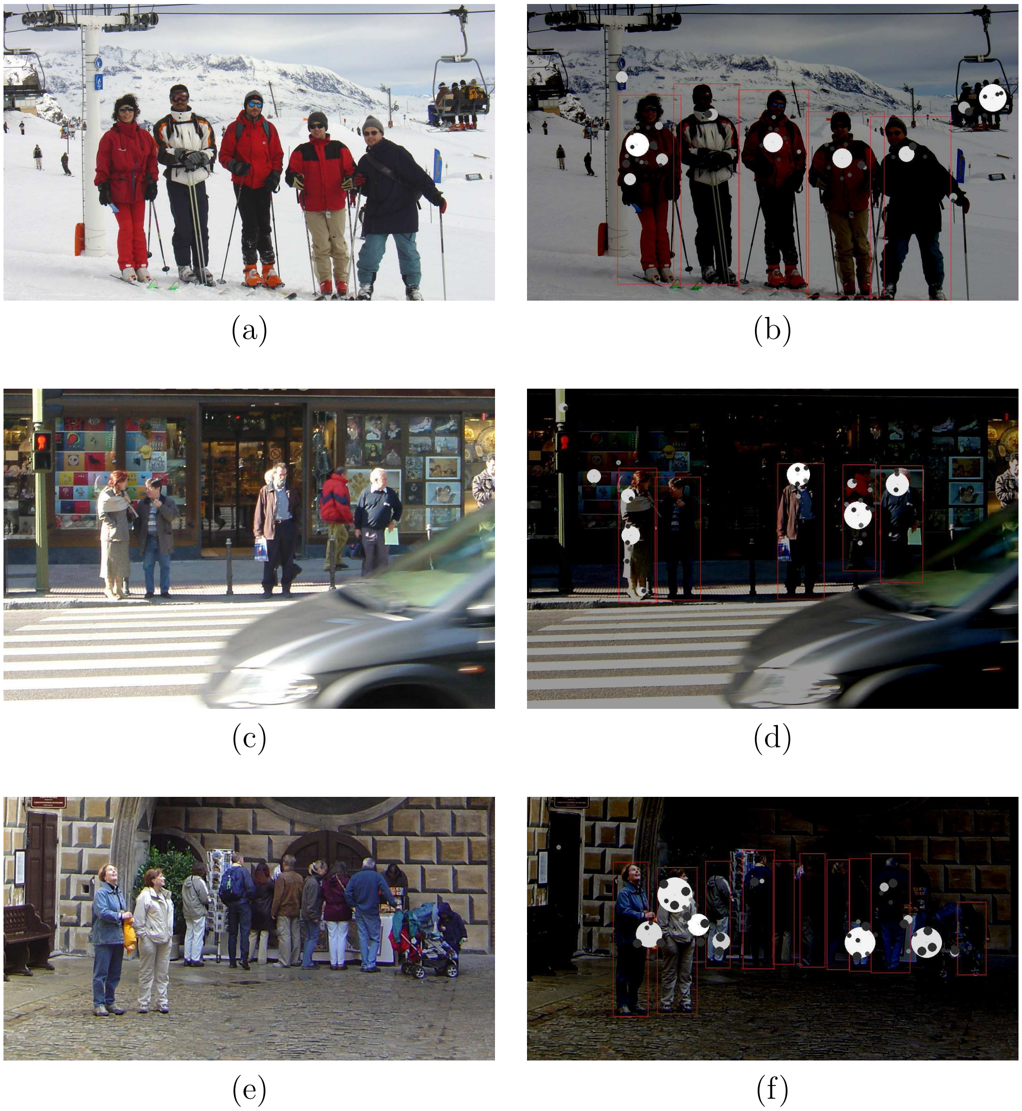
Table 4.4: Correlation coefficients for activation voltage and time

Database	All	$t \leq 600$ms
STIMcoke	-0.3644	-0.9685
STIMautobahn	-0.3688	-0.9759
VOC2005 1	-0.3789	-0.9689
VOC2005 2	-0.3786	-0.9792
VOC2006 Trainval	-0.3642	-0.9583
VOC2006 Test	-0.3619	-0.9581
VOC2007 Trainval	-0.3798	-0.9587
VOC2007 Test	-0.3741	-0.9571
Mean	-0.3713	-0.9656

4.5.2 Visualizing attention

It is helpful to visualize points of attention, their predicted time, and activation voltage. Figure 4.11 shows three sample images. Points of attention display two attributes. The brighter the point of attention, the earlier it was predicted. The larger the point of attention, the higher its activation voltage. These figures show that larger, bright, points are the first to hit objects, with subsequent points of attention revisiting the same location.

Figure 4.12 plots the average activation voltage for all points in a database predicted at a certain time, visually indicating the correlation between high voltages and low times. The mean correlation coefficient between activation voltage and time for all datasets is -0.3713. When only points at times up to 600ms are considered, the correlation increases to -0.9656, although both values indicate a high degree of correlation. Table 4.4 shows the correlation coefficients for all databases.



(a) (b)

(c) (d)

(e) (f)

Figure 4.11: Visualizing points of attention

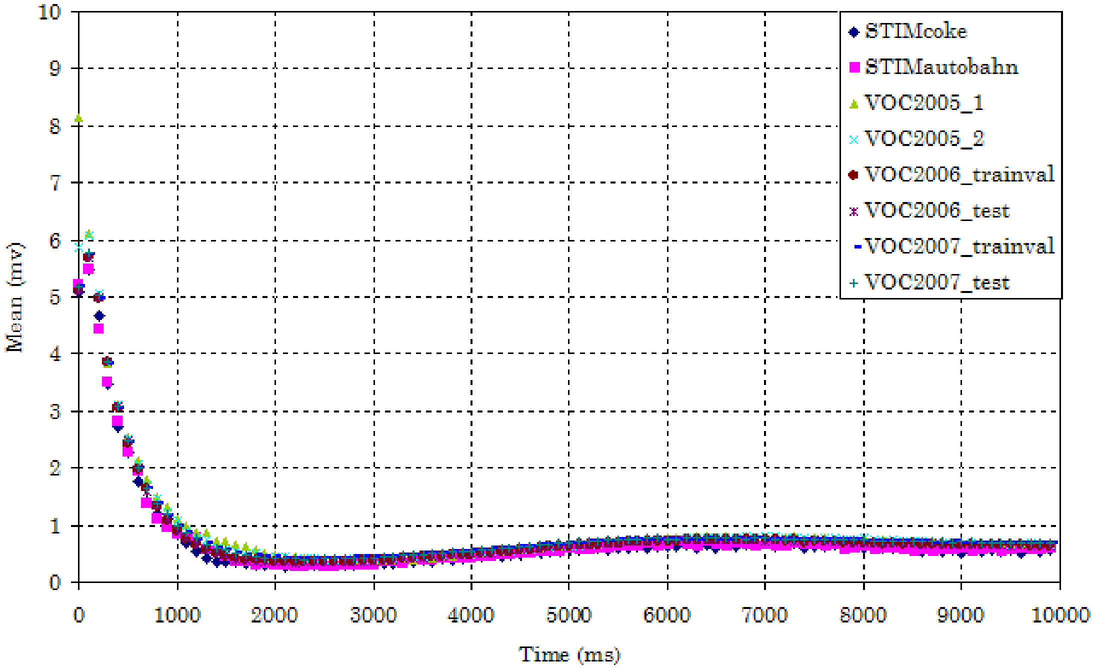


Figure 4.12: Mean voltage vs. time for the selected image databases

4.5.3 Seed points

Seed points are computed by clustering points of attention. The centroids of the resulting clusters are the seed points.

k-means clustering is used [82]. For k-means, k, the number of clusters, must be set. For each cluster an initial centroid is selected, possibly at random. The algorithm iterates for a certain number of times, readjusting cluster membership with each iteration (points are associated with the closest cluster centroid and the centroid is then recomputed).

In this work, k is set dynamically for each image. For each image, the points

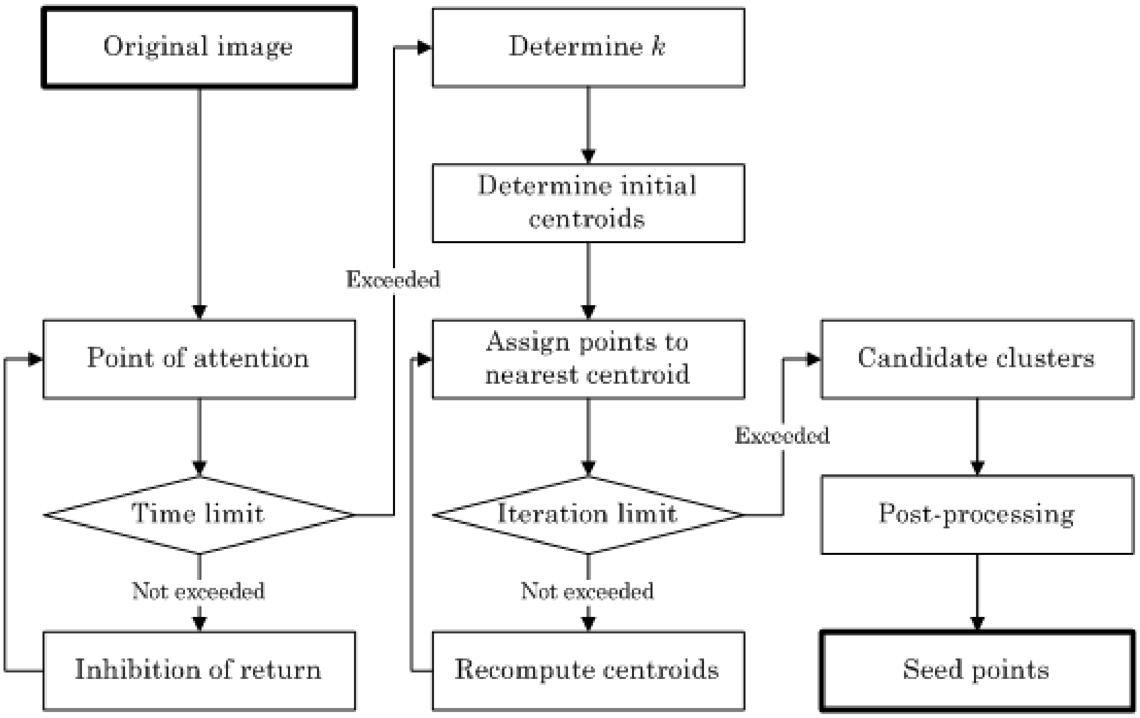


Figure 4.13: Clustering points of attention

of attention are predicted. k is then set to the number of points of attention before time t (determined empirically). If k is large (equivalent to the total number of points of attention), the clustering algorithm will converge to the results of the points of attention. This process is shown in Figure 4.13.

Once k has been set, the locations of the initial centroids must be selected. This can be done in one of four ways:

- **Random**: cluster centroids are assigned randomly
- **Seeded**: cluster centroids are seeded with the locations of the k points with the highest activation voltages

- **Seeded, trimmed**: cluster centroids are seeded with the locations of the k points with the highest activation voltages after points with low activation voltages and late predicted times have been removed

- **Seeded, pruned**: cluster centroids are seeded with the locations of the k points with the highest activation voltages. Subsequently, the number of points belonging to each cluster is calculated. Clusters with very few points are pruned and not included in the results. The cluster is pruned if the difference between the number of points in the cluster and half its standard deviation is negative

The ROC curve for seeded, pruned clusters for all databases is shown in Figure 4.14. It was obtained by varying the time limit threshold of points included in the clusters. A lower threshold results in fewer points of attention being clustered, and vice versa. The results correspond to those obtained for points of attention. Comparative results for the *VOC2005 2* database are shown in Figure 4.15. The curves in Figure 4.15 were obtained the same way as in Figure 4.14. This database was selected for illustration because it is representative of the results of the other databases, but exhibits slightly more differentiation between results. The graph plots five sets of results: points of attention ("normal"), clusters with random seeds, seeded clusters, seeded and trimmed clusters, and seeded and pruned clusters. Points of attention alone exhibit the best performance. Out of the clustering algorithms,

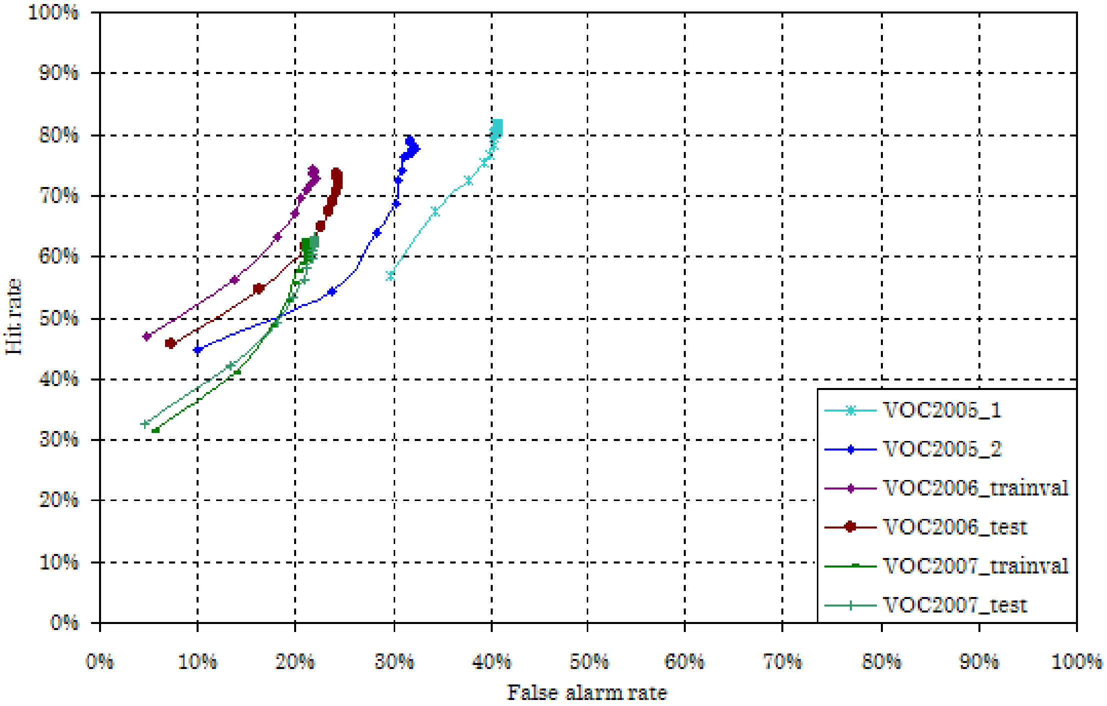


Figure 4.14: ROC curve for seeded, pruned clusters

seeded and pruned clusters perform the best, while randomly-selected cluster seeded perform the worst.

Representative results of clustering points of attention are shown in Figure 4.16. Clusters are illustrates as all points within a blue circle, while the blue point at the center is the centroid if the cluster. Note that these clusters are seeded and pruned. As a result, not every point of attention belongs to a cluster.

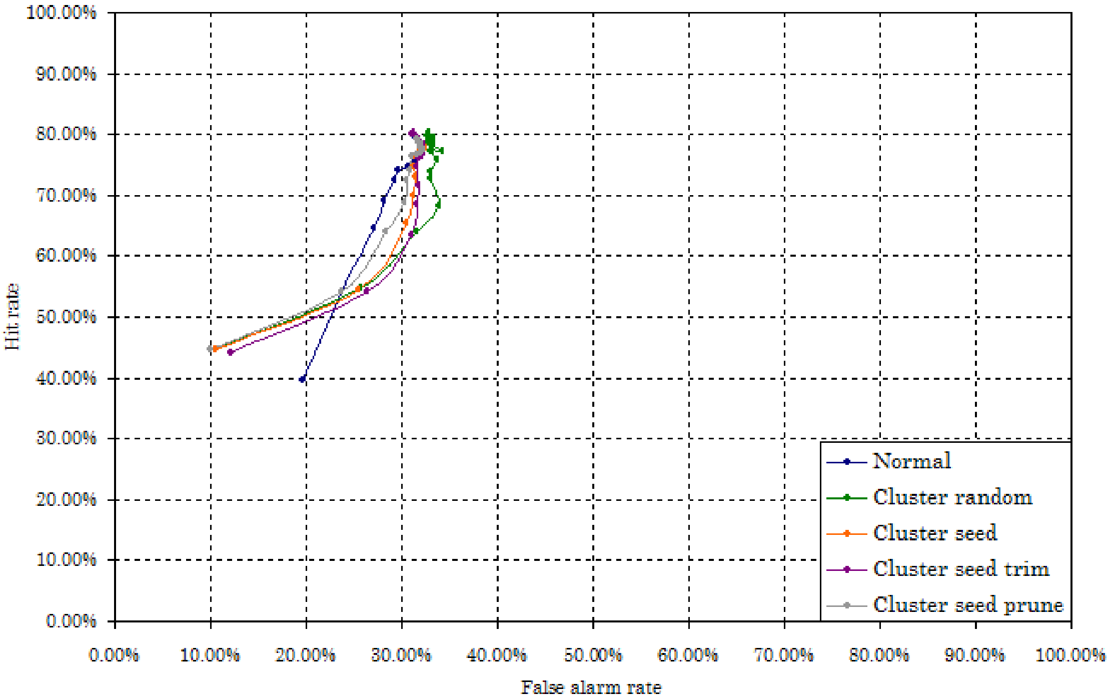
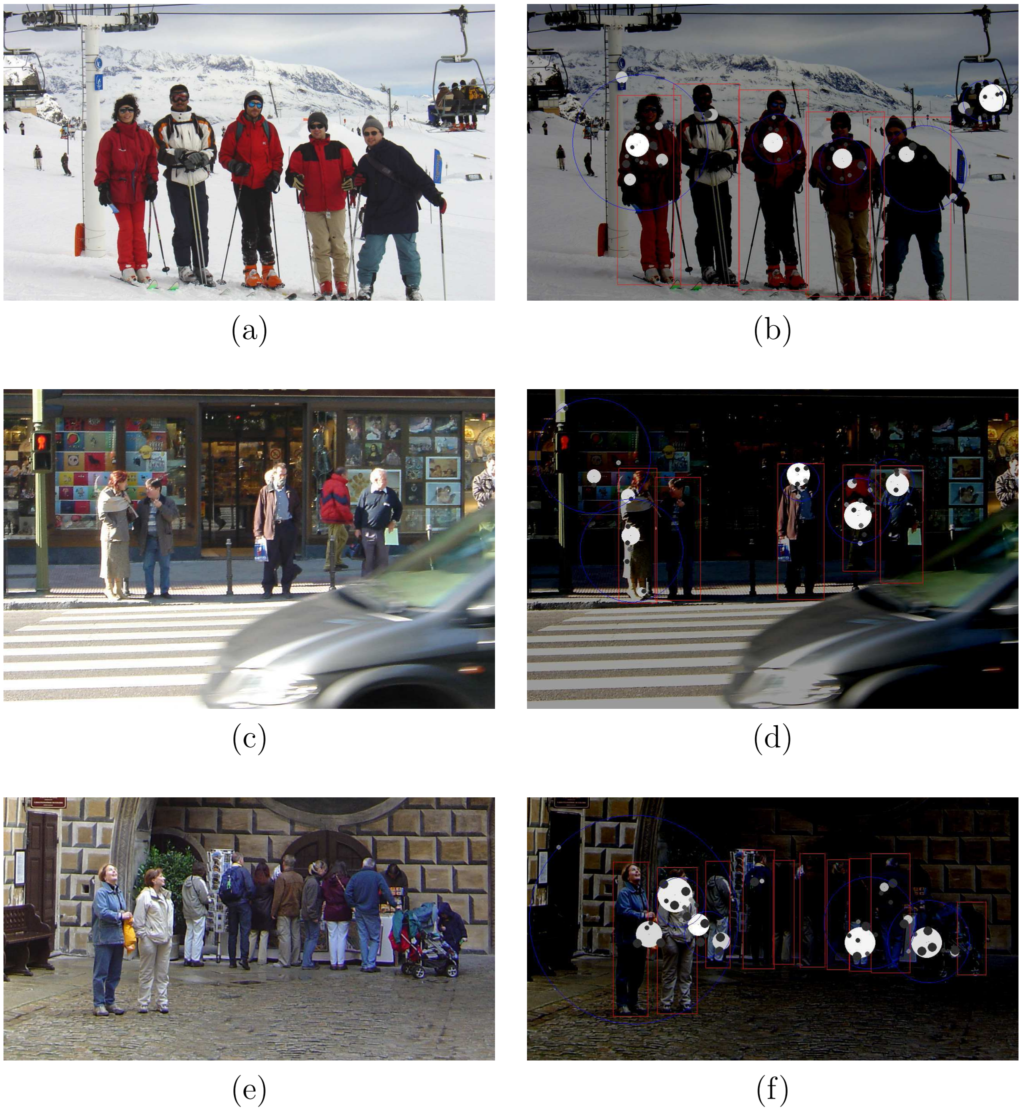


Figure 4.15: Clustering results for the *VOC2005 2* database

4.6 Discussion

The main objective of the experiments in this chapter was to determine seed points for used in a seed-based region-growing algorithm. A computational model of visual attention was used to generate points of attention. These points were then evaluated against a variety of publicly-available image databases in order to measure how well the predicted points of attention match the databases' ground truth. The points were clustered in order to generate seeds for region-growing.

Table 4.5 compares the performance of points of attention against the resulting cluster centroids. Clustering points is unlikely to improve the Hit Rate as


(a)

(b)

(c)

(d)

(e)

(f)

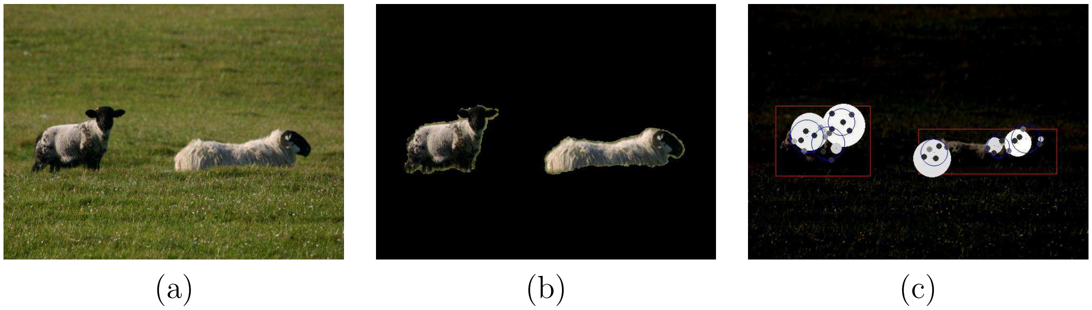
Figure 4.16: Sample images illustrating the computed clusters (blue circles) and their centroids

Table 4.5: Comparing points of attention to cluster centroids

Method	$HitRate_{POA}$	$FalseAlarmRate_{ROI}$	$HitRate_{ROI}$
Points of attention	74.55%	25.41%	68.32%
Cluster centroids	73.76%	26.24%	67.98%

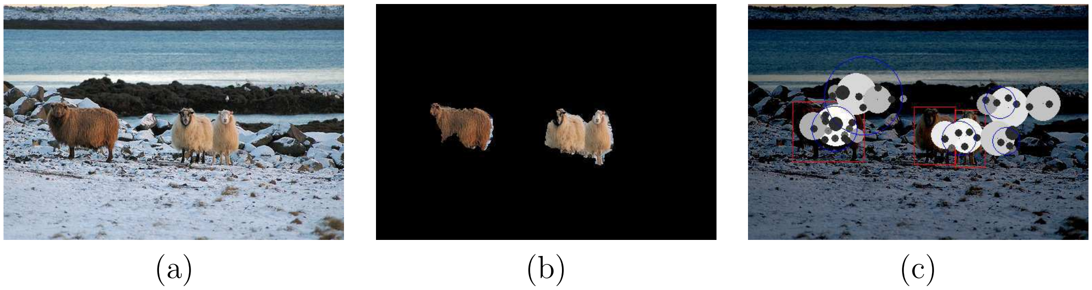
it reduces the number of points used in evaluation, although the decrease is small. Points of attention hit objects in the ground truth 74.55% of the time at a false alarm rate of 25.41%, whereas 68.32% of the objects are hit. Final cluster centroids hit objects in the ground truth 73.76% of the time at a false alarm rate of 26.24%, whereas 67.98% of the objects are hit. From approximately 40 points of attention generated per image, six cluster centroids result.

Sample results for region extraction (related, but beyond the scope of this work) are shown in Figures 4.17, 4.18, and 4.19. Each of the images shows successfully extracted regions. Figure 4.17 is the simplest of the three examples. The method successfully extracts two regions of interest, each corresponding to a sheep. Figure 4.18 is somewhat more difficult. Here, all meaningful objects are extracted, although the two rightmost, adjacent animals are considered to be a single object. Finally, Figure 4.19 is the most difficult case. Here, the cyclists in the foreground are correctly extracted, although those in the background are not salient and thus not extracted.



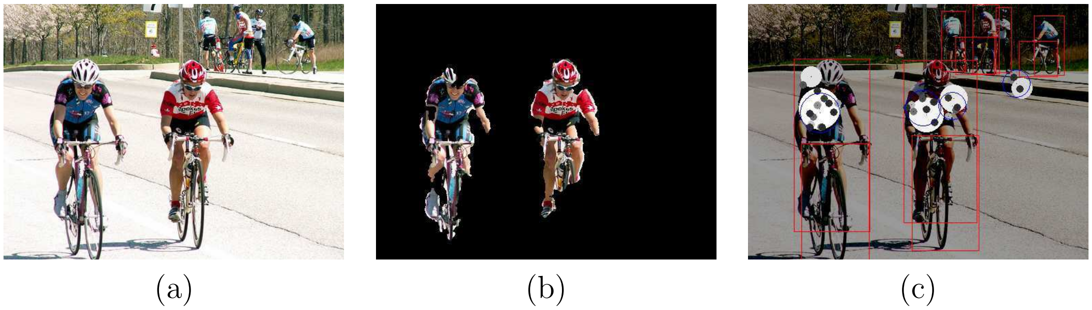
(a) (b) (c)

Figure 4.17: Region extraction results for the *sheep* image



(a) (b) (c)

Figure 4.18: Region extraction results for the *frozen* image



(a) (b) (c)

Figure 4.19: Region extraction results for the *cyclists* image

Chapter 5

PRISM: PERCEPTUALLY-RELEVANT IMAGE SEARCH MACHINE

A picture is worth a thousand words. An interface is worth a thousand pictures.

Ben Shneiderman, computer scientist, b. 1947

5.1 Introduction

This chapter presents a new interface for image retrieval, organization, and annotation. The interface allows queries to be based on image content, keywords, and collaborative filters. Furthermore, the interface allows the system to learn from user actions, improving results over time. This system is referred to as PRISM, the Perceptually-Relevant Image Search Machine.

In PRISM, image retrieval, organization, and annotation are accomplished through a unified set of interface features. For example, the system allows users to spatially organize individual images, and to separate groups of images into different tabs. This user-generated spatial organization of images is also used to compose content-based queries.

A key aspect of the interface is its ability to learn from user actions. Two of the retrieval methods, search by keyword and collaborative filtering, rely on user-provided information. By annotating and organizing their own query images, users contribute to the global performance of the system as well, improving the retrieval results for other users.

PRISM was designed from a *human-centered* perspective. *Human-centered computing* refers to the interaction between human users and the computational machines they use. The field "aims at tightly integrating human sciences (e.g. social and cognitive) and computer science (e.g. human-computer interaction (HCI), signal processing, machine learning, and ubiquitous computing) for the design of computing systems with a human focus from beginning to end" [62]. Human-centered computing examines the user, the task, and the machine as one unit [62].

This chapter presents the motivation (Section 5.2) behind the design of PRISM. Requirements for the interface from the perspective of the included retrieval methods are then discussed in Section 5.3. Section 5.4 presents the user interface.

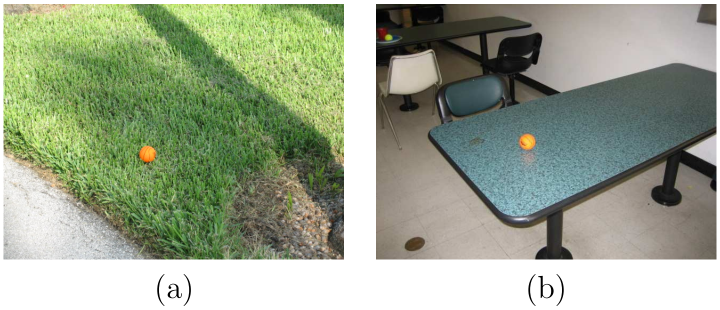
5.2 Motivation

The design of this system was motivated by the desire to create a user interface that can accommodate the attention-based image retrieval method described in Chapters 3 and extended in Chapter 4 of this dissertation.

Queries may either consists of a single example image (query by example – QBE) or multiple example images (query by multiple examples – QBME). QBE is the traditional image search paradigm employed by content-based systems (QBIC is one example of such a system [38]). In QBE, the user must procure an image that is representative of the ones they are seeking. One way to do this is to select an image from the image archive (perhaps starting by displaying images randomly until an appropriate image is found). In QBME, the user is allowed to provide multiple example images. An implementation that uses multiple example images has been proposed by Borba et al. [4]. QBME queries are more complex than QBE, but allow for greater query diversity and specificity. Commonalities between the query images can be extracted and used for retrieval.

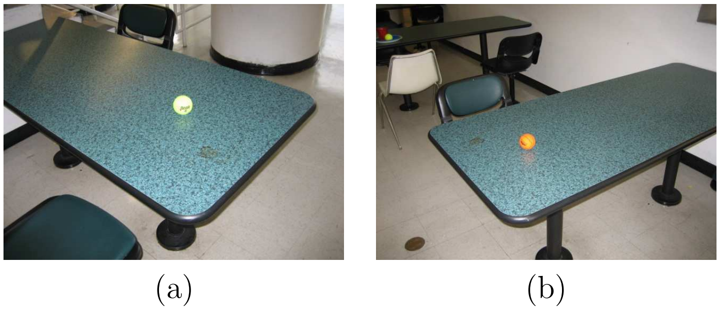
QBME can also be used to construct queries based on either local (regional) or global features. For example, Figure 5.1 shows two images with different global characteristics, but similar regions (the orange miniature basketball). Figure 5.2 shows two images that are similar globally, but contain different regions of interest (a tennis ball in Figure 5.2 (a) and a miniature basketball in Figure 5.2 (b)). By using multiple images, the degree of similarity between the region-based features and the degree of similarity between global features of all query images can be computed. Then, results can be based on the features (local or global) with the greatest degree of similarity.

When querying by multiple example images it is useful to be able to weigh



(a) (b)

Figure 5.1: Similar regions of interest



(a) (b)

Figure 5.2: Similar global features

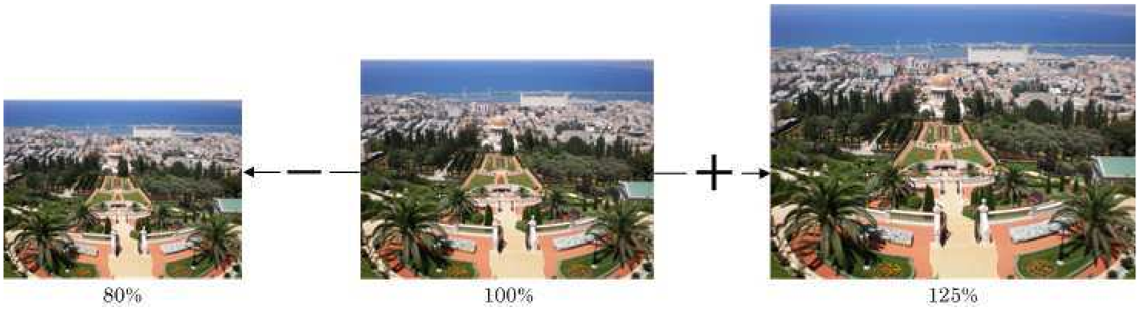
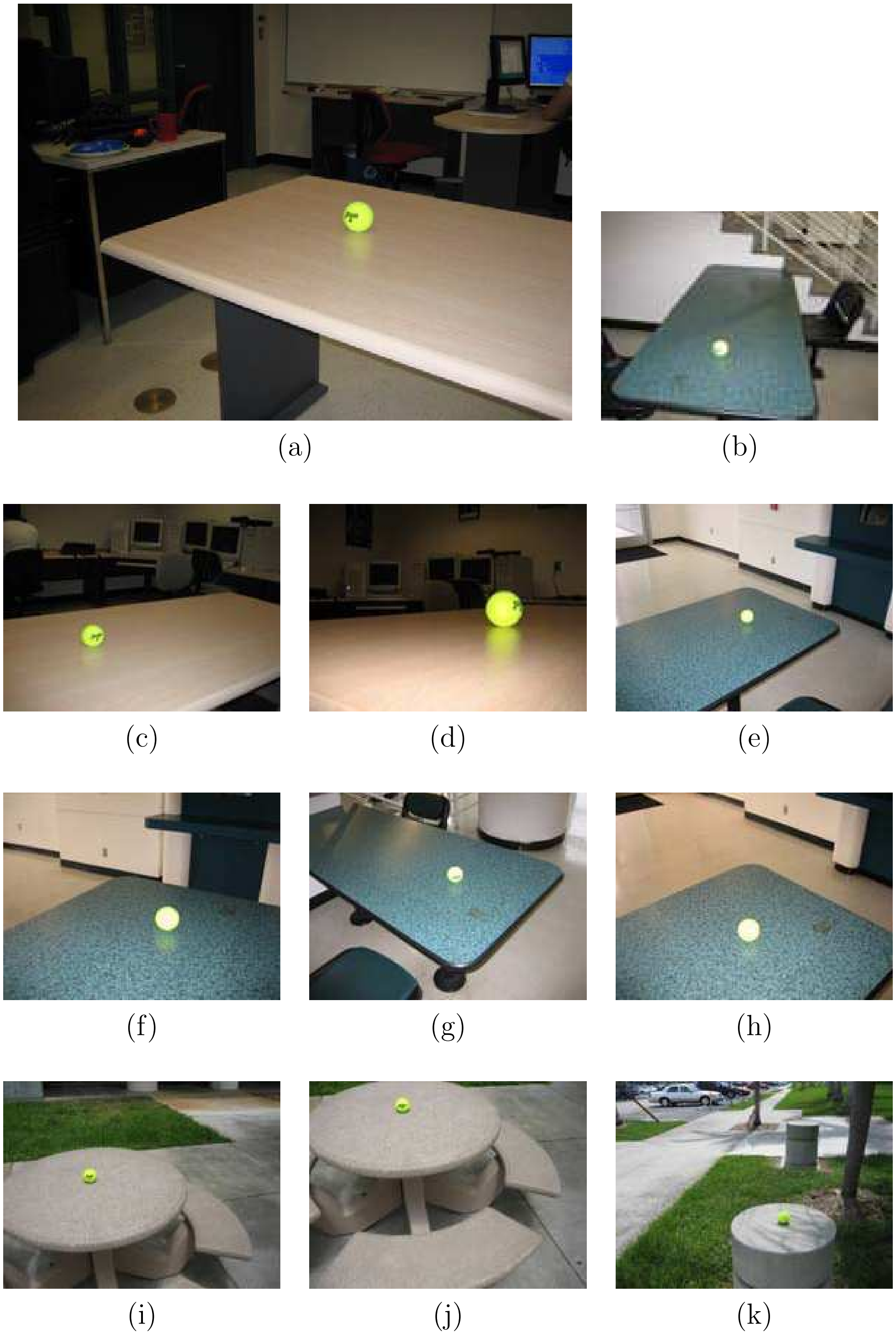


Figure 5.3: Shrinking and enlarging images for querying

images in order indicate each example image's relative importance to the query. This is accomplished by allowing the user to enlarge and shrink images, with larger images receiving more weight in the query. An example is shown in Figure 5.3. Figure 5.4 shows an example query. Two query images are provides, where Figure 5.4 (a) has been enlarged, indicating it is more important to the results than Figure 5.4 (b). The sample results reflect this, with the higher-ranked images (e.g. Figure 5.4 (c) and Figure 5.4 (d)) having similar global features to the more significant query image.

Figure 5.5 shows the integration of PRISM and a query by multiple example images CBIR system. PRISM consists of the user, the query images (multiple images, scaled by the user), and the retrieved images. The system's design allows the option of using query images to select between global and local features, either absolutely or on a sliding scale.

The result of being able to scale multiple images to indicate their importance in a query allows the user to quickly and easily compose expressive queries. By storing and aggregating the queries of multiple users the system is able to collect cues to be used in a collaborative filtering subsystem. When a user puts two images together in the same tab, they imply an association between those images. This association is even stronger if the images are overlapping. In the future if one image is retrieved in a separate query it is more likely that the other image with which it has a past relationship with should also be retrieved. Over time these associations


(a) (b)

(c) (d) (e)

(f) (g) (h)

(i) (j) (k)

Figure 5.4: Searching by multiple example images. (a) and (b) are the query images. Images (c) through (k) are the results.

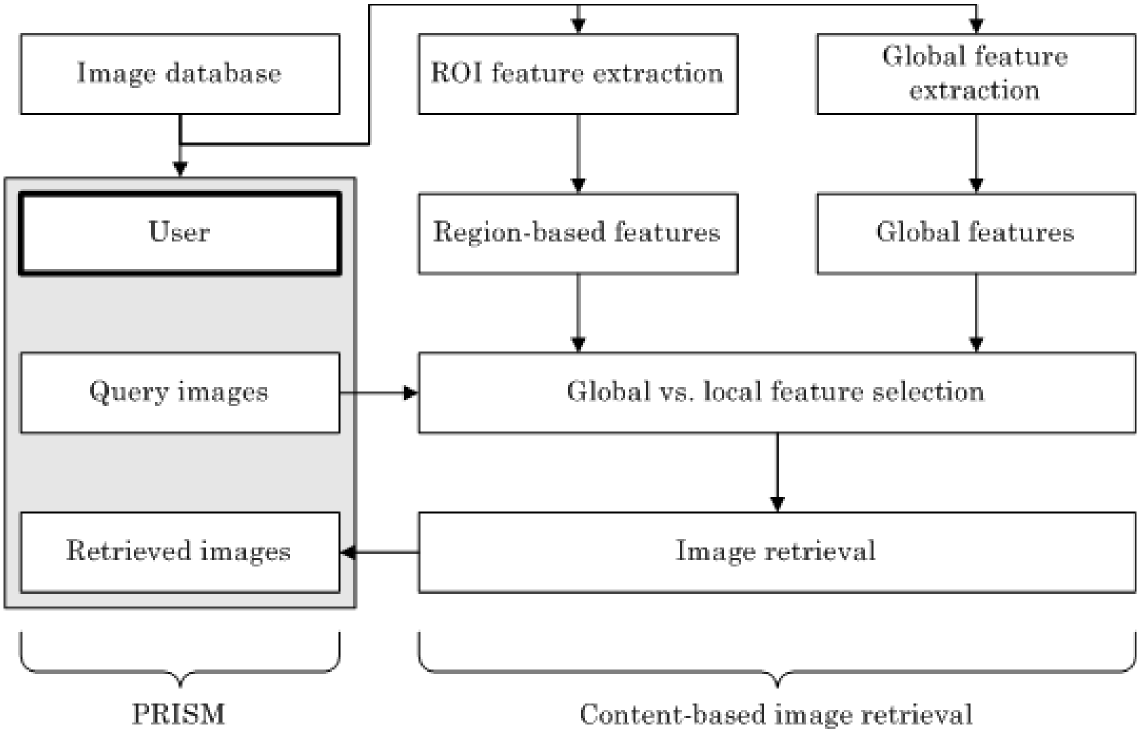


Figure 5.5: Integration of PRISM and a content-based image retrieval system

can be logged and used to improve the efficiency of results. Taken to the extreme, if complete information has been collected, content-based results could be discarded completely and the system will rely solely on information collected from users for retrieval. This collaborative information requires no extra action from the user other than the use of the system to compose queries – it is "free".

The system allows the user to associate keywords with images (Figure 5.6). It is more expensive to collect such annotation. Unlike collaborative filtering, text annotation must be explicitly specified by the user. While collaborative information is collected each time a user composes a query, text annotation is voluntary. The

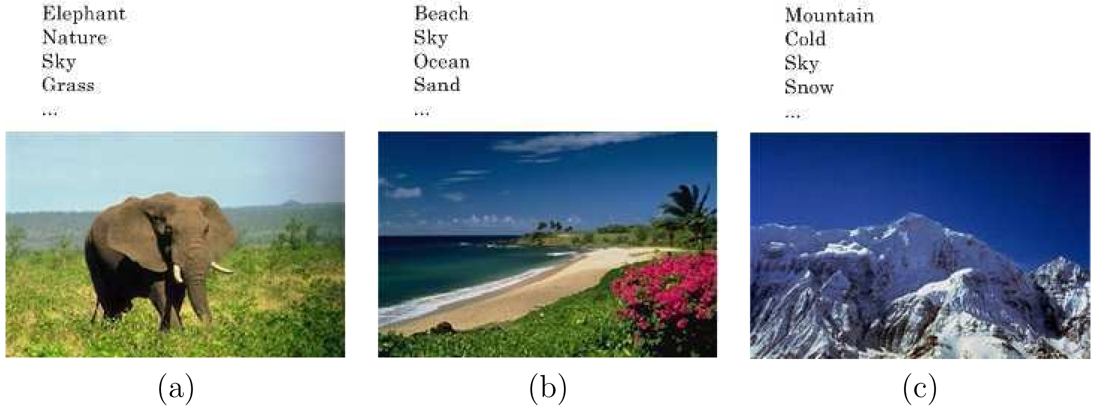


Figure 5.6: Associating keywords with images

size of an image is used to weigh the importance of the annotated text in the query (keywords associated with a larger image weigh more in the results).

The creation of a system providing not only image retrieval functions, but organization and annotation capabilities, as well was the objective of PRISM. Once it had been decided to allow for the composition of queries consisting of multiple example images scaled by the user, the addition of collaborative and annotation capabilities became natural, desirable features to add to the system.

5.3 Requirements

In order to construct a system that combines CBIR, text retrieval, and collaborative filtering in a modular way, a "glue" is needed. Each method must be able to function independently, in the absence of another, but be enhanced by other retrieval methods as well. In our system the glue is embodied by the user interface and the query it generates.

There are requirements for each of the three types of retrieval in our system:

- **Content-based image retrieval**
  - **Query method**: this may be either the direct specification of image features or the procurement of example material from which to extract those features. In systems where example material is provided either one or multiple images may be used. When providing multiple query images a method must be devised to weigh them. *In PRISM, the user selects a single or multiple example images from the image database.*
  - **Global vs. local features**: retrieval can be based on either global features (e.g. the gist of a scene) or local ones (e.g. the characteristics of an object of interest). A variety of feature extraction methods and similarity methods must be considered (please refer to Section 2.4.5 and Section 2.4.6, respectively). *PRISM allows the user to specify a global feature or local feature-based query by scaling multiple example images.*
- **Text retrieval**
  - **Query method**: text retrieval requires a way to input query text into the system. *In PRISM, individual images can be annotated using a separate text field associated with each image. Additionally, each tab can also be annotated, with its given annotation extending to all images. These annotations are used as the basis for the query.*

Table 5.1: Required interface features for each query subsystem in PRISM

Method	Multiple examples	Scale	Position	Text annotation
Content-based	Q	Q		
Collaborative filtering	Q, L		Q, L	
Keyword	Q	Q		Q, L

  - **Annotation**: the system must allow images to be annotated individually. *A text field is associated with each query image and each visible tab.*

- **Collaborative filtering**

  - **Query method**: at least a single image must be specified for querying *The same images specified for use in a content-based query are used to query the collaborative filtering system.*

  - **Association**: multiple images must be able to be associated with each other in order to learn the relationships between images. *PRISM allows multiple images to compose queries. When multiple images are used together in a query collaborative associations between those images are inferred and stored.*

Table 5.1 summarizes the requirements for implementing a single interface that allows the three selected query methods to be simultaneously implemented. In Table 5.1, *Q* indicates a feature which is required for composing a query, whereas

L indicates a feature which is needed for learning (annotation or inferring relationships). In the context of this discussion *learning* is synonymous to modifying metadata in the database through user actions (annotating images and weighing collaborative filters).

Several conclusions can be drawn from Table 5.1. Annotation is not available for content-based retrieval. Content-based retrieval uses unsupervised feature extraction, not user-provided information. Each query method relies on multiple example images for querying, although only collaborative filtering uses multiple example images for recording information from the human user. Scaling is used for querying but does not affect the features stored in the image database. Being able to position images is only needed if collaborative filtering is implemented. Similarly, being able to annotate images using text is only needed for keyword search and retrieval. Most significantly, by including these four user actions (multiple example images, scaling individual images, positioning images, and annotating images), querying and learning from the three selected methods can be accomplished through a single interface. Content-based and content-free retrieval are, by their nature, significantly different. However, they can be integrated by exploiting the commonality in their querying interfaces – by providing a way to query by multiple example images.

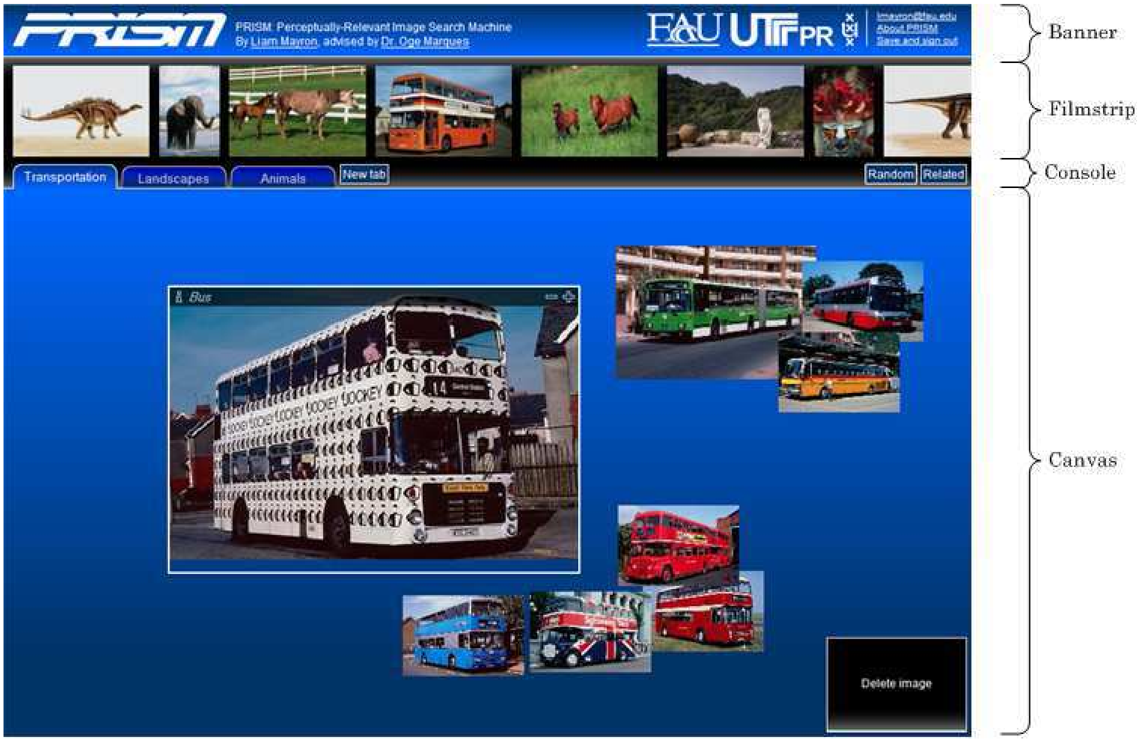


Figure 5.7: The PRISM interface in use

5.4 User interface

The PRISM client's user interface was designed to enable the user to construct expressive queries that meet the requirements defined in Section 5.3 through a set of intuitive, purpose-driven actions [89, 90].

The PRISM interface is shown in figure 5.7. In this Figure, the interface is already in use (the user has already organized several images). The user has created three tabs representing three broad categories ("transportation", "landscapes", and "animals"). The "transportation" tab is displayed. It has been populated by eight images featuring buses. Several images have been enlarged, with the largest being labeled “bus”. This will enable PRISM to search by the text “bus” next time the uses requests a “Related images” query. Furthermore, the user has indicated a stronger relationship between three pictures of single-level buses as opposed to double-decker buses (towards the top-right side in Figure 5.7) by overlapping the images. The PRISM interface is separated into four functional areas:

- **Banner**: The banner is the least frequently accessed portion of PRISM. It is a narrow area across the top of the screen that situates the user and provide them with global control functions to get help, modify user settings, learn about the system, or save their session and exit the system. The banner does not change, whereas all of the other portions of PRISM are mutable.

- **Filmstrip**: The filmstrip is a wide, narrow region between the banner and the console. It is how new images are presented to the user. It either displays random images or related images. Random images are displayed on initial sign in or when the user requests it, whereas related images are shown only at the user’s request. The origin of the image (random, or related to content, text, or collaborative filters) is displayed when the user moves their mouse over the image. The system tries to ensure that the filmstrip is always full. Images are dragged from the filmstrip into either the main canvas (relevant images) or the “Delete Image” box (irrelevant images).

- **Console**: The console is a narrow, horizontal strip just below the filmstrip. It contains a variable number of tabs that the user can create and label to switch between multiple canvases. A button to fetch "Random Images" and another for "Related Images" are placed here as well.

- **Canvas**: The canvas occupies the most space in PRISM – the entire area that remains below the banner, filmstrip, and console. It is initially empty. The user can switch between multiple canvases by selecting the associated tab. An area of the canvas in the bottom-right of the screen is reserved for deleting images that are dragged and dropped on top of it. Once an image is dragged from the filmstrip to the main portion of the canvas it can be moved, resized, or annotated with text (Figure 5.8). Controls for doing so appear when the user moves their mouse over an image. The canvas allows the user to visually compose their query.

Three classes of actions can be performed in PRISM (illustrated in Figure 5.9). They are described as follows:

- **Displaying additional images:** The filmstrip can be populated with either random images or with related images (the results of the execution of the current query) depending on which button the user presses.

- **Add image to query:** The query consists of the images currently displayed in the active tab's canvas canvas area, their relative size, spatial location,

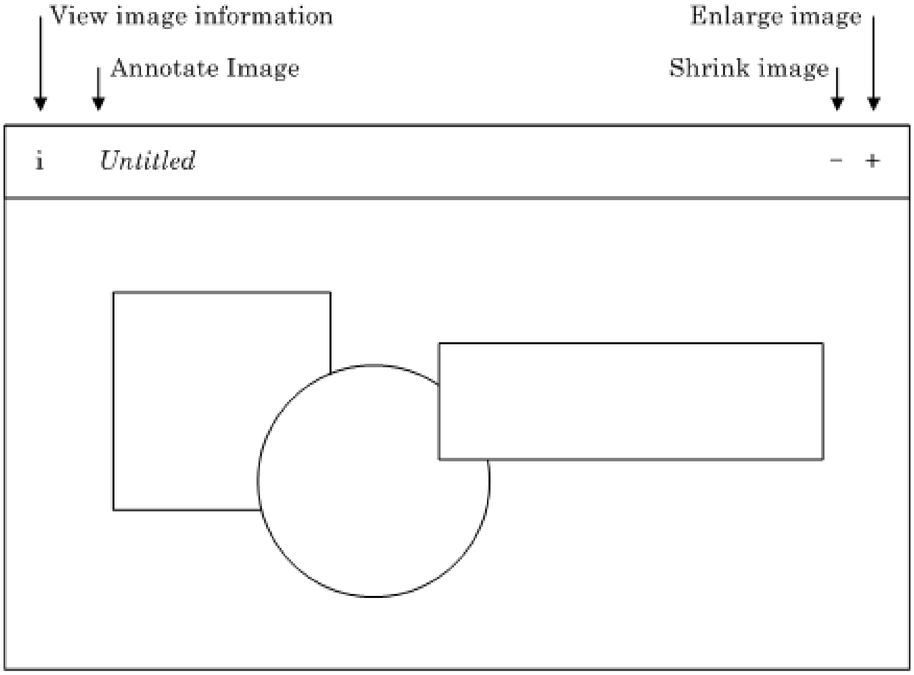


Figure 5.8: Image-specific functionality in PRISM

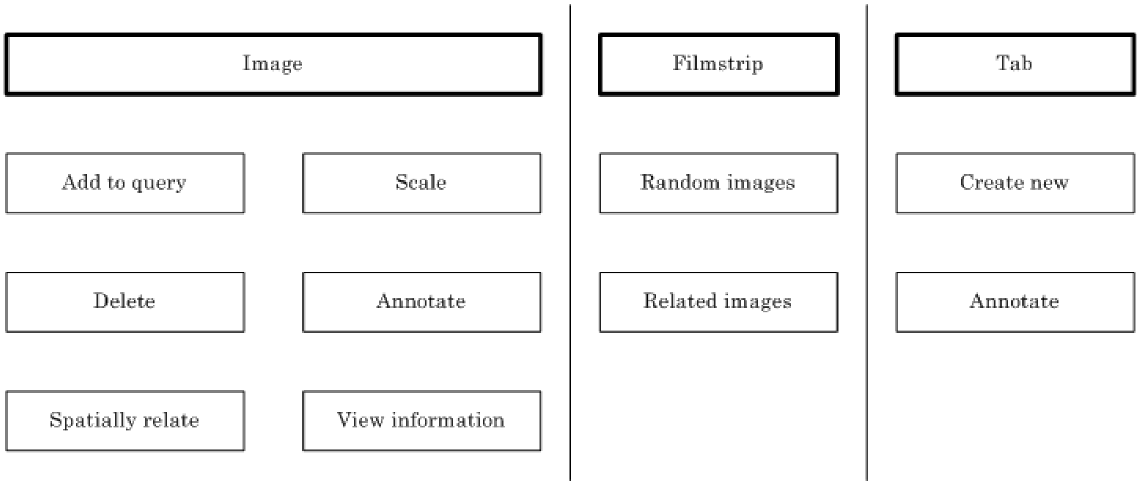


Figure 5.9: Actions in PRISM

individual image annotation, and the active tab's annotation. To add an image to the query the user drags an image originating in the filmstrip and drops it in the canvas area. All images currently visible in the canvas will be used in the next query (the next time the user presses the "Related images" button).

- **Remove image from query:** To remove an image from the current query the user performs the same drag-and-drop action as they did to add the image to the query, but instead moves the image to the "Delete Image" area in the bottom-right corner of the canvas.

- **Create a new, independent query:** The metaphor of using tabs to switch context is used frequently across operating systems and in contemporary applications (e.g. web browsers). A variable number of tabs that can be annotated by the user are provided. Clicking on a tab hides the current query and displays what was last shown on the selected tab, allowing the user to separate semantically-different groups of images. Tabs are also an important for organizing images.

- **Relate images to each other:** The spatial arrangement of images in the canvas is used to determine how strongly the user relates images together. Certain conditions, such as overlapping images, indicate stronger relationships between those images. Being able to rearrange images serves the user's desire to organize images.

- **Change the significance of individual images in a query:** The ability to scale individual images is used to intuitively weight the importance of images to a query. For each image, familiar "+" and "–" controls are displayed when the cursor is moved over the image (Figure 5.8). The ability to magnify an image to display it in more detail or to reduce the size of an image to provide more room for other images is an intuitive action that helps the user and the system as well.

- **Annotate images:** The same interface that appears when a user hovers their cursor over an image that allows the image to be scaled also allows the image to be annotated by replacing the default text with the desired keywords (Figure 5.8).

- **View image information:** Information on individual images can be viewed by clicking the "i" icon in the corner of any image in the canvas area (Figure 5.8).

From the user's perspective, these functions are easily described and allow the organization of images. When used together, these actions form an iterative loop of executing a query, adjusting the query parameters, and executing an improved query, as shown in Figure 5.10.

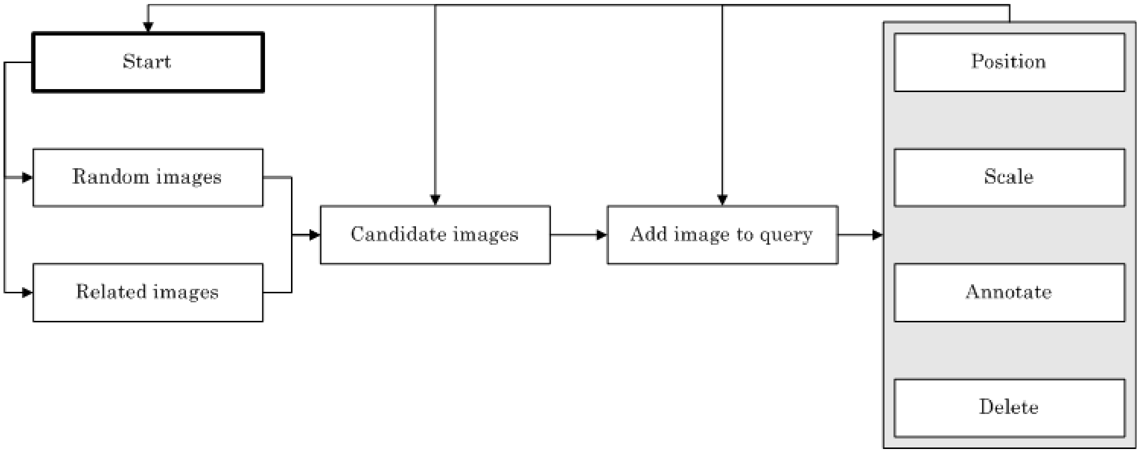


Figure 5.10: The iterative process of PRISM's interface

Each user benefits from creating a unique profile in the system, although a guest account is also available. This provides several capabilities:

- **Concurrency:** Multiple users can use the system at the same time
- **Session interruption:** A user can save their session, sign out, and resume it at a later time
- **Aggregation:** By maintaining each user's unique activity, mainstream and stray users can be better distinguished when collective activity is aggregated

5.5 Discussion

This chapter presented PRISM, an interface for image retrieval, organization, and annotation. It was motivated by the desire to create a complete an interface for image retrieval compatible with the attention-based image retrieval method proposed in Chapter 3 and Chapter 4.

PRISM also includes querying abilities beyond content-based image retrieval. Individual images can be annotated and used to query by keyword. Multiple images can be grouped together and used for collaborative filtering.

The implementation of PRISM is presented in Appendix A. There, the technical details of PRISM, such as the selected implementation language, system architecture, and flow of execution are discussed.

The complete, human-centered, PRISM system includes the user, the technical implementation, and the task of image retrieval. Thus, the user must also be part of the evaluation of PRISM. Chapter 6 presents a game used to evaluate the entire system.

Chapter 6

THE PRISM GAME

My work is a game, a very serious game.

M. C. Escher, artist, 1898 – 1972

6.1 Introduction

The PRISM system combines content-based image retrieval, collaborative filtering, and keyword queries together in a human-centered system. This hybrid approach creates unique challenges in evaluating the quality of the results, as the user is an essential component of the system. The open-ended nature of the interface allows a variety of expressive of queries to be composed. Furthermore, the learning component of the system means that the results change over time. The system relies not only on computer-generated (content-based) information, but on human-generated features (annotation, collaborative information) as well.

A variation of PRISM was created in order to effectively evaluate the system. While the interface has minor changes, the retrieval blocks are streamlined. The most significant difference is the presentation of the system to the user. This variation presents PRISM not only as an image retrieval, organization, and annotation system, but as a *game* with specific objectives. The game also serves to instruct users regarding the use of PRISM by rewarding more effective actions with more points.

It was necessary to explore ways to collect human data in order to establish the manual annotation needed for both collaborative filtering and keyword-based retrieval. There must be a clear benefit to the user providing this information. This benefit may belong to one of two classes:

- **Improved retrieval**: a system that requires human effort to annotate items must provide feedback as to the significance of the person's annotation. It is important to convey the contribution the human is making to the system without overwhelming a nonexpert user. One way to accomplish this is to quickly provide updated results. For example, many online stores provide recommendations of items a user may be interested in when they provide a rating for an item. Once serendipitous results are produced the human has added incentive to continue providing the system their ratings of products in the store. One example of a large database that has successfully collected manual annotation is LabelMe [119]. LabelMe has motivated people to contribute annotation solely with the promise of improving research results for other in the future.
- **Entertainment**: in some cases the retrieval task overlaps with one that provides the user with a diversion. For example, a user searching for music may

enjoy describing songs with keywords which can be used to find similar music or even users with similar tastes. It may even be possible to completely separate the retrieval task from the collection of annotation. In this case, annotation can be presented to the user as a game. The only incentive to participate may be the entertainment value derived from playing the game. This has been successfully demonstrated in the ESP Game [148] and Peekaboom [149], two games that have logged a massive amount of human-generated image annotation data. The ESP Game asks users to type in keywords that describe an image. Each pair of users receive points when they both guess the same keyword. Peekaboom is somewhat different in that two users alternate roles. The objective of Peekaboom is to annotate specific regions of images by having one user reveal a portion of an image that corresponds to a given keyword.

Both LabelMe [119] and Peekaboom [149] have the same objective – provide annotation for objects within images. They are both web-based. Interestingly, despite the different origin of each database, they are comparable. Indeed, each paper cites the other's work. Russel et al. [119] laud the volume of information games can collect but criticizes the quality of the data. Von Ahn et al. [149] faults the quality controls in LabelMe, which are limited to the faith that the person contributing the data is trustworthy. Peekaboom (and the ESP game), on the other hand, prevents cheating and bad data in several ways, foremost by having two people concur. However, the most telling difference between the solutions is the volume of data. As of the end of 2007, LabelMe consists of 161,780 images of which 41,969 are labeled, according to the LabelMe web site. The ESP Game has collected 33,524,492 labels since October 2003, according to its web site. In its first month of operation Peekaboom collected 1,122,998 labels from 14,153 different people [149].

Henceforth, the PRISM Game will be referred to as GAME (as opposed to the standard implementation, still referred to as PRISM).

6.2 Background

Gameplay is defined as the practice of applying actions, controlled by specified rules, within a particular situation, to achieve an objective [74]. This pattern of interactions (perceptual, cognitive, and motor operations) can be referred to as the *gameplay gestalt* [74]. Gameplay has similarly been defined as "all the activities and strategies game designers employ to get and keep the player engaged and motivated" [115].

The concept behind the game is the key factor distinguishing gameplay from performing an otherwise tedious task. For example, it is more enjoyable to "slay the dragon" using a series of keystrokes than to simply press those keys out of context. For similar reasons, games have been applied to education, as a way to motivate students [23]. Games "are examples of user-centered designed that motivate through learning, arousing players' interest (desire to act) and giving them the power of ample expression (pleasure to act)" [23]. Rather than the boredom that is experienced when performing a tedious action, gameplay introduces tension and resolution, leading to positive an negative emotions [21]. While the PRISM Game does not take place in a fantasy world, it does provide a strong, real-world premise (which is absent in the vast majority of games) – by playing the game one is contributing to the PRISM system and improving image search results for others.

Interactivity is an important aspect of gameplay. "It is not enough to just sit and watch and possibly activate some cognitive schemas. Instead, the player must become and active participant. When successful, this type of participation leads to strong gameplay experiences that can have [a] particularly powerful hold on the player's actions and attention" [31]. GAME enables interactivity in its design and persistently displays a timer, score, and progress meter throughout gameplay. Additionally, runs are ranked upon completion.

6.3 Objectives

A game must establish clear objectives from two perspectives: that of the system designer, and of the user. These are the considerations from the system designer's perspective:

- **Which features must be evaluated?** A wide range of variables can be evaluated. Interface features have a significant impact on the user experience.

Overt changes to the interface can be compared between user groups. Alternatively, the behind-the-scenes retrieval methods, such as content-based retrieval and collaborative filtering, can be evaluated.

- **How are outcomes evaluated?** The practitioner must decide the method used to evaluate results. This may be in the form of free responses in a user survey, a survey consisting of statements graded on a Likert scale, or a more subtle method, such as a timer, score, or other type of counter. Covert evaluation methods, such as counting the number of clicks or computing the quality of the user's queries can be employed, or even a combination of evaluation methods can be used.

- **What is the size and composition of the user group?** The number of control and test users must be established. Users may be divided into *control* and *test* user groups. The practitioner must decide if a few users will be enough to evaluate the system, or if many are needed. The level of technical expertise required of the users must also be established.

The player has a different set of objectives:

- **What is the task I must perform?** The user's task must be clearly and unambiguously defined. Furthermore, the complexity, challenge, and time demands must be communicated to the user.

- **How can I evaluate my performance?** The user must know if they are doing poorly or well. This can be communicated during gameplay or after gameplay. During gameplay, a running score, progress meter, or a timer can indicate success or failure. After gameplay a user's performance can be compared to that of other players to produce a relative ranking.

- **How can I improve my performance?** Incentive to replay the game should be established by communicating ways in which they can improve their performance.

- **What incentive do I have to complete this task?** The game must be demonstrated to be either fun, scientifically meaningful, or personally rewarding.

6.4 Gameplay

In order to encourage interactivity and measure progress, GAME displays a timer, score meter, and progress meter throughout gameplay. Additionally, runs are ranked upon completion. These elements provide additional avenues for motivation and emotional attachment beyond simply organizing images to improve image retrieval.

GAME is a single-player experience. From the user's perspective, they can play at any time, from anywhere, without regard for others that are playing at the same time. Their results are kept and compared to those of other players.

GAME was designed to be open-ended in that there are multiple ways to achieve the same objective. From the user's perspective it is up to them how they want to play. The score, progress meter, and a running timer are provided as benchmarks and feedback mechanisms. Good gameplay will result in the additional objective of trying to organize images in an effective, meaningful manner. This will be rewarded with more points. A veteran player may return to GAME in order to increase the score they are awarded or reduce their time, all valid objectives.

Self-motivation is a key aspect of all games, for when motivation disappears a game is no longer fun. When a game is no longer fun it is no longer a game. Thus, an additional objective is simply for the user to enjoy their time with GAME.

In GAME users are first presented instructions. These instructions broadly set out the objectives of GAME and introduce the key elements of the interface. The user is told that the objective of GAME is to organize related images while achieving the highest score. The score depends on the given task. It may be simply be the number of images they are able to organize within a certain period of time, or a more complex composite of multiple metrics. Images are organized by using the elements of the PRISM interface.

An example of good gameplay is shown in Figure 6.1. In this case, the user has nearly completed GAME any organized many images using many of the interface's features. The pictures of horses are all semantically-related. Furthermore, the user has enlarges the most representative images. Finally, images are overlapping,

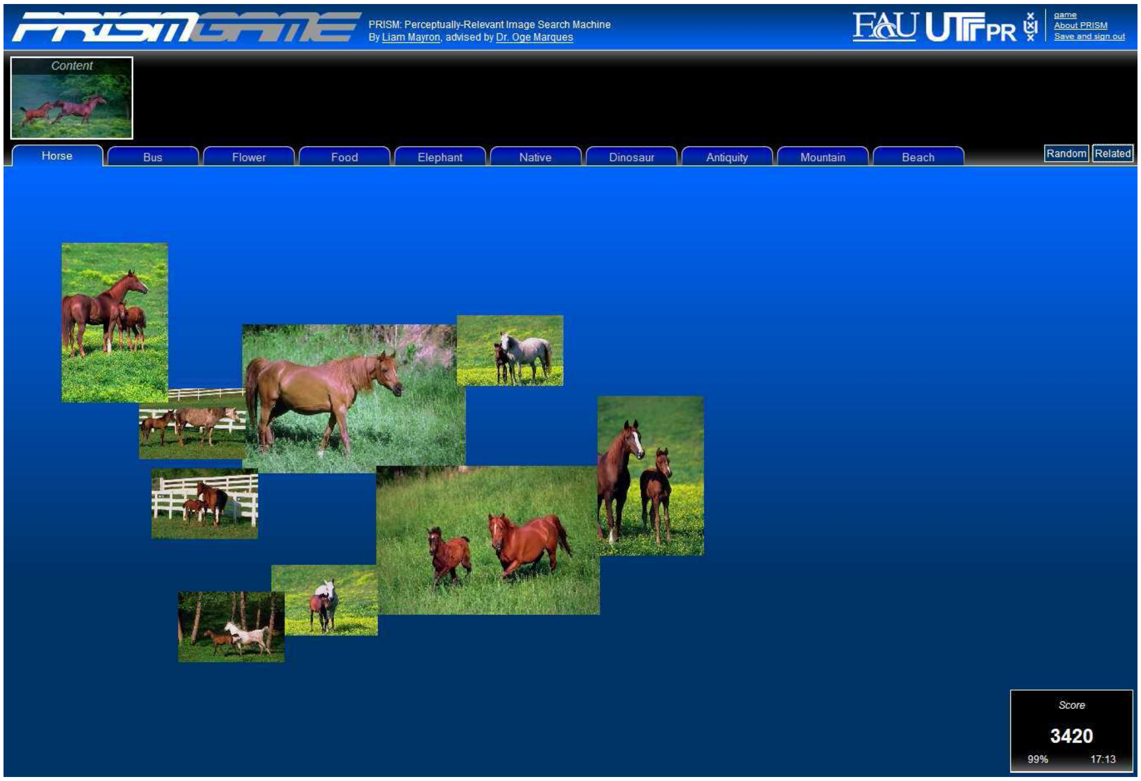


Figure 6.1: An example of good gameplay

indicating a stronger relationship. In contrast, Figure 6.2 shows inefficient gameplay. Here, the user has not effectively organized the images. The images are not related and range from ancient ruins, to horses, to a drawing of a dinosaur. The images have not been scaled to indicate relative importance, nor have they been spatially-associated by overlapping images.

6.5 Implementation

There are four modules which contribute to the complete PRISM game. They are:

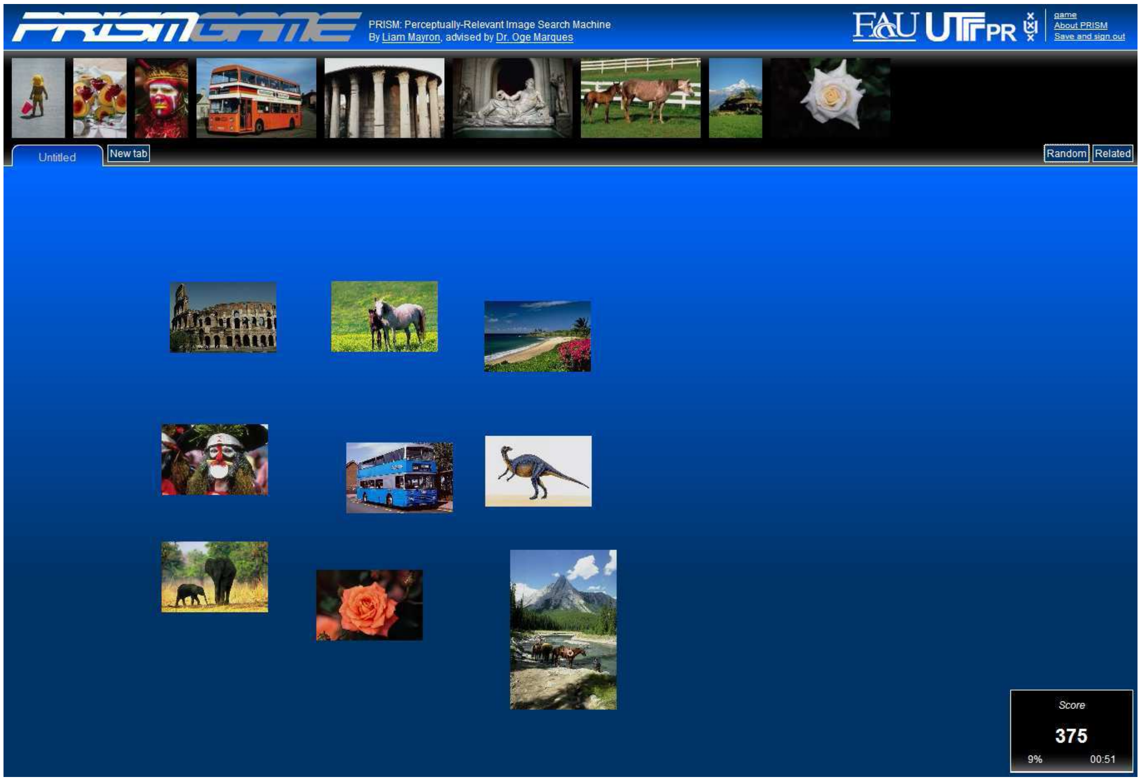


Figure 6.2: An example of a poorly-composed query

- **PRISM**: this is the standard implementation of PRISM upon which GAME is implemented (see Chapter 5 for details on the PRISM interface).

- **GAME**: GAME consists of the variations to the PRISM core.

- **REFEREE**: REFEREE is a web-based administration console for GAME. All aspects of the game, such as the rules, text labels, collaborative filtering information, results, and more can be monitored and modified from REFEREE. The features of REFEREE are described in Section 6.5.1.

- **SITE**: SITE is a web site that introduces users to PRISM before the game and displays results once the game is complete. SITE is presented in Section 6.5.2.

6.5.1 REFEREE

REFEREE is the administration utility for GAME. The following functionality exists in REFEREE:

- **Show all images**: this allows the monitoring of all images in the current database. Thumbnails for all images are displayed in a grid. Clicking on an image displays detailed information on the selected image.

- **Show individual image information**: this mode is illustrated in Figure 6.3. The image is displayed in its original size in the left column. The right column displays six categories of information, three for keywords (generated as ground truth by the author of this dissertation) and three for collaborative filtering (also generated as ground truth by the author of this dissertation). Keywords are displayed representing their original weighing, TF-IDF weighing, and LSA (please see Section 2.5 for a description of TF-IDF and LSA). The words are illustrated as tag clouds [44], displaying more heavily weighed words in larger text sizes and vice-versa. The initial weights were manually assigned. Similar displays are shown for the manually determined collaborative filtering information, showing the default weights (images are always the same size), TF-IDF weighting, and LSA weighing. The number after the LSA headings (e.g. "LSA10" and "LSA17") refers to the number of singular values used in LSA (refer to Section Section 2.5 for more details). Images, not text, are shown for this category. Images are scaled according to their relative weight, similar to the tag cloud presentation of keywords. This mode allows the administrator to visualize the effect each image will have when used as part of a query in GAME.

- **Set keywords**: each image is displayed. Next to each image is a text box whose contents may be edited. If this box is empty no keywords have been associated with an image. Otherwise it is populated with keywords that will be used in GAME for searching. Listing the same keyword multiple times increases the weight that keyword is given for that image.

- **Set collaborative filters**: the interface for setting collaborative filter information is the same as that for setting keywords. The difference is that instead of selecting keywords, the user must associate images using *virtual keywords*.

- **Set image order**: optionally, the administrator can specify the order images are to appear in GAME. While implemented, it was decided to display images randomly rather than in a scripted manner.

- **Set rules**: each rule displays its name, description, and value. The value (positive or negative) can then be updated.

- **Get results**: the score for each run is displayed. When clicked, details of the run (similar to those shown once the game is complete) are displayed. Overall statistics are also shown.

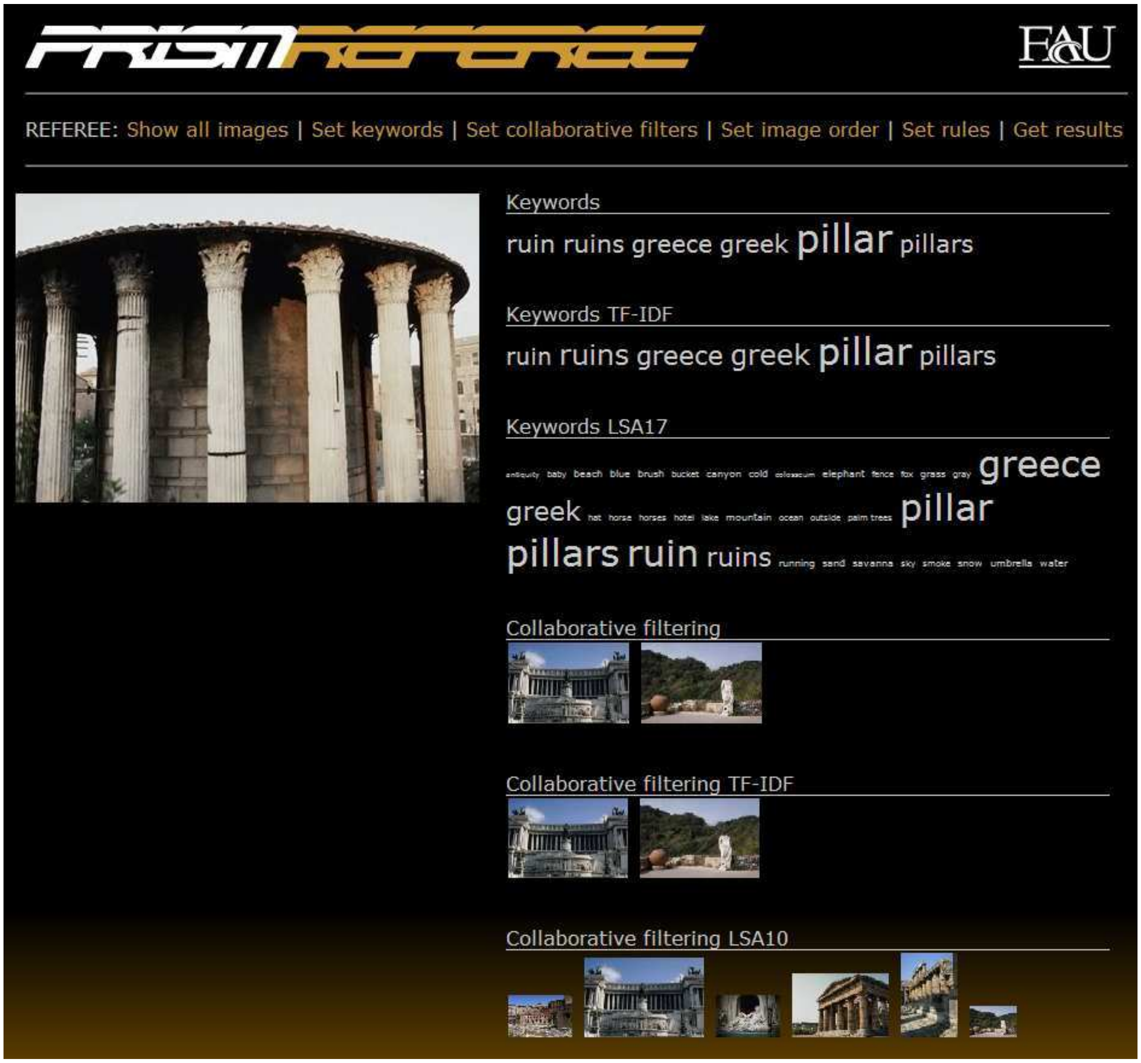
6.5.2 SITE

SITE is a public web site with information about PRISM and GAME. It provides instructions to the user and then directs them to PRISM. The users is returned to SITE once the game is complete. An analysis of their gameplay is then displayed (the incentive for the user to use GAME).

An important component of SITE is the survey the user is prompted to complete. This survey provides valuable feedback which is to be used to assess the system. Some questions can be responded to using a few sentences. However, most questions are scored using a five-point Likert scale [63, 111], formatted with a semantic differential [40]. A survey can only be completed once per run. Once the survey is complete the results are stored in the database for further analysis.

6.5.3 Content-based image retrieval

Each image had four sets of global, color-based histograms extracted (RGB, YCbCr, HSV, and HMMD – see Section 2.4.5). The five considered distance measures are Euclidean distance, Manhattan distance, histogram intersection distance, D1 distance, and cosine similarity (please refer to Section 2.4.6). For evaluation



REFEREE: Show all images | Set keywords | Set collaborative filters | Set image order | Set rules | Get results

Keywords

ruin ruins greece greek pillar pillars

Keywords TF-IDF

ruin ruins greece greek pillar pillars

Keywords LSA17

antiquity baby beach blue brush bucket canyon cold colosseum elephant fence fox grass gray greece greek hat horse horses hotel lake mountain ocean outside palm trees pillar pillars ruin ruins running sand savanna sky smoke snow umbrella water

Collaborative filtering

Collaborative filtering TF-IDF

Collaborative filtering LSA10

Figure 6.3: Displaying an individual image and its associated information in REFEREE

purposes, each image is exhaustively compared against every other image in the database for each distance measure. Once the distance between the given query image and all other images in the database has been computed the results are then ordered from smallest to greatest distance, excluding the query image. Finally, the ordered results are stored in a new table in the database for further evaluation. The HSV color space and D1 distance measure were ultimately selected as they exhibited the best performance.

6.5.4 Keyword-based retrieval

Keywords in GAME are set using REFEREE. For each image none, one, or several keywords were specified. Out of the 100 images in the database, 32 were assigned one or more keywords. A total of 61 unique keywords were used, resulting in 123 keywords being assigned to images. Keywords are terms such as “elephant”, “palm trees”, or “umbrella”. The most common keyword was “sky”, which was associated with 8 images. Keywords may have different initial weights.

Many keywords are assigned to only one image. Using traditional retrieval searching using these terms will only retrieve one result. However, using LSA (see Section 2.5) the semantic relationships between the given term and other terms associated with that image can be extrapolated and more results retrieved. These semantic relationships are derived by analyzing existing image arrangement, stored in the database, and loaded at the beginning of a PRISM session along with the other content-based and collaborative features.

From the original keyword assignments an inverted index is created (Section 2.5). The inverted index allows for the fast searching for images associated with specific terms and was used throughout these experiments.

Keyword information is stored in a database where each row corresponds to an image. A second version of this term-document matrix is also stored in the database, but adjusted using TF-IDF weighing (Section 2.5).

Latent semantic analysis (LSA) (Section 2.5) was also performed. Using PHP, the information was exported from the MySQL database to MATLAB, where the transformation was executed. The output from MATLAB was then inserted into the database again using PHP. The singular value decomposition (SVD) [138] function in MATLAB is key to this capability. SVD creates three matrices from the original LSA data. These three matrices are then reconstructed into one matrix. In LSA, the parameter n represents reconstruction with the n largest singular values. In experiments n ranged from 1 to 20. For text, $n = 17$ demonstrated the best performance.

6.5.5 Collaborative filtering

Collaborative filtering in GAME is very similar to keyword-based retrieval, although there are some differences. REFEREE allows virtual keywords to be assigned to images. A virtual keyword is a unique string with no semantic meaning that is shared by two or more related images. There are manually-assigned 23 virtual keywords (related groups of images) in GAME. Group sizes range from 2 images to 7 images. The most virtual keywords assigned to an image is 3, most images have one or two virtual keywords. 61 images (61%) have been assigned collaborative filtering information.

When performing a collaborative query for a single image, all virtual keywords are looked up in the inverted index. All images associated with those virtual keywords are retrieved. As with keyword retrieval, TF-IDF weighing and LSA with between $n = 1$ and $n = 20$ singular values was computed, in addition to the normal weights. LSA with $n = 10$ was determined to yield the best results for collaborative filtering.

6.6 Experiments and results

The PRISM Game (available online at `http://mlab.fau.edu/prism/site/`) was available for play between February 7 and February 24, 2008. During this period, 28 sessions of the PRISM Game were completed. Seventeen individuals also completed a user survey after their game concluded. 93% of the users (26 out of 28) users searched using the "related images" query. 39% of the users annotated at least one image. Images were annotated 234 times (8.75% of all images). Users were most satisfied with the ability to retrieve images using keywords. Searching for keywords is an overt method (the user

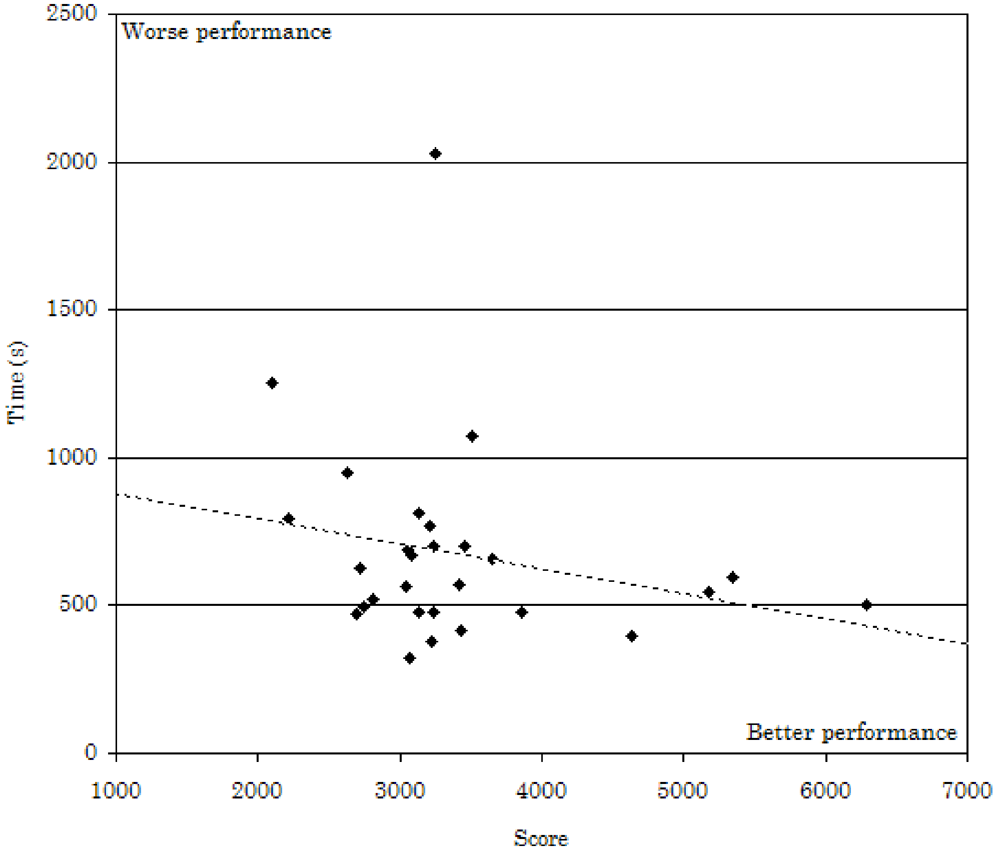


Figure 6.4: Time vs. score for the PRISM Game

explicitly specifies the keywords), whereas content-based retrieval and collaborative filtering are covert, behind-the-scenes methods. The abilities to create new tabs and to annotate tabs were highly rated which led to the conclusion that organization capabilities are important to users. Furthermore, being able to view a large version of the images images in the system was also rated highly. This is a simple feature, but it eases viewing images in detail.

Figure 6.4 plots scores achieved in the PRISM Game versus the time required to complete the game for all users. The *score* metric indicates the efficiency with which the user organized and annotated images, whereas *time* is the amount of seconds taken to complete the task. Higher scores and lower times indicate better performance. The graph demonstrates the inverse relationship between score and time indicating that users who achieved higher scores were able to more efficiently organize images using the features of the PRISM interface, although this is a weak correlation (indicated by the dashed line in Figure 6.4).

However, higher scores do not necessarily correlate to higher satisfaction. In a separate analysis, the correlation coefficient between score and users responses to the question “what is your overall opinion of PRISM?” was computed and shown to be only 0.0874 – highly ranked user performance does not indicate increased satisfaction.

The results of the gameplay surveys are summarized as follows:

- **In general, how would you improve image search?**: Answers to this question ranged from speculating as to the difficulty of image search, to specific suggestions to improve PRISM. Many of the comments regarded searching for images using text, perhaps because this is the image search paradigm that is most readily available today (e.g. Google Image Search [42]).

- **How would you improve PRISM? What features would you add or**

remove?: In the following list, related feedback has been grouped together:

- **Tabs and annotation**: In the non-game implementation the need to move images between tabs is diminished because incorrectly-placed images can be deleted and reappear in subsequent queries. In GAME the delete function has been removed. Given the complexity of the new search interface, adding the ability to move images between tabs may have resulted in more confusion.

- **Technical and performance aspects**: PRISM's nature as an AJAX-based Web application makes it vulnerable to slow responsiveness whenever the client must interact with the server. While this is due to latency across the Internet that is beyond the control of the implementation, it is perceived by the user as latency nonetheless, reducing their satisfaction. PRISM gains much by not being implemented as a traditional, client-based application, although what is lost in responsiveness and performance cannot be overlooked.

- **Features**: In the regular (non-game) implementation of PRISM the filmstrip is indeed "always full" and replenished itself whenever an image is removed. This behavior was modified for GAME. The standard PRISM client was specifically-designed to allow users to save, suspend, and resume their sessions. This too was removed from GAME.

- **Database**: GAME's database was a specifically-selected 100-image collection. Were it any larger users may not have the patience to complete GAME. The standard implementation of PRISM has been designed to work with larger databases (please see Section 4.4 for a discussion of these databases).

- **Ease of use**: Certain users believed they did poorly in the game and thus were disappointed (despite the intended challenge of the game for new users). PRISM can be improved, incorporating more help and on-the-fly feedback in order to reduce the confusion of new users.

6.7 Use cases

Several use cases for future study have been formulated by revising the objectives of Section 6.3 in the context of PRISM and GAME. Three use cases are proposed: image retrieval, query by keyword, and evaluation of the use interface. This is not intended to be an exhaustive list of uses of a game to evaluate an image retrieval system, but a representative one. From a player's perspective, the objectives are consistent:

- **What is the task I must perform?** The user will be told to retrieve as many images of a certain category (sufficiently broad, e.g. "beach scenes") within a specified time limit

- **How can I evaluate my performance?** The more images retrieved, the better the performance

- **How can I improve my performance?** Users can improve their performance by composing better queries (except in the lone case where they are only shown random images, as the composition of the query has no effect)

- **What incentive do I have to complete this task?** By keeping the task short and focused with a clear method to improve performance it is intended that the incentive to complete the task is the entertainment value derived from the process

From the system designer's perspective, the objectives depend on the given task:

- Image retrieval
  - **Which features must be evaluated?** The impact of different retrieval algorithms on user satisfaction will be evaluated
  - **How are outcomes evaluated?** Outcomes will be evaluated in two ways. First, quantitative measures will be recorded. The user will be asked to find as many images of a certain category within a time limit.

The number of images they retrieve will be recorded. In order to obtain qualitative measures, users will be presented a survey containing statements to be graded on a Likert scale.

  - **What is the size and composition of the user group?** The test group will consist of approximately five non-expert users. These users will have their queries resolved using a content-based image retrieval algorithm. On the other hand, the control group (approximately the same size as the test group) will retrieve only random images when querying. There may be multiple test groups, each testing a different image retrieval algorithm.

- Query by keyword

  - **Which features must be evaluated?** The impact of querying by keyword on user satisfaction will be evaluated

  - **How are outcomes evaluated?** Quantitative measures will be recorded. The user will be asked to find as many images of a certain category within a time limit. The number of images they retrieve will be recorded. Qualitatively, users will be presented a survey containing statements to be graded on a Likert scale.

  - **What is the size and composition of the user group?** The test group will consist of approximately five non-expert users. These users will have access to a text box for querying by keywords and will be instructed on its use. The control group (approximately the same size as the test group) will not have access to the text box and will have to complete the same task using content-based retrieval alone.

- The user interface
  - **Which features must be evaluated?** The efficiency and satisfaction of users when using different user interfaces will be evaluated
  - **How are outcomes evaluated?** Quantitative and qualitative measures will be recorded. Again, the user will be asked to find as many images of a certain category within a specified time limit. The number of images they retrieve will be recorded. Users will then be presented a survey containing statements to be graded on a Likert scale in order to obtain qualitative measurements.
  - **What is the size and composition of the user group?** The test group will consist of approximately five non-expert users. These users will have access to the PRISM interface for searching for images and will be instructed on its use. The control group (approximately the same size as the test group) will instead have to complete the task using a plain, traditional interface that displays retrieved images in a grid (as with many Internet image search engines).

6.8 Summary

The feedback that was captured in the context of the game provided several insights to users' perception of the system:

- Users recognize and value text annotation but are not willing to spend much time contributing their own annotation (demonstrated by GAME's records). Images were annotated 234 times by users (only 8.75% of all images acted upon).

- "Hidden" retrieval methods (e.g. content-based and collaborative retrieval) do not have as much of an impact on user opinions of the system as text retrieval, which they can see and actively participate in. Features such as image scaling, which are essential to content-based and collaborative retrieval, were rated as less useful in user surveys than text annotation features.

- Comments show that users become confused and anxious if they cannot perceive their progress in the game or are challenged by the game. This does not apply as much in traditional image retrieval where the object is the user's own intended target images.

- Making the user comfortable with the interface and showcasing the images themselves is an important, yet often-overlooked aspect of image retrieval systems. Users use image retrieval systems to locate image which they intend to look at and inspect in detail. Image retrieval systems should incorporate image browsing tools in order to improve user satisfaction.

- The ability to compose multiple queries in parallel using multiple tabs was appreciated by users. Features associated with the tabbed interface should be a priority when improving the system.
- Users are sensitive to latency and processing delays.

Several improvements can be made to this initial trial. In the PRISM Game all users were rated against each other. Instead, the users should be divided into control and test groups. We learned that the task we assigned users – to organize all images as they see fit – was too broad, ambiguous, and complex. Instead, a smaller, more focus tasks needs to be given to users (e.g. asking users to find several images of a specific type of scene within a certain amount of time). Three principled use cases that take heed of the lessons learned were proposed to evaluate image retrieval, query by keyword, and the user interface.

Chapter 7

CONCLUSION

It is not really difficult to construct a series of inferences, each dependent upon its predecessor and each simple in itself. If, after doing so, one simply knocks out all the central inferences and presents one's audience with the starting-point and the conclusion, one may produce a startling, though perhaps a meretricious, effect.

Sir Arthur Conan Doyle, author, 1859 – 1930

7.1 Conclusion

This dissertation developed a method of searching for objects in images using visual attention. First, a proof of concept that used saliency to detect objects of interest was developed. Subsequently, the proof of concept was expanded into a new method for detecting objects in images. An interface compatible with this method of content-based image retrieval capable of providing additional organization and annotation abilities was implemented. A game was proposed as a method of evaluating the complete system.

An image retrieval system using a computational model of visual attention was implemented and evaluated as a proof of concept design. This system combined the computational model of bottom up visual attention proposed by Itti, Koch, and

Niebur in [61] with that proposed by Stentiford in [134]. These models were used to create masks of the salient regions in images. RGB and HMMD descriptors were extracted. HMMD consistently outperformed RGB in experimental results. The system performed well, detecting 77% of regions of interest at a false alarm rate of 28% in the test database. Several avenues for improvement were identified. Feature extraction, similarity, and clustering algorithms can all be improved in the traditional CBIR directions. Computational models of visual attentions, however, were critical to the performance of this system.

The proof of concept was extended with a new method of using the complete computational model of visual attention, including the inhibition of return, to generate points of attention. These points of attention were then organized into clusters. After postprocessing, the centroids of the resulting clusters were intended for use as seed points for a seed-based region-growing algorithm. The new method was tested using a variety of databases containing objects of interest that are not necessarily salient by design. Approximately 75% of points of attention hit objects at a false alarm rate of approximately 25%.

A new interface for image retrieval was designed. This interface was made compatible with the aforementioned new method for image retrieval by including the abilities to query by multiple example images and weigh each example image's importance to the query. The interface provides retrieval, organization, and annotation capabilities. A game was developed to evaluate the user, user interface, and retrieval methods together as a complete system. An initial user study made several conclusions. Users can easily understand and use retrieval by keyword features, but are not necessarily willing to contribute their own annotation. Subtle methods of retrieval (e.g. collaborative filtering) are perceived as being less effective than more over methods (keyword-based retrieval). Image organization features improved the user experience. In certain cases the interface was used primarily as way to organize images rather than as a retrieval tool. Finally, satisfaction with image retrieval systems improves when features for browsing and viewing images are provided. Such features are often overlooked in image retrieval systems where the focus is the quality of results, not of the retrieval experience of the human end-user.

In summary, this dissertation made the following key contributions:

- The design, implementation, and evaluation of an attention-driven method to extract regions of interest from images containing objects that are salient by design

- The design, implementation, and evaluation of an attention-driven method to detect objects of interest in broad image databases

- The design and implementation of an image organization and retrieval system incorporating visual features, keywords, and collaborative filtering

- The development of a new method for evaluating image organization, annotation, and retrieval systems using a game metaphor

- A study of recent advances in image retrieval
- A study of established, relevant work in cognitive science, concentrating on visual attention, with applications for image retrieval
- A survey of image databases for object-centered image retrieval

It is the author's hope that those the readers of this dissertation, at a minimum, take note of these two points:

- The human visual system is a complex, yet effective image processing system. Researchers interested in all areas of computer vision should keep abreast of the latest advances in vision science, particularly computational models of specific functions of the human visual system. Insights gained from vision science may yet prove to be the missing piece in certain open problems of computer vision.
- The interface of a visual information retrieval system matters. The ultimate goal of these systems is to satisfy the user. Thus, there are considerations beyond the system's pure quantitative performance. The user's understanding of the system's capabilities, and the system's ease of use are vital qualities of an effective retrieval tool.

7.2 Future work

This work unified previous disparate fields in information retrieval and vision science. There are many promising directions for future work.

The three retrieval methods used in this research can be improved. The content-based features used throughout this dissertation were baseline descriptors. Both local and global descriptions of an image must be investigated. The decomposition of an image into a set of local objects and a global context will expand retrieval possibilities, narrowing the semantic gap. Measures of similarity between images should also be investigated.

Improved keyword retrieval methods can be developed, incorporating stemming and other text analysis techniques. Collaborative filtering can be augmented with further traditional collaborative features such as ratings. Furthermore, probabilistic LSA has been shown to improve upon the performance of LSA [48].

The user interface, including the interpretation of the query, is a key area for future work. Numerous suggestions were made by users of the PRISM system. These can all be addressed as part of future improvements to the developed interface. The speed, responsiveness, and robustness of the system can be improved. Image browsing features were received positively by users and should be expanded. Along the same lines, more functionality should be added to the tab-related aspects of the interface. Most significantly, PRISM can be integrated with publicly-accessible photo sharing services. For example, Flickr provides a programmer's interface to develop applications that can import specified image content [155]. Providing the capability for users to use PRISM to retrieve, organize, and annotate *their* images (rather than a pre-selected image database) is an exciting future direction.

Projects such as Open Mind Common Sense [129] collect an abundance of useful information from humans (e.g. statements of common facts such as "airplane can be in the sky"). The use of this knowledge coupled with improved local and global image descriptors has the potential to reduce the semantic gap. For example, a CBIR system may be able to determine that the background of the image in question is the sky and not the ocean. Using knowledge from Open Mind Common Sense, an ambiguous object in this image may be ruled out as being a fish, as fish cannot exist in the sky, and instead identified as an airplane. Using this existing metadata is an interesting direction for related future work.

The use of the computational model of visual attention can be extended in future work. Additional features beyond color, intensity, and orientation can be explored. A model of surprise [57] can be tested instead of saliency. Furthermore, alternatives to clustering the points of attention to generate seed points can be investigated. Finally, top-down information, such as pre-extracted category features or previously-defined heuristics can be used to modulate the model of visual attention.

Appendix A

IMPLEMENTATION DETAILS

> *As soon as we started programming, we found to our surprise that it wasn't as easy to get programs right as we had thought. Debugging had to be discovered. I can remember the exact instant when I realized that a large part of my life from then on was going to be spent in finding mistakes in my own programs.*
>
> Maurice Wilkes, computer scientist, b. 1913

A.1 Introduction

This appendix presents implementation details of PRISM and REFEREE, the administration utility of the PRISM Game.

PRISM was written in PHP, a web-based scripting language. Several components used for offline were created using MATLAB. There were several factors that contributed to the selection of PHP as the main language for this project:

- **Web-based**: as PRISM is a web-based system, it made sense to use a language designed for the Internet. PHP is intended to be integrated with a web server and output directly to a web browser. Additionally, several Ajax implementations have been written for use with PHP (this work uses Xajax [154]).

- **The web browser is the user interface**: using PHP meant that the web browser would be the user interface. Tools for developing user interfaces for web pages such as CSS and JavaScript (collectively known as DHTML) are mature and allow great flexibility in the interface's design.

- **Platform independent**: making PRISM a web-based application implies platform independence on the client side (i.e. PRISM can be used in a variety of web browsers on a variety of devices). Using PHP for the server makes the server platform independent as well (i.e. the server can run on Windows, Linux, or other platforms that support the Apache web server, MySQL, and the image processing libraries of PHP).

- **Database support**: the ability to interface with a MySQL database was key to the implementation of this system.

- **Image processing support**: PHP is able to manipulate most image formats. This allowed tighter integration between the offline image processing components and the online system. For example, histograms could be directly extracted from images and stored in the database without the need for intermediate tools.

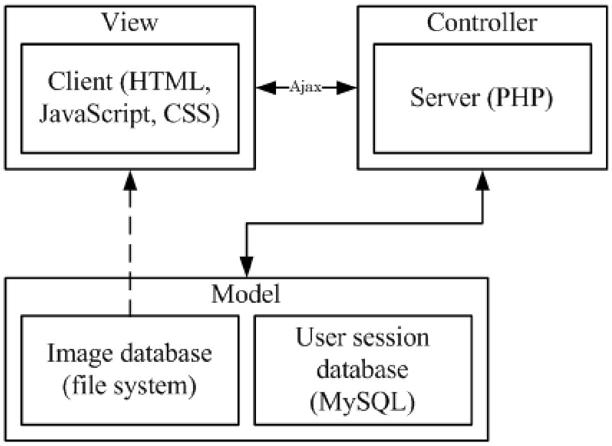

View
Client (HTML, JavaScript, CSS)
Ajax
Controller
Server (PHP)
Model
Image database (file system)
User session database (MySQL)

Figure A.1: The PRISM system architecture

A.2 System architecture

PRISM is implemented as a web-based application. The general architecture follows the Model-View-Controller design pattern [41], as shown in Figure A.1.

The data model consists of two components: a relational database, implemented in MySQL [94], and a file system-based image database. MySQL is a widely-available relational database. The following information is stored in the database and is used by PRISM:

- **User credentials**: users must create an account to use PRISM

- **User session information and preferences**: various parameters of PRISM, and the activation of a special game mode (see Chapter 4) may be adjusted by the user. Additionally, the user may suspend and resume their session at any time. Thus, information on all manipulated images and tabs must be stored.

- **Image database access and display information**: information on the files in the image database, including the names, paths, and dimensions of those files, is maintained.

- **Featured extracted from the image database**: pre-extracted features are stored in the database for quick access when a content-based search is performed.

- **Keywords assigned to the images in the image database**: user-provided keywords, extracted from saved session information, is stored and used in text-based searched. This includes derived information, such as the weighed term-document matrices and LSA-modified matrices.

- **Collaborative filtering information**: similar to the information stored on keywords, collaborative relationships inferred from saved sessions is stored, including derived information.

The file system is far simpler. It is a collection of images organized in either a flat folder system or a hierarchy of folders. Samples of the original images at multiple dimensions are also maintained (e.g. thumbnail representations).

The controller is implemented in PHP [113]. Its purpose is to query the database in response to events in the view. Requests are invoked through AJAX (Asynchronous JavaScript And XML) [43] requests. Results are returned to the client in the same manner. The XAJAX [154] library was used to implement this functionality.

The view consists of a client-side web application combining several popular web technologies [71]: HTML, JavaScript, Cascading Style Sheets (CSS), and Flash. A minimal amount of HTML is used to create several empty container objects. These objects are then populated by JavaScript function calls which, in turn, obtain their content from the server queried over AJAX. CSS is used to manipulate these objects. Flash is used in a limited fashion to trigger certain audio events.

A.3 Organization

PRISM is organized into the following files (excluding external libraries and media resources such as sounds and images). Files are either online and accessed when PRISM is in use, or offline and used for manually-executed batch processing of retrieval features (visual, keyword, and collaborative filters).

- **index.php**: online, this is the inception point of PRISM. It contains the most basic elements needed to construct the view. It consists nearly entirely of empty `<div>` HTML elements that will be manipulated by other elements of the program.

- **script.js**: online, contains the main JavsScript functionality of PRISM. It includes functions to initialize the page, query images, and auxiliary functions.

Additionally, it contains the implementation of key elements of GAME, such as a timer.

- **style.css**: online, contains style information for the elements in the PRISM user interface.

- **inc_config.php**: online, contains configuration information and constants used throughout PRISM. This is the only file that should be modified in normal use. It allows the dynamic specification of query features and image database.

- **inc_db.php**: online, functions controlling PRISM's database connectivity, including additional functions to process queries

- **inc_ir.php**: online, functions that are called at various points in PRISM's execution to process visual features, text features, and collaborative filtering, returning a list of images ranked by relevance. Several other utility functions are also included.

- **inc_prism.php**: online, called from index.php in order to include other required modules

- **inc_ui.php**: online, functions that draw and manipulate the PRISM interface. Contains the user session and all server-side AJAX functionality. Most of the functions are called as responses to events initiated on the client-side in JavaScript.

- **aux_cbir.php**: offline, computes statistics for CBIR.
- **aux_cf.php**: offline, utility functions for processing and formatting collaborative filtering information.
- **aux_color.php**: offline, generates color histograms.
- **aux_game.php**: offline, evaluates aggregate information from the PRISM Game.
- **aux_text.php**: offline, utility functions for processing and formatting keywords
- **aux_textVOC.php**: offline, used in the specific case of having to import ground truth keyword information from a Visual Objects Challenge [109] formatted database.

A.4 Events

PRISM is an event-driven application. The major events are illustrated in Figure A.2. Initially, only the *Start* event can occur, which initializes the system. This triggers the *Welcome* event, displaying a sign on screen. A successful sign on will launch the *Startup – Client* and *Startup – Server* events. Once the system

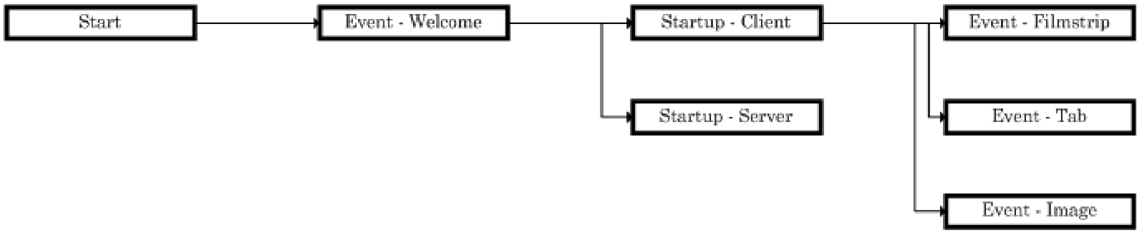


Figure A.2: Key events in PRISM

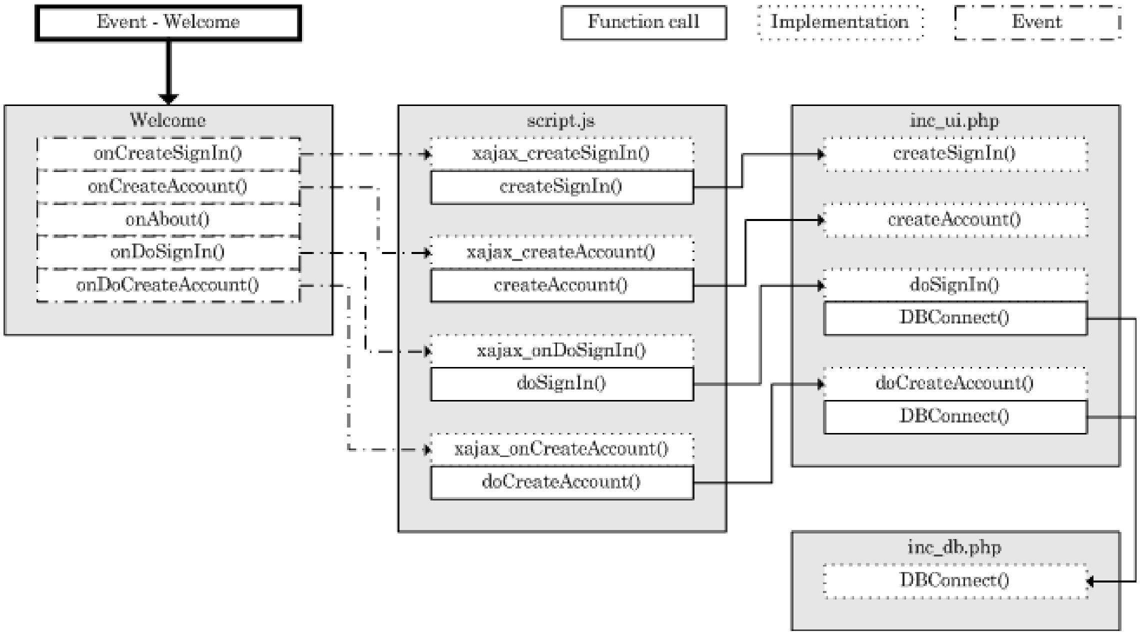


Figure A.3: Implementation of the PRISM sign in method

is initialized it reaches a steady-state where events can be either of the *Filmstrip*, *Tab*, or *Image* category. All events are detailed in the following figures.

The *Welcome* event (Figure A.3) draws the sign in screen. From here a user may sign in, create an account, or retrieve more information on the system.

The *Startup – Client* event (Figure A.4) loads the client application. This

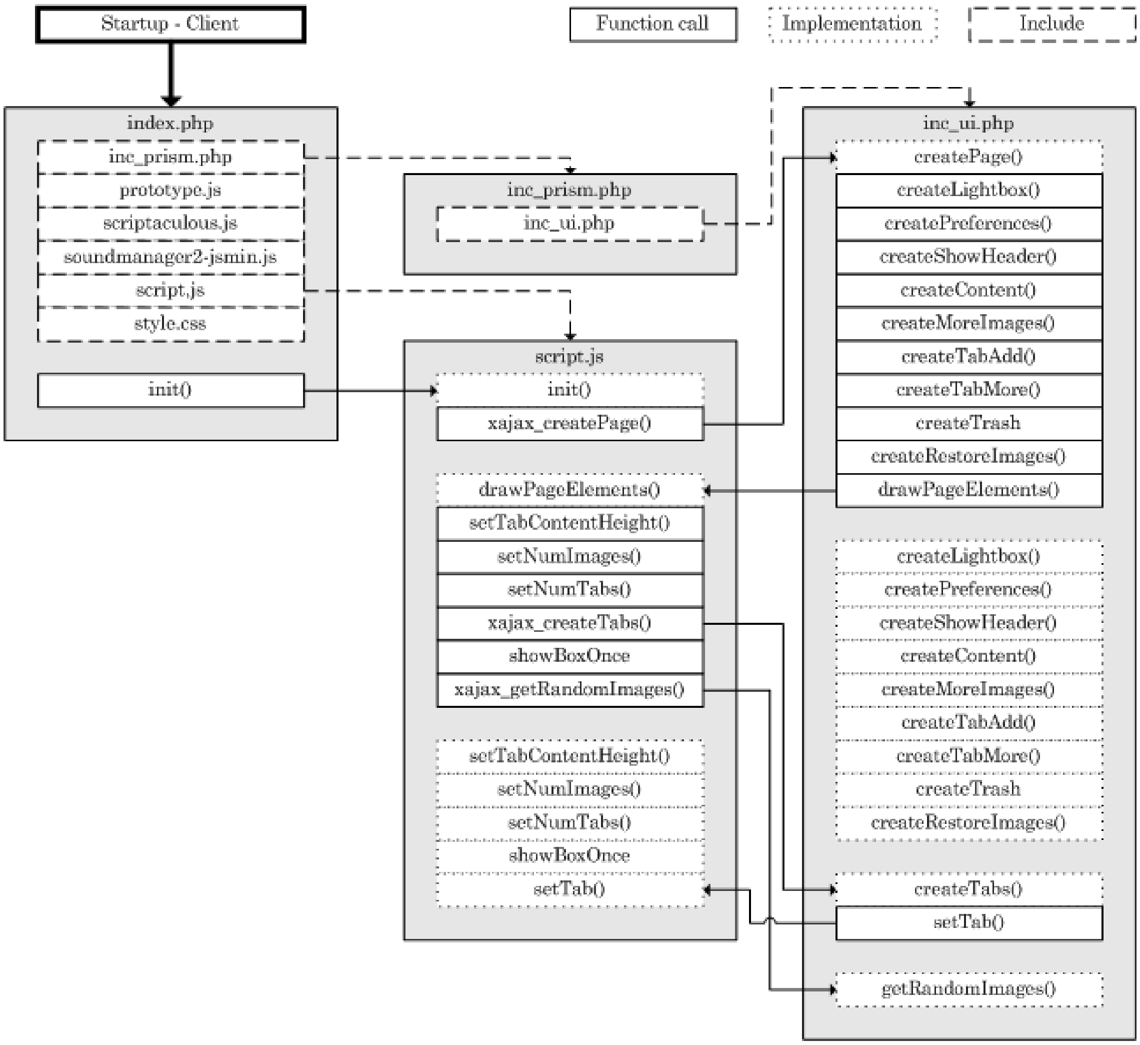


Figure A.4: Initialization of the PRISM client

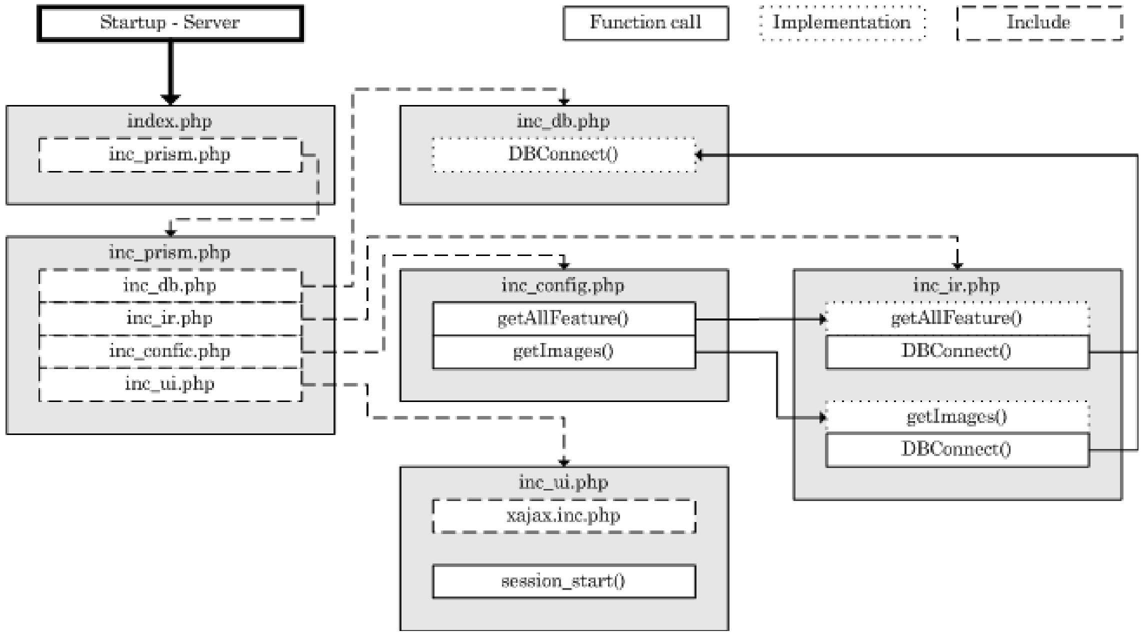


Figure A.5: Initialization of the PRISM server

consists of JavaScript functions on the client side and PHP functions to draw the page (in response to requests from the client) on the server side.

The *Startup – Server* event (Figure A.5) loads the server application. This loads configuration parameters (e.g. the visual, text, and collaborative filtering features to be used, as well as the image database), connects to the database, and initiates a session with the user.

The *Filmstrip* events (Figure A.6) refer to actions that can be performed on the images in the filmstrip. The images may either be hovered upon with the mouse, moved to a tab, or moved to the trash. The two types of queries, *Related* and *Random* are also included as *Filmstrip*-type events as they also manipulate the

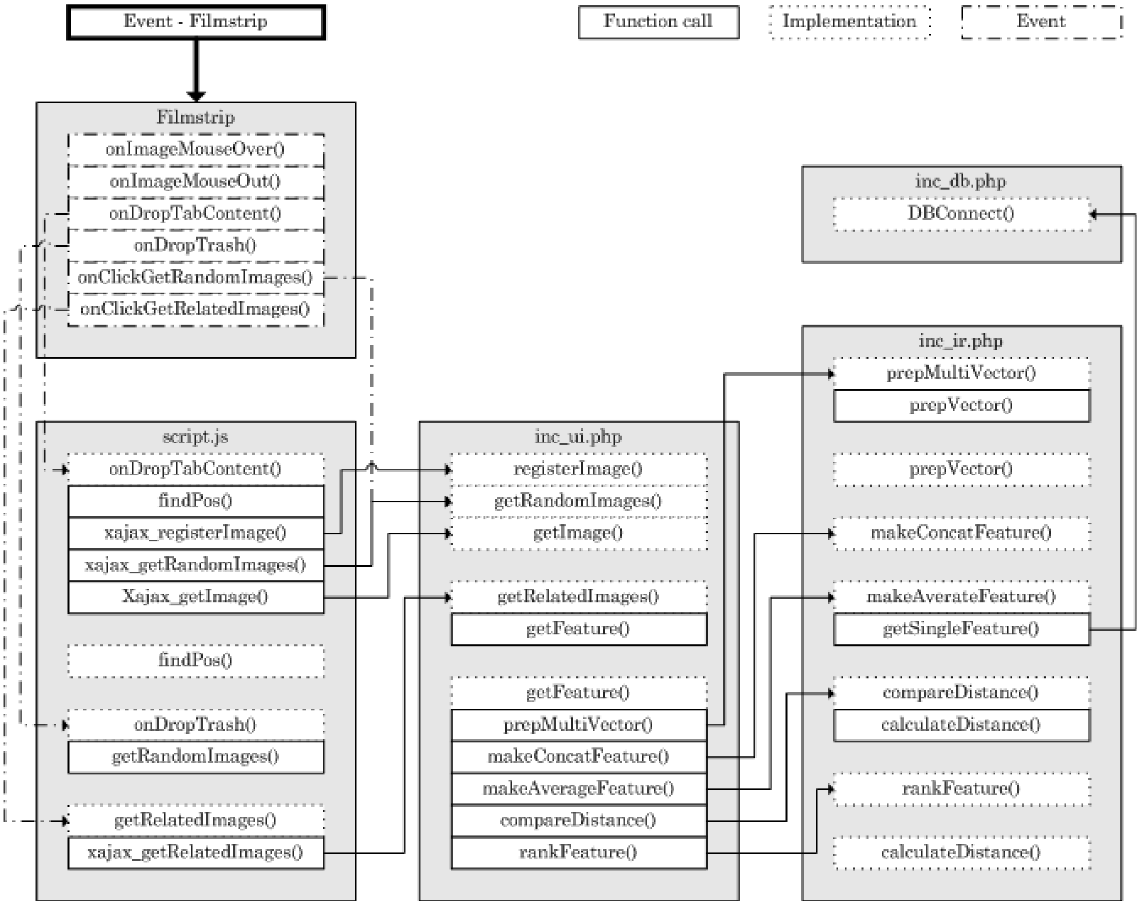


Figure A.6: Implementation of the PRISM filmstrip events

filmstrip.

The *Tab* events (Figure A.7) are either creating a new tab, switching the current tab, or changing the text annotation of a tab.

The *Image* events (Figure A.6) are available for images that have been registered in the canvas area of PRISM. Here, images may be moved to new locations within the canvas, moved to the trash and deleted, scaled (larger or smaller), or annotated. Furthermore, a new view showing more detailed image information may

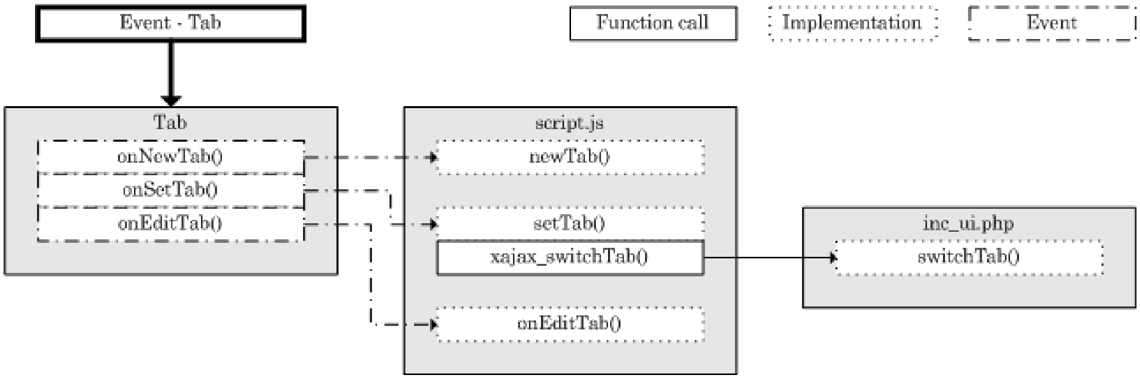


Figure A.7: Implementation of the PRISM tab events

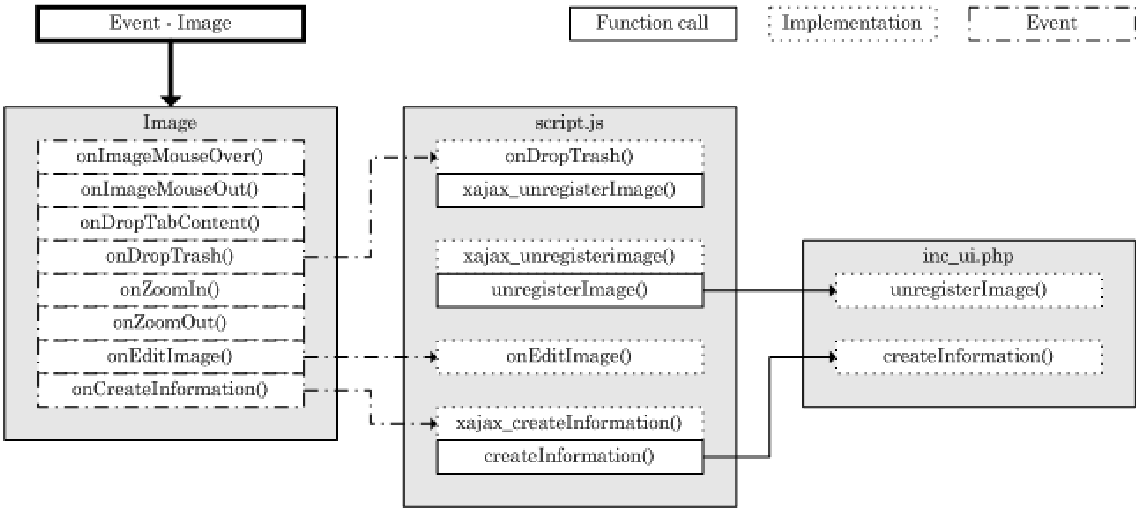


Figure A.8: Implementation of the PRISM image events

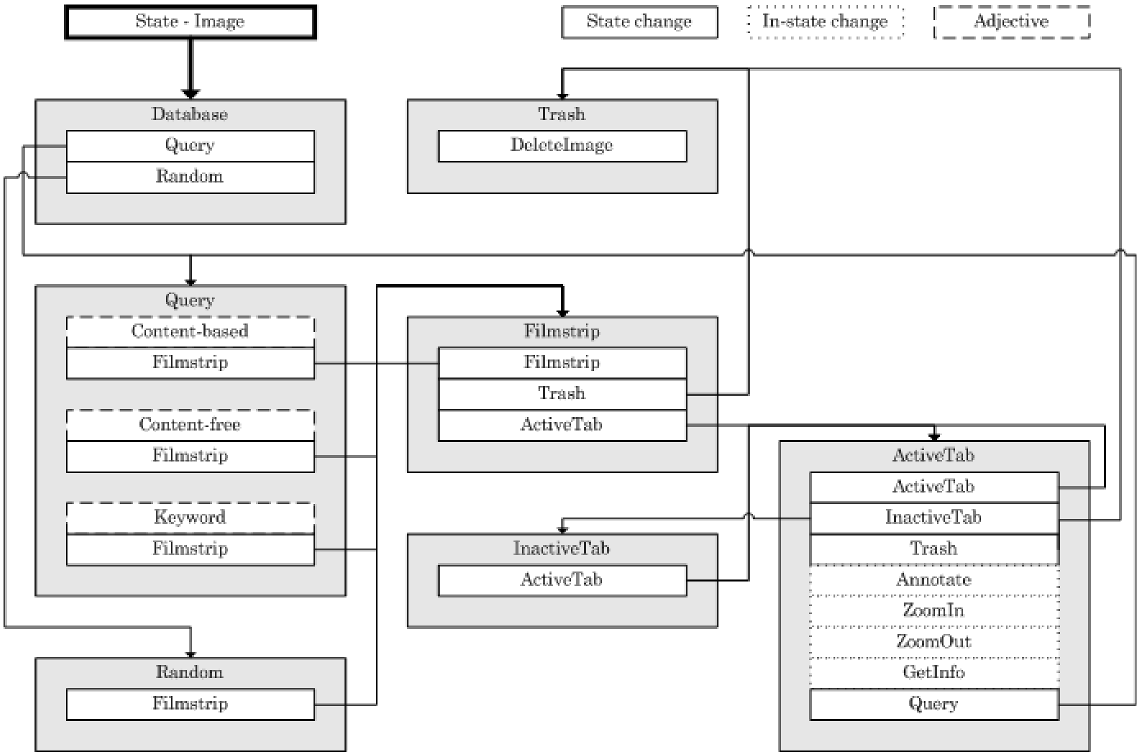


Figure A.9: States of an image in PRISM

be requested.

A.5 States

At any time, an image in PRISM may be in one of seven states (Figure A.9):

- **Database**: the image is idle in the image archive and has not yet been acted upon by the system. It can enter the live system through either a related images query, or being called randomly
- **Query**: the image has been requested in response to a user query

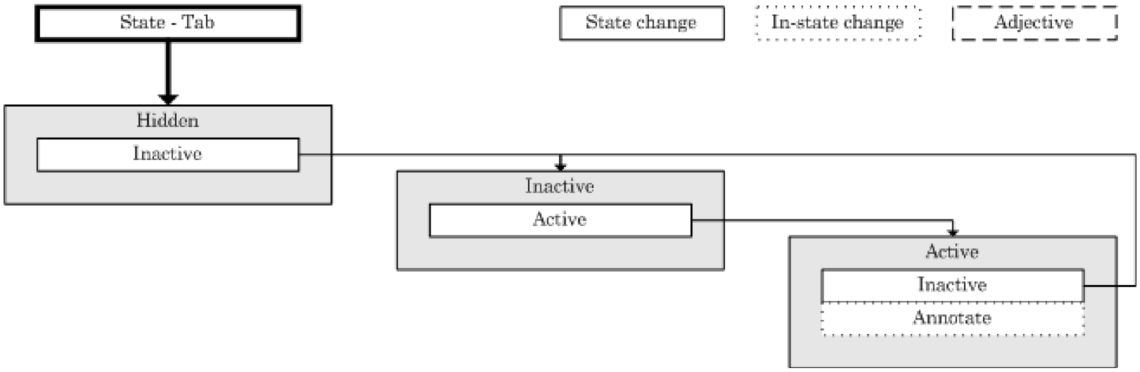


Figure A.10: States of a tab in PRISM

- **Random**: the image has been requested due to a need for a new, random image

- **Filmstrip**: the image is in the filmstrip. From here it may stay in the filmstrip, move to the trash, or move to the active tab

- **Trash**: the image has been deleted and is removed from the user's view of the system

- **Active tab**: the image is in the active tab. It may be manipulated within the active tab (scaled, annotated, etc.), deleted (moved to the trash), or to an inactive tab (if the user switches tabs)

- **Inactive tab**: an image enters the inactive tab state when a user switches tabs. An image in this state cannot be manipulated or deleted, it can only enter the active tab state.

A tab in PRISM has three possible states (Figure A.10):

- **Hidden**: the tab is not available to the user. It is an extra tab pre-created by the system. It may become an inactive tab if the user selects the "New tab" button
- **Inactive**: the tab is visible to the user, but is not the current active tab.
- **Active**: the tab is the currently visible to the user and its associated canvas area and images are also available. In this state the tab itself can also be annotated.

A.6 REFEREE

REFEREE is the administration utility of the PRISM Game. It was designed to be easy to use by someone familiar with the system. The following functions are included in REFEREE:

- `showHeader()`: prints global header information to the screen, such as the REFEREE mark at the top of the screen and the menu available to the REFEREE administrator
- `showBody()`: decodes the action parameter from the URL and executes the action
- `showSingleImage()`: displays detailed information for a single image. Executes six queries: retrieves associated keywords (normal weighing, TF-IDF

weighing, and LSA17 weighing) and associated collaborative filtering information (normal weighing, TF-IDF weighing, and LSA17 weighing). The display of all results is scaled and represented as tag clouds with higher weights resulting in larger displays.

- `showAllImages()`: displays thumbnails of all images in the database

- `setKeywords()`: allows the viewing and adjustment of keywords associated with images in the database for text retrieval. This function has two modes. If POST form data has been received then it is processed as amended keywords and the database is updated. Otherwise, each image is displayed alongside its associated keywords.

- `setCF()`: allows the viewing and adjustment of virtual keywords associated with images in the database for collaborative filtering. This function has two modes. If POST form data has been received then it is processed as amended collaborative information and the database is updated. Otherwise, each image is displayed alongside its associated virtual keywords.

- `setImageOrder()`: allows the order of images displayed in the filmstrip to be specified. Although implemented, this functionality has not yet been used in GAME.

- `setRules()`: displays all rules and parameters, allowing the amendment of their values.

- `getResults()`: queries the database for all completed trials of GAME and displays the results. Clicking a results displays its score report in SITE.

BIBLIOGRAPHY

[1] P. Baldi, P. Frasconi, and P. Smyth. *Modeling the Internet and the Web: Probabilistic Methods and Algorithms.* John Wiley & Sons, May 2003.

[2] A. Bamidele and F. Stentiford. An attention based similarity measure used to identify image clusters. In *2nd Euro. Workshop on the Integration of Knowledge, Semantics and Digital Media Technology*, 2005.

[3] I. Biederman. Perceiving real-world scenes. *Science*, 177(4043):77–80, 1972.

[4] G. B. Borba, H. R. Gamba, L. M. Mayron, and O. Marques. Integrated global and object-based image retrieval using a multiple examples query schema. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, Barcelona, Spain, March 2007.

[5] A. P. Bradley and F. W. M. Stentiford. JPEG 2000 and region of interest coding. *Digital Imaging Computing Techniques and Applications, Melbourne, Australia*, Jan 2002.

[6] V. Bush. As we may think. *The Atlantic Monthly*, 176(1):101–108, 1945.

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

[8] S.-F. Chang, J. R. Smith, M. Beigi, and A. Benitez. Visual information retrieval from large distributed online repositories. *Communications of the ACM*, 40(12):63–71, Dec. 1997.

[9] L. Chen and F. W. M. Stentiford. An attention based similarity measure for colour images. In *ICANN (2)*, pages 481–487, 2006.

[10] Y. Chen and J. Z. Wang. A region-based fuzzy feature matching approach to content-based image retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(9):1252–1267, 2002.

[11] Y. Chen, J. Z. Wang, and R. Krovetz. CLUE: Cluster-based retrieval of images by unsupervised learning. *IEEE Transactions on Image Processing*, 14(8):1187–1201, Aug 2005.

[12] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin. Combining content-based and collaborative filters in an online newspaper. In *Proceedings of ACM SIGIR Workshop on Recommender Systems*, 1999.

[13] C. Colombo and A. Del Bimbo. Visible image retrieval. In V. Castelli and L. D. Bergman, editors, *Image Databases: Search and Retrieval of Digital Imagery*, chapter 2, pages 11–33. John Wiley & Sons, Inc., New York, NY, USA, 2002.

[14] C. Connor, H. Egeth, and S. Yantis. Visual attention: Bottom-up versus top-down. *Current Biology*, 14:850–852, 2004.

[15] M. Cord, J. Fournier, and S. Philipp-Foliguet. Exploration and search-by-similarity in cbir. *Computer Graphics and Image Processing, 2003. SIBGRAPI 2003. XVI Brazilian Symposium on*, pages 175–182, 12-15 Oct. 2003.

[16] R. Datta, W. Ge, J. Li, and J. Z. Wang. Toward bridging the annotation-retrieval gap in image search. *IEEE MultiMedia*, 14(3):24–35, 2007.

[17] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 2008.

[18] P. Davidson and J. Han. Synthesis and control on large scale multi-touch sensing displays. In *NIME*, pages 216–219, 2006.

[19] J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. In *ICML '06: Proceedings of the 23rd international conference on Machine learning*, pages 233–240, New York, NY, USA, 2006. ACM.

[20] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[21] D. Degroot and J. Broekens. Using negative emotions to impair game play. In *Proceedings of the 15th Belgian-Dutch Conference on Artificial Intelligence*, 2003.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[23] G. Denis and P. Jouvelot. Motivation-driven educational game design: applying best practices to music education. In *ACE '05: Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 462–465, New York, NY, USA, 2005. ACM.

[24] T. Deselaers, D. Keysers, and H. Ney. Features for image retrieval: An experimental comparison. *Information Retrieval*, page in press, 2008.

[25] T. Deselaers, T. Weyand, D. Keysers, W. Macherey, and H. Ney. Fire in imageclef 2005: Combining content-based image retrieval with textual information retrieval. In C. Peters, F. C. Gey, J. Gonzalo, H. Mller, G. J. F. Jones, M. Kluck, B. Magnini, and M. de Rijke, editors, *CLEF*, volume 4022 of *Lecture Notes in Computer Science*, pages 652–661. Springer, 2005.

[26] N. Dhavale and L. Itti. Saliency-based multi-foveated MPEG compression. In *Proc. IEEE Seventh International Symposium on Signal Processing and its Applications, Paris, France*, pages 229–232, Jul 2003.

[27] A. Draper, K. Baek, and J. Boody. Implementing the expert object recognition pathway. *Mach. Vision Appl.*, 16(1):27–32, 2004.

[28] H. Eidenberger and C. Breiteneder. Semantic feature layers in content-based image retrieval: implementation of human world features. *Control, Automation, Robotics and Vision, 2002. ICARCV 2002. 7th International Conference on*, 1, 2002.

[29] W. Einhauser and P. Konig. Does luminance-contrast contribute to a saliency map for overt visual attention? *European Journal of Neuroscience*, 17(5):1089–1097, March 2003.

[30] P. G. B. Enser and C. J. Sandom. Towards a comprehensive survey of the semantic gap in visual image retrieval. In *Proceedings of the Second International Conference on Image and Video Retrieval (CIVR)*, pages 291–299, 2003.

[31] L. Ermi and F. Myr. Fundamental components of the gameplay experience: Analysing immersion. In *DIGRA Conf.*, 2005.

[32] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

[33] M. Everingham, A. Zisserman, C. K. I. Williams, and L. V. Gool. The 2007 PASCAL Visual Object Classes Challenge (VOC2006) Results. Technical report, University of Oxford, 2007.

[34] M. Everingham, A. Zisserman, C. K. I. Williams, L. van Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, G. Dorko, S. Duffner, J. Eichhorn, J. D. R. Farquhar, M. Fritz, C. Garcia, T. Griffiths, F. Jurie, D. Keysers, M. Koskela, J. Laaksonen, D. Larlus, B. Leibe, H. Meng, H. Ney, B. Schiele, C. Schmid, E. Seemann, J. Shawe-Taylor, A. Storkey, S. Szedmak, B. Triggs, I. Ulusoy, V. Viitaniemi, and J. Zhang. The 2005 PASCAL Visual Object Classes Challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment (PASCAL Workshop 05)*, number 3944 in Lecture Notes in Artificial Intelligence, pages 117–176, Southampton, UK, 2006.

[35] H. Fang, T. Tao, and C. Zhai. A formal study of information retrieval heuristics. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–56, New York, NY, USA, 2004. ACM Press.

[36] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples an incremental bayesian approach tested on

101 object categories. In *Proceedings of the Workshop on Generative-Model Based Vision*, Washington, DC, June 2004.

[37] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 264–271, Madison, Wisconsin, June 2003.

[38] M. Flickner et al. Query by image and video content: The QBIC system. In M. T. Maybury, editor, *Intelligent Multimedia Information Retrieval*, chapter 1, pages 7–22. American Association for Artificial Intelligence (AAAI), Menlo Park, CA, 1997.

[39] D. Forsyth, J. Malik, and R. Wilensky. Searching for digital pictures. *Scientific American*, 276(6):XX, June 1997.

[40] O. Friborg, M. Martinussen, and J. H. Rosenvinge. Likert-based vs. semantic differential-based scorings of positive psychological constructs: A psychometric comparison of two versions of a scale measuring resilience. *Personality and Individual Differences*, 40(5):873–884, April 2006.

[41] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design patterns: elements of reusable object-oriented software.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1995.

[42] Google, Inc. Google image search. http://images.google.com.

[43] H. Halpin and H. S. Thompson. One document to bind them: combining xml, web services, and the semantic web. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 679–686, New York, NY, USA, 2006. ACM.

[44] M. J. Halvey and M. T. Keane. An assessment of tag presentation techniques. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 1313–1314, New York, NY, USA, 2007. ACM.

[45] E. Hamilton. JPEG File Interchange Format. *C-Cube Microsystems*, 1992.

[46] J. Henderson, J. Brockmole, M. Castelhano, and M. Mack. Visual saliency does not account for eye movements during visual search in real-world scenes. In R. van Gompel, M. Fischer, W. Murray, and R. Hill, editors, *Eye Movement Research: Insights into Mind and Brain.* Elsevier, 2006.

[47] J. L. Herlocker, J. A. Konstan, and *et al.* An Algorithmic Framework for Performing Collaborative Filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference*, pages 230–237. ACM Press, 1999.

[48] T. Hofmann. Probabilistic latent semantic analysis. In *Proc. of Uncertainty in Artificial Intelligence, UAI'99*, Stockholm, 1999.

[49] D. Hoiem, R. Sukthankar, H. Schneiderman, and L. Huston. Object-based image retrieval using the statistical structure of images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 490–497, 2004.

[50] L. J. Hove. Extending image retrieval systems with a thesaurus for shapes. *Norsk Informatikk Konferanse, Stavanger, Tapir Akademisk Forlag*, 2004.

[51] J. Huang. *Color-spatial Image Indexing and Applications.* PhD thesis, Cornell University, 1998.

[52] E. Hyvnen, S. Saarela, A. Styrman, and K. Viljanen. Ontology-based image retrieval. In *WWW (Posters)*, 2003.

[53] Idee Inc. Idee inc. - the visual search company. http://ideeinc.com.

[54] L. Itti. *Models of Bottom-Up and Top-Down Visual Attention.* PhD thesis, California Institute of Technology, Pasadena, California, Jan 2000.

[55] L. Itti. Automatic attention-based prioritization of unconstrained video for compression. In B. Rogowitz and T. N. Pappas, editors, *Proc. SPIE Human Vision and Electronic Imaging IX (HVEI04), San Jose, CA*, volume 5292, pages 272–283, Bellingham, WA, Jan 2004. SPIE Press.

[56] L. Itti. Automatic foveation for video compression using a neurobiological model of visual attention. *IEEE Transactions on Image Processing*, 13(10):1304–1318, Oct 2004.

[57] L. Itti and P. Baldi. A principled approach to detecting surprising events in video. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 631–637, San Siego, CA, Jun 2005.

[58] L. Itti and C. Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40(10-12):1489–1506, May 2000.

[59] L. Itti and C. Koch. Computational modeling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203, Mar 2001.

[60] L. Itti and C. Koch. Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging*, 10(1):161–169, Jan 2001.

[61] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. on PAMI*, 20(11):1254–1259, Nov 1998.

[62] A. Jaimes, N. Sebe, and D. Gatica-Perez. Human-centered computing: a multimedia perspective. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 855–864, New York, NY, USA, 2006. ACM Press.

[63] S. Jamieson. Likert scales: how to (ab)use them. *Med Educ*, 38(12):1217–1218, December 2004.

[64] D. Jobson, Z. Rahman, and G. Woodell. Properties and performance of a center/surround retinex. *IP*, 6(3):451–462, March 1997.

[65] T. Kanade and S. Uchihashi. User-powered content-free approach to image retrieval. In *Proceedings of International Symposium on Digital Libraries and Knowledge Communities in Networked Information Society 2004 (DLKC04)*, pages 24 – 32, March 2004.

[66] L. Kaufman and P. Rousseeuw. *Finding Groups in Data: an introduction to cluster analysis.* Wiley, 1990.

[67] P. Kay and C. McDaniel. The Linguistic Significance of the Meanings of Basic Color Terms. *Language*, 54(3):610–646, 1978.

[68] T. K. Landauer, P. W. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse Processes*, 25:259–284, 1998.

[69] C. Langreiter. retrievr - search by sketch / search by image. http://labs.systemone.at/retrievr/.

[70] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. In *Proceedings of the Workshop on Statistical Learning in Computer Vision*, Prague, Czech Republic, May 2004.

[71] R. Levering and M. Cutler. The portrait of a common html web page. In *DocEng '06: Proceedings of the 2006 ACM symposium on Document engineering*, pages 198–204, New York, NY, USA, 2006. ACM.

[72] Y. Li and L. Shapiro. Object recognition for content-based image retrieval. http://www.cs.washington.edu/homes/shapiro/dagstuhl3.pdf, 2004.

[73] G. Linden, B. Smith, and J. York. Amazon.com recommendations: item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.

[74] C. A. Lindley. The gameplay gestalt, narrative, and interactive storytelling. In *Computer Games and Digital Cultures Conference*, Tampere, Finland, June 2002.

[75] D. Liu and T. Chen. Content-free image retrieval using bayesian product rule. In *IEEE International Conference on Multimedia & Expo*, 2006.

[76] K.-K. Liu, W. Meng, and C. Yu. Discovery of similarity computations of search engines. In *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*, pages 290–297, New York, NY, USA, 2000. ACM Press.

[77] Y.-H. Lu, X.-H. Zhang, J. Kong, and X.-F. Wang. A novel content-based image retrieval approach based on attention-driven model. *Wavelet Analysis and Pattern Recognition, 2007. ICWAPR '07. International Conference on*, 2:510–515, 2-4 Nov. 2007.

[78] W. Y. Ma and B. S. Manjunath. Edge flow: A framework of boundary detection and image segmentation. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 744, Washington, DC, USA, 1997. IEEE Computer Society.

[79] W. Y. Ma and B. S. Manjunath. Netra: a toolbox for navigating large image databases. *Multimedia Systems*, 7(3):184–198, May 1999.

[80] W.-Y. Ma and H. J. Zhang. Benchmarking of image features for content-based retrieval. In *Conference Record of the Thirty-Second Asilomar Conference IEEE on Signals, Systems & Computers, 1998*, volume 1, pages 253–257, 1998.

[81] J. Machrouh and P. Tarroux. Attentional mechanisms for interactive image exploration. *EURASIP Journal on Applied Signal Processing*, 14:23912396, 2005.

[82] J. MacQueen. Some methods for classification and analysis of multivariate observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1(281-297):14, 1967.

[83] O. Marques and B. Furht. *Content-Based Image and Video Retrieval.* Kluwer Academic Publishers, Boston, MA, 2002.

[84] O. Marques and L. M. Mayron. How can the semantic web improve the acquisition and sharing of knowledge? *International Journal of Technology, Knowledge and Society*, 2:129–142, 2006.

[85] O. Marques, L. M. Mayron, G. B. Borba, and H. R. Gamba. On the potential of incorporating knowledge of human visual attention into cbir systems. In *Special Session on Perceptual Visual Processing, IEEE International Conference on Multimedia & Expo (ICME 2006)*, Toronto, Canada, July 2006.

[86] O. Marques, L. M. Mayron, G. B. Borba, and H. R. Gamba. Using visual attention to extract regions of interest in the context of image retrieval. In *44th ACM Southeast Conference (ACMSE2006)*, Melbourne, FL, USA, March 2006.

[87] O. Marques, L. M. Mayron, G. B. Borba, and H. R. Gamba. An attention-driven model for grouping similar images with image retrieval applications. *EURASIP Journal on Advances in Signal Processing*, 2007:Article ID 43450, 17 pages, 2007. doi:10.1155/2007/43450.

[88] S. Mavandadi, P. Aarabi, A. Khaleghi, and R. Appel. Predictive dynamic user interfaces for interactive visual search. In *Proceedings of the 2006 IEEE International Conference on Multimedia & Expo (ICME 2006)*, July 2006.

[89] L. M. Mayron, G. B. Borba, V. Nedovic, O. Marques, and H. R. Gamba. A forward-looking user interface for CBIR and CFIR systems. In *IEEE International Symposium on Multimedia (ISM2006)*, San Diego, CA, USA, December 2006.

[90] L. M. Mayron and O. Marques. Design of a web-based interface for image retrieval. In *International Conference on Web Information Systems and Technologies (WEBIST)*, Barcelona, Spain, March 2007.

[91] P. Melville, R. Mooney, and R. Nagarajan. Content-boosted collaborative filtering. In *Proceedings of the ACM SIGIR Workshop on Recommender Systems*, Sep 2001.

[92] H. Müller, S. Marchand-Maillet, and T. Pun. The truth about corel - evaluation in image retrieval. In *CIVR '02: Proceedings of the International Conference on Image and Video Retrieval*, pages 38–49, London, UK, 2002. Springer-Verlag.

[93] H. Müller, W. Müller, D. M. Squire, S. Marchand-Maillet, and T. Pun. Performance evaluation in content-based image retrieval: overview and proposals. *Pattern Recogn. Lett.*, 22(5):593–601, 2001.

[94] MySQL AB. MySQL AB :: The world's most popular open source database. http://mysql.com.

[95] V. Navalpakkam and L. Itti. Sharing resources: Buy attention, get recognition. In *Proc. International Workshop on Attention and Performance in Computer Vision (WAPCV'03), Graz, Austria*, Jul 2003.

[96] V. Navalpakkam and L. Itti. Modeling the influence of task on attention. *Vision Research*, 45(2):205–231, Jan 2005.

[97] Netflix, Inc. Netflix prize: Home. http://netflixprize.com.

[98] R. Newcombe. An interactive bottom-up visual attention toolkit in Java. http://privatewww.essex.ac.uk/ ranewc/research/visualAttentionJava.html.

[99] D. Noton and L. Stark. Scanpaths in eye movements during pattern perception. *Science*, 171:308–311, Jan. 1971.

[100] A. Oliva. Gist of a scene. In L. Itti, G. Rees, and J. Tsotsos, editors, *Neurobiology of Attention*, chapter 41. Academic Press, Elsevier, 2005.

[101] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Generic object recognition with boosting. Technical Report TR-EMT-2004-01, EMT, TU Graz, Austria, 2004. Submitted to the IEEE Transactions on Pattern Analysis and Machine Intelligence.

[102] N. Oren. Reexamining tf.idf based information retrieval with genetic programming. In *SAICSIT '02: Proceedings of the 2002 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology*, pages 224–234, , Republic of South Africa, 2002. South African Institute for Computer Scientists and Information Technologists.

[103] O. Oyekoya and F. Stentiford. Exploring human eye behaviour using a model of visual attention. In *ICPR (4)*, pages 945–948, 2004.

[104] O. Oyekoya and F. Stentiford. Perceptual image retrieval using eye movements. In *IWICPAS*, pages 281–289, 2006.

[105] S. Palmer. *Vision Science: Photons to Phenomenology.* MIT Press, Cambridge, MA, 1999.

[106] K. Papineni. Why inverse document frequency? In *NAACL '01: Second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies 2001*, pages 1–8, Morristown, NJ, USA, 2001. Association for Computational Linguistics.

[107] D. Parkhurst, K. Law, and E. Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision Research*, 42(1):107–123, 2002.

[108] D. Parkhurst and E. Niebur. Texture contrast attracts overt visual attention in natural scenes. *European Journal of Neuroscience*, 19(3):783–789, February 2004.

[109] PASCAL Network. The PASCAL Object Recognition Database Collection. http://www.pascal-network.org/challenges/VOC/databases.html.

[110] P. Paulson and A. Tzanavari. Combining collaborative and content-based filtering using conceptual graphs. In *Modelling with Words*, pages 168–185, 2003.

[111] G. Pell. Use and misuse of likert scales. *Medical Education*, 39(9):970–970, September 2005.

[112] R. Peters, A. Iyer, L. Itti, and C. Koch. Components of bottom-up gaze allocation in natural images. *Vision Research*, 45(18):2397–2416, 2005.

[113] T. PHP Group. PHP: Hypertext Processor. http://php.net.

[114] Picsearch. Picsearch - image search for pictures and images.

[115] M. Prensky. The motivation of gameplay: The real twenty-first century learning revolution. *On the Horizon, NCB University Press*, 9(5), October 2002.

[116] Z. W. Pylyshyn. *Seeing and Visualizing: It's Not What You Think*. MIT Press, Cambridge, MA, 2006.

[117] L. W. Renninger and J. Malik. When is scene identification just texture recognition? *Vision Research*, 44(19):2301–2311, September 2004.

[118] Riya Inc. Like visual search - find things by appearance with our new likeness technology. http://like.com.

[119] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. *MIT AI Lab Memo AIM-2005-025*, 2005.

[120] U. Rutishauser, D. Walther, C. Koch, and P. Perona. Is bottom-up attention useful for object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11–37, 2004.

[121] I. Rybak, V. Gusakova, A. Golovan, L. Podladchikova, and N. Shevtsova. A model of attention-guided visual perception and recognition. *Vision Research*, 38(15-16):2387–2400, August 1998.

[122] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, 1988.

[123] S. Santini and R. Jain. The graphical specification of similarity queries. *Journal of Visual Languages and Computing*, 7(4):403–421, 1997.

[124] G. Schaefer and M. Stich. UCID - An Uncompressed Colour Image Database. Technical report, School of Computing and Technology, The Nottingham Trent University, Nottingham, United Kingdom, 2003.

[125] G. Schaefer and M. Stich. Ucid - an uncompressed colour image database. In *Storage and Retrieval Methods and Applications for Multimedia 2004*, volume 5307 of *Proceedings of SPIE*, pages 472–480, 2004.

[126] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260, New York, NY, USA, 2002. ACM Press.

[127] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition

and segmentation. In *Proceedings of European Conference Computer Vision (ECCV)*, 2006.

[128] C. Siagian and L. Itti. Biologically-inspired face detection: Non-brute-force-search approach. In *First IEEE-CVPR International Workshop on Face Processing in Video*, pages 62–69, Jun 2004.

[129] P. Singh, T. Lin, E. T. Mueller, G. Lim, T. Perkins, and W. L. Zhu. Open mind common sense: Knowledge acquisition from the general public. In *On the Move to Meaningful Internet Systems, 2002 - DOA/CoopIS/ODBASE 2002 Confederated International Conferences DOA, CoopIS and ODBASE 2002*, pages 1223–1237, London, UK, 2002. Springer-Verlag.

[130] A. Singhal. Modern information retrieval: A brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4):35–43, 2001.

[131] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

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

[133] J. Smith and S. Chang. Searching for images and videos on the world-wide web. Technical Report 459-96-25, Center for Telecommunications Research Technical Report, 1996. see http://www.ctr.columbia.edu/webseek/paper/.

[134] F. Stentiford. An estimator for visual attention through competitive novelty with application to image compression. In *Picture Coding Symposium*, pages 24–27, 2001.

[135] F. Stentiford. An attention-based similarity measure with application to content-based information retrieval. In *Proc. of the Storage and Retrieval for Media Databases Conference, SPIE Electronic Imaging*, 2003.

[136] F. Stentiford. A visual attention estimator applied to image subject enhancement and colour and grey level compression. In *ICPR (3)*, pages 638–641, 2004.

[137] F. Stentiford. Attention-based similarity. *Pattern Recognition*, 40(3):771–783, 2007.

[138] G. Strang. *Linear Algebra and Its Applications.* Brooks Cole, February 1988.

[139] E. A. Styles. *Attention, Perception, and Memory: An Integrated Introduction.* Taylor & Francis Routledge, New York, NY, 2005.

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

[141] Y. Tao and W. Grosky. Spatial color indexing: a novel approach for content-based image retrieval. In *Proceedings in IEEE International Conference on Multimedia Computing and Systems*, volume 1, pages 530–535, 1999.

[142] Y. Tao and W. I. Grosky. Image matching using the OBIR system with feature point histograms. In *Fourth Working Conference on Visual Database Systems (VDB)*, pages 192–197, 1998.

[143] S. Tollari and H. Glotin. Web image retrieval on imageval: evidences on visualness and textualness concept dependency in fusion model. In *CIVR '07: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 65–72, New York, NY, USA, 2007. ACM.

[144] A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 762–769, Washington, DC, June 2004.

[145] A. Traina, J. Marques, and C. Traina. Fighting the Semantic Gap on CBIR Systems through New Relevance Feedback Techniques. *Proceedings of the 19th IEEE Symposium on Computer-Based Medical Systems*, pages 881–886, 2006.

[146] S. Uchihashi and T. Kanade. Content-free image retrieval based on relations exploited from user feedbacks. In *IEEE Int'l Conf. on Multimedia and Expo (ICME '05)*, 2005.

[147] VIMA Technologies. VIMA Technologies: Image search and image categorization software based on image content filtering:. http://vimatech.com.

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

[149] L. von Ahn, R. Liu, and M. Blum. Peekaboom: a game for locating objects in images. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 55–64, New York, NY, USA, 2006. ACM Press.

[150] D. Walther, L. Itti, M. Riesenhuber, T. Poggio, and C. Koch. Attentional selection for object recognition – a gentle way. In *Lecture Notes in Computer Science*, volume 2525, pages 472–479, Nov 2002.

[151] J. Z. Wang, J. Li, and G. Wiederhold. Simplicity: Semantics-sensitive integrated matching for picture libraries. In *VISUAL '00: Proceedings of the 4th International Conference on Advances in Visual Information Systems*, pages 360–371, London, UK, 2000. Springer-Verlag.

[152] T. J. Williams and B. A. Draper. An evaluation of motion in arti.cial selective attention. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, page 85, Washington, DC, USA, 2005. IEEE Computer Society.

[153] Y. M. Wong, S. C. H. Hoi, and M. R. Lyu. An empirical study on large-scale content-based image retrieval. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 2206–2209, July 2007.

[154] Xajax. Xajax php class library - the easiest way to develop asynchronous ajax applications with php. http://www.xajaxproject.org, 2006.

[155] Yahoo! Inc. Welcome to flickr - photo sharing. http://flickr.com.

[156] R. Zhao and W. Grosky. Narrowing the semantic gap - improved text-based web document retrieval using visual features. *IEEE Transactions on Multimedia*, 4(2):189–200, Jun 2002.

[157] R. Zhao and W. I. Grosky. Negotiating the semantic gap: from feature maps to semantic landscapes. *Pattern Recognition*, 35(3):593–600, 2002.

[158] J. Zobel and A. Moffat. Inverted files for text search engines. *ACM Comput. Surv.*, 38(2):6, 2006.

[159] J. Zobel, A. Moffat, and R. Sacks-Davis. Searching large lexicons for partially specified terms using compressed inverted files. In *VLDB '93: Proceedings of the 19th International Conference on Very Large Data Bases*, pages 290–301, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.