Row-wise DCT Plane Sectorization in CBIR

H.B.Kekre
Sr. professor, Computer Engg.
MPSTME,SVKM's NMIMS
Vile Parle West Mumbai-India

Dhirendra Mishra
Associate. professor, Computer Engg.
MPSTME,SVKM's NMIMS
Vile Parle West Mumbai-India

Shikha Shah, Chirag Thakkar, Rohan Shah
B.Tech
MPSTME,SVKM's NMIMS
Vile Parle West Mumbai-India

ABSTRACT
CBIR(Content Based Image Retrieval System) uses the visual information of an image to give the relevant images as the output. In this paper, we have implemented CBIR by the method of generating Feature Vector using Plane Sectorization. The plane of the image is sectorized in four different ways, namely: 4 sectors, 8 sectors, 12 sectors and 16 sectors. For each of these, feature vector is generated by taking the mean value of coefficients of each sector and by augmenting the zeroth and the highest column component for every plane. Taking the Sectorization is performed on DCT transformed image. The results are compared on the basis of absolute difference and Euclidean distance. The evaluation parameters used are LIRS (Length of initial relevant string of images), LSRR (Length of string to recover all relevant images), Precision and Recall. We have also introduced a new parameter LSRI (Longest string of relevant images retrieved). The database used is Wang database which comprises of 1000 images divided into 10 classes. To compare and evaluate the performance of 4, 8, 12 and 16 DCT sectors, we have considered the overall average of precision and recall. Also, in our earlier works [13], we have applied the algorithm of Feature Vector Generation using DCT plane sectorization on Column-wise transformed plane of images. Here, we are applying the same on Row-wise transformed images and have compared the results of both the methods as well.

Keywords:
PRCP (Precision Recall crossover point).

General Terms
CBIR; LIRS; LSRR; LSRI; Euclidean Distance; Sum of Absolute Difference; Precision and Recall; DCT

1. INTRODUCTION
CBIR [4, 5]:There are large databases of over thousands of images but sometimes a user is interested in may be one particular image or images with some particular feature. This is where CBIR is useful. Content Based Image Retrieval is a technique to retrieve the relevant images by analysing the features of the query image. Various features of the image such as: shape [1], color [1], texture [1], edge density [1], etc. are taken into account. The feature vector of the query image is extracted and is compared to the feature vectors of the images in the database. Thus, using the content of the image and comparing them, relevant images are found. Thus CBIR is the technique to retrieve digital images from a large database [2]. There are various applications of CBIR. For example: fingerprintrecognition [8], iris recognition [9], face recognition [10], etc.

2. DISCRETE COSINE TRANSFORM (DCT)
First apply Discrete Cosine Transform on the image. DCT is made up of cosine functions taken over half the interval and dividing this half interval into N equal parts and sampling each function at the center of these parts [7]. The discrete cosine transform matrix is formed by arranging these sequences row wise. The most common DCT definition of 1D Sequence of Length N is:

$$C(u) = \alpha(u) \sum_{x=0}^{N-1} \left(f(x) \cos\frac{\pi(2x+1)u}{2N} \right)$$

…(1) [7]

For u= 0, 1, 2….N-1

Similarly inverse transform is given as:

$$f(x) = \sum_{u=0}^{N-1} \left(\alpha(u) C(u) \cos\frac{\pi(2x+1)u}{2N} \right)$$

...(2) [7]

For x = 0, 1, 2…N-1

And α(u) for both equations (1) and (2) is defined as :

$\alpha(u) = \sqrt{1/N}$ for u= …(3)[7]

$\alpha(u) = \sqrt{2/N}$ for $u \neq 0$ …(4) [7]

3. FEATURE VECTOR
When we compare the contents of an image, we technically compare its feature vector. The feature vector of all the images in the database is calculated and a feature-vector database is generated. Thus when a query image is passed, its feature vector is calculated which is then compared to the feature vector of every image in the database. The similarity measures used for comparison are sum of Absolute difference[6, 17] and Euclidean distance [6,16,18,19].

3.1 Feature Vector Generation

The feature vector is extracted from the DCT transformed image. To form the feature vector plane we take the combination of co-efficient of consecutive odd and even co-efficient of every column and putting even co-efficient on x axis and odd co-efficient on y axis thus taking these components as coordinates we get a point in x-y plane as shown in figure 1. [7]

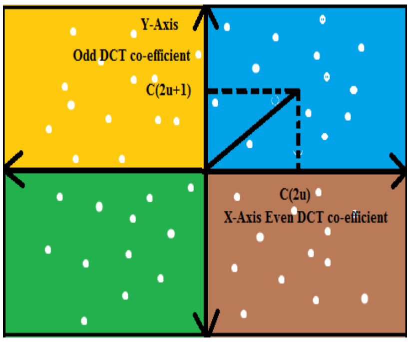

Fig. 1: The DCT Plane used for sectorization [7]

This is the even-odd plane we use for feature vector extraction. Here, we take the mean values to the even-odd components for each sector and then augment the zeroth and highest column components for each sector. We use this method for different sector i.e. 4, 8, 12, and 16 sectors. Also, for each, we do this for each plane i.e. R, G, and B plane.

4. Sectorization

4.1.1 Four DCT Sectors

First we need to plot the points in the even-odd plane. We make use of the following rules:

Table 1. Four DCT Sector Formation [7]

Sign of Even row/column	Sign of Odd row/column	Quadrant Assigned
+	+	I $(0 – 90^0)$
+	-	II $(90^0 – 180^0)$
-	-	III$(180^0- 270^0)$
-	+	IV$(270^0–360^0)$

The even and the odd column components are checked for their signs and are plotted accordingly in either of the four sectors.

Here, the feature vector extracted will be of component size 18. This is because we take the average of even-odd components and augment the average of the zeroth and the highest column component for each sector [3,7]. Since there are 4 sectors and we perform this for all the planes R, G and B, the feature vector will be of component size: (((1*4) + 2))*3=18.

4.1.2 Eight DCT Sectors

The basic method used is the same as Four DCT Sectors but here, we divide each sector obtained in the Four DCT sectors into two equal parts of 45^0 each. Thus, we get Eight DCT Sectors.

Here, the feature vector extracted will be of component size 30. This is because we take the average of even-odd components and augment the average of the zeroth and the highest column component for each sector [3,7]. Since there are 8 sectors and we perform this for all the planes R, G and B, the feature vector will be of component size: (((1*8) + 2))*3=30.

4.1.3 Twelve DCT Sectors

The basic method used is the same as Four DCT Sectors but here, we divide each sector obtained in the Four DCT sectors into three equal parts of 30^0 each. Thus, we get Twelve DCT Sectors.

Here, the feature vector extracted will be of component size 42. This is because we take the average of even-odd components and augment the average of the zeroth and the highest column component for each sector [3,7]. Since there are 12 sectors and we perform this for all the planes R, G and B, the feature vector will be of component size: (((1*12) + 2))*3=42.

4.1.4 Sixteen DCT Sectors

For dividing the plane in Sixteen DCT sectors, we divide each of the eight sectors in two equal parts.

Here, the feature vector extracted will be of component size 54. This is because we take the average of even-odd components and augment the average of the zeroth and the highest column component for each sector [3,7]. Since there are 4 sectors and we perform this for all the planes R, G and B, the feature vector will be of component size: (((1*16) + 2))*3=54.

5. PARAMETERS USED

5.1 Precision

It is the ratio of number of relevant images retrieved to the total number of images retrieved.

$$Precision = \frac{\text{Number of Relevant Images Retrieved}}{\text{Total Number of Images Retrieved}}$$

…(5) [3,14]

5.2 Recall

It is the ratio of number of relevant images retrieved to the total number of relevant images in database.

$$Recall = \frac{\text{Number of Relevant Images Retrieved}}{\text{Total Number of Relevant Images in Database}}$$

…(6) [3,14]

5.3 LIRS

It is the ratio of length of initial relevant string of images to the total number of relevant images retrieved.

$$LIRS = \frac{\text{Length of Initial Relevant String of Images}}{\text{Total Number of Relevant Images Retrieved}}$$

…(7) [3,14]

5.4 LSRR

It is the ratio of length of string to recover all relevant images to the total number of images in the database.

$$LSRR = \frac{\text{Length of String to Recover all Relevant Images}}{\text{Total Number of Images in the Database}}$$

…(8)[3, 14]

5.5 LSRI

It is the ratio of length of the longest string of relevant images retrieved to the total number of relevant images in the Database [13].

$$LSRI = \frac{\text{Longest string of relevant images retrieved}}{\text{Total Number of relevant Images in the Database}}$$

…(9) [13]

6. RESULTS AND DISCUSSIONS

The database used is Wang database [11, 12]. 1000 images have been classified into 10 classes having 100 images of each type. Classes include images of Tribal, Beaches, Monuments, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountains, and Food Dishes.

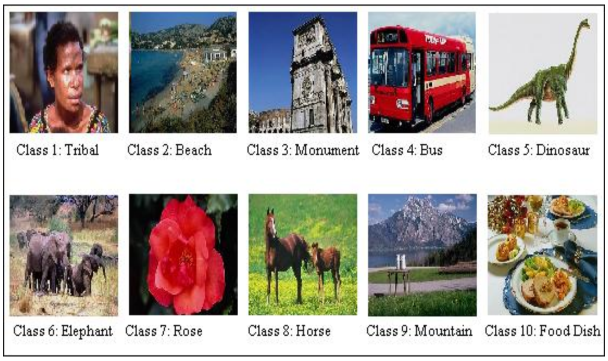

Figure 2.Sample images from Wang database.

6.1 Precision-Recall

The average of the Precision-Recall crossover point is considered for the purpose of generating the plot. Euclidean Distance and Sum of Absolute Difference are used as similarity measures.

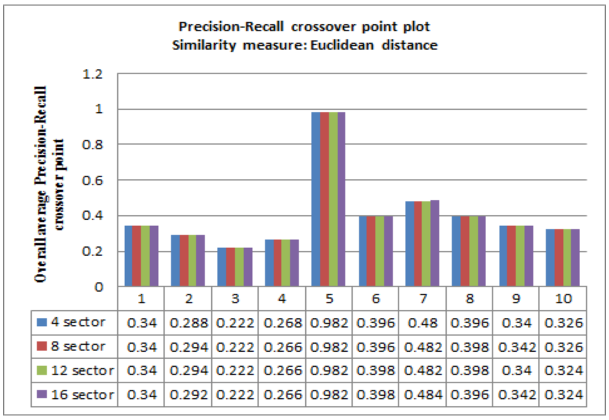

Figure 3. Plot for Precision-Recall Crossover Point using Euclidean distance.

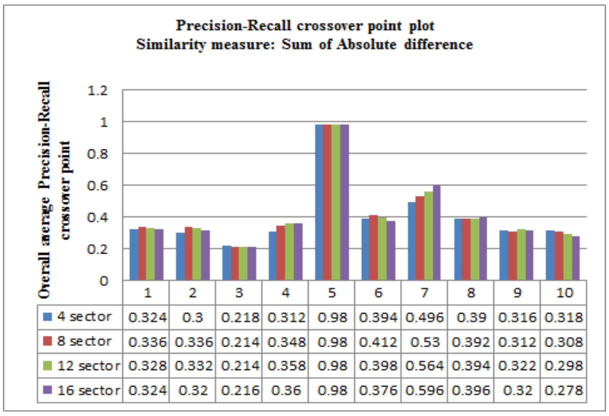

Figure 4. Plot for Precision-Recall Crossover Point using Sum of Absolute difference.

From Figure 3 and 4 we infer that we get better results when we use sum of Absolute difference as the similarity measure as compared to Euclidean distance. But contrary to the overall performance classes 6, 8, 9 and 10 perform marginally better when Euclidean Distance is used as the similarity measure. The best performance is that of class 5 using both the similarity measures.

We can represent the performance of these evaluation parameters in the form of percentage for better understanding.

The top three class-wise performances for the evaluation parameter Precision-Recall with respect to the similarity measure Euclidean Distance are class 5 with the best performance of 98.2% followed by class 7 with the performance of 48.2% followed by class 6 and 8 with the same performance of 39.7%.The top three class-wise performances for the evaluation parameter Precision-Recall with respect to the similarity measure sum of Absolute difference are class 5 with the best performance of 98% followed by class 7 with the performance of 54.65% followed by class 6 with the performance of 39.5%.

On evaluation of the performance of Precision-Recall crossover point for similarity measure Euclidean distance we see that the retrieval rate for: 4 sectors: 40.38%, 8 sectors: 40.48%, 12 sectors: 40.46%, 16 sectors: 40.46 %.Similarly, on evaluation of the performance of Precision-Recall crossover point for similarity measure sum of Absolute difference we see that the retrieval rate for:4 sectors: 40.48 %, 8 sectors: 41.68%, 12 sectors: 41.88%, 16 sectors: 41.66%.

6.2 LIRS

The performance of the evaluation parameter LIRS is shown in Figure 5 and Figure 6 for both the similarity measures Euclidean Distance and Sum of Absolute Difference. Performance of LIRS is compared for all different sectors i.e. 4, 8, 12 & 16 sectors.

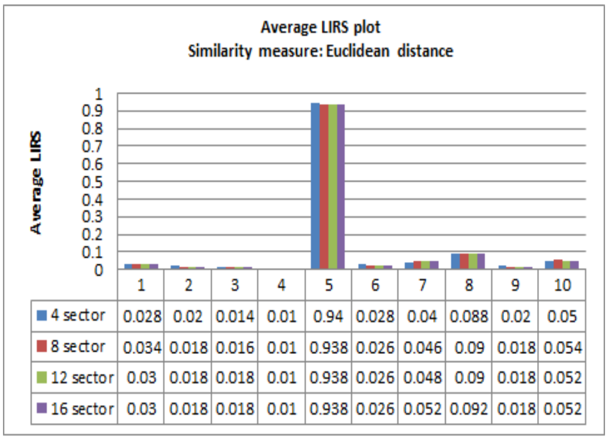

Figure 5. Plot for Average LIRS using Euclidean distance.

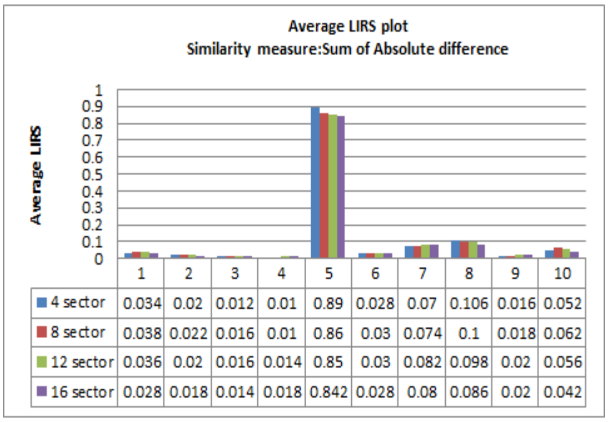

Figure 6. Plot for average LIRS using Sum of Absolute Difference.

From Figure 5 and 6 we infer that we get better results when we use Euclidean distance as the similarity measure as compared to sum of Absolute difference. The best performance is that of class 5 using both the similarity measures.

The top three class-wise performances for the evaluation parameter LIRS with respect to the similarity measure Euclidean Distance are class 5 with the best performance of 93.85% followed by class 8 with the performance of 9% followed by class 10 with the performance of 5.2%. The top three class-wise performances for the evaluation parameter LIRS with respect to the similarity measure sum of Absolute difference are class 5 with the best performance of 86.05% followed by class 8 with the performance of 9.75% followed by class 7 with the performance of 7.65%.

On evaluation of the performance of LIRS for similarity measure Euclidean distance we see that the retrieval rate for: 4 sectors: 12.38%, 8 sectors: 12.5%, 12 sectors: 12.48%, 16 sectors: 12.54%. Similarly, on evaluation of the performance of LIRS for similarity measure sum of Absolute difference we see that the retrieval rate for: 4 sectors: 12.38%, 8 sectors: 12.3%, 12 sectors: 12.22%, 16 sectors: 11.76.

6.3 LSRR

The performance of the evaluation parameter LSRR is shown in Figure 7 and Figure 8 for both the similarity measures Euclidean Distance and Sum of Absolute Difference. Performance of LSRR is compared for all different sectors i.e. 4, 8, 12 & 16 sectors.

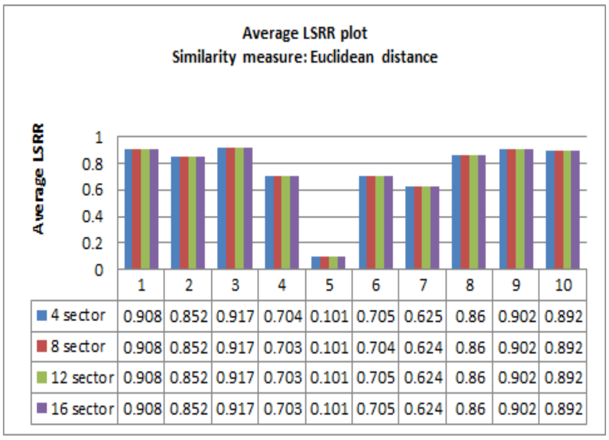

Figure 7. Plot for average LSRR using Euclidean distance.

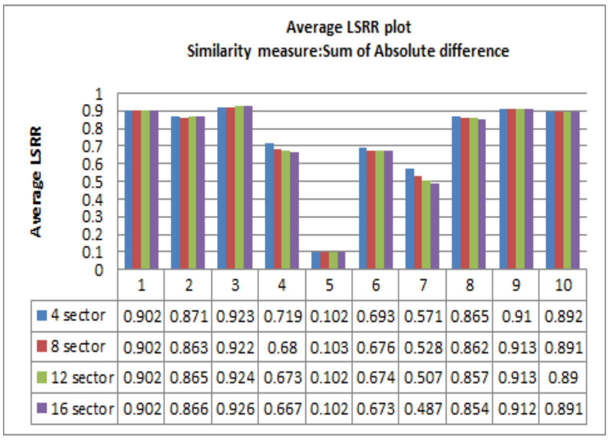

Figure 8. Plot for average LSRR using Sum of Absolute Difference.

From Figure 7 and 8 we infer that we get better results when we use sum of Absolute differenceas the similarity measure as compared to Euclidean distance.But contrary to the overall performance classes 2, 3, 5 and 9 perform marginally better when Euclidean distance is used as the similarity measure. The best performance is that of class 5 using both the similarity measures.

The top three class-wise performances for the evaluation parameter LSRR with respect to the similarity measure Euclidean Distance are class 5 with the best performance of 10.1% followed by class 7 with the performance of

62.43%followed by class 4 with the performance of 70.31% . The top three class-wise performances for the evaluation parameter LSRR with respect to the similarity measure sum of Absolute difference are class 5 with the best performance of 10.24% followed by class 7 with the performance of 52.35% followed by class 6 with the performance of 67.91%.

On evaluation of the performance of LSRR for similarity measure Euclidean distance we see that the retrieval rate for: 4 sectors: 74.67 %, 8 sectors: 74.65 %, 12 sectors: 74.65 %, 16 sectors: 74.65%. Similarly, on evaluation of the performance of LSRR for similarity measure sum of Absolute difference we see that the retrieval rate for: 4 sectors: 74.48 %, 8 sectors: 73.4%, 12 sectors: 73.08%, 16 sectors: 72.81 %.

6.4 LSRI

The performance of the evaluation parameter LSRI is shown in Figure 9 and Figure 10 for both the similarity measures Euclidean Distance and Sum of Absolute Difference. Performance of LSRI is compared for all different sectors i.e. 4, 8, 12 & 16 sectors.

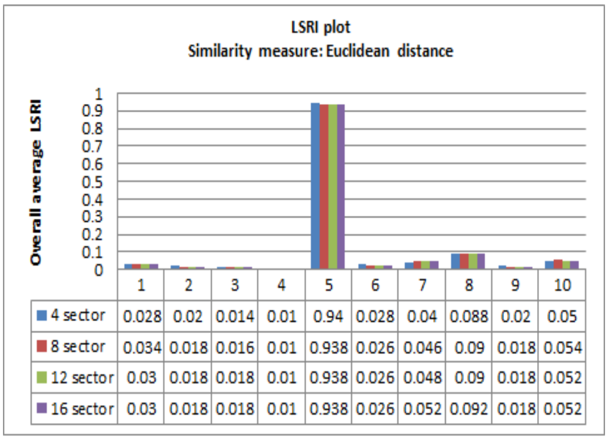

Figure 9. Plot for average LSRI using Euclidean distance.

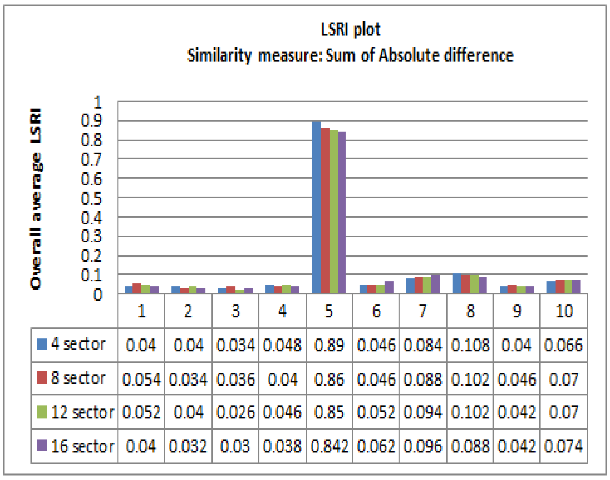

Figure 10 Plot for average LSRI using Sum of Absolute Difference.

Here, as we see, class 5 gives the best output for LSRI which indicates that it retrieves significantly long string of relevant images compared to the other classes. We get better results when we use sum of Absolute difference as the similarity measure as compared to Euclidean distance.

The top three class-wise performances for the evaluation parameter LSRI with respect to the similarity measure Euclidean Distance are class 5 with the best performance of 93.85% followed by class 8 with the performance of 9.1% followed by class 10 with the performance of 6.25% . The top three class-wise performances for the evaluation parameter LSRI with respect to the similarity measure sum of Absolute difference are class 5 with the best performance of 86.05% followed by class 8 with the performance of 10% followed by class 7 with the performance of 9.05%.

On evaluation of the performance of LSRI for similarity measure Euclidean distance we see that the retrieval rate for: 4 sectors: 12.38%, 8 sectors: 14.04%, 12 sectors: 14%, 16 sectors: 14.08%. Similarly, on evaluation of the performance of LSRI for similarity measure sum of Absolute difference we see that the retrieval rate for: 4 sectors: 13.96 %, 8 sectors: 13.76%, 12 sectors: 13.74%, 16 sectors: 13.44%.

6.5 Comparison

Now we compare these results with the results of our earlier work on Column-wise DCT plane sectorization in CBIR for both Euclidean distance and sum of Absolute difference.
Here, we have rounded the percentage of our results for better understanding and comparison.

Table 2: Comparison of Column-wise and Row-wise using Euclidean Distance

		Parameter			
	Sectors	PRCP	LIRS	LSRR	LSRI
Column-wise	4	44%	12%	75%	14%
	8	40%	12%	75%	14%
	12	40%	12%	75%	14%
	16	40%	13%	75%	14%
Row-wise	4	40%	12%	75%	12%
	8	40%	13%	75%	14%
	12	40%	12%	75%	14%
	16	40%	13%	75%	14%

Table 3: Comparison of Column-wise and Row-wise using Sum of Absolute Difference

		Parameter			
	Sectors	PRCP	LIRS	LSRR	LSRI
Column-wise	4	41%	13%	74%	14%
	8	43%	13%	73%	14%
	12	43%	13%	73%	14%
	16	43%	13%	72%	14%
Row-wise	4	40%	12%	74%	14%
	8	41%	12%	73%	14%
	12	41%	12%	73%	14%
	16	41%	12%	73%	13%

From Table 2 and Table 3 we can say that the results for Column-wise are comparatively better than that for Row-wise for both Euclidean distance and sum of Absolute difference. There are a few exceptions: For LIRS 8 sectors Row-wise DCT gives better result for Euclidean distance; For LSRR 16 sectors Row-wise DCT gives better result for sum of Absolute difference.

Considering the similarity measure **Euclidean distance**, from the Table 2 we can draw the following observations: The best result for **PRCP** is given by **Column-wise** approach for **8, 12, and 16 sectors** with the average performance of **44%**. The best result for **LIRS** is given by **Column-wise** approach for **16 sectors**and **Row-wise** approach for **8 and 16 sectors**with the average performance of **13%**. The best result for **LSRR** is given by**all the sectors** of**Column-wise** approachas well as **Row-wise** with the average performance of **75%**. **LSRI** gives good result for **all the sectors** of **Column-wise** approach and **4, 8 and 12 sectors** of **Row-wise** approach with the average performance of **14%**.

Considering the similarity measure **sum of Absolute difference**, from the Table 3 we can draw the following observations: The best result for **PRCP** is given by **Column-wise** approach for **8, 12, and 16 sectors** with the average performance of **43%**. The best result for **LIRS** is given by **Column-wise** approach for **4,8, 12, and 16 sectors**with the average performance of **13%**. The best result for **LSRR** is given by **Column-wise** approach for **16 sectors** with the average performance of **72%**. **LSRI** gives good result for **all the sectors** of **Column-wise** approach and **4, 8 and 12 sectors** of **Row-wise** approach with the average performance of **14%**.

7. CONCLUSION

Therefore, in this paper we have implemented the CBIR technique using the algorithm of Feature Vector Generation for Row-wise DCT transformed image. The Wang database [11,12] is used where we have classified the 1000 images into 10 classes. The evaluation parameters used are Precision, Recall, LIRS, LSRR and LSRI. The results of these parameters are calculated using Euclidean distance and sum of Absolute difference.

On comparison of these results, we can conclude that for the parameter Precision-Recall crossover point we get better resultswhen we use sum of Absolute differenceas the similarity measure as compared to Euclidean distance. The average performance of the Precision-Recall crossover point when we use sum of Absolute difference as the similarity measure is 41% while the average performance of the Precision-Recall crossover point when we use Euclidean distance as the similarity measure is 40%. Also, for the parameter LIRS we get better results when we use Euclideandistance as the similarity measure as compared to sum of Absolute difference. The average performance of LIRS when we use sum of Absolute difference as the similarity measure is 12% while the average performance of LIRS when we use Euclidean distance as the similarity measure is 12%. For the parameter LSRR we get better results when we use sum of Absolutedifference as the similarity measure as compared to Euclidean distance. The average performance of LSRRwhen we use sum of Absolute difference as the similarity measure is 73% while the average performance of LSRR when we use Euclidean distance as the similarity measure is 74%.For the parameter LSRI we get better results when we use sum of Absolutedifference as the similarity measure as compared to Euclidean distance. The average performance of LSRI when we use sum of Absolute difference as the similarity measure is 13% while the average performance of LSRI when we use Euclidean distance as the similarity measure is 13%.

On comparison of results of Row-wise DCT transformed images to Column-wise DCT transformed images, we can conclude that Column-wise DCT transformed images give better results comparatively for both the similarity measure-sum of Absolute difference and Euclidean distance.From Table 2 and Table 3 we can also observe that sum of Absolute distance gives better results as compared to Euclidean distance for all the parameters PRCP, LIRS, LSRR, LSRI.

8. REFRENCES

[1] S. Nandgopalan, Dr. B. S. Adiga , N. Deepak, "A Universal Model for Content-Based Image Retrieval", World Academy of Science, Engineering and Technology 46 2008.

[2] MarjoMarkkula, Marius Tico, BemmuSepponen, KatjaNirkkonen and EeroSormunen, "A Test Collection for the Evaluation of Content-Based Image Retrieval Algorithms - A User and Task-Based Approach", Published in Information Retrieval 4(3/4), 275-294 (2001). *

[3] Dr. H. B. Kekre, Dhirendra Mishra, "Sectorization of Walsh and Walsh Wavelet in CBIR", International Journal on Computer Science and Engineering (IJCSE) Vol. 3 No. 6 June 2011.*

[4] Kato, T., "Database architecture for content based image retrieval in Image Storage and Retrieval Systems" (Jambardino A andNiblack W eds),Proc SPIE 2185, pp 112-123, 1992.

[5] V.N.Gudivada and V.V.Raghavan.: Special issue on content-based image retrieval systems - guest eds. *IEEE Computer*. 28(9) (1995) 18-22 [gr95].

[6] H.B.Kekre, Dhirendra Mishra,"Digital Image Search & Retrieval using FFT Sectors of Color Images" published in International Journal of Computer Science and Engineering (IJCSE) Vol. 02,No.02,2010,pp.368-372 ISSN 0975-3397.

[7] H. B. Kekre, Dhirendra Mishra, "DCT Sectorization for Feature Vector Generation in CBIR" International Journal of Computer Applications (0975 – 8887) Volume 9– No.1, November 2010.

[8] H.B.Kekre, Tanujasarode, VinayaRawool, "Finger Print Identification using Discrete Sine Transform (DST)" International Conference on Advanced Computing & Communication Technology (ICACCT-2008) Asia Pacific Institute of Information Technology, Panipat

India 8-9 Nov 2008H. B. Kekre, Dhirendra Mishra, "DCT Sectorization for Feature Vector Generation in CBIR" International Journal of Computer Applications (0975 – 8887) Volume 9– No.1, November 2010.

[9] H.B.Kekre, SudeepThepade, Juhi Jain and Naman Agrawal, "IRIS Recognition using Texture Features Extracted from Haarlet Pyramid", International Journal of Computer Applications (IJCA) Vol. 11, No.12, pp.01-05, December, 2010.

[10] H.B.Kekre, Kamal Shah, "Application of DCT row and column feature vector for face recognition with comparison to full DCT and PCA", International Journal of Computer Applications in Engineering, Technology and Science (IJ-CA-ETS) , Vol. 1, No.2, 435-439 April/September 2009.

[11] James Z. Wang, Jia Li, GioWiederhold, ``SIMPLIcity: Semantics-sensitive Integrated Matching for Picture LIbraries," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol 23, no.9, pp. 947-963, 2001.

[12] Jia Li, James Z. Wang, ``Automatic linguistic indexing of pictures by a statistical modeling approach," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 9, pp. 1075-1088, 2003.

[13] H.B.Kekre, Dhirendra Mishra, Chirag Thakkar, "Column wise DCT plane sectorization in CBIR," International Journal of Computer Science and Information Technologies (IJCSIT), vol. 3 no. 1, pp. 3229-3235, 2012.

[14] H. B. Kekre, Dhirendra Mishra, "DCT-DST Plane sectorization of Row-wise Transformed color Images in CBIR", International Journal of Engineering Science and Technology Vol. 2 (12), 2010,7234-7244.

[15] Xiang-Yu Huang, Yu-Jin Zhang, Dong Hu, "Image Retrieval Based on Weighted Texture Features Using DCT Coefficients of JPEG Images" ICICS-PCM 2003, 15-18 December 2003.

[16] H.B.Kekre, Dhirendra Mishra, "Four walsh transform sectors feature vectors for image retrieval from image databases", published in international journal of computer science and information technologies (IJCSIT) Vol. 1 (2) 2010, 33-37 ISSN 0975-9646.

[17] H.B.Kekre, Dhirendra Mishra, "Density distribution and sector mean with zero-sal and highest-cal components in Walsh transform sectors as feature vectors for image retrieval", published in international journal of Computer scienece and information security (IJCSIS) Vol.8(4) 2010, ISSN 1947-5500.

[18] H.B.Kekre, Dhirendra Mishra, "Density distribution in walsh transfom sectors ass feature vectors for image retrieval", published in international journal of compute applications (IJCA) Vol.4(6) 2010, 30-36 ISSN 0975-8887.

[19] Arun Ross, Anil Jain, James Reisman, "A hybrid fingerprint matcher," Int'l conference on Pattern Recognition (ICPR), Aug 2002.