The International Journal of Engineering And Science (IJES)
||Volume|| 1 ||Issue|| 2 ||Pages|| 215-220 ||2012||
ISSN: 2319 – 1813 ISBN: 2319 – 1805

# Face Recognition using DCT - DWT Interleaved Coefficient Vectors with NN and SVM Classifier

[1]Divya Nayak (M. Tech.) [1]Mr. Sumit Sharma
[1, 2] *Shri Ram Institute of Technology Jabalpur (M.P.)*

**Abstract**

*Face recognition applications are continuously gaining demands because of their requirements person authentication, access control and surveillance systems. The researchers are continuously working for making the system more accurate and faster in the part of that research this paper presents a face recognition system which uses DWT, DCT interleaved component for feature vector formation & tested the Support Vector Machine and ANN for classification. Finally the proposed technique is implemented and comprehensively analyzes to test its efficiency we also compared these two classification method.*

**Key Words:** *Face Recognition, Support Vector Machine (SVM), Artificial Neural Network (ANN).*

Date of Submission: 06, December, 2012 ⟺ Date of Publication: 25, December 2012

## I Introduction

Increasing security demands are forcing the scientists and researchers to develop more advanced security systems one of them is biometric security system this system is particularly preferred because of its proven natural uniqueness and user has no need to carry additional devices like cares, remote etc. the biometric security systems refers to the identification of humans by their characteristics or traits. Biometrics is used in computer science as a form of identification and access control [1]. It is also used to identify individuals in groups that are under surveillance. One of the biometric identification is done by the face of the person, this method has several application from online (person surveillance) to offline (scanned image identification) etc. face recognition system has its own advantage over other biometric methods that it can be detected from much more distance without need of special sensors or scanning devices.

There are several methods are proposed so far for the face recognition system using different feature extraction techniques or different training approaches or different classification approaches to improve the efficiency of the system. The rest of the paper is arrange as the second section presents a short review of the work done so far, the third section presents the details of technical terms used in the algorithm, the fourth section presents proposed algorithm followed by analysis an conclusion in next chapters.

## II Related Work

This section presents some of the most relevant work recent work presented by other researchers. Ignas Kukenys and Brendan McCane [2] describe a component-based face detector using support vector machine classifiers. They present current results and outline plans for future work required to achieve sufficient speed and accuracy to use SVM classifiers in an online face recognition system. They used a straightforward approach in implementing SVM classifier with a Gaussian kernel that detects eyes in grayscale images, a first step towards a component-based face detector. Details on design of an iterative bootstrapping process are provided, and we show which training parameter values tend to give best results. Jixiong Wang (jameswang) [3] presented a detailed report on using support vector machine and application of different kernel functions (Linear kernel, Polynomial kernel, Radial basis kernel, Sigmoid kernel) and multiclass classification methods and parameter optimization. Face recognition in the presence of pose changes remains a largely unsolved problem. Severe pose changes, resulting in dramatically different appearances, are one of the main difficulties and one solution approach is presented by Antony Lam and Christian R. Shelton [4] they present a support vector machine (SVM) based system that learns the relations between corresponding local regions of the face in different poses as well as a simple SVM based system for automatic alignment of faces in differing poses. Jennifer Huang, Volker Blanz and Bernd Heisele [5] present an approach to pose and illumination invariant face recognition that combines two recent advances in the computer vision field: component-based recognition and 3D morph able models. In preliminary experiments we show the potential of our approach regarding pose and illumination invariance. A Global versus Component based Approach for Face Recognition with Support Vector Machines is presented by Bernd Heisele, Purdy Ho, Tomaso Poggio [6]. They present a component based method and two global methods for face recognition and evaluate them with respect to robustness against pose changes. In the component system they first locate facial components, extract them and combine them into a single feature vector which is classified by a Support Vector Machine (SVM). The two global systems recognize faces by classifying a single feature vector consisting of the gray values of the whole face image. The component system clearly outperformed both global systems on all tests. Yongmin Li, Shaogang Gong, Jamie Sherrah, Heather Liddell [7] presented face detection across multiple views (frontal view, owing to the significant non-linear variation caused by rotation in depth, self-occlusion and self-shadowing). In their approach the view sphere is separated into several small segments. On each segment, a face detector is constructed. They explicitly estimate the pose of an image regardless of whether or not it is a face. A pose estimator is constructed using Support Vector Regression. The pose information is used to choose the appropriate face detector to determine if it is a face. With this pose-estimation based method, considerable computational efficiency is achieved. Meanwhile, the detection accuracy is also improved since each detector is constructed on a small range of views. Rectangle Features based method is presented by Qiong Wang, Jingyu Yang, and Wankou Yang [8] presents an efficient approach to achieve accurate face detection in still gray level images. Characteristics of intensity and symmetry in eye region are used as robust cues to find possible eye pairs. Three rectangle features are developed to measure the intensity relations and symmetry. According to the eye-pair-like regions which have been found, the corresponding square image patches are considered to be face candidates, and then all the candidates are verified by using SVM. Finally, all the faces in the image are detected.

## III Discrete Wavelet Transform (Dwt)

In numerical analysis and functional analysis, a discrete wavelet transform (DWT) is any wavelet transform for which the wavelets are discretely sampled. As with other wavelet transforms, a key advantage it has over Fourier transforms is temporal resolution: it captures both frequency and location information (location in time).

The DWT of a signal ***x*** is calculated by passing it through a series of filters. First the samples are passed through a low pass filter with impulse response ***g*** resulting in a convolution of the two:

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k] g[n-k]$$

The signal is also decomposed simultaneously using a high-pass filter ***h***. The output gives the detail coefficients (from the high-pass filter) and approximation coefficients (from the low-pass). It is important that the two filters are related to each other and they are known as a quadrature mirror filter. However, since half the frequencies of the signal have now been removed, half the samples can be discarded according to Nyquist's rule. The filter outputs are then sub sampled by 2 (Mallat's and the common notation is the opposite, g- high pass and h- low pass):

$$y_{low}[n] = \sum_{k=-\infty}^{\infty} x[k] g[2n-k]$$

$$y_{low}[n] = \sum_{k=-\infty}^{\infty} x[k] h[2n-k]$$

This decomposition has halved the time resolution since only half of each filter output characterizes the signal. However, each output has half the frequency band of the input so the frequency resolution has been doubled.

The DWT and IDWT for an one-dimensional signal can be also described in the form of two channel tree-structured filter banks. The DWT and IDWT for a two-dimensional image x[m,n] can be similarly defined by implementing DWT and IDWT for each dimension m and n separately DWT n [DWT m [x[m,n]], which is shown in Figure 1.

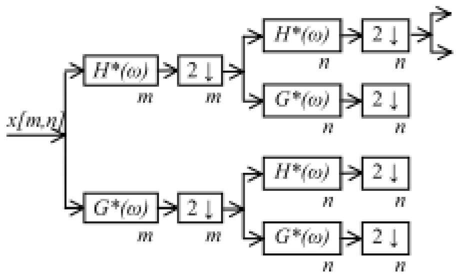


Figure 1: DWT for two-dimensional images [9]

An image can be decomposed into a pyramidal structure, which is shown in Figure 2, with various band information: low-low frequency band LL, low-high frequency band LH, high-low frequency band HL, high-high frequency band HH.

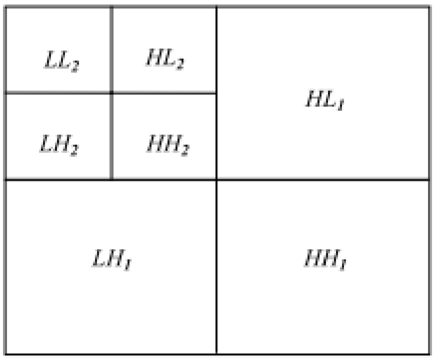


Figure 2: Pyramidal structure

## IV Discrete Cosine Transform (DCT)

A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. DCTs are important to numerous applications in science and engineering, from lossy compression of audio (e.g. MP3) and images (e.g. JPEG) (where small high-frequency components can be discarded), to spectral methods for the numerical solution of partial differential equations.

Formally, the discrete cosine transform is a linear, invertible function $f: R^N \rightarrow R^N$ (where $R$ denotes the set of real numbers), or equivalently an invertible $N \times N$ square matrix. There are several variants of the DCT with slightly modified definitions. The $N$ real numbers $x_0 \ldots x_{N-1}$ are transformed into the $N$ real numbers $X_0 \ldots X_{N-1}$ according to one of the formulas:

$$X_k = \frac{1}{2}(x_0 + (-1)^k x_{N-1}) + \sum_{n=1}^{N-2} x_n cos\left[\frac{\pi}{N-1} nk\right]$$

$$where\ k = 0 \ldots . N-1$$

The 2D DCT is given by

$$y_{mn} = \frac{1}{N^2} u(m) u(n) \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} x_{ij} cos\left[\frac{(2i+1)m}{2N}\pi\right] * cos\left[\frac{(2j+1)n}{2N}\pi\right]$$

$$u(m) = \begin{cases} 1/\sqrt{2}, m = 0 \\ 1, otherwise \end{cases}$$

### 5. Support Vector Machine (SVM)

Support Vector Machines (SVMs) have developed from Statistical Learning Theory [6]. They have been widely applied to fields such as character, handwriting digit and text recognition, and more recently to satellite image classification. SVMs, like ANN and other nonparametric classifiers have a reputation for being robust. SVMs function by nonlinearly projecting the training data in the input space to a feature space of higher dimension by use of a kernel function. This results in a linearly separable dataset that can be separated by a linear classifier. This process enables the classification of datasets which are usually nonlinearly separable in the input space. The functions used to project the data from input space to feature space are called kernels (or kernel machines), examples of which include polynomial, Gaussian (more commonly referred to as radial basis functions) and quadratic functions. By their nature SVMs are intrinsically binary classifiers however there are strategies by which they can be adapted to multiclass tasks. But in our case we not need multiclass classification.

### 5.1 SVM classification

Let $x_i \in R^m$ be a feature vector or a set of input variables and let $y_i \in \{+1, -1\}$ be a corresponding class label, where $m$ is the dimension of the feature vector. In linearly separable cases a separating hyper-plane satisfies [8].

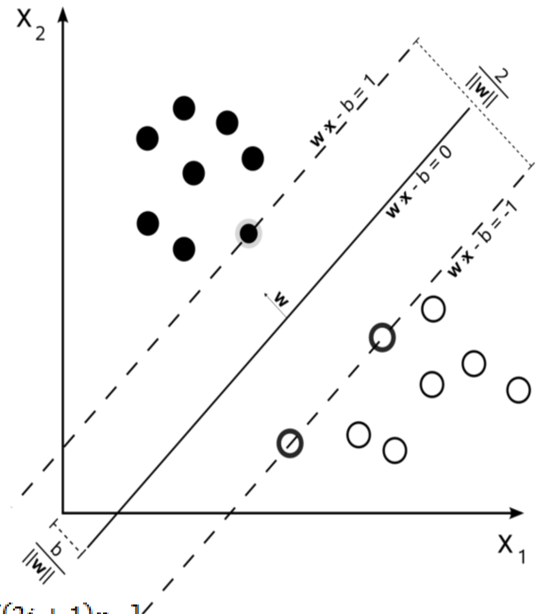


Figure 3: Maximum-margin hyper-plane and margins for an SVM trained with samples from two classes. Samples on the margin are called the support vectors.

$$y_i(\langle w.x_i\rangle + b) \geq 1, i = 1 \ldots n, \ldots .(1)$$

Where the hyper-plane is denoted by a vector of weights ***w*** and a bias term ***b***. The optimal separating hyper-plane, when classes have equal loss-functions, maximizes the margin between the hyper-plane and the closest samples of classes. The margin is given by

$$d(w,b) = \min_{x_i, y_i = 1} \frac{|\langle w.x_i\rangle + b|}{\|w\|} + \min_{x_i, y_i = -1} \frac{|\langle w.x_j\rangle + b|}{\|w\|} = \frac{2}{\|w\|} \ldots \ldots .($$

The optimal separating hyper-plane can now be solved by maximizing (2) subject to (1). The solution can be found using the method of Lagrange multipliers. The objective is now to minimize the Lagrangian

$$L_p(w,b,a) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i y_i (\langle \mathbf{w}.\mathbf{x}_i \rangle + b) + \sum_{i=1}^{l} \alpha_i \dots\dots (3)$$

and requires that the partial derivatives of ***w*** and ***b*** be zero. In (3), $\boldsymbol{\alpha_i}$ is nonnegative Lagrange multipliers. Partial derivatives propagate to constraints$w = \sum_i \alpha_i y_i x_i$ $and$ $\sum_i \alpha_i y_i = 0$.
Substituting ***w*** into (3) gives the dual form

$$L_d(w,b,a) = \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i \alpha_j y_i\ y_j \langle x_i, x_j \rangle \dots\dots (4)$$

which is not anymore an explicit function of **w** or ***b***. The optimal hyper-plane can be found by maximizing (4) subject to $\sum_i \alpha_i y_i = 0$ and all Lagrange multipliers are nonnegative. However, in most real world situations classes are not linearly separable and it is not possible to find a linear hyperplane that would satisfy (1) for all ***i = 1. . . n.*** In these cases a classification problem can be made linearly separable by using a nonlinear mapping into the feature space where classes are linearly separable. The condition for perfect classification can now be written as

$$y_i(\langle w.\emptyset(x_j)\rangle + b) \geq 1, i = 1,\dots,n, \dots\dots (5)$$

where ***Φ*** is the mapping into the feature space. Note that the feature mapping may change the dimension of the feature vector. The problem now is how to find a suitable mapping ***Φ*** to the space where classes are linearly separable. It turns out that it is not required to know the mapping explicitly as can be seen by writing (5) in the dual form

$$y_i\left(\sum_{j=1}^{l} \alpha_i y_i \langle \emptyset(x_j).\emptyset(x_i)\rangle\right) + b \geq 1, i = 1,\dots,n \dots\dots (6)$$

and replacing the inner product in (6) with a suitable kernel function $k(x_j, x_i) = (\emptyset(x_j), \emptyset(x_i))$.

This form arises from the same procedure as was done in the linearly separable case that is, writing the Lagrangian of (6), solving partial derivatives, and substituting them back into the Lagrangian. Using a kernel trick, we can remove the explicit calculation of the mapping *Φ* and need to only solve the Lagrangian (5) in dual form, where the inner product $(x_j, x_i)$ has been transposed with the kernel function in nonlinearly separable cases. In the solution of the Lagrangian, all data points with nonzero (and nonnegative) Lagrange multipliers are called support vectors (SV).

Often the hyperplane that separates the training data perfectly would be very complex and would not generalize well to external data since data generally includes some noise and outliers. Therefore, we should allow some violation in (1) and (6). This is done with the nonnegative slack variable $\zeta_i$

$$y_i(\langle w.\emptyset(x_j)\rangle + b) \geq 1-, i = 1,\dots,n, \dots\dots (7)$$

The slack variable is adjusted by the regularization constant ***C***, which determines the tradeoff between complexity and the generalization properties of the classifier. This limits the Lagrange multipliers in the dual objective function (5) to the range $0 \leq \alpha_i \leq C$. Any function that is derived from mappings to the feature space satisfies the conditions for the kernel function.

The choice of a Kernel depends on the problem at hand because it depends on what we are trying to model.

The SVM gives the following advantages over neural networks or other AI methods (link for more details http://www.svms.org).

SVM training always finds a global minimum, and their simple geometric interpretation provides fertile ground for further investigation.

Most often Gaussian kernels are used, when the resulted SVM corresponds to an RBF network with Gaussian radial basis functions. As the SVM approach “automatically” solves the network complexity problem, the size of the hidden layer is obtained as the result of the QP procedure. Hidden neurons and support vectors correspond to each other, so the center problems of the RBF network is also solved, as the support vectors serve as the basis function centers.

Classical learning systems like neural networks suffer from their theoretical weakness, e.g. back-propagation usually converges only to locally optimal solutions. Here SVMs can provide a significant improvement.

The absence of local minima from the above algorithms marks a major departure from traditional systems such as neural networks.

SVMs have been developed in the reverse order to the development of neural networks (NNs). SVMs evolved from the sound theory to the implementation and experiments, while the NNs followed more heuristic path, from applications and extensive experimentation to the theory.

## V Proposed Algorithm

The proposed algorithm can be described in following steps.

1. Firstly divide the image into 8x8 blocks then take the 2D DCT of the image & then select the components in zigzag manner.

2. Secondly we take the 2D DWT of the image blocks & then select the zigzag components of only LL components.

3. Now the feature vectors are formed by interleaving these two components (even components from DWT and odd from DCT).

| | | |
|---|---|---|
| A1 | A2 | A6 |
| A3 | A5 | |
| A4 | | |

| | | |
|---|---|---|
| B1 | B2 | B6 |
| B3 | B5 | |
| B4 | | |

Figure 4: let the left block represents the DCT components and the right block represents the LL components of DWT.

According to figure 4 the feature vector will be formed by [A1, B1, A2, B2, A3, B3………..].

4. Like above step these vectors are created for all classes of faces.

5. These vectors are used to train the N*(N-1)/2 (N is the number of classes) SVM classifiers as we used one against one method.

6. For detection purpose the input image vectors are calculated in same way as during training and then it is applied on each classifier.

7. Finally the decision is mode on the basis of majority of class returned by N*(N-1)/2 vectors.

## VI Simulation Results

We used the ORL database for testing of our algorithm. The ORL database contains 40 different faces with 10 samples of each face. The accuracy of the algorithm is tested for different number of faces, samples and vector length.

The comparison of the proposed method with Neural Network and PCA based method shows that the proposed method outperforms the other two.

**Table 1: Result for 40 Faces and 10 Samples Each using SVM**

| Face No. | TP | TN | FP | FN | Accuracy | Precision | Recall | F-Meas. |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.6 | 0.9889 | 0.0111 | 0.4 | 0.95 | 0.8571 | 0.6 | 0.7059 |
| 2 | 0.8 | 1 | 0 | 0.2 | 0.98 | 1 | 0.8 | 0.8889 |
| 3 | 1 | 0.9889 | 0.0111 | 0 | 0.99 | 0.9091 | 1 | 0.9524 |
| 4 | 1 | 0.9889 | 0.0111 | 0 | 0.99 | 0.9091 | 1 | 0.9524 |
| 5 | 0.9 | 0.9667 | 0.0333 | 0.1 | 0.96 | 0.75 | 0.9 | 0.8182 |
| 6 | 1 | 0.9444 | 0.0556 | 0 | 0.95 | 0.6667 | 1 | 0.8 |
| 7 | 1 | 0.9778 | 0.0222 | 0 | 0.98 | 0.8333 | 1 | 0.9091 |
| 8 | 0.9 | 1 | 0 | 0.1 | 0.99 | 1 | 0.9 | 0.9474 |
| 9 | 0.8 | 0.9889 | 0.0111 | 0.2 | 0.97 | 0.8889 | 0.8 | 0.8421 |
| 10 | 0.8 | 0.9889 | 0.0111 | 0.2 | 0.97 | 0.8889 | 0.8 | 0.8421 |
| | | | | | | | | |
| Average | 0.88 | 0.9833 | 0.0167 | 0.12 | 0.973 | 0.8703 | 0.88 | 0.8658 |

**Table 2: Result for 40 Faces and 10 Samples Each using NN**

| Face No. | TP | TN | FP | FN | Accuracy | Precision | Recall | F-Meas. |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.9 | 0.9889 | 0.0111 | 0.1 | 0.98 | 0.9 | 0.9 | 0.9 |
| 2 | 0.8 | 1 | 0 | 0.2 | 0.98 | 1 | 0.8 | 0.8889 |
| 3 | 1 | 0.9889 | 0.0111 | 0 | 0.99 | 0.9091 | 1 | 0.9524 |
| 4 | 0.8 | 1 | 0 | 0.2 | 0.98 | 1 | 0.8 | 0.8889 |
| 5 | 0.9 | 0.9889 | 0.0111 | 0.1 | 0.98 | 0.9 | 0.9 | 0.9 |
| 6 | 0.8 | 0.9667 | 0.0333 | 0.2 | 0.95 | 0.7273 | 0.8 | 0.7619 |
| 7 | 0.9 | 0.9889 | 0.0111 | 0.1 | 0.98 | 0.9 | 0.9 | 0.9 |
| 8 | 0.6 | 0.9778 | 0.0222 | 0.4 | 0.94 | 0.75 | 0.6 | 0.6667 |
| 9 | 1 | 0.9889 | 0.0111 | 0 | 0.99 | 0.9091 | 1 | 0.9524 |
| 10 | 0.7 | 1 | 0 | 0.3 | 0.97 | 1 | 0.7 | 0.8235 |
| | | | | | | | | |
| Average | 0.84 | 0.9889 | 0.0111 | 0.16 | 0.974 | 0.8995 | 0.84 | 0.8635 |

**Table 3: Result for Training Time and Matching Time for ANN**

| Number of Faces | ANN Training Time (Sec.) | ANN Matching Time (Sec.) |
|---|---|---|
| 10 | 0.061151 | 0.011579 |
| 20 | 0.13404 | 0.017074 |
| 40 | 0.35419 | 0.030042 |

.

**Table 4: Result for Training Time and Matching Time for SVM**

| Number of Faces | SVM Training Time (Sec.) | SVM Matching Time (Sec.) |
|---|---|---|
| 10 | 0.53541 | 0.067786 |
| 20 | 2.2608 | 0.31951 |
| 40 | 9.2771 | 1.3173 |

## VII Conclusion

This paper presents a DCT, DWT Mixed approach for feature extraction and during the classification phase, the Support Vector Machine (SVM) and Neural Network is tested for robust decision in the presence of wide facial variations. The experiments that we have conducted on the ORL database vindicated that the SVM method performs better than ANN when compared for detection accuracy but when compared for training time and detection time the neural network out performs the SVM. In future we can also compare them with using different kernel functions and learning techniques.

## References

[1] "Biometrics: Overview", Biometrics.cse.msu.edu. 6 September 2007. Retrieved 2012-06-10.

[2] Ignas Kukenys and Brendan McCane “Support Vector Machines for Human Face Detection”, NZCSRSC ’08 Christchurch New Zealand.

[3] Jixiong Wang (jameswang) “CSCI 1950F Face Recognition Using Support Vector Machine: Report”, Spring 2011, May 24, 2011.

[4] Antony Lam & Christian R. Shelton “Face Recognition and Alignment using Support Vector Machines”, 2008 IE.

[5] Jennifer Huang, Volker Blanz and Bernd Heisele “Face Recognition Using Component-Based SVM Classification and Morphable Models”, SVM 2002, LNCS 2388, pp. 334–341, 2002. Springer-Verlag Berlin Heidelberg 2002.

[6] Bernd Heisele, Purdy Ho and Tomaso Poggio “Face Recognition with Support Vector Machines: Global versus Component-based Approach”, Massachusetts Institute of Technology Center for Biological and Computational Learning Cambridge, MA 02142.

[7] Yongmin Li, Shaogang Gong, Jamie Sherrah and Heather Liddell “Support vector machine based multi-view face detection and recognition”, Image and Vision Computing 22 (2004) 413–427.

[8] Qiong Wang, Jingyu Yang, and Wankou Yang “Face Detection using Rectangle Features and SVM”, International Journal of Electrical and Computer Engineering 1:7 2006.

[9] Nataša Terzija, Markus Repges, Kerstin Luck, and Walter Geisselhardt “DIGITAL IMAGE WATERMARKING USING DISCRETE WAVELET TRANSFORM: PERFORMANCE COMPARISON OF ERROR CORRECTION CODES”,