



# **Multimodal Fusion Method for Image Retrieval System**

**By Raniah Ahmad Alghamdi**

**A thesis submitted for the requirements of the degree of  
Master of Computer Science**

**FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY  
KING ABDULAZIZ UNIVERSITY  
JEDDAH – SAUDI ARABIA  
Sha’aban 1435H – June 2014G**

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

قال تعالى:

(( وَمَا تَوْفِيقِي إِلَّا بِاللَّهِ عَلَيْهِ تَوَكَّلْتُ وَإِلَيْهِ أُنِيبُ )) سورة هود: آية 88



# **Multimodal Fusion Method for Image Retrieval System**

**By Raniah Ahmad Alghamdi**

**A thesis submitted for the requirements of the degree of  
Master of Computer Science**

**Supervised By**

**Dr. Mohammad Abdoulshakoor**

**Dr. Mounira Taileb**


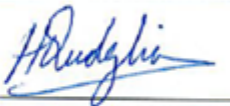


**FACULTY OF COMPUTING AND INFORMATION TECHNOLOGY  
KING ABDULAZIZ UNIVERSITY  
JEDDAH – SAUDI ARABIA  
Sha'aban 1435H – June 2014G**

# Multimodal Fusion Method for Image Retrieval System

By Raniah Ahmad Alghamdi

This thesis has been approved and accepted in partial fulfillment of the requirements for the degree of Master of Computer Science

## EXAMINATION COMMITTEE

	Name	Rank	Field	Signature
Internal Examiner	Prof. Kamal Jambi	Professor	Computer Science	
External Examiner	Dr. Houria Odghiri	Associate Professor	Computer Science	
Co-Advisor	Dr. Mounira Taïleb	Assistant Professor	Computer Science	
Advisor	Dr. Mohammad Abdoulshakoor	Assistant Professor	Computer Science	

KING ABDULAZIZ UNIVERSITY  
Sha'aban 1435H – June 2014G

## **ABSTRACT**

Recently, image retrieval in general and in content-based image retrieval specially became a very important research area used in different fields. From the early days, content-based image retrieval systems suffer from the “semantic gap problem” which is the lack of coincidence between the low level visual features of the image and the high-level human perception. The proposed thesis tries to bridge this gap by designing an image retrieval system for the Web using a multimodal fusion retrieval technique.

The proposed retrieving method utilizes the fusion of the images’ multimodal information (textual and visual) which is a recent trend in image retrieval researches. It combines two different data mining techniques to retrieve semantically related images: clustering and association rules mining algorithm. The semantic association rules mining is constructed at the offline phase where the association rules are discovered between the text semantic clusters and the visual clusters of the images to use it later in the online phase. The experiment was conducted on more than 54,500 images of ImageCLEF 2011 Wikipedia collection. It was compared to an online image retrieving system called MMRetrieval and to the proposed system but without using association rules. The obtained results show that the proposed method achieved the best precision score among different query categories.

## **Dedication**

*This work is dedicated to Mom and Dad, it's impossible to thank you adequately for everything  
you've done...*

*May Allah reward you.*

## **Acknowledgment**

First and above all, I thank Allah, the Almighty, for granting me the capability to complete this thesis. Then, I would like to thank my advisor Dr.Mohammad Abdoulshakoor for his support and helpful comments during the preparation of this thesis. I gratefully acknowledge my advisor Dr.Mounira Taileb for her efforts, patience, motivations, immense knowledge, and never ending guidance; may Allah reward you.

Special thanks goes to my family for their patience and support through all my studies. Last but not the least, I would like to express my sincere gratitude and love to my wonderful parents who deserve the initial honor for their sacrifices, which only Allah knows and can reward.

# TABLE OF CONTENTS

Examination Committee Approval	
ABSTRACT .....	i
Dedication.....	ii
Acknowledgment.....	iii
TABLE OF CONTENTS .....	iv
LIST OF TABLES.....	viii
LIST OF FIGURES .....	ix
LIST OF SYMBOLS AND TERMINOLOGY.....	xi
1 Introduction .....	1
1.1 Motivations .....	2
1.2 Research Problem and Question .....	3
1.3 Goals .....	4
1.4 Methods.....	4
1.5 Contribution .....	5
1.6 Organization of the Thesis .....	5
2 Literature Review .....	7
2.1 Traditional Techniques of Image Retrieval .....	7
2.1.1 Text-Based Image Retrieval (TBIR) .....	7
2.1.2 Content-Based Image Retrieval (CBIR).....	8
2.1.3 Semantic Gap.....	9



2.2	Multimodal Information Fusion.....	10
2.2.1	Fusion Levels.....	11
2.2.2	Fusion Methods .....	12
2.3	Multimodal Fusion in Image Retrieval.....	13
2.3.1	Early Fusion.....	13
2.3.2	Late Fusion .....	14
2.3.3	Trans-Media Fusion.....	17
2.3.4	Image Re-ranking .....	17
2.4	Discussion.....	18
2.5	Association Rules Mining (ARM).....	21
2.5.1	Basic Concepts.....	23
2.5.2	The Process.....	23
2.5.2.1	Discovering Frequent Itemsets .....	24
2.5.2.2	Discovering Association Rules .....	25
2.5.2.3	Apriori Algorithm .....	25
2.6	Association Rules Mining in Image Retrieval .....	28
3	Methodology .....	30
3.1	Planning and Designing the System.....	30
3.1.1	Offline Phase .....	30
3.1.1.1	Features Extraction .....	31
3.1.1.2	Clustering.....	34
3.1.1.3	Association Rules Mining Algorithm in MFAR.....	38
3.1.2	Online phase .....	42
3.1.2.1	Query Modalities and Processing .....	42
3.1.2.2	Retrieve the Related Visual Clusters .....	43

3.1.2.3	Retrieve the Related ARs .....	43
3.1.2.4	Get the Ordered Results List .....	44
3.2	Experiment .....	45
3.2.1	Dataset .....	45
3.2.2	Dataset Topics .....	46
3.2.3	Tools .....	47
3.2.4	Experimental Setup (parameters) .....	48
3.2.5	Steps of the Experiment .....	49
4	Results .....	52
4.1	Generated Association Rules .....	52
4.2	Case Studies .....	55
4.3	Performance measurements .....	57
4.3.1	Precision (P) .....	57
4.3.2	Recall (R) .....	58
4.3.3	Mean Average Precision (MAP) .....	58
4.4	Relevance judgments .....	58
4.5	Performance Results .....	59
5	Discussion and Evaluation .....	63
5.1	The Output of the Offline Phase .....	63
5.2	Response Time .....	64
5.3	Results Evaluation .....	64
5.3.1	In Query by Example Mode .....	64
5.3.2	In Composite Query Mode .....	66
5.4	Discussion .....	67
6	Conclusion and Future Work .....	69

6.1	Conclusion .....	69
6.2	Scope of Future Work .....	70
	References.....	72
	Appendix.....	76

## LIST OF TABLES

Table 2.1 Summary of the researches discussed in section 2.3 .....	19
Table 2.2 Example of transactions of market baskets .....	23
Table 2.3 Apriori Algorithm's notations .....	26
Table 2.4 Apriori algorithm.....	27
Table 3.1 Frequent itemsets mining algorithm based on Apriori .....	41
Table 3.2 Information of the topics of the subset collection .....	50
Table 4.4 The overall values of P@10, P@20, MAP, and Recall of our system without ARs, MMRetrival, and MFAR .....	62
5.1 The number of the best performing topics for each performance measurements of the dataset. .....	67
5.2 The average values of the performance measurements of all the topics difficulty for Sys.1 (MMRetrieval) and Sys.2 (MFAR). .....	67

## LIST OF FIGURES

Figure 2.1 The main processes in CBIR systems [5].....	9
Figure 2.2 A categorization of the multimodal fusion methods [8] .....	13
Figure 2.3 The fusion processes of (a) early fusion; (b) late fusion; (c) trans-media fusion; and (d) images re-ranking.....	19
Figure 2.4 Itemset lattice of $I$ .....	24
Figure 2.5 An illustration of the Apriori principle .....	26
Figure 3.1 Online phase of MFAR .....	31
Figure 3.2 Example of data clustering and the use of the Minimum Bounding Rectangles in direction of the first principal component .....	36
Figure 3.3 Example of K-means clustering .....	37
Figure 3.4 Offline phase of MFAR.....	42
Figure 3.5 Example of an image and its metadata from ImageCLEF 2011 Wikipedia image collection.....	46
Figure 3.6 Topic example: topic# 71 (A) the description of the topic; (B) the five query images; and (C) the relevant images to the queries in the dataset .....	47
3.7 Main GUI of MFAR .....	48
Figure 4.1 Subset of the transaction database for the text clusters (A) $Ct_{192}$ ; and (B) $Ct_{320}$ .....	53
Figure 4.2 ARs containing the text cluster (A) $Ct_{192}$ ; and (B) $Ct_{320}$ .....	53
Figure 4.3 The system interface to display the generated ARs .....	54
4.4 Five case studies .....	56
Figure 4.5 The P@10 values of the dataset topics.....	60
Figure 4.6 The P@20 values of the dataset topics.....	60
Figure 4.7 The Recall values of the dataset topics .....	61
Figure 4.8 The AP values of the dataset topics .....	61

Figure 5.1 The results of the highlighted image query from (A) “colored Volkswagen beetles” - topic# 71, and (B) “two euro coins” - topic# 111.....65

Figure 5.2 Topic 92 “air race” (A) topic description, (B) relevant images of the dataset, and (C) query images.....68

## LIST OF SYMBOLS AND TERMINOLOGY

AR	Association Rule
ARM	Association Rules Mining
CBIR	Content-Based Image Retrieval
CCD	Compact Composite Descriptor
$Cc_i$	Color-based cluster
$Ce_i$	Edge-based cluster
$C_k$	Set of candidate k –itemsets
CombSum	Sum Combination linear fusion method
$Conf(X \rightarrow Y)$	Confidence value of the rule $X \rightarrow Y$
CSD	Color Structure Descriptor
$Ct_i$	Text-based cluster
$Cv_i$	Visual-based cluster
$d(x,y)$	Euclidean distance
EHD	Edge Histogram Descriptor
$I$	Items set
IR	Image Retrieval
$L_k$	Set of large k-itemsets
MAP	Mean Average Precision
MFAR	Multimodal Fusion method based on Association Rules mining
$minconf$	Minimum confidence value
$minsup$	Minimum support value
MPEG-7	Moving Picture Experts Group visual descriptors
NLP	Natural Language Processing

NOHIS	Non Overlapping Hierarchical Index Structure
P	Precision
PDDP	Principal Direction Divisive Partitioning
$s_i$	Normalized score value using Zero-One method
$Supp(X)$	Support value of item or rule
$T$	Transaction database
TBIR	Text-Based Image Retrieval



# 1 Introduction

Today, a huge amount of information exists in electronic formats in the Web and different information repositories, and its size is exponentially growth day after another. This information could be presented in different modalities depending on the application as, for example, text documents, images, audio or videos. With all these vast amount of information, a critical demand arises: what is the use of the information if we cannot find it? Information retrieval systems solve this dilemma. They aim to quickly find useful information within massive data and rank the results by relevance. The type of the retrieval systems depends on the modality of the query and the modality of the retrieved results. Images are important media that exist everywhere and in different applications such as online photo-sharing Websites like Flickr® and Instagram®, Web images like Wikipedia images, specific domains images, or personal collection of images.

Image Retrieval systems (IR) have been used in several applications such as Web images search engines (like Google® and Yahoo!®), fingerprint identification systems, digital libraries, crime prevention systems, medicine and historical researches. They can rely purely on textual metadata such as in Web based image search engines called Text-Based Image Retrieval systems (TBIR). On the other hand, they can filter images based on their visual contents such as colors, shapes, textures or any other information that can be derived from the image itself. Thus, that may provide better indexing and return more accurate results. In that case, IR systems are called Content-Based Image Retrieval systems (CBIR).

Retrieving images based on visual features only led to a gap between the high-level user perceptions and the low-level image features. As a result, the researchers' focus has been shifted to reduce the “semantic gap” between the visual features and the richness of human semantics. Generally, the proverb says, “a picture is worth a thousand words,” and naturally, the interpretation of what we see is hard to express. Also, it is different from person to another depending on his background and needs. In addition, it is even harder to teach a machine how to understand the image contents and the person expectations. For any semantic IR system, to retrieve relevant images to the user query, it is important to offer the user facilities to express his needs, and to index the images in somehow that combines the high level semantic with the low level features. Therefore, there is a great need to design a system that takes benefits from all the related information about the images database.

## **1.1 Motivations**

In addition to the vast volume of images, in different applications especially the Web, there are other important reasons to fuse different modalities in IR systems. We live in a truly multimodal world and humans always take the benefit of each media for sensory interpretation. In Web medium and some other applications, the representation of images can be naturally split into two or more independent modalities such as visual features (color, texture, ... etc.) and textual features (metadata and associated text). So, there is no reason why advantage should not be taken of all available media (images, video, audio, text) to build a useful semantic retrieval system. Moreover, fusion for IR is considered as a trend technique and a novel research area, with very little achieved in the early days of research [1].

The proposed method in this thesis tries to construct a semantic relation between the visual features clusters and the textual features clusters of the images using the association rules mining

algorithm. Using the proposed method for the retrieval process – up to the best of our knowledge – was not used before in the retrieval process in IR systems. The system uses the visual features of the query image to retrieve semantically related text cluster and uses keyword query to support the results, and that does not need to the user feedback. That makes the method more suitable for the Web medium.

## **1.2 Research Problem and Question**

As mentioned previously, the main challenge in IR is the semantic gap. The key problem is how to predict semantic features from primitive visual features. Most of the problems in the current approaches are due to the lack of semantic extraction. There is a need to a system that can interpret the user query according to his/her requirements and make the retrieval operation efficient and accurate. So, choosing the appropriate query modalities to capture the required results is a concern. In addition, constructing a semantic relation between the visual features of the images in the dataset and the textual features for the same dataset is a trivial task and need to study the possible relations that may exist between the two modalities.

As a result, some important questions come to the surface: How can we construct the relationship between primitive and semantic features; and in which level? How can we index the images using their visual and textual features with reducing human intervention and feedback? What is the suitable query format that supplies the system by the required semantic information?

The proposed retrieval system in this thesis tries to address those questions by suggesting and implementing a multimodal fusion method; then, to test the system by conducting an experiment on a carefully selected dataset.

### **1.3 Goals**

The main goal of this thesis is to implement and design a general purpose semantic model for IR system for Web images using multimodal fusion approach. Actually, we want to study if it is possible to reach semantic results by using the visual features of the images. Another aim of this thesis is to review the existing multimodal fusion techniques in multimedia applications and image retrieval systems and to investigate their advantages and weaknesses. That helps to design the architecture of the proposed technique with the suggested algorithm. In addition, we want to compare the efficiency of the proposed design with another well-known multimodal system by implementing the proposed method and conducting an experiment over a general dataset similar to the Web images.

### **1.4 Methods**

In order to satisfy these goals, we have studied the state-of-the-art techniques of IR systems which are TBIR and CBIR, and their advantages and disadvantages. Furthermore, we have investigated the different types of features (visual and textual) that could be used in each system individually, and different clustering and indexing algorithms. For multimodal fusion, we have reviewed the different types of fusion in multimedia applications in general and in IR in particular. To select the appropriate fusion method, there is a need to determine the fusion level, the fusion technique, the used features in the fusion process, the clustering algorithms, the query types, and other things. As a result, a Multimodal Fusion method based on Association Rule mining (MFAR) has been designed and developed for IR. MFAR has been implemented using C#.NET and tested on a subset of ImageCLEF 2011 Wikipedia collection. The experiment results have been evaluated using the precision, recall and the mean average precision

measurements. Finally, the result of MFAR has been compared to the results of another two systems using the same dataset and queries.

## **1.5 Contribution**

The proposed method (MFAR) tries to prove that using the ARs in IR system, which they are constructed between the visual and the textual features of the dataset, will increase the performance of the IR and will provide semantic results. It is considered as a late fusion method. It combines two different data mining techniques for retrieving: clustering and Association Rules Mining (ARM) algorithm. It uses ARM algorithm to explore the relations between text semantic clusters and visual features clusters. The method consists of two main phases: the offline and online phase. In the offline phase, the input is the image dataset which contains the two modalities: the images visual features and their associated text. First, the visual and the textual features should be extracted to run the clustering algorithm independently over them. Then, a modified version of ARM algorithm will identify the relations among the clusters from each modality to construct the semantic Association Rules (ARs). On the other hand, the online phase is the retrieval phase. It uses the generated ARM to retrieve the related images to the query, which could be image query only or image and keyword query.

## **1.6 Organization of the Thesis**

The rest of this thesis is organized as follows. Chapter 2 summarizes the primary research issues and the current state of multimodal fusion in multimedia applications and IR field, and it provides the required background of ARM algorithm. In chapter 3, we introduce the methods used in the different parts of the proposed system and the experimental setup with the used dataset. The generated results after conducting the experiment are presented in chapter 4. The evaluation of the proposed system depending on the experiment results is discussed in chapter 5.

Finally, chapter 6 concludes the thesis and suggests future work. The appendix consists of the final results of the system and an accepted paper in the 16<sup>th</sup> International Conference on Human-Computer Interaction in Greece.

## **2 Literature Review**

This thesis touched on a number of related but different fields. This chapter presents the background material for fusing multimodal information in IR together with the different efforts that have been achieved in this field. Also, the general concept of association rules mining algorithm will be discussed, which is an important background for understanding the proposed method.

### **2.1 Traditional Techniques of Image Retrieval**

The state-of-the-art methods for image retrieval systems include two famous techniques: text-based image retrieval (TBIR) and content-based image retrieval (CBIR). Each method has different advantages and suffers from some drawbacks. The following subsections describes that briefly.

#### **2.1.1 Text-Based Image Retrieval (TBIR)**

In TBIR, the user needs to enter a keyword or phrase to search in the images database as in the existed Web engines such as Google®, Yahoo®, and AltaVista. Such systems depend mainly on the reliable image tags typed by human and on the text correlated to the image in a Web page. Although these tags provide paramount semantic description for the images, most of the Web images come with unreliable tags that lead to incorrect retrieving results. Moreover, the extracted correlated text may unrelated to the image and non-descriptive. Also, these tags and text suffer

from synonymy –which means using different words or phrases to describe the same thing, polysemy –which means using the same words or phrases to describe different things, and ambiguity (multiple interpretations of words or phrases) [2]. In general, any TBIR system needs to a sequence of Natural Language Processing (NLP) steps for indexing. These steps include tokenizing, removing stop words and stemming to create the bag of words model.

### **2.1.2 Content-Based Image Retrieval (CBIR)**

CBIR has become a very interesting research topic in the recent years. “Content-based” means that the technology makes direct use of the visual content of the image rather than relying on human annotation of metadata with keywords. CBIR systems like QBIC [3] and VisualSeek [4] provide the ability to search for a query image by its contents in an image database. The query and the content of the images set are compared and the results are returned based on a similarity algorithm. In fact, it is used mainly for searching in images database of a specific domain, and it is rarely used in Web searching (recently in Google® search engine) or in general purpose system [1].

Figure 2.1 shows the main processes of retrieving images in CBIR. It is grounded on extracting the general visual features of the images in the database and of the query image such as color, shape, and texture. Then for efficient retrieving, each image is represented by a feature vector, or multiple feature vectors, in multidimensional feature space. Each feature vector consists of  $d$  values, which correspond to coordinates in a  $d$ -dimensional space. These feature vectors form the high-dimensional data points in the feature space. To find the similar images vastly, the high-dimensional space need to be indexed. Indexing algorithm is used to index the points of the high-dimensional space. Then, to use this index structure, different retrieval algorithms are designed to perform fast similarity search [5].



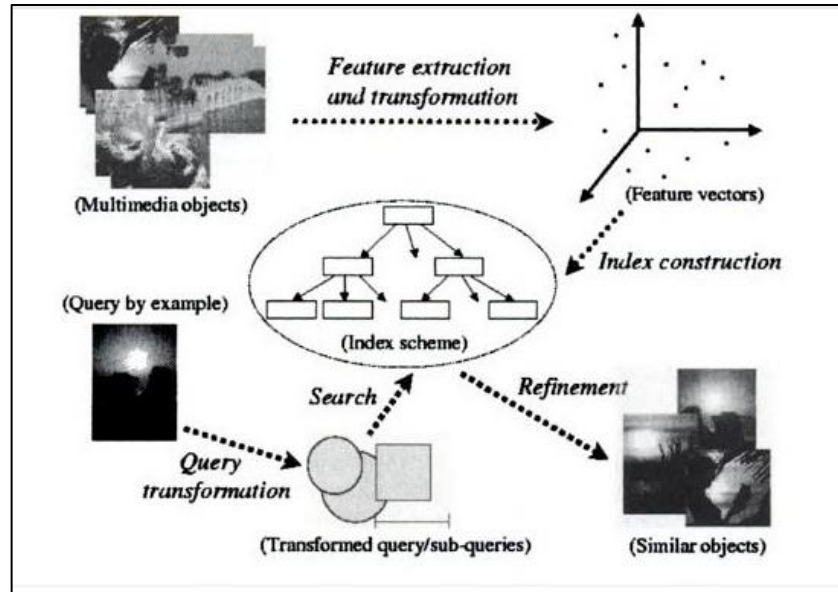


Figure 2.1 The main processes in CBIR systems [5]

### 2.1.3 Semantic Gap

As mentioned in TBIR section, the tags and the related text that belongs to the image contain important semantic information about the image. Most of the images come with unreliable textual information, and typing manually a reliable description for each image in a medium such as the World Wide Web –which contains millions of images–, is inefficient and an expensive solution. Also, having humans who manually enter keywords for images in a large database may not capture every keyword that describes the image. In CBIR, filtering the images based on their content may provide better indexing and return more accurate results in different applications. At the same time, the visual features extracted by the computer are hugely different from the image contents which people understand. It requires the translation of high-level user perceptions into low-level image features and this is the so-called “semantic gap” problem [6]. This problem is the reason behind why the current CBIR systems are difficult to be widely used by users.

Many efforts have been made for bridging this gap by using different techniques. In [7], the authors identified the major categories of the state-of-the-art techniques in narrowing down this

gap. These techniques include: (1) using object ontology to define high-level concepts; (2) using machine learning methods to associate low-level features with query concepts; (3) using relevance feedback into retrieval loop for continuous learning of users' intention; (4) generating semantic template to support high-level image retrieval; (5) fusing the evidences from the text and the visual content of the images. This thesis focuses on the last technique which lately attracted the attention of a lot of researchers. In the next section, fusing multimodal information in general will be discussed first before presenting the accomplished work in multimodal fusion in IR field.

## **2.2 Multimodal Information Fusion**

In the last decades, several research fields found that using information fusion techniques are useful such as in image processing, robotics and pattern recognition [8]. Also, the information retrieval community found the power of fusing multiple information sources on the retrieving performance [9]. As a result, the fusion operation was implemented by several researchers in IR branch using different methods. To analyze these conducted researches, it is important first to define explicitly the “fusion” term. In general, it means “a merging of diverse, distinct, or separate elements into a unified whole”<sup>1</sup>. Several definitions for information fusion process have been proposed in literature depending on the context [10].

In different applications, it is better to fuse multiple modalities in order to reach satisfactory performance instead of using one modality such as in video retrieval and image retrieval. Information fusion has the potential of improving retrieval performance by relying on the assumption that the heterogeneity of multiple information sources allow cross correction of some of the errors, leading to better results [11]. Using different information modalities can provide

---

<sup>1</sup> Merriam Webster Dictionary

complementary information and increases the accuracy of the overall decision making process. It is more robust to use more than one source of information since some modalities can do much to create information while others are unavailable or unreliable [12]. In the other hand, fusing different modalities leads to certain cost and complexity. Different modalities are usually captured in different formats and the processing time of each media is dissimilar. There is a need to study the relation between the modalities if they are correlated or independent [8]. All that characteristics of multiple modalities influence the used fusion process.

To determine the appropriate fusion method, it is important to answer the following basic questions: what is the suitable level to implement the fusion strategy? and how to fuse? These challenges, which may appear in the multimodal fusion process, are stated in the following subsections.

### **2.2.1 Fusion Levels**

Depending on the type of the available information in a certain field, different levels of fusion could be defined. In [13], the authors categorize the fusion levels into two broad types: pre-mapping or early fusion and post-mapping or late fusion. In some literature, they add also the trans-media fusion which is closer to late fusion than early fusion [14]. We attempted to characterize these three families of approaches by distinguishing the inherent steps that they are made.

In the feature level or early fusion approach, the low level features of the modalities are extracted first by the suitable feature extractor. Then, if the extracted feature vectors are not commensurate, the vectors are concatenated into one vector to form one unique feature space [13] [11]. It is the “fusion scheme that integrates unimodal features before learning concepts” [15]. In this approach, the number of features extracted from different modalities may be

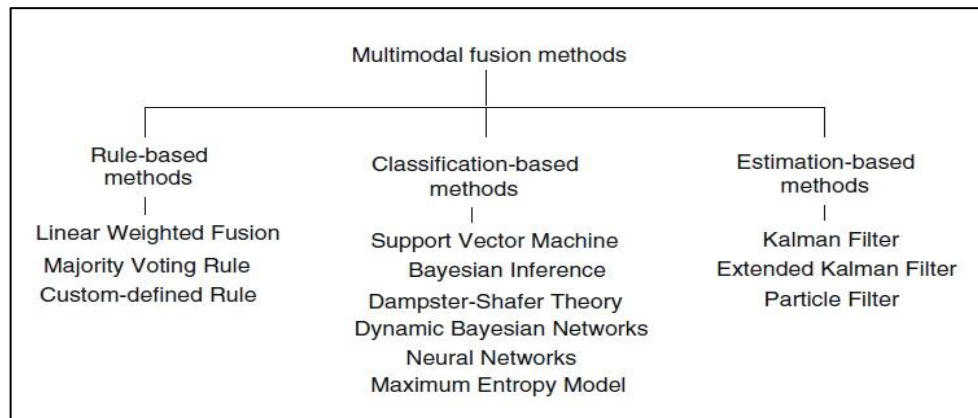
numerous such as visual features, textual features, audio features and motion features. The advantage of this strategy is that it enables a true multimedia representation for all the fused modalities where one decision rule is applied on all information sources. In the other hand, the main drawback is the dimensionality of the resulting feature space which equal to the sum of all the fused subspaces and that leads to the well-known problem the “curse of dimensionality” [11]. Also, the increasing number of modalities and the high dimensionality make them difficult to learn the cross-correlation among the heterogeneous features [8].

On the contrary, late fusion strategies do not act at the level of one representation for all the media features but rather at the level of the similarities among each media. In the decision level or late fusion, the extracted features of each modality is classified using the appropriate classifier then each classifier provides the decision. The classifiers can be of the same type but working with different features (e.g., image and text data), non-homogeneous classifiers working with the same features, or a hybrid of the two types [13]. Unlike feature level fusion, where the features of each modality may have different representation, the decisions usually have the same representation. As a result, the fusion of decisions becomes easier. In addition, it allows for each modality to use the most suitable methods for analyzing and classifying which provides much more flexibility than the early fusion. The main disadvantages of this strategy is that it fails to utilize the feature level correlation among modalities; and using different classifiers and different learning process is time consuming [8].

### **2.2.2 Fusion Methods**

In literature, three different fusion methods are proposed to analyze various multimedia applications: rule-based methods, classification-based methods, and estimation-based methods, as illustrated in figure 2.2 [8]. Choosing the appropriate method for a specific application

depends on the nature of the problem, the nature of the media, and the available parameters. In the rule-based fusion category, a variety of rules are used such as linear weighted fusion (sum and product), MAX, MIN, AND, OR, majority voting, and custom-defined rules for specific applications. On the other hand, the classification-based fusion is used to classify the multimodal observation into one of pre-defined classes. The methods in this category are mentioned in figure 2.2. In some applications, such as the estimating of the moving object in multimedia, the estimation-based fusion is used.



**Figure 2.2 A categorization of the multimodal fusion methods [8]**

### **2.3 Multimodal Fusion in Image Retrieval**

Several researches were achieved in the last decade about using multimodal fusion in IR systems to bridge the semantic gap, and various techniques were proposed for fusing textual and visual information to accomplish that goal. In general, it is possible to categorize them depending on the fusion level to early fusion, late fusion, trans-media fusion and re-ranking researches.

#### **2.3.1 Early Fusion**

As mentioned earlier, in this technique, all the extracted features from different modalities are combined into a single representation. Early fusion could be used without feature weighting like in [16]. The early fusion in [16] is created by concatenating the normalized feature spaces of the

visual and the textual features. Also, the authors compared between the early fusion results and the results of late fusion which uses MINRANK scheme for ranking each document. The results of the experiment show slightly improvement by using early fusion method. In the other hand, feature weighting used in different works in order to provide more weight for specific features. In [17] and [18] as part of ImageCLEF 2006 and 2007, they used the maximum entropy framework to train a logistic model, which can then be used to calculate a score for a query/database image pair. Also, they presented a novel approach to weight features using support vector machines. The entropy-based feature weighting method showed to outperform significantly the performance obtained using a single modality.

In the recent years, kernel-based methods have attracted considerable attention. It could be classified as fusion in intermediate level. Lin and Fuh [19] proposed a kernel-based framework to fuse multimodal information sources for retrieval. Each modality is presented by kernel matrices which combined into an informative one. This approach provided a unified approach for dealing with information fusion among modalities containing large intra-varieties.

### **2.3.2 Late Fusion**

Like early fusion, late fusion starts with extracting the features from each modality. Unlike early fusion, it reduces the features of each model to separately learn concept scores. Then, these scores are integrated to learn concepts [15]. The main drawback of late fusion is the expense of learning for each modality. Late fusion is used widely in IR systems and there is diversity in the proposed methods.

In late fusion approach, the most widely used technique is the rule-based method [20]. Lau *et al.*[20] designed an image retrieval system used for structured XML documents containing heterogeneous images and text information. The content based and the text based search engines

work concurrently, and the final results list are retrieved by fusing the results of each engine. For the text-based system, they used TF-IDF variant; and image features similarity are used for the content-based retrieval system. In the combining process, the results of the text based system are used as the base, while the results of the contents based are used to boost the confidence. Although the system showed improvement in the final retrieval result, they did not investigate the semantic relations between the keywords; and the method it is close to be a filtering method rather than fusion.

Two other different methods for fusing the retrieved results are proposed in [21]. They used a set of generic MPEG-7 descriptors and few other commonly used features. For the image's features, they proposed a fuzzy approach that is based on mapping the features' distances in the feature space to membership functions, and each feature has different rank depending on the location of the query image in the feature space. Then, they fused the features' memberships values and their relevance weights in two methods that could be used to fuse the results in a real-time mode: linear, based on a simple weighted combination, and non-linear, based on the discrete Choquet integral. This system is designed for searching in general purpose database, and it showed improvement in the overall ranking of the retrieved images significantly.

In Scenique[22], a multimodal image retrieval system, providing an integrated query facility, is proposed. It is based on the multi-structure framework which consists of a set of objects together with schema that specifies the classification of objects according to multiple distinct criteria. The retrieved results are the intersection of the retrieved results of image-based with the retrieved results of the text-based followed by images in the text based results only; finally, by the retrieved images of the image based results only. Different rule-based fusion strategies of the outputs of various systems have been studied in [23], such as the maximum combination

(combMAX), the sum combination (CombSun) and the multiplication of the sum and the number of non-zero scores (combMNZ). It showed that using combination of these strategies provides better retrieval results.

The maximum margin model is employed in [24] to capture the dependency information between different modalities. To retrieve the results of an image query, the system searches for the best set of keywords which is represented as a weight vector. Then, the set of images related to those key words are combined with all images annotated with those keywords in the database. They performed online feature level keyword assignment.

In addition, a late fusion method of independent retrieval models (LFIRM) is proposed in [25]. It consists of different independent retrieval systems using the same data set. Each system uses different strategy and different model (either textual features, visual features, or both). After querying, each system returns a ranked list of the retrieved documents. Then, the final results are the combination of all the lists. The results of the experiments showed that using heterogeneous image retrieval systems outperforms the other models with homogeneous systems.

Another examined system is MMRetrieval [26]. It proposes an architecture to retrieve ImageCLEF 2011 Wikipedia Collection. It has an online graphical user interface that brings image and text search together to compose a multimodal and multilingual query. The modalities are searched in parallel, then the results can be fused via several selectable methods. Fusion process consists of two components: score normalization and combination. It provides combination of scores across modalities with summation, multiplication, and maximum. To index the dataset of visual features, the authors used different Compact Composite Descriptors (CCDs).



### **2.3.3 Trans-Media Fusion**

Trans-media fusion is more closer to the late fusion method. Its main idea is to first use one of the modalities (say image) to gather relevant documents (nearest neighbors from a visual point of view), and then to use the dual modalities (text representations of the visually nearest neighbors) to perform the final retrieval. Most proposed methods under this category are based on adopted relevance feedback as in [27] or pseudo-relevance feedback technique as in [28]. In [27], the authors performed the first round retrieval for query set based on the visual features only. In the next rounds, the user's feedback is used to combine the visual and the textual features based on measurement utilized in quantum mechanics and the tensor product of co-occurrence (density) matrices. While in [28], the pseudo-relevance feedback is used to gather the  $N$  most relevant documents from the dataset. To gather the relevant images, it uses some visual similarity measures with respect to the visual features of the query or, reciprocally, a purely textual similarity with respect to the textual features of the query; then to aggregate these mono-modal similarities.

### **2.3.4 Image Re-ranking**

It is another level for fusing the visual and the textual modalities. Image re-ranking first needs to perform the search based on the text query, then the returned list of images is reordered according to the visual features similarity. In other words, image re-ranking constrains the visual system to search among the set of images that was returned by the text-based retrieval instead of dealing with the entire database.

Wei et al. [29] proposed a cross-reference re-ranking strategy for the refinement of the initial search results of text-based video search engines. This method contains three main steps: clustering the initial search results separately for different modality, ranking the clusters by their

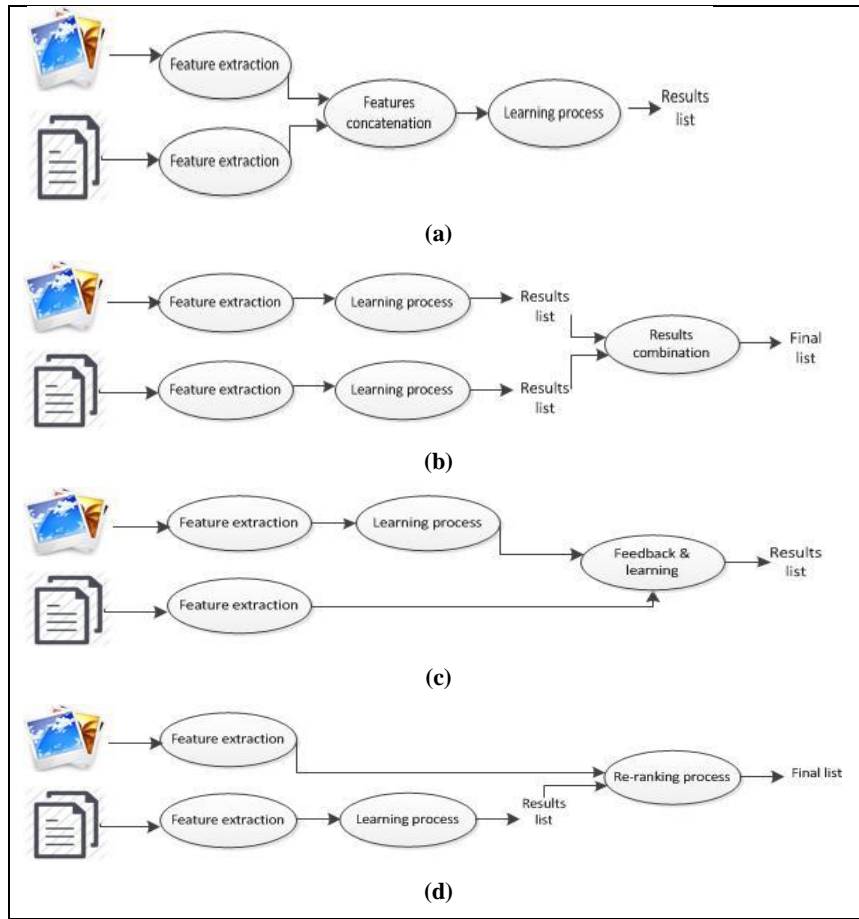
relevance to the query to three predefined rank levels, and hierarchically fusing all the ranked clusters using a cross-reference strategy. The idea behind this method is that, the semantic understanding of video content from different modalities can reach an agreement. While this method deals with the clusters of the modalities, [30] proposed a method that constructs a semantic relation between text (words) and visual clusters using the association rules mining algorithm. They proposed Multi-Modal Semantic Association Rule (MMSAR) algorithm to fuse keywords and visual features automatically for Web IR. It associates a single keyword to several visual feature clusters in inverted file format. Based on the mined MMSARs in inverted files, the results of the text-based image retrieval are re-ranked using the keyword and the MMSARs. The main processes of the four fusion levels are illustrated in figure 2.3, and all the mentioned works of section 2.3 are summarized in table 2.1.

## **2.4 Discussion**

In general, no fusion strategy is optimal, and it should be selected according to the task and the data structure. In image retrieval task, late fusion technique achieved more success than early fusion, and that is because of the well-known problem of the early fusion: curse of dimensionality. Also, the late fusion – either in the result stage or the intermediate stage – provides the ability of individual learning in each modality that increases the gained knowledge from all modalities.

The proposed system in this thesis is considered as a late fusion method. As in [30], MFAR uses ARM for the fusion process. There are three main differences between the method in [30] and the thesis's method. First, while [30] used ARM to construct relations between the keywords and the image clusters, the proposed method uses ARM algorithm to explore the relations between text semantic clusters and image visual feature clusters. Second, their fusion method is used to

semantically re-rank the text-based results while in our work it is used for the retrieval phase. The last main difference is that in our system it is possible to make a query by example image.



**Figure 2.3 The fusion processes of (a) early fusion; (b) late fusion; (c) trans-media fusion; and (d) images re-ranking.**

**Table 2.1 Summary of the researches discussed in section 2.3**

Research	Usage	Fusion method	Used features	Query type
[16]	For database of general images (ImageClef 2008)	Early fusion: Rule-based, linear fusion, concatenating the normalized feature spaces of the visual and the textual features	Color histograms, Texture features, Shape features and text.	Image and text
[17] [18]	IAPR TC-12 photographic collection	Early fusion: classification-based: using support vector machines for the weighted features	Colour Histograms, Global Texture Descriptor, Invariant Feature Histograms,	Image and text

Research	Usage	Fusion method	Used features	Query type
			Tamura Features, GIFT Colour Descriptors, SIFT features, and text	
[19]	For database of general images (COREL)	Early fusion (intermediate level): Kernel-based method	They divided the image representations into three levels, namely, global level, region level, and patch level	Image
[20]	For XML documents (Web)	Late fusion: Rule-based Custom defined <ul style="list-style-type: none"> <li>Result of text based retrieval is the base</li> <li>Image based result used for more confidence</li> <li>The document that appear in both result, should have high rank</li> </ul>	Color histogram, Object Histogram, Hough transform, Texture, UvA Features, and text	Image and text
[21]	For database of general images (COREL)	Late fusion Rule-based <ul style="list-style-type: none"> <li>linear (based on a simple weighted combination)</li> <li>non-linear (based on the discreteChoquet integral)</li> </ul>	A set of MPEG-7 descriptors and other features: Color Structure, Scalable Color, Homogenous Texture, Wavelet Texture, Edge Histogram, and Thesaurus Text Descriptors	Image
[22]	For database of general images (COREL)	Late fusion: Rule-based Intersection of the text based retrieval and content based retrieval	Color, texture and text features	Image and text
[23]	For medical database of ImageCLEF	Late fusion: Rule-based combination of combMAX, combSUM, and combMNZ) strategies	Not mentioned	Image and text
[24]	Berkeley Drosophila embryo image Database	Late fusion: rule-based; max merging	Visual and textual features	Image and text

Research	Usage	Fusion method	Used features	Query type
[25]	For database of general images (IAPRTC12)	Late fusion: rule-based; linear weighted	textual features, visual features, or both	Image or text
[26]	ImageCLEF 2010 Wikipedia Collection	Late fusion: rule-based; linear method: CombSum	Set of visual Compact Composite Descriptors and text	Image and text (multilingual)
[27]	ImageCLEF 2007 photo	Trans-media fusion: rule-based; linear function: tensor product	Visual and textual features	Image
[28]	Set of Wikipedia documents	Trans-media fusion: Rule-based method; linear combination	Visual features and text	Image
[29]	NIST TRECVID'06 benchmark data set	Re-ranking: rule-based method; custom defined (cross-reference)	Visual features: color and edge; and text	Text
[30]	Web images	Re-ranking: rule-based method; custom defined (Multi-Modal Semantic Association Rule)	MPEG7 features and text	Text

In the next subsection, background about ARM algorithm is presented, to help to understand the proposed method.

## 2.5 Association Rules Mining (ARM)

ARM algorithm is a data mining technique. Data mining is the step in the knowledge discovery process that attempts to discover novel and meaningful patterns in data. Knowledge discovery as a process includes an iterative sequence of the following steps [31]:

- **Data Cleansing:** to remove noise and inconsistent data.
- **Data integration:** where multiple data sources may be combined.
- **Data Selection:** where data relevant to the analysis task are retrieved from the database.
- **Data transformation:** where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations for instance.

- **Data mining:** an essential process where intelligent methods are applied in order to extract data patterns.
- **Pattern evolution:** to identify the truly interesting patterns representing knowledge based on some interestingness measures.
- **Knowledge presentation:** where visualization and knowledge representation techniques are used to present the mined knowledge to the user.

The two high-level goals that data miners want to achieve are prediction and description:

- Prediction is used to find patterns which can be used to project the future.
- Description represents discovered patterns to the user in a human-understandable form.

Analyzing association rules between sets of items is useful for discovering interesting relationships hidden in large databases. It is one of the best studied models for data mining [32].

The classical example is the rules extracted from the content of the market baskets. Items are things we can buy in a market, and transactions are market baskets containing several items. Milk, bread, cola, beer and diapers all are examples of the basket's items. Table 2.2, shows several transactions for different customers each with unique identifier TID. In this example, the market basket data is presented in binary format. Each row correspond to a transaction and each column correspond to an item. The collection of all transactions called the transactions database.

For example, a simple association rule extracted from table 2.2 could look as follows:

$$\text{Diapers} \rightarrow \text{Beer}$$

This rule shows that there is a strong relationship between selling diapers with beer because many customers who buy diapers also buy beer. The goal of discovering such rules is to describe the customer's purchase behavior which can aid retail companies to discover cross-sale

opportunities and guide the category management in this way. Besides the market basket data, association rules mining is applicable for different applications of other domains such as bioinformatics, medical diagnosis and Web mining.

**Table 2.2 Example of transactions of market baskets**

TID	Bread	Milk	Diapers	Bear	Eggs	Cola
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

### 2.5.1 Basic Concepts

The problem of mining association rules is stated as following:  $I = \{i_1, i_2, \dots, i_m\}$  is a set of items,  $T = \{t_1, t_2, \dots, t_n\}$  is a transaction database or a set of transactions, each of which contains items of the itemset  $I$ . Thus, each transaction  $t_i$  is a set of items such that  $t_i \subseteq I$ . An association rule is an implication of the form:  $X \rightarrow Y$ , where  $X \subset I$ ,  $Y \subset I$  and  $X \cap Y = \emptyset$ .  $X$  (or  $Y$ ) is a set of items, called itemset. If an itemset contains  $k$  items, it is called  $k$ -itemset. In the association rule of the form  $X \rightarrow Y$ ,  $X$  would be called the antecedent or the left hand side,  $Y$  the consequent or also called the right hand side, as well. It is obvious that the value of the antecedent implies the value of the consequent. Both sides can contain either of a single item or of a subset of items.

### 2.5.2 The Process

The process of mining association rules consists of two main steps. The first step is to identify all the itemsets contained in the data that are adequate for mining association rules. These itemsets have to show at least a certain frequency in the transaction database to be worth mining and are thus called frequent itemsets. The second step is to generate rules out of the discovered frequent itemsets.

### 2.5.2.1 Discovering Frequent Itemsets

To enumerate all the possible itemsets, a lattice structure can be used. Figure 2.4 shows all the possible itemsets for  $I = \{a, b, c, d\}$  in a lattice. In general, if  $I$  contains  $k$  unique items, it is potential to generate up to  $2^k - 1$  different frequent itemsets from it. All the itemsets in the lattice are candidate to be a frequent. To determine that the itemset is frequent, it should satisfy at least the predefined minimum support count. To measure the support count for an itemset, the following formal definition is used:

$$Supp(X) = \frac{count(X)}{N} \quad (\text{Eq. 2.1})$$

where  $N$  is the total number of transactions in the transaction database  $T$  i.e.  $N = count(T)$ . For finding the frequent itemsets, a brute-force approach could be used to calculate the support count for each itemset in the lattice which is computationally expensive since it requires  $O(MNw)$  comparisons, where  $M = 2^k - 1$  is the number of candidate itemsets and  $w$  is the maximum transaction width. Different algorithms attempt to allow efficient discovery of frequent patterns such as the famous *Apriori* algorithm.

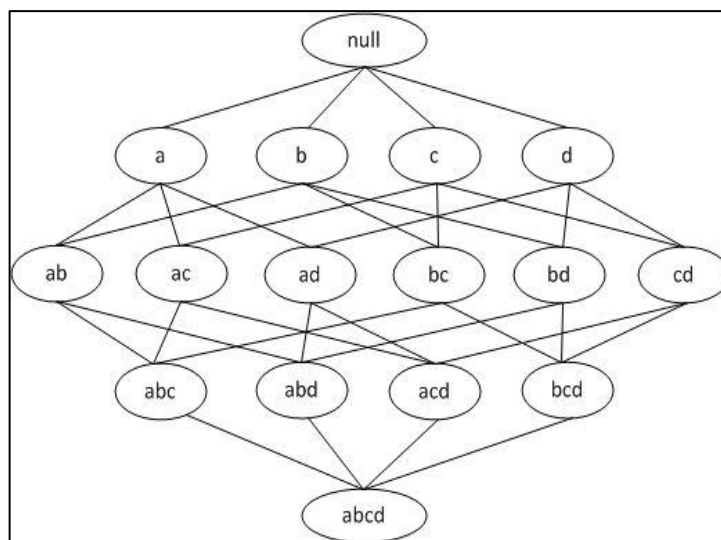


Figure 2.4 Itemset lattice of  $I$



### 2.5.2.2 Discovering Association Rules

After generating all the patterns that meet the minimum support requirements, rules can be generated out of them. For doing so, a minimum confidence has to be defined. The formal definition to calculate the rule confidence is:

$$Conf(X \rightarrow Y) = \frac{count(X \cup Y)}{count(X)} \quad (\text{Eq. 2.2})$$

The confidence of the rule  $X \rightarrow Y$  is a measurement that determines how frequently items in  $Y$  appear in transactions that contain  $X$ , while the support of a rule determines how often a rule is applicable to a given database. The task is to generate all possible rules of the frequent itemsets and then compare their confidence value with the predefined minimum confidence value. Again, different algorithms have been proposed for generating the rules such as *Apriori* algorithm.

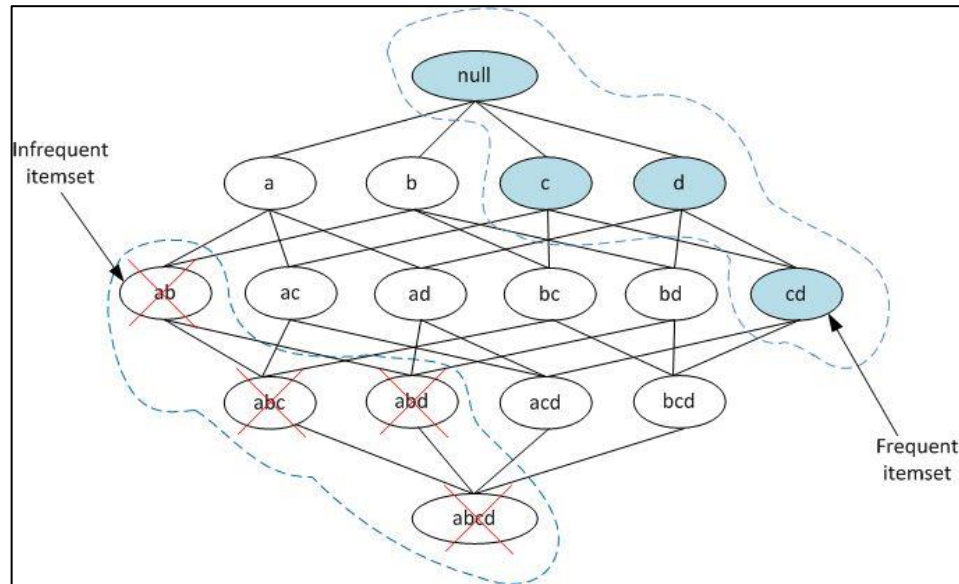
### 2.5.2.3 Apriori Algorithm

The *Apriori* algorithm is the first attempt to mine association rules from a large dataset. It has been presented in [33] for the first time. The algorithm can be used for both, extracting frequent itemsets and also deriving association rules from them.

#### *Frequent Itemsets Generation*

As mentioned earlier, *Apriori* algorithm designed to reduce the computational complexity of frequent itemset generation by reducing the number of candidate itemsets  $M$ . They found an effective method to eliminate some of the candidate itemsets without calculating their support value. To do so, the method stands on the basic theory of *Apriori* algorithm which is "if an itemset is frequent, then all of its subset must also be frequent" [32] which is illustrated in figure 2.5 with frequent itemset  $\{c, d\}$  (shaded sub-graph). Conversely, if an itemset is infrequent according to its support value, then all its supersets are infrequent too and can be pruned

immediately. For example in figure 2.5, the itemset {a, b} is considered to be infrequent. Then, all the supersets that containing {a, b} are pruned.



**Figure 2.5 An illustration of the Apriori principle**

Before representing the pseudo code of the algorithm, the notation of table 2.3 will be used. For each itemset, there is a count field associated with it to store the calculated support value. The pseudo code of the Apriori algorithm for the frequent itemset generation part is given in table 2.4.

**Table 2.3 Apriori Algorithm's notations**

Notation	Definition
$k$ -itemset	An itemset having $k$ items.
$L_k$	Set of large $k$ -itemsets (those with minimum support). Each member of this set has two fields: i) itemset and ii) support count.
$C_k$	Set of candidate $k$ -itemsets (potentially large itemsets). Each member of this set has two fields: i) itemset and ii) support count.

**Table 2.4 Apriori algorithm**

```
1)  $L_1 = \{\text{large 1-itemsets}\};$ 
2) for (  $k = 2 ; L_{k-1} \neq \emptyset ; k++$  ) do
3)   begin
4)      $C_k = \text{apriori-gen}( L_{k-1} );$  // New candidates
5)     for all transactions  $t \in T$  do begin
6)        $C_t = \text{subset}(C_k, t);$  // Candidates contained in  $t$ 
7)       for all candidates  $c \in C_t$  do
8)          $c.\text{count}++;$ 
9)     end
10)     $L_k = \{ c \in C_k \mid c.\text{count} \geq \text{minsup} \}$ 
11) end
12) Answer =  $\cup L_k ;$ 
```

The explanation of the table 2.4 is as following:

- The transactions database is passed over in order to determine the support of each item. As a result of this step, the set of all frequent 1-itemsets,  $L_1$ , will be known.
- Next, the algorithm will iteratively generate new candidate  $k$ -itemsets using the frequent  $(k-1)$ -itemsets found in the previous iteration. To generate the candidate itemsets, the *apriori-gen* function is used which performs two main operations:
  - Candidate generation: it generates new candidate  $k$ -itemsets based on the frequent  $(k-1)$ -itemsets discovered from the previous iteration.
  - Candidate pruning: it eliminates some of the candidate  $k$ -itemsets using the support-based pruning strategy.
- Then, the algorithm needs to make another pass over the database to count the support of the candidates. The *subset* function is used to determine all the candidate itemsets in  $C_k$  that are contained in each transaction  $t$ . A transaction  $t$  is said to contain an itemset  $X$ , if  $X$  is a subset of transaction  $t$ .

- After that, the algorithm will eliminate all the candidate itemsets whose support counts are less than the predefined *minsup*.
- When there are no new frequent itemsets generated, i.e.  $L_k = \emptyset$ , the algorithm stops.

### ***Rules Generation in Apriori Algorithm***

Association rules are allowed to have multiple elements in the antecedent as well as in the consequent. Only frequent (or large) itemsets  $L$  are used to generate the association rules. The procedure starts with finding all possible subsets of the large itemset  $l$ . For each of those subsets, a rule is setup in the form  $z \rightarrow (l - z)$ . Initially, the high confidence association rules with one item in the consequent (or right) part are extracted. Then, all subsets of  $l$  are explored in order to not miss any possible associations. For example, If  $l = \{a, b, c\}$  is a frequent itemset, candidate rules are:  $ab \rightarrow c$ ,  $ac \rightarrow b$ ,  $bc \rightarrow a$ ,  $a \rightarrow bc$ ,  $b \rightarrow ac$ ,  $c \rightarrow ab$ . The number of the candidate rules is equal to  $2^k - 2$  where  $k$  is the length of the itemset, with ignoring the two rules:  $l \rightarrow \emptyset$  and  $\emptyset \rightarrow l$ . In case if a subset  $z$  of  $l$  does not generate a rule with more than or equal the predefined minimum confidence (*minconf*), the subsets of  $z$  should be pruned. This will save computation power that would otherwise be wasted.

## **2.6 Association Rules Mining in Image Retrieval**

In the literature, there are several attempts to couple image retrieval and ARM algorithm. It has been used in image retrieval for different purposes. It was used as a preprocessing strategy for a preliminary reduction of the dimensionality of the pattern space to improve the global search time for CBIR system as in [34]. They used the *Apriori* algorithm in order to discover association rules among the clusters of global MPEG-7 features extracted from the images database.

As mentioned earlier in section 2.3.4, ARM has been used in image re-ranking process proposed in Multi-Modal Semantic Association Rule (MMSAR) algorithm [30]. The common point between the previous works is that the images dataset is Web images i.e. huge and general dataset.

In this thesis, the ARM will be used for the retrieval process. The proposed method (MFAR) tries to provide a semantic IR by constructing a semantic relation between the text semantic clusters and visual feature clusters using *Apriori* algorithm. Then, the generated semantic rules will be used in the retrieval phase. The next chapter shows all the system steps in details along with the used tools and parameters.

## **3 Methodology**

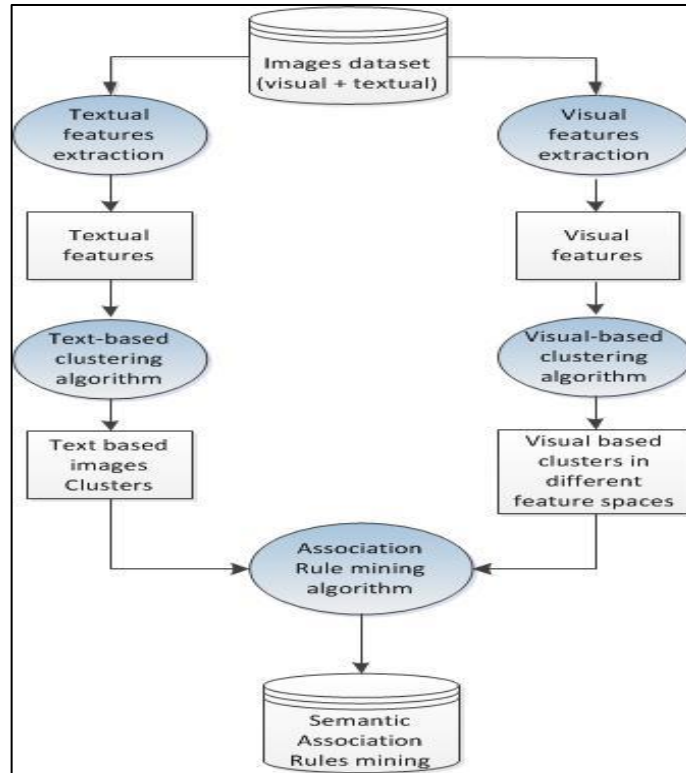
As a result of the previous issues and challenges discussed earlier in chapter 2, an alternative method (MFAR) for fusing multimodal information of general images is proposed. The output of the fusion process is used in the retrieval process to perform semantically better results. This chapter outlines the main stages of the system and the tools employed. It is intended to give an overall picture of the work done, and to explain different aspects of the system design.

### **3.1 Planning and Designing the System**

MFAR consists of two main phases: offline phase and online phase. The next subsections describe in details inputs, outputs and the steps of each phase.

#### **3.1.1 Offline Phase**

As shown in figure 3.1, the input of this phase is the images dataset which contains the two modalities: the images and their associated text. First, the visual and the textual features should be extracted to run the clustering algorithms independently over them. Then, the modified association algorithm will identify the relations among the clusters from each modality to construct the semantic association rules. The following subsections discuss these steps in details.



**Figure 3.1 Online phase of MFAR**

### 3.1.1.1 Features Extraction

#### Visual features extraction

The visual features are defined to capture a certain visual property of the image, either globally for the entire image or locally for a specific group of pixels. In the local feature extraction, a set of features is computed for a set of pixels using its neighborhood pixels. On the other side, the global features are computed to capture the overall characteristics of an image such as color, texture, and shape features. Thus, the overall image is represented by a vector of the feature components where a particular dimension of the vector corresponds to a certain sub-image location. The advantage of using the global features is its high speed for both extracting features and computing similarity [1]. In general, using one global feature is not accurate. So, by

integrating more than one global features, the accuracy could be improved significantly. In general domain image retrieval, it is common to use global features [35].

The goal of this experiment is to illustrate that the proposed fusion approach can improve the retrieval performance over CBIR and other systems. Thus, we did not attempt to optimize the feature extraction component for the used image dataset. Simply, we use a set of generic MPEG-7 (Moving Picture Experts Group) descriptors [36]. The MPEG-7 visual descriptors consist of a varied set of image and video “feature vectors” which describe in a compact fashion various aspects of the visual content. Examples of the features include shape, texture, spatial and temporal location, motion, and color. The features in our system are selected to balance the color and the edge properties of the images. These features are outlined below.

#### ***Color Structure Descriptor (CSD)***

The CSD is extracted from the image in the HMMD (Hue, Max, Min, Diff) color space. It represents an image by both the color distribution of the image (similar to a color histogram) and the local spatial structure of the color. The additional information about color structure makes the descriptor sensitive to particular image features to which is not clear in the color histogram. The CSD is identical in form to a color histogram but is semantically different.

The extraction process of CSD needs three steps. First, a 256-bin color structure histogram is extracted (i.e. accumulated) from an image represented in the 256 cell-quantized HMMD color space. If the image is in another color space, it must be converted to HMMD and re-quantized prior to extraction. Second, if  $N < 256$  is desired (where  $N$  is the number of bins), then bins are unified to obtain a  $N$ -bin color structure histogram. Finally, the values (amplitudes) of each of the  $N$  bins are nonlinearly quantized in accordance with the statistics of color occurrence in



typical consumer imagery. The histogram size is variable and could be 32, 64, 128 or 256. In our experiment, the descriptor size is 64.

### ***Edge Histogram Descriptor (EHD)***

EHD is a useful texture descriptor for similarity search and retrieval with similar semantic meaning. It is designed to represent the spatial distribution, frequency, and directionality of the edges. Each image is divided into  $4 \times 4 = 16$  sub-images. Then, the local-edge distribution for each sub-image can be represented by a histogram. In order to generate the histograms, simple edge detector operators are used to identify edges and group them into five categories: vertical, horizontal, 45 diagonal, 135 diagonal, and isotropic (non-directional). As a result, a total of  $16 \times 5 = 80$  histogram bins are required to represent each image. The experimental results of the MPEG-7 evaluation show that the EHD is quite useful for IR, especially for natural images with non-uniform textures and clip art images.

### **Textual feature extraction**

Before applying clustering methods on unstructured documents collection, we need first to create the vector-space model usually known as bag-of-words model. The central idea of this model is to treat their constituent words (or terms) as features and then describe each document by a vector that represents the frequency occurrence of each term in the document. In this model, no ordering of words or any structure of the text is used. The set of documents is then described by the so-called document-to-term matrix whose  $ij$ -element indicates the frequency (absolute, relative or normalized) of term  $j$  in the document  $i$ . Therefore this matrix has  $ND$  rows and  $NT$  columns where  $ND$  and  $NT$  are respectively the total number of documents and terms. To build the bag-of-words model, it is necessary to perform several linguistic preprocessing steps which include the following components and resources [37].

**Tokenizing the text:** Text tokenization is the task of dividing the text into pieces, called tokens. At the same time, it includes throwing away certain characters, such as punctuation. Tokens are the substrings of consecutive characters that belong together logically. Each token is called term.

**Stop-word remover:** Stop-words (or common words) include terms that are meaningless in the language. They are typically function words (like “is”, “that”, in English) or words that are common in the analyzed body of the text and should be marked as ignored. The general strategy is to collect all the stop words in a list known as a stop list. Then, any token (term) in the documents belongs to that list will be discarded during indexing.

**Stemmer:** Since documents are going to use different forms of a word, such as "organize", "organizes", and "organizing", and there are families of derivationally related words with similar meanings, such as "democracy", "democratic", and "democratization", it would be useful for a search for one of these words to return documents that contain another word in the set. Stemming is the process of returning the words with different grammatical variations into their “base” forms.

### **3.1.1.2 Clustering**

Clustering is the most common form of unsupervised learning. No supervision means that no expert has assigned documents to classes. The goal of dividing the dataset to clusters is to attain high intra-cluster similarity (documents within a cluster are similar) and low-inter cluster similarity (documents from different clusters are dissimilar).

#### **Visual-based clustering**

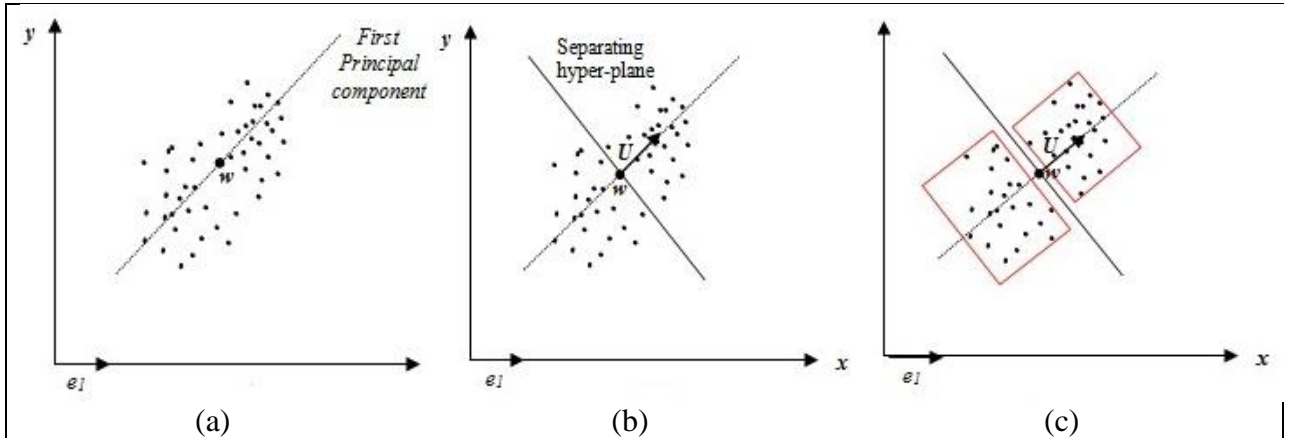
The previous extracted visual descriptors are represented as multidimensional space model. The large quantity of images and the high dimensionality of descriptors need for an efficient

clustering (or indexing) method. To obtain a high-dimensional index, it can be made by using traditional techniques of indexing such as R-tree [38]. Also, it is possible to use a clustering algorithm to form clusters or groups of descriptors, and that clusters are supported by a hierarchical structure. Different high-dimensional index structures have been proposed. The most known and used are data-partitioning based index structure such as SS-tree [39], SR-tree [40], X-tree [41], considered as extensions of R-tree, and space-partitioning based index structure such as k-d-B-tree [42]. Some of the data-partitioning structures suffer from overlapping between bounding regions which influence negatively on the results of query processing. Also, some space-partitioning-based index structures have essential drawback. There is no guarantee of using allocated space which leads to the consultation of few populated or empty clusters [5].

As a result, a high dimensional index technique called NOHIS-tree (Non Overlapping Hierarchical Index Structure) is used in the system [43]. In NOHIS-tree, the overlapping between the bounding forms is avoided and the quality of clusters is preserved. That is satisfied by clustering the high-dimensional descriptors by using data dispersion which guarantee the possibility to avoid empty and few populated clusters by fixing a minimal threshold for the cluster size. In the other hand, the hyper-rectangles bounding forms are directed according to the first principal component which ensures the non-overlapping between any two forms. Also, the results show that NOHIS-tree performs better than SR-tree and sequential scan search when the search is carried out on huge datasets.

This method consists of two phases: offline phase and online phase. In the offline phase, the descriptors are gathered in clusters using the Principal Direction Divisive Partitioning (PDDP) which is one of the divisive hierarchical clustering algorithms. In NOHIS method, they modified the clustering algorithm by using the minimum bounding rectangle (MBR) to avoid overlap.

MBRs are directed according to the principal direction (principal component) used in the clustering algorithm to divide a cluster into two sub-clusters. At the end of this phase, the NOHIS-tree, which is the tree obtained by using PDDP, is constructed. It is not a balanced binary tree. The offline steps are illustrated in figure 3.2 [43]. Then, an adapted k-nearest neighbors algorithm is used to perform a search on NOHIS-tree in the online phase.



**Figure 3.2 Example of data clustering and the use of the Minimum Bounding Rectangles in direction of the first principal component**

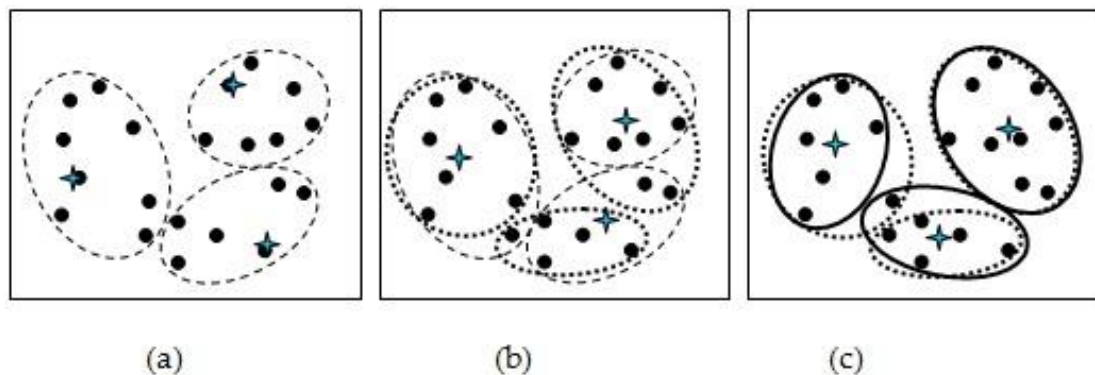
### Text-based clustering

After constructing the bag-of-words model for the related text of the image dataset, we need to group the images based on their related text to set of clusters. Different clustering algorithms are used for text such as flat clustering algorithms and hierarchical clustering algorithms. As it is clear from their names, flat clustering creates a flat set of clusters without any explicit structure that would relate clusters to each other; while hierarchical clustering creates a hierarchy of clusters. Here in the proposed system, there is no need for building a hierarchy of documents. We need only to divide the images to clusters that are semantically related. As a result, K-means algorithm [44] will be used, it is one the most important and well known flat clustering algorithm.

The objective of K-means algorithm is to minimize the average squared Euclidean distance of documents from their cluster centers where a cluster center is defined as the mean or centroid point of the documents in a cluster. The ideal cluster in K-means is a sphere with the centroid as its center of gravity. Ideally, the clusters should not overlap, that means each document is a member of exactly one cluster. We can summarize the clustering algorithm in the following steps:

1. Place  $K$  points (documents) into the space randomly. These points represent initial cluster centroid (seed).
2. Assign each document to the group that has the closest centroid using the distance measurement.
3. When all documents have been assigned, recalculate the positions of the  $K$  based on the current members of its cluster.
4. Repeat steps 2 and 3 until the centroids no longer move or to satisfy the termination condition.

Figure 3.3 shows snapshots from K-means algorithm for a set of points where  $K = 3$ . In our experiment, each text cluster has a unique identifier and a list of the words that belong to the cluster.



**Figure 3.3 Example of K-means clustering**

### 3.1.1.3 Association Rules Mining Algorithm in MFAR

As mentioned previously in chapter 2, to apply the ARM algorithm we need to determine the items set  $I$  and the transaction database  $T$ . In our case, the items set  $I$  is the generated images clusters based on the text (denoted by  $Ct_i$ ) and based on the visual features (denoted by  $Cc_j$  for color-based clusters and  $Ce_k$  for edge-based clusters) where  $i, j$  and  $k$  are the identifiers of the clusters in each modality. After quantifying the features space of all the modalities, we aim to construct the transaction database  $T$  to run the association rules mining algorithm over it.

#### *Constructing the transaction database $T$*

Each transaction  $t$  in  $T$  contains the similar clusters from different modalities. Similarity here means the overlapping degree between the clusters. The similarity of the clusters is estimated in the term of the common images they contain. If the cardinality of the common images set is not equal to zero, the clusters should be combined at the same transaction. It is possible to represent that in the following example:

If  $|Ct_i \cap Cc_j| > 0$ , then add  $\{Ct_i, Cc_j\}$  to  $T$

The hypothesis in constructing  $T$  is that similar clusters tend to be semantically related, therefore, they are combined in the same transaction. We are interested in the association between text clusters and visual feature clusters only. Thus, transactions with similar visual clusters (i.e. in different feature space) and transactions with single cluster should not be included. In our case with using three different features (text and two visual features), we need to make the following number of comparisons to construct  $T$ :

$$\begin{aligned} & \text{no. of text clusters} \times (\text{no. of color clusters} + \text{no. of edge clusters} + (\text{no. of} \\ & \text{color clusters} \times \text{no. of edge clusters})) \end{aligned} \tag{Eq. 3.1}$$

After constructing  $T$ , it is the time to start running the ARM algorithm over it. Since we want to study the relation between the text clusters and the visual clusters, and we want also to find the rules that consist of one text cluster and several visual feature clusters, candidate itemsets  $C_k$  (see section 2.5.2.3) will not start from 1-itemsets. Instead, it will start with 2-itemsets which are the sets that contain similar couple of text cluster and visual cluster. Therefore, only the itemsets containing one text cluster and at least one visual feature cluster are considered. The following are examples of the obtained transactions:  $\{Ct_0, Cc_{111}\}$ ,  $\{Ct_0, Ce_{206}\}$ ,  $\{Ct_0, Cc_{111}, Ce_{173}\}$ .

### ***Calculate support and confidence***

As mentioned earlier in section 2.5.2.1, the role of the support value is to determine how often the rule is applicable to the dataset. While the confidence value determines how frequently items in  $Y$  appear in transactions that contain  $X$  (see section 2.5.2.2). Two different related reasons let us to adjust those formal definitions as in [30]. First, using the standard support/confidence definitions for the semantic rules which are calculated for the entire  $T$ , their support is extremely low, which will affect the generated rules. Second, since we are testing the semantic relations between the text clusters and visual clusters, the calculation of support and confidence is restricted within the result set of the text clusters. Thus, we define the support and the confidence of the rule  $Ct_i \rightarrow Cv_j$  (where  $Cv$  represents the visual cluster) as follows:

$$Supp(Ct_i \rightarrow Cv_j) = \frac{count(Ct_i, Cv_j)}{count(Ct_i)} \quad (\text{Eq. 3.2})$$

$$Conf(Ct_i \rightarrow Cv_j) = \frac{count(Ct_i, Cv_j)}{\max_k(count(Ct_i, Cv_k))} \quad (\text{Eq. 3.3})$$

where  $count(A)$  is the number of itemsets that contain  $A$  in  $T$ . Similarly in case there is more than one item at the right hand side of the rule:

$$Supp(Ct_i \rightarrow \{ Cv_j | j = 1, \dots, m \}) = \frac{count(Ct_i, \{ Cv_j | j = 1, \dots, m \})}{count(Ct_i)} \quad (\text{Eq. 3.4})$$

$$Conf(Ct_i \rightarrow \{ Cv_j | j = 1, \dots, m \}) = \frac{count(Ct_i, \{ Cv_j | j = 1, \dots, m \})}{\max_k(count(Ct_i, Cv_k))} \quad (\text{Eq. 3.5})$$

The modified definitions of support and confidence eliminate the mentioned two problems, and the calculation of support and confidence is restricted within the result set of the text clusters. An example for a possible generated association rule would be  $Ct_0 \rightarrow Cc_4$ . This rule says that if  $Ct_0$  was in a transaction,  $Cc_4$  was in most cases in that transaction too. In other words, there is a strong semantic relation between the images of clusters in the both sides.

### ***Mining frequent itemsets***

We need to use a modified version of the frequent itemsets mining algorithm based on *Apriori* algorithm with Eq. 3.4 and Eq. 3.5 of support and confidence respectively to discover all frequent patterns of the association between text clusters and visual clusters. The frequent itemsets are used later to generate strong ARs. The algorithm is shown in table 3.1 and the used notations are the same as the notations of table 2.3. As stated earlier, the algorithm will not start from 1-itemsets, and that because we want to construct the relationships between text clusters and visual clusters and in case of starting from 1-itemsets it is possible to build relations among visual clusters since they will be treated equally. The minimum support threshold should be given as input to the algorithm. The description of the algorithm steps and the duty of *apriori-gen* and *subset* functions have been mentioned in section 2.5.2.3. The difference here is to insure that each candidate itemsets and each frequent itemsets should have one text clusters at lines 1, 2 and 3 of the algorithm in table 3.1.



### **Generating strong ARs**

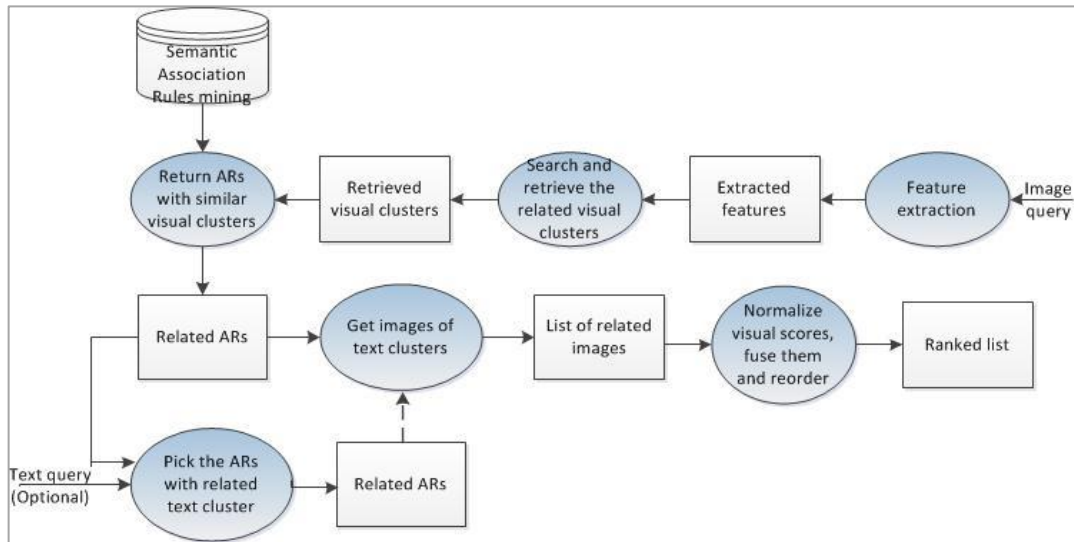
The input of this phase is the generated frequent itemsets  $L$  and the minimum confidence threshold value  $minconf$  and the output is the strong ARs. The generated ARs in our case have one text cluster in the left hand side and one or multiple visual cluster(s) at the right hand side of the AR. There is no need to find all possible subsets of the large itemset  $l$  as in original *Apriori* algorithm (see section 2.5.2.3). Instead, only one possible relation will be generated from each large itemset which contains the text cluster at the antecedent and the rest visual clusters in the consequent. For example, if  $l = \{Ct_1, Cc_3, Ce_1\}$  is a frequent itemset, the candidate rule is:  $Ct_1 \rightarrow \{Cc_3, Ce_1\}$ . If the calculated confidence of the candidate rule using Eq. 3.5 is greater than or equal  $minconf$ , then the rule is strong. Otherwise, it is discarded. Finally, all the generated ARs are stored in the database along with the support and confidence values of each rule which is the final output of this phase.

**Table 3.1 Frequent itemsets mining algorithm based on Apriori**

<p><b>Input:</b></p> <ul style="list-style-type: none"><li>a) The transaction database <math>T</math></li><li>b) <math>minsup</math> threshold</li></ul> <p><b>Output:</b></p> <p>The list of frequently itemsets <math>L</math></p> <ol style="list-style-type: none"><li>1) <math>L_2 = \{(Ct_i, Cv_j) \mid \text{where }  Ct_i \cap Cv_j  &gt; 0 \ \&amp;\&amp; \ (Ct_i, Cv_j).supp \geq minsup\}</math>; //Find all //frequent 2-itemsets</li><li>2) for ( <math>k = 3</math> ; <math>L_{k-1} \neq \emptyset</math> ; <math>k++</math> ) do begin</li><li>3) <math>C_k = \text{apriori-gen}(L_{k-1})</math>; // New candidates with <math>k</math>-itemset with only one text //cluster in it and a combination of frequent sets from <math>L_{k-1}</math></li><li>4) for all transactions <math>t \in T</math> do begin</li><li>5) <math>C_t = \text{subset}(C_k, t)</math>; // Identify all candidates that belong to <math>t</math></li><li>6) for all candidates <math>c \in C_t</math> do</li><li>7) <math>c.count++</math>;</li><li>8) end</li><li>9) <math>L_k = \{c \in C_k \mid c.supp \geq minsup\}</math></li><li>10) end</li><li>11) Return <math>\cup L_k</math>;</li></ol>
--

### 3.1.2 Online phase

It is the retrieval phase. After generating the ARs, it is the time to use it at this phase. The main retrieving processes are illustrated in figure 3.4. In the next sections, each process in this phase is described in detail.



**Figure 3.4 Offline phase of MFAR**

#### 3.1.2.1 Query Modalities and Processing

The used query paradigm in this method is the composite. Composite paradigm involves using one or more of the modalities for querying a system. The basic query model used here is the query by example image since when an image is used as query, all the information it contains is provided to the system. Using a keyword as query is optional. It could be given to the system to support the results generated by the image query. For query image, we need to extract the same visual features that have been extracted from the images of the dataset which are mentioned previously in section 3.1.1.1. As a result, two different vectors should be extracted from the query. For the optional keyword query, we used one keyword and simple text matching to simplify this step.

### 3.1.2.2 Retrieve the Related Visual Clusters

In addition to the generated strong ARs from the offline phase, there is another output from that phase which is NOHIS-tree. It is discussed in section 3.1.1.2. We need to use the same index tree to retrieve the relevant clusters to the query image. In our case, we have two different NOIHS-trees for two different feature spaces. For each feature, the query vector  $q$  will be used to search in the trees and to retrieve the relevant clusters of  $q$ . Relevant cluster is the leaf cluster that contains nearest neighbor(s) objects to  $q$ . The used similarity measurement of images is simply defined as the Euclidean distance between two vectors. The formal definition of Euclidean distance between two vectors  $x = (x_1, x_2, \dots, x_n)$  and  $y = (y_1, y_2, \dots, y_n)$  in  $n$ -dimensional space is as following:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (\text{Eq. 3.6})$$

In our experiment, we will calculate the top 500 nearest neighbors to increase the number of the results and then to return their clusters. The search should be conducted on the trees in parallel. The output of this process is a list of visual clusters from different feature spaces.

### 3.1.2.3 Retrieve the Related ARs

The goal of this process is to use the list of the related visual clusters as input, and then to make sequential search in the ARs mining to retrieve the rules that include that clusters. If the keyword query is provided, the retrieved rules should be filtered to take only the rules containing text clusters that have similar term to the text query. Then, the images' scores in those text clusters should be increased. The dashed arrow in figure 3.4 indicates that it is an optional path.

### 3.1.2.4 Get the Ordered Results List

For all the retrieved ARs, we need to get the images of the text-based clusters. For each image, the relevant score to the query image  $q$  should be calculated if the image is not from the top 500 images for each visual feature. Since the relevance scores are generated from different feature spaces, it is important to normalize the scores before fusing them. Score normalization is very often considered as a preliminary step to data fusion [45]. In our case, we will use Zero-One linear method which maps the scores into the range of [0, 1] [45]. For any retrieval system with  $m$  documents  $d_i$  in the result list where  $1 \leq i \leq m$  where  $m$  is the total number of documents in result list, the Zero-One linear normalization can be done using the following equation:

$$s_i = \frac{r_i - \min\_r}{\max\_r - \min\_r} \quad (\text{Eq. 3.7})$$

where  $\min\_r$  and  $\max\_r$  are respectively the minimal and the maximal score that appear in the results list,  $r_i$  is the raw score of document  $d_i$ . Then, the normalized scores of different modalities should be fused using CombSum method [46] which produces the final scores of the images to reorder them. CombSum (Sum of Combination) is one of a typical data fusion method in information retrieval. To calculate the fused score for each document  $d$  of dataset  $D$ , suppose different results lists  $L_i = \langle d_{i1}, d_{i2}, \dots, d_{im} \rangle$  where  $L_i$  is generated by using different features or different retrieval systems and each document has its relevance scores associated with each of the documents in the list, CombSum uses the following equation:

$$f(d) = \sum_{i=1}^n s_i(d) \quad (\text{Eq. 3.8})$$

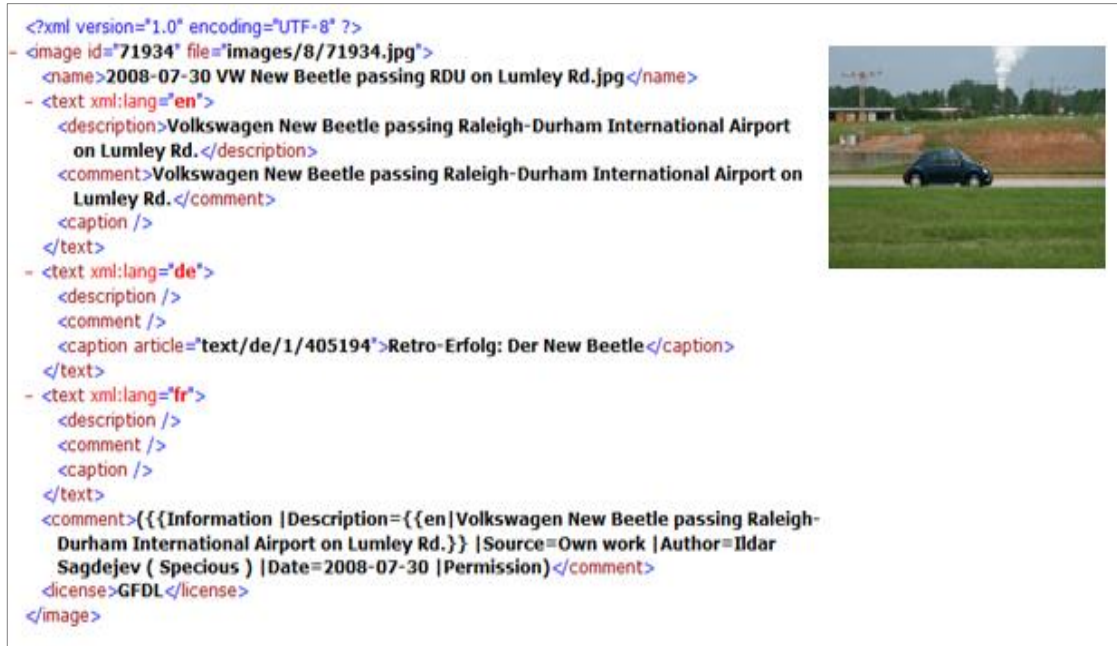
where  $s_i(d)$  is the relevance score of  $d$  in each results list. Then, if there is a keyword query as input, the fused score of each image correlated to term matched the query should be incremented by one. Finally, the final fused list will be reordered based on the fused scores.

## **3.2 Experiment**

In order to evaluate the proposed IR system MFAR, we compared MFAR with two other systems: MMRetrieval and our system without using ARs. As mentioned in section 2.3.2, MMRetrieval is an online system which has GUI and designed to search in ImageCLEF 2011 Wikipedia collection. While our system without using ARs depends totally on the visual features of the images, in MMRetrieval the query could be composite or by example image only.

### **3.2.1 Dataset**

We tested and compared the proposed approach using ImageCLEF 2011 Wikipedia collection. It is a standard collection used by information retrieval community for evaluation purposes. This allows comparison with published results. It consists of 50 topics and 237,434 Wikipedia images along with their user-provided annotations in three different languages (English, French, and German) and the Wikipedia articles that contain these images [47]. Each image has an XML file with its description. An example of an image with its associated metadata is given in figure 3.5. This dataset was our choice because it is a typical example for Web images. In addition, it is available for public use with its ground truth which helps to make fair evaluation with other systems. Because of the abstract semantic content of many of the queries, ImageCLEF 2011 Wikipedia collection is considered to be very difficult for retrieval systems. Since some images in the dataset do not have English description and others do not have description at all, only images with English description are considered in the experiment. Thus, the used dataset is a subset of ImageCLEF 2011 Wikipedia containing 54,545 images.

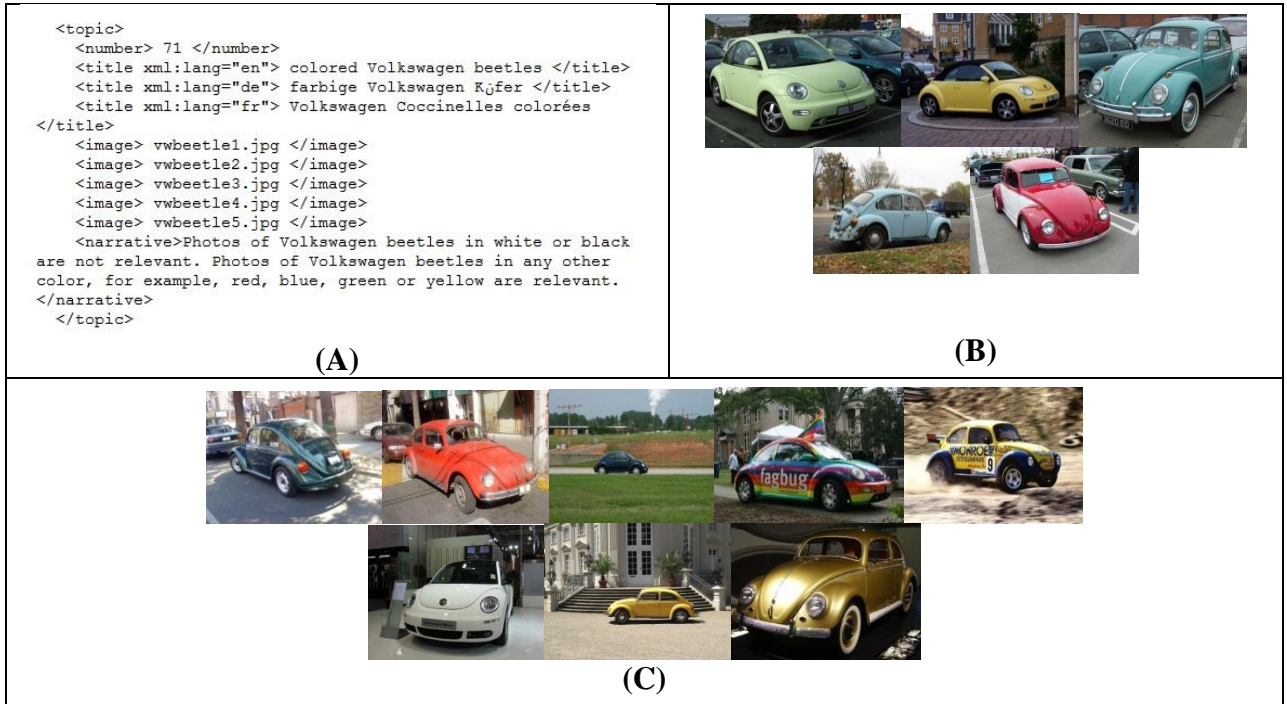


**Figure 3.5 Example of an image and its metadata from ImageCLEF 2011 Wikipedia image collection**

### 3.2.2 Dataset Topics

The 50 topics in the ImageCLEF 2011 Wikipedia collection aim to cover diverse information needs and to have a variable degree of difficulty (Easy: 17 topics, Medium: 12 topics, Hard: 14 topics, and Very hard: 7 topics). Each topic has four to five example images. They were chosen to illustrate, as possible, the visual diversity of the topic. Queries of each topic consist of a multilingual textual part, the query title, and a visual part made of several example images. For assessment, the ground truth of each topic determines if an image is either relevant or not with binary relevance values. Since, the used dataset is a subset of the collection, we have filtered the relevant images to pick the existed images in the subset dataset. Figure 3.6 shows an example of a topic (topic# 71) with its queries, description and relevant images. The topics of the dataset along with their titles, the used text query in the experiment, the number of image examples, and

the number of relevant images in the collection subset are provided in table 3.2 at the end of this chapter (two topics are excluded for ethical reasons).



**Figure 3.6 Topic example: topic# 71 (A) the description of the topic; (B) the five query images; and (C) the relevant images to the queries in the dataset**

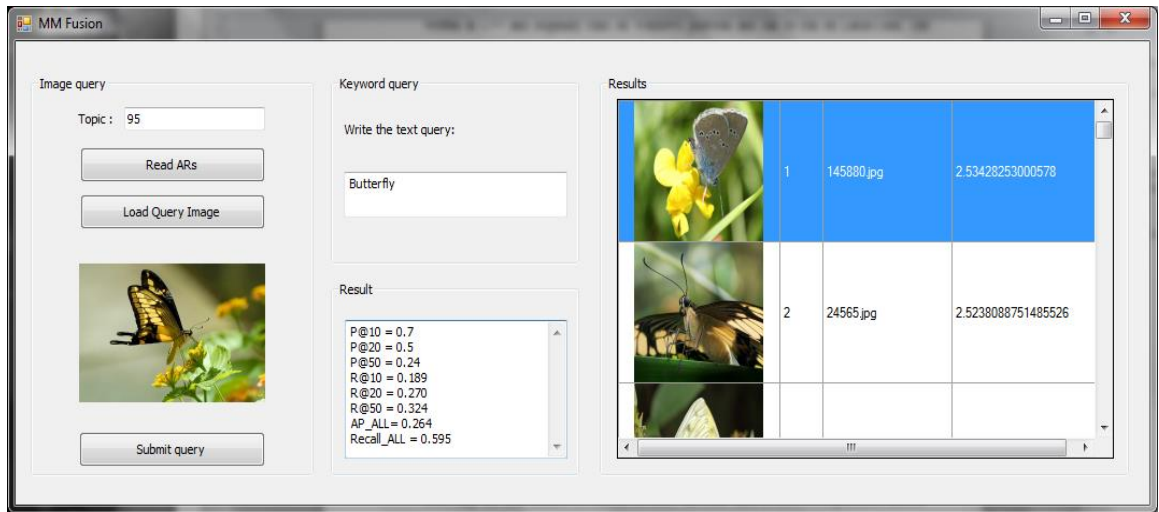
### 3.2.3 Tools

For visual features extraction, the two MPEG-7 descriptors: CSD and EHD are extracted from the dataset using the tool given in [48]. This command line tool for Windows is compiled (C++ and MinGW) using the MPEG-7 Low Level Feature Extraction library. This tool was developed for BilVideo-7 MPEG-7 compatible video indexing and retrieval system. For visual features clustering, NOHIS algorithm library is provided by the author of the algorithm.

For textual features extraction and K-means clustering, Text-Garden software<sup>1</sup> is used. Text-Garden is a software library and collection of software tools for solving large scale tasks dealing

<sup>1</sup> Text-Garden – Text Mining Software Tools. [www.textmining.net](http://www.textmining.net)

with structured, semi-structured and unstructured text. It can be used in different ways covering research and applicative scenarios. Text-Garden is being used by several institutions including British Telecom, Carnegie Mellon University, and Microsoft Research. The code of the library is written in C++ and originally runs on Windows platform and can be run on Linux/Unix. The system prototype of MFAR is developed in C#.NET Framework with simple Graphical User Interface (GUI) for experimental purpose only, as shown in figure 3.7.



### 3.7 Main GUI of MFAR

#### 3.2.4 Experimental Setup (parameters)

The experiment are conducted over MFAR, our system without using ARs, and the online system MMRetrieval. Regarding MFAR, in order to determine the value of *minsupp* and *minconf*, we have run the ARM algorithm five times using different values for *minsupp* and *minconf*. We set *minsupp* and *minconf* to be 2% and 70% respectively because they provided the best association (visually) between the two sides of the ARs. Also according to several experiments on our dataset, we have set the parameters of K-means text clustering algorithm as following:



- The final number of clusters is 1000.
- The seed value is set to be a generated random-number.
- The number of different runs/trials of the algorithm in a search for the best solution is 1 run.

Regarding the NOIHS-tree algorithm, the number of the generated clusters (leaf nodes) has been set to  $\sqrt[2]{54545} = 234$  clusters as stated in [49].

Since MMRetrieval system supports different fusion methods, the well-known method CombSum with MinMax normalization is selected. For our system without using ARs, it depends totally on the visual features of the images. The CSD and EHD scores of each image are fused using CombSum fusion method (see Eq. 3.8).

### **3.2.5 Steps of the Experiment**

Several steps have been done to conduct the experiment in both phases. In the offline phase, we need first to write the extracted vectors of the visual and textual features in external files. Then, the text-based clusters and the visual-based clusters should be written in three different external text files (one for the text-based clusters and the other two for the visual-based clusters) along with their images. Running the adjusted ARM algorithm over the clusters (textual and visual) is the final step in this phase, and then writing the generated strong ARs in a text file. On the other hand, we need to load the ARs from the external file before starting the online phase by pressing the “Read ARs” button. As shown in figure 3.7, the “Submit query” button will be used for retrieving the results. The final ordered results will be displayed in the interface with their scores.

For queries, we have used the examples images of all the topics. For the system without ARs, the query is only the image example. On the other hand, for MFAR and MMRetrieval the query can

be either image only or image with keyword. The text query is restricted to be one keyword only as in table 3.2. MMRetrieval returns the results as qrels (short for "query relevance judgments") file format as following:

*query-number 0 document-id relevance*

where *query-number* is the number of the query, *document-id* is the external ID for the judged documents, 0 is a constant, and *relevance* is the *relevance* assigned to the document for the particular query. *Relevance* is either 0 (non-relevant) or 1 (relevant). Therefore, these files should be processed to evaluate the system performance.

For each topic, all the example images are used in the experiments and the results are calculated. The results and the performance measurements are in the next chapter.

**Table 3.2 Information of the topics of the subset collection**

<b>Topic ID</b>	<b>Topic Title</b>	<b>Text query</b>	<b>No# of Images query</b>	<b>No# of relevant images</b>	<b>Difficulty degree</b>
71	colored Volkswagen beetles	Volkswagen	5	8	Easy
72	skeleton of dinosaur	Dinosaur	5	16	Easy
73	graffiti street art on walls	Graffiti	5	21	Medium
74	white ballet dress	Ballet	5	10	Hard
75	flock of sheep	Sheep	5	10	Easy
76	playing cards	Cards	5	5	Easy
77	cola bottles or cans	Cola	5	6	Easy
79	heart shaped	Heart	5	7	Medium
80	wolf close up	Wolf	4	4	Hard
81	golf player on green	Golf	5	3	Easy
82	model train scenery	Train	5	3	Very hard
83	red or black mini cooper	Cooper	5	3	Medium
84	Sagrada Familia in Barcelona	Sagrada	5	2	Easy
85	Beijing bird nest	Nest	5	4	Hard
87	boxing match	Boxing	5	10	Very hard
88	portrait of Segolene Royal	Segolene	5	6	Easy
89	Elvis Presley	Elvis	4	2	Easy
90	gondola in Venice	Gondola	5	21	Hard

<b>Topic ID</b>	<b>Topic Title</b>	<b>Text query</b>	<b>No# of Images query</b>	<b>No# of relevant images</b>	<b>Difficulty degree</b>
91	freestyle jumps with bmx or motor bike	Bike	5	5	Medium
92	air race	Race	5	4	Medium
93	cable car	Cabling	5	17	Hard
94	roller coaster wide shot	Coaster	5	24	Easy
95	photo of real butterflies	Butterfly	5	37	Hard
96	shake hands	Shake	5	25	Easy
97	round cakes	Cake	5	10	Easy
98	illustrations of Alice's adventures in Wonderland	Alice	4	4	Easy
99	drawings of skeletons	Skeleton	5	14	Hard
100	brown bear	Bear	5	6	Medium
101	fountain with jet of water in daylight	Fountain	5	43	Hard
102	black cat	Cat	5	4	Very hard
103	dragon relief or sculpture	Dragon	5	12	Hard
104	portrait of Che Guevara	Guevara	4	1	Medium
105	chinese characters	Chinese	5	56	Easy
106	family tree	Family	5	15	Medium
107	sunflower close up	Sunflower	5	4	Easy
108	carnival in Rio	Carnival	5	6	Medium
109	snowshoe hiking	Hiking	4	2	Medium
110	male color portrait	Portrait	5	210	Very hard
111	two euro coins	Euro	5	30	Easy
112	yellow flames	Flame	5	23	Hard
113	map of Europe	Europe	5	70	Hard
114	diver underwater	Diving	5	5	Medium
115	flying bird	Flying	5	46	Very hard
116	houses in mountains	House	5	33	Very hard
117	red roses	Rose	4	5	Hard
118	flag of UK	Flag	4	2	Very hard
119	satellite image of desert	Desert	4	21	Hard
120	bar codes	Barcode	4	1	Medium

## 4 Results

This chapter presents the results of the different system phases and the final results of the system. Furthermore, we compared the performance between MFAR, MMRetrival online system and our system without using ARM using different evaluation measurements.

### 4.1 Generated Association Rules

As mentioned previously in section 3.1.1.3, we need to construct the transaction database  $T$ . Each transaction should have one text cluster and one or more visual cluster(s) depending on the overlapping cardinality. In the experiment, the size of  $T$  is 128959 transactions. Figures 4.1.A and 4.1.B show a subset of the database for the text clusters  $Ct_{192}$  and  $Ct_{320}$  respectively. After generating the transaction database, we have to run the ARM algorithm over them. The number of the produced ARs is 6808 rules. Figure 4.2 shows a subset of the rules of the text clusters  $Ct_{192}$  (figure 4.2.A) and  $Ct_{320}$  (figure 4.2.B). These rules of figure 4.2 are generated from the transactions of figure 4.1 using 2% as *minsupp* and 70% as *minconf*.

To study the ARs deeply, the system has another interface for displaying the AR list and the cluster's images in each side of the AR which is shown in figure 4.3. The main two topics of the text cluster  $Ct_{320}$  are "Euro" and "Coin" which are clear in figure 4.3.A. In addition, the right hand side of the highlighted AR in figure 4.3.A is the visual (edge) cluster  $Ce_{217}$  which comprises different circle shapes for diverse objects like coin, plate and round apple pie. Another

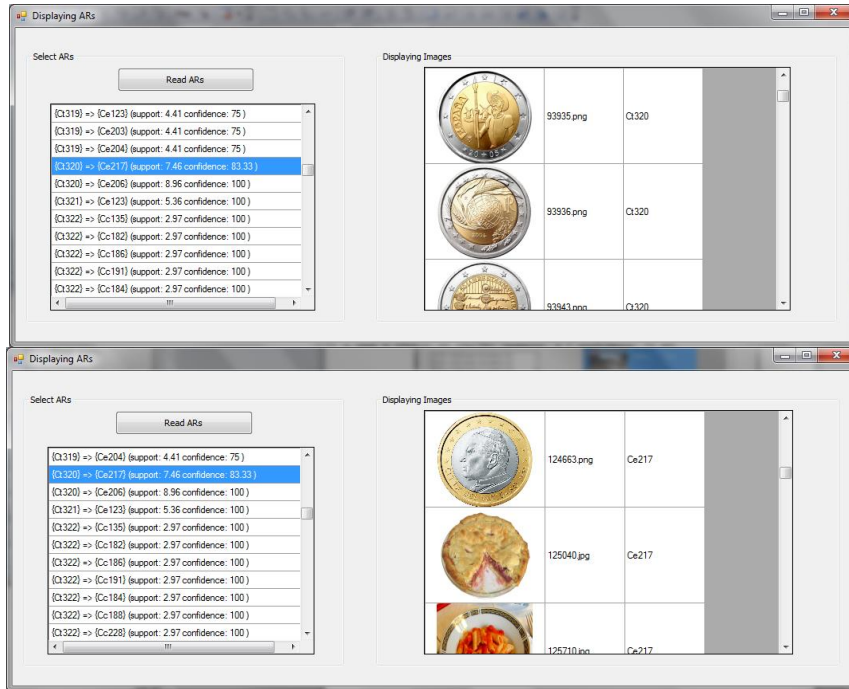
example is shown in figure 4.3.B for the text cluster  $Ct_{192}$  and described by different main labels like “aircraft”, “air”, and “flight” which is associated with three visual clusters, one of them is  $Cc_{180}$ . The color cluster  $Cc_{180}$  contains different images in grayscale of different objects such as “aircraft” and “male portrait”.

{ct192, cc147}	{ct192, ce104}	{ct192, ce225}	{ct192, cc191, ce150}	{ct320, cc124}	{ct320, ce176}	{ct320, cc202, ce189}
{ct192, cc135}	{ct192, ce126}	{ct192, ce222}	{ct192, cc191, ce161}	{ct320, cc139}	{ct320, ce166}	{ct320, cc213, ce217}
{ct192, cc121}	{ct192, ce113}	{ct192, cc147, ce111}	{ct192, cc191, ce141}	{ct320, cc147}	{ct320, ce175}	{ct320, cc213, ce206}
{ct192, cc117}	{ct192, ce111}	{ct192, cc147, ce125}	{ct192, cc201, ce170}	{ct320, cc130}	{ct320, ce189}	{ct320, cc218, ce175}
{ct192, cc122}	{ct192, ce117}	{ct192, cc147, ce222}	{ct192, cc201, ce222}	{ct320, cc125}	{ct320, ce174}	{ct320, cc232, ce121}
{ct192, cc143}	{ct192, ce115}	{ct192, cc135, ce144}	{ct192, cc179, ce111}	{ct320, cc143}	{ct320, ce221}	{ct320, cc228, ce165}
{ct192, cc159}	{ct192, ce128}	{ct192, cc121, ce192}	{ct192, cc179, ce115}	{ct320, cc150}	{ct320, ce190}	{ct320, cc227, ce228}
{ct192, cc142}	{ct192, ce134}	{ct192, cc121, ce200}	{ct192, cc179, ce201}	{ct320, cc157}	{ct320, ce190}	
{ct192, cc126}	{ct192, ce143}	{ct192, cc117, ce155}	{ct192, cc211, ce126}	{ct320, cc154}	{ct320, ce217}	
{ct192, cc144}	{ct192, ce142}	{ct192, cc117, ce213}	{ct192, cc211, ce126}	{ct320, cc162}	{ct320, ce206}	
{ct192, cc136}	{ct192, ce130}	{ct192, cc122, ce160}	{ct192, cc180, ce113}	{ct320, cc174}	{ct320, ce124, ce221}	
{ct192, cc178}	{ct192, ce120}	{ct192, cc143, ce143}	{ct192, cc180, ce128}	{ct320, cc171}	{ct320, cc139, ce217}	
{ct192, cc160}	{ct192, ce158}	{ct192, cc143, ce158}	{ct192, cc180, ce163}	{ct320, cc194}	{ct320, cc147, ce130}	
{ct192, cc182}	{ct192, ce151}	{ct192, cc159, ce117}	{ct192, cc180, ce144}	{ct320, cc177}	{ct320, cc147, ce206}	
{ct192, cc181}	{ct192, ce125}	{ct192, cc159, ce192}	{ct192, cc180, ce175}	{ct320, cc202}	{ct320, cc130, ce206}	
{ct192, cc183}	{ct192, ce154}	{ct192, cc142, ce150}	{ct192, cc180, ce191}	{ct320, cc213}	{ct320, cc125, ce166}	
{ct192, cc192}	{ct192, ce155}	{ct192, cc126, ce211}	{ct192, cc203, ce201}	{ct320, cc218}	{ct320, cc143, ce104}	
{ct192, cc164}	{ct192, ce163}	{ct192, cc144, ce151}	{ct192, cc218, ce154}	{ct320, cc232}	{ct320, cc150, ce174}	
{ct192, cc171}	{ct192, ce153}	{ct192, cc136, ce134}	{ct192, cc196, ce170}	{ct320, cc228}	{ct320, cc150, ce190}	
{ct192, cc172}	{ct192, ce150}	{ct192, cc178, ce120}	{ct192, cc221, ce180}	{ct320, cc227}	{ct320, cc157, ce159}	
{ct192, cc148}	{ct192, ce161}	{ct192, cc178, ce125}	{ct192, cc217, ce216}	{ct320, ce104}	{ct320, cc154, ce217}	
{ct192, cc195}	{ct192, ce144}	{ct192, cc160, ce175}	{ct192, cc220, ce162}	{ct320, ce126}	{ct320, cc154, ce205}	
{ct192, cc189}	{ct192, ce141}	{ct192, cc182, ce222}	{ct192, cc228, ce142}	{ct320, ce121}	{ct320, cc162, ce135}	
{ct192, cc197}	{ct192, ce160}	{ct192, cc181, ce111}	{ct192, cc228, ce153}	{ct320, ce130}	{ct320, cc174, ce126}	
{ct192, cc186}	{ct192, ce180}	{ct192, cc181, ce222}	{ct192, cc228, ce222}	{ct320, ce135}	{ct320, cc174, ce217}	
{ct192, cc177}	{ct192, ce162}	{ct192, cc183, ce125}	{ct192, cc227, ce104}	{ct320, ce154}	{ct320, cc174, ce206}	
{ct192, cc191}	{ct192, ce170}	{ct192, cc192, ce162}	{ct192, cc227, ce113}	{ct320, ce165}	{ct320, cc171, ce176}	
{ct192, cc201}	{ct192, ce175}	{ct192, cc164, ce180}	{ct192, cc227, ce130}	{ct320, ce163}	{ct320, cc194, ce163}	
{ct192, cc179}	{ct192, ce123}	{ct192, cc171, ce162}	{ct192, cc227, ce183}	{ct320, ce159}	{ct320, cc194, ce169}	
{ct192, cc211}	{ct192, ce192}	{ct192, cc172, ce192}	{ct192, cc227, ce219}	{ct320, ce169}	{ct320, cc177, ce154}	
{ct192, cc180}	{ct192, ce213}	{ct192, cc148, ce222}	{ct192, cc227, ce201}			
{ct192, cc203}	{ct192, ce200}	{ct192, cc195, ce216}	{ct192, cc227, ce232}			
{ct192, cc218}	{ct192, ce183}	{ct192, cc189, ce160}	{ct192, cc227, ce225}			
{ct192, cc196}	{ct192, ce226}	{ct192, cc189, ce226}				
{ct192, cc221}	{ct192, ce211}	{ct192, cc197, ce120}				
{ct192, cc217}	{ct192, ce219}	{ct192, cc197, ce125}				
{ct192, cc220}	{ct192, ce191}	{ct192, cc186, ce143}				
{ct192, cc228}	{ct192, ce216}	{ct192, cc177, ce123}				
{ct192, cc227}	{ct192, ce201}	{ct192, cc177, ce213}				
	{ct192, ce232}	{ct192, cc177, ce222}				

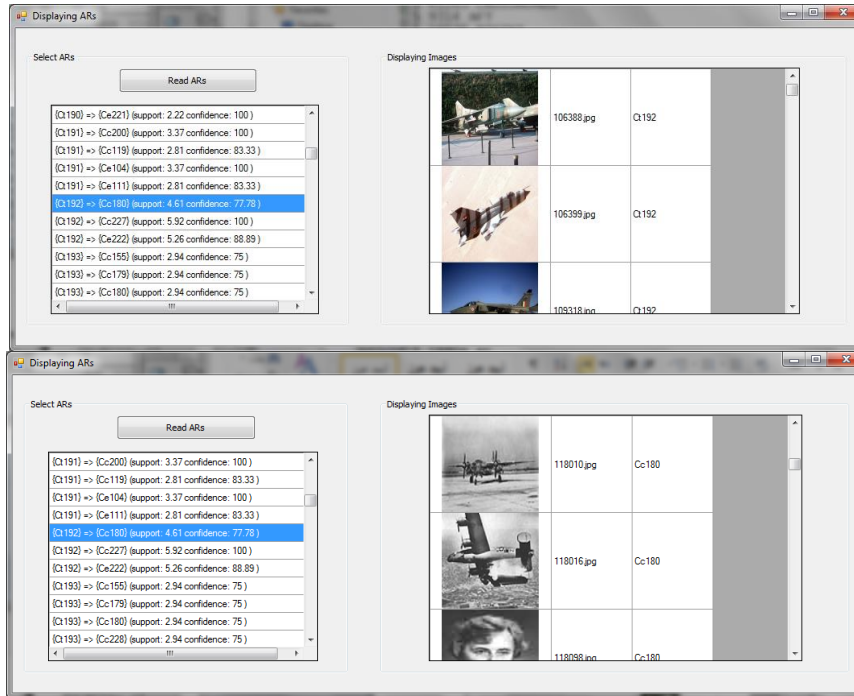
Figure 4.1 Subset of the transaction database for the text clusters (A)  $Ct_{192}$  ; and (B)  $Ct_{320}$

{Ct192}	=>	{Cc180}	(support: 4.61 confidence: 77.78)
{Ct192}	=>	{Cc227}	(support: 5.92 confidence: 100)
{Ct192}	=>	{Ce222}	(support: 5.26 confidence: 88.89)
(A)			
{ct320}	=>	{ce217}	(support: 7.46 confidence: 83.33 )
{ct320}	=>	{ce206}	(support: 8.96 confidence: 100 )
(B)			

Figure 4.2 ARs containing the text cluster (A)  $Ct_{192}$ ; and (B)  $Ct_{320}$



(A)



(B)

Figure 4.3 The system interface to display the generated ARs

## 4.2 Case Studies

In this section, different example queries will be used to show the retrieved ARs and the relations between the query and the retrieved rules. All the ARs shown in figure 4.4 of the case studies are retrieved in the query by image mode and in the composite query (image + keyword) mode as well. Figure 4.4.A shows a query image from the topic number 107 with title “sunflower close up”. Text cluster  $Ct_{645}$  is classified based on different words one of them is “sunflower”. The figure also shows subset of the retrieved ARs of the query which contain the text cluster  $Ct_{645}$ . That means by using the visual features of the query image, it is possible to reach semantically related text clusters. Another query example is shown in figure 4.4.B from the topic number 111. Most of the retrieved ARs for that query include text clusters with “coin” and “Euro” topic ( $Ct_{320}$ ,  $Ct_{484}$ ,  $Ct_{507}$ ). Moreover, the rest three queries in figures 4.4.C, 4.4.D and 4.4.E from the topics 81, 101, and 113 respectively return ARs that combine the visual clusters of the queries to semantically related textual clusters as following:

- For the query of topic 81, the associated text clusters ( $Ct_{47}$ ,  $Ct_{101}$ ,  $Ct_{409}$ ) are described by different words like: “golf”, “green” and “land”.
- For the query of topic 101, the associated text clusters ( $Ct_{289}$ ,  $Ct_{787}$ ,  $Ct_{965}$ ) are described by different words like: “fountain” and “park”.
- For the query of topic 113, the associated text clusters ( $Ct_{127}$ ,  $Ct_{250}$ ,  $Ct_{558}$ ) are described by different words like: “Europe” and “map”.

Furthermore, most of the five example queries of the five topics above retrieved ARs containing the same text clusters. In fact, that was realized in the majority of the topics not just in the examples of figure 4.4.



```
{Ct645} => {Cc180} (support: 2.17 confidence: 75)
{Ct645} => {Cc228} (support: 2.17 confidence: 75)
{Ct645} => {Ce130} (support: 2.17 confidence: 75)
{Ct645} => {Ce169} (support: 2.17 confidence: 75)
{Ct645} => {Ce178} (support: 2.17 confidence: 75)
{Ct645} => {Ce208} (support: 2.9 confidence: 100)
{Ct645} => {Ce196} (support: 2.9 confidence: 100)
```

(A) Image query of topic# 107, “sunflower close up” with the retrieved ARs



```
{Ct320} => {Ce217} (support: 7.46 confidence: 83.33 )
{Ct320} => {Ce206} (support: 8.96 confidence: 100 )

{Ct484} => {Ce217} (support: 7.35 confidence: 100)
{Ct484} => {Ce206} (support: 6.62 confidence: 90)

{Ct507} => {Cc128} (support: 2.96 confidence: 80)
{Ct507} => {Cc187} (support: 2.96 confidence: 80)
{Ct507} => {Ce104} (support: 2.96 confidence: 80)
{Ct507} => {Ce137} (support: 3.7 confidence: 100)
```

(B) Image query of topic# 111, “two euro coins” with the retrieved ARs



```
{Ct47} => {Ce115} (support: 2.94 confidence: 75)
{Ct47} => {Ce174} (support: 3.92 confidence: 100)
{Ct47} => {Ce224} (support: 2.94 confidence: 75)

{Ct101} => {Ce161} (support: 3.3 confidence: 75)
{Ct101} => {Ce173} (support: 4.4 confidence: 100)

{Ct409} => {Cc141} (support: 3.75 confidence: 75)
```

(C) Image query of topic# 81, “golf player on green” with the retrieved ARs



```
{Ct787} => {Cc228} (support: 2.44 confidence: 100)
{Ct787} => {Cc227} (support: 2.3 confidence: 94.12)
{Ct787} => {Ce121} (support: 2.16 confidence: 88.24)

{Ct965} => {Cc175} (support: 4.96 confidence: 92.31)
{Ct965} => {Cc146} (support: 4.55 confidence: 84.62)
{Ct965} => {Cc203} (support: 4.96 confidence: 92.31)
{Ct965} => {Cc176} (support: 5.37 confidence: 100)
{Ct965} => {Ce227} (support: 4.55 confidence: 84.62)

{Ct289} => {Cc151} (support: 2.31 confidence: 71.43)
{Ct289} => {Cc227} (support: 3.24 confidence: 100)
{Ct289} => {Ce186} (support: 2.31 confidence: 71.43)
```

(D) Image query of topic# 101, “fountain with jet of water in daylight” with the retrieved ARs



```
{Ct250} => {Ce173} (support: 5.73 confidence: 100)

{Ct127} => {Cc153} (support: 3.51 confidence: 80)
{Ct127} => {Ce126} (support: 4.39 confidence: 100)

{Ct558} => {Cc168} (support: 2.92 confidence: 80)
{Ct558} => {Ce115} (support: 3.65 confidence: 100)
```

(E) Image query of topic# 113, “map of Europe” with the retrieved ARs

#### 4.4 Five case studies



### 4.3 Performance measurements

As stated earlier, the goal of the system is to perform semantic search in IR system. Thus, after conducting the experiment, efficiency of the method is a concern. To evaluate the accuracy of MFAR, we need to analyze the retrieved results. Thus, different measurements are used to measure the performance of the IR system. The main purpose of measuring the performance is to compare MFAR system with other retrieval systems to determine the success of the proposed design. In the literature, the most widely used evaluation metrics are precision, recall, mean average precision, and recall/precision graph. In recall/precision graph, precision is measured at a set of standard recall points for each topic in the test collection. We used the first three measurements that are commonly used in ImageCLEF [50] since there was a growing trend in many parts of the image retrieval research community to move from the graphical presentations to a single value measure. That is because usually recall/precision graph showed similar characteristics of each plotted system which is not appropriate in comparing process [51].

#### 4.3.1 Precision (P)

In information retrieval domain, precision measures the accuracy of the retrieved result. It is defined as the ratio of the number of retrieved relevant images to the total number of retrieved images list; and it could be represented as following [50]:

$$Precision = \frac{|relevant\ documents \cap retrieved\ documents|}{|retrieved\ documents|} \quad (\text{Eq. 4.1})$$

Since most of the IR systems return the results as ranked lists and users rarely examine more than the first returned images, the precision definition was needed to be adapted. A common approach for measuring precision over a ranked images list is to measure at a fixed rank position by ignoring all documents retrieved below the fixed position. Precision at a fixed rank  $n$  (symbolically  $P(n)$ ,  $P@n$  or  $P_n$ ) is simply defined as:

$$P(n) = \frac{r(n)}{n} \quad (\text{Eq. 4.2})$$

Where  $r(n)$  is the number of relevant documents in the top  $n$  images. Commonly, the value of  $n$  could be 5, 10, 20, 30, 50, or 100.

### 4.3.2 Recall (R)

Recall is the fraction of the documents that are relevant to the query which are successfully retrieved [50]. It could be represented as in the following equation:

$$\text{Recall} = \frac{|relevant\ documents \cap retrieved\ documents|}{|relevant\ documents|} \quad (\text{Eq. 4.3})$$

### 4.3.3 Mean Average Precision (MAP)

It is the most popular evaluation measure for the last two decades [50]. It provides a single-figure measure of quality across recall levels. We need first to calculate the Average Precision (AP) of each topic using the following equation:

$$AP = \frac{\sum_{rn=1}^N (P(rn) \times rel(rn))}{R} \quad (\text{Eq. 4.4})$$

Where  $N$  is the number of images retrieved,  $rn$  is the rank number;  $rel(rn)$  returns either 1 or 0 depending on the relevance of the image at  $rn$ ;  $P(rn)$  is the precision measured at rank  $rn$  and  $R$  is the total number of relevant images for this particular topic. Then to calculate MAP, the mean of the produced AP scores for all of the topics is taken.

## 4.4 Relevance judgments

As mentioned previously, the dataset comes with ground truth file for all the fifteen topics in *qrel* format. Since our dataset is a subset of ImageCLEF 2011 Wikipedia collection, the relevant images in the *qrel* file of each topic need to be filtered to calculate the performance of MFAR

and the other two systems. Therefore, we have prepared the relevant image set of each topic using the ground truth file.

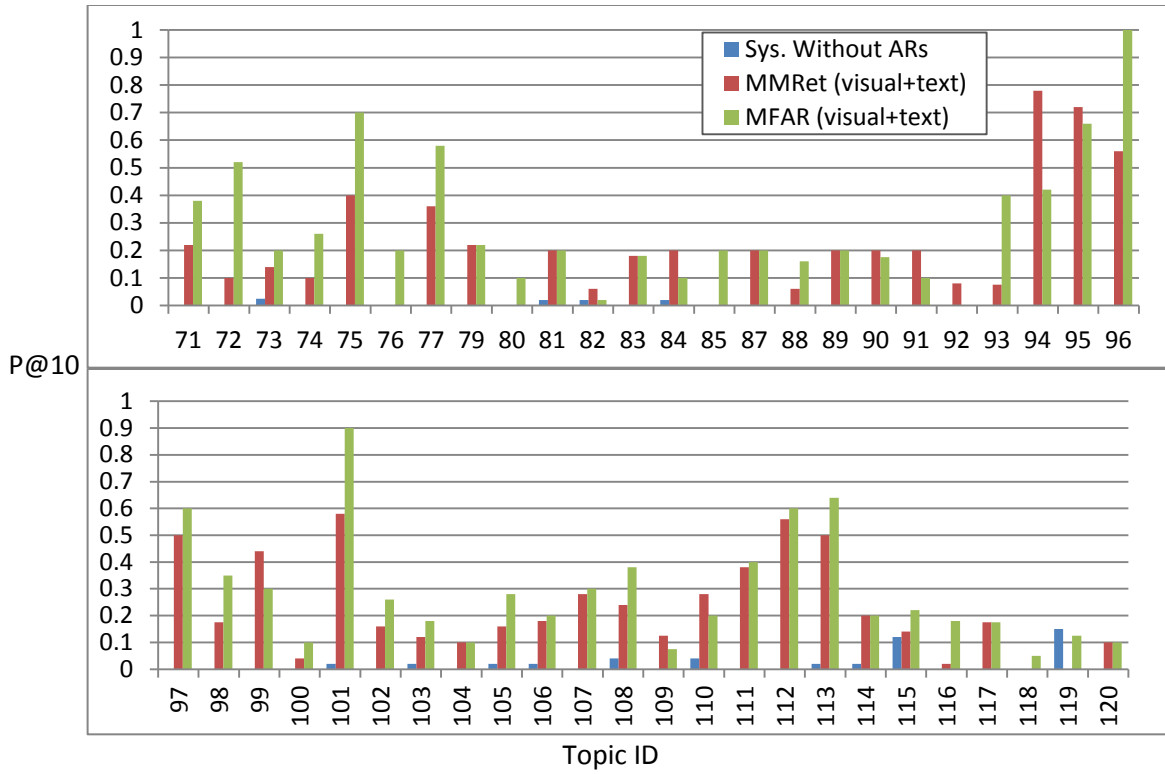
#### **4.5 Performance Results**

To evaluate the results of MFAR, we have conducted the experiment on other two systems: MMRetrival and our system without using the ARM. MMRetrieval – as mentioned in section 2.3.2 – is a multilingual and multimodal online system, it provides a flexible indexing of text and visual modalities as well as different fusion strategies (score combination and score normalization). MMRetrieval was introduced first in ImageCLEF 2010 in the Wikipedia Image Retrieval task. Then, it was developed in ImageCLEF 2011. ImageCLEF<sup>1</sup> is an initiative for evaluating cross-language and multimodal image retrieval systems in a standardized manner thus allowing comparison between the various approaches. Since MMRetrieval system supports different fusion methods, the well-known method CombSum with MinMax normalization is selected. We used the same query images and keywords in this system as in MFAR (see table 3.2). For the system without ARs, we used the visual features of MFAR and CombSum fusion method to fuse the visual scores with the same indexing technique. Nonetheless, the queries are only images. On the other hand, for MFAR and MMRetrieval, the query can be either image only or image with keyword.

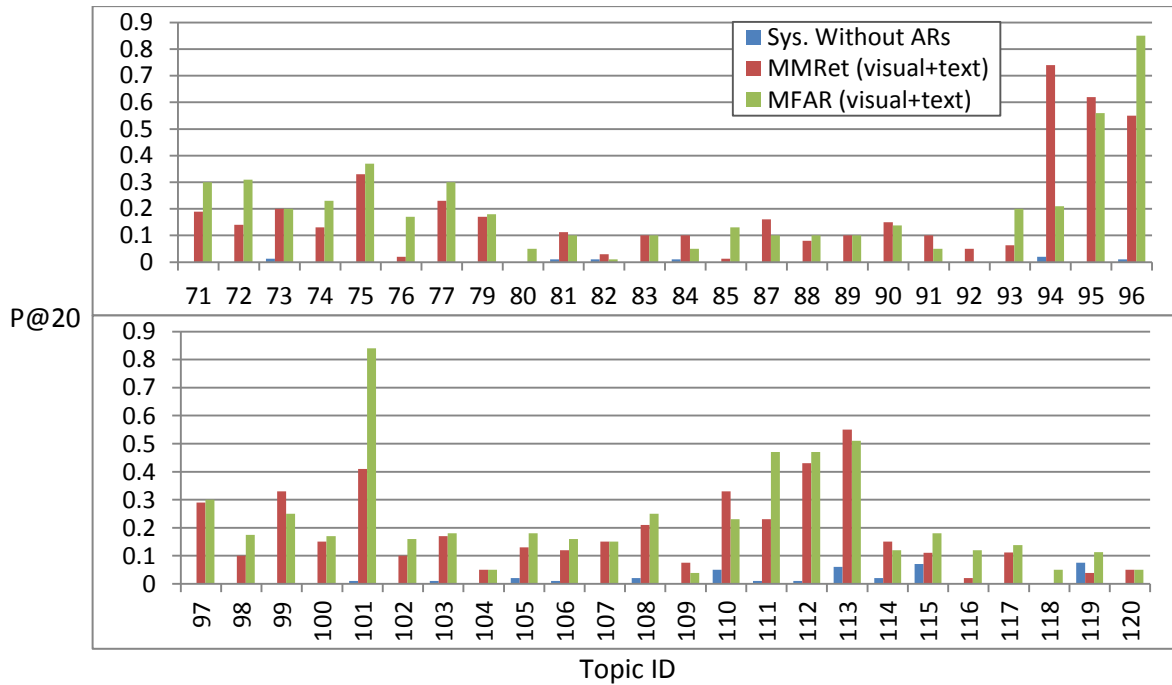
The full results of P@10, P@20, R of the retrieved results, and Average Precision (AP) of all the dataset topics of MFAR, MMRetrieval, and our system without ARs are given in the Appendix; and they are illustrated in figures 4.5, 4.6, 4.7, and 4.8 respectively. Each value in the tables of the Appendix represents the average of the performance measurements values for the five (or four) query images contained in the topic.

---

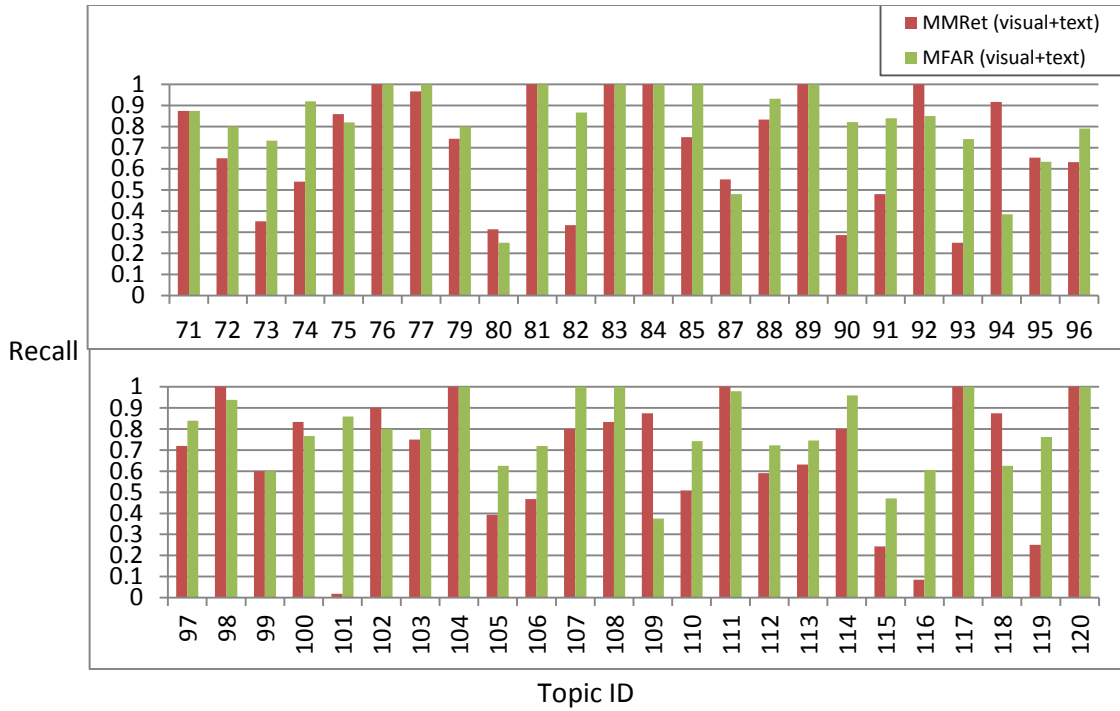
<sup>1</sup> [www.imageclef.org/](http://www.imageclef.org/)



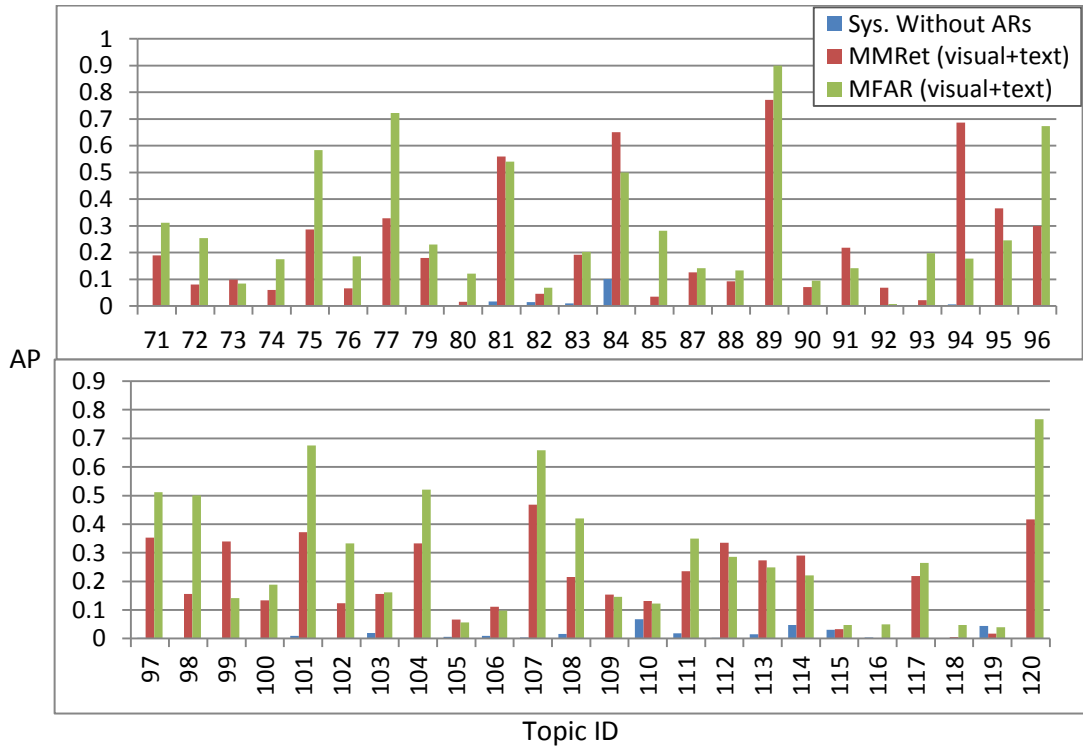
**Figure 4.5 The P@10 values of the dataset topics**



**Figure 4.6 The P@20 values of the dataset topics**



**Figure 4.7 The Recall values of the dataset topics**



**Figure 4.8 The AP values of the dataset topics**

In our experiment, the system without ARs returns all the dataset in a ranked list which means the Recall values of the retrieved results are 1 for all the queries. That is unlike MMRetrieval and MFAR since they return subset of the dataset. Thus, we have calculated the recall value for MMRetrieval and MFAR only. In addition, table 4.1 shows the overall values of P@10, P@20, Recall and MAP of the dataset for the three systems.

**Table 4.4 The overall values of P@10, P@20, MAP, and Recall of our system without ARs, MMRetrieval, and MFAR**

Sys. without ARs			MMRetrieval				MFAR			
P@10	P@20	MAP	P@10	P@20	MAP	R	P@10	P@20	MAP	R
0.008	0.005	0.006	0.243	0.168	0.242	0.691	0.294	0.212	0.288	0.802

The next chapter discusses the formal evaluation of the above results of the offline and the online phases.

## 5 Discussion and Evaluation

This chapter discusses the output of the offline phase and the retrieved results of the online phase presented and illustrated in the results chapter. Also, it shows an evaluation of MFAR system and compare its results with the other two systems.

### 5.1 The Output of the Offline Phase

As mentioned in section 4.1, most of the produced ARs showed a strong relationship between the text cluster, at the left hand side, and the visual cluster, at the right hand side. In some cases, the semantic relation between the both sides is not clear. That because the semantic topic of the text cluster is not classified properly and the cluster has various and unrelated topics. To improve the text clusters, we may need to use another semantic text clustering algorithm like hierarchical clustering. Also, we found that the visual clusters of the rules are associated with different text clusters. That is because each visual cluster consists of many topics; and here appears the importance of using the textual query besides the visual query.

In addition, all the generated ARs consist of one visual cluster in the right side. One reason of that could be the number of the used visual features. In any case, we did not offer any special consideration for the AR with multiple visual clusters in consequent side. Moreover, we have noted that the confidence value does not reflect the strength of the rule. Thus, the AR with

greater confidence values does not show all the time a stronger semantic relationship. As a result, we did not add the confidence value to the images scores.

## **5.2 Response Time**

Despite the goal of the proposed method is to improve the accuracy of the retrieved results and to provide a semantic IR, the response time is important and it should be taken into account. All the steps used in MFAR, in the online phase, to fuse the textual and the visual features lead to the increase of the response time of the query due to its additional processing. The processes of the offline phase which are: extracting the features, running the clustering algorithm, building the  $T$ , and generating the strong ARs are time consuming but do not affect the response time of the system.

In general, the response time is in an acceptable range and it could be a subject for more improvement by changing the used data structure. In fact, after retrieving the related ARs, one property of using MFAR is the ability to search in a subset of the dataset, not in the hole dataset. Thus, that minimizes the number of keyword comparisons.

## **5.3 Results Evaluation**

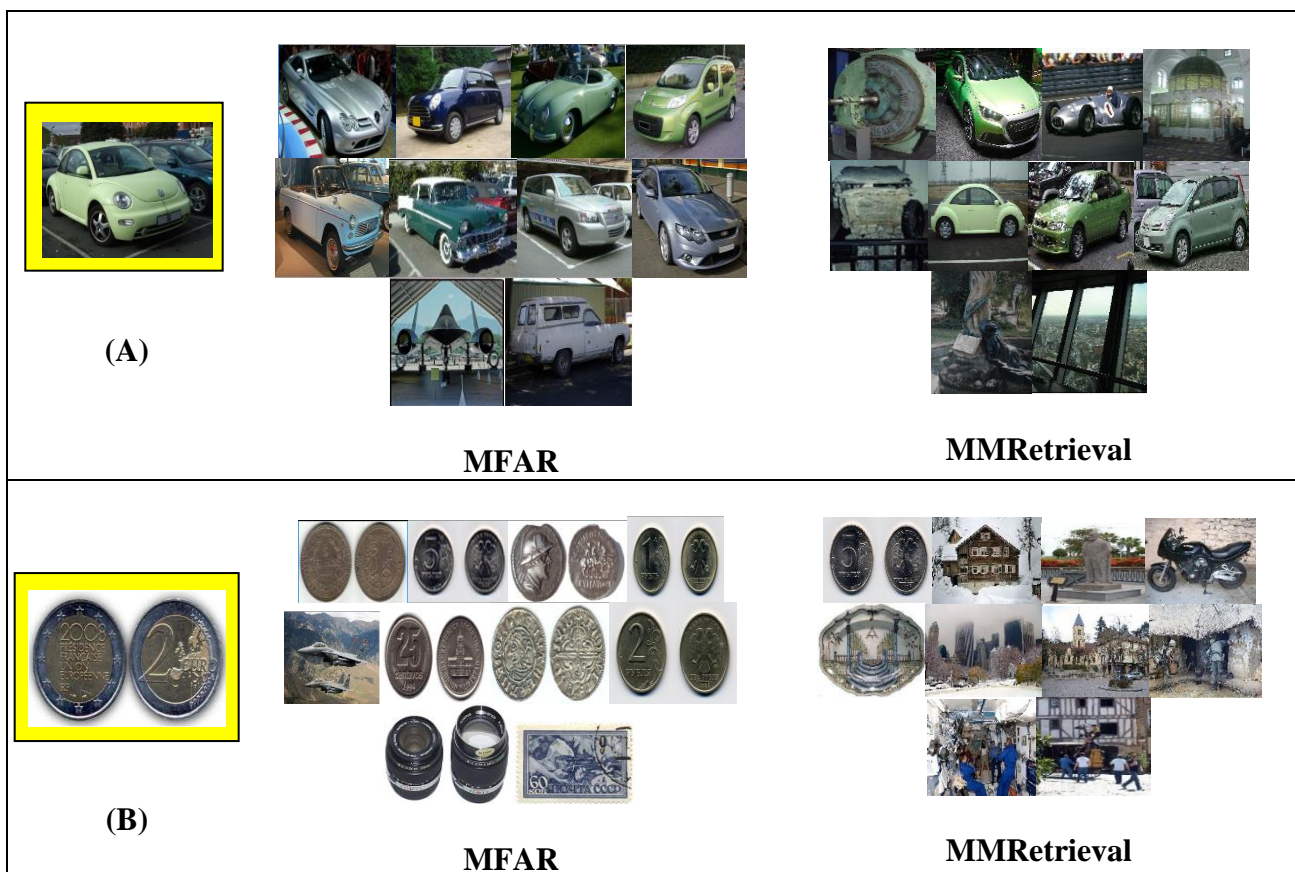
This section discusses the results of the experiments over MFAR and MMRetrieval using the two query modes: by example image query mode and composite query mode.

### **5.3.1 In Query by Example Mode**

The proposed system and MMRetrieval system have been evaluated with an image query only without using text query. The proposed system performed better semantic results than MMRetrieval system and provided better precision values than MMRetrieval. Figure 5.1 shows different examples of results obtained using both systems with query by example mode. In



MFAR, the top ten results of the two examples show semantically related images which are not dependent only on the visual features of the query image since they are from various colors and shapes. In contrast, the top ten results of MMRetrieval system with image query only clearly depend on the visual features (color) of the query only. The precision with image query mode in both systems is lesser than the systems with composite query. That because the images of the text clusters of the retrieved ARs in MFAR are ordered based on the visual features similarity which may give lesser scores to the semantically related images with different visual features. Thus, providing the text query to the system will give better scores to the semantically related text clusters and to the related images.



**Figure 5.1** The results of the highlighted image query from (A) “colored Volkswagen beetles” - topic# 71, and (B) “two euro coins” - topic# 111.

### 5.3.2 In Composite Query Mode

It is clear from section 4.5 that the multimodal runs of MFAR and MMRetrieval significantly outperform the mono-modal (visual) runs of the system without ARs. That shows obviously the importance benefits of combining multiple modalities in IR systems. Furthermore, the multimodal runs of MFAR outperform the multimodal runs of MMRetrieval and the visual runs of the system without ARs in most of the topics. In fact, as shown in table 5.1, MFAR performs the best AP values in 32 topics while MMRetrieval showed better AP in 15 topics than the other two systems. In addition, in the meaning of P@10 and P@20, MFAR outperforms the runs of the other two systems in 28 topics, MMRetrieval provides better in 10 and 14 topics respectively, and the neutral topics are nine and six topics respectively. In 26 topics, MFAR provides the better Recall values with 10 neutral topics.

Originally, the dataset topics are classified based on the AP values per topic averaged over all query runs as following:

- Easy:  $MAP > 0.3$ .
- Medium:  $0.2 < MAP \leq 0.3$ .
- Hard:  $0.1 < MAP \leq 0.2$ .
- Very hard:  $MAP < 0.1$ .

Table 5.2 presents some statistics on the topic over the classes indicating their difficulty for the multimodal runs of MMRetrieval and MFAR. From the last row of the table, the MAP values of the two systems are in an acceptable range in each difficulty class. Moreover, the MAP value of MFAR in hard topics and very hard topics exceeds the ranges and give higher scores than the range limit by 0.226 and 0.116 respectively. In general, the MAP scores of MFAR are better than the other system in all difficulty classes. Also, we can realize from the table that the P@10 and

P@20 values of MFAR in all the difficulty classes outperform the performance of MMRetrieval. One interesting note from the table is the precision value of the medium topics which is less than the precision of the hard topics.

**5.1 The number of the best performing topics for each performance measurements of the dataset.**

	<b>Sys. Without ARs</b>	<b>MMRetrieval</b>	<b>MFAR</b>
<b>No# of topics with best P@10</b>	1	10	28
<b>No# of topics with best P@20</b>	0	14	28
<b>No# of topics with best AP</b>	1	15	32
<b>No# of topics with best Recall</b>	—	12	26

**5.2 The average values of the performance measurements of all the topics difficulty for Sys.1 (MMRetrieval) and Sys.2 (MFAR).**

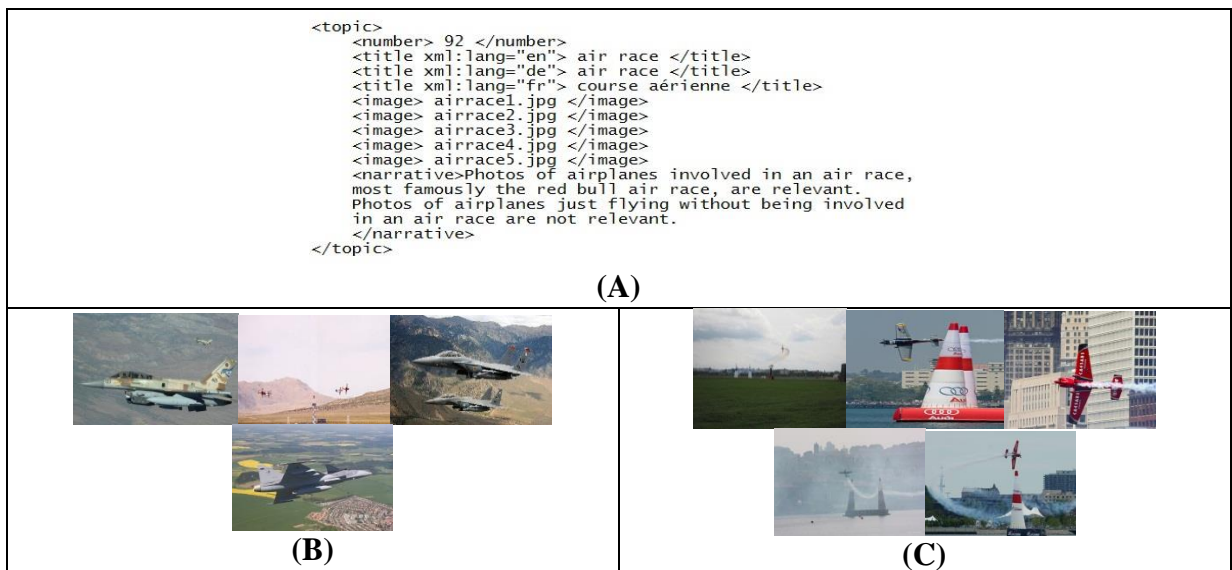
	<b>Easy topics</b>		<b>Medium topics</b>		<b>Hard topics</b>		<b>Very Hard topics</b>	
	<b>Sys. 1</b>	<b>Sys. 2</b>	<b>Sys. 1</b>	<b>Sys. 2</b>	<b>Sys. 1</b>	<b>Sys. 2</b>	<b>Sys. 1</b>	<b>Sys. 2</b>
<b>P@10</b>	0.286	0.399	0.150	0.155	0.270	0.363	0.123	0.161
<b>P@20</b>	0.197	0.231	0.129	0.179	0.234	0.215	0.107	0.174
<b>MAP</b>	0.331	0.441	0.201	0.252	0.176	0.226	0.067	0.116

**5.4 Discussion**

The multimodal runs show significant improvement in the results' performance of MFAR and MMRetrieval. That supports the results of the previous researches which indicate the improvement of the retrieval results with the multimodal systems than the mono-modal systems [7, 9, 11].

MMRetrieval is a powerful system that showed success in ImageCLEF 2011. To make a fair evaluation between MFAR and MMRetrieval, we needed to use the same image and text query.

With the images queries provided by the dataset and the text queries shown in table 3.2, MFAR satisfied the best performance results. Although of the progress in MFAR results, it satisfies a zero P@10 value for topic number 92. In the other side, MMRetrieval, performed zero P@10 in five topics (see appendix). To investigate the reason of this low performance of MFAR in topic 92, we have studied this topic from three sides (1) the description of the topic, (2) the topic’s relevant images in the dataset, and (3) the query images which are shown in figure 5.2. The topic is about the “air race” and the query images show that clearly, but the relevant images from the ground truth file of the dataset are not about air racing. The relevant images mostly show “military aircraft” which does not match the topic of “air racing”.



**Figure 5.2 Topic 92 “air race” (A) topic description, (B) relevant images of the dataset, and (C) query images.**

The next chapter presents the final conclusion of the proposed method, the possible improvements, and the future works.

## 6 Conclusion and Future Work

### 6.1 Conclusion

This thesis presented a new multimodal fusion method for IR to address the semantic gap problem which is the main goal of this study. Multimodal fusion method in IR was used in four different fusion levels: early, late, trans-media and re-ranking level. Late fusion method is the most widely used method since it allows for each modality to use the most suitable methods for analyzing and classifying each modality which provides much more flexibility. The proposed late fusion method MFAR uses ARM algorithm in IR system for the Web images to construct semantic relations between image clusters based on the visual features and the images clusters based on textual features for the same dataset. The hypothesis in constructing the transaction database  $T$  and thus the strong ARs is that the similar clusters (textual and visual) that satisfied the predefined *minsupp* and *minconf* values seem to be semantically related.

The results in chapter 4 show the correctness of the hypothesis in most of the rules. After constructing the ARs in the offline phase, the retrieval process should be started with example image query in the online phase. The method gives the ability to retrieve images that are semantically related by using the extracted visual features of the query image and by exploring the related ARs from the mining. It is possible to use a keyword query to support the results. The

results show that the precision, recall, and the MAP values of MFAR system are better than MMRetrieval system and the system without ARs.

Despite MFAR with image query only mode performed better precision values than the other two systems, the results are low comparing to the multimodal runs. We found that supplying the IR with composite query provides the system by more evidences to increase the scores of the related images.

We think that the ability to make semantic search using image query is one of the main strength points of MFAR. In addition, the ability to enclose the search in a subset of the dataset which is most probably semantically related to the query is another point. In the other hand, after studying the generated ARs, we have found a concern that needed to be addressed in the calculation of the support/confidence value (Eq. 3.4 and 3.5). A few of text clusters with very small result set (small denominator of the equations) possibly get a high support and confidence. This drawback could be relieved by defining a minimum count threshold *min\_count* to filter out these text clusters in the later rules mining process.

## **6.2 Scope of Future Work**

The future work can be classified into two categories: improving the processes of MRAF system, and generalizing the proposed method to be used in other applications. The following points could improve the system performance:

- Try another text clustering algorithm to improve the semantic meaning for each cluster.
- Add more visual descriptors to capture more visual features.
- Use of WordNet's information about different senses of a word. WordNet contains one or more senses for a word. For each sense there exists information about conceptual

relations (like synonymy, hyponyms, ... etc.) which helps to improve text query processing.

- Demonstrate the use of minimum count value for text clusters in Eq. 3.4 and 3.5, as stated above.
- Conduct the experiment over the full size of the dataset which is not fully described (not all the images have a related text) to study the generated ARs and if the un-described images are associated with the appropriate text cluster or not.
- Allow the ability to provide the system by multiple query images at the same time and fuse the results of each query to generate one final list.
- Improve the system with image query mode without keyword query. The use of pseudo-relevance feedback technique is one of the suggested solutions. The correlated terms of the top retrieved ARs could be used to make a feedback text query.

On the other hand, we think that the proposed method could be used in image annotator system. After clustering the unannotated images, we can use the same modified version of the ARM algorithm, used in this thesis, to associate the image clusters with the text clusters of the annotated images.

## References

- [1] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, pp. 1-60, 2008.
- [2] A. F. Smeaton and I. Quigley, "Experiments on using semantic distances between words in image caption retrieval," presented at the Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, Zurich, Switzerland, 1996.
- [3] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, *et al.*, "Query by Image and Video Content: The QBIC System," *Computer*, vol. 28, pp. 23-32, 1995.
- [4] J. R. Smith and S.-F. Chang, "VisualSEEK: a fully automated content-based image query system," presented at the Proceedings of the fourth ACM international conference on Multimedia, Boston, Massachusetts, United States, 1996.
- [5] C. Yu, "Introduction," in *High-Dimensional Indexing*. vol. 2341, C. Yu, Ed., ed: Springer Berlin Heidelberg, 2003, pp. 1-8.
- [6] L. Bittner, "R. Bellman, Adaptive Control Processes. A Guided Tour. Princeton University Press.," *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 42, pp. 364-365, 1962.
- [7] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recogn.*, vol. 40, pp. 262-282, 2007.
- [8] P. Atrey, M. Hossain, A. El Saddik, and M. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimedia Systems*, vol. 16, pp. 345-379, 2010.
- [9] S. Wu and S. McClean, "Performance prediction of data fusion for information retrieval," *Information Processing & Management*, vol. 42, pp. 899-915, 2006.
- [10] H. Boström, S. F. Andler, M. Brohede, R. Johansson, A. Karlsson, J. van Laere, *et al.*, "On the Definition of Information Fusion as a Field of Research," University of Skövde, School of Humanities and Informatics, Skövde2007.
- [11] A. Depeursinge and H. Müller, "Fusion Techniques for Combining Textual and Visual Information Retrieval ImageCLEF." vol. 32, H. Müller, P. Clough, T. Deselaers, and B. Caputo, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 95-114.
- [12] H. Lee, B. Lee, K. Park, and R. Elmasri, "Fusion Techniques for Reliable Information: A Survey," *JDCTA*, vol. 4, pp. 74-88, 2010.
- [13] C. Sanderson and K. K. Paliwal, "Information Fusion and Person Verification Using Speech & Face Information," IDIAP2002.
- [14] S. Clinchant, J. Ah-Pine, and G. Csurka, "Semantic combination of textual and visual information in multimedia retrieval," presented at the Proceedings of the 1st ACM International Conference on Multimedia Retrieval, Trento, Italy, 2011.
- [15] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders, "Early versus late fusion in semantic video analysis," presented at the Proceedings of the 13th annual ACM international conference on Multimedia, Hilton, Singapore, 2005.



- [16] M. Ferecatu and H. Sahbi, "TELECOMParisTech at ImageClefphoto 2008: Bi-modal text and image retrieval with diversity enhancement," presented at the Working Notes for the CLEF 2008 workshop, 2008.
- [17] T. Deselaers, T. Weyand, and H. Ney, "Image Retrieval and Annotation Using Maximum Entropy," in *Evaluation of Multilingual and Multi-modal Information Retrieval*. vol. 4730, C. Peters, P. Clough, F. Gey, J. Karlgren, B. Magnini, D. Oard, *et al.*, Eds., ed: Springer Berlin Heidelberg, 2007, pp. 725-734.
- [18] T. Gass, T. Weyand, T. Deselaers, and H. Ney, "FIRE in ImageCLEF 2007: Support Vector Machines and Logistic Models to Fuse Image Descriptors for Photo Retrieval," in *Advances in Multilingual and Multimodal Information Retrieval*. vol. 5152, C. Peters, V. Jijkoun, T. Mandl, H. Müller, D. Oard, A. Peñas, *et al.*, Eds., ed: Springer Berlin Heidelberg, 2008, pp. 492-499.
- [19] L. Yen-Yu and F. Chiou-Shann, "Multimodal kernel learning for image retrieval," in *System Science and Engineering (ICSSE), 2010 International Conference on*, 2010, pp. 155-160.
- [20] C. Lau, D. Tjondronegoro, J. Zhang, S. Geva, and Y. Liu, "Fusing Visual and Textual Retrieval Techniques to Effectively Search Large Collections of Wikipedia Images Comparative Evaluation of XML Information Retrieval Systems." vol. 4518, N. Fuhr, M. Lalmas, and A. Trotman, Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 345-357.
- [21] H. Frigui, J. Caudill, and A. C. Ben Abdallah, "Fusion of multi-modal features for efficient content-based image retrieval," in *Fuzzy Systems, 2008. FUZZ-IEEE 2008. (IEEE World Congress on Computational Intelligence). IEEE International Conference on*, 2008, pp. 1992-1998.
- [22] I. Bartolini and P. Ciaccia, "Scenique: a multimodal image retrieval interface," presented at the Proceedings of the working conference on Advanced visual interfaces, Napoli, Italy, 2008.
- [23] Z. Xin, A. Depeursinge, and H. Muller, "Information Fusion for Combining Visual and Textual Image Retrieval," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, 2010, pp. 1590-1593.
- [24] G. Zhen, Z. Zhongfei, E. P. Xing, and C. Faloutsos, "A Max Margin Framework on Image Annotation and Multimodal Image Retrieval," in *Multimedia and Expo, 2007 IEEE International Conference on*, 2007, pp. 504-507.
- [25] H. J. Escalante, C. A. H. rnadez, L. E. Sucar, and M. Montes, "Late fusion of heterogeneous methods for multimedia image retrieval," presented at the Proceedings of the 1st ACM international conference on Multimedia information retrieval, Vancouver, British Columbia, Canada, 2008.
- [26] K. Zagoris, A. Arampatzis, and S. A. Chatzichristofis, "[www.MMRetrieval.net](http://www.MMRetrieval.net): a multimodal search engine," presented at the Proceedings of the Third International Conference on SIMilarity Search and APplications, Istanbul, Turkey, 2010.
- [27] L. Kaliciak, D. Song, N. Wiratunga, and J. Pan, "Combining Visual and Textual Systems within the Context of User Feedback," in *Advances in Multimedia Modeling*. vol. 7732, S. Li, A. Saddik, M. Wang, T. Mei, N. Sebe, S. Yan, *et al.*, Eds., ed: Springer Berlin Heidelberg, 2013, pp. 445-455.
- [28] R. Besançon, P. Hède, P.-A. Moellic, and C. Fluhr, "Cross-Media Feedback Strategies: Merging Text and Image Information to Improve Image Retrieval Multilingual Information Access for Text, Speech and Images." vol. 3491, C. Peters, P. Clough, J.

- Gonzalo, G. Jones, M. Kluck, and B. Magnini, Eds., ed: Springer Berlin / Heidelberg, 2005, pp. 919-919.
- [29] W. Shikui, Z. Yao, Z. Zhenfeng, and L. Nan, "Multimodal Fusion for Video Search Reranking," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, pp. 1191-1199, 2010.
- [30] R. He, N. Xiong, L. T. Yang, and J. H. Park, "Using Multi-Modal Semantic Association Rules to fuse keywords and visual features automatically for Web image retrieval," *Inf. Fusion*, vol. 12, pp. 223-230, 2011.
- [31] U. Fayyad, U. Fayyad, G. Piatetsky-shapiro, G. Piatetsky-shapiro, P. Smyth, and P. Smyth, "Knowledge Discovery and Data Mining: Towards a Unifying Framework," pp. 82-88, 1996.
- [32] R. Agrawal, T. Imieli, #324, ski, and A. Swami, "Mining association rules between sets of items in large databases," *SIGMOD Rec.*, vol. 22, pp. 207-216, 1993.
- [33] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," presented at the Proceedings of the 20th International Conference on Very Large Data Bases, 1994.
- [34] A. Kouomou-Choupo, L. Berti-Equille, and A. Morin, "Multimedia indexing and retrieval with features association rules mining," Taipei, 2004.
- [35] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, pp. 1349-1380, 2000.
- [36] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7: multimedia content description interface*: Wiley, 2002.
- [37] C. D. Manning, P. Raghavan, and H. Schtze, *Introduction to Information Retrieval*: Cambridge University Press, 2008.
- [38] A. Guttman, "R-trees: a dynamic index structure for spatial searching," presented at the Proceedings of the 1984 ACM SIGMOD international conference on Management of data, Boston, Massachusetts, 1984.
- [39] D. A. White and R. Jain, "Similarity Indexing with the SS-tree," presented at the Proceedings of the Twelfth International Conference on Data Engineering, 1996.
- [40] N. Katayama and S. i. Satoh, "The SR-tree: an index structure for high-dimensional nearest neighbor queries," *SIGMOD Rec.*, vol. 26, pp. 369-380, 1997.
- [41] S. Berchtold, D. A. Keim, and H.-P. Kriegel, "The X-tree: An Index Structure for High-Dimensional Data," presented at the Proceedings of the 22th International Conference on Very Large Data Bases, 1996.
- [42] J. T. Robinson, "The K-D-B-tree: a search structure for large multidimensional dynamic indexes," presented at the Proceedings of the 1981 ACM SIGMOD international conference on Management of data, Ann Arbor, Michigan, 1981.
- [43] M. Taileb, S. Lamrous, and S. Touati, "Non Overlapping Hierarchical Index Structure," *Proceedings of World Academy of Science: Engineering & Technolog*, vol. 39, 2008.
- [44] J. B. MacQueen, "Some Methods for Classification and Analysis of MultiVariate Observations," presented at the Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability, 1967.
- [45] S. Wu, "Score Normalization," in *Data Fusion in Information Retrieval*. vol. 13, ed: Springer Berlin Heidelberg, 2012, pp. 19-42.

- [46] S. Wu, "Observations and Analyses," in *Data Fusion in Information Retrieval*. vol. 13, ed: Springer Berlin Heidelberg, 2012, pp. 43-71.
- [47] T. Tsirikas, A. Popescu, and J. Kludas, "Overview of the Wikipedia Image Retrieval task at ImageCLEF 2011," ed. Amsterdam: In the Working Notes for the CLEF 2011 Labs and Workshop, 2011.
- [48] Bas, x, M. tan, x, H. am, Gu, *et al.*, "Bilvideo-7: an MPEG-7- compatible video indexing and retrieval system," *MultiMedia, IEEE*, vol. 17, pp. 62-73, 2010.
- [49] M. Taileb and S. Touati, "NOHIS-Tree: High-Dimensional Index Structure for Similarity Search," *World Academy of Science, Engineering and Technology*, vol. 59, pp. 351 - 358, 2011.
- [50] M. Sanderson, "Performance Measures Used in Image Information Retrieval," in *ImageCLEF*. vol. 32, H. Müller, P. Clough, T. Deselaers, and B. Caputo, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 81-94.
- [51] G. Salton, "The state of retrieval system evaluation," *Inf. Process. Manage.*, vol. 28, pp. 441-449, 1992.

## Appendix

### A. The Results of the Experiment

#### 1. The P@10 values of the dataset topics

Topic ID	Sys. Without ARs	MMRetrieval (visual+text)	MFAR (visual+text)
71	0	0.22	0.38
72	0	0.1	0.52
73	0.025	0.14	0.2
74	0	0.1	0.26
75	0	0.4	0.7
76	0	0	0.2
77	0	0.36	0.58
79	0	0.22	0.22
80	0	0	0.1
81	0.02	0.2	0.2
82	0.02	0.06	0.02
83	0	0.18	0.18
84	0.02	0.2	0.1
85	0	0	0.2
87	0	0.2	0.2
88	0	0.06	0.16
89	0	0.2	0.2
90	0	0.2	0.175
91	0	0.2	0.1
92	0	0.08	0
93	0	0.075	0.4
94	0	0.78	0.42
95	0	0.72	0.66
96	0	0.56	1
97	0	0.5	0.6
98	0	0.175	0.35
99	0	0.44	0.3
100	0	0.04	0.1

<b>Topic ID</b>	<b>Sys. Without ARs</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>101</b>	0.02	0.58	0.9
<b>102</b>	0	0.16	0.26
<b>103</b>	0.02	0.12	0.18
<b>104</b>	0	0.1	0.1
<b>105</b>	0.02	0.16	0.28
<b>106</b>	0.02	0.18	0.2
<b>107</b>	0	0.28	0.3
<b>108</b>	0.04	0.24	0.38
<b>109</b>	0	0.125	0.075
<b>110</b>	0.04	0.28	0.2
<b>111</b>	0	0.38	0.4
<b>112</b>	0	0.56	0.6
<b>113</b>	0.02	0.5	0.64
<b>114</b>	0.02	0.2	0.2
<b>115</b>	0.12	0.14	0.22
<b>116</b>	0	0.02	0.18
<b>117</b>	0	0.175	0.175
<b>118</b>	0	0	0.05
<b>119</b>	0.15	0	0.125
<b>120</b>	0	0.1	0.1

**2. The P@20 values of the dataset topics**

<b>Topic ID</b>	<b>Sys. Without ARs</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>71</b>	0	0.19	0.3
<b>72</b>	0	0.14	0.31
<b>73</b>	0.013	0.2	0.2
<b>74</b>	0	0.13	0.23
<b>75</b>	0	0.33	0.37
<b>76</b>	0	0.02	0.17
<b>77</b>	0	0.23	0.3
<b>79</b>	0	0.17	0.18
<b>80</b>	0	0	0.05
<b>81</b>	0.01	0.113	0.1
<b>82</b>	0.01	0.03	0.01

<b>Topic ID</b>	<b>Sys. Without ARs</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>83</b>	0	0.1	0.1
<b>84</b>	0.01	0.1	0.05
<b>85</b>	0	0.013	0.13
<b>87</b>	0	0.16	0.1
<b>88</b>	0	0.08	0.1
<b>89</b>	0	0.1	0.1
<b>90</b>	0	0.15	0.138
<b>91</b>	0	0.1	0.05
<b>92</b>	0	0.05	0
<b>93</b>	0	0.063	0.2
<b>94</b>	0.02	0.74	0.21
<b>95</b>	0	0.62	0.56
<b>96</b>	0.01	0.55	0.85
<b>97</b>	0	0.29	0.3
<b>98</b>	0	0.1	0.175
<b>99</b>	0	0.33	0.25
<b>100</b>	0	0.15	0.17
<b>101</b>	0.01	0.41	0.84
<b>102</b>	0	0.1	0.16
<b>103</b>	0.01	0.17	0.18
<b>104</b>	0	0.05	0.05
<b>105</b>	0.02	0.13	0.18
<b>106</b>	0.01	0.12	0.16
<b>107</b>	0	0.15	0.15
<b>108</b>	0.02	0.21	0.25
<b>109</b>	0	0.075	0.038
<b>110</b>	0.05	0.33	0.23
<b>111</b>	0.01	0.23	0.47
<b>112</b>	0.01	0.43	0.47
<b>113</b>	0.06	0.55	0.51
<b>114</b>	0.02	0.15	0.12
<b>115</b>	0.07	0.11	0.18
<b>116</b>	0	0.02	0.12
<b>117</b>	0	0.112	0.138
<b>118</b>	0	0	0.05
<b>119</b>	0.075	0.038	0.113

<b>Topic ID</b>	<b>Sys. Without ARs</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>120</b>	0	0.05	0.05

### 3. The AP values of the dataset topics

<b>Topic ID</b>	<b>Sys. Without ARs</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>71</b>	0.0012	0.19	0.311
<b>72</b>	0.001	0.08	0.254
<b>73</b>	0.003	0.098	0.084
<b>74</b>	0.001	0.06	0.175
<b>75</b>	0.001	0.287	0.583
<b>76</b>	0.0001	0.066	0.186
<b>77</b>	0.0004	0.328	0.723
<b>79</b>	0.001	0.18	0.23
<b>80</b>	0.0012	0.016	0.121
<b>81</b>	0.017	0.56	0.54
<b>82</b>	0.014	0.046	0.069
<b>83</b>	0.01	0.192	0.203
<b>84</b>	0.101	0.65	0.5
<b>85</b>	0.001	0.035	0.282
<b>87</b>	0.001	0.126	0.142
<b>88</b>	0.0014	0.092	0.133
<b>89</b>	0.0001	0.771	0.9
<b>90</b>	0.0012	0.071	0.095
<b>91</b>	0.001	0.218	0.141
<b>92</b>	0.0002	0.068	0.007
<b>93</b>	0.001	0.022	0.198
<b>94</b>	0.006	0.686	0.178
<b>95</b>	0.004	0.366	0.246
<b>96</b>	0.002	0.3	0.673
<b>97</b>	0.003	0.353	0.512
<b>98</b>	0.0001	0.156	0.501
<b>99</b>	0.001	0.339	0.141
<b>100</b>	0.001	0.134	0.189
<b>101</b>	0.01	0.372	0.675
<b>102</b>	0.0002	0.124	0.333

<b>Topic ID</b>	<b>Sys. Without ARs</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>103</b>	0.02	0.156	0.162
<b>104</b>	0.0003	0.333	0.521
<b>105</b>	0.006	0.067	0.056
<b>106</b>	0.01	0.111	0.098
<b>107</b>	0.004	0.468	0.658
<b>108</b>	0.016	0.215	0.420
<b>109</b>	0.0001	0.154	0.146
<b>110</b>	0.068	0.131	0.122
<b>111</b>	0.018	0.236	0.35
<b>112</b>	0.003	0.335	0.286
<b>113</b>	0.015	0.274	0.249
<b>114</b>	0.047	0.29	0.221
<b>115</b>	0.031	0.033	0.047
<b>116</b>	0.004	0.002	0.05
<b>117</b>	0.002	0.219	0.265
<b>118</b>	0.0001	0.005	0.048
<b>119</b>	0.044	0.017	0.04
<b>120</b>	0.0004	0.417	0.767

#### 4. The Recall values of the dataset topics

<b>Topic ID</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>71</b>	0.875	0.875
<b>72</b>	0.65	0.8
<b>73</b>	0.352	0.733
<b>74</b>	0.54	0.92
<b>75</b>	0.86	0.82
<b>76</b>	1	1
<b>77</b>	0.967	1
<b>79</b>	0.743	0.8
<b>80</b>	0.313	0.25
<b>81</b>	1	1
<b>82</b>	0.333	0.867
<b>83</b>	1	1
<b>84</b>	1	1



<b>Topic ID</b>	<b>MMRetrieval (visual+text)</b>	<b>MFAR (visual+text)</b>
<b>85</b>	0.75	1
<b>87</b>	0.55	0.48
<b>88</b>	0.833	0.933
<b>89</b>	1	1
<b>90</b>	0.286	0.822
<b>91</b>	0.48	0.84
<b>92</b>	1	0.85
<b>93</b>	0.25	0.741
<b>94</b>	0.917	0.385
<b>95</b>	0.654	0.633
<b>96</b>	0.632	0.792
<b>97</b>	0.72	0.84
<b>98</b>	1	0.938
<b>99</b>	0.6	0.6
<b>100</b>	0.833	0.767
<b>101</b>	0.018	0.86
<b>102</b>	0.9	0.8
<b>103</b>	0.75	0.8
<b>104</b>	1	1
<b>105</b>	0.393	0.625
<b>106</b>	0.467	0.72
<b>107</b>	0.8	1
<b>108</b>	0.833	1
<b>109</b>	0.875	0.375
<b>110</b>	0.509	0.742
<b>111</b>	1	0.98
<b>112</b>	0.591	0.722
<b>113</b>	0.632	0.746
<b>114</b>	0.8	0.96
<b>115</b>	0.243	0.47
<b>116</b>	0.085	0.606
<b>117</b>	1	1
<b>118</b>	0.875	0.625
<b>119</b>	0.25	0.762
<b>120</b>	1	1

**B. The Accepted paper in the 16<sup>th</sup> International Conference on Human-Computer Interaction in Greece. Jun, 2014.**

## **Towards Semantic Image Retrieval Using Multimodal Fusion with Association Rules Mining**

**Abstract.** This paper proposes a semantic retrieving method for an image retrieval system that employs the fusion of the textual and visual information of the image dataset which is a recent trend in image retrieval researches. It combines two different data mining techniques to retrieve semantically related images: clustering and association rule mining algorithm. At the offline phase of the method, the association rules are discovered between the text semantic clusters and the visual clusters to use it later in the online phase. To evaluate the proposed system, the experiment was conducted on more than 54,500 images of ImageCLEF 2011 Wikipedia collection. The proposed retrieval system was compared to an online system called MMRetrieval and to the proposed system but without using association rules. The obtained results show that our proposed method achieved the best precision and mean average precision.

**Keywords:** Image Retrieval, Multimodal Fusion, Association Rules Mining, Clustering.

### **1 Introduction**

Today, a huge amount of images exists in electronic formats on the Web and in different information repositories; and their size is exponentially growing day after another. Thus, we need for an efficient Image Retrieval system (IR) to get access to these images. IR could rely purely on textual metadata which may produce a lot of garbage in the results since users usually enter that metadata manually which is inefficient, expensive and may not capture every keyword that describes the image. On the other hand, the Content-Based Image Retrieval (CBIR) could be used to filter images based on their visual contents such as colors, shapes, textures or any other information that can be derived from the image itself which may provide better indexing and return more accurate results. At the same time, these visual features contents extracted by the computer may be different from the image contents that people understand. It requires the translation of high-level user views into low-level image features and this is the so-called “semantic gap” problem. This problem is the reason behind why the current CBIR systems are difficult to be widely used for retrieving Web images. A lot of efforts have been made to bridge this gap by

using different techniques. In [1], the authors identified the major categories of the state-of-the-art techniques in narrowing down the ‘semantic gap’ one of them is to fuse the evidences from the text and the visual content of the images. Fusion in IR is considered as a novel area, with very little achievements in the early days of research [2]. Truly, we live in a multimodal world, and there is no reason why advantage should not be taken of all available media to build a useful semantic IR system. This paper tries to narrow down this gap and enhance the retrieval performance by fusing the two basic modalities: text and visual features. To determine the appropriate fusion method, it is important to answer the following basic questions: what is the suitable level to implement the fusion strategy? And how to fuse the multimodal information?

The proposed method is a Multimodal Fusion method based on Association Rules mining (MFAR). It is considered as a late fusion. This method combines two different data mining techniques: clustering and Association Rules Mining (ARM) algorithm. It uses ARM to explore the relations between text semantic clusters and image visual features clusters by applying *Apriori* algorithm. The method consists of two main phases: offline and online. The offline phase identifies the relations among the clusters from different modalities to construct the semantic Association Rules (ARs). On the other hand, the online phase is the retrieval phase. It uses the generated ARM to retrieve the related images to the query.

The rest of the paper is categorized as following. The next section will review the current information fusion approaches and how they fused different modalities. Section three gives the required background about ARM algorithm. Then section four describes the proposed method in detail. The experiment and the conclusion are presented at sections five and six respectively.

## **2 Related Work**

Information retrieval community found the power of fusing various information sources on the retrieving performance [3]. Information fusion has the potential of improving retrieval performance by relying on the assumption that the heterogeneity of multiple information sources allows cross-correction of some of the errors, leading to better results [4]. In literature, the fusion of the visual and the textual features was performed in different levels of the retrieval process which are early fusion, late fusion, trans-media fusion and at re-ranking level.

### **2.1 Early Fusion**

This method first extracts the low level features of the modalities using the suitable feature extractor. Then, the extracted vectors are concatenated into one vector to form one unique feature space. The advantage of this strategy is that it enables a true multimedia

representation for all the fused modalities where one decision rule is applied on all information sources. Early fusion could be used without feature weighting such in [5]; they concatenated the normalized feature spaces of the visual and the textual features. On the other hand, feature weighting was used in different works in order to provide more weight for specific features. In [6] and [7] as part of ImageCLEF 2006 and 2007 respectively, they presented a novel approach to weight features using support vector machines. The main drawback of early fusion is the dimensionality of the resulting feature space which is equal to the sum of all the fused subspaces which leads to the well-known problem the “curse of dimensionality” [8]. Also, increasing in the number of modalities and the high dimensionality make them difficult to learn the cross-correlation among the heterogeneous features [9].

## **2.2 Late Fusion**

Late fusion (or decision level) strategies do not act at the level of one representation for all the modalities features but rather at the level of the similarities among each modality. The extracted features of each modality are classified using the appropriate classifier; then, each classifier provides a decision. Unlike early fusion, where the features of each modality may have different representation, the decisions usually have the same representation. As a result, the fusion of the decisions becomes easier. The main disadvantage of this strategy is that it fails to utilize the feature level correlation among modalities. Also, using different classifiers and different learning process is expensive in term of time and learning for each modality.

Late fusion is used widely in image retrieval systems, and there is a diversity in the proposed methods. The most widely used technique is a rule-based method [10-16]. In [16], web application called MMRetrieval is proposed which has an online graphical user interface that brings image and text search together to compose a multimodal and multilingual query. The modalities are searched in parallel, and then the results can be fused via several selectable methods. Fusion process consists of two components: score normalization and combination. It provides a combination of scores across modalities with summation, multiplication, and maximum.

## **2.3 Trans-media Fusion**

In this method, the main idea is to use first one of the modalities (say image) to gather relevant documents (nearest neighbors from a visual point of view) and then to use the dual modalities (text representations of the visually nearest neighbors) to perform the final retrieval. Most proposed methods under this category are based on adopted relevance feedback or pseudo-relevance feedback techniques as in [17]. The authors in [17] used the pseudo-relevance feedback to gather the  $N$  most relevant documents from the dataset using some visual similarity measures with respect to the visual features of the query or,

reciprocally, using a purely textual similarity with respect to the textual features of the query, then aggregate these mono-modal similarities.

## 2.4 Image Re-ranking

In image re-ranking level, we need first to perform the search based on the text query. Then, the returned list of images is reordered according to the visual features similarity. In [18], the cross-reference re-ranking strategy is proposed for the refinement of the initial search results of text-based video search engines. While [18] method deals with clusters of the modalities, [19] proposed a method that construct a semantic relation between text (words) and visual clusters using the ARM algorithm. They proposed Multi-Modal Semantic Association Rules (MMSAR) algorithm to fuse key-words and visual features automatically for Web image retrieval.

MFAR in this paper is considered as a late fusion method. There are three main differences between the method of [19] and MFAR proposed method: (1) MFAR uses ARM algorithm to explore the relations between text semantic clusters and image visual feature clusters; (2) the fusion method in MFAR is used at the retrieval phase not for re-ranking the results; (3) it is possible in MFAR to make a query by example image. In literature, there are several attempts to couple image retrieval and association rules mining algorithm. First, it is used as a preprocessing strategy for a preliminary reduction of the dimensionality of the pattern space to improve the global search time for CBIR system as in [20]. Second, as mentioned earlier, ARM has been used in image re-ranking process [19].

The next section will present the required background about ARM algorithm, which helps to understand the proposed method.

## 3 Basics of Association Rules Mining Algorithm

ARM is a data mining technique useful for discovering interesting relationships hidden in large databases. The classical example is the rules extracted from the content of the market baskets. Items are things we can buy in a market, and transactions are market baskets containing several items. The collection of all transactions called the transactions database. Besides the market basket data, association rules mining are applicable for different applications of other domains such as bioinformatics, medical diagnosis and Web mining.

The problem of mining association rules is stated as following:  $I=\{i_1, i_2, \dots, i_m\}$  is a set of items,  $T=\{t_1, t_2, \dots, t_n\}$  is a transaction database or a set of transactions, each of which contains items of the itemset  $I$ . Thus, each transaction  $t_i$  is a set of items such that  $t_i \subseteq I$ . An association rule is an implication of the form:  $X \rightarrow Y$ , where  $X \subset I$ ,  $Y \subset I$  and  $X \cap Y = \emptyset$ .  $X$  (or  $Y$ ) is a set of items, called itemset. If an itemset contains  $k$  items, it is called

$k$ -itemset. It is obvious that the value of the antecedent implies the value of the consequent. The process of mining association rules consists of two main steps. The first step is to identify all the itemsets contained in the data that are adequate for mining association rules. To determine that the itemset is frequent, it should satisfy at least the predefined minimum support count. To measure the support for an itemset, the following formal definition is used:

$$Supp(X) = \frac{count(X)}{N} \quad (1)$$

Where  $N$  is the total number of transactions in the transaction database  $T$  i.e.  $N = count(T)$ . The second step is to generate rules out of the discovered frequent itemsets. For doing so, a minimum confidence has to be defined. The formal definition to calculate the rule confidence is given by the following equation:

$$Conf(X \rightarrow Y) = \frac{count(X \cup Y)}{count(X)} \quad (2)$$

The confidence of the rule  $X \rightarrow Y$  is a measurement that determines how frequently items in  $Y$  appear in transactions that contain  $X$ . Different algorithms attempt to allow efficient discovery of frequent patterns and for strong ARs such as the famous *Apriori* algorithm [21] which will be used later in MFAR.

## 4 Methodology

MFAR consists of two main phases: online phase and offline phase. The next subsections describe in details the inputs, the outputs and the steps of each phase.

### 4.1 Offline Phase

The input of this phase is the image dataset which contains two modalities: the images and their associated text. First, the visual and the textual features are extracted to run the clustering algorithm independently over them. Then, the modified ARM algorithm will identify the relations among the clusters from each modality to construct the ARs (see figure 1.a).

For visual features extraction, we used a set of generic MPEG-7 descriptors [22]. The features are selected to balance the color and the edge properties of the images. After extracting the visual features, images of the dataset are represented separately as objects in multidimensional space models for each visual feature. For textual features, they were obtained by applying the standard Bag-of-Words technique which needs to perform several linguistic preprocessing steps (tokenization, removing stop words, and stemming). Then, each document is described by a vector of constituent terms that represents the

frequency occurrence of each term in the document which construct the vector-space model.

The large quantity of images and the high dimensionality of the visual descriptors need for an efficient clustering (or indexing) algorithm. The high dimensional index technique called NOHIS (Non Overlapping Hierarchical Index Structure) [23] is used for the indexing process which generates the NOHIS-tree. Then, an adapted k-nearest neighbors search is used for retrieving. On the other hand, K-means algorithm will be used for the textual features.

To apply the ARM algorithm, we need first to determine the items set  $I$  and the transaction database  $T$ . In our case, the items set is the generated images clusters based on the text (denoted by  $Ct_i$ ) and based on the visual features (denoted by  $Cc_j$  for color-based clusters and  $Ce_k$  for edge-based clusters) where  $i, j$  and  $k$  are the identifiers of the clusters in each modality. After quantifying the features space of each modality, we aim to associate the text clusters and the visual feature clusters. Thus, we need to construct the transaction database  $T$  first to run the ARM algorithm over it.

Each transaction in  $T$  contains the similar clusters from different modalities. Similarity here means the overlapping degree between the clusters. If the cardinality of the common images set is not zero, the clusters combine at the same transaction. It is possible to represent that in the following example: If  $|Ct_i \cap Cc_j| > 0$ , then add  $\{Ct_i, Cc_j\}$  to  $T$ . The hypothesis in constructing  $T$  is that similar clusters tends to be semantically related; therefore, they are combined at the same transaction. We are interested in the association between text clusters and visual feature clusters only. Each transaction contains a text cluster and at least one visual cluster. The following are examples of the obtained transactions:  $\{Ct_0, Cc_{111}\}$ ,  $\{Ct_0, Ce_{206}\}$ ,  $\{Ct_0, Cc_{111}, Ce_{173}\}$ .

Two different reasons let us adjust the formal definitions of support and confidence (definitions (1) and (2)). First, using the standard support/confidence definition for the semantic rules, which is calculated for the entire  $T$ , will affect the generated rules because their support is extremely low. Second, the calculation of support and confidence is restricted within the result set of the text clusters because we are testing the semantic relations between the text clusters and visual clusters. Thus, we define the support and the confidence of the rule  $Ct_i \rightarrow Cv_j$  (where  $Cv$  represents the visual cluster) as follows:

$$Supp(Ct_i \rightarrow Cv_j) = \frac{count(Ct_i, Cv_j)}{count(Ct_i)} \quad (3)$$

$$Conf(Ct_i \rightarrow Cv_j) = \frac{count(Ct_i, Cv_j)}{\max_k(count(Ct_i, Cv_k))} \quad (4)$$

Where  $count(A)$  is the number of itemsets that contain  $A$  in  $T$ . Similarly in case there is more than one item at the right hand side of the rule is given by (5) and (6):

$$Supp(Ct_i \rightarrow \{Cv_j | j=1, \dots, m\}) = \frac{count(Ct_i, \{Cv_j | j=1, \dots, m\})}{count(Ct_i)} \quad (5)$$

$$Conf(Ct_i \rightarrow \{ Cv_j | j=1, \dots, m \}) = \frac{count(Ct_i, \{ Cv_j | j=1, \dots, m \})}{\max_k(count(Ct_i, Cv_k))} \quad (6)$$

We need to use a modified version of frequent itemsets mining algorithm based on *Apriori* algorithm with definitions (5) and (6) of support and confidence to discover all frequent patterns of the association between text clusters and visual feature clusters. The algorithm is in table 1. The algorithm do not start from 1-itemsets; that because we want to construct the relationships between text clusters and visual clusters; and in case starting from 1-itemsets, it is possible to build relations among visual clusters since they will be treated equally. The minimum support threshold should be given to run the algorithm.

Here, *apriori-gen* function is used to perform three main operations: (1) candidate generation; (2) candidate pruning; and (3) insuring that each candidate itemset should have one text cluster. The *subset* function is used to determine all the candidate itemsets in  $C_k$  that are contained in each transaction  $t$ . A transaction  $t$  is said to contain an itemset  $X$  if  $X$  is a subset of transaction  $t$ .

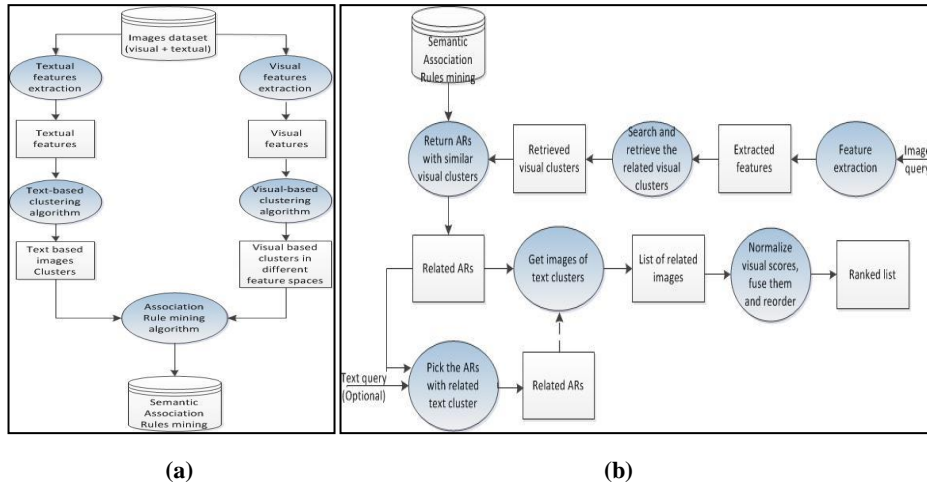


Fig. 1. The offline (a) and online phase (b) of MFAR



**Table 1.** Frequent itemsets mining algorithm based on Apriori

<p><b>Input:</b></p> <ul style="list-style-type: none"> <li>a) The transaction database <math>T</math></li> <li>b) <math>minsup</math> threshold</li> </ul> <p><b>Output:</b></p> <p>The list of frequently itemsets <math>L</math></p> <ol style="list-style-type: none"> <li>1) <math>L_2 = \{(C_{t_i}, C_{v_j}) \mid \text{where }  C_{t_i} \cap C_{v_j}  &gt; 0 \ \&amp;\&amp; \ (C_{t_i}, C_{v_j}).supp \geq minsup\}</math>; //Find all frequent 2-itemsets</li> <li>2) for (<math>k = 3 ; L_{k-1} \neq \emptyset ; k++</math>) do begin</li> <li>3) <math>C_k = \text{apriori-gen}(L_{k-1})</math>; // New candidates with <math>k</math>-itemset with only one text cluster in it and a // combination of frequent sets from <math>L_{k-1}</math></li> <li>4) for all transactions <math>t \in T</math> do begin</li> <li>5) <math>C_t = \text{subset}(C_k, t)</math>; // Identify all candidates that belong to <math>t</math></li> <li>6) for all candidates <math>c \in C_t</math> do</li> <li>7) <math>c.count++</math>;</li> <li>8) end</li> <li>9) <math>L_k = \{c \in C_k \mid c.supp \geq minsup\}</math></li> <li>10) end</li> <li>11) Return <math>\cup L_k</math>;</li> </ol>
--

To generate strong ARs, the generated frequent itemsets  $L$  and the minimum confidence threshold value  $minconf$  should be used as input to the generating algorithm. The ARs in our case have one text cluster in the left hand side and one or multiple visual cluster(s) at the right hand side. There is no need to find all possible subsets of the large itemset  $L$  as in the original *Apriori* algorithm. For example, if  $l = \{C_{t_1}, C_{c_3}, C_{e_1}\}$  is a frequent itemset, candidate rule is  $C_{t_1} \rightarrow \{C_{c_3}, C_{e_1}\}$ . If the calculated confidence of the candidate rule using (6) is greater than or equal  $minconf$ , then the rule is strong; otherwise, it is discarded. Finally, all the generated ARs are stored in the database along with the values of support and confidence for each rule which is the final output of this phase.

#### 4.2 Online phase.

This phase uses the generated ARM of the offline phase. The main processes are illustrated in figure 1.b. The basic query model used here is the query by example image since when image is used as query, all the information it contains is provided to the system. Using a keyword as a query is optional. It could be provided to the system to support the results that generated by the image query. For the query image, we need to extract the same visual features that have been extracted from the image dataset. For the optional keyword query, we used one keyword and simple text matching to simplify this step.

We need to use the same index NOHIS-trees of the offline phase to retrieve the relevant clusters to the query image for each visual descriptor. In our case, we have two different NOHIS-trees for two different feature spaces. For each feature, we calculated the top 500 nearest neighbors and returned their clusters. The search should be conducted on the trees in parallel. The output of this process is a list of visual clusters from different feature spaces.

Then, the next process “retrieve ARs with similar visual clusters” gets the list of the related visual clusters as input; and then it uses them to make a search in the ARM to find the rules that contain these clusters. If the keyword query was provided, the retrieved rules should be filtered to pick the rules which contain text clusters that have similar term to the text query. Then, the images’ scores in those text clusters should be increased. The dashed arrow in figure 1.b indicates that it is an optional path.

For all the retrieved ARs, we need to get the images of the text-based clusters. For each image, the relevant score to the query image  $q$  should be calculated if the image is not from the top 500 images for each visual feature. Regarding score normalization, we used Zero-One linear method which maps the scores into the range of [0, 1] [24]. The normalized scores of different modalities should be fused using CombSum method [24]. Then, if there is a keyword query as input, the fused score of each image that correlated to term similar to the keyword query should be incremented by one. Finally, the fused list will be reordered based on the fused scores.

## 5 Experiment

### 5.1 Experimental Setup and Tools

MFAR has been evaluated using ImageCLEF 2011 Wikipedia collection. It consists of 50 topics and 237,434 Wikipedia images along with their user-provided annotations in three different languages [25]. Since some images in the dataset do not have English description and others do not have a description at all, only images with English description are considered. Thus, the used dataset is a subset of ImageCLEF 2011 Wikipedia which contains more than 54,500 images. Some example topics of the dataset along with their titles, the used text query, the number of image queries in the topic and the number of relevant images in the subset dataset are given in table 2.

For visual features extraction, the two MPEG-7 descriptors: Color Structure Descriptor (CSD) and Edge Histogram Descriptor (EHD) are extracted from the dataset using the tool given in [26]. For textual features extraction and K-means clustering, Text-Garden software is used<sup>1</sup>. To cluster the extracted visual features, NOHIS algorithm library is provided by the author of the algorithm. The system prototype is developed in C#.NET

---

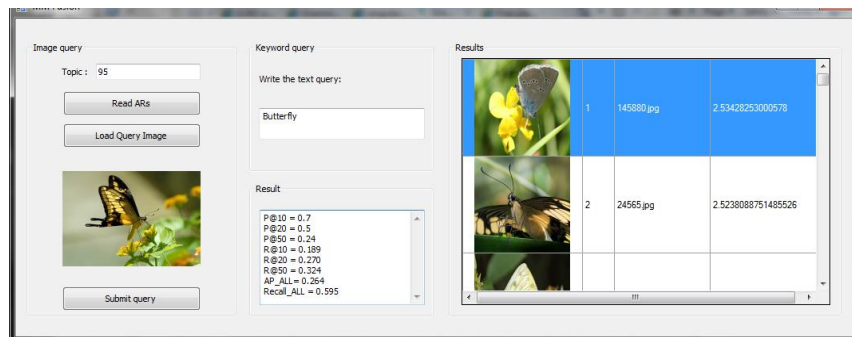
<sup>1</sup> Text-Garden – Text Mining Software Tools. <http://www.textmining.net>

Framework with simple GUI for experiment purpose only (see figure 2). Based on different experiments, we set *minsupp* and *minconf* to be 2% and 70% respectively.

MFAR was compared to our system without using ARs and to the online system MMRetrieval<sup>1</sup> [16]. Since MMRetrieval system supports different fusion methods, the well-known method CompSum with MinMax normalization is selected. We used the example images of all the dataset topics. For our system without ARs, the queries are only images. On the other hand, for MFAR and MMRetrieval, the query can be either image only or image with keyword. The text query is restricted to be one word.

**Table 2.** Information of some topics of the subset collection

Topic ID	Topic Title	Text query	No.# of query images	No# of relevant images
85	Beijing bird nest	Beijing	5	8
95	photo of real butterflies	Butterfly	5	37
107	sunflower close up	Sunflower	5	4
111	two euro coins	Euro	5	30
115	flying bird	Flying	5	46



**Fig. 2.** Main GUI of MFAR

## 5.2 Experimental Results and Discussion

For evaluation, we used the Precision (P) at fixed rank (10 and 20), and the Mean Average Precision (MAP) [27]. The values of P@10, P@20 and Average Precision (AP) of five random categories (with different difficulty levels; and they are not the best results) are

<sup>1</sup> <http://mmretrieval.nonrelevant.net>

given in table 3. Each value in the table represents the average of the precisions for the five example images contained in the topic. In addition, table 4 shows the overall values of P@10, P@20 and MAP for all topics of the dataset. The results show that MFAR with composite query (image + keyword) performs better precision and MAP than the other two systems. Furthermore, the proposed system and MMRetrieval system have been evaluated with an image query only without using text; and the proposed system performs acceptable semantic results comparing to MMRetrieval system and provides better precision results than MMRetrieval. The precision values with image query mode in both systems are lesser than the systems with composite query.

We examined the retrieved ARs for different visual queries to study the relations between the image query and the rules. One example is an image from topic 107 with title “sunflower close up”. Text cluster  $Ct_{645}$  is classified based on different words one of them is “sunflower”. The retrieved ARs for the query in the two query modes: query by image only and the composite query contain rules that associate  $Ct_{645}$  text cluster to visual clusters consists of sunflower pictures. That means by using the visual features of the query image, it is possible to reach the text cluster which is semantically related.

In addition, we found that by using MFAR the search operation concentrate on the images subset that included in the retrieved ARs of the submitted query which increases the chance of retrieving a semantically related results.

**Table 3.** P@10, P@20 and AP of 5 different topics in: (1) Sys.1: our system without ARs (visual), (2) Sys.2: MMRetrieval system (visual + text), and (3) Sys.3: MFAR (visual + text)

Topic ID	P@10			P@20			AP		
	Sys.1	Sys.2	Sys.3	Sys.1	Sys.2	Sys.3	Sys.1	Sys.2	Sys.3
85	0	0	0.2	0	0.013	0.13	0.001	0.035	0.282
95	0	0.72	0.66	0	0.62	0.56	0.004	0.366	0.246
107	0	0.28	0.3	0	0.15	0.15	0.004	0.468	0.658
111	0	0.38	0.4	0.01	0.23	0.47	0.018	0.236	0.350
115	0.12	0.14	0.22	0.07	0.11	0.18	0.031	0.033	0.047

**Table 4.** The overall values of P@10, P@20, and MAP of our system without ARs, MMRetrieval system, and MFAR

Sys. without ARs			MMRetrieval			MFAR		
P@10	P@20	MAP	P@10	P@20	MAP	P@10	P@20	MAP
0.011	0.009	0.010	0.243	0.168	0.242	0.294	0.212	0.288

## 6 Conclusion and Future Work

In this proposed method, we used association rules mining algorithm in our image retrieval system to construct semantic relations between image clusters based on the visual features and the image clusters based on textual features for the same dataset. From information fusion perspective, we have used late fusion technique. The online phase uses the constructed ARs from the offline phase. Then, the retrieval process requires an example image query to start. The method gives the ability to retrieve images that are semantically related by using the extracted visual features of the query image and by exploring the related ARs from the constructed ARM. To support the results, it is possible to use a keyword query. The results show that the precision value of our proposed system is better than MMRetrieval system and the system without association rules.

The future work will involve using different clustering algorithm to improve the accuracy of the text clusters. The system with image query mode without keyword query needs for further improvements. Using pseudo-relevance feedback technique is one suggested solution. The correlated terms of the top retrieved ARs could be used to make feedback text query. Also, it is possible to generalize the proposed method to use it for image annotation system by associating the unannotated images with the semantically related text cluster.

## 7 References

1. Liu, Y., Zhanga, D., Lua, G., Ma, W-Y.: A survey of content-based image retrieval with high-level semantics. *Pattern Recognition*, Vol. 40, No. 1, pp. 262-282. (2007)
2. Datta, R., Joshi, D., LI, J., Wang, J. Z.: Image retrieval: Ideas, influences, and trends of the new age”, *ACM Computing Surveys (CSUR)*. 40(2):1-60. (2008)
3. Wu, S., McClean, S.: Performance prediction of data fusion for information retrieval. *Information Processing, Management*. 42(4): pp. 899-915. (2006)
4. Müller, H., Clough, P., Deselaers, Th., Caputo, B.: *ImageCLEF* (ser. The Springer International Series on Information Retrieval), vol. 32, pp.95 -114. Springer-Verlag. (2010)
5. Ferecatu, M., Sahbi, H.:TELECOM ParisTech at ImageClefphoto 2008: Bi-modal text and image retrieval with diversity enhancement. In: *Working Notes of CLEF 2008*, Aarhus, Denmark. (2008)
6. Deselaers, T., Weyand, T., Ney, H.: Image retrieval and annotation using maximum entropy. In *Evaluation of Multilingual and Multi modal Information Retrieval*. pp. 725-734. (2007)
7. Gass, T., Weyand, T., Deselaers, T., Ney, H.: FIRE in ImageCLEF 2007: Support vector machines and logistic models to fuse image descriptors for photo retrieval. In: *CLEF 2007 Proceedings. Lecture Notes in Computer Science (LNCS)*. vol 5152. Springer, pp 492-499. (2007)
8. Bellman, R.: *Adaptive Control Process: A Guided Tour*. Princeton University Press (1961)
9. Atrey, P. K., Hossain, M. A., Saddik, A. E., Kankanhall, M. S.: Multimodal fusion for multimedia analysis: A survey. *Multimedia Syst*. vol. 16, no. 3, pp.1432 -1882. (2010)

10. Lau, C., Tjondronegoro, D., Zhang, J., Geva, S., Liu, Y.: Fusing visual and textual retrieval techniques to effectively search large collections of wikipedia images. *International Journal of Business Intelligence and Data Mining*. pp. 345-357. (2007)
11. Frigui, H., Caudill, J., Ben Abdallah, A.: Fusion of multi-modal features for efficient content-based image retrieval. *IEEE World Congress on Computational Intelligence*. pp. 1992-1998. (2008)
12. Bartolini, I., Ciaccia, P.: Scenique: a multimodal image retrieval interface. In *Proceedings of the working conference on Advanced visual interfaces*. ACM, Italy. pp. 476-477. (2008)
13. Zhou, X., Depeursinge, A., Müller, H.: Information fusion for combining visual and textual image retrieval. In *Proceedings of the 20th International Conference on Recognizing Patterns in Signals, Speech, Images, and Videos*. pp. 1590 - 1593. (2010)
14. Tong, H., He, J. R., Li, M. J., Zhang, C. S., Ma, W. Y.: Graph-based multi-modality learning. In *Proceeding of the ACM Int. Conf. on Multimedia*. pp.862 -871. (2005)
15. Escalante, H.J., Hernandez, C., Sucar, E., Montes, M.: Late fusion of heterogeneous methods for multimedia image retrieval. In: *Proceeding of MIR, ACM, Vancouver, Canada*. pp. 172–179. (2008)
16. Zagoris, K., Arampatzis, A., Chatzichristofis, S. A.: *www.MMRetrieval.net: a multimodal search engine*. In *proceedings of the Third International Conference on Similarity Search and Applications*. Turkey. (2010)
17. Ah-Pine, J., Bressan, M., Clinchant, S., Csurka, G., Hoppenot, Y., Renders, J.: Crossing textual and visual content in different application scenarios. *Multimedia Tools and Applications*, 42(1). pp.31–56. (2009)
18. Wei, S., Zhao, Y., Zhu, Z., Liu, N.: Multimodal Fusion for Video Search Reranking. *IEEE Transactions on Knowledge and Data Engineering*. v.22 n.8, pp.1191-1199. (2010)
19. He, R., Xiong, N., Yang, L., Park, J.: Using multi-modal semantic association rules to fuse keywords and visual features automatically for web image retrieval. In: *International conference on information fusion*. (2011)
20. Kouomou-Choupo, A., Berti-Equille, L., Morin, A.: Multimedia indexing and retrieval with features association rules mining. In *IEEE International Conference on Multimedia and Expo (ICME04)*, pp. 1299-1302. (2004)
21. Agrawal, R., Imielinski, T., and Swami, A., Mining association rules between sets of items in large databases. In *Proceedings of ACM SIGMOD-93*, pp. 207–216. (1993)
22. Manjunath, B.S., Salembier, P., Sikora, T.: *Introduction to MPEG-7: multimedia content description interface*. Wiley. (2002)
23. Taileb, M., Lamrous, S., Touati, S.: Non Overlapping Hierarchical Index Structure. *International Journal of Computer Science*, vol. 3 no. 1, pp. 29-35. (2008)
24. Wu, S.: *Data Fusion in Information Retrieval*. Springer, Heidelberg. (2012)
25. Tsirikia, T., Popescu, A., Kludas, J.: Overview of the Wikipedia Image Retrieval Task at ImageCLEF 2011. In: *Working Notes of CLEF 2011, Amsterdam, The Netherlands*. (2011)
26. Bastan, M., Cam, H., Gudukbay, U., Ulusoy, O.: BilVideo-7: An MPEG-7 Compatible Video Indexing and Retrieval System. *IEEE MultiMedia*. vol. 17, no. 3, pp. 62-73. (2010)
27. Müller, H., Clough, P., Deselaers, Th., Caputo, B.: *ImageCLEF (ser. The Springer International Series on Information Retrieval)*, vol. 32, pp.81 -92. Springer-Verlag. (2010)

# أسلوب دمج المعلومات من وسائط متعددة لأنظمة استرجاع الصور

رانية أحمد غانم الغامدي

## الملخص

في الآونة الأخيرة، أصبحت أنظمة استرجاع الصور بصفة عامة وأنظمة استرجاع الصور بناء على المحتوى (CBIR) بصفة خاصة مجال بحث هام يُستخدم في مختلف المجالات. منذ الأيام الأولى، وأنظمة CBIR تعاني من وجود "مشكلة الفجوة الدلالية" الذي هو عدم وجود تطابق بين خصائص الصورة البصرية وبين النتائج التي يريدها ويتوقعها المستخدم. هذه الرسالة المقترحة تحاول سد هذه الفجوة من خلال تصميم نظام لـ CBIR للويب باستخدام تقنية دمج المعلومات من عدة وسائط.

أهم الدوافع لتنفيذ هذه الدراسة بالإضافة للحجم الهائل من الصور في الوسائط الرقمية هو أن دمج عدة وسائط يعتبر اتجاه جديد في أنظمة استرجاع الصور ويحتاج لمزيد من الدراسة. وحيث أن الصور في تطبيقات مختلفة تتواجد مع نصوص كتابية لها علاقة بالصورة فلا يوجد سبب لعدم استخدام كل المعلومات المتوفرة عن الصورة ودمجها لتنفيذ نظام استرجاع يوفر المعنى الدلالي للصورة من خلال استخدام صفاتها المرئية.

إنّ فالمشكلة الأساسية هي كيف يتم التكهن بالمعنى الدلالي للصورة من خلال صفاتها المرئية. غالبية الطرق الحالية تفتقر للقدرة على استخراج المعنى الدلالي من الصورة بفاعلية. لذلك فهناك حاجة لدراسة طريقة الاستعلام المناسبة للمستخدم وكيفية استخدامها لاسترجاع الصور ذات العلاقة. كنتيجة لذلك، العديد من الاسئلة ظهرت للسطح منها: كيف يمكن بناء العلاقة بين صفات الصور المرئية وصفاتها النصية في قاعدة صور كبيرة؟ وفي أي مرحلة؟ كيف يمكن فهرسة الصور باستخدام صفاتها المرئية والنصية مع التقليل من تدخل المستخدم؟ ماهي افضل طريقة للاستعلام والتي ستوفر المعلومات اللازمة لاسترجاع الصور ذات العلاقة؟

الهدف الرئيسي من هذه الرسالة هو تصميم وتنفيذ نظام يساهم في سد الفجوة الدلالية في أنظمة استرجاع الصور العامة كالصور المتواجدة في الشبكة العنكبوتية. أيضا من اهم اهداف الرسالة دراسة الأعمال الحالية في مجال دمج المعلومات في تطبيقات الوسائط المتعددة بصفة عامة وفي أنظمة استرجاع الصور بصفة خاصة ودراسة نقاط القوة والضعف في كل طريقة. نريد بالإضافة لذلك لدراسة الطريقة المقترحة والتحقق من فاعليتها بمقارنتها بطرق أخرى وتنفيذ التجربة على مجموعة صور منتقاة بعناية وتطابق مواصفات صور الشبكة العنكبوتية.

الأسلوب المقترح لدمج معلومات الصورة النصية والمرئية لأنظمة استرجاع الصور - والتي هي الاتجاه الحديث في أبحاث استرجاع الصور- يجمع بين اثنين من تقنيات تنقيب البيانات لاسترداد الصور ذات الصلة لغويا: خوارزمية منجم قواعد التجميع (ARM) وخوارزمية المجموعات (clustering) ويسمى MFAR. منجم قواعد التجميع الدلالي يتم إنشاؤه في المرحلة الأولى offline حيث يتم في هذه المرحلة اكتشاف قواعد الارتباط بين مجموعات الصور المقسمة بناء على العلاقات اللغوية للنص و مجموعات الصور المقسمة بناء على المحتوى البصري. هذا المنجم يتم حفظه لاستخدامه لاحقا في مرحلة استرجاع الصور online.



للتأكد من فاعلية النظام المقترح، تم تنفيذ النظام باستخدام لغة C#.NET وباستخدام العديد من الأدوات لاستخراج الصفات المرئية والنصية للصور و لتقسيم الصور الى مجموعات clusters ثم أُجريت التجربة على 54545 من صور ImageCLEF 2011 ويكيبيديا. تم مقارنة نتائج MFAR مع نتائج نظام MMRetrieval و هو نظام على الانترنت تم انشاؤه لاسترجاع نفس مجموعة الصور المستخدمة في التجربة. والنظام الثاني الذي تمت المقارنة به هو نفس النظام المقترح في الرسالة ولكن دون استخدام منجم قواعد التجميع. أظهرت النتائج المتحصل عليها أن الطريقة المقترحة قد حققت أفضل قيمة للدقة Precision, Recall and Mean Average Precision بين فئات استعلام مختلفة.

## المستخلص

في الآونة الأخيرة، أصبحت أنظمة استرجاع الصور بصفة عامة وأنظمة استرجاع الصور بناء على المحتوى (CBIR) بصفة خاصة مجال بحث هام يُستخدم في مختلف المجالات. منذ الأيام الأولى، وأنظمة CBIR تعاني من وجود "مشكلة الفجوة الدلالية" الذي هو عدم وجود تطابق بين خصائص الصورة البصرية وبين النتائج التي يريدها ويتوقعها المستخدم. هذه الرسالة المقترحة تحاول سد هذه الفجوة من خلال تصميم نظام لـ CBIR للويب باستخدام تقنية دمج المعلومات من عدة وسائط. الهدف الرئيسي من هذه الرسالة هو تصميم وتنفيذ نظام يساهم في سد الفجوة الدلالية في أنظمة استرجاع الصور العامة كالصور المتواجدة في الشبكة العنكبوتية. أيضا من اهم اهداف الرسالة دراسة الأعمال الحالية في مجال دمج المعلومات في تطبيقات الوسائط المتعددة بصفة عامة وفي أنظمة استرجاع الصور بصفة خاصة ودراسة نقاط القوة والضعف في كل طريقة.

الأسلوب المقترح لدمج معلومات الصورة النصية والمرئية لأنظمة استرجاع الصور - والتي هي الاتجاه الحديث في أبحاث استرجاع الصور- يجمع بين اثنين من تقنيات تنقيب البيانات لاسترداد الصور ذات الصلة لغويا: خوارزمية منجم قواعد التجميع (ARM) وخوارزمية المجموعات (clustering) ويسمى MFAR. منجم قواعد التجميع الدلالي يتم إنشاؤه في المرحلة الأولى offline حيث يتم في هذه المرحلة اكتشاف قواعد الارتباط بين مجموعات الصور المقسمة بناء على العلاقات اللغوية للنص و مجموعات الصور المقسمة بناء على المحتوى البصري. هذا المنجم يتم حفظه لاستخدامه لاحقا في مرحلة استرجاع الصور online.

للتأكد من فاعلية النظام المقترح، تم تنفيذ النظام باستخدام لغة C#.NET وباستخدام العديد من الأدوات لاستخراج الصفات المرئية والنصية للصور و لتقسيم الصور الى مجموعات clusters ثم أجريت التجربة على أكثر من 54500 من صور ImageCLEF 2011 ويكيبيديا. تم مقارنة نتائج MFAR مع نتائج نظام MMRetrieval و هو نظام على الانترنت تم انشاؤه لاسترجاع نفس مجموعة الصور المستخدمة في التجربة. والنظام الثاني الذي تمت المقارنة به هو نفس النظام المقترح في الرسالة ولكن دون استخدام منجم قواعد التجميع. أظهرت النتائج المتحصل عليها أن الطريقة المقترحة قد حققت أفضل قيمة للدقة Precision, Recall and Mean Average Precision بين فئات استعلام مختلفة.



# أسلوب دمج المعلومات من وسائط متعددة لأنظمة استرجاع الصور

رانية أحمد غانم الغامدي

بحث مقدم لنيل درجة الماجستير في العلوم تخصص علوم حاسبات

إشراف

د. محمد عبدالشكور أمين

د. منيرة محمد طييب

كلية الحاسبات وتقنية المعلومات

جامعة الملك عبدالعزيز

جدة - المملكة العربية السعودية

شعبان 1435هـ - يونيو 2014م

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

قال تعالى:

(( وَمَا تَوْفِيقِي إِلَّا بِاللَّهِ عَلَيْهِ تَوَكَّلْتُ وَإِلَيْهِ أُنِيبُ )) سورة هود: آية 88



# أسلوب دمج المعلومات من وسائط متعددة لأنظمة استرجاع الصور

رانية أحمد غانم الغامدي

بحث مقدم لنيل درجة الماجستير في العلوم تخصص علوم حاسبات

كلية الحاسبات وتقنية المعلومات  
جامعة الملك عبدالعزيز  
جدة - المملكة العربية السعودية  
شعبان 1435هـ - يونيو 2014م