Retrieval and Ranking of Biomedical Images using Boosted Haar Features

Chandan K. Reddy and Fahima A. Bhuyan

Abstract— **Retrieving similar images from large repository of heterogeneous biomedical images has been a difficult research task. In this paper, we develop a retrieval system that uses Haar features as its weak classifiers and builds strong training models using the adaboost algorithm. Our system is trained for each image category separately and the final boosted model is stored during the training phase. In the test phase, the most similar images for a given query image are computed using these boosted models. The main advantages of the proposed system are (1) cheap computation of the most relevant features for each image category and (2) fast retrieval of similar images for a given query image. Using performance metrics such as sensitivity and specificity, our results demonstrate the robustness and accuracy of the proposed system.**

I. Introduction

Understanding the medical anatomical structure and extracting features for the retrieval of similar images from large heterogeneous databases has been a challenging research task. For expediting medical and clinical analysis, a new approach to improve the efficiency of content-based medical image retrieval task is developed in this paper. A typical radiology department generates between 100,000 to 10,000,000 images per year [13] and the creation of the large digital image databases increased by the recent advances in PACS (Picture Archival and Communications System) system [15]. A comprehensive survey on this topic about the usage of various imaging techniques and evaluation measures that are appropriate for the retrieval task is reported in [13], [16].

From the biomedical imaging point of view, there are wide range of applications that are being developed in image-producing departments such as Pathology, Hematology and Dermatology etc. In Pathology, most of the work has been done on color changes and texture of microscopic images. In Hematology department, machine learning techniques have been applied for discriminating normal and abnormal blood cells. Although, many of these applications focus on detection and classification tasks, only a few of them have explored the use of advanced machine learning and data mining for image retrieval tasks. A fast retrieval system that can help the medical experts to identify similar images and organize massive collection of images in a systematic manner, will tremendously benefit the biomedical community. Researchers have also developed systems for specific tasks such as retrieving similar pulmonary nodules in Computed tomography images [9].

In general, retrieval systems must be able to retrieve and rank similar images (with respect to a query image) in almost real-time. Also, such a retrieval engine can afford to take relatively higher training time to build models that are specific to individual biomedical image categories. One form of additive modeling, namely the AdaBoost(Adaptive Boosting) has proven to be one of the most efficient off-the-shelf classifier that works fast during the query-time. In this paper, we present a boosting based training algorithm and demonstrate its performance on six different categories of biomedical images. The rest of this paper is organized as follows: Section II gives some relevant background on medical image retrieval problem and Section III gives some basic information about the wavelet (Haar) features which are primary components of our algorithm. Section IV describes the boosting framework for retrieval and ranking of biomedical images. Section V gives the experimental results of our algorithm on real-world biomedical image database. Finally, Section VI concludes our discussion with future research directions.

II. Relevant Background

Image retrieval had been a research topic since early 80s. Some of the most popular commercial image retrieval systems are IBM's QBIC (Query by Image Content) [4] and Virage [1]. A comprehensive review of existing image retrieval systems developed till date are given in [3]. In spite of the extensive research in image retrieval, the field of biomedical image retrieval is still at the stage of infancy. This is mainly due to the fact that biomedical image data is not as profuse as standard image dataset and usually the biomedical images pose some other inherent challenges such as being noisy. In recent years, more developments in the field have resulted in the availability of medical data for public usage and standardization of such medical databases is one of the primary goals in the near future [2]. Our training and testing datasets were collected from IRMA (Image Retrieval in Medical Application) database[1]. IRMA [7], is one of the few frameworks which provide a partial implementation of image retrieval in biomedical applications. This system uniquely classifies biomedical images and allows to test and measure the performance of the classification [10]. We used a subset of this database for training different categories of images and Haar-like features were used to build models for specific categories.

C. K. Reddy and F. A. Bhuyan are with the Department of computer Science at Wayne State University, Detroit, MI - 48202. reddy@cs.wayne.edu

[1] courtesy of TM Deserno, Dept. of Medical Informatics, RWTH Aachen, Germany.

One of the main problems for the retrieval task is to precisely define the representative features that define the visual similarity based on the anatomical structures of different categories of biomedical images. Grey level co-occurrence matrices (GLCM) for extracting second order statistics from an image were used successfully by Haralick [5], [6]. Some of the other popular techniques include signal processing based approaches for texture feature extraction using Gabor wavelet filters [18], [12]. In this paper, we are more interested in reducing the retrieval time taken for a given query image and thus, we applied boosting methodology to retrieve medical images from large-scale databases in a robust and efficient manner. We used Haar features which were originally proposed by Viola and Jones [19] for constructing a face detector. Later, Lienhart et al. introduced novel rotated haar like features [11], [8]. For the task of image retrieval, boosting framework was originally developed by Tieu and Viola [17]. The framework proposed in this paper has two main advantages:

- The system is generic and can be used for a wide range of biomedical applications such as retrieving tumor images (assuming that appropriate training images are provided). This is a task-dependent system and it is not sensitive to any specific set of features used.
- The retrieval time taken is significantly lower compared to other models proposed in the literature.

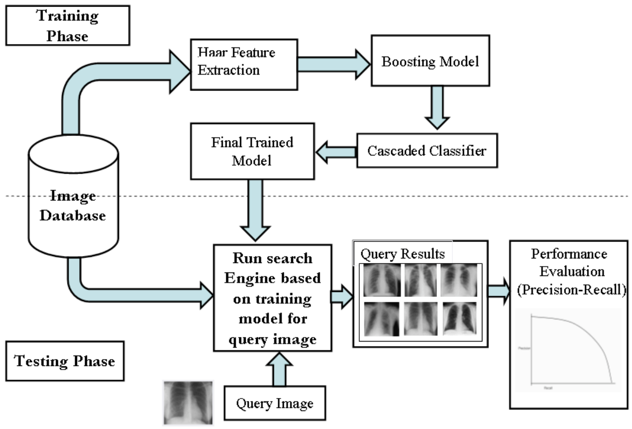


Fig. 1. Block Diagram of a Bio-medical Image Search Engine.

The block diagram of our biomedical image search engine is shown in Fig. 1. A robust training (boosting) algorithm is used to build a trained model that can effectively classify each category of biomedical images. For a given query image during the testing phase, the corresponding trained model contains the most relevant features and computes/displays the similar images to the end-user. Finally, the system is evaluated using standard information retrieval metrics such as sensitivity, specificity and accuracy [20]. Performance of this system significantly depends on the informativeness of each feature. Also, the feature extraction module that we used in our work is robust enough to subtle variations and other affine transformations such as rotation and scaling.

The key steps of building a biomedical image retrieval system developed in this paper are as follows:

1) Efficient extraction of simple wavelet (Haar) features.
2) Application of boosting based training algorithm to model each biomedical image category.
3) Computation of the most similar images (with respect to a query image) from the image repository using the boosted Haar feature values and their weights.

III. Boosted Haar Features

One of the main reasons of using the Haar features in the proposed system is the relatively lower computational time taken to compute these features. This section deals with efficient computation of these Haar features for an image. Compared to other popularly used features such as Gabor features, Haar features are much cheaper to compute. Recent work in the literature has demonstrated that Haar wavelets are powerful image features for object recognition [19]. The two-dimensional Haar decomposition of a square image with n^2 pixels consists of n^2 wavelet coefficients, each of which corresponds to a distinct Haar wavelet. The basic Haar like features was first presented by Viola-Jones and later on enhanced feature set containing rotational features was presented by Lienhart and Maydt [11]. In our system, we used Intel@OpenCV library, which is a library of programming functions mainly aimed at real-time computer vision [14]. The library supports extensive image manipulation functions as well as detailed implementation of Adaboost functionality. For the sake of completion and to make this paper self-contained, we describe in detail the features that are are used in our work. Fig. 2 shows examples of Haar features used in our work. These features were taken from [11] and are given as follows:

1) Four edge features,
2) Eight line features,
3) Two center-surround features.

The computation of number of features for a feature window of size 24X24 is given by Eq. (1) and (2) for all of the prototypes.

$$\mathcal{X}Y.\left(W+1-w\frac{X+1}{2}\right)\left(H+1-h\frac{Y+1}{2}\right) \qquad (1)$$

For rotated features the computations will be [11], [8]:

$$\mathcal{X}Y.\left((W+1)-z\frac{X+1}{2}\right)\left(H+1-z\frac{Y+1}{2}\right) \qquad (2)$$

where z=$w+h$.
Here, $X = \left\lfloor \frac{W}{w} \right\rfloor$ and $Y = \left\lfloor \frac{H}{h} \right\rfloor$, which is the maximum scaling factor in x and y direction.

Using feature prototypes in Fig. 2 and Eqs. (1) and (2), the total number of raw features [11] within 24 X 24 image sub-window can be calculated as shown in Table I. The main computational advantage of using Haar features is obtained by first computing the integral image which is an intermediate representation of the simple rectangular features of an image. If $I(x, y)$ is the original image and the integral image $II(x, y)$ is computed by using the following equation:

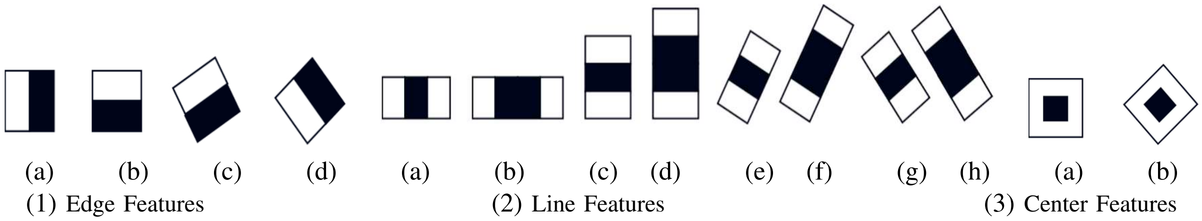


Fig. 2. Feature prototypes of simple haar-like and center-surround features. Black areas are negative and white areas are positive weights. These features were initially proposed in [11].

TABLE I
TOTAL NUMBER OF RAW FEATURES COMPUTED WITHIN A SUB-WINDOW OF SIZE 24 X 24.

Feature Type	w/h	X/Y	Number of Features
$1a : 1b$	2/1 : 1/2	12/24: 24/12	43,200
$1c : 1d$	2/1 : 1/2	8/8	8,464
$2a : 2c$	3/1 : 1/3	8/24: 24/8	27,600
$2b : 2d$	4/1 : 1/4	6/24: 24/6	20,736
$2e : 2g$	3/1 : 1/3	6/6	4,356
$2f : 2h$	4/1 : 1/4	4/4	3,600
$3a$	3/3	8/8	8,464
$3b$	3/3	3/3	1,521
Total			117,941

$$II(x,y) = \sum_{x^{'} \leq x, y^{'} \leq y} I(x^{'}, y^{'}) \tag{3}$$

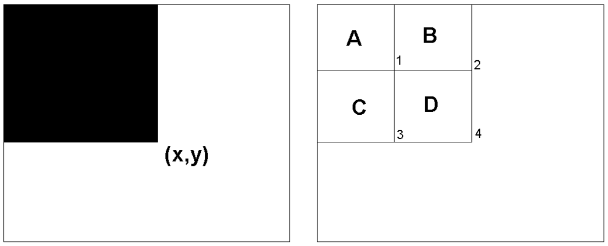


Fig. 3. Integral Image Computation: After integrating the pixel at (x, y), it contains the sum of all pixel values in the shaded rectangle. The sum of the pixel values in rectangle D is $(x_4, y_4) - (x_2, y_2) - (x_3, y_3) + (x_1, y_1)$.

IV. BOOSTING ALGORITHM FOR FAST RETRIEVAL

We will now describe the training algorithm (see Algorithm 1) which is based on the boosting methodology. We train each category of images separately and store the corresponding weights and the weak classifiers for each classifier. For a given query image in the testing phase, the system will identify the class to which it belongs to and then retrieves the top-k ranked images from the image repository based on these weak classifiers and their corresponding weights for a particular category. In the standard Adaboost framework, classification task is performed using several weak classifiers which subsequently develop into stronger models after certain number of iterations. We closely follow the procedure developed in [19], where highly selective features are extracted from the pool of features by minimizing the classification error. The error rate is computed for each feature f_j using the corresponding weak classifier $h_j(x)$ based on a certain threshold value θ_j and polarity p_j (indicating the direction of the inequality sign) for all the training samples (x_1,y_1),(x_2,y_2),..,(x_n,y_n).

Algorithm 1 Boosting for training

```
Input: Given example images ($x_1$,$y_1$),($x_2$,$y_2$),..,($x_n$,$y_n$) where $y_i$=$0, 1$ for negative and positive examples respectively.
Output: Trained Model $(H(x))$
Algorithm:
  • Initialize weights, $D_1(i) = \frac{1}{2m}, \frac{1}{2l}$ for $y_i$=0,1 respectively, where $m$=number of negatives and $l$=number of positives respectively.
  • For round $t = 1, ........., T$
      1) $D_t(i) = \frac{D_t(i)}{\sum_{j=1}^{N} D_t(j)}$
         (so that $D_t$ is a probability distribution)
      2) For each feature, $j$
         $h_t = argmin_{h_j} \epsilon_j = \sum_i D_t(i) \cdot I[y_i \neq h_j(x_i)]$
         where
            $h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) \leq p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$
      3) Set $\alpha_t = \frac{1}{2} log \frac{1-\epsilon_t}{\epsilon_t}$
      4) Update:
         $D_{t+1}(i) = \frac{D_t(i) exp(-\alpha_t y_i h_t(x_i))}{Z_t}$
  • The final strong classifier is:
         $H(x) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$
```

$$h_j(x) = \begin{cases} 1 & \text{if } p_j f_j(x) \leq p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

The feature which gives the lowest error rate amongst all these features is selected corresponding to that weak classifier (See Eq.(4)). A larger weight is associated with better classification functions and a smaller weight with less distinctive functions.

$$h_t = argmin_{h_j} \epsilon_j = \sum_i D_t(i) \cdot I[y_i \neq h_j(x_i)] \tag{4}$$

After building the predictive models for each category, a query image is given to the system and the algorithm will assign a category using the $H(x)$ value. In our implementation, we run the classifiers for each category independently in a random order and the successive classifier is evoked only when its predecessor assigns a negative class label to the given query image. We also return the rank vector for the correctly identified class, which delineates how similar the training images are with respect to the query image using the α values of that particular category. A rank function calculates a complete ranking of the set containing the training images with respect to the query image. The function iteratively computes the similarity of the training images from the query image and orders the training set based on the similarity value, i.e. assigns the most similar image the highest rank using the weights of the selected category. These final results for each image in the particular category are calculated by computing the summation of the product of α and the outcome of each classifier for that particular image.

V. Experimental Results

All our experiments were run in Visual Studio 2005 environment on a Pentium IV 2.8 GHz machine. The evaluation is done using metrics such as sensitivity, specificity and accuracy. The sensitivity measures the proportion of actual positives which are correctly identified, whereas, specificity measures the proportion of negatives which are correctly identified (given in Eq. (5) and Eq. (6)).

$$sensitivity = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (5)$$

$$specificity = \frac{True\ Negatives}{True\ Negatives + False\ Positives} \quad (6)$$

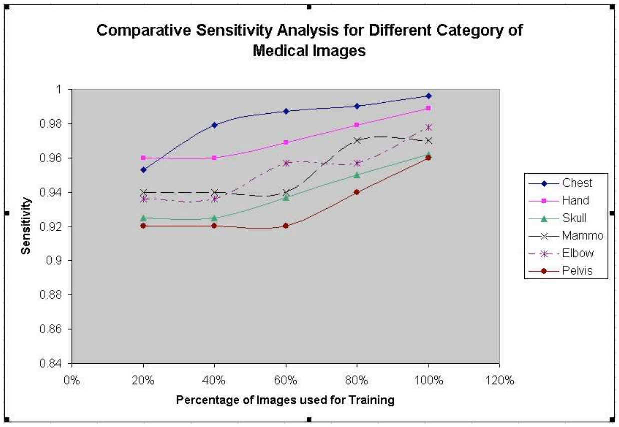


Fig. 5. Sensitivity Analysis of Different Categories of Biomedical Images.

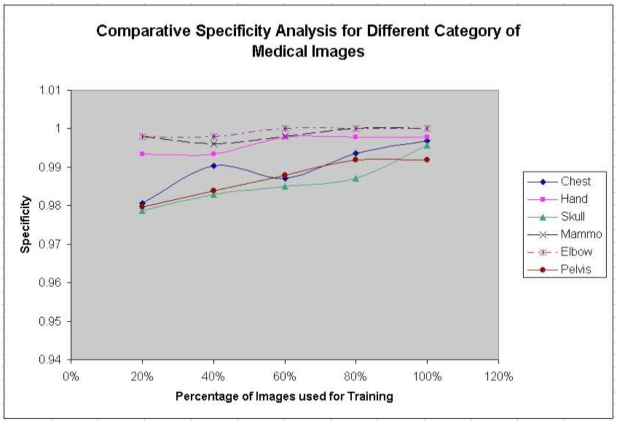


Fig. 6. Specificity Analysis of Different Categories of Biomedical Images.

Our retrieval system is trained using six categories namely chest, hand, skull, mammogram (mammo), elbow and pelvis (see Fig. 4). During the retrieval phase, we run the query images through a series of trained classifiers to acquire the category to which the image belongs to. To demonstrate the performance of our system, for each image category, we performed five-fold cross validation with 80% of images from the repository for training and the rest of the images for testing. We measured the system performance when trained with only partial amounts of training data. The sensitivity and specificity values are computed when only 20%, 40%, 60%, 80% and 100% images were randomly selected and used for training in each category. Fig. 5 shows the sensitivity values for all images in these categories. From this graph, we can observe that as the number of images in the training phase is increased, the true positive rate also increases. This monotonically increasing behavior of the performance function denotes the fact that, as we incorporate more training images into the system, it will become more robust and performs better on test images. From the sensitivity graph, we also observe that on an average 97% of the positive images are classified as positive irrespective of the category. Interestingly, it can be noticed that the chest and the hand categories reached 98% sensitivity after 80% of training images whereas other categories (such as pelvis and skull) had lower sensitivity values. Coincidentally, for these two categories, we need relatively more images to train our system. Hence, the sensitivity analysis will provide some valuable intuition about the complexity of each image category. This might be due to the fact that these categories contain noisy images or images with either homogeneous texture. The specificity graph (see Fig. 6) shows the percentage of total number of negative images that have been correctly classified which is usually around 98%.

TABLE II
Performance comparison of six different biomedical image categories.

Image Category	No. of Features	Training Data size	Test Data size	Accuracy
Chest	63	956	239	0.981
Hand	49	389	98	0.976
Skull	188	319	80	0.954
Mammo	39	132	33	0.97
Elbow	55	188	47	0.972
Pelvis	97	201	51	0.952

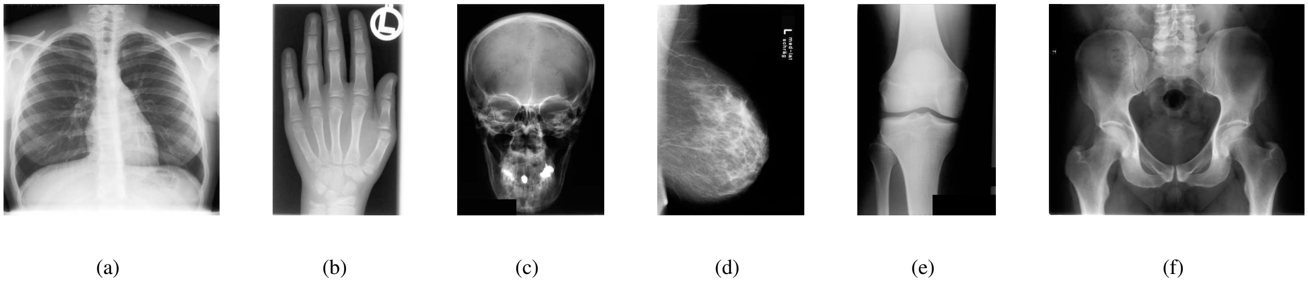

(a) (b) (c) (d) (e) (f)

Fig. 4. Six different Categories of Bio-medical Images used in this paper. (a) Chest (b) Hand (c) Skull (d) Mammogram (e) Elbow (f) Pelvis

Table II gives the performance comparison of six different biomedical image categories. The number of features extracted in each category during the training phase is also reported. This gives an idea about the complexity of the image category that is being considered. For more complicated images (such as skull and pelvis), the number of weak classifiers (or features) required for training is larger. Since, we need to compute only few weak haar features (less than 200 for any given category) during the testing phase (compared to 117,941 features in the training phase), the retrieval task does not consume a lot of time. This is one of the main advantages of using this boosting based framework compared to other traditional approaches for biomedical image retrieval. Accuracy for each category is also reported. One can observe that pelvis and skull image categories have the lowest accuracy rates compared to other categories, whereas the chest images have the best performance compared to the other categories.

When the query image matches with the positive images belonging to a specific class, it displays images from the pool belonging to the retrieved category, with the highest similar value based on our ranking function calculation. In Fig. 7, we show top ranked chest images by using the similarity algorithm. We also validated the classification performance by testing the system with noisy and distorted data and partially clipped images. These kinds of images, if incorporated in the pool of test images, may result in slight decline in the specificity and sensitivity values. Any test image category that has not been incorporated in the training pool prior to the testing phase may also trigger false categorization of the query image. This may happen due to substructure similarity between the query image and the putative category of images that has been identified. Fig. 8 shows an example of noisy and distorted skull query image which retrieved the misclassified pelvis images from the repository of training images. Due to space limitations, we are unable to show the user-interface and other image based query results of our system.

VI. Conclusions and Future Research

In this paper, we developed a novel biomedical image retrieval system that uses boosting method for building prediction models for different categories of biomedical images. Haar-like features were used as weak classifiers for the Adaboost algorithm which automatically selects the most informative features in all the categories. Performance measuressuch as sensitivity and specificity were used to evaluate the efficiency of our system and more insights were provided with respect to the images that are easy/hard to be retrieved. As a continuation of this work, we will analyze the features obtained by our boosting model for each category and identify any correlation that exists between different categories of images. A medical diagnosis system will be developed by combining this image retrieval system the data will be integrated with some other forms of patient information.

Acknowledgments

We would like to thank Dr. Thomas Deserno, RWTH Aachen, Germany for providing the medical image database used in this paper. This work is partially funded by the Wayne State University faculty research award.

References

[1] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. F. Shu. The virage image search engine: An open framework for image management. *Storage and Retrieval for Image and Video Databases IV*, 2670:76–87, 1996.

[2] ImageClefMed Database. http://ir.ohsu.edu/image/.

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

[4] 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*, 28(9):23–32, 1995.

[5] R. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[6] P. Howarth and S. Ruger. Evaluation of texture features for content-based image retrieval. *In Proceedings of the International Conference on Image and Video Retrieval*, pages 326–334, 2004.

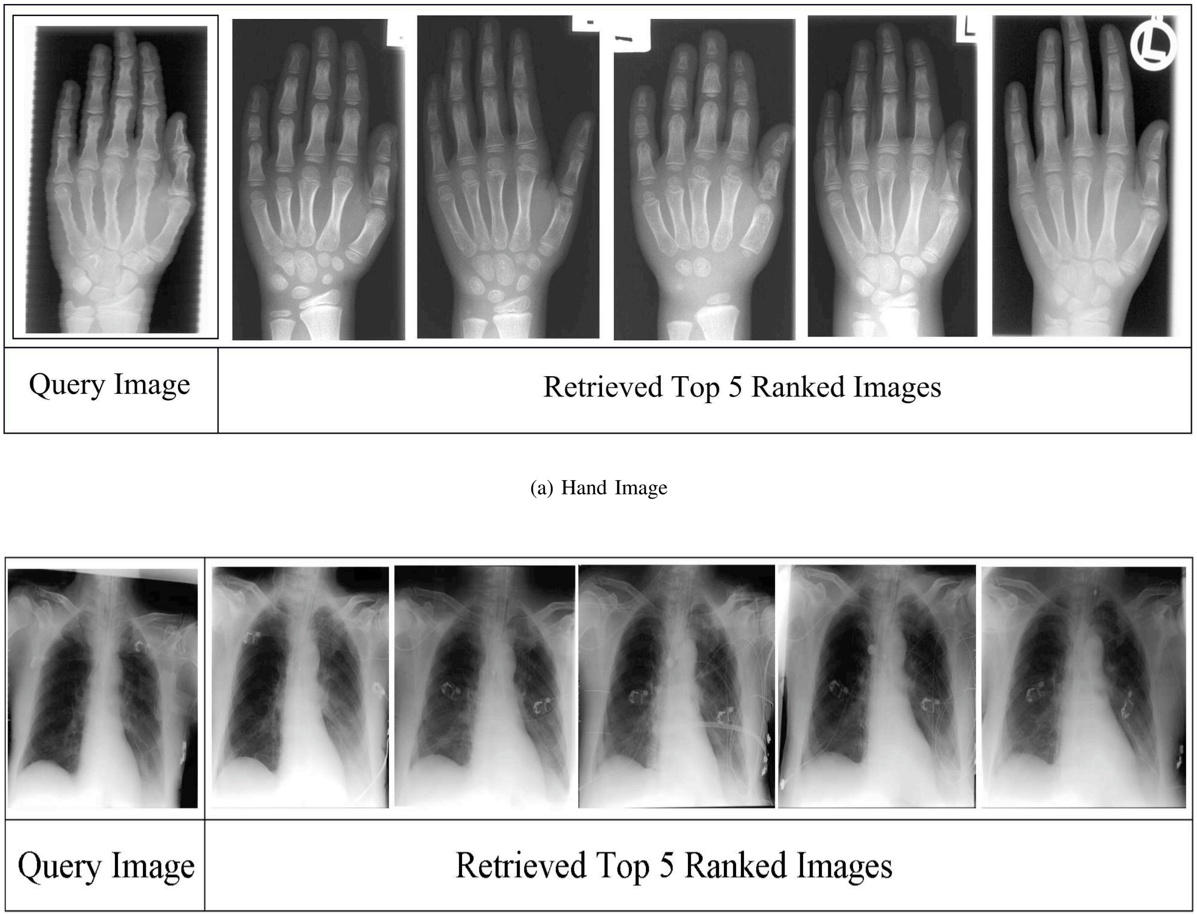
[7] D. Keysers, J. Dahmen, H. Ney, and T. M. Lehmann. A statistical framework for model-based image retrieval in medical applications. *Journal of Electronic Imaging*, 12(1):59–68, 2003.

[8] A. Kuranov, R. Lienhart, and J. Maydt. An empirical analysis of boosting algorithms for rapid objects with as extended set of haar-like features. *Intel Technical Report MRL-TR*, 1, 2002.

[9] M.O. Lam, T. Disney, D.S. Raicu, J. Furst, and D. S. Channin. Brisc - an open source pulmonary nodule image retrieval framework. *Journal of Digital Imaging*, 20(1):63–71, 2007.

[10] T. M. Lehmann, H. Schubert, D. Keysers, and B.B. Wein. The irma code for unique classification of medical images. *In Proceedings of the SPIE Conference on Medical Imaging*, pages 109–117, 2003.

[11] R. Liehart and J. Maydt. An extended set of haar-like features for rapid object detection. *Proceedings of International Conference on Image Processing*, 1:900–903, 2002.



(a) Hand Image

(b) Chest Image

Fig. 7. Results of our system. Images retrieved by our system for a given query image in different categories.

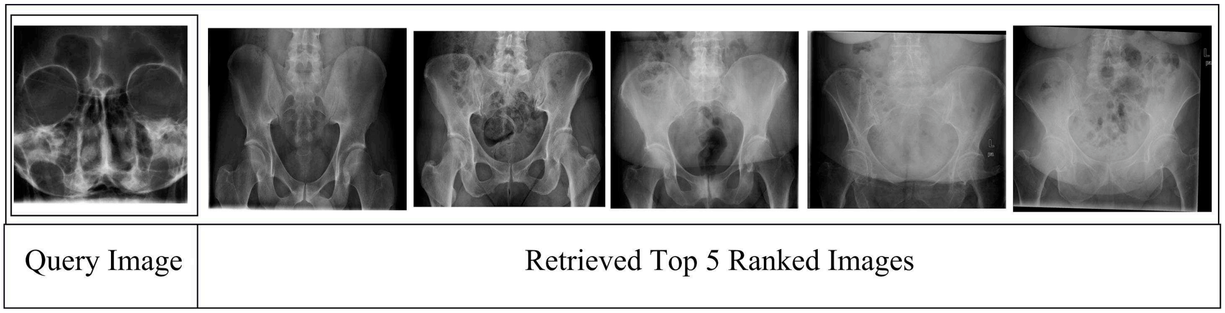


Fig. 8. A noisy query image that retrieved incorrect/irrelevant images.

[12] B. Manjunath and W. Ma. Texture features for browsing and retrieval of image data. *IEEE Tranactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.

[13] H. Muller, N. Michoux, D. Bandon, and A. Geissbuhler. A review of content-based image retrieval systems in medical applications-clinical benefits and future directions. *International Journal of Medical Informatics*, 73(1):1–23, 2004.

[14] Intel Open Source Computer Vision Library (OpenCV). http://www.sourceforge.net/projects/opencvlibrary/.

[15] H. Qi and W.E. Snyder. Content-based image retrieval in pacs. *Journal of Digital Imaging*, 28(2):81–83, 1999.

[16] H. D. Tagare, C. Jaffe, and J. Duncan. Medical image databases: a content-based retrieval approach. *Journal of American Medical Informatics Association*, 4(3):184–198, 1997.

[17] K. Tieu and P. A. Viola. Boosting image retrieval. *International Journal of Computer Vision*, 56(1-2):17–36, 2004.

[18] M. Turner. Texture discrimination by gabor functions. *Biological Cybernetics*, 55(2-3):71–82, 1986.

[19] P. A. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[20] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, second edition edition, 2005.