The customized-queries approach to CBIR

Jennifer G. Dy[a], Carla E. Brodley[b], Avi Kak[c], Chi-Ren Shyu[d], Lynn S. Broderick[e]

School of Electrical and Computer Engineering, Purdue University[a,b,c,d]

Department of Radiology, University of Wisconsin Hospital[e]

ABSTRACT

This paper introduces a new approach called the "customized-queries" approach to content-based image retrieval (CBIR). The customized-queries approach first classifies a query using the features that best differentiate the major classes and then customizes the query to that class by using the features that best distinguish the subclasses within the chosen major class. This research is motivated by the observation that the features that are most effective in discriminating among images from different classes may not be the most effective for retrieval of visually similar images within a class. This occurs for domains in which not all pairs of images within one class have equivalent visual similarity. We apply this approach to content-based retrieval of high-resolution tomographic images of patients with lung disease and show that this approach yields 82.8% retrieval precision. The traditional approach that performs retrieval using a single feature vector yields only 37.9% retrieval precision.

Keywords: Image retrieval, medical images, feature selection, clustering.

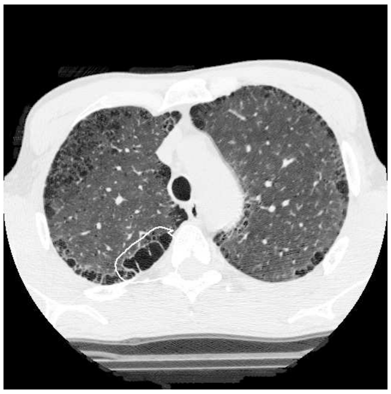
1. INTRODUCTION

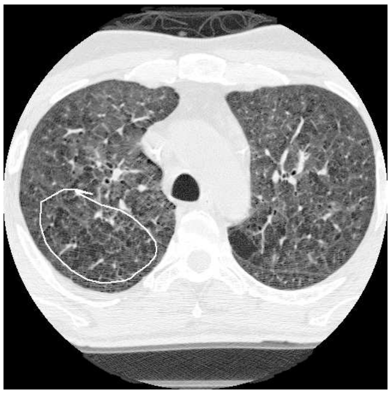
An effective approach to CBIR represents each image in the database by a vector of feature values.[1–4] During retrieval the query image's feature vector is compared to the database vector via the chosen indexing scheme. For such approaches, the choice of features to include in the image characterization is a critical factor in their ability to achieve high retrieval precision. We observed that the features that are most effective in discriminating among images from different classes may not be the most effective for retrieval of visually similar images within a class. This occurs for domains in which not all pairs of images within one class have equivalent visual similarity. For example in the domain of transportation classification, the features that best distinguish airplanes from cars differ from the features that best distinguish commercial jets and stealth fighters. Such domains are appropriate candidates for our approach.

In this paper, we present an automated approach that forms a hierarchical representation of each image for storage and retrieval. Our approach first learns which features best discriminate the images among different classes and then for each class, learns which features retrieve the most visually similar images within that class. In Section 2 we describe a supervised learning procedure to find the features at the first level and an unsupervised learning procedure to find the features that retrieve the most visually similar features within a class. Note that the second step requires unsupervised learning because we are not given subclass labels.

We have applied and evaluated this approach within ASSERT,[5] a CBIR system for medical images. Our database of interest is high-resolution computed tomographic (HRCT) images of the lung. Each image in our database has a disease class label. This enables us to use supervised learning in the first level of our hierarchy. Moreover, we can apply our approach to this domain because this domain reflects the property mentioned earlier, i.e. although a given set of features may be ideal for the disease categorization of a query image, those features may not always retrieve the images that are most similar to the query image. A query image may differ visually from other images within the same disease class on account of the severity of disease and other such factors. Figure 1 illustrates this point. Figure 1a shows an image of a patient with Paraseptal Emphysema (P-Emphysema) and Figures 1 b, c and d show images of three patients that have Centrilobular Emphysema (C-Emphysema). Notice that within the class C-Emphysema, Figure 1d is visually dissimilar to Figures 1b and 1c. A feature that distinguishes P-Emphysema from C-Emphysema is the distance of the "pathology bearing region (PBR)" from the boundary of the lung, whereas the features that best discriminates the images within class C-Emphysema are those that measure the gray level intensity of the PBR. The "pathology bearing region (PBR)"[5] is the region marked by the physician as the region of interest or diseased region. The PBR's are encircled by a white boundary as shown in Figure 1.

Correspondence: Email: *dy,brodley,kak,chiren*@ecn.purdue.edu, *lsbroderick*@facstaff.wisc.edu.





(a) (b)

(c) (d)

Figure 1. *(a) P-Emphysema image. (b) C-Emphysema image in subclass 1. (c) C-Emphysema image in subclass 1. (d) C-Emphysema image in subclass 2.*

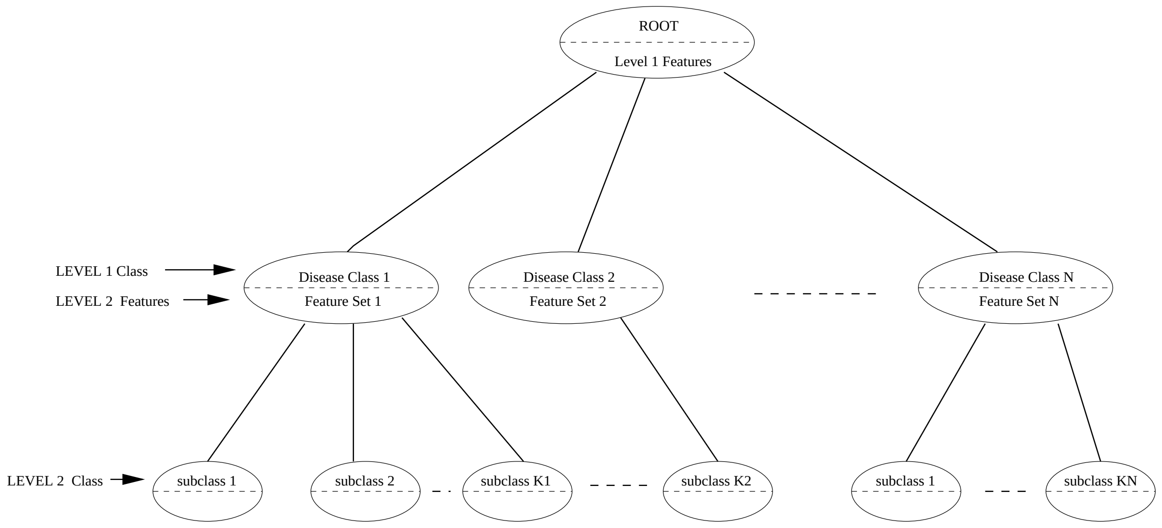


Figure 2. *Customized-Queries Approach.*

Forming a hierarchy of features for retrieval and storage has been explored by other researchers, but their end goals for doing so differ from ours. For example in the FourEyes system,[8] highly structured objects in images, such as buildings and trees, are represented hierarchically to facilitate structural comparisons with a query image. In "Texture features and learning similarity" by Ma and Manjunath,[9] they used a hybrid neural network algorithm to learn similarity by clustering in the texture feature space and then fine tuning the clusters using supervised learning. Their approach builds a hybrid neural network classifier that is applied during retrieval to classify the query as one of the given classes. Then they select the n most similar images within that class cluster using Euclidean distance. Note that the same feature set is used both for classification and for retrieval after classification. Our approach differs in that we do not require the feature sets for classification and retrieval to be the same. Chen and Bouman[6] developed an approach that organizes images in "similarity pyramids" by grouping images with the closest distances together. The organization is used for indexing and browsing purposes. In contrast, we group images according to disease classes and subclasses in order to emulate how expert radiologists would categorize them. Furthermore, we use different feature sets for comparing similarity at each level and for each class. In their case, they used the same feature set and similarity metric throughout the organization of their hierarchy or "pyramid".

We refer to our approach as the *Customized-Queries* approach because it allows the system to "customize" the feature set to the query image. By customization we mean that only those features that retrieve the most similar images within a disease class are used. In other words, our approach transcends retrieval merely on the basis of class identity. After determining class identity of a query image, it goes further and retrieves those images on the basis of visual similarity.

2. THE CUSTOMIZED-QUERIES APPROACH

The customized-queries approach models the database as a hierarchy of classes and the queries as a hierarchy of features (Figure 2). Each node consists of two parts: The set of images in the class (shown as the upper partition of the node) and the set of features for indexing the next level (in the lower partition). Our current implementation consists of two levels but the approach can easily be extended to more levels. The root (Level 0) of the hierarchy contains all the images in our collection and the Level 1 index features here are the features that best differentiate the images according to their major class to identify the query's major class. Each child node (called Level 1) contains images belonging to a single major class and the set of features used to retrieve visually similar images within that class. The classes need not be mutually exclusive, i.e. images can belong to more than one class. Finally, each Level 2 node contains images from a subclass of its parent Level 1 class. Note that these subclasses are not provided, but must be learned. In Section 2.1, we describe the retrieval procedure. In Sections 2.2 and 2.3 we present our approach to finding the Level 1 and Level 2 features.

2.1. The retrieval procedure

Using the Level 1 features, we first classify the image using a one nearest neighbor (1-NN) classifier. We used 1-NN because in a comparison to 2, 3, 4, 5-NN and decision trees, 1-NN yielded the smallest classification error as measured by a ten-fold cross-validation. After classifying our query image, we then use the Level 2 features associated with the image's query class to retrieve the n most similar images as defined by Euclidean distance to the query image. In the next two subsections, we will answer the following questions:

1. How does one determine the discrimination boundaries among classes?

2. Within each class, how does one determine the discrimination boundaries among visually similar subclasses?

3. And, how does one customize the features within each subclass?

2.2. Customizing level 1 features

We treat each level as a separate classification and feature selection problem. On the first level we use the given image classes as our categories. In our domain of interest, this corresponds to the disease pathology assigned to each image. These pathology class labels are confirmed diagnoses obtained from medical records, hence we can consider these as ground truth labels.

To find the Level 1 features, we first extract all features from the query image. We call these the base features, $F_B = \{F_1, F_2, ..., F_N\}$. Then we used Feature Subset Selection (FSS)[10] wrapped around Instance-Based (IB) (a 1-nearest-neighbor classifier) using MLC++[11,12] (we chose the forward direction option) to find the subset of features from F_B that best discriminates the Level 1 classes. Other inducers such as decision trees or neural networks could be used in place of IB. FSS is wrapped around IB because this is the inducer used to classify the query at Level 1. This means that we are using Euclidean distance as our dissimilarity metric and that we are using nearest neighbor to identify which class our query belongs to. FSS is a greedy search algorithm that finds the best set of features for delineating the different classes. FSS wrapped around IB is illustrated in Figure 3. To estimate the classification error, FSS uses ten-fold cross-validation, which randomly partitions the dataset into ten mutually exclusive subsets. Classification error is computed with each partition (or fold) as the test set and the rest as the training set. This is repeated ten times, one for each fold. The final estimate for the classification error is the average of these estimates.[11] For our dataset using the FSS features gave a classification accuracy of $93.33\% \pm 0.70\%$ which was slightly better than using all the features with $92.67\% \pm 0.79\%$ classification accuracy. FSS chose eleven Level 1 features, which is a substantial reduction from using all of 256 possible features. As implied by the FSS algorithm, we need to know the class label of each observation before the best features can be chosen.

2.3. Customizing the level 2 features and clustering the level 2 classes

When categories at a particular level are not available or known, we can use unsupervised clustering to establish the classes. On the first level we used the different disease classes assigned to the images in our database. At the second level we are not given class labels, so we used k-means[14,13,15] to cluster the images within each disease class. Applying k-means requires that we find k, the optimal number of clusters.

In addition to choosing k, we need to determine which features produce the best clusters. Since we cannot tell which features are best before knowing the clusters and vice versa, we used a greedy search algorithm that tries to maximize the interclass distance and minimize the intraclass distance to simultaneously find k and the feature set. Specifically, our algorithm tries to minimize the $trace(S_w)/trace(S_b)$ criterion.[14] S_w is the within-class scatter matrix and S_b is the between class scatter matrix, and they are defined as follows:

$$S_w = \sum_{i=1}^{k} P_i E\{(X - M_i)(X - M_i)^T / \omega_i\} = \sum_{i=1}^{k} P_i \Sigma_i$$

$$S_b = \sum_{i=1}^{k} P_i (M_i - M_o)(M_i - M_o)^T$$

Figure 3. FSS wrapped around IB.

Input: classes, base features, data
Output: set of features

1 $Queue = \{\}$. $Queue$ keeps track of the feature subset at every iteration.

2 $F = F_B$.

3 Find the best feature $f \in F_B$ when used together with the features in $Queue$ gives the smallest classification error.

4 If the old classification error (IB) $-$ new classification error (IB) $> \epsilon$ $(default = 0.001)$,
then enqueue f to $Queue$,
$F = F - f$,
goto 3.
Else
return $Queue$.

$$M_o = E\{X\} = \sum_{i=1}^{k} P_i M_i$$

where P_i is the probability of class ω_i, X is a random data vector, M_i is the mean vector of class ω_i, M_o is the total mean across all data points, ω_i is the class ω_i, Σ_i is the covariance matrix of class ω_i, and $E\{\cdot\}$ is the expected value operator. $trace(S_w)$ measures how scattered the samples are from their cluster means, and $trace(S_b)$ measures how scattered the cluster means are from the total mean.[14] We would like the distance between each pair of samples in a particular class to be as close together as possible and the class means to be as far apart as possible with respect to the similarity metric we are using, i.e. inverse Euclidean distance. Hence, we select k and the n features that minimizes our criterion.

Our algorithm first considers each feature one at a time to assess its ability to cluster the data into k classes using the k-means algorithm. We then sort the features in increasing order based on the value of our criterion for the clusters generated by the feature. Next, we pick the first n features that are not pairwise correlated by more than 50%. We need to check for correlation because many of our base features are highly correlated. We chose n to be the number of Level 1 features chosen by FSS as described in Section 2.2. We then cluster the data using k-means applied to the n selected features and then compute the $trace(S_w)/trace(S_b)$ criterion. We repeat this procedure for k ranging from two to ten. Since the criterion function is a decreasing function with respect to k,[14] we choose k to be the smallest k for which the decrease in criterion value is no longer as significant as the previous decrease. For example Figure 4 shows the $trace(S_w)/trace(S_b)$ criterion value with respect to k for C-Emphysema, resulting in a choice of $k = 5$.

Notice that this method obtains different features and different clusters for each k. Once we establish the clusters obtained by the selected n features and the chosen k, we define each of the k clusters to be a distinct subclass. Using the generated subclass labels, we apply FSS to customize the features to be our Level 2 features for that class. We chose to customize the features rather than retain the features used for clustering to attempt to improve our ability to discriminate among the subclasses and to reduce the number of features required. We ran FSS wrapped around a 1-NN classifier as that is what our retrieval procedure uses at Level 2 to retrieve visually similar images.

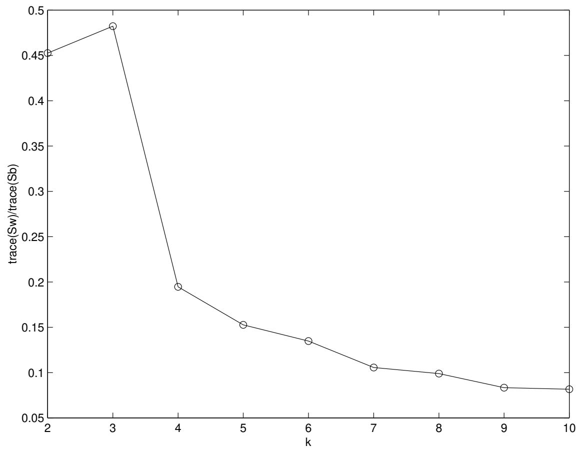


Figure 4. The $trace(S_w)/trace(S_b)$ criterion value with respect to k for C-Emphysema.

3. EXPERIMENTAL RESULTS

Our current database consists of 312 HRCT lung images from 62 patients. These images yield 615 pathology bearing regions (PBR).[5] PBR's are local image regions marked by the physician as pathological. A single image may have several PBR's and these PBR's may have different diagnoses. Throughout the experiment we considered each PBR as a data point, i.e. a single image with three PBR's gives us three data points. We used the 256 implemented features from ASSERT as our base features.[5] These features are measures of geometric properties, histogram, and texture measures of the local pathology bearing regions and of the global image.[5] Table 1 enumerates the base features. Table 2 shows the Level 1 and Level 2 features chosen by the methods described in Section 2. Table 2 does not include diseases with fewer than ten elements. Because the class distribution of diseases in our database is not uniform, we duplicated the data in minority classes before applying FSS in order to avoid biasing the features toward discriminating only the most prevalent class(es) at Level 1. See Table 3 for the original distribution.

Our experiments compare using the Level 1 features for retrieval (the traditional approach) to our customized-queries approach, which first classifies the query image using the Level 1 features and then uses the Level 2 features from the resulting class for the retrieval. Specifically, we compare the results of the following two methods:

Method 1: The traditional approach: Level 1 features are used for retrieval across the entire database using Euclidean distance.

Method 2: The customized queries approach: Level 1 features classify the query image as on of the Level 1 classes. Then retrieve the nearest neighbors within that class as measured by Euclidean distance over the corresponding Level 2 features.

In assessing the performance of Method 2, we assume an ideal classifier to classify the query as a Level 1 class. We did this to isolate the effect of using the appropriate Level 2 features in retrieving the images, i.e. the utility of customizing a query. This assumption is not too limiting since the classification accuracy we obtained from a ten-fold cross-validation applied to a 1-NN classifier using the Level 1 features is $93.33\% \pm 0.70\%$.

To determine which method is best, the radiologist in our team (Dr. Broderick) was asked to evaluate the retrieval results of the two systems. Throughout the test, the radiologist was not informed as to which method produced the retrieved images. We used eighteen images selected randomly from the C-Emphysema class, five from P-Emphysema, and one from each of EG, IPF, Bronchiectasis, Sarcoid and Aspergillus as test query images. We chose eighteen from C-Emphysema because it is the largest class in our collection (51% of our database is of class C-Emphysema). The four images ranked most similar to the query image were retrieved for each method. Note that all images of the query patient are excluded from the search. The user can choose from five responses: strongly-agree (SA), agree (A), not sure (NS), disagree (D) and strongly-disagree (SD) for each retrieved image. To measure the performance of each method, the following scoring system was used: 2 for SA, 1 for A, 0 for NS, -1 for D and -2 for SD. The results are summarized in Table 4.

Table 1. Base features.

Features			Description
Global Features			
F1, F2			gray level mean, standard deviation (std.)
F3			area
F4–F19			histogram
Local Region Features			
F20, F21			centroid (column, row)
F22, F23			area (whole image, lung only)
F24, F25, F26			axis (long, short, orientation)
F27, F28			distance from boundary (Euclidean, Mahalanobis)
F29			closest fissure
F30, F31			deviation from fissure
F32, F33			distance from fissure (Euclidean, Mahalanobis)
F34			orientation from fissure
F165–F167			mean, std., maximum of segmented lung
F168–F175			area histogram of segmented lung
F176–F191			gray histogram of segmented lung
original image	equalized image	gamma corrected $\gamma = 2.2$ image	
F35, F36	F92, F93	F192, F193	mean, standard deviation
F37, F38	F94, F95	F194, F195	mean - global mean, std. - global std.
F39	F96	F196	(mean-global mean)/global std.
F40	F97	F197	(std.-global std.)/global std.
F41–F56	F98–F113	F198–F213	histogram
F57–F61	F114–F118	F214–F218	texture energy at distance 1, 1.414, 2, 2.828 and 3
F62–F66	F119–F123	F219–F223	texture entropy at distance 1, 1.414, 2, 2.828 and 3
F67–F71	F124–F128	F224–F228	texture homogeneity 1 at distance 1, 1.414, 2, 2.828 and 3
F72–F76	F129–F133	F229–F233	texture homogeneity 2 at distance 1, 1.414, 2, 2.828 and 3
F77–F81	F134–F138	F234–F238	texture contrast at distance 1, 1.414, 2, 2.828 and 3
F82–F86	F139–F143	F239–F243	texture correlation at distance 1, 1.414, 2, 2.828 and 3
F87–F91	F144–F148	F244–F248	texture cluster at distance 1, 1.414, 2, 2.828 and 3
F149–156	F157–F164	F249–F256	edge

Table 2. The Level 1 and Level 2 features.

Level 1 Features	F1, F3, F8, F9, F29, F32, F134, F136, F155, F205, F229		
Level 2 Features			
Disease Class	Feature Set	Disease Class	Feature Set
Alveolar Proteinosis	F22, F25, F177	Hemorrhage	F1
Aspergillus	F6	IPF	F72, F167, F220
Bronchiectasis	F20	Panacinar	F44, F106, F19
Bronchiolitis Obliterans	F20	Paraseptal Emphysema	F1, F14, F193, F244
Centrilobular Emphysema	F1, F35, F194	PCP	F20
EG	F1, F167	Sarcoid	F94

Table 3. The disease class distribution.

Disease Class	Frequency	Disease Class	Frequency
Alveolar Proteinosis	17	Hemorrhage	10
Aspergillus	12	IPF	51
Bronchiectasis	14	Panacinar	24
Bronchiolitis Obliterans	12	Paraseptal Emphysema	54
Centrilobular Emphysema	314	PCP	19
EG	57	Sarcoid	16
Aspergilloma	4	Fibrosis	3
Metastatic Calcification	8		

Table 4. Results.

Disease Class	Method 1					Method 2				
	SA	A	NS	D	SD	SA	A	NS	D	SD
C-Emphysema	28	9	5	2	28	58	5	8	0	1
P-Emphysema	2	0	4	0	14	18	0	1	0	1
IPF	5	0	0	0	3	4	0	0	1	3
EG	0	0	0	0	4	0	4	0	0	0
Sarcoid	0	0	0	0	4	0	0	0	0	4
Aspergillus	0	0	0	0	4	4	0	0	0	0
Bronchiectasis	0	0	0	0	4	3	0	0	0	1
total	35	9	9	2	61	87	9	9	1	10

Method 1 obtained a total of -45 points, while Method 2 garnered 162 points. If SA and A are considered as positive retrievals and the rest as negative retrievals, Method 1 obtained 37.93% retrieval precision. Whereas, Method 2 obtained 82.76% precision. Notice that the Method 1 precision is not the same as the accuracy obtained for the Level 1 classifier because there were cases where the radiologist did not mark SA or A even though the PBR's had the same diagnosis as the query. From these results, we can see that customized queries dramatically improve retrieval precision for this domain. This is highlighted by the following example. Figure 5a shows the query image with the pathology bearing region marked. Figure 5b shows the rank 1 image retrieved by Method 1. Figure 5d is the rank 1 image retrieved by Method 2. Both Figures 5b and 5d received a response of "strongly-agree". However, for the rank 2 images, shown in Figures 5c and 5e, Figure 5c produced by Method 1 was evaluated as "disagree" while Figure 5e, by Method 2, was evaluated as "strongly-agree."

Our results show that it is not enough to retrieve images based on just the disease class. In addition, we need to find the best image within that class on the basis of visual similarity. Obtaining better precision is very important in medical image retrieval, because once the best match is found doctors can compare the medical history of the two patients and hopefully be able to use the same diagnosis and treatment. Moreover as shown in Table 2, the feature set that best discriminates the disease classes is different from the feature set that best discriminates between visually similar subclasses within each disease class. Hence, there exists a need for customized queries.

4. CONCLUSION

We introduced a customized-queries approach to CBIR which first classifies a query using the features that best differentiate the Level 1 classes and then customizes the query to that class by using the features that best distinguish the Level 2 classes within the chosen Level 1 class. Our approach was motivated by the observation that the image features that work best to discriminate among different classes are different from the features needed to retrieve the most visually similar images within each class. For the domain of HRCT images of the lung, our results show that the customized-queries approach yields 82.8% retrieval precision, whereas the traditional single feature vector approach yields only 37.9% retrieval precision. Moreover, the traditional approach requires that one finds a set of

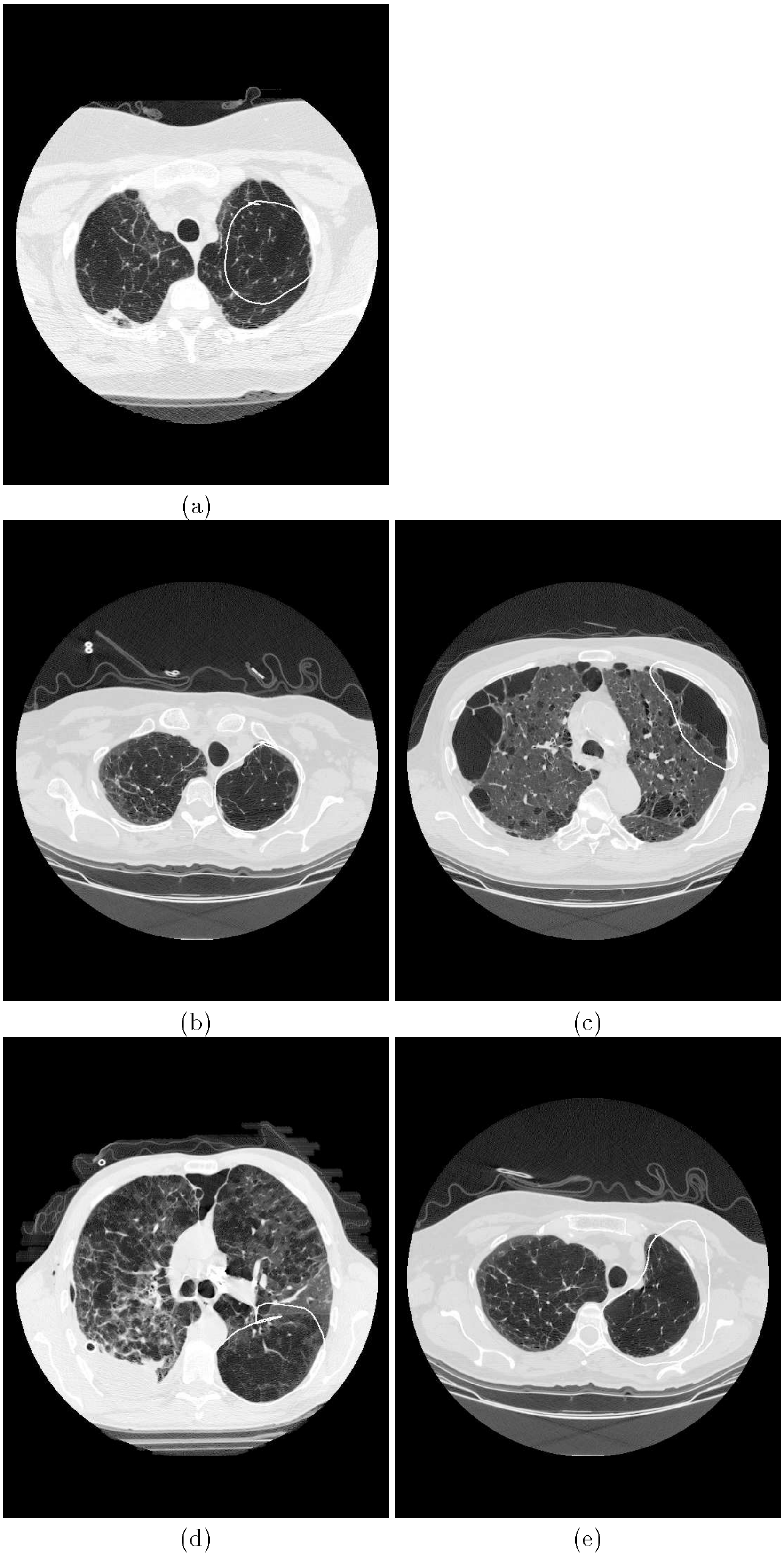


Figure 5. *(a) Query image. (b) Rank 1 image of Method 1. (c) Rank 2 image of Method 1. (d) Rank 1 image of Method 2. (e) Rank 2 image of Method 2.*

features that both correctly classify the query and retrieve visually similar images. Whereas the customized-queries approach divides the problem into two parts. First optimize the features for Level 1 classification, then optimize the features for retrieving visually similar images within that class.

An additional benefit of this approach is that it organizes our database hierarchically into clusters. We noticed that given a collection of objects, humans tend to organize these objects by grouping similar objects together and then organize these groups in different semantic levels. A classic example is biological taxonomy. If we would like to query images by their content, this is a logical representation of the database. For example, in digital image library, we might first group outdoor images and indoor images. Then outdoor images could be further classified as images of beaches, mountains, trees, and so on. Organizing the database this way aides both indexing and browsing.

Our future research will address automatic search for the best Level 1 classifier and find an elegant way to simultaneously solve the clustering and feature selection problem. Our future research will also incorporate indexing and browsing in this approach.

ACKNOWLEDGMENTS

We thank Mark Flick for his helpful comments and Dr. Alex Aisen for helping build our database. This research is supported by the National Science Foundation under Grant No. IRI9711535 , the Showalter Fund, and the National Institute of Health under Grant No. 1 R01 LM06543-01A1.

REFERENCES

1. M. Flickner, H.Sawhney, W. Niblack, J. Ashley, B. Dom, Q. Huang, 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.
2. P. M. Kelly, T. M. Cannon, and D. R. Hush, “Query by image example: The CANDID approach,” *SPIE Vo. 2420 Storage and Retireval for Image and Video Databases III*, pp. 238-248, 1995.
3. L. Taycher, M. Cascia, and S. Sclaroff, “Image digestion and relevance feedback in the image rover WWW search engine,” *Proc. Visual 1997*, San Diego, December 1997.
4. A. Pentland, R. W. Picard, S. Sclaroff, “Photobook: Tools for content-based manipulation of image databases,” *SPIE Storage and Retrieval for Image and Video Databases II*, No. 2185, February 1994.
5. C. R. Shyu, C. E. Brodley, A. C. Kak, A. Kosaka, A. Aisen and L. Broderick, “Local versus global features for content-based image retrieval,” *Proc. IEEE Workshop of Content-Based Access of Image and Video Databases*, pp. 30-34, Santa Barbara, CA, June 1998.
6. J. Chen, C. A. Bouman, and J. C. Dalton, “Similarity pyramids for browsing and organization of large image databases,” *Proceedings of SPIE/IS&T Conference on Human Vision and Electronic Imaging III*, volume 3299, San Jose, CA, January 26-29 1998.
7. Y. Rui, T. S. Huang, S. Mehrotra, and M. Ortega, “A relevance feedback architecture for content-based multimedia information retrieval systems,” *Proc. of IEEE Workshop on Content-based Access of Image and Video Libraries*, in conjunction with *IEEE CVPR '97*, 1997.
8. T. P. Minka and R. W. Picard, “Interactive learning using a 'society of models',” *M.I.T. Media Laboratory Perceptual Computing Section Technical Report No. 349, PAMI Special Issue of Pattern Recognition on Image Database: Classification and Retrieval*, 1995.
9. W. Y. Ma and B. S. Manjunath, “Texture features and learning similarity,” *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 425-430, 1996.
10. R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artificial Intelligence Journal*, Vol. 97, No.s 1-2, pp. 273-324, 1997.
11. R. Kohavi and D. Sommerfield, “Machine learning library in C++,” *SGIMLC++ Utilities 2.0*, http://www.sgi.com/Technology/mlc, October 7, 1996.
12. R. Kohavi and D. Sommerfield, “Data mining using MLC++ a machine learning library in C++,” in the proceedings of *Tools with AI 1996*, pp. 234-245, 1996.
13. T. M. Stough and C. E. Brodley, “Image feature reduction through spoiling: Its application to multiple matched filters for focus of attention,” *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, Newport Beach, CA: AAAI Press, 1997.

14. K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Second Edition, Chap. 10, pp. 441-503, Academic Press, Inc., San Diego, California, 1990.
15. R. J. Schalkoff, *Pattern Recognition: Statistical, Structural and Neural Approaches*, Chap. 5, pp. 114-116, John Wiley & Sons, Inc., 1992.