Chapter 1

Introduction

If there is a defining characteristic of the new digital communications media, that characteristic is the potential for interactivity [116, 117]. In the digital world, communication is synonymous with computation: fast processors are required at both ends of the pipe to transform the massive amounts of audio, video, and textual information characteristic of modern applications into a compact and error-resilient bitstream which can be transmitted rapidly and reliably. Digital decoders are therefore intelligent, giving the user the ability to actively search, browse through, or even publish information, instead of passively "tuning in" to what is going on a broadcast channel.

Such a shift with regards to the control over the communications process unveils a new universe of requirements and possibilities for representing audio-visual content. Because digital media are characterized by ubiquitous networking, computation, storage capacity, and absence of geographical barriers, the user has instantaneous access to a virtually unlimited amount of information. But while the ability to tap into such an infinite source of resources is exciting, it can also lead, in the absence of appropriate indexing and navigation mechanisms, to a significant degree of frustration and helplessness.

A significant challenge is, therefore, to design representations that can support not only efficient transmission and storage of information but also filtering, sorting, classification, retrieval, summarization, browsing, and manipulation of content. The challenge is particularly strong when the content to represent does not lend itself to unambiguous textual descriptions. Today, we simply cannot understand the structure contained in a sound recording or an image, and this limits our ability in various areas.

The issues of image understanding and representation are central to this thesis, where we address the problem of how to design systems for retrieving information from large image repositories. While text can be a powerful aid, experience reveals that traditional text-based search engines fall short of fulfilling all the requirements of visual retrieval. The problem is that images can have multiple interpretations and text annotations usually say more about the interpretation of the person that created them than the one that may be relevant for someone else's search. The alternative that we pursue here is to design *content-based image retrieval* (CBIR) systems, i.e. systems that allow users to express their queries directly in terms of visual attributes.

While, ideally, we would like CBIR systems to understand natural language scene descriptions, e.g. "show me all the pictures of a tiger running in the wild," we simply do not yet understand images well enough for this to be feasible. The only viable alternative, in the short-term, is to request less from the machine and more from the users. An increasingly popular solution [129, 43, 118, 32, 7, 164] is to build systems that can make judgments of *visual similarity* and place on the user the burden of guiding the search. This is accomplished through an iterative process where the user provides examples, the machine suggests matches and, from those, the user selects the next round of examples. The obvious advantage is that the CBIR system is now much simpler to design. The main drawback is that, because visual similarity is not the same as *semantic similarity*, the matches returned by the machine will not always be what the user is looking for. Since this increases the risk of user frustration, the next step is to give retrieval systems the ability to react intelligently to the user feedback, i.e. to *learn* from the user interaction [132].

There are, therefore, two fundamental problems to be addressed. First, the design of the visual recognition architecture itself and, second, the design of learning mechanisms to facilitate the user interaction. Obviously, the two problems cannot be solved in isolation since the careless selection of the recognition architecture will make learning more difficult and vice-versa.

1.1 Contributions of the thesis

This thesis presents a unified solution to visual recognition and learning by formulating recognition as a decision theoretical problem, where the goal is to *minimize the probability of retrieval error*. Besides providing an objective performance criteria for the design and evaluation of retrieval systems, the new formulation also leads to solutions that are optimal in a well-defined sense, and can be derived from the well understood principles of Bayesian inference. The resulting Bayesian recognition architecture has several attractives. First, it is based on a universal recognition language (the language of probabilities) that provides a computational basis for the integration of information from multiple content sources and modalities. In result, it becomes possible to build systems that simultaneously account for text, audio, video, or any other modalities. Second, because learning is a consequence of the ability to integrate information over time, this language also provides a basis for designing learning algorithms. It therefore becomes possible to design retrieval systems that can rely on user-feedback both to retrieve images faster and to adapt themselves to their users' preferences. Third, all the integration can be performed through belief propagation algorithms that, although optimal in the decision-theoretic sense, are extremely simple, intuitive, and easy to implement. Finally, as an architecture for visual recognition, it generalizes current retrieval solutions, solving some of the most challenging problems faced by these: joint modeling of color and texture, objective guidelines for controlling the trade-off between feature transformation and feature representation, and unified support for local and global queries without requiring image segmentation.

1.1.1 An architecture for visual recognition

An architecture for visual information retrieval is composed by three fundamental building blocks: 1) a transformation from the image pixel space into a feature space that provides sufficient discrimination between the image classes in the database, 2) a feature representation that describes how each image populates the feature space, and 3) a retrieval metric that relies on this feature representation to infer similarities between images. Even though significant attention has been recently devoted to each of these individual components, there have been significantly fewer attempts to investigate the interrelationships among them and how these relationships may affect the performance of retrieval systems.

In fact, current retrieval solutions can be grouped into two major disjoint sets: on one hand, a set of representations that evolved from texture analysis and are tailored for texture and, on the other hand, a set that grew out of object recognition and is tailored for color. We refer to the former as *texture-based retrieval* and to the latter as *color-based retrieval*. Retrieval approaches in these two classes vary widely with respect to the emphasis placed on the design of the individual retrieval components. For example, because most texture databases consist of homogeneous images, it is reasonable to assume that the associated features will be Gaussian distributed. The Gaussian density is thus commonly used for feature representation (although usually in implicit form), and simple metrics such as the Euclidean or the Mahalanobis distance serve as a basis to evaluate similarity. Given these feature representation and similarity function, the main goal of texture analysis is to find the set of features that allow best discrimination between the texture classes in the database. This goal can be made explicit in the formulation of the problem [176, 183, 40] or implicit [134, 94, 104, 96, 102, 109].

Unlike texture-based retrieval, feature selection has not been a critical issue for color-based retrieval, where the features are usually the pixel colors themselves. Instead a significant amount of work has been devoted to the issue of feature representation, where several approaches have been proposed. The majority of these are variations on the color histogram initially proposed for object recognition [172], e.g. the color coherence vector [126], the color correlogram [66], color moments [167], etc. While each feature representation may require a specific similarity function, the most commonly used are L^p distance norms in feature representation space. Among these, the L^1 distance between color histograms, also known as histogram intersection [172], has become quite popular.

While they have worked well in their specific domains, these representations break down when applied to databases of generic imagery. The main problem for texture-based solutions is that, since generic images are not homogeneous, their features cannot be accurately modeled as Gaussian and simple similarity metrics are no longer sufficient. On the other hand, color-based solutions are plagued by the exponential complexity of the histogram on the dimension of the feature space, and are applicable only to low-dimensional features (e.g.

pixel colors). Hence, they are unable to capture the spatial dependencies that are crucial for texture characterization.

In the absence of solutions that can account for both color and texture, retrieval systems must resort to different features, representations, and similarity functions to deal with the two image attributes [43, 118, 164, 44, 175], making it difficult to perform joint inferences with respect to both. The standard solution is to evaluate similarity according to each of the attributes and obtain an overall measure by weighting linearly the individual distances. This opens up the question of how to weigh different representations on different feature spaces, a problem that has no easy solution.

Ideally, one would like to avoid these problems altogether by designing a retrieval architecture capable of accounting for both color and texture. An obvious, but often overlooked, statement is that carefully designing any of the individual retrieval modules is not sufficient to achieve a good overall solution. Instead, the design must start from an unambiguous performance criteria, and all modules designed with the goal of optimizing the overall performance. In this context, we start by posing the retrieval problem as one of optimal decision-making. Given a set of database image classes and a set of query features, the goal is to find the map from the latter to the former that *minimizes the probability of retrieval error*.

This is shown to be interesting in two ways. First, it leads to a Bayesian formulation of retrieval and a probabilistic retrieval criteria that either generalizes or improves upon the most commonly used similarity functions (Mahalanobis distance, L^p norms, and minimum discrimination information, among others). Second, it provides objective guidelines for the selection of both the feature transformation and representation. The first of these guidelines is that the most restrictive constraints on the retrieval architecture are actually imposed on the feature transformation. In fact, optimal performance can only be achieved under a restricted set of *invertible transformations* that leaves small margin for feature optimization. The second guideline is that performance quality is directly related to the quality of density estimates, which is in turn determined by the feature representation.

A corollary, of great practical relevance, of these guidelines is that there is less to be gained from the feature transformation than from accurate density estimation. This, and the fact that the difficulty of the latter increases with the dimensionality of the space, motivates us to restrict the role of the former to that of dimensionality reduction; i.e. we seek the feature transformation that achieves the optimal trade-off between invertability and dimensionality reduction. This leads to the well known principal component analysis which is, in turn, well approximated by perceptually more justifiable multi-resolution transformations that are common in the compression literature, e.g. the discrete cosine transform (DCT).

On the other hand, we devote significant attention to the issue of feature representation. We notice that a good representation should be 1) expressive enough to capture the details of multi-modal densities that characterize generic imagery, and 2) compact enough to be tractable in high-dimensional spaces. Viewing the standard Gaussian and histogram representations as particular cases of the generic family of mixture models reveals that insufficient number of basis functions, poor kernel selection, and inappropriate partitioning of the space are the major reasons behind their inability to meet these requirements. The Gaussian mixture then emerges as a unifying feature representation for both color and texture, eliminating the problems associated with the standard approaches. Further observation that a mixture model defines a family of embedded densities leads to the concept of embedded multi-resolution mixtures (EMM). These are a family of embedded densities ranging over multiple image scales that allow explicit control of the trade-off between spatial support and invariance.

Overall, the retrieval architecture composed by the Bayesian similarity criteria, the DCT feature transformation, and the embedded mixture representation provides a good trade-off between retrieval accuracy, invariance, perceptual relevance of similarity judgments, and complexity. We illustrate all these properties with an extensive experimental evaluation on three different databases that stress different aspects of the retrieval problem[1]: the Brodatz texture database, the Columbia object database, and the Corel database of stock photography. In all cases, the new approach outperforms the previous best retrieval solutions both in terms of objective (precision/recall) and subjective (perceptual) evaluation.

[1]The experimental set up is discussed in detail in Appendix A.

1.1.2 Learning from user interaction

While a sophisticated architecture for evaluating image similarity is a necessary condition for the design of successful retrieval systems, it is by no means sufficient. The fact is that the current understanding of the image analysis problem is too shallow to guarantee that any retrieval system (no matter how sophisticated) will always find the desired images in response to a user query. In result, retrieval is usually an interactive process where 1) the user guides the search by rating the systems suggestions, and 2) the system refines the search according to those ratings. Conceptually, a retrieval system is nothing more than an interface between an intelligent high-level system (the user's brain) that can perform amazing feats in terms of visual interpretation but is limited in speed, and a low-level system (the computer) that has very limited visual abilities but can perform low-level operations very efficiently. Therefore, the more successful retrieval systems will be those that make the user-machine interaction easier.

The goal is to exploit as much as possible the strengths of the two players: the user can provide detailed feedback to guide the search when presented with a small set of meaningful images, the machine can rely on that feedback to quickly find the next best set of such images. To enable convergence to the desired target image, the low-level system cannot be completely dumb, but must know how to *integrate* all the information provided by the user over the entire course of interaction. If this were not the case, it would simply keep oscillating between the image sets that best satisfied the latest indication from the user, and convergence to the right solution would be difficult.

This ability to learn by integrating information must occur over various time scales. Some components maybe hard-coded into the system from the start, e.g. the system may contain a specialized face-recognition module. However, hard-coding leaves small room for *personalization*. Not all users are interested in the same visual concepts and retrieval systems should be able to respond to the individual user requirements. Therefore, most components should, instead, be learned over time. Since users tend to search often for the concepts that are of greatest interest to them, examples of these concepts will be available. Hence, it is in principle possible for the system to build internal concept representations and become progressively more apt at recognizing specific concepts as time progresses. We refer to such mechanisms as *long-term learning* or *learning between retrieval sessions*, i.e. learning that does not have to occur on-line, or even in the presence of the user.

Information must also be integrated over short-time scales, e.g. during a particular retrieval session. In the absence of *short-term* or *in-session learning*, the user would have to keep repeating the information provided to the retrieval system from iteration to iteration. This would be cumbersome and extremely inefficient since a significant portion of the computation performed by the latter would simply replicate what had been done in previous iterations. Unlike long-term learning, short-term learning must happen on-line and therefore has to be fast.

In this thesis, we show that the Bayesian formulation of the retrieval problem leads to very natural procedures for inference and learning. This is not surprising since probabilistic representations are the soundest computational tool available to deal with uncertainty and the laws of probability the only principled mechanism for making inferences in its presence. We illustrate this point by designing both short- and long-term learning mechanisms that can account for both positive and negative user-feedback and presenting experimental evidence that illustrates the clear benefits of learning for CBIR.

1.2 Organization of the thesis

The standard thesis format asks for an introduction chapter, a chapter of literature review, and then a sequence of chapters describing the contributions, experimental validation, and conclusions. In this document, we deviate from this standard format in at least two ways.

First, because one of the points of the thesis is to demonstrate that a significant part of what has been proposed in the retrieval literature are sub-optimal special cases of the Bayesian formulation now introduced, we do not include a standard review chapter. Instead, we establish connections to previous work as we discuss Bayesian retrieval. We believe that this will be easier for the reader than simply including a large review section and referring to it in the chapters ahead. Since the organization of the thesis follows the fundamental structure of a retrieval system, a reader interested in reviewing particular sub-topics (e.g. feature sets commonly used to characterize texture) can simply skip ahead to the corresponding chapter. Second, instead of having a chapter entirely devoted to experimental validation, we include experimental results as we go along. This allows us to build on the experimental results of the earlier chapters to motivate the ideas introduced in subsequent ones.

The organization of the thesis is as follows. In Chapter 2, we introduce the Bayesian retrieval criteria and show that it generalizes or outperforms most of the similarity measures in common use in the retrieval literature. In Chapter 3, we discuss how Bayesian similarity provides guidelines for feature selection and representation and discuss previous retrieval strategies in light of these guidelines. We conclude that the two most prevalent strategies have strong limitations for retrieval from generic image databases and devise an alternative strategy. This strategy is then implemented in Chapters 4 and 5 where we present solutions for feature transformation and representation.

The issue of local vs. global similarity is addressed on Chapter 6, where we show that Bayesian retrieval provides a natural solution to local queries. However, we also point out that for global queries the straightforward implementation of the Bayesian criteria is usually too expensive. To correct this problem, we devise efficient approximations that are shown to achieve similar performance to that of the exact Bayesian inferences. The ability to account for local queries is a requirement for learning, which is then discussed in Chapters 7 and 8. In Chapter 7, we present a short-term learning algorithm that can exploit both positive and negative user feedback to achieve faster convergence to the target images. In Chapter 8, we present a long-term learning algorithm that gives retrieval systems the ability to, over time, tailor themselves to the interests of their users.

Finally, in Chapter 9 we describe the practical implementation of all the ideas in the "Retrieval as Bayesian Inference" (RaBI) image retrieval system, and we present conclusions and directions for future work in Chapter 10. A discussion of the experimental set up used to evaluate retrieval performance is presented in Appendix A.