CHAPTER 3

RESEARCH METHODOLOGY

3.1 INTRODUCTION

The research methodology and the framework to be used for the proposed sports video classification system are discussed in this chapter. The proposed system classifies the commonly interested five sports category: baseball, volleyball, football, cricket and tennis. The following sections describe the mathematical background of NSST, SVM and KNN. In this study, the proposed methodology considers several methodologies that have been done by previous researchers that relates to the thesis. In order to achieve better performance, new techniques have been proposed based on methodologies given in this chapter.

3.2 BASIC SIGNAL TRANSFORMS

In digital image processing, there are numerous transforms applied on digital images in order to analyze and manipulate the images. Still images are commonly described as two-dimensional signals. Hence this section presents various popular 2-D transformation techniques.

3.2.1 DISCRETE FOURIER TRANSFORM

The Discrete Fourier transform (DFT) is a special case of Z-transform. It is the sampled Fourier transform that requires an input signal that is discrete. The discrete version of a continuous signal is often created by sampling it. DFT produces the sine and cosine components of an image. A digital image contains point samples of a band-limited continuous 2D signal. According to the uniform sampling theorem, a

signal is band-limited if it contains no energy at frequencies higher than some band limit. The output of DFT represents the image in the frequency domain, while the input image is the spatial domain equivalent. In the image equivalent of the Fourier coefficients, each point represents a particular frequency contained in the spatial domain image (Theodoridis and koutroumbas 2008). For an image X of size $N * N$, the 2-D discrete Fourier transform is given by (3.1)

$$X(a,b) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m,n).\, e^{-j2\pi(a\frac{m}{N}+b\frac{n}{N})} \tag{3.1}$$

The set of Fourier Coefficients corresponding to the digital image can be retransformed to the spatial domain using the inverse Fourier transform, which is given by (3.2)

$$x(m,n) = \frac{1}{N} \sum_{a=0}^{N-1} \sum_{b=0}^{N-1} X(a,b).\, e^{j2\pi(a\frac{m}{N}+b\frac{n}{N})} \tag{3.2}$$

3.2.2 DISCRETE COSINE TRANSFORM

The DCT (Ahmed et al., 1974) is a Fourier related transform, similar to the discrete Fourier transform, which uses only real numbers. It expresses a signal in terms of a sum of cosine functions oscillating at different frequencies. An obvious distinction between a discrete cosine transform and a discrete Fourier transform is that the discrete cosine transform uses only cosine functions, while the discrete Fourier transform uses both cosine and sine functions. The DCT is utilized in many applications, with the most notable being the MPEG Audio Layer-3 (MP3) music format, as well as the Joint Photographic Experts Group (JPEG) image format. For a

2-D signal, like an image, x of size N* N, the DCT (Theodoridis and koutroumbas 2008) is given by (3.3)

$$C(u,v) = a(u).a(v)\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} x(m,n).\cos(\frac{\pi(2m+1)u}{2N}).\cos(\frac{\pi(2n+1)v}{2N}) \qquad (3.3)$$

while the inverse 2D discrete cosine transform is given by

$$x(m,n) = \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} a(u).a(v).C(u,v).\cos(\frac{\pi(2m+1)u}{2N}).\cos(\frac{\pi(2n+1)v}{2N}) \qquad (3.4)$$

where

$$a(k) = \begin{cases} \sqrt{\frac{1}{N}}, k = 0 \\ \sqrt{\frac{2}{N}}, 1 \le k \le N-1 \end{cases} \qquad (3.5)$$

3.2.3 DISCRETE WAVELET TRANSFORM

The basis of the wavelet transform called wavelets, are scaling functions and wavelet functions (Vetterli and Kovacevic 1995). Different wavelet transforms are based on different wavelet basis functions. Many canonical families of orthogonal wavelet basis functions exist, such as the Haar wavelets, the Daubechies wavelets (Daubechies 1990) and the Morlet wavelets. Each wavelet family consists of various distinct sets of wavelet basis functions. For example, the Daubechies family wavelets are commonly denoted by "Daubechies-N", where N is the order. Compared with the sinusoid function which is smooth and symmetric with infinite time duration, wavelets may be asymmetric and are fast changing with limited duration. Therefore wavelets can efficiently represent discontinuity in a signal. In practice, images are

represented usually in the discrete domain. All the continuous transforms are necessarily extended to their corresponding discrete forms.

In the 2D multiresolution wavelet transform, one must introduce a scaling function $\Phi(x,y)$ and a wavelet function $\psi(x,y)$ that are separable. That is

$$\Phi(x, y) = \Phi(x)\, \Phi(y) \tag{3.6}$$

and

$$\psi(x, y) = \psi(x)\, \psi(y) \tag{3.7}$$

In addition, the combinations of these functions are defined as

$$\Phi(x, y)_1 = \Phi(x)\, \Phi(y) \tag{3.8}$$

$$\psi(x, y)_h = \Phi(x)\, \psi(y) \tag{3.9}$$

$$\psi(x, y)_v = \Psi(x)\, \Phi(y) \tag{3.10}$$

$$\psi(x, y)_d = \Psi(x)\, \psi(y) \tag{3.11}$$

Using a similar extension of the properties, one can now define the filters h and g for the basic functions. The filters are h_r, h_c. g_r and g_c for $\Phi(x)$, $\Phi(y)$, $\psi(x)$ and $\psi(y)$ respectively. The subscripts r and c stand for row and column and indicate the direction in which the filters are applied. So, the low pass and detail images are given by

$$C_1 = h_r\, h_c\, C_0 \tag{3.12}$$

$$d_h = d_{1,1} = h_r\, g_c\, C_0 \tag{3.13}$$

$$d_v = d_{1,2} = g_r\, h_C\, C_0 \tag{3.14}$$

and

$$d_d = d_{1,3} = g_r\, g_c\, C_0 \tag{3.15}$$

Computationally, the 2D wavelet Transform of an image is taken in two parts. First, the 1D wavelet transform is taken along the image pixel rows by multiplying each row by the appropriate low and high pass filters *h* and *g*. The low pass and detail groups are then down sampled by two. The second steps in the 2D wavelet transform are accomplished by taking the 1D wavelet transform along the columns of each of the reordered low and high pass filtered groups. These are again accomplished by multiplying both the low pass and high pass filtered groups by the same low pass and high pass filters discussed above. The columns are again down sampled by two. The result of this operation is a decomposition which has a low pass L image in quadrant one, a vertical error image D_v in quadrant two, a horizontal error image D_h in quadrant three, and a diagonal error image D_d in quadrant four. Thus, the first wavelet decomposition of the image is complete. Further wavelet transformations of the resulting low pass image will result in multi-resolution wavelet transform decomposition. Many types of wavelets are available. For example, the JPEG 2000 compression standard uses the bi orthogonal CDF 5/3wavelet for lossless compression and a CDF 9/7 wavelet for lossy compression (Daubechies 1990).

3.2.4 FILTER BANKS

The simplest type of filter bank is one consisting of only two filters. In this case, input filter is decomposed into two sub bands by using a low pass and a high pass filter. These filters are called “analysis” filters. The inverse of this procedure is the synthesis of two sub bands into one signal. The filters used for this procedure are called “synthesis” filters. The band width of each sub band is limited in fewer frequencies than the original signal (Park et al., 2004). As a consequence, the sampling rate can be reduced before any further processing. Reducing the sampling

rate allows for more efficient processing of the signals. The original signal is reconstructed by increasing the sampling rate of the sub bands before applying the synthesis filters.

The analysis filters along with the down sampling system by 2 constitute the analysis section of the filter bank. The up sampling system by 2 along with the synthesis section of filter bank forms the synthesis section. When the filter bank consists of two quadrature mirror filters, then the decomposition procedure is reversible and approximately perfect reconstruction of the original signal is possible. An example of filter bank having 2 channels is shown in Figure 5. L_a is the low pass analysis and H_a is the high pass analysis filters, while L_s is the low pass synthesis and H_s is the high pass synthesis filters. $y_{low}[n]$ is low pass signal and $y_{high}[n]$ is high pass signals. Both are obtained from the decomposition procedure. The reconstructed signal is $X'[n]$.

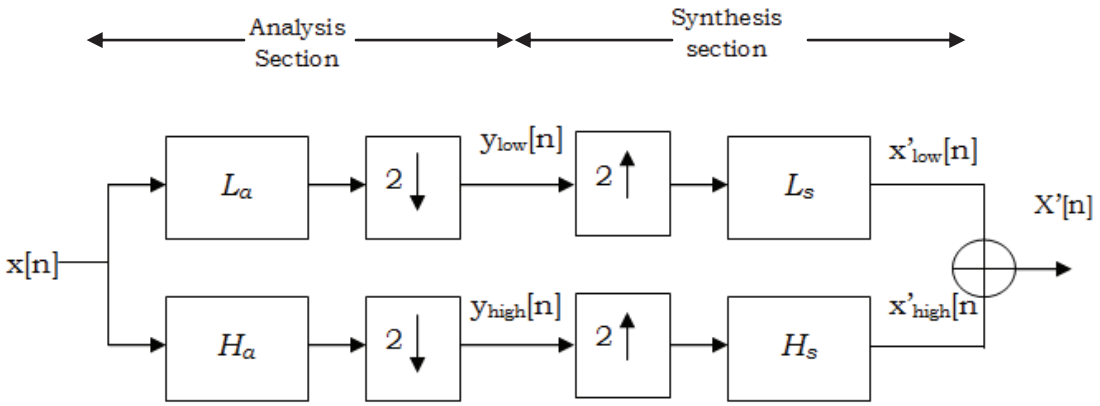


Figure – 5: Example of a 2 channel filter bank.

3.2.5 CONTOURLET TRANSFORM

By combining the Laplacian Pyramid (LP) and the Directional Filter Bank (DFB), Do and Vetterli (2004) proposed the Contourlet transform, which decomposes the spectrum into trapezoid-shaped sub-bands. A one level LP decomposes a input image into a down sampled low pass sub-band and a high pass sub-band, and then the high pass sub-band is decomposed into various directional sub-bands by the DFB, to avoid the division of low frequency regions, at each stage the contour transform removes the low frequency component from the high pass sub-band before implementing the DFB decomposition. If this procedure is iterated on the low pass sub-band, then a multi-scale and multi direction decomposition of images is achieved. By taking the advantages of both the LP and DFB, the contourlet transform can efficiently capture high frequency directional information in images, such as oriented edges.

The spectrum of an original image could be decomposed into four scales, which are then divided into one, two, four and eight directions sub-bands from coarser scales to finer scales, respectively. Due to the abundant number of small absolute values coefficient, the Contourlet transform is a sparse expression of images. Each sub-band of the Contourlet transform, which consists of a pairwise trapezoid shaped regions symmetric to the origin in Figure, corresponds to an oriented basis function. Therefore, the Contourlet transform can offer basis functions oriented at 2^k different directions at each scale, where k is an arbitrary positive integer. Rich and flexible multi scale and oriented basis functions allow the Contourlet transform to effectively

represent smooth contours. However, the Contourlet transform has the drawback of 4/3 redundancy in it's over sampling ratio, which comes from the LP.

3.3 DISCRETE SHEARLET TRANSFORM

This new representation is based on a simple and rigorous mathematical framework which not only provides a more flexible theoretical tool for the geometric representation of multidimensional data, but is also more natural for implementation. In addition, the Shearlet approach can be associated to a multi-resolution analysis. In this section, the development of discrete implementations of the Shearlet transform to obtain the Discrete Shearlet Transform is explained. The theory of composite wavelets, recently introduced by their collaborators (Guo et al., 2004, Guo et al., 2006 and Guo et al., 2006a),provides an especially effective approach for combining geometry and multi scale analysis by taking advantage of the classical theory of affine systems. In dimension n = 2, the affine systems with composite dilations are the collections of the form:

$$A_{AB}(\psi) = \left\{ \psi_{j,l,k}(x) = |\det A|^{j/2}\ \psi(B^l A^j x - k) : j,l \in Z k \in Z^2 \right\} \qquad (3.16)$$

Where $\psi \in L^2(\Re^2)$, A, $B$are 2*2 invertible matrices and $|\det B| = 1$.The elements of this system are called composite wavelets if $A_{AB}(\psi)$ forms a Parseval frame (also called tight frame) for $L^2(\Re^2)$; that is,

$$\sum_{j,l,k} |\left\langle f, \psi_{j,l.k} \right\rangle|^2 = \| f \|^2, \qquad (3.17)$$

For all $f \in L^2(\Re^2)$, the dilations matrices A^j are associated with scale transformations, while the matrices B^l are associated to area preserving geometrical transformations, such as rotations and shear. This framework allows one to construct Parseval frames whose elements range not only at various scales and locations, like wavelets, but also at various orientations. Let us consider a special example of composite wavelets in $L^2(\Re^2)$ called Shearlets. These are collections of the form (3.17) where $A = A_0$ is the anisotropic dilation matrix and $B = B_0$ is the shear matrix, which are given by

$$A_0 = \begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}, B_0 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

For any $\xi = (\xi_1, \xi_2) \in \hat{\Re}^2, \xi_1 \neq 0, let \psi^{(0)}$ be given by

$$\hat{\psi}^{(0)}(\xi) = \hat{\psi}^{(0)}(\xi_1, \xi_2) = \hat{\psi}_1(\xi_1)\hat{\psi}_2(\frac{\xi_1}{\xi_2}), \tag{3.18}$$

where $\hat{\psi}_1, \hat{\psi}_2 \in C^\infty(\hat{\Re}), \sup p\hat{\psi}_1 \subset [-\frac{1}{2}, -\frac{1}{16}] \cup [\frac{1}{16}, \frac{1}{2}] and \sup p\hat{\psi}_2 \subset [-1,1]$.

This implies that $\hat{\psi}^{(0)}$ is C^∞ and compactly supported with sup $p\hat{\psi}^{(0)} \subset [-\frac{1}{2}, \frac{1}{2}]^2$.

In addition,

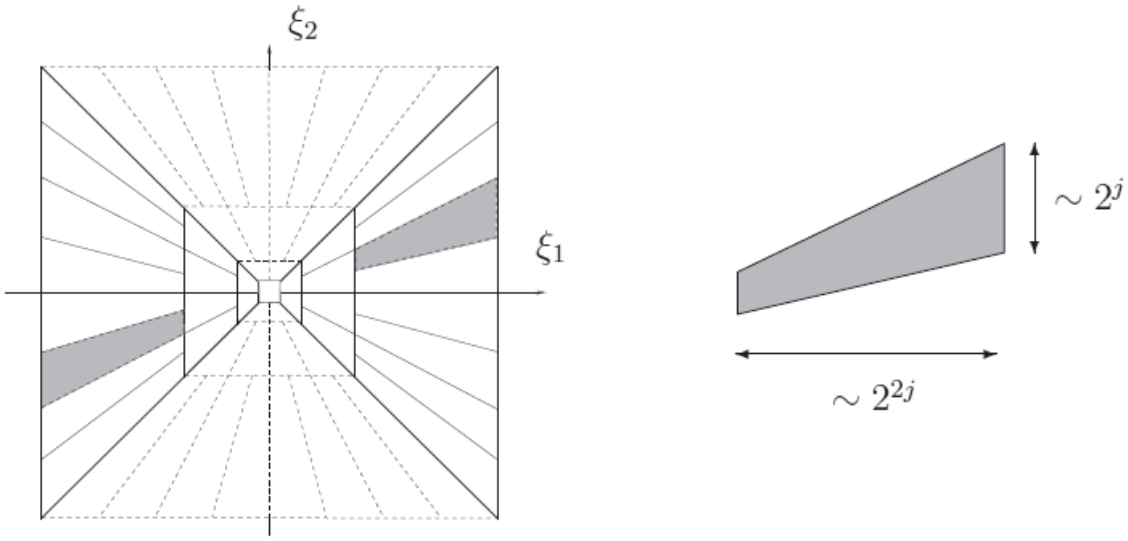
$$\sum_{j \geq 0} \left|\hat{\psi}_1(2^{-2j}\omega)\right|^2 = 1, \ for |\omega| \geq \frac{1}{8}, \text{and for each } j \geq 0, \tag{3.19}$$

$$\sum_{l=-2^j}^{2^j-1} \left|\hat{\psi}_2(2^j\omega - l)\right|^2 = 1, \ for |\omega| \leq 1 \tag{3.20}$$

From the conditions on the support of $\hat{\psi}_1$ and $\hat{\psi}_2$ one can easily observe that the functions $\hat{\psi}_{j,k,l}$ have frequency support:

$$\sup p\hat{\psi}^{(0)}_{j,k,,l} \subset \left\{ \begin{array}{l} (\xi_1,\xi_2) : \xi_1 \in [-2^{2j-1},-2^{2j-4}] \\ \cup [2^{2j-4},2^{2j-1}], \left| \frac{\xi_1}{\xi_2} + l2^{-j} \right| \le 2^{-j} \end{array} \right\} \tag{3.21}$$

That is, each element $\hat{\psi}_{j,k,l}$ is supported on a pair of trapezoids, of approximate size $2^{2j} * 2^{j}$ oriented along lines of slope $l2^{-j}$ (see Figure 6(b)).



(a) (b)

Figure – 6: (a) The tiling of the frequency plane induced by the Shearlet

(b) The frequency support of a Shearlet that satisfies parabolic scaling

The equations (3.20) and (3.21) imply that

$$\left\{ \begin{array}{l} \sum_{j\ge-} \sum_{l=-2^j}^{2^j-1} \left| \hat{\psi}^{(0)}(\xi A_0^{-j} B_0^{-l}) \right|^2 = \sum_{j\ge 0} \sum_{l=-2^j}^{2^j-1} \left| \hat{\psi}_1(2^{-2j}\xi_1) \right|^2 \\ \left| \hat{\psi}_2(2^j \frac{\xi_2}{\xi_1} - l) \right|^2 = 1 \end{array} \right\} \tag{3.22}$$

For $(\xi_1,\xi_2)\in D_0$, where $D_0=\left\{(\xi_1,\xi_2)\in\hat{\Re}^2:|\xi_1|\geq\frac{1}{8},\left|\frac{\xi^2}{\xi^1}\right|\leq 1\right\}$. That is, the functions $\left\{\hat{\psi}^{(0)}(\xi A_0^{-j}B_0^{-l})\right\}$ form a tiling D_0.This is illustrated in Figure 6(a). This property, together with the fact that $\hat{\psi}^{(0)}$ is supported inside $\left[-\frac{1}{2},\frac{1}{2}\right]^2$, implies that the collections:

$\left\{\psi^{(0)}_{j,l,k}(x)=2^{3j/2}\psi^{(0)}(B_0^lA_0^jx-k):j\geq 0,-2^j\leq l\leq 2^j-1,k\in Z^2\right\}$, is a Parseval frame for $L^2(D_0)^V=\left\{f\in L^2(\Re^2):\hat{f}\subset D_0\right\}$. Similarly, a Parseval frame can be constructed for $L^2(D_1)^V$,

Where D_1 is the vertical cone $D_1=\{(\xi_1,\xi_2)\in\hat{\Re}^2:|\xi_2|\geq\frac{1}{8},|\frac{\xi_1}{\xi_2}|\leq 1\}$. Let

$$A_1=\begin{pmatrix}2&0\\0&0\end{pmatrix},B_1=\begin{pmatrix}1&0\\1&1\end{pmatrix},$$

and $\psi^{(1)}$ be given by $\hat{\psi}^{(1)}(\xi)=\hat{\psi}^{(1)}(\xi_1,\xi_2)=\hat{\psi}_1(\xi_2)\hat{\psi}_2(\frac{\xi_1}{\xi_2})$, where, $\hat{\psi}_1,\hat{\psi}_2$ are defined as above. Then the collection

$$\left\{\psi^{(1)}_{j,k,l}(x)=2^{3j/2}\psi^{(1)}(B_1^lA_1^jx-k):j\geq 0,-2^j\leq l\leq 2^j-1,k\in Z^2\right\}\qquad(3.23)$$

is a Parseval frame for $L^2(D_1)^V$,

Finally, let $\psi \in C_0^{\infty}(\Re^2)$ be chosen to satisfy

$$G(\xi) = |\hat{\varphi}(\xi)|^2 + \sum_{j\geq 0}\sum_{l=-2^j}^{2^j-1} \left|\hat{\psi}^{(0)}(\xi A_0^{-j} B_0^{-l})\right|^2 \chi D_0(\xi) + \sum_{j\geq 0}\sum_{l=-2^j}^{2^j-1} \left|\hat{\psi}^{(1)}(\xi A_1^{-j} B_1^{-l})\right|^2 \chi D_1(\xi) = 1, \ for \xi \in \hat{\Re}^2, \tag{3.24}$$

where, χD denotes the indicator function of the set D. This implies that $\varphi \subset \left[-\frac{1}{8}, \frac{1}{8}\right]^2$, $with |\hat{\varphi}(\xi)| = 1 \ for \xi \in \left[-\frac{1}{16}, \frac{1}{16}\right]^2$, and the set $\{\varphi(x-k) : k \in Z^2\}$ is a Parseval frame for $L^2(\left[-\frac{1}{16}, \frac{1}{16}\right]^2)^v$. Observe that, by the properties of $\psi^{(d)}, d = 0,1,$ it follows that the function $G(\xi) = G(\xi_1, \xi_2)$ is continuous and regular along the lines $\xi_1 / \xi_2 = \pm 1$. It will be convenient to describe the collection of Shearlets presented above in a way which is more suitable to derive its numerical implementation. For $\xi = (\xi_1, \xi_2) \in \hat{\Re}^2, j \geq 0, l = -2^j, \ldots 2^j - 1,$ let

$$W_{j,l}^{(0)}(\xi) = \begin{cases} \hat{\psi}_2(2^j \xi_2 / \xi_1 - l) \aleph D_0(\xi) + \hat{\psi}_2(2^j \xi_1 / \xi_2 - l + 1) \aleph D_1(\xi) & if l = -2^j \\ \hat{\psi}_2(2^j \xi_2 / \xi_1 - l) \aleph D_0(\xi) + \hat{\psi}_2(2^j \xi_1 / \xi_2 - l - 1) \aleph D_1(\xi) & if l = 2^j - 1 \\ \hat{\psi}_2(2^j \xi_2 / \xi_1 - l) & otherwise \end{cases} \tag{3.25}$$

and

$$W_{j,l}^{(1)}(\xi) = \begin{cases} \hat{\psi}_2(2^j \xi_2 / \xi_1 - l + 1) \aleph D_0(\xi) + \hat{\psi}_2(2^j \xi_1 / \xi_2 - l) \aleph D_1(\xi) & if l = -2^j \\ \hat{\psi}_2(2^j \xi_2 / \xi_1 - l - 1) \aleph D_0(\xi) + \hat{\psi}_2(2^j \xi_1 / \xi_2 - l) \aleph D_1(\xi) & if l = 2^j - 1 \\ \hat{\psi}_2(2^j \xi_2 / \xi_1 - l) & otherwise \end{cases} \tag{3.26}$$

For $1-2^j \le l \le 2^j - 2$, each term $W_{j,l}^{(d)}(\xi)$ is a window function localized on a pair of trapezoids, as illustrated in Figure 6(a). When $l = -2^j \; or \; l = 2^j - 1,$ at the junction of the horizontal cone D_0 and the vertical cone D_1, $W_{j,l}^{(d)}(\xi)$ is the superposition of two such functions. Using this notation, for $j \ge 0, -2^j \le l \le 2^j - 1, k \in Z^2, d = 0,1$, we can write the Fourier transform of the Shearlets in the compact form

$$W_{j,l}^{(d)}(\xi) = 2^{3j/2} V(2^{-2j}\xi) W_{j,l}^{(d)}(\xi) e^{-2\pi i \xi A_d^{-j} B_d^{-l} k}, \tag{3.27}$$

Indeed, one can easily verify that

$$\sum_{d=0}^{1} \sum_{l=-2^j}^{2^j-1} \left| W_{j,l}^{(d)}(\xi_{1,\xi_2}) \right|^2 = 1, \tag{3.28}$$

And from this it follows that

$$|\hat{\psi}(\xi_1, \xi_2)|^2 + \sum_{d=0}^{1} \sum_{j \ge 0} \sum_{l=-s^j}^{2^j-1} |V(2^{2j}\xi_1, 2^{2j}\xi_2)| \left| W_{j,l}^{(d)}(\xi_{1,\xi_2}) \right|^2 = 1, \tag{3.29}$$

for $(\xi_1, \xi_2) \in \hat{\Re}^2$.

An $N * N$ image consists of a finite sequence of values, $\left\{ x[n_1, n_2]_{n_1,n_2=0}^{N-1,N-1} \right\}$ where $N \in \mathbb{N}$. Identifying the domain with the finite group Z_N^2, the inner product of image x, y: $Z_N^2 \to \mathbb{C}$ is defined as

$$(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} x(u, v) \overline{y(u, v)} \tag{3.30}$$

Thus the discrete analog of $L^2(\mathbb{R}^2)$ is $l^2\mathbb{Z}_N^2$. Given an image $f \in l^2(\mathbb{Z}_N^2)$, let $\hat{f}[k_1,k_2]$ denote its 2D Discrete Fourier Transform (DFT):

$$\hat{f}[k_1,k_2] = \frac{1}{N}\sum_{n_1,n_2=0}^{N-1} f[n_1,n_2]e^{-2\prod i(\frac{n_1}{N}k_1+\frac{n_1}{N}k_2)} \tag{3.31}$$

The brackets in the equations $[\cdot,\cdot]$ denote arrays of indices, and parentheses $(\cdot,\cdot)$ denote function evaluations. Then the interpretation of the numbers $\hat{f}[k_1,k_2]$ as samples $\hat{f}[k_1,k_2] = \hat{f}(k_1,k_2)$ is given by the following equation from the trigonometric polynomial.

$$\hat{f}(\xi_1,\xi_2) = \sum_{n_1,n_2=0}^{N-1} f[n_1,n_2]e^{-2\prod i(\frac{n_1}{N}\xi_1+\frac{n_1}{N}\xi_2)} \tag{3.32}$$

First, to compute

$$\hat{f}(\xi_1,\xi_2)\overline{V(2^{-2j}\xi_1,2^{-2j}\xi_2)} \tag{3.33}$$

In the discrete domain, at the resolution level j, the Laplacian pyramid algorithm is implemented in the time domain. This will accomplish the multi scale partition by decomposing $f_a^{j-1}[n_1,n_2], 0 \le n_1,n_2 < N_j - 1$, into a low pass filtered image $f_a^j[n_1,n_2]$, a quarter of the size of $f_a^{j-1}[n_1,n_2]$,and a high pass filtered image $f_d^{j-1}[n_1,n_2]$.Observe that the matrix $f_a^{j-1}[n_1,n_2]$ has size $N_j * N_j$,where $N_j = 2^{-2j}N$ and $f_a^0[n_1,n_2] = f[n_1,n_2]$ has size $N*N$. In particular,

$$\hat{f}_d^j(\xi_1,\xi_2) = \hat{f}(\xi_1,\xi_2)\overline{V(2^{-2j}\xi_1,2^{-2j}\xi_2)} \tag{3.34}$$

Thus, $f_d^j[n_1,n_2]$ are the discrete samples of a function $f_d^j[x_1,x_2]$, whose Fourier transform is $\hat{f}_d^j(\xi_1,\xi_2)$. In order to obtain the directional localization the DFT on the pseudo-polar grid is computed, and then one-dimensional band-pass filter is applied to the components of the signal with respect to this grid. More precisely, the definition of the pseudo-polar co ordinates $(u,v)\in\mathbb{R}^2$ as follows:

$$(u,v) = (\xi_1,\frac{\xi_2}{\xi_1}), if(\xi_1,\xi_2)\in D_0 \tag{3.35}$$

$$(u,v) = (\xi_1,\frac{\xi_1}{\xi_2}), if(\xi_1,\xi_2)\in D_1 \tag{3.36}$$

After performing this change of coordinates, $g_j(u,v) = \hat{f}_d^j(\xi_1,\xi_2)$ is obtained and for $l = 1-2^j,..2^j-1$:

$$\begin{aligned}\hat{f}(\xi_1,\xi_2) &= \overline{V(2^{-2j}\xi_1,2^{-2j}\xi_2)W_{jl}^{(d)}(\xi_1,\xi_2)} \\ &= g_j(u,v)\overline{W(2^j v-l)}\end{aligned} \tag{3.37}$$

This expression shows that the different directional components are obtained by simply translating the window function W. The discrete samples $g_j[n_1,n_2] = g_j(n_1,n_2)$ are the values of the DFT of $f_d^j[n_1,n_2]$ on a pseudo-polar grid. That is, the samples in the frequency domain are taken not on a Cartesian grid, but

along lines across the origin at various slopes. This has been recently referred to as the pseudo-polar grid. One may obtain the discrete Frequency values of f_d^j on the pseudo-polar grid by direct extraction using the Fast Fourier Trans-form with complexity $ON^2 \log N$ or by using the Pseudo-polar DFT.

A special form of DST named NSST is adopted for the proposed sports video classification system which exhibits excellent shift invariance and highly redundant decomposition. Also, it eliminates the effect of acquisition noise effectively. The non-subsampled version is easily implemented from DST by removing up and down sampler in the Laplacian pyramid decomposition structure.

3.4 KNN CLASSIFIER

Machine learning is a branch of computer science where algorithms are used that allows a computer to extract relevant data from patterns and use that data to make intelligent decisions. It is especially useful in situations where patterns may be very complex and not feasible for a human to develop instructions for every possible situation or even recognize the patterns. In the case of classification, the decision would be to what class the data belongs. Computer learning falls under two broad categories: supervised and unsupervised learning. Unsupervised learning is used when there is no class data available for a dataset. In this case objects are partitioned so as to best cluster the data.

Supervised learning is used in situations where there is some sample data available with appropriate decisions that can be used as a training set. Classifiers often operate in two phases. The training phase is where the relationship between certain

features and outcomes is determined and optimized. This is often a long and computationally intensive process. The operating phase is when the training data is put to use to classify an object. This is usually much quicker. Possibly the most important component of a classification routine is the feature vector. The feature vector is a set of scalar quantities that describe an object. The success of a machine learning algorithm depends on the choice of a feature vector. The algorithms work by comparing the feature vector of a test object with those of objects already classified. If the data in the feature vector is not appropriate for the classification task, it will fail.

Usually the initial choice of a feature vector is not the best one. Some features may not contribute to the classification task or might be made redundant by other features. Attempting to classify with these features can not only significantly increase computation time, but can make classifications less accurate. In order to mitigate this problem, a feature reduction step should take place. A good feature reduction process will result in faster learning due to less data, higher accuracy and better generalization to other data sets. There are two approaches of choosing a feature vector from all available features, top- down and bottom-up. The former takes a vector of all features and removes them one by one, testing the classification accuracy at each step. The bottom-up approach does the opposite. It starts with an empty vector and adds features to it one by one. Classifiers can be either soft or hard. A Hard classifier classifies an object without giving a probability. The assumption is made that an object that meets a certain criteria always belongs to a particular class. Soft classifiers give a probability of their classification. The assumption made is that sometimes objects with similar features may belong to different class. The algorithm for the basic KNN classifier is as follows:

BEGIN

Input: $Data = \{(x_1, c_1), ..., (x_N, c_N)\}$ and $x = (x_1, ..., x_n)$ new instance to be classified

1. For each labeled instance (x_i, c_i) calculate $d(x_i, x)$.

2. Order $d(x_i, x)$ from lowest to highest, $(i = 1, ..., N)$

3. Select the K nearest instances to $x: Data_x^K$

4. Assign to X the frequent class in $Data_x^K$

END

The proposed sports video classification system uses the following distance measures to evaluate the performance.

3.4.1 EUCLIDEAN DISTANCE

The Euclidean distance measures calculation is as follows: Let us consider $u = (x_1, y_1)$ and $v = (x_2, y_2)$ are two points. The Euclidean distance between these two points is given by

$$Euclidean\ distance\ (u, v) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \qquad (3.38)$$

If the points have n-dimensions such as $u=(x_1, x_2, x_3, \ldots x_n)$ and $v=(y_1, y_2, y_3, \ldots y_n)$ then the generalized Euclidean distance formula between these points is

$$Euclidean\ distance\ (u, v)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2+\ldots+(x_n-y_n)^2} \quad (3.39)$$

3.4.2 CITY BLOCK DISTANCE

Let us consider $u=(x_1, y_1)$ and $v=(x_2, y_2)$ are two points. The city block distance between these two points is given by $cityblock(u, v)=|x_1-x_2|+|y_1-y_2|$. If the points have n-dimensions such as $u=(x_1, x_2, x_3, \ldots x_n)$ and $v=(y_1, y_2, y_3, \ldots y_n)$ then the generalized city block distance formula between these points is

$$cityblock(u, v)=|x_1-y_1|+|x_2-y_2|+\cdots\ldots\ldots|x_n-y_n|=\sum_{i=1}^{n}|x_i-y_i| \quad (3.40)$$

3.4.3 COSINE DISTANCE

Let us consider X and Y where $X=(x_1, x_2, x_3, \ldots x_n)$ and $Y=(y_1, y_2, y_3, \ldots y_n)$ then $\cos\theta$ may be consider as the cosine of the vector angle between X and Y in n dimension. The cosine of the vector angle between X and Y is given by

$$COSINE(X, Y)=\frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\sqrt{\sum_i y_i^2}} \quad (3.41)$$

One important property of cosine angle is that it gives a metric of similarity between two vectors unlike Manhattan distance and Euclidean distance, both of which give metrics of dissimilarities. Also $COSINE(X, Y) \in [0,1]$, this makes it easy to combine the distance between two images using multiple features.

3.4.4 CORRELATION DISTANCE

A correlation is single number that describes the degree of relationship between two variables X and Y where $X = (x_1, x_2, x_3, \ldots x_n)$ and $Y = (y_1, y_2, y_3, \ldots y_n)$. The correlation r between X and Y is defined by

$$r = \frac{N\sum xy - \left(\sum x\right)\left(\sum y\right)}{\sqrt{\left[N\sum x^2 - \left(\sum x\right)^2\right]\left[N\sum y^2 - \left(\sum y\right)^2\right]}} \tag{3.42}$$

3.5 SUPPORT VECTOR MACHINE

In the following sections, a learning algorithm called support vector machine (SVM) is discussed. It has many good properties which are described in (Erasto, 2001) as follows: Basically, it is a linear discriminate function and has fast shattering dimension and a low Vapnik Chervonenkis. The theories of large margin classifier and Vapnik Chervonenkis theory can partially justify the usage of SVM. The main properties of SVM are as follows:

1. It performs well and good in real world situations as presented by Gunn (1998).

2. It is easy to understand and implement.

3. It is robust as proposed by (Smola et al., 2000).

4. It has a global solution while others have more than one local solution as presented by Crisp (1999).

3.5.1 THE LINEARLY SEPARABLE CASE

Let us consider, a linear discriminant function for a linearly separable training set $Z = \{(x_i, y_i)\}_{i=1}^{n}$ of the form

$$x \mapsto w^T x + b,\ w \in \Re^d,\ b \in \Re \tag{3.43}$$

for which the corresponding decision function $t = \text{sgn}(w^T x + b)$ has the property $\hat{e}_n(t) = 0$. In order to separate the training set with no error, many linear classifiers are available. However the main aim is to choose the best one based on margin (Erasto, 2001). To maximize, the minimum margin w and b the training set, Z must satisfy the following condition

$$\min_{x_i \in X} \left| w^T x_i + b \right| = 1 \tag{3.44}$$

Hyperplanes of the above form are usually called canonical hyper planes. The geometric margin $\bar{\gamma}$ is defined as

$$\bar{\gamma}_i = \frac{y_i (w^T x_i + b)}{\| w \|_2} \tag{3.45}$$

It can be seen that maximizing the minimal geometric margin reduces to minimizing the norm of the weight vector. The optimal separating hyper plane is the

one that satisfies the equation (3.46) in which the geometric minimum margin is maximized.

$$y_i(w^T x_i + b) \geq 1,\ i=1,\ldots,n \tag{3.46}$$

The problem of finding the optimal separating hyper plane can be written in the form

$$\begin{aligned} &\min_{w,b} && \frac{1}{2}\|w\|_2^2 \\ &s.t. && y_i(w^T x_i + b) \quad i = 1,\ldots,n \end{aligned} \tag{3.47}$$

Instead of solving the above minimization problem, one can solve the dual maximization problem as given in (3.48) because of the fact that the dual problem is independent of the dimension of x and instead scales with the number of observations. This is a very good property, especially when x is high dimensional and the number of observations remains small.

$$\begin{aligned} &\max_{\alpha} && \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i a_j y_i y_j (x_i^T x_j) \\ &s.t. && \alpha_i \geq 0, && i=1,\ldots,n \\ & && \sum_{i=1}^{n} \alpha_i y_i = 0 \end{aligned} \tag{3.48}$$

Solving the constrained optimization problem (3.47) gives a solution $\hat{\alpha}$ in terms of which the optimal weight vector $\hat{w}$ is

$$\hat{w} = \sum_{i:\hat{a}_i>0} \hat{\alpha}_i x_i y_i \tag{3.49}$$

and the optimal bias term $\hat{b}$ is given by

$$\hat{b} = -\frac{1}{2}\left[\min_{y_i=1}(\hat{w}^T x_i) + \max_{y_i=-1}(\hat{w}^T x_i)\right] \tag{3.50}$$

For numerical reasons, it is often better to use more support vectors for calculating $\hat{b}$ and in the case where the number of support vectors from classes -1 and 1 are equal we can set

$$\hat{b} = -\frac{1}{2\{|i|\,\hat{\alpha}_i > 0\}|}\sum_{i:\hat{\alpha}_i>0} \hat{w}^T[x_i 1(y_i = 1) + x_i 1(y_i = -1)] \tag{3.51}$$

Training vectors x_i for which they $\hat{\alpha}_i$ are strictly positive are called support vectors. From the Kuhn-Tucker complementary condition, it follows that the points x_i for which α_i are positive must satisfy

$$y_i(\hat{w}^T x_i \hat{b}) = 1 \tag{3.52}$$

So that all support vectors lie on a hyper plane at functional distance 1 from the optimal separating hyper plane as shown in Figure 7. Because of this property, the number of support vectors can be very small. Another important property of support vectors is the following: if we remove all the data points (x_i, y_i) for which the x_i's are not support vectors, the same solution $\hat{w}, \hat{b}$ is obtained. Thus, the support vectors give, in some sense, a compact representation of the data. Also the SVM ignores non-informative data and considers only informative data points from point of view of the optimal hyper plane, that is, points lying on the hyper planes which have unit functional distance from the optimal separating hyper plane.

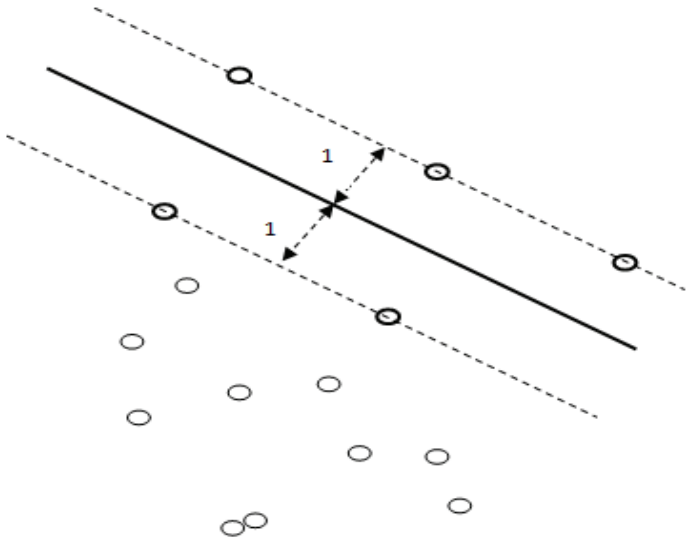


Figure – 7: The optimal separating hyper plane for linearly separable case.

In Figure 7, the solid line is the optimal separating hyper plane. Support vectors (highlighted with an extra circle) are the points which lie on hyper planes (the dashed lines) that have unit (functional) distance to the optimal separating hyper plane. Losing the quality of the estimate is very useful, especially when very large datasets are considered. Of course, this property can be used to classify a previously unseen vector x' only after the optimal hyper plane has been found and training must be done using the whole dataset. For the optimal parameters, $\hat{w}, \hat{b}$ the classifier t is given by

$$t(x) = \text{sgn}(\hat{w}^T x + \hat{b}) \tag{3.53}$$

Another possibility is to use the optimal parameters as follows,

$$t(x) = h(\hat{w}^T x + \hat{b}), \quad where\ h(s) \begin{cases} -1 & if\ s \le -1, \\ 0 & if\ -1 \le s, < 1, \\ 1 & if\ s > -1 \end{cases} \tag{3.54}$$

which can be interpreted to give a naive posterior probability estimate of classification.

3.5.2 THE LINEARLY NON-SEPARABLE CASE (SOFT MARGIN)

Let us assume that one cannot separate the data without a misclassification error using the class of linear classifiers but still try to find a linear discriminant function. By the error of observation ξ the amount by which the discriminant function fails to reach the (functional) unit margin is shown in Figure 8. Formally,

$$\xi_i := \max\{0, 1 - y_i(\hat{w}^T x_i + \hat{b})\}, \tag{3.55}$$

and misclassification takes place when $\xi_i > 1$.

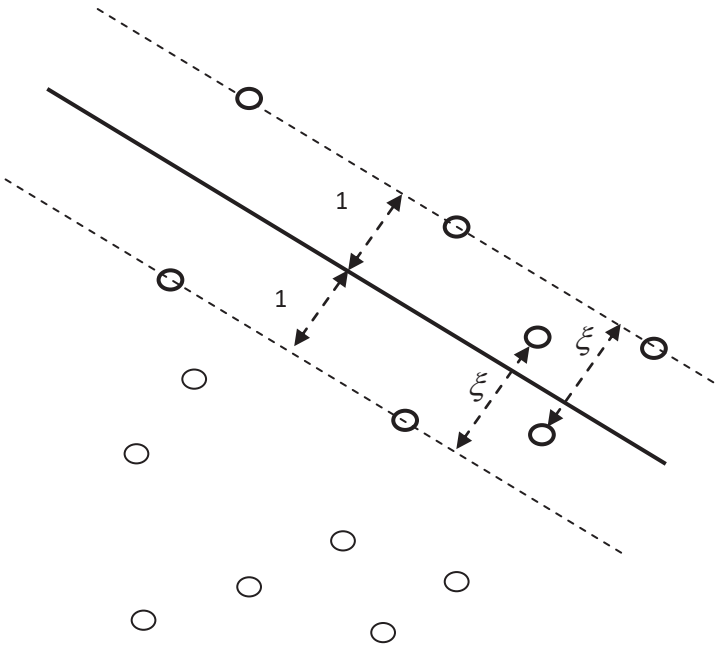


Figure – 8: The optimal separating hyper plane for linearly non-separable case

In the linearly non-separable case, the optimal hyper plane is defined to be the hyper plane which maximizes the geometric margin and minimizes some functional $\theta(\xi)$ of the errors. The functional $\theta(\xi)$ is usually of the form

$$\theta(\xi) = \sum_{i=1}^{n} \xi_i^{\sigma} \tag{3.56}$$

where σ is the some small positive constant. Usually the value $\sigma = 1$ is used since the corresponding dual does not involve ξ and therefore offers a simpler optimization problem. The constraint now assumes the form

$$y_i(w^T x_i + b) \geq 1 - \xi_i,\ i=1,\ldots,n \tag{3.57}$$

Where, $\xi_i \geq 0$. Selecting $\sigma = 1$ the optimization problem is

$$\min_{w,b,\xi} \frac{1}{2}\|w\|_2^2 + C\sum_{i=1}^{n}\xi_i$$

$$s.t. \quad y_i(w^T x_i + b) \geq 1 - \xi_i, \qquad i=1,\ldots,n$$

$$\xi_i \geq 0, \qquad i=1,\ldots,n \tag{3.58}$$

Here C is a positive trade-off parameter which intuitively defines how important it is to avoid misclassification errors. Crisp (1999) have suggested in that every observation has its own trade-off parameter C_i, the second term in the objective

being $\sum_{i=1}^{n} C_i \xi_i$. Again, instead of solving direct optimization problem (3.58), consider the corresponding dual problem with the objective function

$$W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i^T x_j), \tag{3.59}$$

to be maximized with the constraints

$$0 \le \alpha_i \le C, \qquad i=1,..., n$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{3.60}$$

The only difference between the duals in the linearly separable and the non-separable cases is that in the non-separable case the coefficients α_i have an upper bound C. From the Kuhn-Tucker complementary condition it also follows that $\hat{\alpha}_i = c$ if and only if $\xi_i > 0$ and thus the vectors x_i with $\xi_i > 0$ are support vectors. The standard form of the SVM can be used in two-way classification. However, in real-life situations it is often necessary to separate more than two classes at the same time. The well-known example is the classification of Hand written characters. The standard SVM can be extended from the binary two-class problem to classification tasks with $k > 2$ classes. New theoretical foundations of multiclass classification systems are presented by (Guermeur et al., 2000).

3.5.3 SVM CLASSIFICATION

The simplest extension of the SVM to a k-class problem is to separate the observations from class j from the rest for every $j = 1,\ldots,k$. Here the "rest" means that all the observations from classes other than j are combined to form one class. The optimal hyper plane that separates samples from the class j and the combined class is found by using the standard SVM approach (Guermeur et al., 2000). The optimal separating hyper plane discriminating the class j and the combined class is defined by

$$x^T \hat{w}^j + \hat{b}_j, \qquad j = 1,\ldots,k \tag{3.61}$$

where, the superscript in $\hat{w}^j$ stands for the class which should be separated from the other observations. The decision rule t^j that assigns the vector x to the class j or to the combined class is

$$t^j(x) = \operatorname{sgn}(g^i(x) \tag{3.62}$$

where, $g^j(x) = x^T \hat{w}^j + \hat{b}_j$. After all the k optimal separating hyper planes defined by $(\hat{w}^j, \hat{b}_j)$, $j = 1,\ldots,k$ have been found, the final classifier t_k is

$$t_k(x) = \operatorname{argmax}_j(t^j(x)) \tag{3.63}$$

This approach of assigning the class label using argmax-rule is usually called voting. In this case voting is performed by giving every classifier a vote of size one and the unknown label is decided to be the index of the classifier that gives the only positive vote. If there are no positive votes or if there is more than one classifier

t_j with positive vote, then no decisions about the class label is made. The one-versus-all approach has drawbacks that can be rather serious. In (3.63) the argmax-rule is not well-defined since there is not always a unique solution.

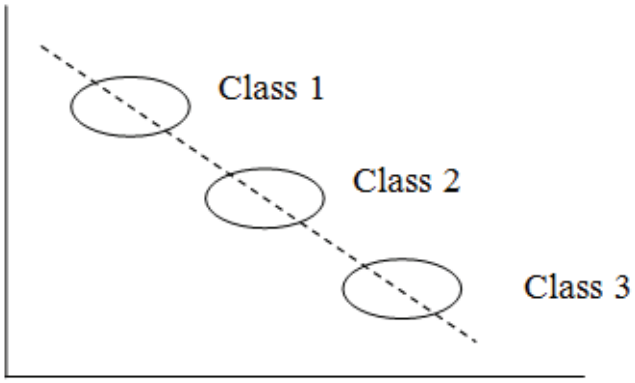


Figure – 9: Three classes aligned approximately along a line.

The main difficulty in this approach is that the outputs of the classifiers t^j are binary values, e.g., $t^j(x) = t^h(x)$ even if $g^j(x) >> g^h(x) > 0$. The usual way to handle this problem is to ignore the sign-operator and to use the argmax-rule for the $g^j(x)$'s. In this approach the index of the largest component of the discriminant vector $(g^1(x).....g^k(x))$ is assigned to the vector X. Even with the winner-takes-all modification, the one-versus-all approach has problems which occur for example when the classes are approximately aligned along a line. Figure 9 illustrates such a situation where the class 2 cannot be reasonably separated.

In this chapter, multi resolution analysis based on NSST is discussed. This transform is applied to various image processing applications such as compression (Lim, 2009), de-noising (Easley et al., 2006), enhancement (Negi and Labate, 2010)

and edge detection (Yi et al., 2009).SVM is a well known machine learning approach to automated learning of pattern classifiers.

3.6 TYPICAL COMPUTER VISION SYSTEM

Figure 10 shows a computer vision system and its various components. Firstly, the images are acquired by using a standard camera. In the preprocessing stage, the acquired image is de-noised, enhanced or segmented by the properties of each region, depending on particular application. The next stages are feature extraction and classification where the classifying patterns are identified by the extracted features.

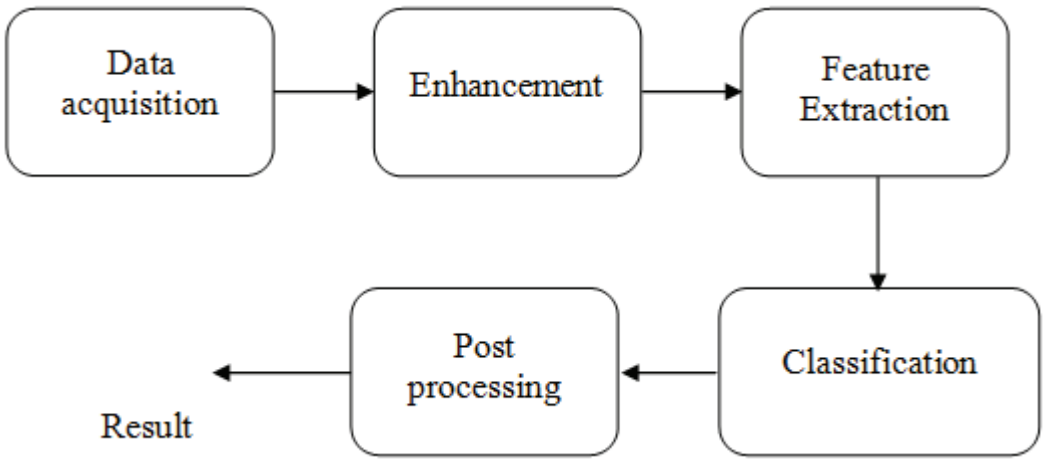


Figure – 10: Components of a computer vision system

3.7 PROPOSED SPORTS VIDEO CLASSIFICATION SYSTEM

The automatic classification of sports video is growing exponentially due to its tremendous commercial demand among the various video collections such as commercial, cartoon, drama and news. It aids in efficient storage, quick browsing, and retrieval of large collection of video data without losing important aspect. Video classification provides automatic interpretation of video objects to assist computer

vision and automatic management system. On account of similarity between various sports video, extraction of cognitive information is considered as a challenging task in sports video classification.

In this section, an efficient sports video classification system is proposed using the distribution of edge strengths extracted by exploiting NSST. In addition, multi class SVM classifier is employed for automated video classification. On account of current terrific interests on sports, there are five sports video such as cricket, volleyball, basketball, football and tennis are considered for this study. The success of any pattern recognition system relies on the appropriate design of two computational modules: feature extraction and classification. These two modules are discussed in the following subsections in detail.

3.7.1 FEATURE EXTRACTION

In any machine learning and pattern recognition approaches, feature extraction is considered as a critical process because the features obtained from this process directly influence the efficacy of the classification process. Also, it is defined as the first stage of intelligent image analysis, which tends to remove the redundant data and posses more intrinsic content of the original data.Thus, the task of feature extraction emphasizes the significant image information.

Feature extraction serves not only to reduce the dimensionality but also to capture the characteristics of the input pattern. The selected features are very effective for preserving the class separability. Hence, the extracted features with high discriminative nature will assist the classification system effectively while the lack of discrimination reduces the accuracy of the classification system. Figure 11 depicts the

various computational blocks of the proposed feature extraction module. The property of the selected features must satisfy the following conditions:

- The features should carry enough information about the each sports video.
- The feature should be differentiated from one category to all others and vice versa.
- They should be simple and easy to compute.

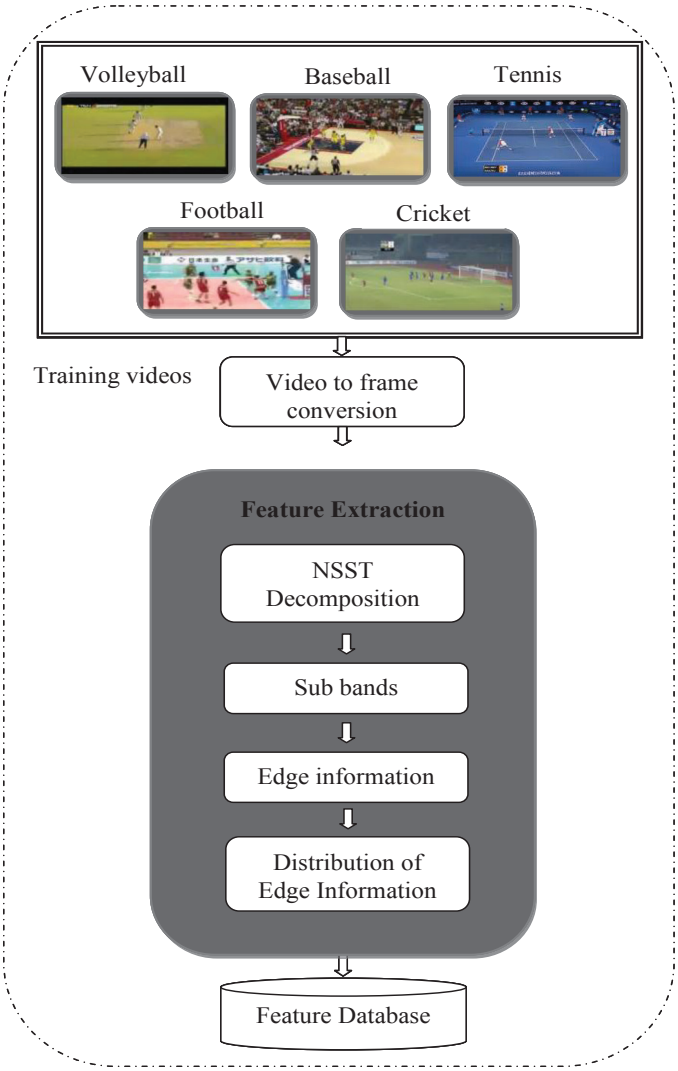


Figure – 11: Proposed sports video classification system - Feature extraction phase

The efficacy of the proposed sports video classification mainly depends on the discriminative nature of the features/attributes used for the representation of video contents. The efficient video content discrimination is achieved by the proposed system by exploiting multi-scale geometric characteristics of NSST as feature extraction technique. The optimal representation of image edges and the geometric features captured from the input multidimensional data by NSST can improve the classification accuracy of the proposed sports video classification system. The proposed system uses NSST as feature extraction technique. The reasons behind them are listed below:

- It possesses efficient accuracy for detection of edge orientation. The geometry of edges information is precisely captured in NSST transform by exploiting anisotropic dilations and multiple orientations.

- It is a multi-scale transform and has well organized multi-scale structure. Also, the affine mathematical structure of NSST is same as traditional wavelets.

- It provides a stable decomposition and reconstruction algorithm for video frames with very low complexity.

The edges features in an image are considered as prominent features for the analysis and recognition of many image processing and computer vision algorithms. In order to extract the edge features, the recent multi-scale mathematical framework named NSST is exploited. The multi-scale transformation is applied to each frame of training video at pre defined level of decomposition and directions. The obtained NSST sub-bands have edge strength information which is considered for feature set

evaluation. Figure 12 shows the decomposition of input video frames from various sports categories by NSST at 2-level 2-direction.

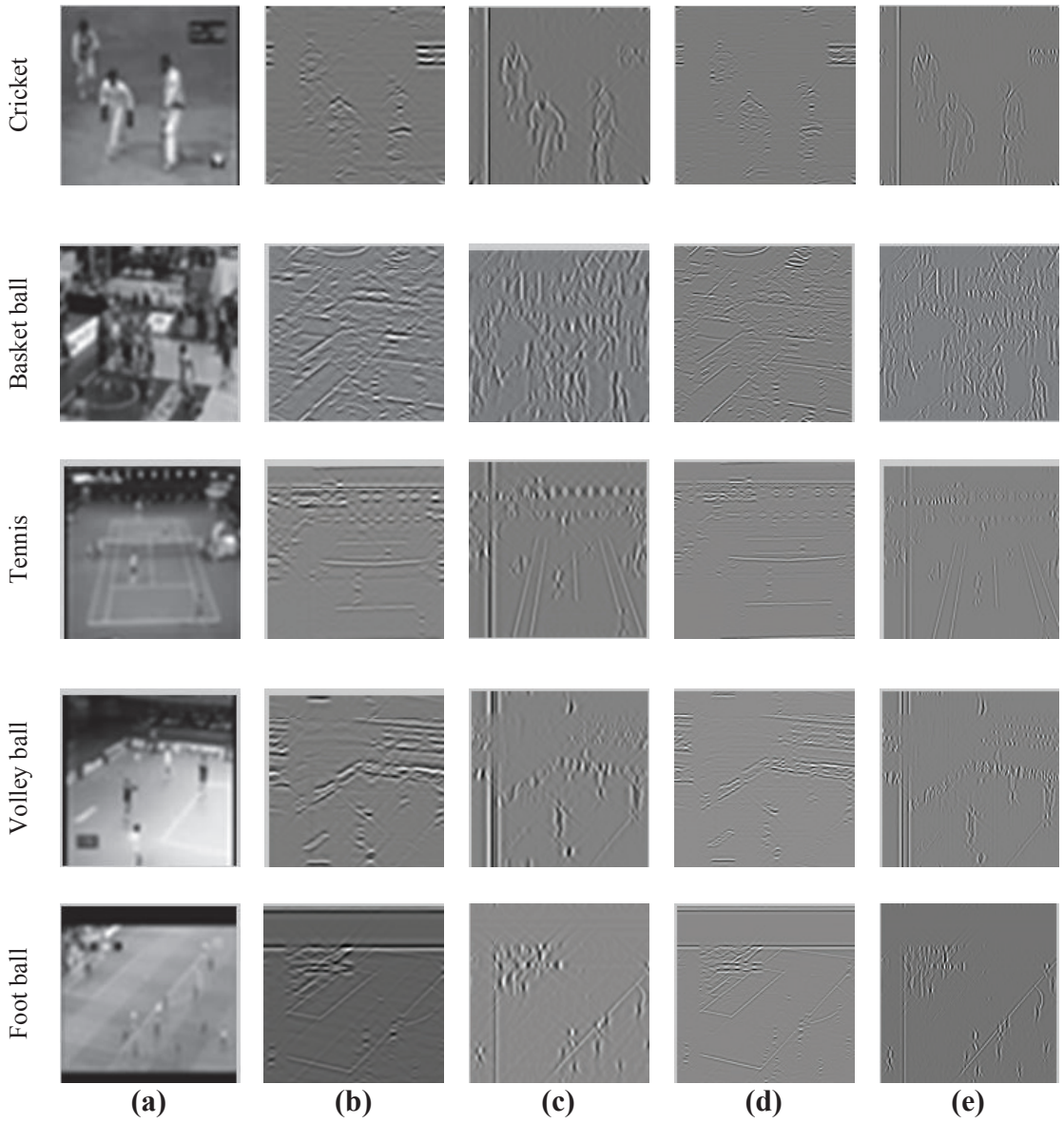


Figure – 12: NSST sub-bands at 2-level with 2-direction (a) sub-bands of low frequency (b) - (e) Sub-bands of high frequency

From Figure 12, it is observed that the edges of each sports video are clearly extracted by the application of NSST decomposition. The high dimensional edge strength features (size of each sub-band is same as size of the input frame) of

each sub-band are reduced by computing the distribution of edge strengths into 10 bins based on their appropriate magnitude ranges. This means that each video frame is represented by 10 features. Finally, the mean of distribution of edge strengths is computed as each video consists of 500 frames and used as features by the proposed system. To find the feature set that provides paramount discriminating power, the process of NSST is tested by increasing the decomposition level with various directions from 2 to 64

3.7.2 CLASSIFICATION

The classification task is the final stage of the proposed NSST and SVM classifier based sports video classification system where the unknown sports video is labeled into one of the predefined class of sports video category. Most of the classification system often operates in two phases: The training phase is where the relationship between certain features and outcomes is determined and optimized. This is often a long and computationally intensive process. The classification phase is when the training data is put to use to classify an object. This is usually much quicker. Supervised model of multiclass SVM classifier is employed in this proposed method for its generalization and discriminative learning approaches. The classification module of the proposed sports video classification system using NSST and multiclass SVM-RBF classifier is shown in Figure 13.

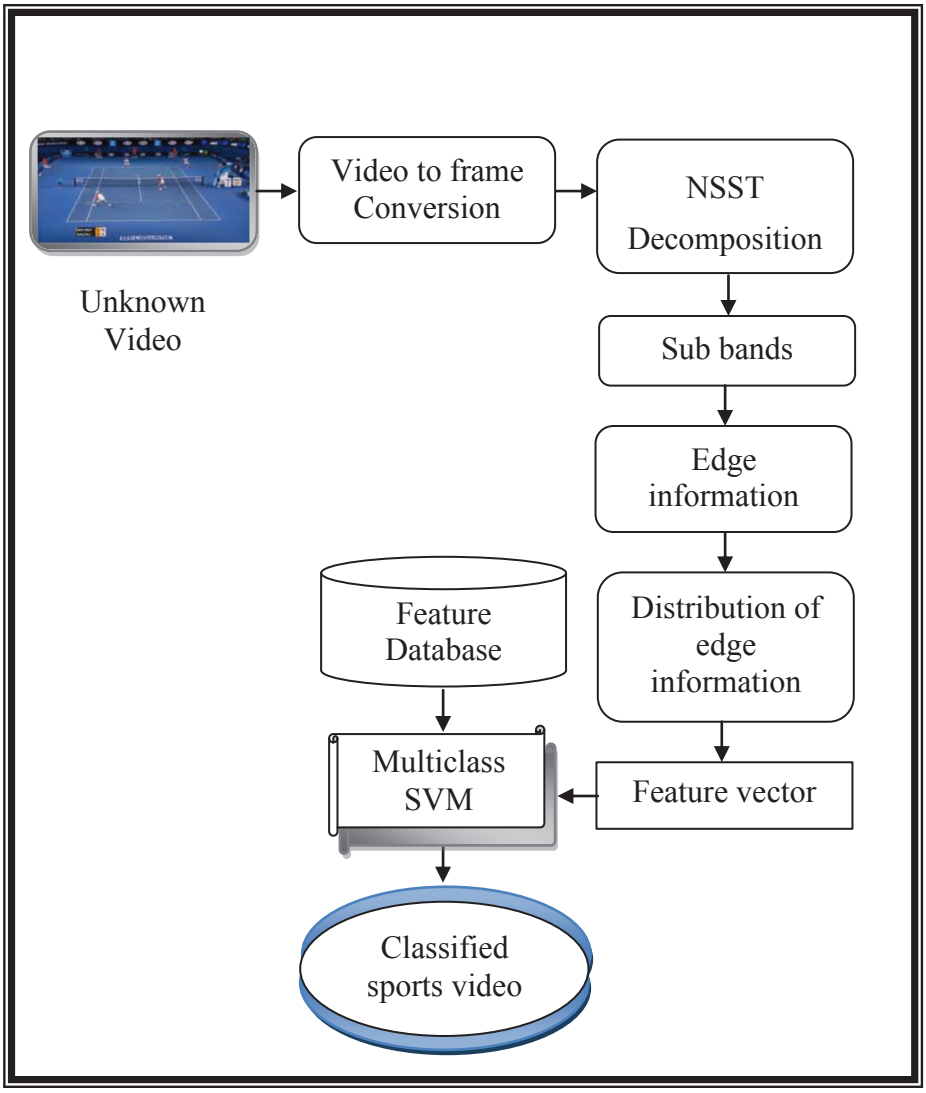


Figure – 13: Classification phase of the proposed sports video classification system

While comparing SVM with Artificial Neural Network (ANN), SVM gives the better performance in most of the cases and they are furnished below:

- SVM is very simple to analyze and derive theoretically than ANN.
- In the learning stage, SVM maps the training data to high dimensional space using various kernels like linear, radial basis functions. Finally a hyper plane

is selected that separates the training data which has maximum margin and minimum error

- "Error optimization" is the main difference between SVM and ANN.
- The main objective of learning stage in ANN is to minimize the error for a particular set of weight values whereas in SVM is to adjust the capacity of the machine.
- The number of hidden layers in ANN is same as support vectors in SVM.

In this study, the multiclass SVM classifier uses Radial Basis Function (RBF) kernel for better classification. The reason is that the RBF-SVM is suitable if the number of features is very low. The linear mapping does not improve the performance if the number of features is very low and it is a time consuming process to map the data. The SVM algorithm is as follows.

1. Calculate the RBF kernel based on the training samples and ensure that the function is symmetric.
2. Create the Hessian matrix based on kernel by using the targets or groups.
3. Calculate the optimal separating hyper plane by minimizing the cost function (3.49) using quadratic programming.
4. Select the non-zeros of values that minimize the cost function as support vectors.
5. Calculate the parameters of the separating line from the support vectors.

6. Classify the test samples by using the support vectors and the parameters of the separating line. If the points lie on the boundary, then it is assigned to one class, or else assigned to another class.

The unknown video undergoes into feature extraction as same as training video feature extraction. The extracted feature from test video and stored feature database is given to the multi class SVM classifier, where the unknown video is categorized by it. For classification, SVM separates the features into given number of classes by constructing hyper plane.

3.8 EXPERIMENTAL SETUP

3.8.1 PROGRAMMING ENVIRONMENT

The simulation of the proposed sports video classification system is implemented in MATLAB R2013b and tested on Intel architecture under the windows operating system. MATLAB was created by Cleve Moler, the chairman of the computer science department in the late 1970s at the University of New Mexico. It is a high level language used for visualization, numerical computation and programming. It can be used in various applications such as image and video processing, signal processing, control systems and communications.

3.8.2 PERFORMANCE MEASURES

The performance of the proposed sports video classification system discussed in section 3.7 is determined by confusion matrix and classification accuracy. The definition for these two performance measures are discussed below.

3.8.2.1 CLASSIFICATION ACCURACY

In data mining, the most common parameter used to assess the performance of any system is classification accuracy. It gives the accuracy of the proposed classification system. The classifier performance is better if it has higher classification accuracy and given in eq. (3.58).

$$Accuracy\,(\%) = \frac{correctly\ classified\ \ sports\ \ videos}{total\ sports\ \ videos\ tested} X100 \quad (3.64)$$

3.8.2.2 CONFUSION MATRIX

For a two class problem, confusion matrix is constructed by using the classifier outcome with the actual one. Table I shows the confusion matrix.

TABLE – I CONFUSION MATRIX

Test outcome	**Actual Class**	
	Class A	**Class B**
Class A	True Positive	False Positive
Class B	False Negative	True Negative

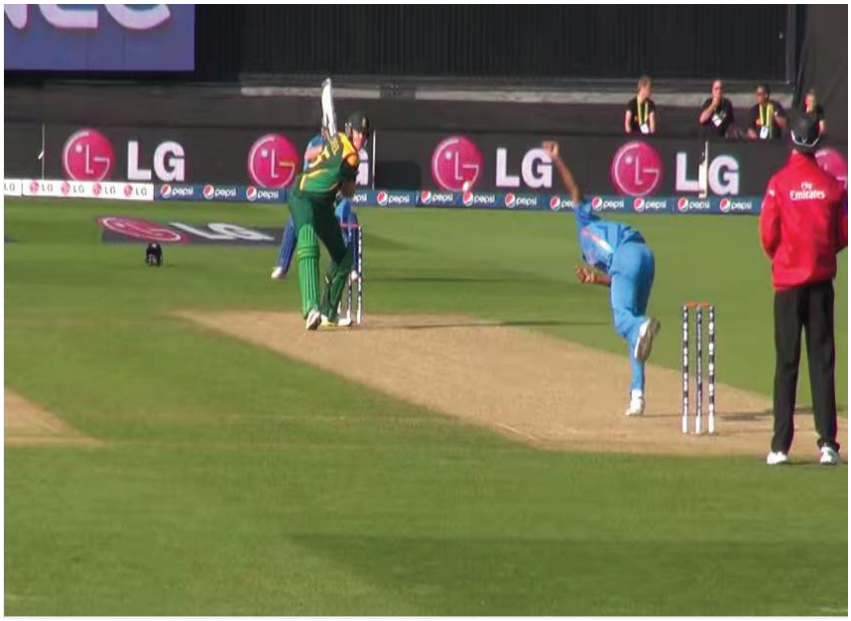
In table I

- **True positive**: class 1 object is correctly classified.
- **False Positive**: class 2 object is incorrectly classified
- **True Negative**: class 2 object is correctly classified
- **False Negative**: class 1 object is incorrectly classified

3.8.3 DATA SET

The evaluation of the proposed NSST and multiclass SVM classifier based sports video classification system is carried on a database which has five kinds of sports video such as volleyball, basketball, tennis, cricket and football. The data sets

are obtained from TV broadcast channels in different sessions. Totally, 500 video clips are collected for these experiments, in which 100 video clips are selected for each sports video. Each of the sports clip composed of 500 frames at the resolution of 128 x 128 pixels with 20 seconds duration. Among 100 video clips from each sport genre, 50 video clips are used to train the multiclass SVM classifier and remaining 50 video clips are used to test the proposed system by random selection process. Figure 14 shows the sample frames from five sports category: cricket, basketball, tennis, volley ball and foot ball videos in the database.



(a)


SAC 5 LAC 7 1ST 7:59 03
FOXSPORTSWEST.COM

(b)


ROLEX
ROLEX
BNP PARIBAS
ATP WORLD TOUR
SHANGHAI
SHANGHAI
FEDERER 40
MURRAY 60 Ad

(c)

(d)

(e)

Figure – 14: (a) – (e) Sample video frames from five sports categories