Open Access

Combining Textual and Visual Information for Image Retrieval in the Medical Domain

Yiannis Gkoufas*, Anna Morou* and Theodore Kalamboukis*

Department of Informatics, Athens University of Economics and Business, Greece

Abstract: In this article we have assembled the experience obtained from our participation in the imageCLEF evaluation task over the past two years. Exploitation on the use of linear combinations for image retrieval has been attempted by combining visual and textual sources of images. From our experiments we conclude that a mixed retrieval technique that applies both textual and visual retrieval in an interchangeably repeated manner improves the performance while overcoming the scalability limitations of visual retrieval. In particular, the mean average precision (MAP) has increased from 0.01 to 0.15 and 0.087 for 2009 and 2010 data, respectively, when content-based image retrieval (CBIR) is performed on the top 1000 results from textual retrieval based on natural language processing (NLP).

Keywords: Information storage and retrieval, data fusion, content based image retrieval, digital libraries.

1. INTRODUCTION

The explosion of information in the last 20 years over the Internet has made information seeking for both textual and visual objects a very hot topic of research. In the medical domain, in particular, the vast volumes of visual information produced every day in hospitals in connection with the existence of digital Picture Archiving and Communications Systems (PACS) make the need imperative for advanced ways of searching, i.e., by moving beyond conventional text-based searching towards combining both text and visual features in search queries. Indeed biomedical information comes in several forms: as text in scientific articles, as images or illustrations from databases and Electronic Health Records (EHR). Although many methods and tools have been developed, still, we are far from an effective solution especially in the case of image retrieval from large and heterogeneous databases. One way towards the improvement of current retrieval facility is data fusion. Data fusion is generally defined as the use of techniques that combines data from multiple sources and gather that information in order to achieve inferences, which will be more efficient and accurate than if they are achieved by means of a single source.

It is evident from the literature that there is a lot of room for improvement in image retrieval. For example, techniques for image annotation with semantic information, is an active research topic. Furthermore, given that the text accompanying the images is usually a short paragraph, techniques for documentation and query expansion may be needed to overcome the language ambiguity, such as polysemy and synonymy.

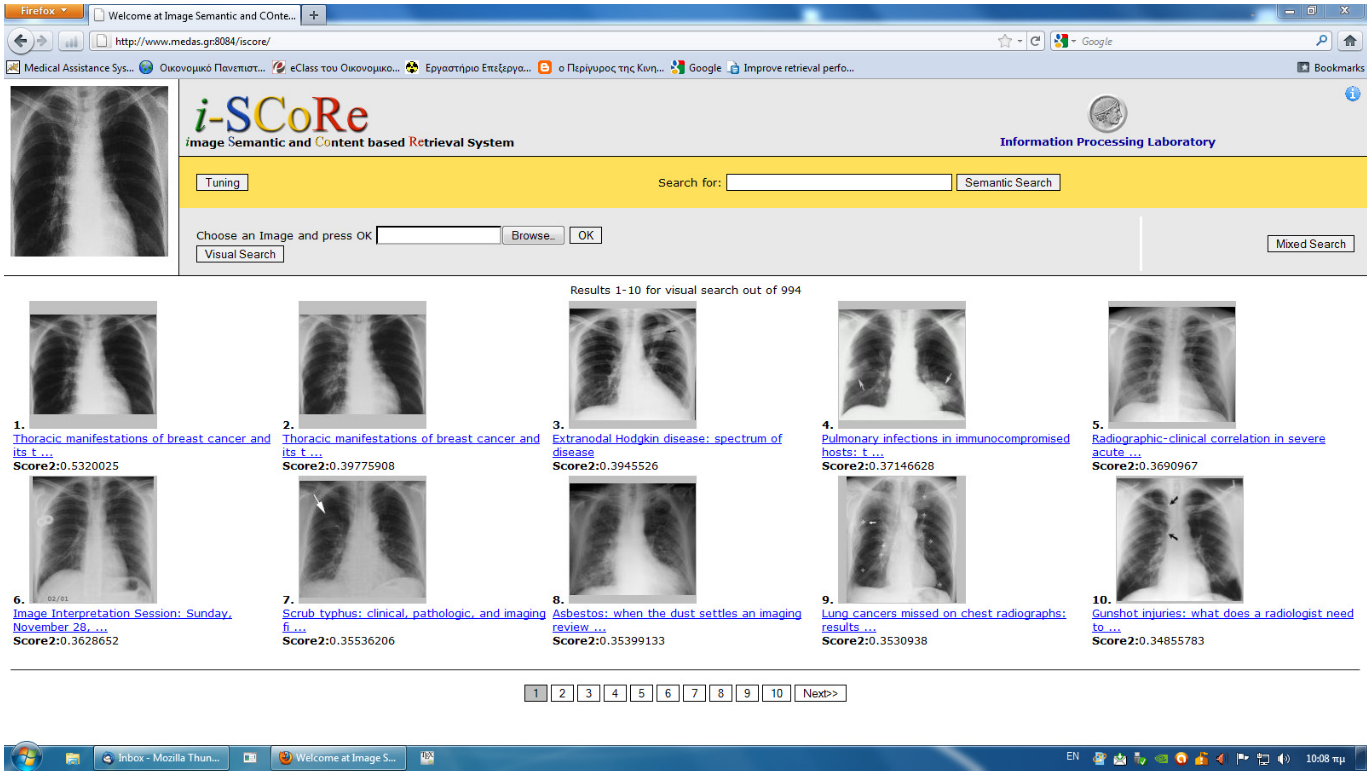
This article is an overview of the experience we have obtained through our participation in the imageCLEF Ad-Hock task in the last two years. In particular we present ways to improve retrieval performance by making use of textual as well as visual information. This information is extracted from an image itself and from textual descriptions like caption or from references to an image of an article, and ontologies. Thus to achieve our goal we combine techniques of information retrieval, content-based image retrieval (CBIR) and natural language processing (NLP). Our objective is to aid diagnosis by finding similar cases for a patient using several resources in the literature and in databases of EHR. A detailed account on imageCLEF 2009 and 2010 with the results of the official runs from all the participants and conclusions can be found in [1, 2].

To demonstrate our techniques, we have developed our own search engine (*i*-score), a hybrid system that uses both visual and textual resources. Our framework is built upon the Lucene[1]search engine and provides several ways to combine textual and visual search results. The system is capable of: (i) starting a text-based search of an image database, and refining the results using image features; (ii) starting a visual search (query by example) and applying relevance feedback with textual features that accompany an image; and, (iii) merging the results of independent text and image searches. The retrieved results can be viewed as thumbnails in a grid view sorted by relevance (Fig. **1**). Such a system may be used for computer-aided diagnosis, medical education and research purposes.

In what follows we report results from the databases used within the ImageCLEF track, evaluation forum, in the last two years. The results were evaluated using the trec_eval[2] package developed for evaluation of retrieval results within TREC. In section 2 we review the most common data fusion techniques. In sections 3 and 4 we describe our retrieval methods, followed by the section where we present our experimental results and finally conclusions are drawn with proposals for further work.

*Address correspondence to these authors at the Department of Informatics, Athens University of Economics and Business, Greece;
Tel: +302108203575; Fax: +302108676265;
E-mails: gkoufas@aueb.gr, morou@aueb.gr, tzk@aueb.gr

[1] http://lucene.apache.org/java/docs/index.html
[2] http://trec.nist.gov/trec_eval/


Firefox
Welcome at Image Semantic and COnte...
http://www.medas.gr:8084/iscore/
Google
Medical Assistance Sys...
Οικονομικό Πανεπιστ...
eClass του Οικονομικο...
Εργαστήριο Επεξεργα...
ο Περίγυρος της Κιvη...
Google
Improve retrieval perfo...
Bookmarks
i-SCoRe
image Semantic and Content based Retrieval System
Information Processing Laboratory
Tuning
Search for:
Semantic Search
Choose an Image and press OK
Browse..
OK
Visual Search
Mixed Search
1 2 3 4 5 6 7 8 9 10 Next>>
Inbox - Mozilla Thun...
Welcome at Image S...
EN
10:08 πμ

Results 1-10 for visual search out of 994

1. Thoracic manifestations of breast cancer and its t ...
Score2:0.5320025

2. Thoracic manifestations of breast cancer and its t ...
Score2:0.39775908

3. Extranodal Hodgkin disease: spectrum of disease
Score2:0.3945526

4. Pulmonary infections in immunocompromised hosts: t ...
Score2:0.37146628

5. Radiographic-clinical correlation in severe acute ...
Score2:0.3690967

6. Image Interpretation Session: Sunday, November 28, ...
Score2:0.3628652

7. Scrub typhus: clinical, pathologic, and imaging fi ...
Score2:0.35536206

8. Asbestos: when the dust settles an imaging review ...
Score2:0.35399133

9. Lung cancers missed on chest radiographs: results ...
Score2:0.3530938

10. Gunshot injuries: what does a radiologist need to ...
Score2:0.34855783

Fig. (1). A screenshotof *i*-score, our semantic and content-based image retrieval system on an image-query (top left corner).

2. DATA FUSION TECHNIQUES

Data fusion, is defined as the use of techniques that combine data from multiple sources in order to achieve inferences, which will be more efficient and accurate than the retrieval results achieved by means of a single source. We distinguish three types of fusion algorithms:

(a) those that combine from different retrieval systems;

(b) those that fuse from different document representations and

(c) those that combine from several sources (databases).

Traditionally the methods used for data fusion are based either on the similarity values of the documents across the ranked lists, or the ranks of the documents across the lists. The main factors related to the design of data-fusion algorithm deal with the existence or the absence of the `three effects': skimming effect, chorus effect, and dark horse effect. Vogt and Cottrell [3] described those effects as follows:

Chorus effect: this effect suggests that for a particular document if it is retrieved by more systems than another document it will be "better". "Better" means that the document has a higher probability to be relevant. This is considered as a very significant effect and any data-fusion algorithm should take this effect into account.

Skimming effect: relevant documents are most likely to occur on the top of the retrieved list for each individual retrieval system, so any fusion algorithm that chooses the top ranked documents from each individual retrieval system is expected to be more efficient.

Dark horse effect: usually different retrieval systems retrieve different number of relevant documents. This effect assumes that a good fusion algorithm should treat the systems which retrieve a larger number of relevant documents differently than other systems which don't retrieve a large number of relevant documents. This means that we should give more importance (or weight) to a retrieval system based on the number of relevant documents it has retrieved.

We are interested in fusion methods that use more than one resource and in particular the sources with a large variation on performance. Such fusion techniques may be used on image retrieval from both textual and visual features. So far it has been proved inside the ImageCLEF track that text–based systems overwhelmly outperformed visual systems, sometimes by up to a factor of ten [2]. It is therefore important to determine optimal fusion strategies allowing overall performance improvement over the constituent systems.

The classical approaches such as CombMAX, CombSUM, CombMNZ[4] have been commonly employed in the literature for fusion tasks. However, these three methods have their limitations. On the one hand, CombMAX favors the documents highly ranked in one system (Dark Horse Effect) and is thus not robust to errors. On the other, CombSUM and CombMNZ favor the documents widely returned to minimize the errors (Chorus Effect) but in this way non-relevant documents can obtain high ranks if they are returned by few systems. Two other important issues of fusion are the normalization of the input scores [4, 5] and the tuning of the respective weights (i.e. contributions) given to each system [6].

A good introduction of the classical approaches to data fusion is given in [7]. In our experiments we concentrate basically on linear fusion methods which are briefly described in the next section.

3. LINEAR COMBINATION FUNCTIONS

The most simple and effective fusion method is the CombSUM, which sums up all the scores of a document, from all the retrieval lists:

$$CombSUM(q,d) = \sum_i score_i(q,d) \tag{1}$$

where $score_i$ is the similarity score of the document to the query for the i-th retrieval system. Since different retrieval systems generate different ranges of similarity scores, it is necessary to normalize the similarity scores of the documents. A normalization proposed by Lee [8] is defined as Eq. (2):

$$NormScore_i = \frac{score_i - MinScore}{MaxScore - MinScore} \tag{2}$$

All the variables are related to a given query q in a given resultant list. Whereas $MaxScore$ and $MinScore$ are the maximum and minimum scores in the resultant list, respectively; $score_i$ refers to the score that a document d obtained initially; and $Normscore_i$ the normalized score that d should obtain.

The CombMAX and CombSUM rules both have drawbacks. CombMAX is not robust to errors as it is based on a single run for each image. CombSUM has the disadvantage of being based on all runs and thus includes runs with low performance. However, the best fused runs of the test data are obtained by using CombSUM with logarithmic rank normalization.

Many researchers have experimented with updated versions of CombSUM, where a weight is assigned to each retrieval strategy according to its performance on the training data. *WeightedSUM* is a general linear combination formula as defined by:

$$WeightedSUM(q,d) = \sum_i w_i NormScore_i(q,d) \tag{3}$$

where w_i is a weight proportional to the performance of the i-th retrieval component.

Several weighting schemes have been proposed in the literature. Thompson (1993) [9] used this weighted linear combination method to fuse results in TREC-1. He found that the combined results, weighted by performance level, performed better than a combination using a unified weight (CombSum). Bartell *et al*. (1994) [10] used a numerical optimization method, conjugate gradient, to find good weights of different systems. The simplest one is the selection of the best performing values on a set of training examples. Another approach is to use w_i for the performance of the i-th retrieval system measured by the Mean Average Precision (MAP) value [11, 12], again on a set of training data. A third approach uses a combination of MAP and recall values [13]. Wu and McClean[14] use both system performance and dissimilarities between results. Here, the dissimilarity weight is defined as the average dissimilarity between the system in question, and all other systems over a group of training queries.

In the following section, we use the linear combination method proposed in [6] where the weight w_i of each system is determined by a function of its performance. A good choice is to use power functions of the performance of all the participating systems in a fusion process $w_i = MAP_i^p$. In our experiments, we have used several values for the power p. From most of these experiments it seems that a value of $p > 1$ always improves the performance in the fused results.

4. TEXTUAL AD-HOC RETRIEVAL

The ad hoc task involves retrieving relevant images using the text associated with each image query. For this task we have investigated several similarity functions [15] with the Lucene search engine: the default similarity function, the BM25 [16] and several other variants. BM25 is a very successful weighting scheme based on the probabilistic model of information retrieval. These two methods are the most commonly used to retrieve documents with multiple fields. The simplest approach to retrieval is to ignore the structure of the documents, by simply merging all the data from the documents in one field and then perform standard information retrieval. The alternative is to perform individual retrieval for each field separately, and then form the sum of the resulting ranked lists to produce a single combined document list for the output. In this latter method of fusion the fields maybe weighted prior to merging at indexing time. The BM25F combination approach uses a simple weighted summation of the multiple fields of the documents to form a single field for each document in the usual way. The importance of each document field is determined empirically. As we shall see in the next section the frequency of each term appearing in each field is multiplied by a scalar constant representing the importance of this field, and the components from all the fields are summed to form the overall representation of the term for indexing.

For indexing, Lucene search engine is used, with a default analyzer which performs tokenization, removes stop words, transforms words to lower case, and performs stemming using the Porter stemmer.

4.1. The BM25F Scoring Function

In a vector space model the general scoring function defined by the $TF \times IDF$ model is given by:

$$SCORE(q,d) = \sum_{t \in d} idf(t) \times tf(t,d) \tag{4}$$

where $tf(t,d)$ is the frequency of the term t inside a document d and $idf(t)$ denotes the number of documents that contain the term t. If a document, d, is organized into fields then term frequencies are calculated for each field separately. If f denotes a field in a document d then:

$$tf(t,d) = \sum_{f \in d} w_f \times tf_f(t,d) \quad (5)$$

where w_f is the weight or boost factor of the field f, and $tf_f(t,d)$ is the frequency of term t in the field f of a document d. This definition allows the use of the $TF \times IDF$ model to calculate the relevance of structured documents.

BM25F is an extension of BM25 scoring function adapted for structured documents. The impact of term frequencies to retrieval has been discussed in the BM25. Although it is evident that the probability of relevance of a document increases together with the frequency of query terms inside a document this increase is not linear. This is the reason why scoring functions use an increased saturated factor to estimate the weight of a query term. The intuition behind this is that the gain we get when seeing a term first time inside a document is greater from what we gain if we see the same term further down in the same document. This non-linear relation maybe logarithmic or a more complex function like the parameter k_1 used with the BM25. An example of such a function used with BM25 is:

$$\frac{tf(t,d)}{k_1 + tf(t,d)} \quad (6)$$

where k_1 is a constant which controls the linear increase of the frequency of term $tf(t,d)$.

An implementation of BM25F as was proposed by Perez-Iglesias *et al.* is given in [17]. First a normalized frequency of term *t* for each field, *f*, is calculated from Eq. (7):

$$tf_f(t,d) = \frac{count_f(t,d)}{1 + b_f\left(\frac{l_{d,f}}{l_f - 1}\right)} \quad (7)$$

where $count_f(t,d)$ is the number of occurrences of the term *t* in the field *f* of a document *d*, $l_{d,f}$ is the length of the field and l_f is the average length of the field.

The parameter b_f is similar to *b* of the BM25 model. The frequencies of the fields are combined linearly with the boost factors w_f :

$$tf(t,d) = \sum_{f \in d} w_f \times tf_f(t,f) \quad (8)$$

From these relations we get the BM25F scoring function:

$$BM25F(q,d) = \sum_{t \in q \cap d} \frac{tf(t,d)}{k_1 + tf(t,d)} \times idf(t) \quad (9)$$

where $tf(t,d)$ is defined in Eq.(5).

The default similarity function of the Lucene search engine that is suitable for retrieval of structured documents is based on a linear combination of the scores of each field of a document.

$$SCORE(q,d) = \sum_{f \in d} score(q,f) \quad (10)$$

where

$$score(q,f) = \sum tf_f(t,d) \times idf(t) \times w_f \quad (11)$$

and $tf_f(t,d) = \sqrt{count(t,f)}$. From these scoring functions we observe that with the Lucene default function the boosting factors w_f are applied before the linear combination of the $tf_f(t,d)$ values which may affect the retrieval performance.

5. VISUAL AD-HOC RETRIEVAL

LIRE (Lucene Image Retrieval)[3] is a light weight open source Java library for content based image retrieval [18]. It provides a simple way to retrieve images based on their color and texture characteristics. The LIRE creates a Lucene index of image features for CBIR.

The following low level features have been used individually or in several combinations with our databases:

1) **CEDD** (Color and Edge Directivity Descriptor), [19] incorporates color and texture information in a histogram. (144 elements of features).

2) **Color Histogram**, a representation of the distribution of RGB and HSV color space in an image. (512 elements of features).

3) **ColorOnly** contains the scalable color and color layout descriptors.

   a. The **scalable color** descriptor is a color histogram in HSV color space, which is encoded by a Haar transform. (64 elements of features)

   b. The **color layout** descriptor represents a spatial distribution of color of visual signals in a very compact form. (10 elements of features)

4) **Auto color correlation** is based on color (HSV color space) and includes information upon color correlation in an image. (16 features).

5) A combination of the color layout descriptor and edge histogram descriptor. The **edge histogram** descriptor represents the spatial distribution of five types of edges, namely four directional edges and one non-directional edge. It can retrieve images with similar semantic meaning. (80 features).

6) **FCTH (Fuzzy color and texture histogram)** [20] is another descriptor that combines in one histogram, color and texture information. (192 features).

7) **Gabor filter**, [21] is a linear filter used for edge detection. (60 features).

8) **Tamura texture**, [22] is consisted of six texture features corresponding to human visual perception:

[3] http://www.semanticmetadata.net/lire/.

coarseness, contrast, directionality, line-likeness, regularity, and roughness. (18 features).

6. EXPERIMENTAL RESULTS

As we have already mentioned we are interested in retrieval strategies with a large difference in effectiveness on the fused lists as it happens to be the case in image retrieval from both visual and textual sources. We use CombSUM and WeightedSUM with $w_i = MAP_i$ as a baseline method for our experiments. Extensive experiments conducted by [6] with TREC data, have concluded that a series of power functions $w_i = MAP_i^p$, with p between 2 to 8 are always better than the simple weighting schema with $p = 1$.

Following the CLEF practice four metrics are used to evaluate the fused retrieval results, including the MAP, the precision at the top of k retrieved images $k = 5,10,30$ and the number of retrieved and relevant images. Since the number of documents judged to be relevant is small in comparison with the size of collections, the binary preference (bpref) retrieval evaluation metric computed by trec_eval is also used, which appears to be more robust than MAP.

6.1. Data Collections

Throughout our examination tests we have used image collections from the imageCLEF Ad-Hoc task over the last two years (2009-2010). Both collections, which are actually almost the same, were made accessible by the Radiological Society of North America[4](RSNA). The 2009 database contained a total of 74,902 images, whilst the 2010 contained 77,506 images. In both collections, images are accompanying with a small text (figure caption). Also the PubMed IDs were also made available with each image thus we had access to the MeSH (Medical Subject Headings) terms created by the National Library of Medicine for PubMed[5].

The image-based topics were created using realistic methods and search topics were identified by surveying actual user needs. Twenty-five queries were selected as the topics for ImageCLEFmed 2009. Similarly, in 2010, sixteen topics were selected from those each retrieved at least one relevant image by the system. Each textual topic is accompanied by 2 to 4 sample images from other collections of ImageCLEFmed. Also with each topic, a French and a German translations of the original textual description were provided.

6.2. Multi-Field Textual Retrieval Results

In Table **1** we present the baseline results using Lucene's default similarity function with both databases of the years 2009 and 2010. All the textual information inside a document is concatenated into one unstructured field. The total number of relevant images in the 2009 database is 2362 and in 2010 database 999. In Table **4** we repeat the same process in reverse order, that is we use the values of the weights w_i, estimated from year 2010 queries to combine the results in the year 2009 database.

The weight of each field equals to the performance of the corresponding field over all the queries. As performance measure for each field the MAP was used. These values are given in Table **2**.

Table **3** presents the results from multi-field retrieval. Three fields are used: title, caption and MeSH terms. In Table **3** the values of the weights estimated with the 2009 queries are used for fusion of the results of the year 2010 queries.

In Table **4** we repeat the same process in the reverse order, i.e., we use the values of the weights w_i, estimated from year 2010 queries to combine the results in the year 2009 database.

6.3. Visual Multi-Feature Retrieval Results

Following the same steps for CBIR, Table **5** summarizes the performance from each individual feature. We have used all the features described in Section 5. By the term DefaultDoc we denote a combination of color layout and edge histogram, by ExtensiveDoc a combination of color layout, edge histogram and scalable color and finally by Fastdoc the color layout.

Out of several combinations of these features, we present in Table **6**, four combinations which give the best results. In Table **7** we used the performance results from the year 2009 queries presented in Table **5** for the combination of the results of the year 2010 queries. We mention here that for multi-image queries the simple CombSUM scoring function is used as defined by:

$$SCORE(q, \mathrm{Im}\, age) = \sum_{j=1}^{k} score(q, i_j) \qquad (12)$$

where the images $\{i_1, .. i_k\}$ represent the query.

6.4. Fusion from Both Visual and Semantic Sources

Table **8** presents the results of fusion from both semantic and visual retrieval. These two approaches have a significant difference in retrieval effectiveness. For this particular fusion task we have used two different approaches. One with linear combination of the results defined by Eq. (13):

$$SCORE(q,d) = w_1 * score_{textual}(q,d) + w_2 * score_{visual}(q,d)$$
(13)

where $w_1 = 0.39$ is the MAP value from 2009 textual retrieval task (Tables **3** and **4**) and $w_2 = 0.01$ the MAP value from visual retrieval (Tables **6** and **7**). From Table **8** we observe that the contribution of the visual results is so small that they leave the results in the textual lists unaltered. The second fusion approach is a filtering task of CBIR on a set of images retrieved from a textual query. In Table **9** results are presented from the two databases. The top 1000 documents retrieved from the textual queries are used for the CBIR. The documents are re-ranked according to their content based scores.

[4] http://www.rsna.org/

[5] http://www.pubmed.gov/

Table 1. Performance of Textual Retrieval with One Field

Datasets	MAP	Bpref	P@5	P@10	P@30	rel_ret
2009	0.4025	0.4281	0.6480	0.5840	0.5547	1870/2362
2010	0.3680	0.7249	0.5125	0.4125	0.2729	769/999

Table 2. Performance of Textual Multi-Field Retrieval on Title, Caption and MeSHterms

Datasets	Fields	MAP	Bpref	P@5	P@10	P@30	rel_ret
2009	Title	0.1130	0.1551	0.2750	0.3292	0.2750	943
	Caption	0.3348	0.3687	0.6960	0.6120	0.5373	1528
	MeSHterms	0.1015	0.1928	0.1250	0.1792	0.1528	928
2010	Title	0.1176	0.3187	0.1000	0.0875	0.0979	456
	Caption	0.3206	0.6544	0.5500	0.4937	0.3458	742
	MeSHterms	0.1102	0.3232	0.1250	0.1313	0.0917	396

Table 3. Fusion of Multi-Field Retrieval Results on the 2009 with $w_i = MAP_i^p$ and $p = 1,2$. The Same Weighted Parameters were Applied on the 2010 Data-Collection

Datasets	Power	MAP	Bpref	P@5	P@10	P@30	rel_ret
2009	p=1	0.3954	0.4169	0.6880	0.6280	0.5587	1882
	p=2	0.3799	0.4045	0.6880	0.6240	0.5640	1780
2010	p=1	0.3380	0.7496	0.5125	0.4437	0.3250	784
	p=2	0.3326	0.7212	0.5375	0.4937	0.3521	747

Table 4. Fusion on the Multi-Field Retrieval Results on the 2010 with $w_i = MAP_i^p$ and $p = 1,2$. The Same Weighted Parameters were Applied on the 2009 Data-Collection

Datasets	Power	MAP	Bpref	P@5	P@10	P@30	rel_ret
2010	p=1	0.3423	0.7498	0.5250	0.4312	0.3208	785
	p=2	0.3333	0.7269	0.5250	0.4812	0.3417	757
2009	p=1	0.3943	0.4163	0.6880	0.6240	0.5560	1877
	p=2	0.3829	0.4062	0.6880	0.6280	0.5680	1730

Table 5. CBIR Performance on Single Features on the Year 2009 Data Collection

Features	MAP	Bpref	P@5	P@10	P@30	rel_ret
DefaultDoc	0.0097	0.0367	0.0320	0.0360	0.0360	302
ExtensiveDoc	0.0086	0.0301	0.0160	0.0280	0.0320	261
CEDD	0.0054	0.0322	0.0080	0.0280	0.0213	204
FastDoc	0.0035	0.0263	0.0080	0.0160	0.0187	171
FCTH	0.0030	0.0267	0.0080	0.0080	0.0133	185
ColorOnly	0.0024	0.0209	0.0160	0.0160	0.0120	121
AutoColorCorrelation	0.0017	0.0156	0.0160	0.0120	0.0053	93
ColorHistogram	0.0014	0.0151	0.0080	0.0080	0.0080	93
Tamura	0.0012	0.0281	0.0000	0.0120	0.0120	115
Gabor	0.0002	0.0160	0.0000	0.0000	0.0027	55

Table 6. CBIR Performance with Fusion on Three Features, $w_i = MAP_i^p$ and $p = 1,2$ on the Year 2009 Collection

Features	Power	MAP	Bpref	P@5	P@10	P@30	rel_ret
DefaultDoc/Tamura/Gabor	p=1	0.0098	0.0369	0.0320	0.0360	0.0373	301
	p=2	0.0097	0.0367	0.0320	0.0360	0.0360	302
Default/ExtensiveDoc/CEDD	p=1	0.0100	0.0326	0.0400	0.0320	0.0333	298
	p=2	0.0100	0.0324	0.0400	0.0240	0.0373	300
Default/Extensive/Fast	p=1	0.0097	0.0324	0.0320	0.0320	0.0360	290
	p=2	0.0096	0.0326	0.0320	0.0280	0.0333	287
Default/Extensive/Fast/CEDD	p=1	0.0101	0.0323	0.0320	0.0240	0.0373	302
	p=2	0.0102	0.0326	0.0400	0.0280	0.0360	303

Table 7. CBIR Performance with 2010 Collection and Fusion on the Same Features-Weights Learned on the Year 2009 Collection

Features	Power	MAP	Bpref	P@5	P@10	P@30	rel_ret
DefaultDoc/Tamura/Gabor	p=1	0.0095	0.0222	0.0500	0.0250	0.0104	63
	p=2	0.0097	0.0229	0.0500	0.0250	0.0104	60
Default/ExtensiveDoc/CEDD	p=1	0.0084	0.0203	0.0375	0.0312	0.0167	62
	p=2	0.0086	0.0219	0.0375	0.0188	0.0146	61
Default/Extensive/Fast	p=1	0.0098	0.0221	0.0375	0.0250	0.0125	59
	p=2	0.0097	0.0227	0.0375	0.0250	0.0146	61
Default/Extensive/Fast/CEDD	p=1	0.0074	0.0198	0.0250	0.0250	0.0167	57
	p=2	0.0087	0.0216	0.0375	0.0188	0.0146	60

Table 8. Data Fusion from Semantic and Visual Retrieval

Database	Features	MAP	Bpref	P@5	P@10	P@30	rel_ret
2009	DefaultDoc/Tamura/Gabor	0.3984	0.4218	0.6880	0.6320	0.5587	1899
	Default/ExtensiveDoc/CEDD	0.3990	0.4222	0.6880	0.6320	0.5587	1902
	Default/Extensive/Fast	0.3988	0.4220	0.6880	0.6320	0.5587	1900
2010	DefaultDoc/Tamura/Gabor	0.3424	0.7498	0.5250	0.4312	0.3208	785
	Default/ExtensiveDoc/CEDD	0.3424	0.7498	0.5250	0.4312	0.3208	785
	Default/Extensive/Fast	0.3424	0.7498	0.5250	0.4312	0.3208	785

Table 9. CBIR Performance on the Top 1000 Results Returned from Textual Retrieval, Fusion with *p*=1

Database	Features	MAP	Bpref	P@5	P@10	P@30	rel_ret
2009	DefaultDoc/Tamura/Gabor	0.1419	0.2684	0.2160	0.2280	0.2027	1881
	Default/ExtensiveDoc/CEDD	0.1496	0.2789	0.2240	0.2400	0.2147	1881
	Default/Extensive/Fast	0.1431	0.2782	0.2320	0.2240	0.2053	1881
2010	DefaultDoc/Tamura/Gabor	0.0870	0.1444	0.1500	0.1313	0.0979	785
	Default/ExtensiveDoc/CEDD	0.0778	0.1607	0.1250	0.1500	0.1083	785
	Default/Extensive/Fast	0.0765	0.1373	0.1375	0.1375	0.0979	785

7. DISCUSSION

Most systems simply use textual features to find similar images. Our goal is to improve the performance of multi-modal (text and image) information retrieval by combining both visual and semantic retrieval methods. However, from one side, semantic retrieval has reached to a point with no further improvement over the last few years, and from the other side visual retrieval still has very poor performance and far from been acceptable for commercial use. Combinations of these two approaches may raise the issue of search engines to a new dimension particularly in the field of retrieving medical information. To this respect, we have run a number of experiments from approaches of either independently or in combination. From our experimental results we can conclude that multi-field retrieval on textual data is always beneficial.

Certainly, there is still free space for improvements. One such improvement may be in the choice of the weighting parameters of a linear combination model. In our experiments we estimated the weights of the contributed systems in the fusion function by the performance of each individual system. We intend to estimate these weights by applying machine learning techniques upon a set of training queries. Such an approach may offer some additional and desirable properties for adaptability to the user profile. Furthermore there is a lot of room for improvement by incorporating knowledge from other resources using ontologies and thesauruses, like UMLS, for query expansion and lexical entailment. Captions may also be enriched by references to figures from inside the articles. Finally compound words may be split-up to extend the queries as well as the documents. Some of these propositions are currently under investigation and others will be dealt with in the near future.

Similarly several techniques may improve the visual retrieval. It seems that global features of images do not have a good discrimination value. Thus techniques for image segmentation using local features may improve CBIR while keeping the complexity to acceptable levels. An interesting result for CBIR comes from Table **9** where CBIR is restricted to the top 1000 images returned by an initial textual query. This approach not only improves significantly the performance of CBIR but also makes the method scalable to large image collections.

ACKNOWLEDGEMENT

None declared.

CONFLICT OF INTEREST

None declared.

REFERENCES

[1] Muller H, Kalpathy-Cramer J, *et al.* Overview of the CLEF 2009 medical image retrieval track. Proceedings of the 10th international conference on Cross-language evaluation forum: multimedia experiments.; 2009 Corfu, Greece. Springer-Verlag, Berlin, Heidelberg 2010; 72-84.

[2] Muller H, Kalpathy-Cramer J, Eggel I, *et al.* Overview of the CLEF 2010 medical image retrieval track. Working Notes of CLEF 2010 Padua, Italy, September 2010.

[3] Vogt CC, Cottrell GW. Fusion *via* a Linear Combination of Scores. Information Retrieval 1999; 1: 151-73.

[4] Zhou X, Depeursinge A, Muller 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 2010; pp. 1590-3

[5] Wu S, Crestani F, Bi Y. Evaluating Score Normalization Methods in Data Fusion. Asia Information Retrieval Symposium (AIRS) Singapore 2006; pp. 642-8

[6] Wu S, Bi Y, Zeng X, Han L. Assigning appropriate weights for the linear combination data fusion method in information retrieval. Info Process Manage 2009; 45: 413-26.

[7] Christensen HU, Ortiz-Arroyo D. Applying data fusion methods to passage retrieval in QAS. Proceedings of the 7th international conference on Multiple classifier systems. MCS'07, Springer-Verlag, Berlin, Heidelberg 2007; 82-92.

[8] Lee JH. Combining multiple evidence from different properties of weighting schemes. Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '95, ACM, New York, NY, USA 1995; 180-88.

[9] Thompson P. Description of the PRC CEO algorithm for TREC-2. The Second Text Retrieval Conference. NIST Special Publication 500-215 1993; 271-4

[10] Bartell BT, Cottrell GW, Belew RK. Automatic Combination of Multiple Ranked Retrieval Systems. Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. SIGIR 1994; 173-81

[11] He D, Wu D. Toward a Robust Data Fusion for Document Retrieval. Proceedings of the 2008 IEEE International Conference on Natural Language Processing and Knowledge Engineering. IEEE NLP-KE 2008; pp. 1-8.

[12] Alzghool M, Inkpen DZ. Cluster-based Model Fusion for Spontaneous Speech Retrieval. Proceedings of the ACM SIGIR Workshop on Searching Spontaneous Conversational Speech, 2008 July, Singapore; 4-10

[13] Alzghool M, Inkpen DZ. Model Fusion Experiments for the CLSR Task at CLEF 2007. In: C. Peters *et al.*, Eds. Advances in Multilingual and Multimodal Information Retrieval, 2008, Springer-Verlag Berlin Heidelberg, LNCS 5152: 695-702

[14] Wu S, McClean S. Performance prediction of data fusion for information retrieval. Information Processing Management 2006; 42: 899-915.

[15] StougiannisA, Gkanogiannis A, Kalamboukis T. IPL at imageCLEF2010. Working Notes of CLEF 2010 Padua, Italy, September 2010.

[16] Jones KS, Walker S, Robertson SE. A probabilistic model of information retrieval: development and comparative experiments. Information Processing Management 2000; 36: 779-808.

[17] Perez-Iglesias J, Perez-Aguera JR, Fresno V, Feinstein YZ. Integrating the Probabilistic Models BM25/BM25F into Lucene. Computer Research Repository (CoRR), abs/0911.5046 (2009) available from: http://arxiv.org/PS_cache/arxiv/pdf/0911/0911.5046v2.pdf

[18] Lux M, Chatzichristofis SA. Lire: Lucene image retrieval: an extensible java CBIR library. In: Proceeding of the 16th ACM international conference on Multimedia. MM '08, New York, USA 1085-8.

[19] Chatzichristofis SA, Boutalis YS. CEDD: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval. Proceedings of the 6th international conference on Computer vision systems. ICVS'08, Springer-Verlag, Berlin, Heidelberg 2008; 312-322.

[20] Chatzichristofis SA, Boutalis YS. FCTH: Fuzzy Color and Texture Histogram - A Low Level Feature for Accurate Image Retrieval. Proceedings of the 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services IEEE Computer Society, Washington, DC, USA 2008; pp. 191-6.

[21] Ng CBR, Lu G, Zhang D. Performance Study of Gabor Filters and Rotation Invariant Gabor Filters, Proceedings of the 11th International Multimedia Modelling Conference, MMM '05, 2005; 158-62.

[22] Tamura H, Mori S, Yamawaki T. Textural features corresponding to visual perception. IEEE Trans Syst Man Cybern 1978; 8(6): 460-72.

Received: May 15, 2011 Revised: May 20, 2011 Accepted: May 24, 2011