International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

A Scale Invariant Human Motion Detection System using Wavelet Based Feature Extraction

K. Padma Vasavi

***Abstract* – Visual analysis of human motion is one of the most active research topics in computer vision. This strong interest is motivated by a wide range of applications like intelligent video surveillance. There is a need for a motion detection system in which action representation and search scheme shall be able to tolerate enormous variance. In this paper a human motion detection system based on the 2-D wavelet features of history edge images of motion is proposed. This system can effectively encode shape and motion patterns of the image. Furthermore, a booster tree classifier is used for classifying different motions in a scene.**

***Keywords* – Action, Edge Map, Feature Extraction Thresholding, Wavelets.**

I. INTRODUCTION

Automated video surveillance addresses real-time observation of people and vehicles within a busy environment, leading to a description of their actions and interactions [1]. The technical concepts include moving object detection and tracking, object classification, human motion analysis, and activity understanding, touching on many of the core topics of computer vision, pattern analysis, and artificial intelligence. Among the techniques used in the video surveillance, human motion detection is a very important step because it is the first step for any automatic video surveillance system. Motion detection is defined as the action of sensing physical movement in a given area. Human motion detection algorithms aim at finding moving human beings in a given scenario [6]. They are also concerned in computing trajectories of moving human and identifying the unusual behavior in a given area. But, perceiving the human motion is very difficult because, people come in all different shapes, sizes and color. They wear vastly different clothing, move frequently and rapidly sometimes, gather in large and small numbers and interact in a world with varied lighting conditions and clutter. These factors make it difficult to generate general purpose robust algorithms to detect humans. The human detection algorithms are classified into two types: namely, the shape based methods and the motion based approaches. The shape based approaches are further classified into monolithic and part based methods. The monolithic approaches use either structured edge based models or classifiers in combination with shape encoding features. However, the detection rate drop significantly in the presence of occluded human beings. The part based methods take the help of parts of human body for detecting humans in a scene [2].

On the other hand, the motion based approaches detect the humans based on the change detection by using a statistical modeling for back ground. In the general frame work of human motion analysis, the captured video frames undergo motion segmentation to detect the regions that correspond to moving objects such as vehicles and people in a natural scene. This is followed by an object classification which aims to perform a classification between human and non-human objects. This process is followed by human tracking which estimates the trajectory of an object in the image plane as it moves around a scene. Then action recognition algorithms are used to understand different actions of humans in the scene from which a semantic description of the actions may be concluded [7]. Generally, in natural video surveillance scenario, the frame under surveillance may be heavily crowded, or may have a severely disordered background with shadows, large variations in stances and people or it may have large amount of data to be analyzed and so on. This makes the human motion detection a compellingly challenging task [3-5,8].

Therefore, all these problems have to be addressed jointly in a practical action detection approach in which action representation and search scheme shall be able to tolerate enormous variance and it should be computationally feasible as well. In this paper, a human motion detection system based on the 2-D wavelet features of history edge images of motion is proposed. This system can effectively encode shape and motion patterns of the image. Furthermore, a booster tree classifier is used for classifying different motions in a scene.

The rest of the paper is organized as follows: The methodology of the proposed motion detection system is explained in section 2. In section 3 the experiments conducted and results obtained are presented. Finally section 4 draws the conclusion of the paper.

II. PROPOSED METHODOLOGY

The proposed methodology can be understood easily from the block diagram shown in Fig.1.

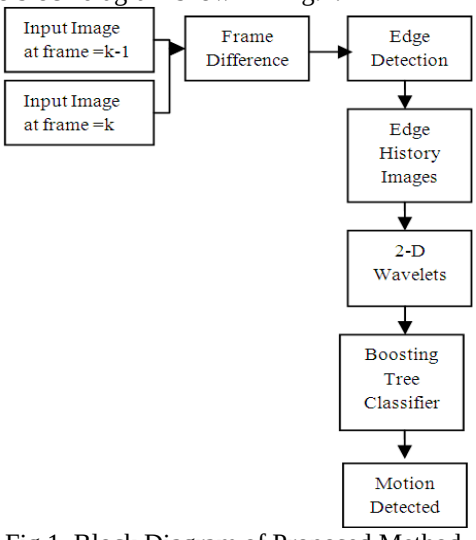

Fig.1. Block Diagram of Proposed Method

International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

The images from two successive frames are subtracted from each other in order to estimate the amount of motion between them. Then, an edge detection algorithm that can differentiate between all the straight and curved edges is applied on to the difference image. The edge maps thus obtained are stored for all the k frames in a video sequence of events. A 2D wavelet transform is then applied on each history image of edge to extract the motion features. The features thus extracted are given to boosting tree classifier for determining each action in human motion.

A. Edge Detection

The edge detection of the difference between two successive frames in the images is carried out in order to estimate the motion. The edge detection scheme must be a robust one because the information obtained at this stage is useful in classifying the motion at higher levels of processing the data. The procedure for edge detection employed in this paper is as follows:

Thresholding for image segmentation is generally made based on information contained in a gray level histogram of a given image. The aim of this approach is to identify the bottom of the histogram that positively separates the two groups or sub divisions. However, whether a pixel is an edge pixel or not depends not only on the gray values of the pixel, but also on gray values of the surrounding pixels. After approximating the gradient vector, one should not use the magnitudes of the derivatives alone for defining the appropriateness of a pixel to be an edge pixel, though it has been done in that way with many edge detectors. Threshold values may be selected based on the gray levels in the neighborhood of pixels which may vary from image to image. So, the changes in the neighborhood of a pixel also should be used in this analysis. In order to automatically vary the threshold normalization of gradient magnitudes is to be done with respect to the neighboring pixels gradient magnitude, and then it is to be confirmed whether the obtained value is large or not. A normal way of doing such normalization in any process is to use suitable statistical principles. This method of normalizing the gradient strength at each pixel locally before thresholding results in the elimination of the uncertainty, and thereby produces consistent, strong and smooth edges. The variance in the image is estimated by the equation 1.

$$V^2 = \frac{1}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(C_{i,j} - \nabla C(i,j) \right)^2 \quad (1)$$

Where $C_{i,j}$ is the magnitude of the pixel (i,j) of the contrast image obtained from GLCM and $\nabla C(i, j)$ is the gradient of the contrast image [8].

For every pixel location (x, y), the variance–Covariance matrix along horizontal direction $\nabla_x C$ (x, y) & vertical direction $\nabla_y C(x, y)$ is given by V(x ,y).
Where V (x,y) is given by the equation 2.

$$V(x,y) = \begin{pmatrix} \upsilon_{11}(x,y) & \upsilon_{12}(x,y) \\ \upsilon_{12}(x,y) & \upsilon_{22}(x,y) \end{pmatrix} \quad (2)$$

Where,

$$\upsilon_{11}(x,y) = \frac{1}{2\pi mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(\frac{x-i}{h} \right)^2 \nabla_x C(x,y) \quad (3)$$

$$\upsilon_{22}(x,y) = \frac{1}{2\pi mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(\frac{y-j}{h} \right)^2 \nabla_y C(x,y) \quad (4)$$

$$\upsilon_{12}(x,y) = \frac{1}{2\pi mn} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(\frac{x-i}{h} \right)^2 \left(\frac{y-j}{h} \right)^2 \nabla_x C(x,y) \nabla_y C(x,y) \quad (5)$$

Then the normalization parameter N(x, y) for each pixel is obtained by dividing the absolute magnitude of each pixel by its variance w.r.to its neighborhood pixels and is given in equation 6.

$$N(x,y) = \left(\nabla_x C \ \nabla_y C \right) V^{-1}(x,y) \left(\nabla_x C \ \nabla_y C \right)^T \quad (6)$$

The value of *N (x, y)* is determined for each pixel and if its value is found to be sufficiently large then that pixel *(x, y)* is considered to be an edge pixel.

Then a double thresholding technique which automatically chooses the upper threshold value to be equal to the mean of the normalized magnitude of all the pixels and the lower threshold be equal to ninety two percent of the upper threshold is applied to get the edge pixels. Then all the edge pixels are bridged to get a smooth edge map.

B. Motion History of Edge Images

A motion history image is used to represent the movement of 'motion. In a motion history image, the intensity of the pixel at a point is the function of the motion history at that point. The brighter intensity values of the pixels correspond to the most recent motion. In this paper, the motion history images of the edge maps obtained from previous section are used to represent the movement of motion as it is easy to compute the motion history images of edge maps. Furthermore, because of the pre-processing done before getting the edge maps, the motion history images will be free of noise. The motion history of the edge maps E^t_k (x, y) is given by equation 7.

$$M_k^t(x,y) = \begin{cases} 1 \ if \ E_k^t(x,y) = 1 \\ \beta(t) M_k^{t-1}(x,y) \ otherwise \end{cases} \quad (7)$$

Where, β is the updating function given by equation 8.

$$\beta(t) = \frac{1}{1 + exp^{-t}} \quad (8)$$

However, the data obtained till this step is huge and the complexity of computation would be very high during classification for this huge amount of data. So, the data is converted into feature space using 2D discrete wavelet transforms which is explained in the next sub section.

C. Feature Extraction using 2D Discrete Wavelet Transform

The dyadic wavelet transform decomposes a signal into a set of signals at different resolution levels. The information at the finer resolution levels is strongly affected by noise. In order to reduce this effect on the zero-crossing representation, only a few low- resolution levels, excluding the coarsest level, are used. This makes

International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

the representation robust in a noisy environment and reduces the number of computations required.

The wavelet transform allows for the decomposition of a signal using a series of elemental functions called wavelets and scaling which are created by scaling and translations of a base function known as the mother wavelet.

$s \in \Re^+$

$u \in \Re$

$$\psi_{s,u}(x) = \frac{1}{\sqrt{s}}\psi\left(\frac{x-u}{s}\right) \quad (9)$$

where's' governs the scaling and 'u' the translation. The wavelet decomposition of a function is obtained by applying each of the elemental functions or wavelets to the original function.

$$wf(s,u) = \int_{\Re} f(x)\frac{1}{\sqrt{s}}\psi^*\left(\frac{x-u}{s}\right)dx \quad (10)$$

Wavelets in practice are applied as high pass filters while scaling are equal to low pass filters. As a result of this, the wavelet transform decomposes the original image in to a series of different scales called trends and fluctuations. The trends are averages versions of the original image and the fluctuations contain the high frequencies at different scales or levels [10]. The decomposition of the image by the application of the wavelets is illustrated with the help of the block diagram shown in Fig.2.

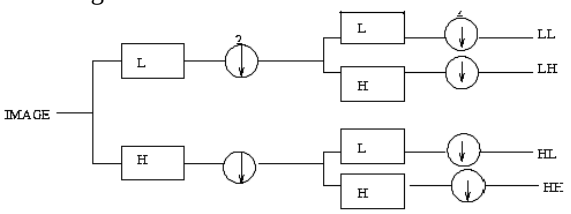

Fig.2: Block Diagram to Illustrate the Wavelet Decomposition

D. Boosting Tree Classifier

A decision tree is a sequential application of cuts splits the data into nodes, where the final nodes (leafs) classify an event as signal or background as shown in Fig.3.

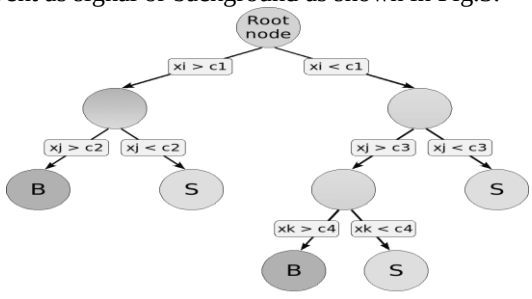

Fig.3. Decision Tree

Boosted decision trees combine a whole forest of decision trees, derived from the same sample, using different event weights. The process of boosting is depicted in Fig.4. The training sample is given to a number of week classifiers and the output of each of these classifiers is given to another strong classifier to get the final classification. This process of boosting the classification from a number of week classifiers by means of a strong classier is called "Adaptive Boosting Classifier" or "ADABOOST" classifier.

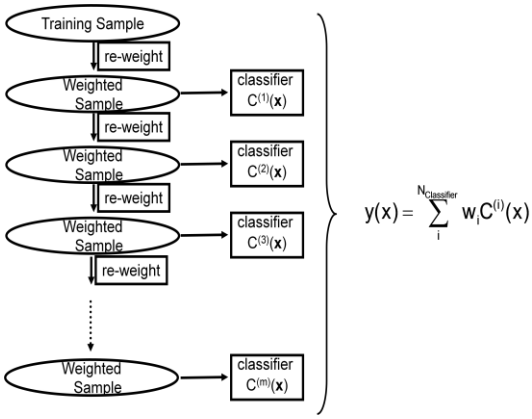

Fig.4. Boosting Illustration

AdaBoost re-weights events misclassified by previous classifier by using the equation 11.

$\frac{1-f_{err}}{f_{err}}$ with:

$$f_{err} = \frac{\text{misclassified events}}{\text{all events}} \quad (11)$$

AdaBoost weights the classifiers also using the error rate of the individual classifier according to equation 12

$$y(x) = \sum_{i}^{N_{Classifier}} log\left(\frac{1-f_{err}^{(i)}}{f_{err}^{(i)}}\right)C^{(i)}(x) \quad (12)$$

In this paper an AdaBoost classifier is used to classify different actions made by human beings.

Finally, the algorithm for Scale Invariant Motion Detection (SIMODE) is given below:

Algorithm_SIMODE

Input: A video sequence of k frames

Output: Motion Detection

Step1: Frame Differencing

For i=1;i<k i++

$D_t^i = f_t^i - f_t^{i-1}$ (Subtract two successive frames)

Step 2: Edge Detection using statistical thresholding

While (i!=K)

Do

1) Determine the gradient of D(i) along horizontal and vertical directions
2) Normalize the magnitude of the gradient using variance and co-variance matrix and obtain the normalized value as given in equation 6
3) Apply a double thresholding according to the chi-square distribution of the data
4) Obtain the edge map

Step 3: Motion History Images of Edge Maps

Obtain the motion history of the edge images using equation 7

Step 4: Feature extraction using Wavelets

The relevant features are extracted by using wavelets as defined in equation 11

Step 5: Classification of Human Action

International Journal of Electronics Communication and Computer Engineering
Volume 5, Issue 3, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Finally, classification of different human actions is done by make using of an AdaBoost classifier as described by equation 12.

The algorithm described above is applied on different bench mark videos taken from Weizmann data set. The experiments and results are furnished in the next section.

III. Experiments and Results

A. Results

All the experiments are performed on the Weizmann data set, which consists of nine different people performing ten different actions, namely: run, walk, skip, jump, jack, gallop sideways, wave two hands, wave one hand, and jumping jack. In this paper, the proposed method is used to recognize the human motion in different commonly made actions like walking, running and jumping. The simulations are performed using an intel i5 core processor with 3.16GHz clock for an image resolution of 180x144 (as in the case of Weizmann data set). Furthermore, the performance of the human motion detection system can be evaluated with respect to its accuracy in frames or in video. In this paper, the performance of the proposed algorithm is evaluated w.r.t the accuracy in frames.

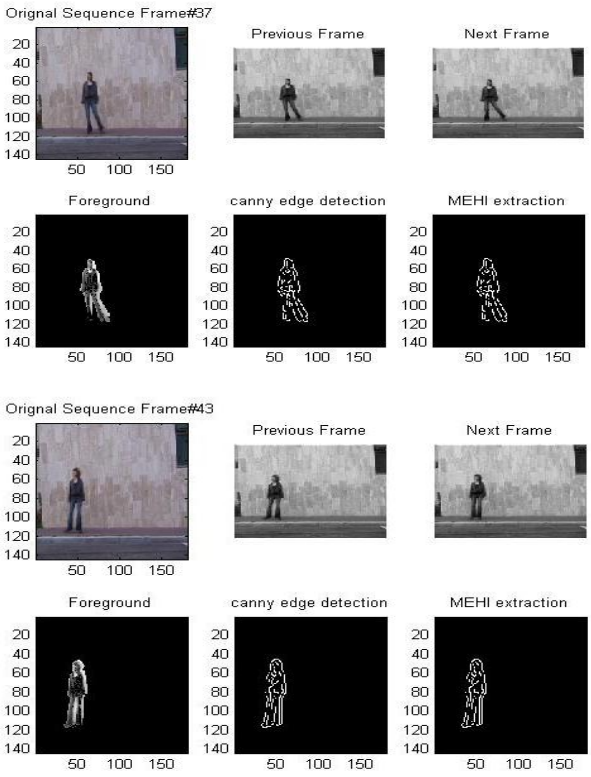

Fig.5: Edge Maps and Motion history Images of Edge Maps for Side Walk Video

The first video sequence taken into consideration for experimentation is the "Side walk" sequence. The edge maps obtained and the motion history of the edge maps for some frames is shown in Fig.5, and Fig.6 shows identification of Side walk motion.

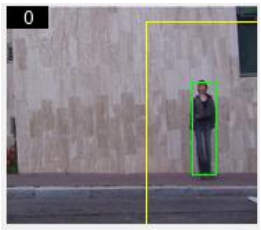

Fig.6. Identification of Side Walk

The next video sequence is the 'jog' video sequence. The corresponding edge maps and the motion history images of edge maps are shown in Fig.7. The identification of the "Jog" motion in the video is shown in Fig.8.

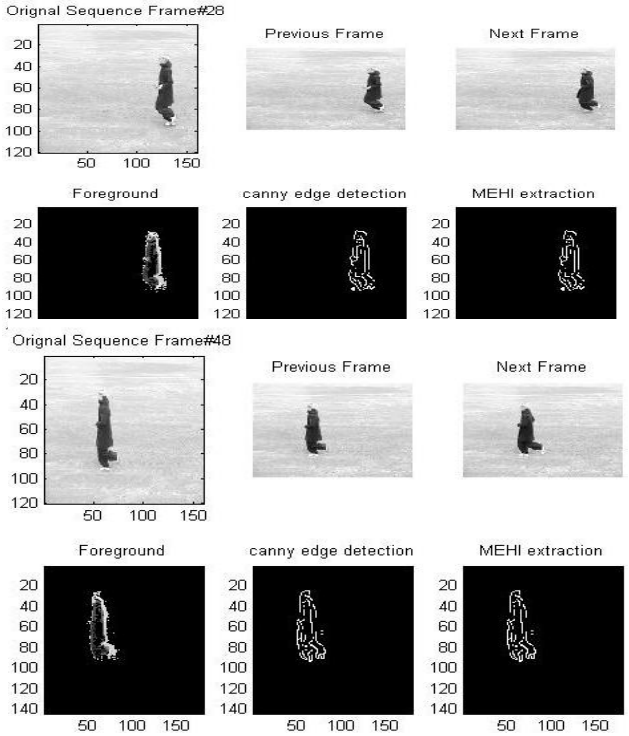

Fig.7. Edge Maps and Motion history Images of Edge Maps for Jog Video

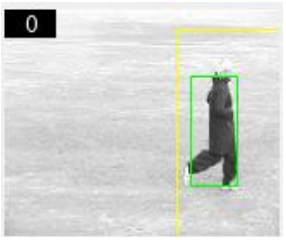

Fig.8. Identification of Jogging Motion

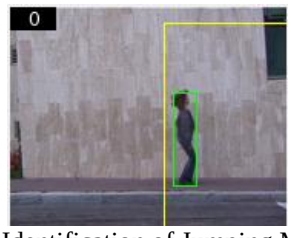

Fig.9. Identification of Jumping Motion

Similarly, the proposed algorithm is tested on other video sequences like the "jump, "run" and "walk" and the results obtained after identification of the specific motion in corresponding video is shown from Fig. 9 to Fig.11 respectively

Fig.10. Identification of Running Motion

Fig.11. Identification of Walking Motion

B. Performance Evaluation

The performance evaluation of any classification algorithm is done by making use of a confusion matrix.

A Confusion Matrix is a 2X2 matrix that, is used to tabulate the results of 2-class supervised learning problem and entry **(i ,j)** represents the number of elements with class label **i**, but predicted to have class label j as shown in Fig.12[6]**.**

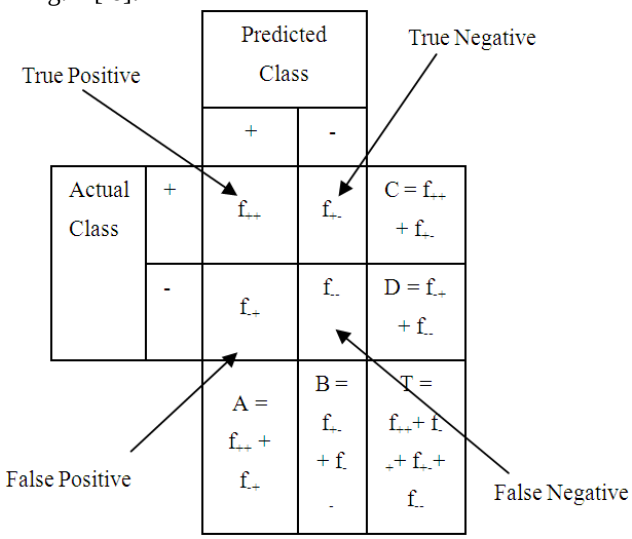

Fig.12: A 2x2 Confusion Matrix

A Confusion matrix is calculated to evaluate the performance of the proposed SIMODE algorithm. 23 video samples with three different actions like "walk" "run" and "jump" are taken into consideration. Class 1 is for the action "walk", class2 is for the action "run" and class 3 is for the action "jump". From the confusion matrix shown in Fig.13, it is clearly understood that an overall accuracy of 91.3% is obtained for correct recognition and 8.7% of the actions are not correctly identified. The best recognition is done for "Walk" video and the least recognition was done for "run" action.

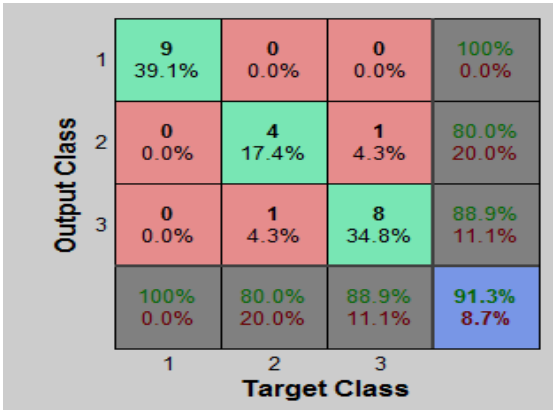

Fig.13. Confusion Matrix for Performance Evaluation of SIMODE Algorithm

IV. Conclusion

In this paper, we demonstrated the effectiveness of an efficient action representation based on a set of History Images of Edge Maps and presented complete human action detection system for recorded videos. We implemented the algorithm for feature extraction and classification of action using Weizmann Dataset and obtained the results which are independent of individual's appearance. The results obtained were satisfactory and performed the pattern recognition with 91.3% accuracy. In future it may be extended for other commonly made human actions like Cell to ear, putting down an object and pointing towards someone.

Acknowledgment

The author would like to express her sincere thanks to DST for sponsoring the project. The author wishes to express her profound sense of gratitude to Sri. K. V. Vishnu Raju garu Chairman, SVES, for his encouragement towards innovation and research. She also would like to express her sincere thanks to Sri Ravichandran Rajagopal, Vice-Chairman, SVES, for supporting at all times of need. She also would like to thank Dr. G. Srinivasa Rao garu, Principal, SVECW for facilitating with a good research environment.

References

[1] Ming Yang, Fengjun Lv, Wei Xu, Kai Yu and Yihong Gong, "Human Action Detection by Boosting Efficient Motion Features", in ICCV Workshops, IEEE 2009.

[2] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features", In *VS-PETS'05*, pages 65–72, Beijing, Oct. 15-16, 2005.

[3] National Institute of Standards and Technology (NIST): "TRECVID 2008 Evaluation for Surveillance Event Detection", http://www.nist.gov/speech/tests/trecvid/2008/ and http://www.nist.gov/speech/tests/trecvid/2008/doc/eventdet08-evalplan-v07.htm#tasks, 2008.

[4] M. D. Rodriguez, J. Ahmed, and M. Shah. "Action MACH a spatiotemporal maximum average correlation height filters for action recognition", In *CVPR'08*, Anchorage, AK, June 23-28, 2008.

[5] E. Shechtman and M. Irani. "Space-time behavior based correlation", In *CVPR'05*, volume 1, pages 405–412, San Diego, CA, June 20-25, 2005.
[6] M. Han, W. Xu, H. Tao, and Y. Gong, "An algorithm for multiple object trajectory tracking", In *CVPR'04*, volume 1, pages 864–871, Washington, DC, Jun.27-Jul.2 2004.
[7] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms", In *CVPR'98*, pages 232–237, Santa Barbara, CA, June 23-25, 1998.
[8] A. Fathi and G. Mori, "Action recognition by learning mid-level motion features", In *CVPR'08*, Anchorage, AK, June 23-28, 2008.
[9] Yun LI, C. Mohammed, C. Y. Suen, "A Threshold selection based on Multi Scale and Gray level Co-Occurrence Matrix Analysis" Proc of Eighth International Conference on Document Analysis and Recognition, vol 2, pp. 575-578,2005
[10] S. Mallat and W. L. Hwang, "Singularity Detection and Processing with Wavelets", IEEE Trans. Inf. Theory, vol38,no.2, pp 617-643,March,1992
[11] F.H. Y. Chan,F. K. Lam, Hui Zhu, " Adaptive Thresholding by Variational method", IEEE Trans. On Image Processing, vol 7 no3, pp 468-473,1998

AUTHOR'S PROFILE

K. Padma Vasavi
was born in India. She has done her under graduation in Electronics and Instrumentation Engineering from Kakatiya University, Warangal.
She did her Masters in Digital Systems and Computer Electronics in JNTU, Hyderabad. She has submitted her Ph.D. thesis to JNTUH, Hyderabad.
Mrs. K. Padma Vasavi is working at Shri Vishnu Engineering College for Women in the department of ECE. She has published around 10 papers in various international journals which were published reputed journals like Springer and Elsevier She has presented around 20 papers in various International conferences in India and abroad.
She has received many prizes for the projects mentored by her both at national and International level contests.
She has visited University of Massachussets, Lowell, USA, University of North Hampshire, USA, Georgia Tech University, Atlanta as a part of faculty exchange program in her University.
She has received grants from DST to pursue a research project in BCI.
She also received funding from DST to pursue her research on human motion detection and tracking. She is a Fellow of IETE and is a life member in ISTE, BMI.
Her research interests include Signal and Image Processing, Pattern Recognition, and Video Surveillance.