ISSN:0975-9646
Suresh Kumar D et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (2) , 2012,3378 - 3381

# Novel Dissimilarity Algorithm for Content Based Image Retrieval

Suresh Kumar. D#1,Venkateswarlu..B[2],Mohan Rao. C.P.V.N.J[2]

[1]*MCA Department, Gayatri Vidya Parishad College for Degree & P.G Courses(A),Andhra University, Visakhapatnam, India,*
[2]*CSE Department, Avanthi Institute of Information Technology, JNTU Kakinada, Viskahapatnam,India*

**Abstract—Target search in content-based image retrieval (CBIR) systems refers to finding a specific (target) image such as a particular registered logo or a specific historical photograph. Existing techniques, designed around query refinement based on relevance feedback (RF), suffer from slow convergence, and do not guarantee to find intended targets. To address these limitations,we propose several efficient query point movement methods. We prove that our approach is able to reach any given target image with fewer iterations in the worst and average cases. We propose a new index structure and query processing technique to improve retrieval effectiveness and efficiency. We also consider strategies to minimize the effects of users' inaccurate RF. Extensive experiments in simulated and realistic environments show that our approach significantly reduces the number of required iterations and improves overall retrieval performance. The experimental results also confirm that our approach can always retrieve intended targets even with poor selection of initial query points.**

**Index Terms—Content-based image retrieval, relevance feedback, target search, index structures.**

## INTRODUCTION:

Content-Based Image Retrieval(CBIR) has received much attention in the last decade, which is motivated by the need to efficeintly handle the immensely growing amount of data. Many CBIR systems have been developed, including QBIC[3], Photo book, MARS[4], Netra, Pic Hunter, Blob world, Visual SEEK, simplicity and others. In a typical CBIR system, low-level visual image features(e.g., color texture, and shape) are automatically extracted for image descriptions and indexing purposes. To search for desirable images, a user presents an image as an example of similarity, and the system returns a set of similar images based on extracted features.

In CBIR systems with relevance feedback (RF),[4] a user can mark returned images as positive or negative, which are then fed back into the systems as a new refined query for the next retrieval. The process is repeated until the user is satisfied with the query result. Such systems are effective for many practical CBIR applications. There are two general types of image search: target search and category search. The goal of target search is to find a specific(target) image, such as a registered logo, a historical photograph, or a particular painting. The goal of category search is to retrive a given semantic class or genre of images, such as scenery images or skyscrapers. In other words a user uses target search to find a known image. In contrast, category search is used to find relevant images the user might not be aware ahead of time. We focus on target search in this paper. Two orthogonal issues in CBIR research are efficiency and accuracy. For instance, indexing techniques, may improve the efficiency of the search process. Their retrieval accuracy, however, depends on the effectiveness of the visual features used to characterize the database images.

An effective CBIR system, therefore needs to have both an efficient search mechanism and accurate set of visual features. Addressing the effectiveness of visual features is beyond the scope of this paper. We assume that the Euclidean distances between the images reflect their semantic similarity, and focus on investigating new search techniques to improve the efficiency of target search. Existing target search techniques re-retrieve previously examined images(i.e., those retrieved in the previous iterations) when they again fall within the search range of the current iteration.

In this paper, we propose the dissimilarity algorithm.[1] Again our goals of our target search methods are avoiding local maximum traps, achieving fast convergence, reducing resource requirements, and guaranteeing to find target images. Reconsidering already checked images is one of the several shortcomings of existing techniques that leads to the local maximum trap problem and slow convergence, the idea of finding dissimilarity between the target image and each image of the search space. We can get the dissimilarity value by using the dissimilarity function. From calculated dissimilarity between target image to every image of the search space, By seeing the results of dissimilarity we can easily find out which image is having less dissimilarity with target image, we conclude, that image is the resultant image according to the query passed by the end user. Suppose if we have 100 images in our search space, then any one of the image in that search space can have less dissimilarity when comparing the dissimilarity of the target image to the remaining images. By using this method we can easily find out the image which is close to our target image. Based on the threshold value fixed by the user, he can get that many resultant images. Suppose if the search space contains 100 images, if the user need 6 resultant images for the query posted. Then if he pass his threshold is 6, then he can get first 6 images which are having less dissimilarity in ascending order. Based on these dissimilarity values the user can get the required resultant images as how many he needed.

In this paper the target image is divided into n parts. Each part of that image can contain any object. Suppose if the user knows that what he has to have in the resultant image, he can put all such objects as the attributes of our image dataset.

For ex., the user wants to see the image of sunrise, the user have some attributes of sun, sea, tree, mountains, birds, clouds, people, then these are all the objects(attributes) we should have in our resultant images. So our target image should contain all the above mentioned attributes. Then we are taking our target image with all such attributes. Now we are checking the remaining images, whether they contain these attributes are not. If Image1($I_1$) contains Object1($O_1$) then 'P'(positive) or binary '1' will be stored in that place. Suppose if Image1($I_1$) doesn't contain Object2($O_2$) then 'N'(negative) or binary '0' will be stored in that place in that way for Image1 we are checking the remaining all objects. We continue this process for all the images in our database. Then we get dissimilarity matrix for our images in the current search space.

Reconsidering already checked images is one of the several shortcomings of existing techniques that leads to local maximum trap problem and slow convergence. By using Novel dissimilarity algorithm for CBIR we can find the dissimilarity of all images existing in our search space. So in a single iteration after calculating the dissimilarity of target image to remaining all images, by seeing the dissimilarity score we can give the resultant image which is having less dissimilarity with target image. So by using this algorithm we can easily find more similar image to out target image in a very less iterations.

## METHODOLOGY:

A binary variable has only two states: 0 or 1, where 0 means that the variable is absent, and 1 means that it is present. Treating binary variables as if they are interval-scale can lead to misleading clustering results. Therefore, methods specific to binary data are necessary for computing dissimilarities [4]. We have two approaches for predicting the expressions for any genes:

a) Symmetric dissimilarity

b) Asymmetric dissimilarity

**a) Symmetric dissimilarity:**

This approach involves computing a dissimilarity matrix from the given binary data. is if all binary variables are thought of as having the same weight, we have the 2-by-2 contingency table of table1, where q is the number of variables that equal 1 for both objects I and j, r is the number of variables that equal 1 for object I but that are 0 for object j, s is the number of variables that equal 0 for object I but equal 1 for object j, and t is the number of variables that equal 0 for both objects I and j. The total number of variables is p, where p = q+r+s+t[1].

A binary variable is symmetric if both of its states are equally valuable and carry the same weight; that is, there is no preference on which outcome should be coded as 0 or 1. One such example could be the attribute gender having the states male and female. Dissimilarity that is based on symmetric binary variables is called **Symmetric Binary Dissimilarity.** Its dissimilarity measure, defined in the following equation, can be used to access the dissimilarity between object i and object j.

D(I,j) = (r+s)/(q+r+s+t) ----------(eq-i)

**A Contingency table for binary variable for any of the disease is given as[4]:**

| | | object j | | |
|---|---|---|---|---|
| | | 1 | 0 | sum |
| object i | 1 | q | r | q+r |
| | 0 | s | t | s+t |
| | sum | q+s | r+t | p |

Fig.1 A Contingency Table

**b) Asymmetric dissimilarity:**

Asymmetric binary variables can be given as

D(i,j) = (r+s)/(q+r+s) ----- (eq-ii

Here t is considered unimportant and thus ignored in the computation. Since probability for negative values can be high.[4]From the above Dissimilarity, the Similarity for two persons can be calculated as S(i,j) = 1 – D(i,j) This paper works with asymmetric dissimilarity, because any two persons i,j can have disease caused gene has at most negative. So, Asymmetric dissimilarity is the best possible algorithm comparing with similar dissimilarity. In asymmetric dissimilarity anyone can observe that there is no such't' **(when both object i=0, object j=0).** Assuming that Asymmetric dissimilarity can give better results comparing with Symmetric dissimilarity.

**Example of Dataset:**

TABLE I

EXAMPLE DATA SET

| Persons | Gene1 | Gene2 | Gene3 | Gene4 |
|---|---|---|---|---|
| A | P | N | P | P |
| B | N | P | P | N |
| C | P | P | N | N |
| ….. | …. | …. | …. | .. |

TABLE II

**A List of Notations:**

| Notation | Description |
|---|---|
| P | Positive, Binary value1 |
| N | Negative, Binary value 0 |
| i, j | Two Persons |
| q | When corresponding gene expression for i=1 & j=1 |
| r | When corresponding gene expression for i=1 & j=0 |
| s | When corresponding gene expression for i=0 & j=1 |
| t | When corresponding gene expression for i=0 & j=0 |
| p | Total sum of q,r,s,t i.e., p=q+r+s+t |
| D(i,j) | Dissimilarity of two persons i and j |
| S(i,j) | Similarity of two persons i and j |

**DISSIMILARITY ALGORITHM:**

**Algorithm**: Dissimilarity of Two Persons
**Input**: binary variables 0,1 from DataSet D.
**Output**: q,r,s,t,p,D(i,j)
q←0, r←0, s←0, t←0, $Sum_1$←0, $Sum_2$←0, P
Declare I,J Values
If i=1 and j=1 then q←q+1
Endif
Else
If i=1 and j=0 then r←r+1
Endif
Else
If i=0 and j=1 then s←s+1
Endif
Else
If i=0 and j=0 then t←t+1
Endif
Endif
Break
$Sum_1$←q+s,Sum2←r+t
P=Sum1+Sum2;
D(i,j) = (r+s)/(q+r+s);
End.

For Example, Let us consider multiple genes for multiple persons, which can be tested for dissimilarities among them.

TABLE III
Image Data Set

| Attributes(A) / Images (I) | $A_1$ | $A_2$ | $A_3$ | $A_4$ | $A_5$ | $A_6$ |
|---|---|---|---|---|---|---|
| **I(target)** | P | P | P | P | P | P |
| **$I_1$** | N | P | N | P | N | P |
| **$I_2$** | P | N | P | P | N | P |
| **$I_3$** | N | N | P | P | P | N |
| **$I_4$** | P | P | P | P | N | P |
| **$I_5$** | N | N | P | P | P | N |
| **$I_6$** | N | N | N | P | N | P |

Hence according to the table 1 we can derive the dissimilarities for n number of persons as,
D(I,J)=(r+s)/(q+r+s).(i.e., from eq-ii)

From Table II, it is clear that for two images I and $I_1$
q=3, r=3, s=0
D ($I,I_1$) =(3+0)/(3+3+0)=3/6=0.50(approx)
For Images I and $I_2$
q=4, r=2, s=0
$D(I,I_2)$ = (2+0)/(4+2+0) = 2/6= 0.34(approx)
For Images I and $I_3$
q=3, r=3, s=0
$D(I,I_3)$ = (3+0)/(3+3+0) = 3/6 = 0.50
For Images I and $I_4$
q=5, r=1, s=0
$D(I,I_{4)}$ = (1+0)/(5+1+0) = 1/6 = 0.23(approx)
For Images I and I5
q=3, r=3, s=0
$D(I,I_{5)}$ = (3+0)/(3+3+0) = 3/6 = 0.50
For Images I and $I_6$
q=2, r=4, s=0
$D(I,I_{6)}$ = (4+0)/(2+4+0) = 4/6 = 0.67(approx)

Since Sim(I,J) = 1-D(I,J)
$Sim(I,I_1)$ = 1- 0.50 = 0.50
$Sim(I,I_2)$ = 1 - 0.34 = 0.66
$Sim(I,I_3)$ = 1 – 0.50 = 0.50
$Sim(I,I_4)$ = 1 – 0.23 = 0.77
$Sim(I,I_5)$ = 1 – 0.50 = 0.50
$Sim(I,I_6)$ = 1 – 0.67 = 0.23

From the above calculated results the images I and $I_4$ have the less dissimilarity and more similarity so that $I_4$ is the nearest similar image of our target image.

**CONCLUSION:**

From the above calculated results it is clear that the images I(target image) and $I_4$ is having less dissimilarity, it means that $I_4$ is the most similar image for our target image I. might be extended for cluster analysis. We can use this novel dissimilarity algorithm for calculating dissimilarity between target image and the remaining images in the search space, in addition to that, we can find dissimilarity for each image to each other image in the search space. By using that we can find out which are most similar images in the search space. If we know that information if we pass any kind of query then our system can send all the similar images as resultant images according to the query posted by the user. finally we can conclude that our method can give best suitable, most similar images than the previous methods within fewer iterations. So it reduces the computation time than the previous methods. Because in previous methods we have more iterations and it is somehow difficult to get the target image comparing with our proposed method. Dissimilarity with Symmetric Dissimilarity, Asymmetric Dissimilarity will give better results because, in Symmetric Dissimilarity there is a case of 't'(i.e., both object i=0 & object j=0).For healthy human being always the disease causing gene is negative. For most of the healthy human beings there is a chance of negative so many times. So we need not take't'. So Asymmetric Dissimilarity can easily identify who two get similar disease in future.

**REFERENCES:**

[1] *Data Mining Concepts & Techniques*, Second Edition, Jaiwei Han and Micheline Kamber Textbook Pgno: 389-393.
[2] T. Gevers and A. Smeulders, "Content-Based Image Retrieval: An Overview," Emerging Topics in Computer Vision, G. Medioni and S.B. Kang, eds., Prentice Hall, 2004.
[3] M. Flickner, H.S. Sawhney, 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," Computer, vol. 28, no. 9, pp. 23-32, Sept. 1995.
[4] M. Ortega-Binderberger and S. Mehrotra, "Relevance Feedback Techniques in the MARS Image Retrieval Systems," Multimedia Systems, vol. 9, no. 6, pp. 535-547, 2004.
[5] Y. Rui, T. Huang, M. Ortega, and S. Mehrotra, "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval," IEEE Trans. Circuits and Systems for Video Technology, vol. 8, no. 5, pp. 644-655, 1998.
[6] W.Y. Ma and B. Manjunath, "Netra: A Toolbox for Navigating Large Image Databases," Proc. IEEE Int'l Conf. Image Processing (ICIP '97), pp. 568-571, 1997.
[7] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 24, no. 8, pp. 1026-1038, Aug. 2002.

[8] J.R. Smith and S.-F. Chang, "VisualSEEk: A Fully Automated Content-Based Image Query System," Proc. Fourth ACM Int'l Conf. Multimedia (MULTIMEDIA '96), pp. 87-98, 1996.

[9] J.Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-Sensitive Integrated Matching for Picture Libraries," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no. 9, pp. 947-963, Sept. 2001.

[10] I.J. Cox, M.L. Miller, T.P. Minka, T.V. Papathomas, and P.N. Yianilos, "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical Experiments," IEEE Trans. Image Processing, vol. 9, no. 1, pp. 20-37, 2000.

**EXPECTED RESULT**

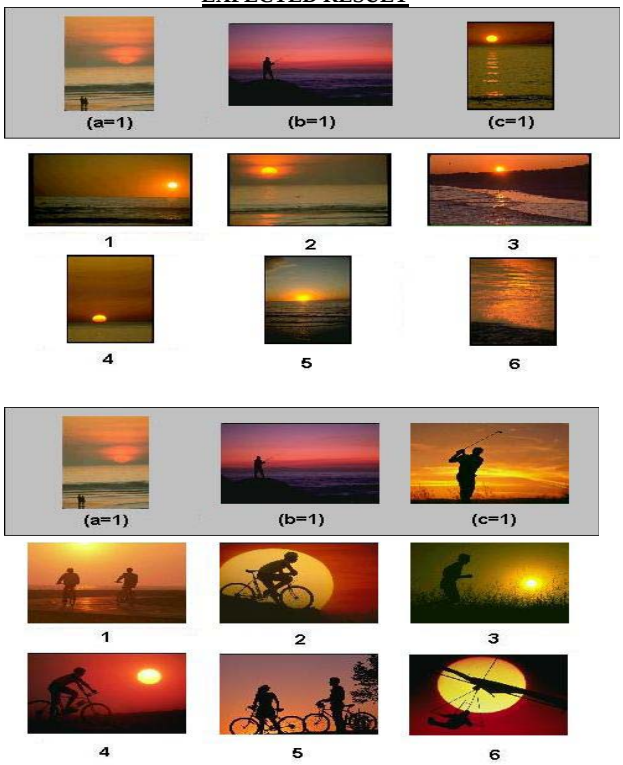


Figure 2: Query 1 and Query 2: Top row contains evidence images followed by the six most relevant images as predicted by the RLN

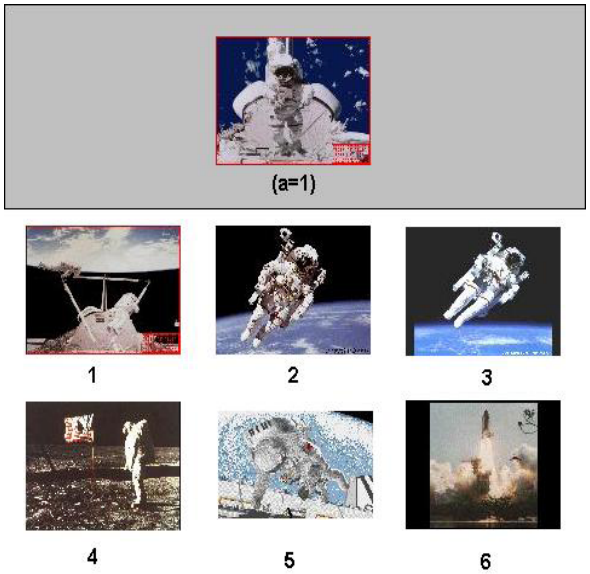


Figure 3: Query 3 and Query 4: Top row contains evidence images followed by the six most relevant images as predicted by the RLN.

**AUTHORS LIST:**

Suresh Kumar .D working as Asst. Professor of MCA Department in Gayatri Vidya Parishad College For Degree & P. G Courses (A), Visakhapatnam. He is Pursuing his M.Tech (CSE) from Avanthi Institute of Engineering & Technology, Narsipatnam, Visakhpatnam(Dist). He is doing his project work on Data Mining. His Areas of Interest are Data Warehousing & Mining, Bio-Informatics, Web Technologies.

Venkateswarlu Bondu received the Master's Dergee in Computer Science and Systems Enigineering fromAndhra University College of Engineering, pursuing Ph.D in Computer Science in Andhra University.
He is an Associate Professor in the Department of Compute Science in Avanthi Institute of Engineering and Technology. His research areas of interests are Software Engineering & Data Modelling. He can be reached at iambondu@gmail.com

**Dr. C.P.V.N.J Mohan Rao** is Professor in the Department of Computer Science and Engineering and Principal of Avanthi Institute of Engineering & Technology - Narsipatnam. He did his PhD from Andhra University and his research interests include Image Processing, Networks, Information security, Data Mining and Software Engineering. He has guided more than 50 M.Tech Projects and currently guiding four research scholars for Ph.D. He received many honors and he has been themember for many expert committees, member of many professional bodies and Resource person for various organizations.