HMM Classifier for Human Activity Recognition

MS. Kanchan Gaikwad

Department of Computer, MGM College of Engineering, Kamothe, Panvel.
kanchan.gawande@gmail.com
Mr.Vaibhav Narawade (ME IT)
vnarawade@gmail.com
Department of Information Technology, PVPPSCOE, Sion, Mumbai

ABSTRACT:

The rapid improvement in technology causes more attention towards to Recognizing of human activities from video. These new technological growth has made vision-based research much more interesting and efficient than ever before. This paper present novel HMM (Hidden Markov Model) based approach for Human activity recognition from video. There are different approaches of HMM to recognize action of human from video. Like threshold and voting to automatically and effectively segment and recognize complex activities, segment and recognize complex activities and for simple activities we use Elman Network (EN) and two hybrids of Neural Network (NN) and HMM, i.e. HMM-NN and NN-HMM.

KEY WORDS:

Human Activity recognition, Hidden Markov Model, Hybrid model of HMM, Image capturing from Video, complex activity.

INTRODUCTION:

Automatically recognizing human activities from video is important for applications such as automated surveillance systems and smart home applications. Several human activity recognition methods [1][2][3][4][5][6] were proposed in the past few years to classify single human activities such as walking, skipping, sitting down, etc. Human activity recognition (HAR) research has been on the rise because of the rapid technological development of the image-capturing software and hardware, in addition to the omnipresence of reasonably low-cost high-performance personal computers. The main goal of this recognition is used to develop the different application which make human machine interaction is easy and interesting.

In the journey of developing algorithms for human activity recognition, some new developed algorithms adds some new features in previously developed algorithm. In this paper, we present a novel HMM-based approach that uses ***threshold*** and ***voting to*** automatically and effectively segment and recognize complex activities. And also survey on two hybrids of Neural Network (NN) and HMM, i.e. **HMM-NN** and **NN-HMM.** This paper also compares their performance with that of the traditional HMM.

DOI : 10.5121/cseij.2012.2403

In this paper section1 gives introduction about HAR and gives motto of paper. over view of traditional HMM classifier in section 2, Section 3 gives overview about HMM-based approach that uses ***threshold*** and ***voting*** and section 4 gives over view about HMM-NN and NN-HMM.

Section 5 contains the review of how object can be detected from other ways. The result of all methods as conclusion in section 6. Sections 7 contain references.

Section 2

TRADITIONAL HMM CLASSIFIER

A **hidden Markov model** (**HMM**) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (*hidden*) states. An HMM can be considered as the simplest dynamic Bayesian network. The logic behind the HMM was developed by L. E. Baum and coworkers. It is nearly dependent on an earlier work on optimal nonlinear filtering problem (stochastic processes) proposed by Ruslan L. Stratonovich, who was the first to describe the forward-backward procedure.[14]

In a regular Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a *hidden* Markov model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states. Note that the word 'hidden' is refers for the state sequence through which the model is passes, not for the parameters of the model.

Hidden Markov models are especially known for their application in temporal pattern recognition such as speech, handwriting, gesture recognition, part-of-speech tagging, musical score following partial discharges and bioinformatics.

A hidden Markov model can be considered a generalization of a mixture model where the hidden variables (or latent variables), which control the mixture component to be selected for each observation, are related through a Markov process rather than independent of each other.

Let us now formally define an HMM. We indicate the experiential symbol sequence as $\mathbf{x} = x_1 x_2 \ldots x_L$ and the underlying state sequence as $\mathbf{y} = y_1 y_2 \ldots y_L$, where y_n. is the essential state of the $\boldsymbol{n}$ the observation x_n. Each symbol x_n indicate a finite number of possible values from the set of observations $\mathbf{O} = \{O_1 O_2,\ldots,O_N\}$ and each state y_n takes one of the values from the set of states $\mathbf{S} = \{1,2,\ldots,M\}$, where N and M denote the number of different observations and the number of different states in the model, respectively.[14] We consider that the hidden state order is a time-homogeneous first-order Markov chain. These indicate that the probability of entering state j in the next time point depends only on the current state i, and that this probability does not change over time. Therefore, we have

For all states $i, j \in S$ and for all $n \geq 1$. The "Transition probability" is defined as transition from state i to state j , and we denote it by $t(i, j)$. For the starting state y_1, we define the *initial state probability* as $\pi(i) = \boldsymbol{P}\{y_1 = i\}$ for all $i \in S$. The probability that the n th observation will be $x_n = x$ depends only on the underlying state y_n, hence

$$P\left\{y_{n+1}=j \middle| y_n=i, y_{n-1}=i_{n-1}, \ldots, y_1=i_1\right\}=P\left\{y_{n+1}=j \middle| y_n=i\right\}=t(i,j) \quad (1)$$

$$P\left\{x_n=x \middle| y_n=i, y_{n-1}, x_{n-1}, \ldots\right\}=P\left\{x_n=x \middle| y_n=i\right\}=e\left(x \middle| i\right) \quad (2)$$

for all likely explanation $x \in \boldsymbol{O}$, all state $i \in S$, and all $n \geq 1$. This defines *emission probability* of x at state i, and we denote it by $e(x \mid i)$. The three probability measures $t(i, j)$, $\pi(i)$, and $e(x \mid i)$ completely specify an HMM. For our reference, we note the set of such parameters as $\boldsymbol{\Theta}$.

Based on such parameters, we can now calculate the probability for HMM will generate the observation sequence $\mathbf{x} = x_1\, x_2 \ldots x_L$ with the underlying state sequence $\mathbf{y} = y_1\, y_2 \ldots y_L$. This joint probability $P\{\mathbf{x},\mathbf{y} \mid \boldsymbol{\Theta}\}$ can be computed by

$$P\left\{x, y \middle| \Theta\right\}=P\left\{x \middle| y, \Theta\right\} P\left\{y \middle| \Theta\right\}, \quad (3)$$

Where

$$P\left\{x \middle| y, \Theta\right\}=e\left(x_1 \middle| y_1\right) e\left(x_2 \middle| y_2\right) e\left(x_3 \middle| y_3\right) \ldots e\left(x_L \middle| y_L\right) \quad (4)$$

$$P\left\{y \middle| \Theta\right\}=\pi(y_1)\, t(y_1, y_2)\, t(y_2, y_3) \ldots t(y_{L-1}, y_L). \quad (5)$$

As we can see, computing the observation probability is straightforward when we know the underlying state sequence.

Section 3

THRESHOLD-BASED HMM

To make sure that our single activity models do not mistakenly assign a single activity label to a clip with two activities, we determine a threshold Tj for the conditional probabilities of each activity j.[9] this algorithm uses these thresholds to reject assigning an activity label to a sequence U if all the conditional probabilities,

$P(U \mid A_j)$, for the different models fall below the corresponding thresholds T_j. $\overline{P}(U \mid A_j)$ is obtained from $P(U \mid A_j)$ by normalizing w.r.t. the number of frames. In order to determine the thresholds T_j, we use a set of single activity video clips and determine the conditional probabilities $P(X \mid A_j)$ for each clip X based on the correct model j. These probability values represent values that we need to accept. We also use a set of video clips such that each clip has two activities. These are examples

of cases that we do not want the system to classify. For each of these clips *Y*, we calculate two conditional probabilities $P(Y \mid A_i)$ and $P(Y \mid A_k)$ based on the models that correspond to the activities, *i* and *k*, in the clip. These probability values represent values that we need to reject. Then for each activity *j*, we select a threshold T_j that minimizes the number of misclassified cases. All the conditional probabilities used in this training are normalized w.r.t. the number of frames in the corresponding video clip as follows:

$$P(X \mid A_j) \quad P(X \mid A_j)/length(X) \tag{6}$$

In the above discussion we used both *X* and *Y* to denote video clips, and in the conditional probability expressions they represent the corresponding feature vector sequences. Based on this idea, the recognition result can be obtained as follows:

$$A_{Final} = \begin{cases} A & \text{if} \quad P(U \mid A_j) > Tj \\ Reject & \text{if} \quad P(\bar{U} \mid A_j) < Tj \end{cases} \tag{7}$$

where

$$A = \arg\max_{A_i \in \text{all activities}} P(U \mid A_j)$$

where $P(U \mid A_j)$ is the conditional probability for activity *j*, and is computed by:

$$P(U \mid A_j) \quad \max_i (P(U \mid A_{ji}), \quad i = 1,..., N)$$

where *U* is a sequence of feature vectors of an unknown activity, and *N* is the number of different viewing directions, in this work *N* is set to eight.

And $J = \arg\max_J P(U \mid A_j)$.

Applying Threshold based HMM to HAR

In this algorithm,[9] activity segmentation and recognition are combined in one process. During training, we train the *threshold-based* HMM*s* for each single activity separately. Then, during recognition we slide a window of length *N* over the sequence of frame features and classify the activity represented by the sequence in the window, see Figure 3. For a video clip with *M* frames we obtain a set of results r_i, $i = 1, 2, \ldots, M-N+1$, where result r_i is the activity assigned to window w_i. The result is used as a vote assigned to each frame in this window. We shift the window frame by frame and repeat the classification process. This will result in obtaining *N* results, r_j, for frame f_i, where $i-N+1 < j < i+1$. These classification results are considered as votes and we classify the activity of a frame by the activity that has maximum votes.

A low-pass filter was applied to smooth the voting curves as shown in Figure 6 in order to obtain the final segmentation and recognition results. Figure 6 shows two examples of voting results (after being filtered). Four curves (solid, dashed, point, star) represent votes for four activities (walking, standing up, sitting down, and writing on a white board) obtained separately for each frame. Sometimes the frames in a window contain frames from two different activities. The recognition results for these clips can be inaccurate and can induce errors in the final segmentation and recognition results. This is the reason for using *threshold-based* HMM*s* in our work.

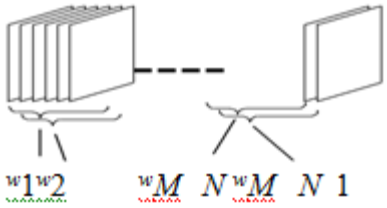

Figure 1. Sliding windows through the sequence of frames.

Section 4

HMM-NN HYBRID

In case with the traditional HMM classifier, it consider only trained to maximize the likelihood of producing its training examples it does not consider minimize the probability that produced by the model. This gives a negative force on the recognition capability. To improve the accuracy and result of HMM we integrate MLP at the output of HMM.

MLP is trained as a classifier with EBP can estimated the Bayes optimal distinguish function and on benefit of the discriminative training of the MLP, the weak point in the discrimination capability of maximum likelihood training of the HMM could be overcome. Hence as result of recognition of performance is increases.

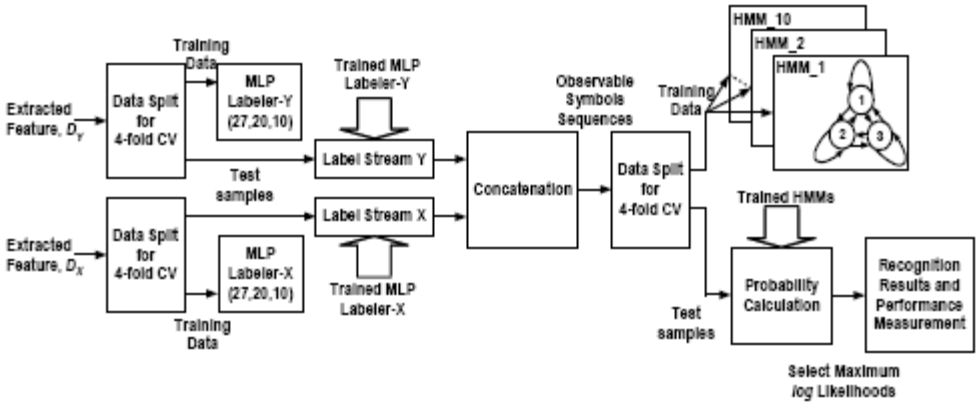

Figure 2: Block diagram of the NN-HMM hybrid

Applying HMM-NN hybrid to HAR

For the HMM stage of the hybrid, a three-state ergodic topology identical to traditional HMM classifier systems was used, for ease of comparison. The same training and recognition algorithms were also employed as described in the HMM HAR system. But, instead of assessing the system performance right at the end of HMM stage, its outputs obtained from the Forward algorithm were passed to the input layer of the MLP, and recognition performance was evaluated only at the end of the hybrid system.

For each test sample, the HMM stage output ten *log*-likelihood functions $P(O|\lambda_k)$, where k=1, 2, É, 10. They were then passed to the single-hidden-layered MLP, which has ten input neurons, 50 hidden neurons (this configuration was obtained heuristically, based on the bestperformance obtained) and ten output neurons, one to represent each class. The number of hidden units was actually varied from ten to 100, by steps of five, in the experiments to obtain the optimal network configuration.

NN-HMM Hybrid

In our final proposal, we incorporated two MLPs as labelers for the traditional HMM classifier, resulting in the NN-HMM hybrid.[10] The advantage of such a hybrid system over the traditional HMM classifier is that the MLP, being both trainable and discriminative, outperforms the ordinary vector quantized and improves the overall recognition capability. The benefit, looking from the MLP point of view, is that HMM will add some dynamic features to the MLP, giving it the capability of handling dynamic HAR problems with the same efficiency and finesse it normally handles static pattern recognition problem.

Applying NN-HMM hybrid to HAR

Two identical MLPs were implemented as labelers for the HMM stage, namely Labeler-Y and Labeler-X (Figure 6). The ten output indices of Labeler-Y were assigned labels ‘1’, ‘2’,’….’, ‘10’ to represent class ‘1’ to class ‘10’ of our human activity, respectively. Likewise, the output indices of Labeler-X were named ‘11’, ‘12’, …., ‘20’ representing class ‘1’ to class ‘10’, respectively. Each MLP labeler was trained with the modified EBP algorithm to classify vectors for one feature, i.e. either the differences in the x- or the y-coordinates between adjacent frames. The number of hidden units employed in each of the MLP labelers was varied from ten to 100, in steps of five, in the experiments.

To incorporate the MLP output information in the ensuing HMM stage, the straightforward yet effective winner-take-all labeling strategy was applied to the MLP labelers. It took into account the highest scoring output by passing only the label of the top scoring output to the HMM. The HMM then used the resulting label streams as the observation sequences, just as observation symbols from codebooks. Same as the traditional HMM system, the label streams from the MLP were concatenated to facilitate splitting of the data into four subsets for the training and evaluation of the three-state erotic HMM classifier. As before, one HMM, λ_k (where k=1, 2, É, 10), was trained specifically for each class of activity and training was via the Baum-Welch method of parameter re-estimation that maximized the likelihood function.

Section 5

Object Detection from video

The motion of object can be detected after the object is detected from video. Tracking the activity or object from sequence video frames this is the main goal of Video tracking. Blob tracking, kernel-based tracking, Contour tracking are some common target representation and localization algorithms. Ruolin Zhang [11] has proposed adaptive background subtraction about the video detecting and tracking moving object. He use median filter to achieve the background subtraction. This algorithm is used for both detecting and tracking moving objects in sequence of video. This algorithm never support for multi feature based object detection. Hong Lu and Hong Sheng Li [12] were introduced a new approach to detect and track the moving object. The define motion model and the non-parameter distribution model are utilized to represent the object and then the motion region of the object is detected by background difference while Kalman filter estimating its affine motion in next frame. The author shows Experimental results and proof the new method can successfully track the object under such case as merging, splitting, scale variation and scene noise. The author Bayan [13] talks about adaptive mean shift for automated multi tracking. The benefit of Gaussian mixture model is that it extracted Foreground image from video frame sequence it also eliminate the shadow and noise from video sequence. It is helpful in initializing the object trackers. As a result of this filter it reduces the search area and the number of iterations to meet for the new location of the object. The advantage of Gaussian mixture model as it reduces the background from video and hence we can track the object easily. The object can trap from video by changes in size and shape.

Section 6

RESULT OF ALL METHODS AND CONCLUSION

Recognition using the traditional HMM

Since there is no simple theoretically correct way of choosing the number of states, *S*; it was varied from three to ten in the experiment.[10] We fixed the number of symbols *M* at 111 based on the simplified 'quantization' process and used the Forward algorithms to compute the various likelihood functions. The best classification result of 87% is obtained when *S*=3, as shown in Figure 8, the plot of HMM recognition rate versus number of states, *S*. This reveals that in the HMM classifier, the higher number of states does not necessarily imply better performance. On the contrary, a mere three-state model is sufficient to classify our selected human activities, using two one-dimensional sequential features derived from tracking the estimated head centroid (x- and y-coordinates). Hence, the three-state ergodic topology is chosen for the conventional HMM, HMM-NN hybrid and the NN-HMM hybrid classifiers, for easy comparison.

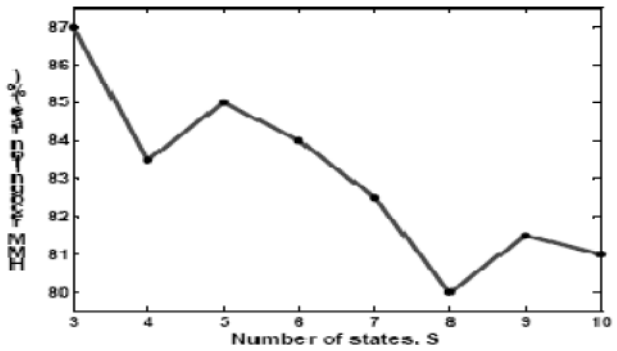

Figure 3: HMM recognition rate as a function of the number of states, *S.*

Recognition using the HMM-NN

In this hybrid, in order to obtain the optimal network configuration, the number of hidden units was varied from ten to 100, in steps of five. The recognition rate as a function of the number of MLP hidden units is plotted as shown in Figure 4. The highest performance is achieved when 50 hidden neurons are used in the MLP stage, yielding a recognition rate of 96.5%. The configurations with more than 50 hidden units had probably over fitted the problem and the MLP actually remembered the training examples, resulted in poorer recognition rate. As such, the 50 hidden-units architecture will be used in the HMM-NN classifier for comparison.

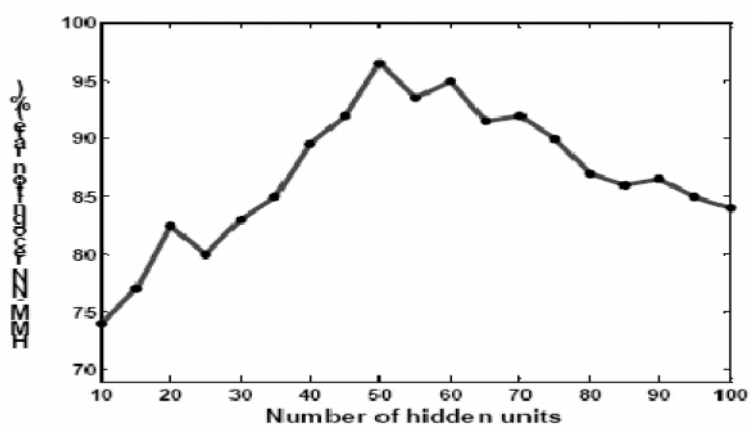

Figure 4: HMM-NN detection rate as a function of the number of MLP hidden units.

Recognition using the NN-HMM

Two identical MLPs were implemented as labelers for the HMM stage in this hybrid. Both used the modified EBP algorithm but each was trained to classify vectors for one feature, i.e. either the differences in the x- or the y-coordinates between adjacent frames. [10]The number of hidden units employed in each of the MLP labelers was varied simultaneously from ten to 100, in steps of five. The hybrid systems recognition rate and the labelers classification rate, both as functions of the number of the hidden units, are plotted in Figure 5. It was noticed that when each of the MLP labelers had 30 hidden units, the best classification results of 96% was obtained at the

output of the labelers. However, the best performance of the entire hybrid system does not peak there Ð it achieved the highest recognition rate of 95% when there are only 20 hidden units in each labeler. The 30-hidden-unit configuration had most likely memorized the training patterns and resulted in inferior overall performance.

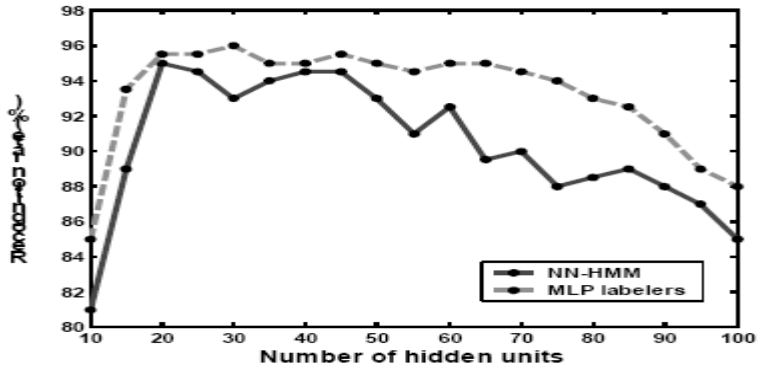

Figure 5 : NN-HMM recognition rate and labelers classification rate as functions of the number of MLP hidden units.

Recognition using Threshold-based hmm

In this paper, [9] we proposed an algorithm for activity segmentation and recognition from video clips containing complex activities. Both motion and shape features were used to represent human activities. We used threshold based HMMs to reject classifying the activity in a given sequence of frames if the evidence is not strong. We used a voting based algorithm for segmentation and recognition of activities. In our experiments, we experimented with videos that contain two or more activities. The activities included walking, sitting down, standing up, and wiring on a white board. The results showed that our algorithm is effective for segmenting and recognizing complex activities independent of the viewing direction.

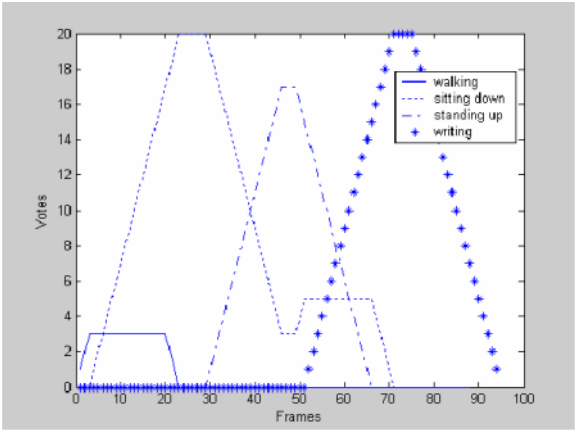

Figure 6. (a) Voting results for complex activity.

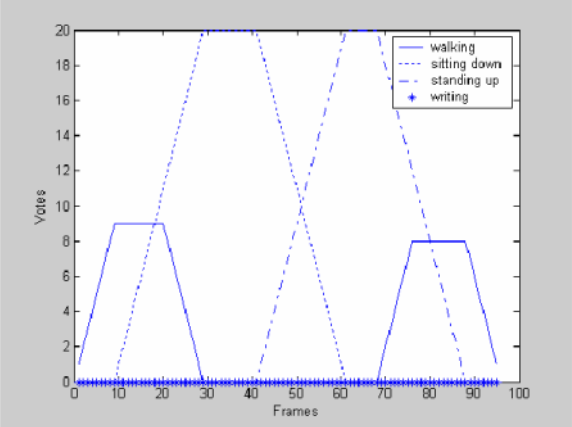

Figure 6. (b) Voting results for complex activity.

Section 7

REFERENCES:

[1] O. Masound, N. Papanikolopoulos, “Recognizing Human activities”, IEEE Conference on Advanced Video and Signal Based Surveillance, PP. 157-162, Miami, Florida, July 21-22,2003.

[2] F. Niu, M. Abdel-Mottaleb. “View-Invariant Human Activity Recognition based on Shape and Motion features”, IEEE Sixth International Symposium on Multimedia Software Engineering, pp. 546-556, Miami, FL, Dec.13-15, 2004.

[3] N. Oliver, E. Horvitz and A. Garg, “Layered Representation for Human Activity Recognition”, Proceedings Ninth IEEE ICCV, PP. 641-648, 2003.

[4] R. Hamid, Y. Huang, I. Essa, “ARGMode–Activity Recognition using Graphical Models”, Conference on Computer Vision and Pattern Recognition Workshop, Volume 4, PP. 38-45, Madison, Wisconsin, June 16-22, 2003.

[5] J. Ben-Arie, Z. Wang, P. Pandit, S. Rajaram, “Human Activity Recognition Using Multidimensional Indexing”, IEEE Trans. on PAMI, Volume 24 , Issue 8, PP. 1091-1104, August 2002. PP. 82--98, 1999.

[6] Y. Yacoob, M. J. Black, “Parameterized modeling and recognition of activities,” Journal of Computer Vision and Image Understanding, vol. 73, no. 2, PP. 232-247, 1999.

[7] D. M. Gavrila, “The visual analysis of human movement: a survey”, Computer Vision and Image Understanding, vol. 73(1), pp. 82-98 (1999).

[8] I. Essa, "Computers Seeing People", AI magazine, vol. 20(1), pp. 69-82 (1999).

[9] Feng Niu and Mohamed Abdel-Mottaleb,” Hmm-based segmentation and recognition of human activities from Video sequences” 0-7803-9332-q23-5/05/$20.00 ©2005 IEEE.

[10] Henry C. C. Tan and Liyanage C. De Silva “Human Activity Recognition by Head Movement using Elman Network and Neuro-Markovian Hybrids”, Palmerston North, November 2003

[11] Ruolin Zhang , Jian Ding,(2012) “Object Tracking and Detecting Based on Adaptive Background Subtraction”, Stevens Institute of Technology, Hoboken.

[12] Hong Lu, Hong Sheng Li, Lin Chai, Shu Min Fei, Guang Yun Liu, (2011) “Multi-Feature Fusion Based Object Detecting and Tracking”, journal on Applied Mechanics and Materials.

[13] C. Beyan, A Temizel, (2012), “Adaptive mean-shift for automated multi object tracking” IET on Computer Vision.

[14] https://en.wikipedia.org/wiki/Hidden_Markov_model