

A Model of Multimodal Fusion for Medical Applications

S. Yang¹, I. Atmosukarto¹, J. Franklin², J. F. Brinkley², D. Suci¹, and L. G. Shapiro¹
¹Dept. of Computer Science & Engineering, University of Washington, Seattle, WA 98195
²Dept. of Biological Structure, University of Washington, Seattle, WA 98195

ABSTRACT

Content-based image retrieval has been applied to many different biomedical applications¹. In almost all cases, retrievals involve a single query image of a particular modality and retrieved images are from this same modality. For example, one system may retrieve color images from eye exams, while another retrieves fMRI images of the brain. Yet real patients often have had tests from multiple different modalities, and retrievals based on more than one modality could provide information that single modality searches fail to see. In this paper, we show medical image retrieval for two different single modalities and propose a model for multimodal fusion that will lead to improved capabilities for physicians and biomedical researchers. We describe a graphical user interface for multimodal retrieval that is being tested by real biomedical researchers in several different fields.

Keywords: content-based retrieval, multimodality data, biomedical imaging, fusion

1. INTRODUCTION

The term CBIR (content-based image retrieval) designates the retrieval of images based on their visual similarity to a user-supplied query image or to a set of user-specified image features. CBIR has been, and can be widely used in many areas, such as photograph archives, medical diagnosis, the military, retail catalogs, and remote sensing systems. Common methods for extracting content from images usually make use of color, texture and shape features. There are many CBIR systems that have been developed for use in industry, such as the IBM QBIC system⁷, or in research labs, such as VisualSeek⁸ and Photobook⁹.

CBIR is also useful in medical applications, where it can be used for teaching, research, or as a diagnostic aid. There are a large number of projects on developing CBIR systems for medical research, using melanoma images, pathology slides, CT images and other venues. Our previously-developed systems³ aid biologists and doctors in biomedical research by retrieving similar eye images, similar skull images, or similar fMRI exams. This kind of system may even help doctors with diagnosis in the future. This can lead to new knowledge and eventually better medical techniques.

Patients at today's medical centers encounter a variety of imaging modalities, as more and more tests are accomplished through digital imaging. At various times in life and spanning various different medical problems or just routine examinations, a patient may accumulate many different types of images. For example, our research group has worked with slit-lens images of subjects' eyes, 3D meshes from CT scans of children's skulls, 3D meshes from 6-camera stereo of children's heads, 3D fMRI activation images and 4D raw fMRI images of the human brain, and 3D head and neck images of cancer patients. For each of these applications, each with its own modality, we have developed distance measures that allow a given patient's image data to be compared to a database of image data of the same type from other patients in order to retrieve similar ones. Our retrieval systems can then use these similarity measures plus simple hard constraints, such as "EYE=LEFT" to retrieve patients with similar images. Such systems can be used to help with diagnosis, to look at the effects of treatments, or to fuel research studies that hope to correlate either genetic factors or outcomes with the appearance of the phenotype. Figure 1 shows a user interface from a (mouse) eye retrieval system that we built for researchers at the UW Eye Lab who study the lens of the eye and are looking at how genetic variation affects the development of cataracts. It includes four distance measures related to three different types of features: 1) the ring pattern, 2) the intensity profile of a horizontal line through the center, and 3) the histogram of the western quadrant².

This research was supported by the National Science Foundation under Grant Number DBI-0543631. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

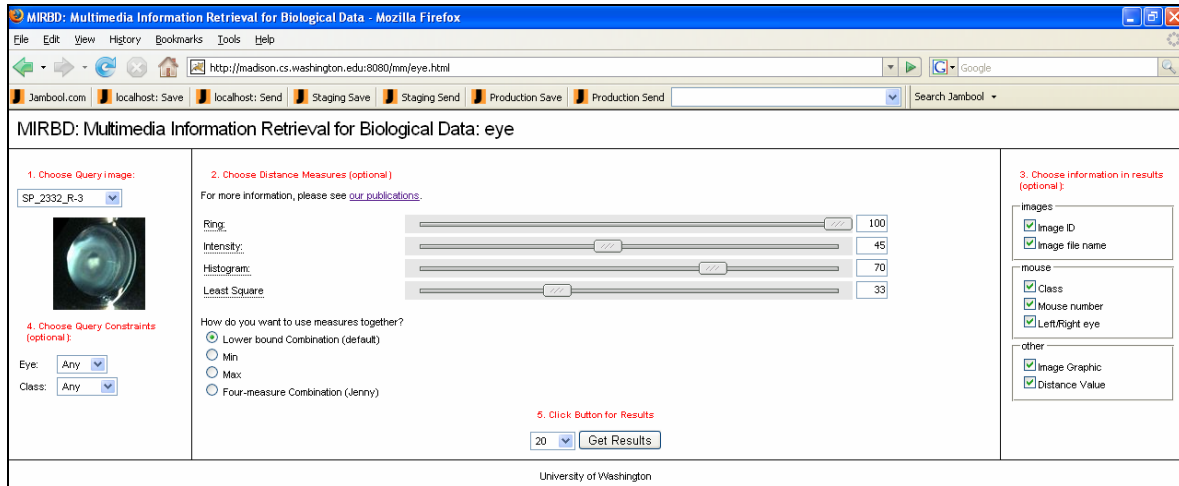


Fig. 1. The graphical user interface for the mouse eye retrieval system. The user has selected a query image, no constraints, a linear combination of the distance measures, and the attributes to be returned.

In this web-based interface, the user selects an eye image, sets constraints as desired for which eye (left or right) and the allowable genetic strains (classes) of mice to be retrieved (SPARC (Secreted Protein Acidic and Rich in Cysteine Knockout), SC1 (Synaptic Cleft Knockout), and WT (Wild Type)), sets weights from 0 to 100 on each of the four distance measures, chooses a method for combining the selected distance measures, and checks the attributes (including images) that should be returned. The methods available for combining the distance measures are 1) weighted linear combinations, 2) weighted minimum, 3) weighted maximum, and 4) concatenation of all (46) attributes of the four distance measures into a single vector with Euclidean distance. In Figure 1, the user has selected an image, chosen no constraints, selected the lower bound combination of the four distance measures, and asked for the attributes ImageID, Class, Mouse Number, Image Filename, Left/Right Eye, Image, and Distance to be returned. Figure 2a shows the first five results for this query, while Figure 2b shows results of a query to a similarly-structured skull retrieval system.

Results, ordered by Distance Measure:

ImageID	Class	Mouse number	Image filename	Left/Right Eye	Image	Distance
3	SPARC	2332	SP129KOf01-18-03#2332R-3.jpg	R		0
33	SPARC	2346	SP129KOf02-01-03#2346R-6.jpg	R		1.258139
1	SPARC	2332	SP129KOf01-18-03#2332R-10.jpg	R		1.719991
8	SPARC	2332	SP129KOf01-18-03#2332R-9.jpg	R		1.725825
329	WT	2380	WT-129 #2380 L28.jpg	L		1.827335

Results, ordered by Distance Measure:

exam_id	class	date	patient	a_plane	f_plane	m_plane	distance
1	metopic	null	1003				0.004989
23	metopic	null	4018				0.016458
3	metopic	null	1008				0.017658
7	sagittal	null	1076				0.027819
2	metopic	null	1006				0.067457

Fig. 2. a) Results of the mouse eye query of Figure 1. b) Results of a skull retrieval query showing 3 planes of skull CT scans.

The mouse eye retrieval system and the skull retrieval system have a simple design similar to the designs of most content-based image retrieval systems in the literature, but they are unusual in that they combine similarity-based retrieval with traditional database retrieval in a single system³. They allow multiple distance measures, but they are limited to single modalities and to a small number of fixed hard constraints.

Given that real patients will have images of their eyes, skulls, brains, and other anatomical entities, a more general retrieval system that can handle multiple types of images and retrieve patients based on similarity over multiple modalities is needed. For this purpose, we have built a prototype multimedia information retrieval system. In this paper we will describe the design of the system, justify the design choices, and give examples of anticipated use.

2. SYSTEM FRAMEWORK

A new patient comes to the hospital with a complaint that must be diagnosed. He/she is given several exams involving images, such as CT, MRI, fMRI, PET or ultrasound. For each modality, the patient may have more than one exam, at different times. The doctor or medical researcher would like to retrieve from the patient database those former patients that have similar exam results to the new patient, and use their information as a reference to help analyze or study the condition of the new patient.

For this scenario, we need a multi-modality system, which can measure the similarity between two patients considering all their exam images. The system needs to be able to measure similarity of multiple images pairs, multiple exams, and multiple modalities. We have developed a multi-modality system that can implement such retrieval tasks. Our system stores in its database all exam data of existing patients. Each time a new query comes in with a set of exam images, it automatically measures the similarity of these images to images in the database, and combines the similarity values at multiple levels. It determines the similarity of the query patient to all patients in the database, ranks them and outputs the results.

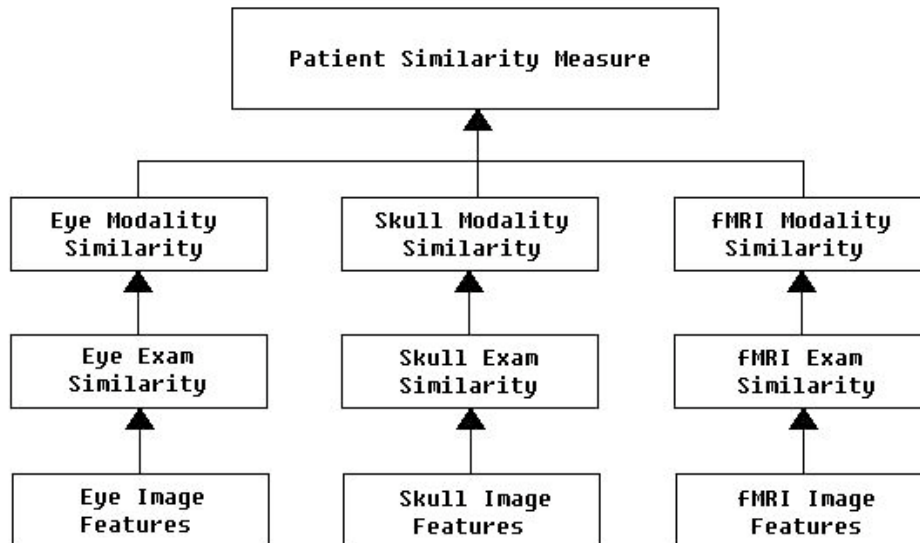


Fig. 3. Structure of the system

Figure 3 is a diagram showing the framework of the system. As is shown in the diagram, the multi-modality system has four levels: combining image feature similarity into image similarity; combining image similarity into exam similarity; combining exam similarity into modality similarity; and combining modality similarity into patient similarity.

At the bottom level, features of the exam image are extracted, and feature similarity between two exam images (coming from two patients) is measured respectively using the different features. Then, this module combines the feature similarity of the two images into one combined image similarity value. At the next level, with image similarity values calculated, similarity of two exams (each consisting of several exam images) is measured by combining the similarity values of each image pair from these two exams. Then at the next level, the system measures similarity of the same

modality of two patients. This modality similarity is calculated by combining similarity values of all exam pairs coming from the same modality of two patients. Last, at the highest level, the similarity of two patients is computed by combining the similarity values of all their modalities, which is calculated in the last level.

3. DATA STORAGE

Our test databases are implemented using the Postgres database system¹⁰. The database consists of two parts. The first part is a relational database with relations for patient information, such as age and gender, and for each of the different imaging studies that can be performed. Figure 4 shows a sample of the relational schemas that such a system might contain. Images are kept outside the database system and are referenced through their file names, which are stored in the relations. The Skull_Exam relation contains file names for both a full CT image of the skull and three 2D slices that were extracted at medically-defined planes and are used by some of our distance measures.

Patient		Skull_Exam		Eye_Exam	
Patient_id	Unique identifier	Exam_id	Unique identifier	Exam_id	Unique identifier
Name	Pseudonym	Patient_id	Ref to patient	Patient_id	Ref to patient
Age	Integer years	Exam_date	date	Exam_dat	date
Gender	M or F	Modality	CT/MRI	Side	R or L
		Image	Path to file	Image	Path to File
		A_plane	Path to File		
		F_plane	Path to File		
		M_plane	Path to File		

Fig. 4. Sample relational schemas for the multimodality system.

The second part includes feature values for all exam images of the existing patients, which is pre-computed and imported into the system before a retrieval task is run. Figure 5 shows a sample of the data file storing feature values for the existing exam image.

Image_id	Unique identifier
Ring	Numeric vector
Intensity	Numeric vector
Histogram	Numeric vector
LeSquare	Numeric vector

Fig. 5. Sample image feature schemas for the multimodality system.

4. SIMILARITY MEASURES

4.1 Image distance measures

In previous work, Yuen et al.² proposed four vector-valued features for comparison of eye images: ring description, intensity profile, histogram of western quadrant of the lens, and segmented least squares fit to the intensity profile. The features were used to discriminate between the different known genotypes of mice whose lenses were being studied. Lin et al.⁶ developed features for comparing the shape similarity of skull images. The features were scaphocephaly severity indices, polar representation, cranial spectrum, and symbolic shape descriptors. Details are given in the referenced papers. These and other biomedical image distance measures motivated our current work and are used to illustrate our multimodality retrieval system.

4.2 The probabilistic method

Similarity measures in our work are calculated under a probabilistic framework. Results from distance measures are converted to similarity values, producing numbers between 0 and 1 that can be thought of as probabilities. The system (at any level) is given n probabilities and n associated weights. At the single instance level, the probabilities come from the values assigned by the different distance measures. At the multiple instance level, the probabilities come from different examinations with the same modality. At the multiple modality level, the probabilities come from the different data modalities.

A straightforward approach to combining two or more probabilities (coming from similarity values) at a lower level into one probability value at a higher level, is the following probabilistic rule:

$$C(p_1, \dots, p_k) = 1 - \prod_{i=1 \dots k} (1 - p_i)$$

in which p_1, \dots, p_k are similarity values from the lower level, and $C(p_1, \dots, p_k)$ is the combined similarity value. However, this general rule to calculate the combined similarity value does not allow the user to specify weights on the probabilities being combined as is common in CBIR systems for computer vision applications. Thus, we included additional factors in order to account for the weights.

4.3 Fagin's algorithm

A scoring rule is an assignment of a value to every tuple. Fagin's algorithm⁵, which was proposed in 2000, is concerned with the issue of how to modify a scoring rule to apply to the case where weights are assigned to the importance of each argument. It provides an explicit formula for incorporating weights that can be applied no matter what the underlying scoring rule is. Here, the underlying scoring rule for the un-weighted case would be our probabilistic formula.

We chose to apply Fagin's weighting formula for fuzzy queries in our work for several reasons. First, when all of the weights are equal, the combination function reduces to the probabilistic framework for the unweighted case: $C(p_1, \dots, p_n) = 1 - \prod_{i=1 \dots k} (1 - p_i)$. Second, if a particular argument has zero weight, then the argument can be dropped without affecting the value of the result. Last, the value of the result is a continuous function of the weights.

4.4 Fagin's algorithm applied to the probabilistic formula

A significant feature of our system is that the user is allowed to assign weights to different part of the similarity measures that are to be combined into one. Based on the probabilistic equation in part 2 and the scoring rule in part 3, we incorporate weights into the probabilistic scoring rules:

Let $P = \{p_1, p_2, \dots, p_n\}$ be the sequence of probabilities and $W = \{w_1, w_2, \dots, w_n\}$ be the corresponding weights, where $w_1 \geq w_2 \geq \dots \geq w_n$. The weighted probabilistic combination function, motivated by the work of Fagin⁵, is defined by

$$C(W, P) = (w_1 - w_2)f(p_1) + 2(w_2 - w_3)f(p_1, p_2) + 3(w_3 - w_4)f(p_1, p_2, p_3) + \dots + n w_n f(p_1, p_2, \dots, p_n),$$

$$\text{where } f(p_1, \dots, p_k) = 1 - \prod_{i=1 \dots k} (1 - p_i).$$

4.5 Image, exam, modality, and patient distances

Using the weighted probabilistic combination function in the last section, the distances for image, exam, modality, and patient are defined with respect to a weight assignment $W = (w_1, \dots, w_n)$ at each level.

The image distance combines all its distance measures d_1, d_2, \dots, d_n :

$$I = (w_1 - w_2) d_1 + 2(w_2 - w_3) (1 - (1 - d_1)(1 - d_2)) + \dots + n w_n (1 - \prod_{i=1 \dots k} (1 - d_i)).$$

For both the eye image and skull image database, $n=4$, and the values for the four image feature distance measures are d_1, d_2, d_3 and d_4 (with weights from the largest to the smallest $w_1 \geq w_2 \geq w_3 \geq w_4$); then image distance I combines the four distance measures:

$$I = (w_1 - w_2) d_1 + 2(w_2 - w_3)(1 - (1 - d_1)(1 - d_2)) + 3(w_3 - w_4) (1 - (1 - d_1)(1 - d_2) (1 - d_3)) + 4 w_4 (1 - (1 - d_1)(1 - d_2) (1 - d_3) (1 - d_4)).$$

The eye exam distance combines distance measures for both the left and right eye images I_1 and I_2 :

$$E_{\text{Eye}} = (w_1 - w_2) I_1 + 2w_2 (1 - (1 - I_1)(1 - I_2)).$$

The skull exam distance combines distance measures for the three image slices that we extract from the full CT exam I_1, I_2, I_3 :

$$E_{\text{Skull}} = (w_1 - w_2) I_1 + 2(w_2 - w_3)(1 - (1 - I_1)(1 - I_2)) + 3w_3 (1 - (1 - I_1)(1 - I_2) (1 - I_3)).$$

The modality distance combines distance measures for all three exams E_1, E_2, E_3 over a single modality:

$$M = (w_1 - w_2) E_1 + 2(w_2 - w_3)(1 - (1 - E_1)(1 - E_2)) + 3w_3 (1 - (1 - E_1)(1 - E_2)(1 - E_3)).$$

The patient distance combines distance measures for all three modalities. Right now, we only have eye and skull data, so the patient distance here combines distance measures for the eye modality and the skull modality M_1 and M_2 :

$$P = (w_1 - w_2) M_1 + 2w_2 (1 - (1 - M_1)(1 - M_2)).$$

Once we have all three modalities M_1, M_2 and M_3 , the combination will become:

$$P = (w_1 - w_2) M_1 + 2(w_2 - w_3)(1 - (1 - M_1)(1 - M_2)) + 3w_3 (1 - (1 - M_1)(1 - M_2) (1 - M_3)).$$

The relationship among image distance (I), exam distance (E), modality distance (M) and patient distance (P) is shown in Figure 6.

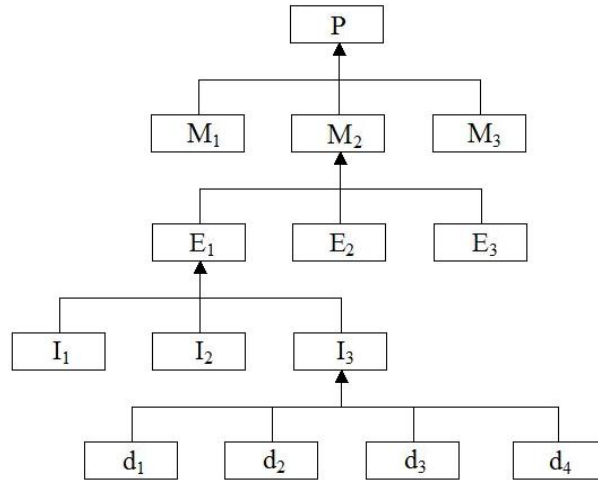


Fig. 6. Image, exam, modality and patient distance

5. IMAGE INDEXING METHOD

Our image retrieval system uses an efficient image indexing scheme⁴ and returns a similarity relation in which the returned images are ordered by similarity to the query. The indexing scheme takes advantage of the triangle inequality to reduce the number of direct comparisons in a threshold style database search. Given a query image, it outputs a value for each database image corresponding to a lower bound on the distance between that image and the query image. The

system then sequences the images in increasing order of their calculated lower bounds. Experimental evidence suggested that the first images in such a sequence are the ones most likely to be the best matches to the query.

In our work, we use this indexing scheme for all image retrieval tasks. Each time that an image is compared to other images in the database, the bounds of their similarity calculated from the triangle inequality are used to take the place of the real distance values.

6. GRAPHICAL USER INTERFACE DESIGN

The user interacts with our system through a sequence of graphical user interfaces. The system initiates with a main GUI to start patient retrieval, as shown in Figure 7. The left part of the main GUI is composed of “Input Information” and “Output Information”. In “Input Information”, a user will first choose a patient ID (ie “1” in the figure), and basic information about the patient ID he/she chooses will show up. In “Output Information”, the user can choose items that he/she wants to view in the retrieval results. The middle part of the main GUI “Image Constraints” shows part of the exam images of the patient, as well as links to other GUI pages which are used to edit weights for different modules, exams, and image features. In the right part of the main GUI “Relational Constraints”, the user can impose some constraints on the patient’s attributes, such as age, gender, etc. The images shown in Fig 7 come from one eye exam and one skull exam of the patient (in this prototype system the eye exams are actually from mouse data). If the user wants to view more exam images of the patient, he/she can click on the button “More Images” to go to the image GUI to see them (Figure 8).

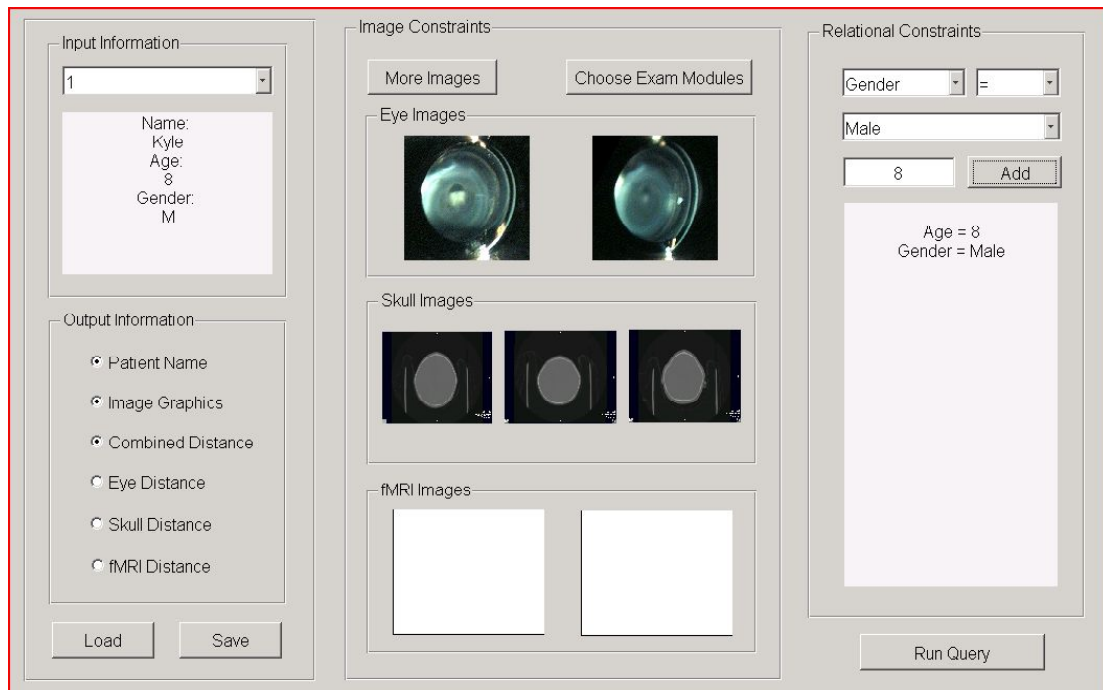


Fig. 7. Main GUI of the system interface after choosing a patient

Next, the weights for all modules, exams, and image features can be edited. By clicking “Choose Exam Modules”, the next GUI page will show up to allow the user to specify weights for different modules, as is shown in Figure 9. In this interface, weights for different image modules can be assigned using the sliders. After a weight combination is specified, the user can click on the “Preview” button to see the order in which patients would be returned under the current weight combination, as illustrated in the figure. The user can click on the circular button below each image pair (an eye image and a skull image) to view more images of the patient.

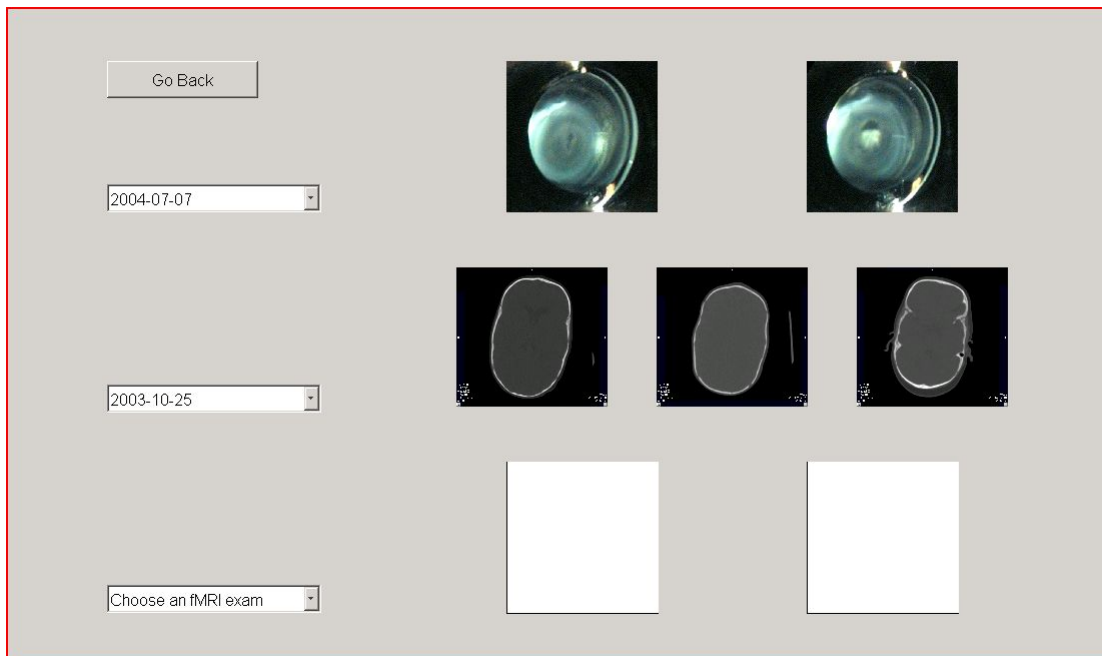


Fig. 8. GUI showing exam images of a patient

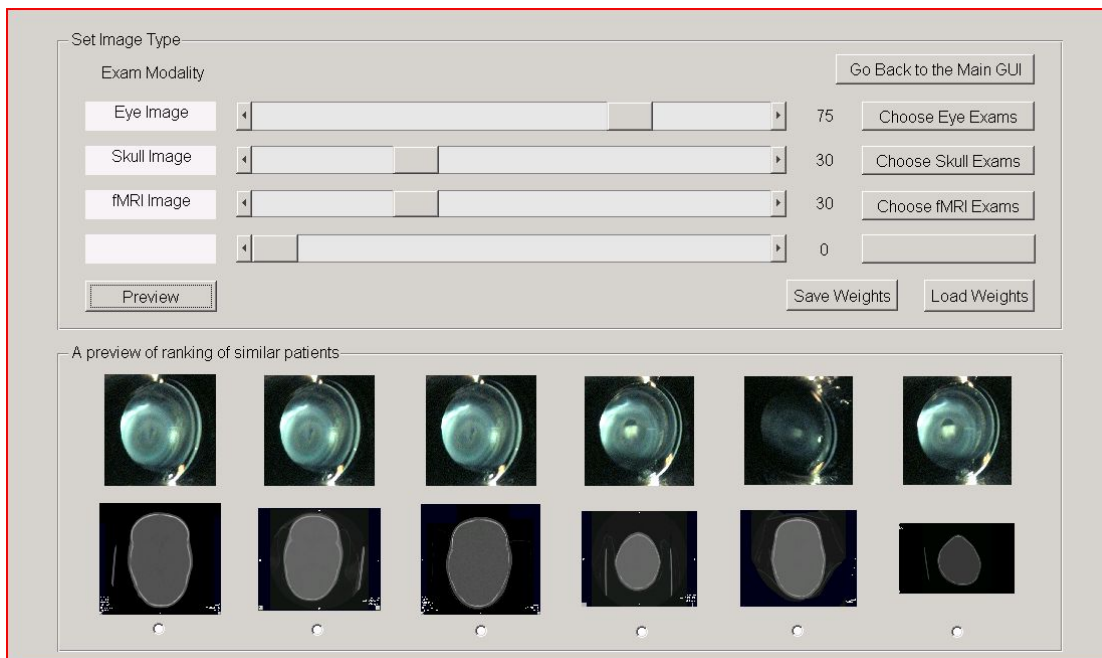


Fig. 9. GUI for setting modality weights after “Preview”

At this point, if the user is satisfied with the weight combination and wants to reserve it for future use, he/she can click on the button “Save Weights” and save it with a specific name, as in Figure 10.

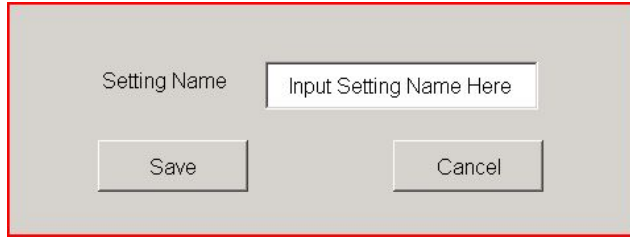


Fig. 10. GUI for saving weight combination

He or she can also load a weight combination that has been previously saved in the database (Figure 11).



Fig. 11. GUI for loading weight combination

After the weights are set in this GUI, the user can choose a modality and proceed to the next level to assign weights for different exams in that modality. For example, when “Choose Eye Exams” is selected, the interface of Figure 12 comes up. As in the previous level, the user can also adjust weights using the sliders, save and load weight combinations, preview on the patient ordering to see one of the exam images of this module of the ordered patients, and click on the circular button below each image to view more images of the patient, as shown in the figure.



Fig. 12. GUI for setting modality weights after “Preview”

By choosing an exam date (one exam within this modality of the query patient), and clicking on the button “Choose Image Features”, the user goes to the last level to edit weights for different image features (Figure 13). At this level, the

user will first choose an image for this exam (such as “left/right eye” for eye images, “A/F/M plane” for skull images), and then edit weights for different image features using the sliders. “Preview”, “Save Weights” and “Load Weights” are also applied here.



Fig. 13. GUI for setting feature weights after “Preview”

Each time that weights are set in a level, the user can either go to the next level, or go back to the previous level. After all the weights are edited, the user can return to the main GUI, and run the query.

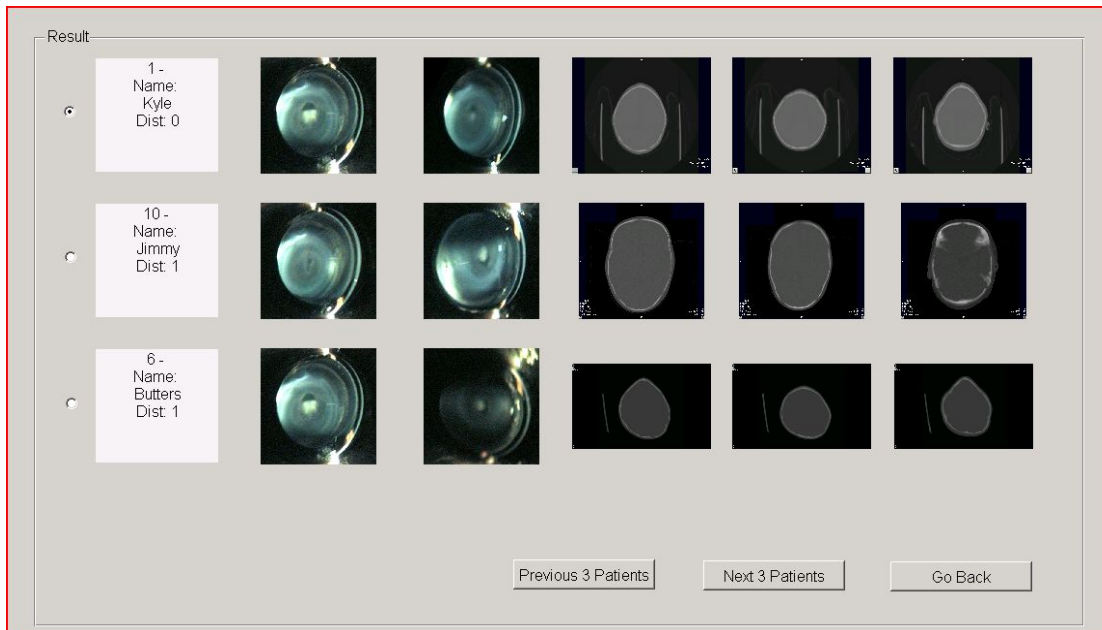


Fig. 14. Resultant GUI showing ranked items

Selected output information for the patients will be shown on the query GUI (Figure 14), and the user can click on the circular button in front of each patient to view more images of the patient. After viewing the results, the user can also go back to adjust the weights to try again.

7. EVALUATION

Our evaluation was done with a group of 5 users. The users were first given instructions on how to use the system and then asked to perform a series of query tasks. After that, they were asked to fill out a questionnaire with questions concerning complexity, efficiency and capabilities of the system.

The tasks consist of queries that require a user to adjust weights in all levels, save and load weight combinations, and try the preview function. Figure 15 shows the set of tasks the users were asked to perform. The questionnaire asks the user to evaluate the system on several aspects: system complexity, system capabilities, task capabilities, and user suggestions. In the system complexity part, we asked the user to rate the complexity of learning to operate the system, ease of use, and ability to construct queries from 1 to 5 (from difficult to easy). According to feedback from the users, our system received an average of 4~4.5 in all the three aspects, which shows that it is a relatively easy to use system.

1. Find patients similar to the query patient using only one eye image of the first eye exam and all eye distance measures with equal weights;
2. Find patients similar to the query patient with a hard constraint added;
3. Find patients similar to the query patient using only one eye image of the first eye exam and all eye distance measures with different weights;
4. Find patient similar to the query patient using both eye images of the first eye exam and all eye distance measures, with different weights;
5. Load and save weight settings in each level while performing query 4;
6. Use the preview to adjust setting of the weights while performing query 4;
7. Find patients similar to the query patient using both eye images of all eye exams and all eye distance measures, with different weights for the exams;
8. Find patient similar to the query patient using only skull images of all skull exams and all skull distance measures, with different weights for the exams;
9. Find patients similar to the query patient using all modalities (both eye images and skull images), with equal weights for different modalities;
10. Find patients similar to the query patient using all modalities (both eye images and skull images), with different weights for all modalities.

Fig. 15. User tasks for evaluation

In the system capabilities part, the users were asked to rate the system speed from 1 to 3 (from slow to fast) and to rate the system reliability from 1 to 3 (from unreliable to reliable). In the task capabilities part, the users were asked to estimate whether the tasks that the system solves would be useful, rating from 1 to 3 in several aspects, such as the “save/load” ability, the “preview” ability, etc. Average scores on these questions were 1.5, indicating a need for some improvement. We note that this was only a pre-evaluation test of our system in which the subjects were computer science graduate students, rather than the biologists, psychologists and doctors for whom the system was designed. Our formal testing will involve both groups.

In the suggestion part, the users gave a lot of suggestions for possible improvements to our system. Most of them were focused on how to make the interface more understandable and easier to use.

8. CONCLUSION

This work is novel in its approach to multimodality fusion in medical image retrieval. It is the first system we know of to retrieve patients based on multiple imaging modalities, multiple instances of each modality, and multiple distance measures for each. The use of Fagin's combination method for weighted probabilities is new to the image retrieval community. Our unified methodology for organization and retrieval of biomedical data from scientific experiments that makes use of both the similarity-based retrieval methodology of CBIR systems and the efficiency of relational database systems is another advantage of the work.

There are still places where the system can be improved. For example, according to the users' feedback, the interface can be more convenient if a user can run the query from any point of the sub-GUIs, rather than having to "go back" to the main GUI to run the query. It would also be better if more illustration is accessible through the interface about what each button means and how to adjust setting. We can also develop methodology to calculate modality similarity between patients with different number of exams, rather than merely with the same number of exams.

In the future, we will also try to use the system for real patient exam images, and we will also try to apply the idea of multimodality fusion in other applications.

REFERENCES

- (1) H. Muller, N. Michous, D. Bandon, A. Gerssbuhler, "A Review of Content-Based Image Retrieval Systems in Medical Applications," *International Journal of Medical Informatics*, Vol. 73, 2004, pp. 1-23.
- (2) J. Yuen, Y. Li, L. G. Shapiro, J. I. Clark, E. Arnett, E. Helene Sage, J. F. Brinkley, "Automated, Computerized, Feature-Based Phenotype Analysis of Slit Lamp Images of the Mouse Lens," *Experimental Eye Research*, 2008.
- (3) I. Atmosukarto, R. S. Travillian, J. D. Franklin, L. G. Shapiro, J. F. Brinkley, D. Suci, J. I. Clark, M. L. Cunningham, "A Unifying Framework for Combining Content-Based Image Retrieval with Relational Database Queries for Biomedical Applications," *Society of Imaging Informatics in Medicine Annual Symposium*, 2008.
- (4) A. Berman and L. G. Shapiro, "A Flexible Image Database System for Content-Based Retrieval," *Computer Vision and Image Understanding*, Vol. 75, Nos. 1-2, 1999, pp. 175-195.
- (5) R. Fagin, "Fuzzy Queries in Multimedia Database Systems," *Proceedings of the ACM Symposium on Principles of Database Systems*, 1998, pp. 1-10.
- (6) H. J. Lin, S. Ruiz-Correa, L. G. Shapiro, M. L. Speltz, M. L. Cunningham and R. W. Sze, "Predicting Neuropsychological Development from Skull Imaging" *IEEE International Conference of the Engineering in Medicine and Biology Society*, 2006.
- (7) M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele and P. Yanker, "Query by Image and Video Content: the QBIC System" *IEEE Computer*, Vol. 28, 1995, pp. 23-32.
- (8) J. R. Smith and S. Chang, "VisualSEEK: a Fully Automated Content-Based Image Query System", *International Multimedia Conference*, 1997, pp. 87-98.
- (9) A. Pentland, R. Picard and S. Sclaroff, "Photobook: Content-Based Manipulation of Image Databases" *International Journal of Computer Vision*, Vol. 18, 1996, pp. 233-254.
- (10) K. Douglas, *PostgreSQL*, Second Edition, Sams, 2005.