Texture Classification using Multiresolution Decomposition of Sorted DCT and Fractional DCT Coefficients

Neha Aggarwal & Tarun Kr. Rawat
Netaji Subhas Institute of Technology, Delhi University
E-mail: nehaaggarwal.1987@gmail.com, tarundsp@gmail.com.

Abstract - In this paper, a method is proposed which uses the multiresolution decomposition on the sorted Fractional DCT (FrDCT) and standard DCT coefficients for feature vector extraction of textures. The method shows better classification rates for DCT than conventional DCT based algorithm.

Index Terms—DCT, Fractional DCT, Texture Classification, Multiresolution Decomposition

I. INTRODUCTION

Texture features are capable of providing lot of information and characteristics about the image. These features can help in classification of the textures which can further lead to better ways of image segmentation [1-2]. This classification and segmentation has many applications in the world of image analysis, like in text detection [3], object classification [4], medical imaging (identifying effected regions) [5], palm print / fingerprint identification [6].

Many algorithms have been devised for texture analysis, in particular texture classification. Many approaches used wavelet transform to extract the feature vectors [7-8]. Ahmadian and Mostafa in [7] explained the application of Gabor wavelet transform for texture classification and it showed better results than dyadic wavelet transform. But with better classification rate, computational time was also increased. In reference [8], Arivazhagan, Ganesan and Subash Kumar explained texture classification with ridgelet transform. But in this case, for high accuracy rate, more textural features were required.

Then DCT was used in many algorithms. In reference [9], Sim, Kim and Park proposed an algorithm using mask for texture descriptors. This method gave fast feature extraction but accuracy rate was compromised.

Huang and Chang in [10] proposed a method to extract feature vectors based on multiresolution decomposition of DCT coefficients. They showed improved results as compared to wavelet transforms with better classification rate and less computational effort. In this paper, we propose an algorithm, which uses the multiresolution decomposition on sorted fractional and standard DCT coefficients for feature vector extraction. The results show that DCT coefficients give better results as compared to the fractional DCT (FrDCT) and other algorithms.

The remaining paper is organized as following: Section II gives brief introduction about Fractional DCT and multiresolution decomposition. It also explains the feature vector extraction. Section III gives the algorithm. Section IV gives the Simulation and experimental results on the Brodatz database. Then the conclusions are given in section V.

II. PROPOSED MODEL

A. Brief Introduction on Fractional DCT

If $x$ is a matrix of dimension $N \times N$, then the standard DCT [11] is computed by,

$$X = C \times x$$

(1)

where $C$ is the transformation matrix given by
For the fractional operator, the transformation matrix \( C \), is decomposed using the diagonalised form [12]. The transformation matrix for fractional DCT \( \alpha \) is given as,

\[
C_\alpha = 2\Re \left[ \sum_{n=1}^{N} U_n e^{i\frac{\pi}{N} n \alpha} \right]
\]

(3)

where \( \Re \) is Real Operator and \( e^{i\frac{\pi}{N} n \alpha} = \lambda_n \), the eigenvalues of the DCT transformation matrix \( C \), such that for fractional matrix,

\[
C_\alpha = U A^\alpha U
\]

(4)

with \( A \) as a diagonal matrix with as diagonal entries \( \lambda_n \) and \( U \) is a unitary matrix with columns as eigen vectors of \( C \).

For \( \alpha = 0.5 \), is given by \( C_{0.5} \).

\[
C_{0.5} = 2\Re \left[ \sum_{n=1}^{N} U_n e^{i\frac{\pi}{N} n \times 0.5} \right]
\]

(5)

for \( 8 \times 8 \) blocksize, FrDCT matrix is given as

\[
\begin{bmatrix}
0.7033 & 0.3144 & -0.1100 & 0.0381 & 0.0965 & 0.4107 & 0.0796 & 0.4567 \\
0.0683 & 0.7431 & 0.4347 & 0.1207 & 0.1701 & -0.1212 & -0.0936 & -0.4327 \\
0.4916 & -0.2384 & 0.4664 & -0.0783 & -0.4766 & -0.4903 & -0.0643 & 0.0787 \\
0.3512 & -0.1322 & -0.3912 & 0.3933 & 0.0758 & -0.2358 & 0.5179 & -0.4711 \\
0.2121 & -0.3665 & 0.1209 & 0.2088 & 0.7171 & -0.1608 & -0.4713 & -0.0126 \\
-0.1217 & -0.2374 & 0.4590 & 0.6641 & -0.1799 & 0.4839 & -0.1027 & -0.0091 \\
0.1531 & -0.2806 & 0.3441 & -0.5805 & 0.2344 & 0.3801 & 0.3356 & -0.3611 \\
-0.2290 & 0.0718 & 0.2911 & 0.0405 & 0.3567 & -0.3407 & 0.6060 & -0.4955 \\
\end{bmatrix}
\]

B. Multi-resolution reordering of sorted FrDCT block

In the proposed algorithm, the \( N \times N \) FrDCT block is first row-wise sorted. The sorted block is then reordered into \((3 \log_2 N + 1)\) subbands [10]. For a coefficient \( X_{\alpha}(u,v) \) let \( 2^{a-1} \leq u < 2^a \) and \( 2^{b-1} \leq v < 2^b \), where \( a,b \) are integers, then \( X_{\alpha}(u,v) \) will be placed in the \( S_i \) subband and \( i \) is calculated as,

\[
i = \begin{cases} 
0, m = 0 \\
(m-1) \times 3 + \left( \frac{a}{m} \right) \times 2 + \frac{b}{m}, \text{otherwise} 
\end{cases}
\]

(3)

where \( m = \max(a,b) \).

Reordering of a \( 8 \times 8 \) block is shown in figure 1.

\[
\begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
50 & 51 & 52 & 53 & 54 & 55 & 56 & 57 \\
58 & 59 & 60 & 61 & 62 & 63 & 64 & 65 \\
66 & 67 & 68 & 69 & 70 & 71 & 72 & 73 \\
74 & 75 & 76 & 77 & 78 & 79 & 80 & 81 \\
\end{array}
\]

Figure 1. Reordering of an 8X8 block.

C. Feature Vector Representation

For creating feature vector, mean (\( \mu \)) and standard deviation (\( \sigma \)) of every subband is calculated. If \( S_i \) is a specific subband, then

\[
\mu_i = \frac{1}{\int \int |S_i(x,y)| dx dy} \int \int |S_i(x,y)| dx dy
\]

(8)

and

\[
\sigma_i = \sqrt{\frac{1}{\int \int |S_i(x,y)| dx dy} \int \int |S_i(x,y)|^2 dx dy - \mu_i^2} \int \int |S_i(x,y)| dx dy
\]

(9)

The feature vector is then represented as,

\[
f = \left[ \mu_0, \sigma_0, \mu_1, \sigma_1, ..., \mu_K, \sigma_K \right]
\]

(10)

where \( K = (3 \log_2 N + 1) \)

III. ALGORITHM:

1. For a given test texture image, compute FrDCT (or DCT) of the image using the block size of \( 8 \times 8 \) and fraction \( \alpha \).

\[
X_{0.5} = C_{0.5} \times x
\]

(6)
2. The transformed matrix is then sorted in ascending order and then multiresolution decomposition is done to get 10 subbands.
3. For every subband, mean and standard deviation is calculated to form the feature vector of the test image \( f_t \).
4. Euclidean distance \([10]\) between \( f_t \) and \( f_d \) is calculated, where \( f_d \) is the feature vector of an image texture from the database and \( \alpha(f_k) \) is standard deviation of respective features in image database.
5. Based on minimum distance, the texture is classified.

\[
d(f_t, f_d) = \sum_{k} \frac{|f_k^t - f_k^d|}{\alpha(f_k)}
\]

IV. SIMULATION RESULTS

The image database used for the simulation is the Brodatz Album \([13]\). 50 images from the database were considered. Each image was divided into 20 randomly positioned images giving us a data set of 1000 images. For training the database, 10 sub images for each class were used. Remaining 10 from each class were used for testing purposes.

For training the database, \( i \) (\( i = 1, 2 \ldots 10 \)) texture cuts per class is used. So, 10 databases are created based on number of texture cuts per class used. For calculating the feature vector of a class, the mean value is taken over the feature vectors of the same class. Thereby generating feature vectors of same length for each class.

Classification is said to be correct if the algorithm classifies correctly that the test image belongs to which class. Classification Rate is calculated as per the following formula,

\[
\text{Classification Rate} = \frac{\text{No. of correct classification done}}{\text{Total number of test cases}}
\]

Figure 3: Comparison on the basis of texture cuts per class.

V. CONCLUSION

In this paper we presented an algorithm which can produce a feature vector set for an image texture from the DCT and FrDCT coefficients. We showed that algorithm involving sorting and multiresolution (S-MRDCT and S-MRFrDCT) gives better classification performance as compared to the conventional DCT method (DCT) and multiresolution algorithm MRDCT.

Due to energy compaction property, the results are best when \( \alpha = 1 \), i.e. for DCT.
Every subband was used to create a feature vector of length 20. All the simulations are done using all the feature vectors. For reducing calculation effort, lesser feature vectors or feature vectors with fewer elements, can be used.

REFERENCES


