Without culture, and the relative freedom it implies, society, even when perfect, is but a jungle. This is why any authentic creation is a gift to the future.

– Albert Camus.

University of Alberta

Image Cultural Analytics Through Feature-Based Image Exploration and Extraction

by

Parisa Naeimi

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

Fall 2011
Edmonton, Alberta

Examining Committee

Eleni Stroulia, Computing Science

Ken Wong, Computing Science

Geoffrey Rockwell, Humanities Computing and Philosophy

To the image that speaks louder than thousand words

Abstract

Today, we are witnessing the proliferation of digital media, whether through mass digitization efforts, or through the publishing activities of the multitude of social-platform users. Much of this data is not the product of "formal work", but rather the product of artistic and social activity and the nascent field of "cultural data mining" focuses on methods for analyzing this data to extract and visualize interesting patterns that give us insights of how our culture evolves. This field adopts and combines methods from image processing, data mining, probability and statistics, and visualization to extract and represent information about culture.

In this research, we report on a image feature-based approach to the cultural-analytics problem, whereby we analyze images in terms of a set of low-level and mid-level features, *i.e.,* colors, textures, and shapes, and examine the correlations of these features with high-level semantic information of cultural significance. In our current implementation, using a faceted search interface, a combination of one or more visual features or tags can be used to perform image- and/or tag-based queries on an image repository. Our results on some sample data sets, obtained from publicly available sources, show that aggregated image features can be used for inferring higher-level cultural information.

Acknowledgements

There are many people who I would like to thank them all for their generous help and share of their knowledge and experience with me during this journey. Without their support and guidance, accomplishing this study seemed impossible.

First of all, I would like to express my utmost gratitude to my supervisor Professor Eleni Stroulia for her support and constructive feedbacks during this research. She gave me the opportunity to explore this subject area and insightfully guided me throughout the way; I can not find any appropriate word to express my deepest feeling to her.

I am also very grateful to have Professor Stan Reucker as my advisor who supported me from the early stages of this project by helping with defining the scope of the project, preparing two initial Flickr data sets, and advising on some experiments on Flickr images. In addition, I would like to thank Professor Geoffrey Rockwell and Joyce Yu for providing the video game data set.

Many thanks go to my examining committee Dr. Ken Wong and Dr. Geoffrey Rockwell for their time, effort, and feedbacks on this thesis. Also I like to thank Amin Jorati for useful comments on image similarity modeling.

As a member of SERL group, I would like to thank all the amazing people of this group. I do appreciate all the cooperation and fun moments we had together. You made this journey very enjoyable to me.

Finally, I would love to thank my lovely parents for their lifetime support and encouragement. All my sincere thanks go to my dear Hossein for his support, encouragement, and constructive comments on this research as well.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

The impact of the digital revolution in our era is undeniable. Letters and mail has been replaced by e-mail; telephone calls have been supplemneted by instant messaging; friendships are transferred to or built over social-networking sites; massive amounts of digitized and born-digital audio and video content can be found on line. As the earlier Internet services, such as email, are enhanced and newer content-sharing and social-networking services such as YouTube, Flickr, Facebook, and Google+ are introduced, different aspects of our personal, social and business life are increasingly affected by these changes. It is becoming increasingly the case that we cannot imagine a day without access to these services or if that occurs we feel a significant interruption in many parts of our daily activities. We are arbitrarily or necessarily so exposed to the web that a track of most of our real-life activities can be found in the form of digital content accumulated on the web. In general, it seems that the web can be considered as a projection, even though imperfect, of our real-life relationships and activities to the digital space.

1.1 Motivation and Problem

As much of our communications and interactions are digital, we can learn a lot about our civilization from this huge collection of information available on the web, and this is why, according to Manovich, it is time to apply data analysis, mining, and visualization methods to the huge cultural content available on the web to inspect and understand cultural trends [14]. The field of cultural analytics puts forward a

methodology of applying computational methods of analysis, mining and visualization to the study of culture, motivated by two important reasons. The first reason is that so much of our cultural production nowadays is, and is increasingly becoming, digital. The second one, and probably more important, is that this new methodological perspective could bring a new type of systematicity in cultural studies and generate new insights that could spark further enthusiasm in the field.

The core research objective of the cultural-analytics agenda is to turn the products of our culture into data that can be automatically processed and analyzed. And as a massive amount of easily available, including cultural artifacts are digital images, an important question arises: "what information about our culture and its evolution might be inferred from visual data, and how?". This is a very challenging question as the types of information of interest to those who analyze cultural phenomena are rather abstract and not obviously related to the low-level features of images. In contrast, the low-level features can be automatically extracted from images and videos using standard (well-known) image and video processing techniques. These standard image-processing tools and techniques are rather mechanical in the sense that a series of operations are applied on the input visual data to extract some structural indicators or transform the input into another visual representation. Taking these into account, then the above general question might be broken into a series of questions as follows:

1. How might one relate low-level, automatically (easily) extractable features to high-level abstract cultural semantics or concepts?
2. Is it necessary to explicitly establish these relations or can some trends be found just by analysis of these low-level features collectively?
3. How can higher-level images features help to improve the results? And how may they affect the system's quantitative and qualitative performance?

These are the types of questions that we attempt to answer, at least partially, in this thesis. In our approach to the problem we will see how some basic image features such as image-intensity statistics can be used to find some interesting insights and trends among fairly large image data sets. We also show the importance and challenges of involving higher-level image features in analysis process.

1.2 Contributions

We approach the task of cultural analytics as one of recognizing culture-specific patterns and trends in image collections relying on image characteristics, contents, and features. In this regard, we have developed a web-based prototype that integrates a variety of components for extraction of different features, namely color, textures, shapes and objects, from images. These features can potentially be used for inferring higher-level information relevant to cultural trends. The system supports interactive exploration of large image data sets through similarity-based image clustering using the extracted features and/or image meta-data.

We have developed our system prototype based on existing open-source image-processing libraries including ImageJ[1] and OpenCV[2]. In specific, we use ImageJ for basic image-processing tasks and low-level features extraction whereas OpenCV is used for higher-level image features detection. In some cases, for example in face and corner detection, we have evolved the original implementations to improve the performance of the original feature extractors.

To calculate the similarity of images, we break the image into a set of blocks and use the features across these blocks in these calculations. Turning the similarity calculation into an assignment problem, we have developed, to some extent, scale- and rotation-invariant distance metrics for each of these features, and methods for calculating image similarities from several sets of extracted features.

We have integrated these components within a web-based prototype with visual faceted search in order to support image exploration and clustering. These facets are defined with respect to the image features extracted and stored in the data base. The interface enables image-based query where the user can upload one or more images with regard to selected facets to retrieve the desirable results. These visual facets can be combined with the textual facet to explore the image data set in order to examine some hypothesis, classifying images, or selecting subsets of the images for further analysis.

Finally, we have used our prototype to explore some questions inspired from the

[1] Image processing and analysis in Java, `http://rsbweb.nih.gov/ij/`
[2] OpenCV Wiki, `http://opencv.willowgarage.com/wiki/`

cultural-analytics field in sample data sets obtained from free data sources available on the web such as IMDB and Flickr. We have used different APIs and open source libraries to extract images and associated data from these image repositories. Our experimental and analysis results reveals interesting pattern and trends within these data showing the applicability and soundness of our approach to the problem.

1.3 Outline

The remaining parts of this thesis are organized as follows.

Chapter 2 reviews earlier work in four areas including content-based image retrieval (CBIR), searching and organizing image data sets, faceted search, and cultural analytics that are closely related to this research topic. There is much interesting research in these areas, especially in CBIR and faceted search, and our focus and special attention is on the most recent and the closely related ones.

Chapter 3 describes our feature-based approach to image cultural analytics in detail. Specifically it presents the process of extracting low-level features including color-, texture- and structure-related features, and some higher-level object features. We describe in detail how these features might be related to some cultural phenomena, events, or concepts and see how they can be used collectively to explore the data set. We also discuss our method of calculating image similarities from a given collection of features based on modeling it as an assignment problem.

Chapter 4 explains the architecture of the prototype system. It focuses on our three-tier architecture namely presentation layer, business logic layer, and the persistence layer. We explain the different components of these layers and their interrelationships and briefly describe the different open source components we have integrated into the system.

Chapter 5 describes our experience with using our system functionalities in examining and extracting some typical cultural patterns. We have implemented some extractor tools for retrieving images and their meta-data from publicly available data sets such as Flickr and IMDB. We present our experimental results on some of these data sets and our data from some other resources including a small set of top

100 video games.

Chapter 6 is devoted to concluding remarks and possible future work and extensions. Specifically, we explain how integrating a statistical analysis and visualization component into our prototype system can turn our system into a generic, valuable tool for image data exploration and cultural analytics.

Chapter 2

Background and Related Work

This chapter reviews the related work and background in four specific areas related to the topic of this research. The first section discusses research and advancements in the area of content-based image retrieval, including retrieval techniques and similarity measures. The second section reviews the strategies for searching, organizing, and browsing large image data sets. The third section presents advances in faceted search especially those ones that are directly related to content-based image retrieval. Finally, the last section focuses on the nascent field of cultural analytics and the related advancements.

2.1 Content-Based Image Retrieval

Content-based image retrieval (CBIR), methods and techniques for retrieving images from large image data sets using image visual content, has been a very active area of research since 1990s. Early work on this field can be tracked back to the late 1970s, when the subject of study was almost exclusively text-based image retrieval (TBIR) using textual annotation of images. In fact, manually adding descriptors to large image data sets was time-consuming, difficult, and error-prone. With the emergence of the Internet, the use of new techniques became necessary to access to the huge volume of digital images available on the web. This has attracted many researchers to work on the techniques of visual information extraction, indexing, user query, interaction and retrieval methods [5, 6, 12].

Visual information extraction, or simply feature extraction, can be considered as

the first step in all CBIR systems. These features can be classified as global or local content descriptors. Global ones represent the visual features of the whole image as opposed to local descriptors that describe the visual features of a segment, a region or a specific object. In general, color, texture, and shape are most commonly used visual content descriptors. Color information is usually represented in the form of a color histogram extracted from pixel intensities. Texture refers to the presence of some spatial patterns and also interaction of colors in an image. For example, similar edge orientation over all or part of image pixels can be classified as a textural feature. Shape can be described as a geometrical representation that is conveyed by colors, intensity patterns, or a texture. Shape can be considered as a mid- or even high-level feature in comparison to color and texture. Houses, cars, and human faces are examples of such features [5, 23].

CBIR systems support different user query techniques and methods namely keyword, free-text, image, graphics and composites. In the keyword query method, the user searches images by inputting one or more words. This is one of the most popular image search methods used by Google, Bing, and Yahoo!image search engines. The extended free-text technique allows the user to use complex phrase, sentence, question or a story for searching inside large image data sets. Query by example refers to the use of an external image (uploaded to the image repository or linked through its URL) to retrieve similar images from data set. For example TinEye [1] uses this technique for finding different variations of an image. In graphics-based methods, the system enables the user to draw a sketch either manually or by using computer and the system uses this graphics for retrieving information. Finally, the composite methods use a combination of two or more of the above-mentioned techniques to query the image database [5].

CBIR systems calculate visual similarities between a query image and the images in their database. This is done using various distance metrics and methods including Minkowski-form distance, quadratic-form distance, Mahalanobis distance, Kullback-leibler divergence (KL), and Jeffrey divergence (JD). The Minkowski distance and the quadratic-form distance are the most commonly used similarity met-

[1] TinEye, `http://www.tineye.com/`

rics for image retrieval. The Minkowski distance is a generalized form of Euclidean and Manhatan distance. It is used in cases where all dimensions of image features are independent of each other and have similar importance. The quadratic-form distance considers the pair-wise similarity and dependency of different features in the feature vector. This comes from the fact that certain pairs of features are perceptually more similar than others and encoding this information in similarity calculation yields a better result. The Mahalanobis metric is appropriate where image feature dimensions are independent of each other but have different significance. The Kullback-Leibler (KL) metric is a non-symmetric measure used to calculate the difference in the feature distribution of one image with respect to another image. The Jeffrey Divergence (JD) is the symmetric version of the KL distance. These are used for calculating texture similarity [12].

Having huge image databases implies a good indexing mechanism to improve performance. In order to have a good indexing schema, dimension reduction is applied to reduce the dimensions of visual feature vector. Principal Component Analysis (PCA), Independent Component Analysis (ICA), Karhunen-Loeve (KL) transform, and neural networks are the most common methods for dimension reduction. The index tree is built on the reduced feature vector using some indexing structures such as R-tree, R*-tree, quad-tree, and K-d-B tree [5].

There are many CBIR systems that provide image search and retrieval services for different purposes and applications. Here we review some of the existing systems having some special or novel aspects [23]:

- TinEye is a reverse image search engine that enables the user to search for the origin of an image, its modified or higher resolution versions, and also the usage of the image in the web by uploading a version of the image. It is the first image search engine that uses image identification technology rather than meta-data for search. As a search result, it does not retrieve the similar images, it actually brings the exact match especially those that are cropped, flipped, rotated, modified, or even resized. Similarly to the other search engines, it crawls the web to add new images to its current database.

- MIRROR [2] [29] is an interactive CBIR system that uses MPEG-7 [3] technologies. It extracts color and texture and converts this information into MPEG-7 data stream. The system supports hierarchical image browsing, query by example, random browsing, and user relevance feedback.

- CORTINA [4] [8] is a large-scale image retrieval system, having more than 10 million images. It uses a combination of color and texture in its similarity search and enables the user to refine image results with relevance feedback. It supports manual annotation, segmentation, near duplicate image detection, and face detection. It also supports keyword query, query by example, browsing images, and category-based search (semantics-based image clustering) functionalities.

- CIRES [5] [10] is a CBIR system for digital image libraries that, in addition to color and texture, uses image structure as a higher level semantics in calculating image similarity. The system allows the user to select an image from a set of image collection and submit the query, and after browsing the results to the user, it uses user relevance feedbacks to refine the results. Having structure as a feature enables the system to support different query ranges from structural objects, namely bridges, buildings, and towers, to natural scenes, such as sky, trees, and river.

In this research we use a combination of different image features namely color, texture, and shape for image search and retrieval. We calculate these features both for the whole image and image blocks. For the similarity calculation, we have modeled the problem as an assignment problem [3, 27] where the cost of assigning a block of query image to a block of an image in the data set is defined based on the similarity of the features extracted for these blocks. As we will see in subsequent chapters, our approach leads to satisfactory results.

[2] MIRROR, `http://www.ee.cityu.edu.hk/~mirror/`
[3] MPEG-7, `http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm`
[4] CORTINA, `http://vision.ece.ucsb.edu/multimedia/cortina.shtml`
[5] CIRES, `http://amazon.ece.utexas.edu/~qasim/research.htm`

2.2 Searching and Organizing Image Data sets

In the last decade, we have seen substantial advancement in the area of content-based image retrieval. New techniques, algorithms, search methods, and systems were developed based on the image visual features. However, there has not been enough attention to human-centric factors considering how users organize and browse the image collections. In other words, current systems focus more on supporting query techniques rather than browsing techniques [19]. To improve image retrieval systems, it is necessary to understand and consider human factors that affect the search strategies specially in multimodal image retrieval [20, 28]. These have been the inspiration of a class of research that try to improve the way that image data handled, presented, and browsed by the users.

As a recent work, Westman *et al.* [28] have studied search strategies in multimodal image retrieval system by analyzing the queries and search tactics. Participants in their case study were image journalism professionals and non-professionals. The result of their study revealed that users tend to combine textual and visual search methods in searching image collections. The user type and the type of the given task have also an impact on the usage of different query modes. For example, in searching known items, users combined text, color, and category in their queries but for conceptual search tasks they used several text and category type queries. Professional users were interested in using color queries and often switched the query mode while non-professionals were willing to use sketch queries and often edited the content of the query. Moreover, based on the transaction logs analysis, they extracted some interesting patterns such as "sequence of querying, viewing result images and saving results in a workspace" or "repeating the same query type".

Choi *et al.* [4] were interested in studying the analysis of users' queries for image retrieval in American history in order to determine what image attributes are important. They collected pre- and post-test questionnaires and interviews from 38 faculty members and graduate students of American history. The analysis result showed that over half of the search requests were general or nameable. In addition, most of the search terms were related to a person, thing, event, or condition.

Their study suggests that the topic of the image (what the image is about) and the objects in the image (what is depicted in the image) can be considered as optimal descriptors to access the image. They also recommend using shape, example, or color search as a promising approach rather than a keyword search in the context of image retrieval systems.

Rodden *et al.* [20] have also conducted a study to find possible answers to questions such as "How do people like to organize and browse their digital photographs, and how is it different from their non-digital collections?" and whether "advanced multimedia processing and techniques are useful in this context". For this study, they have used a tool called Shoebox; software with some basic browsing features such as folders, thumbnails, and timelines and some advanced features such as content-based image retrieval and speech recognition for voice annotation. During six months, thirteen users participated in the study including interviews, questionnaires, and analysis of usage statistics gathered from Shoebox. The result of the study showed that organizing digital photos is much easier than non-digital photos considering simple browsing features. Participants were also more interested in using the basic features of the system and did not use that much the advanced features of the system. Moreover, they preferred to browse their collections by events rather than querying them based on a specific property such as visual property of an image. In general, efficiency, reliability, and a good design are important features of such systems that are aimed at managing image collections. These kinds of systems can be equipped with two important features: automatically arranging photos in a chronological order and also displaying large number of thumbnails all together.

In another study, Rodden *et al.* [19] were interested in examining whether arranging thumbnails based on their mutual similarity is helpful for browsing images. In this regard, they conducted two experiments, employing designers as participants. In the first experiment, they examined the participants' preference between visual and caption similarity-based arrangement. The result of the experiment showed that similarity arrangement is more useful when users want to work on a specific subset while caption-based is helpful in categorizing the image data set based on the meaning. The second experiment was about comparing similarity-

based arrangement versus random arrangement. The analysis result revealed that similarity-based arrangement breaks the data set into simple genres or groups and it makes adjacent images seem as a unit. Random arrangement is useful when there is not any specific requirement for image browsing and it can highlight differences of adjacent images. Moreover, some users prefer to have access to both arrangements and use them interchangeably.

2.3 Faceted Search

Exploring large image data sets requires an appropriate user interface, so that, on one hand, the user has the freedom and flexibility to choose images based on a variety of features, and on the other, the system can efficiently retrieve large data sets meeting the user's criteria. In general, user-provided tags do not carry sufficient information about the image visual properties and the task of tagging is time-consuming and costly. In this regard, using faceted search, an attractive alternative to "text box" search and navigating the information through items' categorization can help the users become quickly familiar with the scope of the content [1]. Moreover, it seems that users find the faceted-search interfaces very helpful and they prefer them to traditional search interfaces [31].

In a recent work, Lin *et al.* [30] present an entity-based faceted browsing called ImageSieve that benefits from automatically extracting image descriptions referred as named entities (NE). Name, location, facility, occupation, and organizations are examples of NEs. When the user enters the query term, the system in addition to all related search results (image plus description) returns NEs retrieved from the photos' description. Then based on the frequency of NEs, the system lists and also classifies them in four categories of *who*, *where*, *when*, and *what*. In this step, the user can start faceted browsing by selecting one or more interesting NEs as well as the initial query term to refine the search results. In each step of the search process, the NEs will be automatically updated based on the latest search results.

Zwol *et al.* [33] introduce a system called MediaFaces currently available in Yahoo!'s image search engine that supports faceted exploration of large media

collections for semi-structured sources. The system extracts objects and facets from Yahoo!network resources and ranks them based on analysis of search query logs and the behaviour of Flickr tagging in order to retrieve the most relevant facets for a user query. Then the user selects the most appropriate facet to narrow down the image search result.

Villa *et al.* [25] have developed FacetBrowser that aims at doing complex search by supporting multiple search facets simultaneously. Instead of having pre-defined categories or meta-data that implies a lot of pre-processing, it gives the users the ability to define multiple facets during the search process. Each facet can be considered as a sub-topic in a larger search. By organizing such facets, the user can create a sequence of tasks (searches) which all together complete a complex search.

Also Muller *et al.* [18] have designed a faceted search prototype called VisualFlamenco based on the Flamenco framework [6] by introducing visual facets that provide the user with more guidance. Examples of such facets are a dominant color in the whole image or in a particular region of the image. These facets can bridge the semantic gaps exists in other systems by demonstrating the meaning of the facets. Yee *et al.* [31] have also developed a Flamenco-based faceted search interface that allows users to explore data along conceptual dimensions (image descriptors) using a hierarchical faceted meta-data structure. The interface is relatively slow but its category-based approach gives the user more insight about the content.

Overall, faceted-search interfaces provide a more convenient, natural environment to the users and the search results of such systems are far better than the traditional systems. However, due to some pre-processing the response time of faceted search systems may be increased compared to a traditional keyword-based approach. We have used the faceted search in our content-based image retrieval prototype in such a way that we consider a visual facet corresponding to each of the extracted image features and a textual facet for image tags. The visual facets are low-level features such as color and texture, and the higher level conceptual features such as shape and specific objects. The visual and textual facets can be used either

[6] Flamenco, `http://flamenco.berkeley.edu/`

individually or combined together. The system enables the user to upload an image for each group of visual facets and compares the features of query image(s) against the extracted features of existing data set and retrieves the most relevant results.

2.4 Cultural Analytics

Lev Manovich [13] director of the Software Studies Initiative at the University of California, San Diego (UCSD) first proposed the term "Cultural Analytics" in 2007, to refer to the process of "automatically retrieving large amounts of features or concepts from large data sets to understand cultural trends with the use of quantitative measures and interactive visualization techniques". His team has conducted several studies, analyzing art paintings, films, comics, web comics, cartoons, video games and recently one million manga pages [7] as their first large-scale study, to find typical cultural trends and patterns. For example as their initial experiments on cultural analytics, they show how some features of art paintings such as forms, color brightness and saturation change over time and how these are related to different painting styles such as realism, impressionism and abstraction. They have also found that movie length correlates with the movie's country of origin, by examining a set of feature films of USA, France and Russia over a specific period. They have conducted similar studies on videos obtained from computer games playing sessions [16].

As an example of cultural analytics on online videos, Zepel from UCSD [32] also has done some analysis on a small data set of online video ads. from the U.S. 2008 presidential election. At that time, Barak Obama and John McCain used online videos and YouTube Channels as a medium to broadcast their speeches, interviews and debates. Based on these online videos, she has tried to highlight the differences between TV ads and Web ads and also the differences betweens commercials of two candidates. The result of her analysis shows that commercials created for the web are less dynamic than those created for TV, since Web commercials include more content, graphics, and text. In another comparison between the two candidates' ads, McCain's TV ads seem visually more aggressive and radical.

Douglass *et al.* from UCSD [7] have done cultural analysis on the Japanense comic books called manga pages. the Manga data set was downloaded from oneManga.com that has been one of the most active "scanlation" web sites till July 2010. Scanlation (scan plus translation) is the term used for the unofficial translated version of manga pages by fans and communities. The reason for the appearance of scanlation was that the official translation of manga pages was slow and English readers were eager to have the translation ready as quick as possible. The original manga audience in Japan was boys, girls, young men and women while in the unofficial translation the number of series for girls' audience are fewer and the older group audience are very small. Also the oneManga web-site audience includes mostly males in the 18-24 age groups. Based on some computational techniques, some characteristic of scanalation are as follows. The oneManga web-site does not contain all books and all pages of a specific title. Since different people are scanning the manga pages and doing the translation, there are some distinct differences namely in cover art, table of contents, and also a deliberately added one extra page at the end of each chapter that includes people involved in the translation plus their roles as translators, cleaners, and typesetters. These credit pages are different from the story pages having bold visual designs in terms of color, text and logo. However, there are also very few credit pages that are similar to their original story pages. The challenge here was how to automatically differentiate the original story pages from credit pages in one million images. They used a combination of digital image analysis and file meta-data analysis to distinguish between these two types. Other than that, they have compared the official manga pages translation by their publisher with the unofficial version from the oneManga web site as another interesting analysis. To highlight these differences, they have applied some image processing algorithms and revealed some of the differences and variations in terms of the text of translation, the abbreviated or short text presenting sound effects, and some modifications or even replacements in the original graphics.

Another cultural analytics related work, Moretti [17] introduced the idea of "distant reading" for literary history as a new approach for analyzing data from books without reading them to answer some questions like "How has literature

been changed during years?" and "Are there any patterns that can be revealed from these changes?". In this regard, they have done some analysis on literary history to find patterns on different genres, their appearance, their life span, and also their behaviours across world. In one of their studies about novels of countries such as Britain, Japan, Italy, Spain, and Nigeria in the years between 1700 and 2000, they found similar patterns and behaviours in these countries showing a sharp increase in the appearance new novels in those years *e.g.* one novel per month or even per week.

As another type of cultural analytics, we can mention Tang *et al.* [22] that have studied email usage attributes such as number of messages, percentage of messages classified in folders and so on among different countries and geographic regions, using a research prototype called *bluemail*. They found that European users tend to organize their emails in folders more than Asian users. Finally, as related to on-line tools, one could consider Internet services such as Google Trends[7] that presents the frequency of a search term across different parts of the world; Nielsen BlogPulse[8] that mines more than 100 million blogs to find what people are interested in and also what people think about specific brands [15]; IBM History Flow[9] that uses histories from Wikipedia edit's pages to visualize the contribution of different authors; and Google Web Analytics[10] that gives detail statistical information about web sites visitors. These can also be viewed as examples of cultural analytics as they also shed light into how our activities and cultural preferences change over time.

In this thesis, we try to take advantage of all these research trends in understanding image similarities and extracting cultural inter-relationships, and general trends among the given image data. We have exploited variations and extensions of similar ideas in integration with our feature-based image exploration and faceted search prototype. We have used all these together to investigate existence of patterns or trends in some thematic cultural image data sets publicly available.

[7]Google Trends, `http://www.google.com/trends`

[8]Nielsen BlogPulse, `http://www.blogpulse.com/`

[9]IBM History Flow, `http://www.research.ibm.com/visual/projects/history_flow/`

[10]Google Web Analytics, `http://www.google.com/analytics/`

Chapter 3

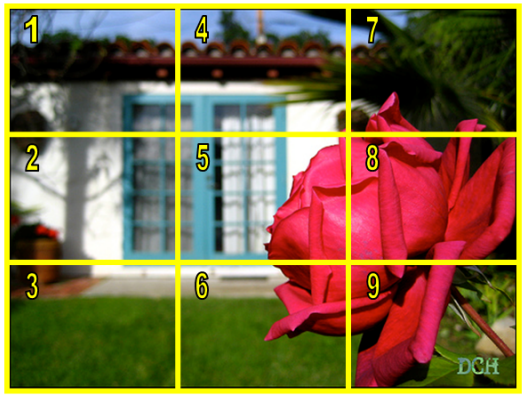
Feature-Based Approach to Image Cultural Analytics

There are several well established image-processing and computer-vision techniques that can be used to extract low-level features such as color, texture, shape, and spatial features [21]. The question then becomes to establish some relationship between low-level, easily extractable image features and abstract concepts in order to enable the use of these features in cultural analytics. Such an approach necessarily involves two distinct phases: (a) feature extraction and (b) systematic analysis of the correlations between features and culturally-relevant abstract concepts. It is this latter step, namely the study of correlating the values and variations of low-level features with high-level phenomena that can be observed as a cultural-analytics activity. Such studies potentially can be conducted over large thematic image data sets collected over time and across different geographical areas [16].

In the following subsections, we explain the low-level features that we currently extract and store in the database, intuitively motivate how these features may constitute evidence of cultural phenomena, and describe how these features are processed for some typical image-analysis tasks.

3.1 Color-Related Features

Color intensities and their variations are valuable source of information that often can be associated with some concepts of cultural significance. For example in fes-



Segment	Whole image	B1	B2	B3	B4	B5	B6	B7	B8	B9
Mean	89.05	84.46	136.14	64.60	109.41	148.29	80.11	37.12	83.06	61.50
StdDev	64.83	65.18	71.23	51.95	75.91	54.22	51.03	36.09	37.54	36.32
Mode	0	0	252	44	6	255	56	8	109	59

Figure 3.1: Dividing an image into 3×3 blocks and calculating statistics for each block

tivals and wedding ceremonies, people tend to wear bright colors. There are also many lights in these places, which will likely be reflected as bright colors in related images. More importantly, in different cultures people may use different colors for these ceremonies. In general, images classified under different topics or concepts somehow show similar intensity/color properties relevant to that class. For example, images that are taken from natural scenes are usually different in terms of color, themes and composition compared to images taken from man-made structures. As another example, in the movie industry, based on different genre, cover photos have different color properties *e.g.* horror and thrilled movies often have darker posters while animations and comedy movies have brighter posters. These differences can be highlighted and used in analysis, understanding and classification of images.

Image-color histograms are well known tools that can be used to describe the color-related information. Our tool integrates ImageJ for extracting image histograms (gray scale and three channels of RGB color space) and some associated statistical information including mean, median, standard deviation, and minimum and maximum color values. Since finding similar images is based on color statistics rather than the true colors of an image, we have chosen to divide an image into blocks that gives us more samples and more information from color distribution.

Therefore, we divide an image into $n \times m$ blocks and also calculate histograms and associated statistics for each block. We extract image histogram both for the whole image and all blocks. Figure 3.1 presents a sample image that is divided into 9 blocks and the color intensity statistics of the whole image and all related blocks. These features are used to calculate color similarity of images based on the general feature matching process explained in Section 3.4.

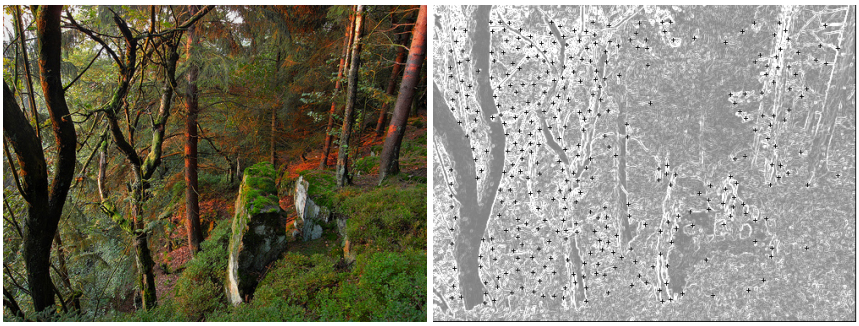
3.2 Texture and Structure-Related Features

Structure-related features are also fairly easy to extract and can provide strong cues with respect to the theme of the image, namely whether it represents a natural, architectural or social scene. More likely than not, architectural scenes tend to have more lines than natural scenes, as a result of meeting surfaces, and corners. Similarly, indoor social scenes are more likely to contain special structural features. Moreover, in close-up images corners are clearer comparing to long shot images captured at a long distance.

Our tool extracts a set of features related to the texture and the structure of an image, including corners, and horizontal, vertical and slanted lines. It uses the Harris corner detector [9], which is implemented as a plug-in for ImageJ. The Harris corner detector [1] is a well-known interest point detector that is invariance to scale, rotation, illumination variation, and image noise. The algorithm approximates variation of intensities in all directions based on the image differentials calculated along with the vertical and horizontal directions. Having large intensity changes in one or more directions is considered as a good interest point. Examples of such interest points are corners, isolated points of maximum or minimum local intensity, and line endings.

Running this algorithm on different image data sets, we found that it often detects more interest points in long-shot, highly textured images than in close-ups. Figure 3.2 shows examples of a long-shot scene, a highly-textured close-up, and a close-up image. Image (a) is the picture of a forest where a large number of inter-

[1] Harris corner detector, `http://www.cse.yorku.ca/~kosta/CompVis_Notes/harris_detector.pdf`



(a) Long shot

(b) Highly textured

(c) Close-up

Figure 3.2: Sample of highly-textured, close-up, and long-shot and their corresponding corners detected by Harris corner detector

est points are detected by Harris Corner in it. Image (b) is a close-up but since it has a lot of texture, as can be seen, here also a large number of interest points are detected. Image (c) is a close-up of a flower with a house as the background. For this image few number of interest points are detected by the algorithm. Considering these differences, it is perceived that this feature can be used in classifying and categorizing images.

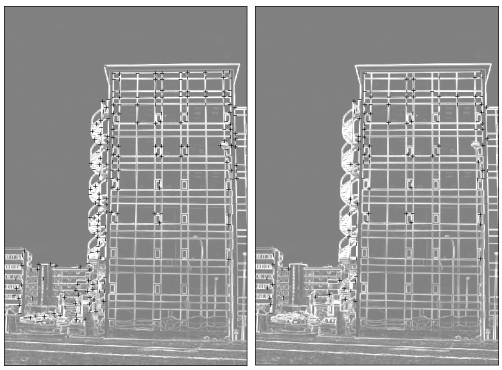
Defining corners as the intersection of two edges, corner-detection algorithms, including the Harris corner detector, often detect interest points rather than just

0	0	0	0	0	0	0
0	0	0	0	0	0	0
-1	-2	-2	-2	-2	0	0
3	3	3	3	-2	0	0
-1	-1	-1	3	-2	0	0
0	0	-1	3	-2	0	0
0	0	-1	3	-1	0	0

0	0	-1	3	-1	0	0
0	0	-1	3	-2	0	0
-1	-1	-1	3	-2	0	0
3	3	3	3	-2	0	0
-1	-2	-2	-2	-2	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

Figure 3.3: Samples of filters that are used for detecting lower-right and upper-right L-junctions from initially detected corners

(a) Natural scene, left: initially detected corners right: accepted corners, ratio = 5%



(b) Architectural scene, left: initially detected corners right: accepted corners, ratio = 50%

Figure 3.4: Comparing initially detected corners with accepted corners for two different type of images

corners conforming to this definition [2]. As result, a post-processing step is required to filter out undesired points, from the originally detected corners. Figure 3.3 presents samples of the filters we have used for this purpose. The filter that are shown in this figure are respectively used for extracting upper-right and lower-right L-shape corners. Other filters are generated in the same way.

Figure 3.4 compares the initially detected interest points (those that are detected by Harris corner detector) and the accepted ones (those that have passed the filters) in two different images: a natural and an architectural scene. For each of these scenes, the left image shows the initially detected interest points and the right image shows those that are accepted as right corners. As expected, the number of accepted points for both images is smaller than the number of initially detected

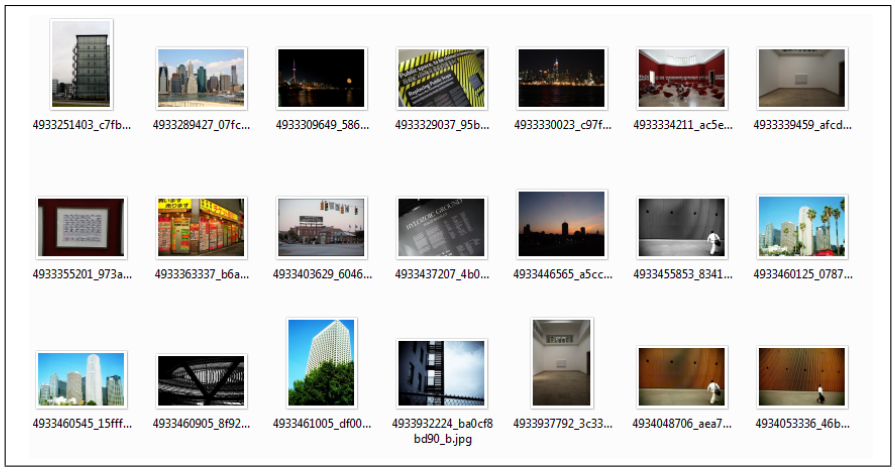


Figure 3.5: Samples of images with the percentage of accepted corners more than 30% (avg = 39%)

-1	-1	-1
-1	8	-1
-1	-1	-1

(a)

-1	-1	-1
2	2	2
-1	-1	-1

(b)

-1	2	-1
-1	2	-1
-1	2	-1

(c)

-1	-1	2
-1	2	-1
2	-1	-1

(d)

2	-1	-1
-1	2	-1
-1	-1	2

(e)

Figure 3.6: Samples of filters are used for detecting (a) points (b) horizontal lines (c) vertical lines (d) 45° slanted lines (e) 135° slanted lines

points. However, the ratio of accepted to the initially detected ones in the first image (Figure 3.4a, a natural scene) is only 5 percent whereas this ratio for the second image (Figure 3.4b, an architectural scene) is about 50 percent. Figure 3.5 shows samples of images where the percentage of accepted corners to the total initially detected corners is more than 30 percent. Again we can see that most of these images belong to or related to the structural or architectural scenes. Similarly, we observed that for those images that this ratio is small, many of them are natural scenes or objects. Noting to these differences, it clearly can be seen that this ratio can assist in classifying images. For this reason, we store the number of initially detected interest points as well as the number of accepted corners in the database.

Finally, we use a set of filters, similar to the filters used in [11], on the edge map of the images to capture vertical, horizontal, and slanted lines. Figure 3.6 shows samples of the filters we have used for this purpose. Here, same as with color-related features, we split the original image into blocks, we apply the same process

to the individual blocks and the whole image and store the calculated features in the database.

3.3 Object-Level Features

Object-level features, such as simple geometrical shapes or human and animal faces, can be considered as higher-level information, as compared to the low-level features described in the previous sections. We are interested in extracting these features considering that they can provide strong cues regarding the theme of the image, whether it is social (with human faces) or natural (with animals) or even architectural (containing special architectural objects) scene, and more importantly they convey some of these abstract/high-level concepts by themselves. Therefore, they are more directly relevant in image-analysis and image-understanding tasks. However, it is often more difficult (computationally and methodologically) to extract these features.

Here, as a good example of object-level features, we explain our experience with detecting human faces. To detect human faces, we use the Viola-Jones object-detection algorithm [26] that is implemented in the computer-vision open-source library OpenCV. It can determine the location and size of human faces in arbitrary images and possibly detect some associated properties. We have integrated the original function in our system but we extended it with several pre- and post-processing to improve the results of the original implementation. These improvements are rather important in the overall performance considering that this function potentially can be trained to detect a variety of other object classes as well. The original implementation of the algorithm was unable to detect small faces of people in far distance (Figure 3.7a), tilted faces (Figure 3.7b), and also cropped faces (Figure 3.7c). We improved the effectiveness of the original algorithm by applying some simple transformations on the input images as follows:

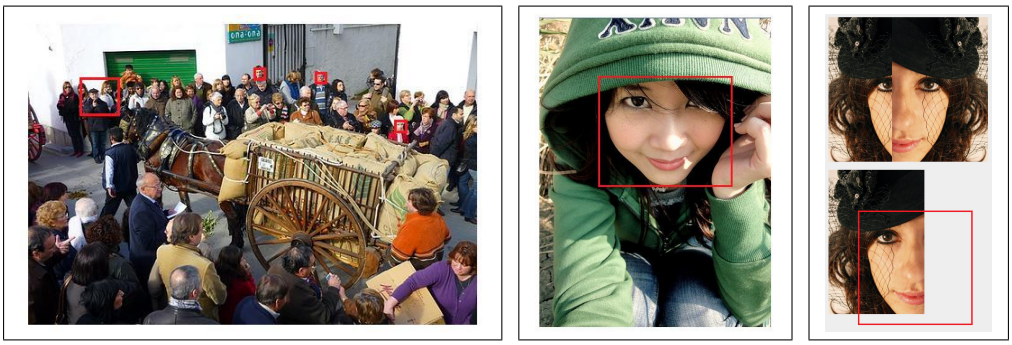
- Proportionally resizing width and height of images (scaling image) with different coefficients for the purpose of detecting smaller faces. For example, four faces are detected in Figure 3.7a by scaling the image by a factor of 4

while no faces are detected on the original image.

- Horizontally or vertically resizing images with different coefficients for the purpose of detecting slightly rotated faces. For example, the slightly tilted face in Figure 3.7b is detected only when the image is vertically resized.
- Flipping the picture horizontally and placing it to the left or the right of the original image with different offsets for detecting faces in cropped images. For example, Figure 3.7c shows a half cropped face that will be detected using this technique.

Applying these transformations to the input images significantly improves the efficiency of the algorithm. For example, on one of our data sets composed of 50 images, of which 21 have people, the original implementation recognizes only 12 images as containing a face: 10 of them are correct and 2 are false positives. Applying the aforementioned transformations, 25 images are detected as having faces, with 17 correct detections and 8 false positives. To reduce the effect of false positives, we have assigned a confidence level to each detected face depending on the type of the transformation that is applied on the original image. For example, we assign less confidence to those faces that are detected with larger scaling factors.

As a further refinement, we also pass extracted faces through a skin model filter in YC_bC_r color space [24]. In this color space, Y is the brightness (luma/luminance), C_b is blue minus luma ($B-Y$), and C_r is red minus luma ($R-Y$). The confidence level assigned to the detected faces is modified proportionally to the output of the skin-model filter, but we do not eliminate any detected faces as they constitute valuable information extracted from the image. Figure 3.8 shows an image where the face detection algorithm incorrectly detects a face. The initial confidence assigned to this face is 1 since this face has been detected by applying Viola-Jones algorithm in the original image (without applying any transformation). Considering the image colors in the detected face area, this is a good example where the skin filter can help in removing (reducing the effect of) such obvious false positive detection. In this example, after applying the skin filter, the confidence of the detected face drops to 0.0015 which is almost equal to zero.



(a) Small faces (b) Tilted face (c) Cropped face

Figure 3.7: Improving face detection efficiency by scaling, horizontally/vertically resizing, and flipping images

Figure 3.8: Sample of a wrongly detected face, initial confidence = 1 (detected by original algorithm), face confidence after applying skin filter = 0.0015

3.4 Calculating Image Similarity

Having several heterogeneous features enables multidimensional comparison of images in order to evaluate their similarity. In addition, recalling that we divide images in a set of blocks and calculate different features not only for the original image as a whole, but also for each image block as well, we developed a general method "summarizing" image similarity across all these extracted features. We have modeled this problem as an assignment problem that finds the "best match" between the set of the first image blocks and a permutation of the second image blocks. More specifically, assuming F_1 and F_2 are two sets defined as follows:

- F_1 sets of feature vectors which are calculated for blocks of image I_1.

- F_2 sets of feature vectors which are calculated for blocks of image I_2.

Since number of blocks in both images are equal, then F_1 and F_2 are of the same size, $|F_1| = |F_2|$. The feature vector may contain one or more features depending on the number of facets involved in the similarity calculation. For example, it can be just a scalar representing average intensity of a block or a vector of average block intensity, number of detected corners, number of slanted line segments, and so on. Having these two sets, we try to find a bijection $f : F_1 \rightarrow F_2$ such that the cost function

$$\sum_{a \in F_1} C(a, f(a))$$

be minimized where $C : F_1 \times F_2 \rightarrow R$ is the cost of assigning a feature vector in F_1 to a feature vector in F_2.

A cost function must be defined for different features and their combination as a feature vector. For example, the color difference between two image blocks B_1 of image I_1 and B_2 of image I_2 is defined as the sum of the absolute difference between the mean of all three color channels in these blocks. In other words, the cost of assigning block $B_1 \in I_1$ to block $B_2 \in I_2$ in terms of color is defined as:

$$C(B_1, B_2) = \sum_{i \in \{R,G,B\}} \left| M_i^{B_1} - M_i^{B_2} \right|$$

where M_i^B is the mean of color channel i over block B. Similarly, appropriate cost functions are defined for corner- and edge-related features, based on the normalized number of detected features in each block. More specifically, the distance between two blocks in terms of their corners and edges is calculated as the absolute difference between the numbers of the corners and lines detected in each one of them. The weighted sum of these individual cost functions can be considered as cost function for feature vectors.

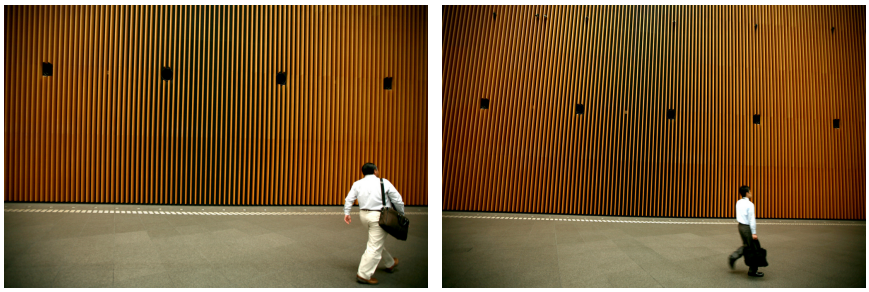
We follow the same approach for calculating image similarities for object-level features (such as human faces). For such features, the feature sets F_1 and F_2 are the set of objects detected in image I_1 and I_2, respectively, and the assignment cost is defined as the difference in the confidence of the detected objects. Here, we equalize size of F_1 and F_2 by adding dummy objects (zero confidence objects) to the smaller set if necessary.

	Confidence	K 1.0	L 1.0	M 0.89
B	1.0	C(B,K) = 0	C(B,L) = 0	C(B,M) = 0.11
D	1.0	C(D,K) = 0	C(D,L) = 0	C(D,M) = 0.11

Figure 3.9: Modeling image similarity as an assignment problem based on detected objects (here faces) and their confidence

Figure 3.9 illustrates modeling image similarity as an assignment problem based on detected faces. The image on the left has 4 detected faces all having the same confidence level equal to 1; the image on the right has 3 faces with two of them having confidence equal to 1 and the third one equal to 0.89. The cost of assigning a detected face in one image (for example face B) to a face in the second image (for example face K), $C(B, K)$, is calculated as the absolute difference in their corresponding confidences. As already mentioned here we add a dummy face with confidence zero to the right image to equalize the size of two feature sets. Following this procedure for this simple example, the similarity of two images in terms of the detected faces is calculated as 1.11.

Modeling the problem of image-similarity calculation as an assignment problem has several advantages over using a strict distance metric. It makes the similarity calculation, to some extent, scale- and rotation-invariant and helps in finding images containing similar structure, texture, and shapes regardless of the capturing viewpoint. For example, Figure 3.10 shows two images that are very similar. For these two images, fixed block matching similarity (FBMS) method calculates simi-



(a) (b)

Figure 3.10: Sample of very similar images, FBMS value = 9.42, ABMS value = 7.34

(a) FBMS value = 69.39, ABMS value = 16.32 (b) FBMS value = 61.22, ABMS value = 20.75

Figure 3.11: Sample of partly similar images captured from a scene from different points of view

larity value as 9.42 and calculating similarity through modeling it as an assignment problem (ABMS) gives similarity 7.34 which is close to FBMS value. This shows that the performance of both methods is similar whenever they applied on closely similar images (also see comparison of these two methods on movie data set in Section 5.3.2.3). On the other hand, when these algorithms are applied on images that are partly similar or taken from different view points, then ABMS shows much better performance (better detects the similarity of such images) than FBMS. For example, for the images shown in Figure 3.11 ABMS calculates the similarity values as 16.32 and 20.75 for the first and second sets, respectively, whereas FBMS does not show much similarity between these images. From these two examples it can be inferred that, in general, ABMS will be able to detect (partially) rotated, flipped, cropped or distorted images better than FBMS method.

Since the assignment problem establishes a one-to-one correspondence between blocks, to some extent, the relative location of the blocks are also taken into account in the similarity calculation process. More precisely, considering that the adjacent blocks in an image are expected to be more similar, and also similar images follow partly/generally similar patterns, it is more probable that the min-cost assignment algorithm matches locationally closer blocks to each other in similar images. Finally, this model properly fits some features such as faces or other detected objects where depending on the intended query, location of the detected object might be of less significance than its existence.

Chapter 4

Application Architecture

We have developed an application prototype that can be used for image exploration, classification, and retrieval from large image data sets. The application can also be used as a tool for performing different (statistical) analysis and visualization tasks on the retrieved images. Figure 4.1 illustrates the three-layer architecture of our web-based application. These include presentation layer, application logic layer, and data-management layer. Each of these layers and their functionalities/components are explained in subsequent sections. We also briefly explain the technologies we have used in different layers.

4.1 Presentation Layer

The presentation layer is a web-based graphical user interface (GUI). This layer is responsible for uploading images for image-based queries, sending the query

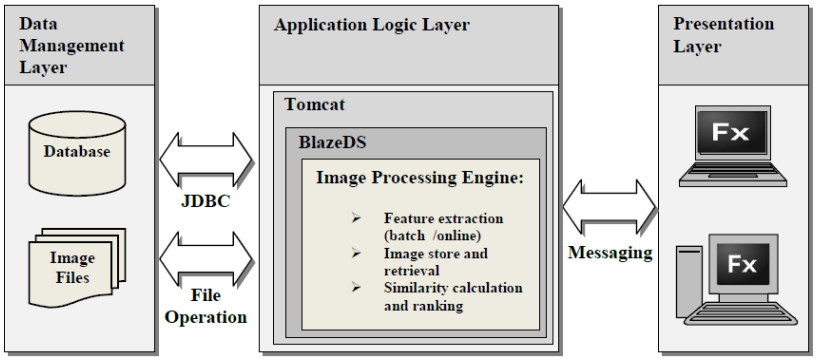


Figure 4.1: Prototype architecture

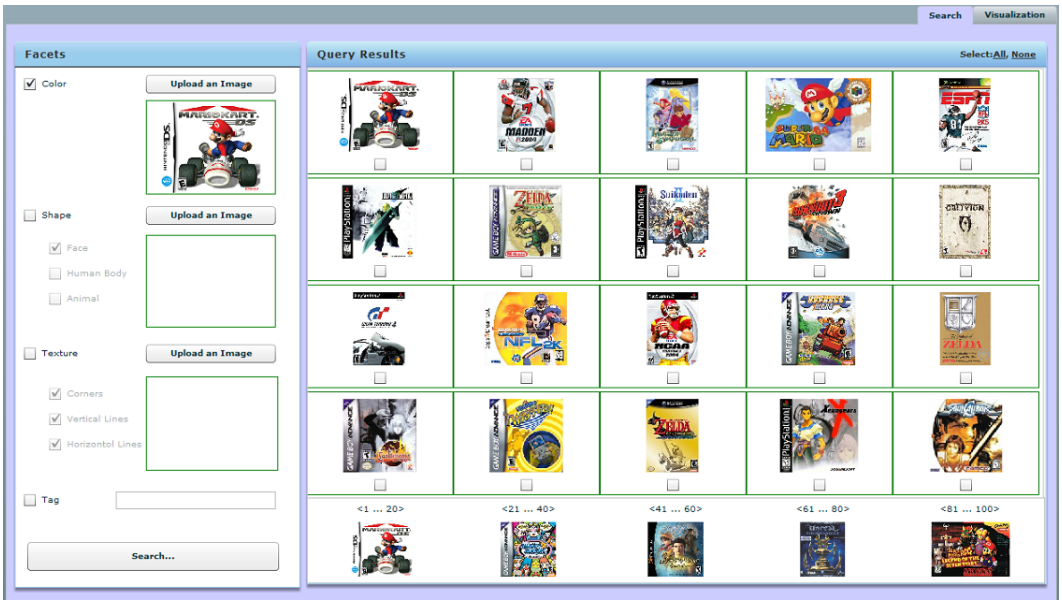


Figure 4.2: Web-based graphical user interface

(images or tags) to the middle layer for query processing, and enabling the user to browse the query results. Figure 4.2 presents a snapshot of application user interface implemented using Flex [1]. The user interface supports visual faceted search by enabling the user to upload one or more images as examples of the desired visual facets. Here, the system supports three visual facets: color, shape, and texture which can be combined with a semantic facet *i.e.,* image tags. For each of the visual facets, the system allows the user to upload an image containing objects or features of interest. For example, the user may upload an image having three human faces for the "face" facet to find similar images in terms of the number of faces existing in there (see Section 4.4 for more details and sample usages of the system).

4.2 Application-Logic Layer

The application-logic layer, implemented in Java, provides the core functionality of the system (image feature extraction, similarity calculation and so on). In this layer, Apache Tomcat is used as a web server and BlazeDS[2] as a Java Remoting and Web Messaging technology that facilitates the connection between the server

[1] Adobe Flex, `http://www.adobe.com/products/flex/?sdid=FFSBR`
[2] BlazeDS, `http://opensource.adobe.com/wiki/display/blazeds/BlazeDS`

and the presentation layer. Remote Object Service is one of the key components of BlazeDS. It provides the Flex application with remote procedure calls to the Java server via AMF3 protocol which is much faster than SOAP since it is a binary protocol. We use ImageJ, a Java-based image processing program, and also OpenCV, an open source computer vision library for the image feature extraction in this layer.

The image features are extracted either in a batch process or during an image-based query. The batch process is useful when the database is initially populated for a given data set. The batch process is a simple loop over all images of the data set, extracting image features and storing them in database tables.

The features that are typically extracted for images during the batch process or query are those features that are explained in the previous chapter. These features correspond to the visual facets we are provided in the user interface. As a result, the performance of the system will be limited by the type and the extent of the extracted features and how they contribute to the overall similarity calculations. In other words, at the moment we provide visual faceted search at the highest level and most of the decisions regarding to the use of different features are decided by the program logic. However, the interface could be extended to allow the user to decide on the type of subfeatures on each category. For example, the user could decide if the standard deviation of the image color histogram also needs to be involved in similarity calculation or if all features are treated equally or he/she could put more focus or weight on some specific features. The user could also be involved in a hierarchical decision process based on different facets and their corresponding features. It should be clarified that for the case of query by image, image features are calculated only for the given image(s). Since all the features of image of the data set are already calculated, the query time mainly depends on the performance of similarity calculation and the result are often returned within a reasonable time.

4.3 Persistence and Data-Management Layer

The data-management layer is responsible for maintaining the image files and their associated features, tags, and properties. We use a MySQL database to store image

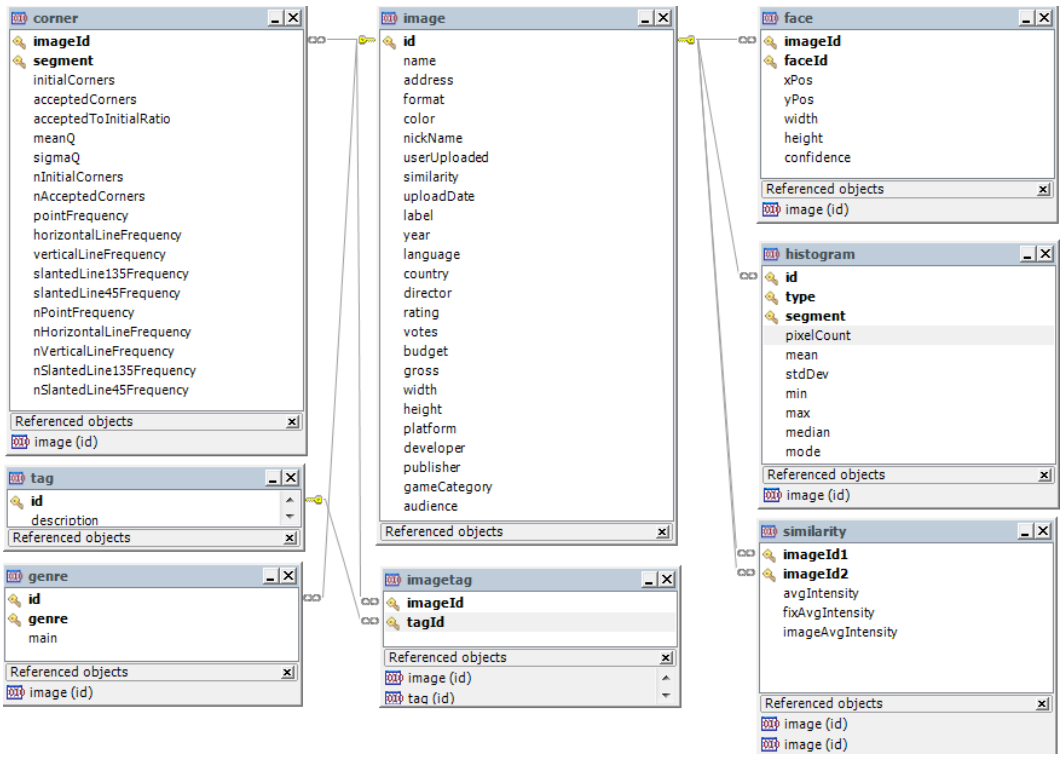


Figure 4.3: Database schema

tags and other general properties. All image features described in Chapter 3 are precomputed and stored in the database as well. The images themselves are separately stored in a specific directory on the disk and only their location is stored in the database. Figure 4.3 presents the database schema of the application. The database has several tables including *image*, *tag*, *imagetag*, *histogram*, *face*, *corner* and *similarity*. Here *image* is the main table that keeps track of image properties such as image name, URL, format, and if it is a color image. There are also some domain-specific properties that currently we store them in this table but in general they could be stored in a property list table. The *imagetag* table is used for storing semantic tags assigned to each image. These are usually the image tags that are assigned by users (owners or viewers) to the images of the data set. The image visual feature data are stored in *histogram*, *face* and *corner* tables. For each image, the color statistics, namely mean, standard deviation, minimum, maximum, and mode, are stored in *histogram* table for the whole image and all its blocks. The *face* table is used to store the face location, size, and confidence of the detected faces on im-

Figure 4.4: Browsing capabilities of the user interface

age. In general, this table could be replaced with a shape table where information of different shapes and objects could be stored in there. The *corner* table is used to store texture-related features. These currently include frequency and density of detected corners, and horizontal, vertical, and slanted line segments for the entire image and its blocks. Again, these three tables are corresponding to the three main visual facet categories implemented in this application.

4.4 Typical Usage of the System

When the user uploads one or more images, they are stored on the server and all their features are extracted from the uploaded image(s) and stored in the database. These features are then compared against the features of the existing images in the database. For each class of features similarity is calculated using the minimum cost assignment algorithm described in Section 3.4 and top N images which have the highest similarity will be returned to the presentation layer. This search and exploration functionalities repeatedly can be used to select an appropriate subset of images from the given image data set for further analysis.

Figure 4.4 presents a snapshot of the system user interface. The interface con-

tains three main panels: (A) Facets panel, (B) Query Results panel, and (C) Navigation panel. The facets panel supports four facets, three visual facets (A1, A2, and A3), and one textual facet (A4). As already mentioned for each of the visual facets, the user can upload an image containing the desired feature corresponding to that facet. This is done through the "Upload an Image" button (A5) next to each of the visual facets. For the textual facet (A4), the user is simply allowed to enter one or more words. When the user clicks on "Search" button (A6), depending on the selected facet, similar/related images are retrieved and shown as thumbnails (B1) in the query results panel (panel B) and in a descending similarity order. Query results panel offers different functionalities including auto-zoom (B2), selecting, and deselecting images (B3). The auto-zoom provides the magnified version of the image when the user click on a thumbnail in the result set. The magnified version is accompanied by image meta-data if such information is available in the database. The select/deselect capability enables the user to prune images in the query result panel. Finally, the navigation panel (panel C) at the bottom of query result area gives the user the possibility of navigating among the returning results. In fact, this navigation bar shows the first image of every k consecutive images packed together as a group in the query results (here we consider $k = 20$). The user can move forward or backward by simply clicking on these group representatives.

Figures 4.5 to 4.8 present sample usages of the system for different visual and textual facets. Figure 4.5 shows a screenshot of the system when the user has used the color facet and uploaded an image to retrieve images of similar color themes. This is the search result on a small set (100 samples) of top video game cover photos (see Section 5.2 for further details). Note that the returned images have more or less the same color combinations and appearance.

Figure 4.6 shows the result of using the shape (face) facet of the system in image exploration. Here, the data set is composed of 450 images downloaded from Flickr. In this example, the user is looking for images having only one face by uploading a query image that has only face. The face detection algorithm detects one face in the query image with confidence equal to 1. Therefore, the query returns all the images with one face detected in them as the most relevant (highest similarity/lowest

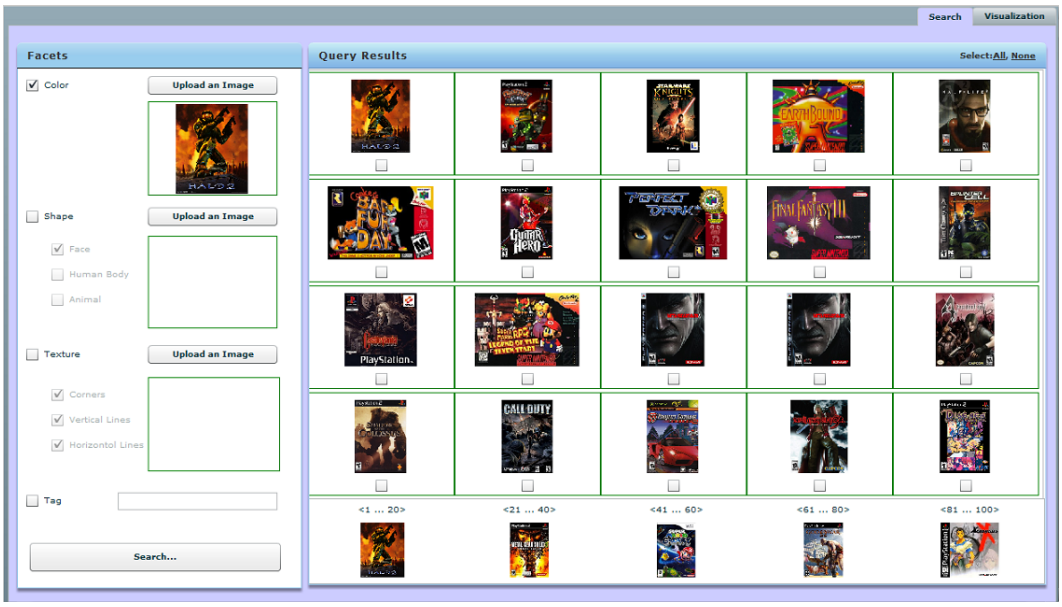


Figure 4.5: Search image data set based on color facet

distance) results.

Figure 4.7 shows a sample query result using texture facet. The data set is the one that is used in previous example (search by shape facet) but the query image does not exist in the data set. Here, the query image is a high-rise building with clear structural features such as vertical and horizontal edges and lines. The query result also has the same structure mainly belonging to the urban areas with some high-rise buildings.

Figure 4.8 shows the results for a typical meta-data search. The word “park” is used for this search. For the tag-based search, the system returns only those images that have the query word(s) in their meta-data.

In general, the faceted-search capability allows the user to build the data set itself based on a single feature, a single concept (tag) or a combination of visual features and concepts. For example, the user may choose to extract the collection of images that have red color and tagged as “wedding”. Then, the percentage of the images that are tagged as “India” can be calculated as well. In this manner, the user can investigate the hypothesis that red is the dominant color in wedding ceremonies in India, and possibly to see whether there is any change in the value of this hypothesis over time. Extending different capabilities of our system, similar

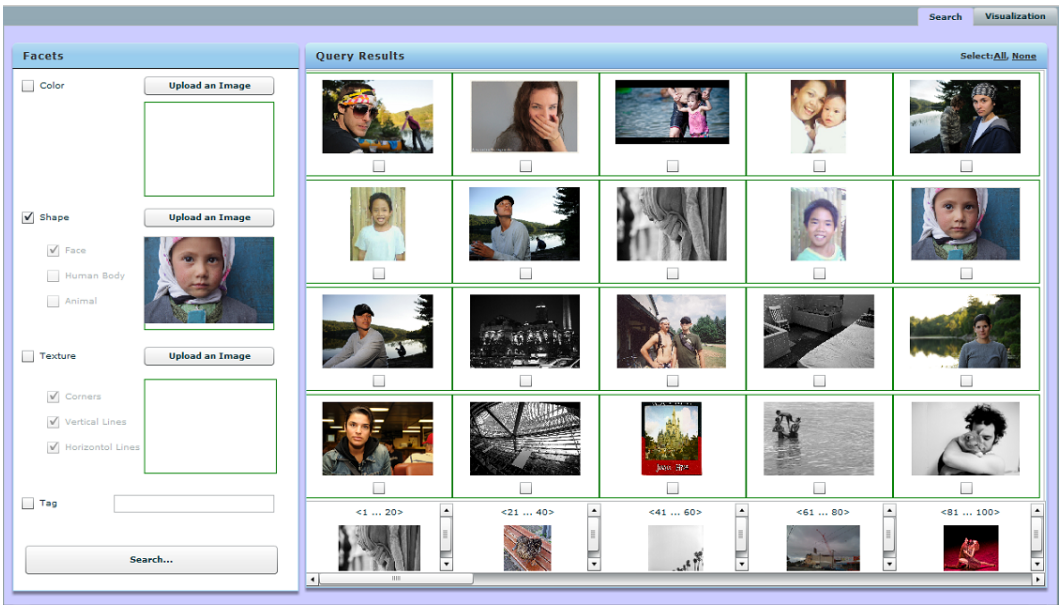


Figure 4.6: Search image data set based on shape facet

experiments can be designed and performed to extract more abstract concepts from data without involving the user with detailed technical aspects of the problem.

Ideally, the application could be equipped with a statistical analysis and visualization component (as provisioned in the user interface) to do some typical analysis on extracted and pruned images. For example, the user could find the relevance or trends of changes for a given property such as color intensity, number of faces (or other interesting objects) and so on. At the moment, we are using separate statistical analysis and visualization tools such as R [3], and MS Office tools for doing these analysis and visualization tasks.

[3]The R project for statistical computing, `http://www.r-project.org/`

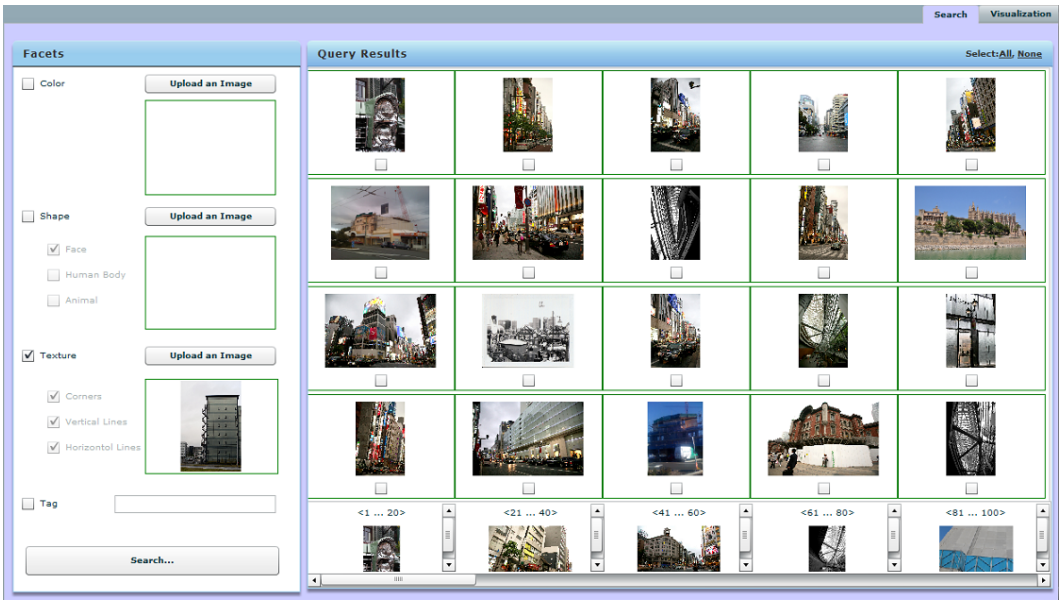


Figure 4.7: Search image data set based on texture facet

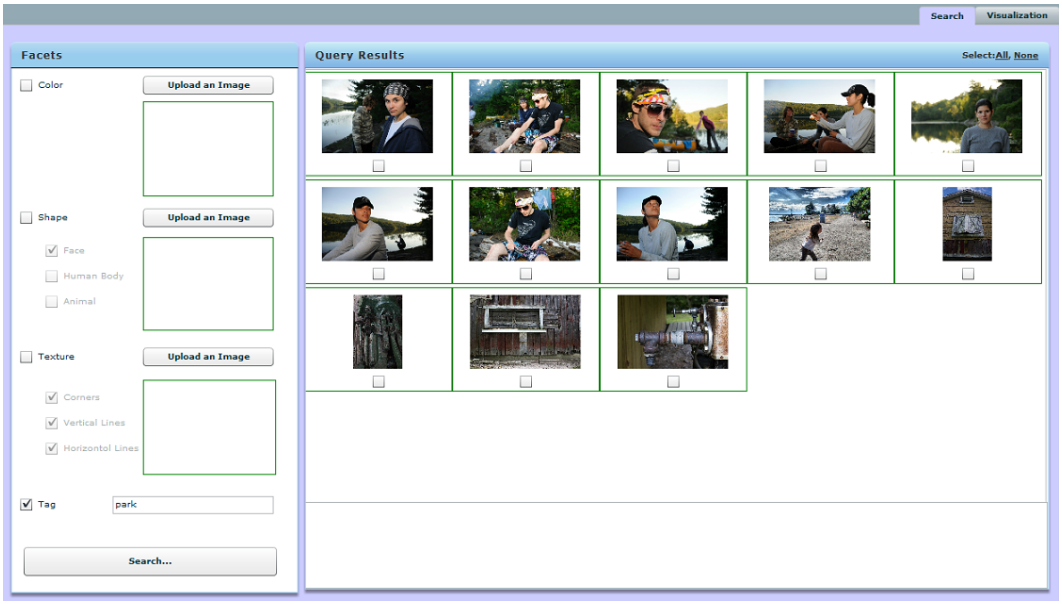


Figure 4.8: Search image data set using textual facet

Chapter 5

Image Cultural Analytics Case Studies

Previous work on image cultural analytics has usually focused on particular domains and limited feature-extraction tools and analysis techniques. Here, we are interested in a general purpose, more automated, and flexible analysis tool that can be adopted by a user to find cultural trends or examine cultural hypotheses in arbitrarily selected, possibly large image data sets. However, this approach, in addition to the technical challenges, involves some practical difficulties and limitations. For example, providing appropriate data for the system will still remain as a big challenge. Even though there are several publicly available data repositories, there are several limitations in properly accessing those data. These include privacy issues, bandwidth limitations, and lack of consistent methods for accessing different data sources.

Our feature-based approach to the cultural-analytics activity provides the user an environment for exploring the image data and their properties for the purpose of finding possible trends and relationships between those data and interesting cultural concepts. In order to evaluate different functionalities and component of our system, we have collected several data sets from publicly available image repositories and web sites and have performed several (cultural) studies on these data. We were interested in image repositories such as Flickr since they include much of user-generated content that, to some extent, reflect some cultural aspects of our social life. We were also interested in data sources such as IMDB and video game

that can be considered as a by-product of the professional cultural-related activities. Millions of people, especially non-professionals, create and share the cultural contents on Flickr. In contrast, IMDB (as well as video game data set) is a more thematic image repository that contains a lot of professional-generated content related to professional (art) activities. For each of these we have written extractor tools to download the images and related meta-data using their publicly available APIs and open source programs. However, as explained below, we typically had several limitations and constraints in accessing high volume of data on these resources and in general it was really challenging to create an appropriate data set for the purpose of our analyses. In the following subsections, we explain some of these challenges and how they have been addressed. We also present the results of the studies we have done on these collected data using different capabilities of our system.

5.1 The Flickr Study

Flickr is a huge image repository, used by people all over the world and as such it constitutes a very fertile environment for cultural analytics. We initially had two small data sets manually prepared from Flickr images, each of them containing 50 images[1]. However, the images in these data sets covered a wide variety of different subjects and concepts, including images of people in groups or individuals at different occasions, close-ups, long shots, cropped and rotated faces, and so on. Most of the pictures used in Chapter 3 are from these two data sets. As a result, they helped us a lot in evaluating different feature extraction techniques and improving the performance of our algorithms and methods.

5.1.1 Data Set

We used Flickr to create some larger data sets for using in some analysis on user-assigned tags validity and relevance (see Section 5.1.2). To that end, we have written a Java program using the flickrj API [2] to retrieve images and all provided meta-data for each image. In this program, we search based on two criteria namely a

[1] These two small data sets were collected by Dr. Reucker.

[2] flickrj API, `http://sourceforge.net/projects/flickrj/`

given tag, *e.g.* "boy", and within a specific time period. Nevertheless, we encountered two issues in extracting images: First, the Flickr API restricts the number of downloads to at most 500 images per call. Second and the most important one, it seems that the API returns a consecutive list of images that are uploaded within a short time period by a specific user. As a result, returned images are often very similar/correlated (for example, pictures of the same person with different poses or pictures of the same place from different angles), making it very difficult to automatically obtain an appropriate sample of uploaded images. To address this problem, we repeatedly searched Flickr for a very limited number of images for a series of consecutive time periods. For example, for the data sets used in analysis of Section 5.1.2 for each of the given tags we have extracted a few images per week for all weeks within the given time period and have put them together to create the data set. In this way, we have reduced the chance of having very similar highly correlated images inside the data set.

5.1.2 Analysis

Here, we discuss some comparative analysis we performed between Flickr images' tags and underlying image features to examine how Flickr's photo-sharing service is used and how valid are tags assigned to the images. In general, we were interested to answer some questions similar to the following ones:

1. How do people tag images? Do they use specific terms directly related to the subject of image or prefer to use implicit descriptions?
2. How valid are the tags assigned to the images? Do tags properly describe the image concept or they are completely irrelevant?
3. Is there any difference between people understanding of different concepts over time and across geographical locations?

In this study, and in order to examine the images' tags validity and trends, we selected a collection of images tagged with the words "people", "man", "woman", "boy", and "girl" and compared them specifically in terms of the number of detected faces. These images were downloaded from Flickr between years 2004 (Flickr

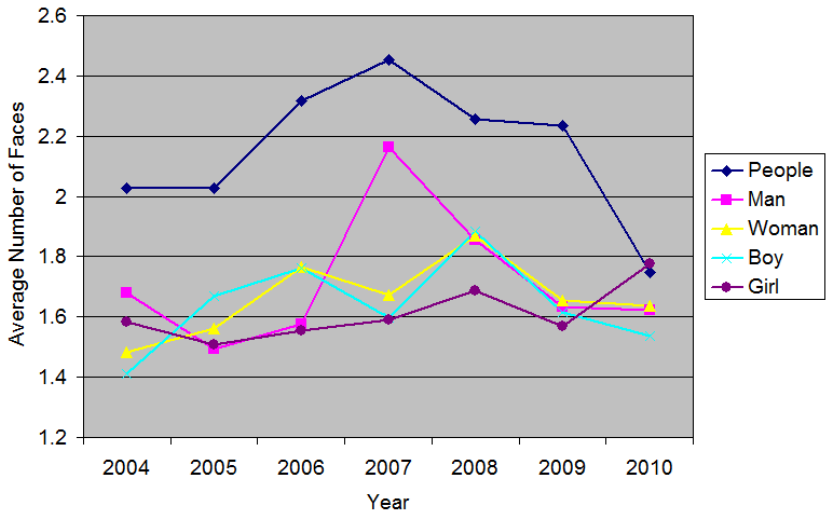


Figure 5.1: Average number of faces detected per year for human-related tags

launching year) and 2010 (about 250 images per year in each group). Running our batch process over these images, we extracted the features and stored them in a database. From the stored values we found that the face detection algorithm detects faces in about 60 percent of all these classes of images with no significant change over different years (this percentage includes false positive and true negatives but we may assume that the algorithm has similar behavior in all these groups). We also extracted the average number of detected faces for each tag per year, based on the set of images in which our system has detected faces. Figure 5.1 compares these calculated averages for different tags and their changes over time. Except from the year 2010, the average number of faces detected for "people" is clearly more than the number for the other tags. T-test analysis also shows significant difference in the number of detected faces in the "people" group and the other tags. For instance, the P-value for comparing "people" and "man" and for "people" and "woman" are 3.59149E-07 and 1.5789E-09, respectively. This shows the images that have more number of faces is more probable to be tagged as people, though the difference is reduced in more recent years.

The above-extracted information can be very useful in finding out how people understand different concepts (for example the difference between the concept of "people" and "man") and how their understanding may change over time. Consid-

Table 5.1: Video Game sample data

Cover Photo	Title	Year	Category	Audience	Platform
	Starcraft	1997	Real time strategy	Teen	PC
	Legend of Zelda: Ocarina of Time	1998	Single Player/Action Adventure	Pre-teen to Young Adult	Nintendo EAD
	Resident Evil 4	2005	Third Person shooter	Mature	Cube, PS2, Wii
	Super Mario Galaxy	2007	Platforming	General Audience	Wii

ering the limitations of the Flickr API, as already explained it is difficult to create an appropriate subsample of uploaded images (see Section 5.1.1) and therefore these findings cannot be easily generalized. This analysis, however, provides some evidence for the ability of our system to extract such cultural trends.

5.2 The Video Game Cover Study

For this study, we used a small data set of video game covers[3]. These are 100 top video games from different categories that have attracted many audiences from all different age groups. As a result, even though the data set is small, it is quite interesting for cultural analysis.

5.2.1 Data Set

The video game data set is small but contains interesting information about the games including cover image, game category, audience, platform, developer, publisher and so on (see Table 5.1 for a small sample of these data). The main problem we had with using this data set was the method that had been used to categorize the data items. Some of the tags were ambiguous, having overlapping intervals or concepts. The age intervals used to categorize audiences into age groups, or the words

[3]The data set was collected by Joyce Uu under supervision of Dr. Rockwell.

Table 5.2: Video Game categories

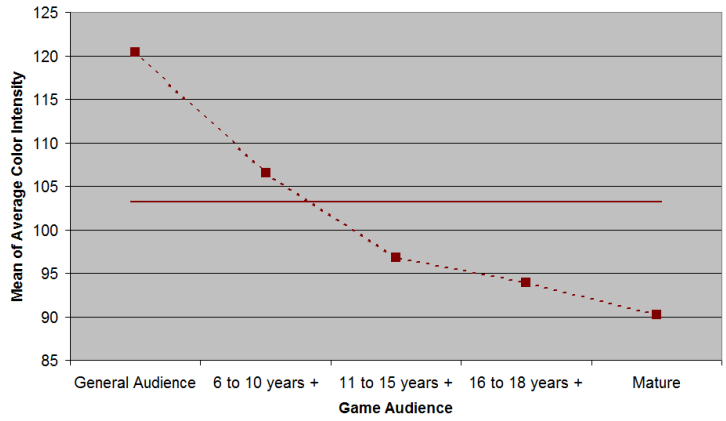
Original Category Name	Count	New Category Name	Count
6 and older	1	6 to 10 years +	8
7 years +	2		
8 years +	2		
8 years +	3		
11 years +	2	11 to 15 years +	13
12 years +	10		
15 years +	1		
16 years +	9	16 to 18 years +	31
17 years +	1		
18 years +	4		
Pre-teen to Young Adult	3		
Teen	10		
Teen and up	1		
Teen to Young Adult	3		
General Audience	7	General Audience	32
All ages	25		
Mature	16	Mature	16
Total	100		100

used to specify the game categories were examples of such ambiguities. Moreover, for some categories or labels, there were not enough samples to be used as a valid basis for analysis. Having these problems, even though the data set was small, we decided to manually "fix" the labels based on the information we obtained from the web. For example, we combined all the age groups younger than 10 in one group since they had very few number of samples in these age groups. Similar rearrangements are applied to other categories or labels (see Table 5.2 for more details on these category mappings). About the game categories, we have considered a video game as a sample of all categories to which it belongs. For example, if a game has been assigned to "Action" and "Adventure" categories, then we have considered it as a sample of both of these categories.

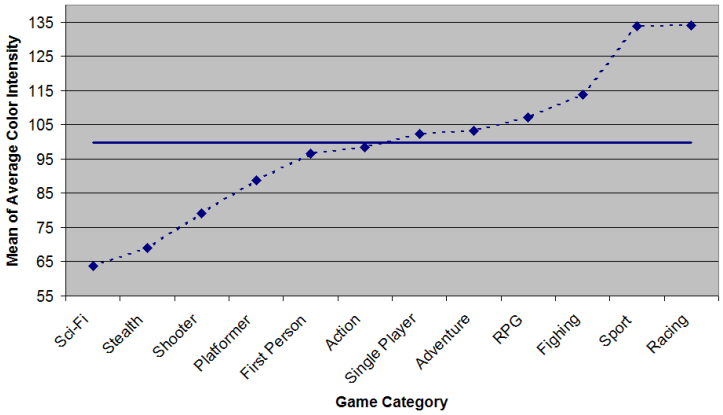
5.2.2 Analysis

With this data set, we wanted to see if there is any meaningful relationship between image features and game audience and categories. In particular, we were interested in the following questions.

1. Are children's games more colorful or brighter than others?
2. Is there any correlation between number of faces and game category?
3. Are there more or less faces in violent games?



(a) Mean of average color intensity of video game cover photos for audiences of different age groups



(b) Mean of average color intensity of video game cover photos for different game categories

Figure 5.2: Analysis of top 100 video games, changes in the mean of average color intensity for different game categories and audiences

4. Is any correlation between size of face "real-estate" and game category?
5. Does any correlation exist with respect to angles and lines and different attributes of video game?

In the following analyses, we have explored some of these questions, specifically questions one to three. First, we examined the relationship of the cover photos brightness and targeted audience in terms of different age groups. Figure 5.2a shows the changes in the mean of the average color intensity of video game cover photos with respect to their audience. Here, average color intensity is taken from the gray channel histogram statistics. Recall from Section 3.1 that the average color inten-

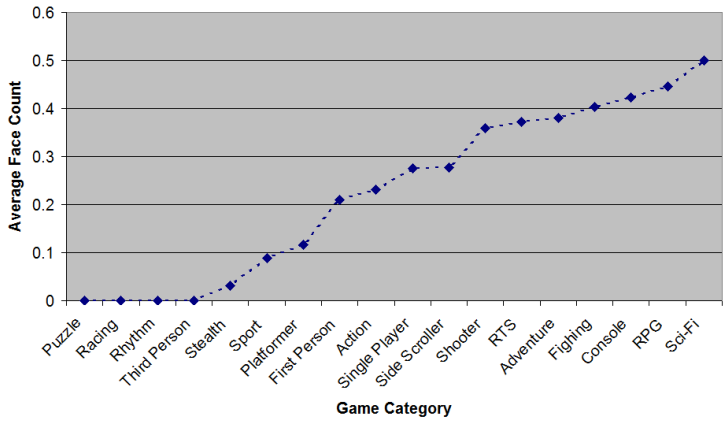


Figure 5.3: Analysis of top 100 video games, average number of detected faces for different game categories

sity represents the brightness/darkness of an image. In fact some distinctions can be perceived for the games developed for audience of different ages. For example, average intensity of video games categorized as "mature" is clearly lower than (or equivalently the cover photos of "mature" category are in average darker than) video games of children younger than 10 years old. More interestingly, the mean of the average intensity of cover photos constantly decreases with the increase in the age of children groups. However, it is difficult to suggest a general trend of intensity changes, first because of the ambiguity in the categorization (as explained in Section 5.2.1) and, second because of the small number of samples in some age groups.

We performed a similar analysis with respect to the game categories. Figure 5.2b shows the result of this analysis. As depicted in this figure, the covers of video game in the "Sport" and "Racing" categories are in average brighter than other categories by far. On the other end, video games under "Sci-Fi", science fiction, and "Stealth" categories are meaningfully darker than the other groups (these two ends are about 35 levels below and above the overall average color intensity). The cover average color intensity for these categories properly represents the underlying concept. The average color intensity level for the other game categories are also more or less representative of their own group.

Table 5.3: IMDB sample data

Cover Photo	Title	Year	Language	Genre	Rating	Gross
	Avatar	2009	English, Spanish	Action, Adventure, Fantasy, Sci-Fi	8.2	760,505,847
	UP	2009	English	Animation, Adventure, Comedy, Drama, Family, Fantasy	8.4	293,004,164
	Toy Story 3	2010	English, Spanish	Animation, Adventure, Comedy, Family, Fantasy	8.6	414,806,932
	Alice in Wonderland	2010	English	Action, Adventure, Family, Fantasy	6.6	334,185,206

For the case of image faces, our analysis shows that the face-detection algorithm does not detect any faces or detect negligible number of faces for game categories such as “Puzzle”, “Racing”, and “Stealth”. On the other hand, more faces are detected on the covers of more violent games such as “Shooter” and “Fighting”, as well as, for “Adventure”, “Sci-Fi”, and “RPG” (Role Playing Game) categories (see Figure 5.3). Nevertheless, it should be clarified that the face-detection algorithm often fails to detect cartoon faces on video game cover photo considering that many of these faces are somehow covered by masks or make-ups and bold features. In the case of this data set, the algorithm detects faces in 45 percent of the images whereas more than 80 percent of the cover photos have face. The average number of detected faces (weighted by face confidence) is 0.29 which is a small fraction (0.13) of the actual average. Such images could be analyzed using other algorithms that are able to for example detect human bodies regardless of the shape or the clarity of the face.

5.3 The IMDB Study

IMDB, the Internet Movie Database, is an online database for information related to visual entertainment media such as movies and television shows, as well as in-

Table 5.4: Single factor ANOVA test for comparing genres average intensity.

Groups	Count	Sum	Mean	Variance
Action	1223	128847.10	105.3533	1555.775
Adventure	1096	126095.20	115.0503	1586.520
Animation	183	22948.16	125.3998	1202.086
Comedy	2451	328876.00	134.1803	1631.264
Drama	3581	418547.30	116.8800	1854.355
Horror	770	70014.90	90.9284	1407.467
Musical	362	47993.96	132.5800	1534.799
Romance	1676	215046.30	128.3093	1817.230

ANOVA: Single Factor			
Source of Variation	SS	df	MS
Between Groups	1645408	7	235058.3
Within Groups	19172630	11334	1691.603
F = 138.9559	$F_{critical}$ = 2.010395	P-value = 2.9E-197	

formation of other related entities such as actors, directors and producers, fictional feature characters, movie user ratings and reviews. In addition, the cover photos for almost all of the movies are also available making these data a valuable source for image exploration and analysis. The advantage of using IMDB over other data sources such as Flickr is that all this information is related to a specific domain and are freely accessible. In fact, all the IMDB information can be downloaded and analyzed.

5.3.1 Data Set

As a more thematic image collection, we focused on analyzing movie posters in IMDB. To that end, we studied the IMDB movie data set and extracted information about movies for seven consecutive decades from 1940-2010 based on top-US-grossing feature films. We ignored the years before 1940 since the data was sparse for those years. The data set we studied involved 7000 movies, 500 top box-office movies for each of five years, downloaded from the IMDB website by using our data-extraction program written based on IMDBPHP. IMDBPHP is a web scraping tool that uses the template of the web page to extract corresponding information fields. Considering the recent changes in the structure of IMDB web pages, we

had several problems using the original program and we had to modify the original source to make it consistent with the new template.

For each movie, in addition to its cover photo, we retrieved its title, the year when it was released, language, country of origin, genre, the user ratings, number of votes, budget and gross income, and director (see Table 5.3 for a sample). It is worth mentioning here that movies are usually associated with more than one genre. Therefore, we considered all the genres of a movie and gave all of them the same importance considering that the IMDB web site does not explicitly mention which genre is the main one.

5.3.2 Analysis

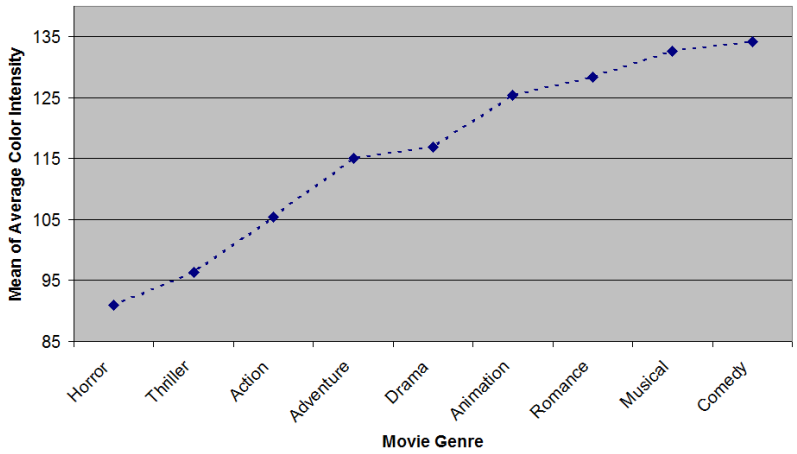
We analyzed our IMDB data set in several ways to explore the following questions.

1. Is there any association between movies' genre and the color or brightness of their cover posters?
2. How might the box-office performance and/or user ratings be related to cover-photo attributes, such as number of faces, and color scheme?
3. How similar/distinct are the images within a genre and across different genres? Are these similarities and distinctions changing over time?

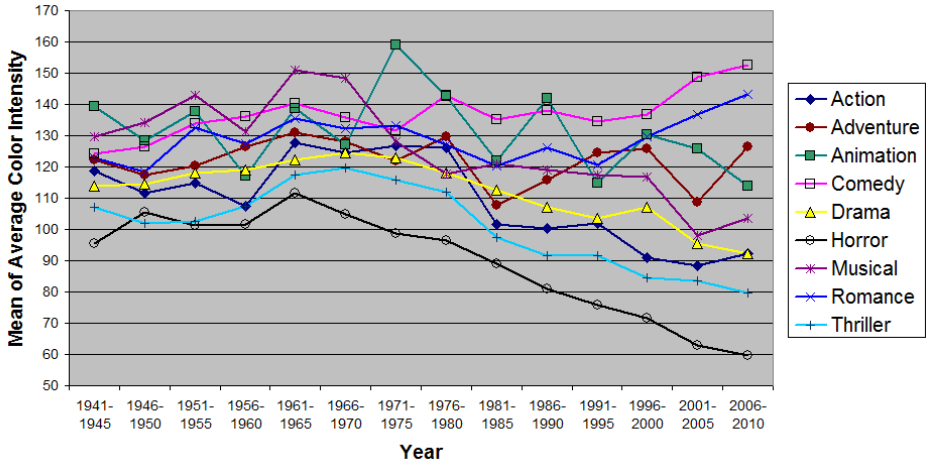
Given the rich IMDB data set (relative to our other data sets), we found several interesting correlations between the features of the movies' cover posters and other movies' attributes and some appealing trends over time and across different genres. These include the relation of cover photo color scheme and number of faces contained with the movies' genres, their box-office performance and their ratings which are all explained in detail in the following subsections.

5.3.2.1 Relation of Genre and Color Intensity of Movie Cover Photo

Figure 5.4 shows the correlation between the genre and the color intensity of the movie cover photo. The implicit question we were interested in answering was "are thrillers or horror films more likely to have dark posters than comedies? and what types of colors are other genres associated with?". As expected and perceived in



(a) Mean of the average color intensity of movie covers over the study period (1940-2010) for different movie genres



(b) Changes in the mean of the average color intensity of different movie genres over time

Figure 5.4: Analysis of IMDB data set 1940-2010, comparing average color intensity of different genres and their changes over time

Figure 5.4a, the mean of the average color intensities, as well as standard deviation (see Table 5.4), are significantly different for different movie genres. On one side of the graph, there are lively, cheerful and romantic movies with bright cover photos whereas some genres such as horror and thriller are on the other side of the graph with dark cover photo. The ANOVA test also shows statistically significant difference between cover intensities of different genres (P-value = 2.9E-197).

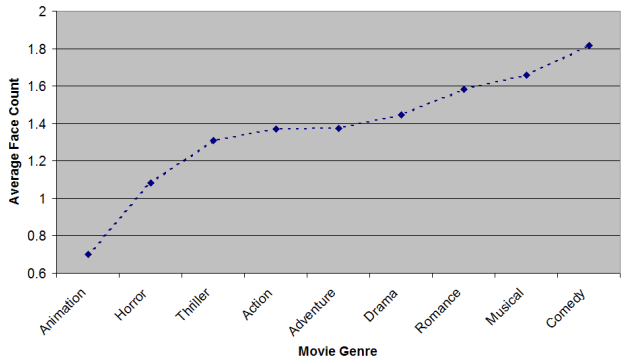
The above observation is about all the movies studied within the specified time

Figure 5.5: Sample of movie's cover photos presenting an abstract concept rather than focusing on a specific star

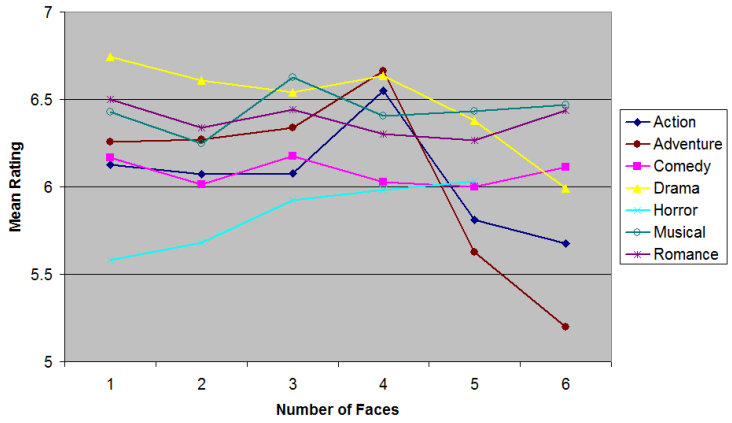
period (1940-2010). What is more interesting is the trend of intensity changes over time (see Figure 5.4b). For example, as one can see in this figure, horror, thriller, and drama cover photos initially were brighter but have become increasingly darker in recent years. In contrast, some other genres such as comedy and romance after some fluctuations over time have started to become brighter in recent years. In addition, no specific trend can be detected for adventure movies which possibly means that the cover designers have been quite "adventurous" in this genre. As another observation, a slight increase in the average intensity of almost all movie genres can be perceived from the early years of the study toward 1960s. Finally, it seems that emergence of computer graphics in 1990 has influenced the design of cover photos as the color intensities more appropriately represent the corresponding genre in recent years. In fact, cover photos show more diversity in recent years compared to for example 1940s or 1960s where it seems that all cover photos are very close in terms of the average intensity values. It is interesting to note that the maximum difference in the mean of the average intensity of different genres is almost doubled from 1940s to 2006 and after (the max difference in 1940s is about 45 intensity levels while it is about 95 in the top movies of the last five years).

5.3.2.2 Relation of Gross Income and User Ratings with Number of Stars

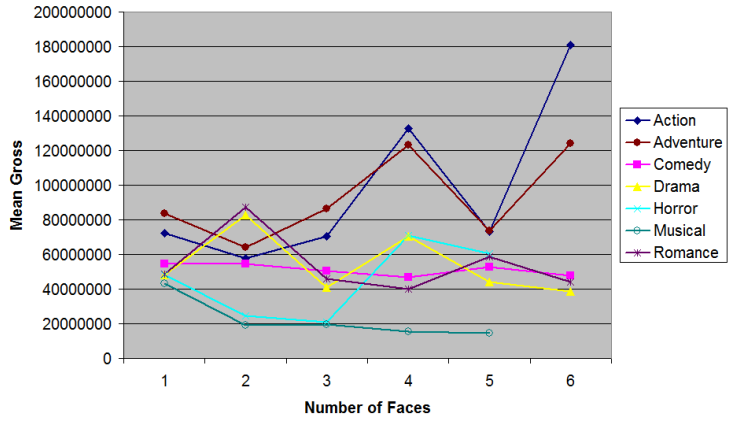
It is a common assumption that having famous actor(s) or actress(es) in a movie will attract larger audiences and will increase the movie income. Assuming that faces on a cover photo mainly represent faces of the main actors, then an interesting question



(a) Overall average face count over the study period (1940-2010) for different movie genres



(b) Mean rating of different movie genres with respect to the number of detected faces



(c) Mean gross of different movie genres with respect to the number of detected faces

Figure 5.6: Analysis of IMDB data set 1940-2010, relation of user rating and gross income with number of detected faces in cover

here is “how the number of faces in a movie cover (the number of stars) impact the box office and also the user rating?”. We studied the correlation between the number of detected faces on the cover photo and the movie’s user rating and gross income. Figure 5.6 shows some of the result of these analyses. Figure 5.6a shows the average number of detected faces for different genres over the whole study period (1940-2010). Similar with the color-intensity analysis, this graph shows that the number of faces detected in comedy, musical, and romance genres are more than others and thriller and horror are among those genres that have the least number of detected faces. The only difference with Figure 5.4a is that animation movies have the smallest number of detected faces. Further analysis on this graph should involve the performance of the face detection algorithm in the sense that the algorithm might be less efficient on darker images as well as cartoon faces (as already mentioned in Section 5.2.2). We have not done these types of analysis on these images as this needed inspection of algorithm results to obtain some ground truth on different categories to compare with automatically detected statistics.

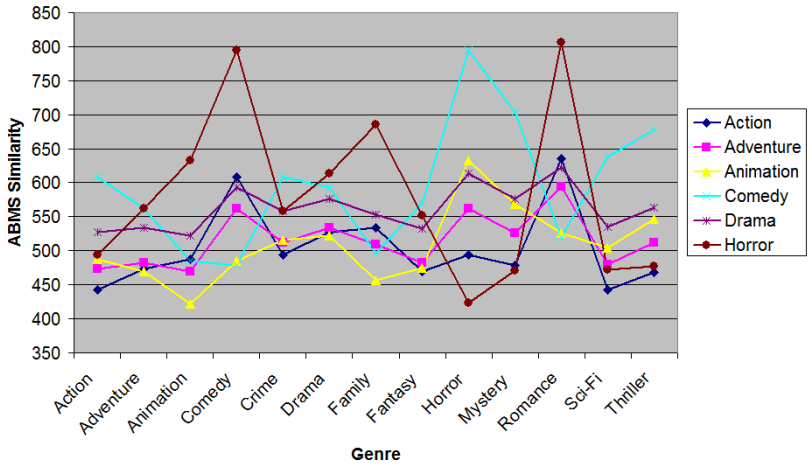
Figure 5.6b shows the mean user-rating changes with the number of detected faces for different genres. Here, we ignored the movie covers having more than six detected faces since such covers contradict our underlying assumption in the sense that they often illustrate an abstract concept rather than being character-centric. Figure 5.5 shows some sample of such cover images where a concept somehow related to the meaning of the movie title is represented by including many faces in the image. Taking this into account, as perceived, in action and adventure movies the user rating increases with the number of detected faces with a peak at four and then rapidly drops. Here, the peak point probably can be interpreted as the optimal number of stars that could be involved in an action/adventure movie to attract the most interest of the audiences. On the other hand, it seems that the mean user ratings of some other genres such as comedy, romance and musical are not sensitive to the number of faces, though it should be considered that the most number of faces are detected in these genres which should be considered on any further analysis on these results. The other observations in this graph are the direct relationship of the user rating with the number of detected faces for horror genre whereas we can see a

reverse relationship in the drama movies. The latter one probably can be explained by the intuition that the drama genre is typically more character-centric than some others.

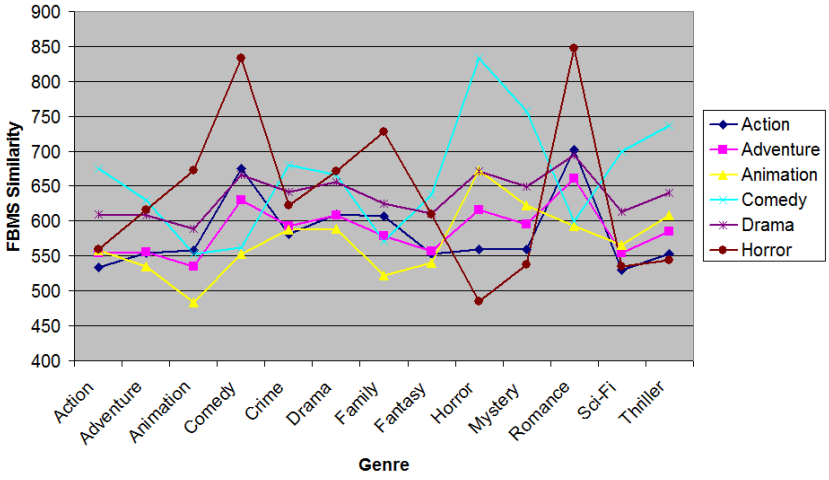
Figure 5.6c shows the same analysis for the mean gross income; there, rather as expected, we can see some trends more or less similar to those were detected for mean user rating. For example, action and adventure movies show an increase in their gross income with the number of detected faces where the peak is at 4 (though it is difficult to justify this trend for the last column of the graph *i.e.,* where number of detected faces are 6). In the same way, comedy genre does not show any significant sensitivity to the number of detected faces and horror genre gross income has a reverse relationship (with a moderate slope) with number of detected faces.

5.3.2.3 Intra- and Inter-Genre Similarities

To examine the intra- and inter-genre similarity for different genres, first we compared ABMS similarity measure (calculating similarity through modeling it as an assignment problem) versus FBMS (fixed block matching similarity) method (see Section 3.4 for further details). Figure 5.7a and 5.7b show inter- and intra-group similarities for six genres (Action, Adventure, Animation, Comedy, Drama, and Horror) versus themselves and some other genres (Crime, Family, Fantasy, Mystery, Romance, Sci-Fi, and Thriller) respectively for ABMS and FBMS methods. As perceived, for this data set, the capability of these methods to differentiate between different groups is almost the same. In other words, even though the absolute similarity value for FBMS is higher than the corresponding value of ABMS, they put different genres at the same distance relative to each other. In general, regardless of some small differences it looks like that the whole graph is shifted up by a similarity value of about 50 when the bottom graph (FBMS) is compared to the upper one (ABMS). This shows that ABMS method is as good as FBMS on conventional data where no transformation or distortion is applied or exist in there. However, as shown via some examples in Section 3.4, ABMS outperforms FBMS similarity method where applied on partly similar or rotated and translated images.

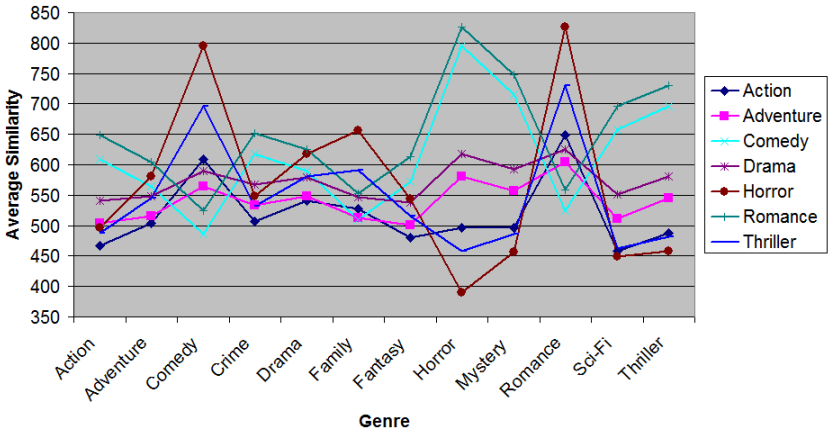


(a) ABMS Similarity

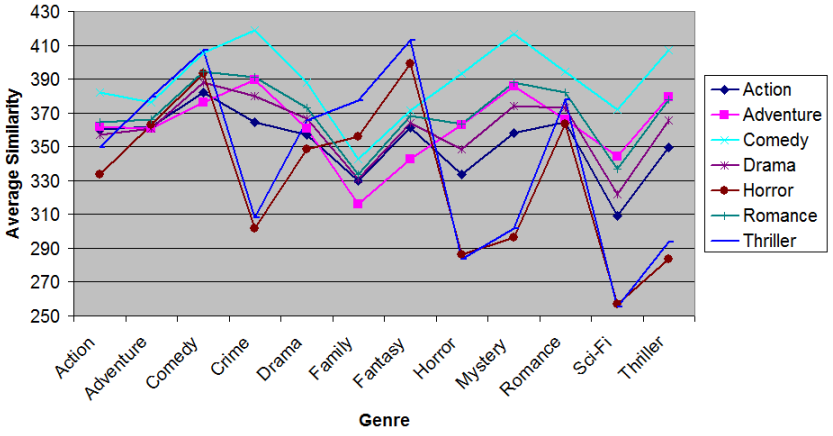


(b) FBMS Similarity

Figure 5.7: Comparing ABMS (calculating similarity through modeling it as an assignment problem) with FBMS (calculating similarity through fixed block matching) similarity calculation methods



(a) 2006-2010



(b) 1941-1945

Figure 5.8: Intra- and inter-genres similarities over different time periods using ABMS similarity measure

We examined the intra- and inter-group similarity for different genres over time to see how these groups respond to ABMS similarity measure and answer questions such as "whether the images within a genre are more similar to each other than images from different genres?". In this regard, we calculated the similarity between pairs of different genres (including the genre self-similarity) over different time periods. Figure 5.8 shows samples of these calculations for the first and the last half decades (1941-1945 and 2006-2010) of the study time interval.

As shown in Figure 5.8a, for the last five years (2006-2010) cover photos often show the most similarity with the images belonging to the their own genre. For example, "Action", "Comedy", and "Horror" genres take the most similarity (the minimum similarity measure value) with the images of their own group. The exceptions are those genres that are commonly assigned to many different movies. For example, according to our data set about 50 percent of the movies are tagged as "Drama". As a result, cover photos of drama movies equally look similar to the movies and all other genres (there is no significant fluctuation in the line representing the "Drama" genre). The other exceptions are more or less related to those genres that are conceptually related. For instance, "Action" has the most similarity with itself but is very close to "Sci-Fi" as well, or "Thriller" is closely similar to "Horror" and "Sci-Fi" genres.

In contrast to the results of the last five years, and consistent with the results depicted in Figure 5.4, no specific distinction is revealed for the cover photos of the first five years of the study period (1941-1945) (see Figure 5.8b). Comparing Figures 5.8a and 5.8b we can see that the intra/inter-genre similarities become more distinct and stable over time. For example, see how all lines are accumulated over each other for the first three genres ("Action", "Adventure", "Comedy") of Genre axis. Also note the significant difference in the range of similarity values on the vertical axis (Average Similarity) in these two charts. Putting all these observations together, we can see how interesting insights can be revealed from such image collections.

Chapter 6

Conclusion and Future Work

In this thesis we examined a feature-based approach to image cultural analytics. Our work makes the following contributions to this area.

- First, we have developed a prototype application that enables the user to explore the image data set by combining several visual features and textual tags through a faceted search interface. This flexible interface supports users in their task of identifying similar images and their associated properties. The program can be used to explore image data sets based on a set of precomputed low- to mid-level features stored in a database where different content and/or tag-based image queries can be made through relating the features of the image under question to these precomputed values. This enables the user to classify or arrange images based on different facets (features) for further analysis and examine higher-level concepts of cultural significance within large image data sets.

- Second, we have written several extractor tools based on some APIs and open source programs to download images and all associated data from different publicly available image data sources. We have used these extractors to create several data sets from IMDB and Flickr.

- Third, we have used our prototype system in conducting several studies on the created data sets including movie cover posters (IMDB data), samples of Flickr images, and cover photos of some top video games. Specifically we have run several analyses on IMDB data as a more comprehensive, thematic

data set. In this domain, we validated our image-similarity metric which is developed based on modeling similarity distance as an assignment problem. We did this through comparing our similarity measure with a fixed block (feature) matching similarity method and also establishing that indeed the metric is higher within a genre than across different genres.

- Finally, we explored the application of our system in cultural analytics to find correlations of (a) poster color schemes with movie genres, (b) faces appearing in posters and the movie's rating and income; indeed, through our analysis we found significant statistical correlations between these low-level image features that our system extracts, and the higher-order concepts of genre, rating and income. Furthermore, we noticed interesting trends in the evolution of these correlations over time and across different movie genres. We also found some interesting trends on other data sets such as sample of Flickr images and top video games.

Our initial analyses provide strong evidence that our system is indeed useful in studying and extracting culturally-relevant information from large digital image repositories. The framework potentially can be extended by adding necessary processes for supporting more visual facets. In this regard, and as part of future work, we are planning to add support for extracting more mid-level facets such as human body detectors, various geometric shapes, and possibly animal faces or bodies detectors. These facets will enable to enhance image classification and find more relevant results. The faceted search mechanism itself can be improved by enabling a hierarchical refining capability based on the facets selected at each level.

We also plan to extend our system to support interactive data visualization and possibly statistical analysis of the image features. In the current version we were mainly depending on the external tools such as R project for statistical computing and MS Office Excel for statistical analysis and visualization of the results. Some development tools such as AXIIS data visualization framework [1] can be used to provide an interactive visualization component for the system. Moreover, the R

[1] AXIIS data visualization framework, `www.axiis.org`

Import/Export capability and its API probably can be used to automate the statistical analysis and exploration. Integrating such tools into the system will guide the user in selecting more distinctive facets in order to more efficiently drill down in a given data set.

Bibliography

[1] Faceted Search with Solr — Enterprise Search support for Apache Lucene and Solr by Lucid Imagination.

[2] Web Article. Corner detection. `http://en.wikipedia.org/wiki/Corner_detection`. Date of Access: September 26, 2011.

[3] Y. Q. Cheng, V. Wu, R. T. Collins, A. R. Hanson, and E. M. Riseman. Maximum-weight bipartite matching technique and its application in image feature matching. In *In Proc. SPIE Visual Comm. and Image Processing*, 1996.

[4] Y. Choi and E. M. Rasmussen. Searching for images: the analysis of users' queries for image retrieval in American history. *Journal of the American Society for Information Science and Technology*, 54(6):498–511, 2003.

[5] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2):1–60, May 2008.

[6] R. Datta, J. Li, and J. Z. Wang. Content-based image retrieval: approaches and trends of the new age. In *MIR '05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 253–262, New York, NY, USA, 2005. ACM.

[7] J. Douglass, W. Huber, and L. Manovich. Understanding scanlation: how to read one million fan-translated manga pages. Forthcoming in Image and Narrative, Winter 2011.

[8] E.D. Gelasca, J. D. Guzman, S. Gauglitz, P. Ghosh, J. Xu, E. Moxley, A. M. Rahimi, Z. Bi, and B. S. Manjunath. Cortina: Searching a 10 million + images database. Technical report, Sep 2007.

[9] C. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151, 1988.

[10] Q. Iqbal and J. K. Aggarwal. Cires: A system for content-based retrieval in digital image libraries. In *in Invited Session on Content-based Image Retrieval: Techniques and Applications, 7 th International Conference on Control Automation, Robotics and Vision (ICARCV*, pages 205–210, 2002.

[11] P. R. Kalva, F. Enembreck, and A. L. Koerich. Web image classification based on the fusion of image and text classifiers. *Document Analysis and Recognition, International Conference on*, 1:561–568, 2007.

[12] F. Long, H. Zhang, and D. D. Feng. Fundamentals of content-based image retrieval. In *Multimedia Information Retrieval*, 2002.

[13] L. Manovich. Cultural analytics: Analysis and visualizations of large cultural data sets. http://www.manovich.net/cultural_analytics.pdf, May 2007. White Paper, With contributions from Noah Wardrip-Fruin, Date of Access: September 26, 2011.

[14] L. Manovich. Cultural analytics: A new field that combines arts, media and IT. http://www.khaleejtimes.com/biz/inside.asp?xfile=/data/marketing/2009/April/marketing_April28.xml&section=marketing, 2009. Date of Access: September 26, 2011.

[15] L. Manovich. *How to Follow Global Digital Cultures, or Cultural Analytics for Beginners*. Transaction Publishers (English version) and Studienverlag (German version), 2009.

[16] L. Manovich and J. Douglass. *Visualizing Temporal Patterns in Visual and Interactive Media*. MIT Press, 2012.

[17] F. Moretti. *Graphs, maps, trees: abstract models for a literary history*. Verso, 2005.

[18] W. Muller, M. Zech, and A. Henrich. Visualflamenco: Faceted browsing for visual features. *Multimedia Workshops, International Symposium on*, 0:71–72, 2007.

[19] K. Rodden, W. Basalaj, D. Sinclair, and K. Wood. Does organisation by similarity assist image browsing? In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '01, pages 190–197, New York, NY, USA, 2001. ACM.

[20] K. Rodden and K. R. Wood. How do people manage their digital photographs? In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '03, pages 409–416, New York, NY, USA, 2003. ACM.

[21] P. Stanchev. Using image mining for image retrieval. *International Journal Information Theories & Applications*, 2003.

[22] J. C. Tang, T. Matthews, J. Cerruti, S. Dill, E. Wilcox, J. Schoudt, and H. Badenes. Global differences in attributes of email usage. In *IWIC '09: Proceeding of the 2009 international workshop on Intercultural collaboration*, pages 185–194, New York, NY, USA, 2009. ACM.

[23] R. C. Veltkamp and M. Tanase. Content-based image retrieval systems: A survey. Technical report, Department of Computing Science, Utrecht University, 2002.

[24] V. Vezhnevets, V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. In *Proc. Graphicon*, pages 85–92, 2003.

[25] R. Villa, N. Gildea, and J. Jose. Facetbrowser: a user interface for complex search tasks. In *Proceeding of the 16th ACM International Conference on Multimedia - MM '08*, pages 489–498, 2008.

[26] P. Viola and M. Jones. Robust Real-time Object Detection. *International Journal of Computer Vision - to appear*, # 2002.

[27] S. Wang. *A robust CBIR approach using local color histograms*, 2001.

[28] S. Westman, A. Lustila, and P. Oittinen. Search strategies in multimodal image retrieval. In *IIiX '08: Proceedings of the second international symposium on Information interaction in context*, pages 13–20. ACM, 2008.

[29] K. M. Wong, K. W. Cheung, and L. M. Po. Mirror: an interactive content based image retrieval system. In *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*, pages 1541 – 1544 Vol. 2, may 2005.

[30] P. Brusilovsky H. Daqing Y. Lin, J. Ahn and W. Real. Imagesieve: exploratory search of museum archives with named entity-based faceted browsing. In *Proceedings of the 73rd ASIS&T Annual Meeting on Navigating Streams in an Information Ecosystem - Volume 47*, ASIS&T '10, pages 38:1–38:10, Silver Springs, MD, USA, 2010. American Society for Information Science.

[31] K. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. *Proceedings of the SIGCHI conference on Human factors in computing systems*.

[32] T. Zepel. *Cultural Analytics at Work: The 2008 US Presidential Online Video Ads*, pages 234–249. Institute of Network Cultures, Amsterdam, 2011.

[33] R. V. Zwol and B. Sigurbjornsson. Faceted exploration of image search results. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 961–970. ACM, 2010.