Detection of Moving Images Using Neural Network

P. Latha, L. Ganesan, N. Ramaraj, and P. V. Hari Venkatesh

***Abstract*—**Motion detection is a basic operation in the selection of significant segments of the video signals. For an effective Human Computer Intelligent Interaction, the computer needs to recognize the motion and track the moving object. Here an efficient neural network system is proposed for motion detection from the static background. This method mainly consists of four parts like Frame Separation, Rough Motion Detection, Network Formation and Training, Object Tracking.

This paper can be used to verify real time detections in such a way that it can be used in defense applications, bio-medical applications and robotics. This can also be used for obtaining detection information related to the size, location and direction of motion of moving objects for assessment purposes. The time taken for video tracking by this Neural Network is only few seconds.

***Keywords*—**Frame separation, Correlation Network, Neural network training, Radial Basis Function, object tracking, Motion Detection.

I. Introduction

In this paper a motion detection algorithm is presented in the compressed domain with a low computational cost. Here it is assumed that video is compressed by using motion JPEG (MJPEG). First the movie is converted into a sequence of frames. As per basic block diagram shown in Fig. 1, the sequence of compressed image frames [3] from the given input video is given to cross correlation network and neural network to detect rough motion.

Then for the purpose of minimizing error and getting accurate output a Feed-Forward Neural Network [1], [5] is formed. As nature of output is known, supervised learning is done and formed network is trained using Radial Basis Function, to detect the level of the motion.

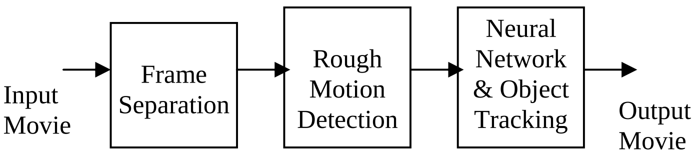

Fig. 1 Basic Block Diagram

P. Latha is with Department of EEE, Govt. College of Engineering, Tirunelveli – 7.

L. Ganesan is with Department of CSE, ACCET, Karaikudi, India.

N. Ramaraj is with G. K. M. Engineering College, Chennai, India.

P. V. Hari Venkatesh is with Department of EEE, Govt. College of Engineering, Tirunelveli-7 (e-mail: plathamuthuarj@gmail.com).

The objective of this paper is achieved in different chapters as outlined here. Section II gives an introduction to frame separation; Section III analyzes the Cross Correlation Network. Level of motion detection and the basics of neural networks are analyzed in Section IV. The result analysis is given in Section V and further details of this paper are given in the succeeding chapters.

II. Frames Separation

Frame separation is the foremost step in motion detection [3]. Actually the input is given to the entire network in the form of a movie. Next the frames are extracted from the input movie and converted to images.

This process includes store the input movie (.avi file), fix the information about the no. of frames, width and height of the frame, image quality etc. In MATLAB a GUI which is an interaction between the user and computer, is designed for frame Separation. Given input movie is as shown in Fig. 2 and some of the frames which are stored in the current directory are shown in the Fig. 3.

Fig. 2 Input Movie

Fig. 3 Input Movie with frames

III. Correlation Network

In this correlation network, first the two frame images are sub divided into four equal parts each. A two dimensional cross correlations is calculated using the formula given below between each sub image with its corresponding part in the other image.

$$r = \frac{\sum_m \sum_n \left(A_{mn} - \overline{A}\right)\left(B_{mn} - \overline{B}\right)}{\left(\left(\sum_m \sum_n \left(A_{mn} - \overline{A}\right)^2\right)\left(\sum_m \sum_n \left(B_{mn} - \overline{B}\right)^2\right)\right)^{1/2}}$$

Where
r = Correlation value
A = matrix representing pixel values of first frame
B = matrix representing pixel values of second frame
Such that
Size (A) = Size (B)

$\overline{A}$ = Average of (A)
$\overline{B}$ = Average of (B)

This process produces four values ranging from -1 to 1 depending on the difference of the two correlated images. The value is 1 if there is no motion and this value keeps on decreasing as the level of motion increases. The goal of this division is to achieve more sensitivity so the minimum value of correlation can be used as variance value which shows maximum motion. If the value lies between the ranges 0.9 to 0.8 then it is low level motion ,if it is nearer to 0.5 then it is medium level motion else it is high level motion.

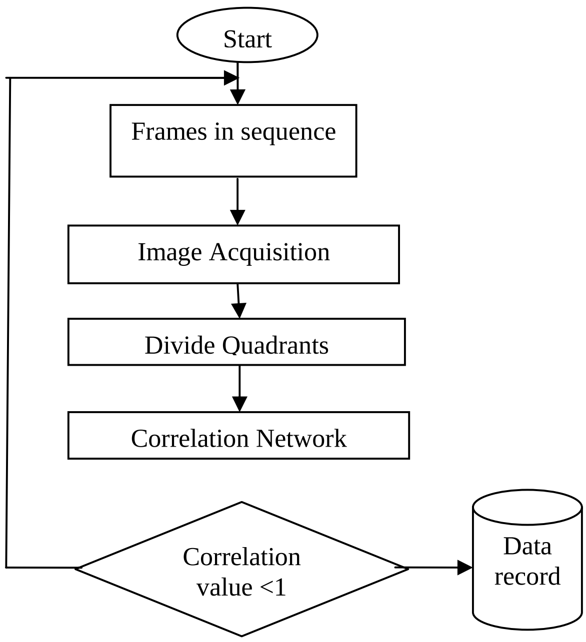

Fig. 4 Basic steps of Correlation Network

Fig. 4 explains all the procedures followed in the correlation Network. For the purpose of correlating, the first two images are considered. These images are resized to get equal ordered matrices for the purpose of ease in quadrant division. Comparing the pixel values when they are in RGB form, it is quite difficult and complex as each pixel consists of three values(Red intensity, Green intensity and Blue intensity at that point). So they are converted to gray images where the pixel comparison is flexible. Then these two images will be divided into 4 equal quadrants.

First quadrant of the first image with the corresponding quadrant of the second image is compared. The pixel values of these two quadrants are used to obtain the correlation value. Similarly the other three quadrants are compared and the formula is applied to get the remaining three values. These four values are checked with the correlation value (r =1).If it is less than 1, depending upon the range, the type of the motion is detected. Among the four values obtained, the minimum value represents maximum motion. This entire process is repeated for all remaining consecutive images and the value of "r" is calculated for all cases.

IV. Detection of Level of Motion Using Neural Networks

Level of motion refers to the degree of movement there is between two image frames. There are three levels as Low level, Medium Level, High level. If there is very little change in pixel values (i.e. Movement is very less) between two frames, then the motion is said to be low level motion.

This chapter explains about the design and training of neural network which takes the variance value as input and gives output as either 1 or 0. ie. whether the motion is really present or not.

Basic steps followed for the actual network formation [2], [4], [8] and training of the neural network are:

- Basic network assumption
- Calculating bias and weights
- Error calculation and determination of actual error &
- New network formation and training

Weight of an input is a number which when multiplied with an input gives the weighted input. This weight determines the effect that each input has on decision-making. Also bias value is a number, which is calculated from the difference between observed output and actual output and is given to the network in order to reduce the error. By using the correlation values, weights are applied to an input to get weighted inputs. The error goal is assumed here to be 0. Net input function is then calculated for a layer's net input by combining its weighted input and bias values.

Here the net input to the `radbas` transfer function is the vector distance between its weight vector **w** and the input vector **p**, multiplied by the bias *b*. RADBAS is a transfer function which is used to calculate a layer's output from its net input. The command RADBAS (n) calculates the transfer function using the formula:

$$\textbf{Radbas (n)} = e^{-n^2}$$

Where n is the product of the correlation value and the bias value. Using this, the layer's output is calculated which gives nothing but the observed output. The aim of error calculation and determination of actual error is to decide whether the observed output coincides with the target vector .The target vector is nothing but the expected result which is to determine whether motion is really present or not. So actual error is calculated by the difference between the observed output and target vector.

Initially the error is calculated between the input vector and the target vector by using the formula

$$e = \frac{(P' * d)^2}{(dd - PP')}$$

Where p = input vector, d = target vector, dd = square of target vector, pp = square of input matrix .The error, e thus obtained will be in matrix form.

After calculating this error the actual error has to be calculated. First the column having the maximum error in the error matrix is taken. This error is used to calculate the weights and bias to be given between the layers. The radial basis transfer function is calculated and the output was determined. Finally the error, e is minimized to get SSE, which is the difference between Target Vector and observed output Vector.

The actual error calculated is compared with the error goal if it is very high then a neural network has to be formed. Since the error is found to be high in the basic network and the nature of the output for a particular input is already known, training is carried out using Supervised Learning. It presents the network with a number of inputs and their corresponding outputs and checks how closely the actual outputs match the desired ones. Also it modifies the parameters to better approximate to the desired outputs.

Thus the input, which is the variance value between two frames, is given to the final network and it is determined whether motion is really present or not. The entire process starting from the basic network assumption till the new network formation is repeated for the remaining frames. Finally when motion is detected the next stage is the Object Tracking.

V. Object Tracking

In this paper, the output of neural network [7] is applied to object tracking. The object tracking is performed in two stages. The first stage is the detection of the regions where the motion had been detected with in each frame and the next stage is the conversion of the frames back to a movie. If the above said steps are followed, then on playing the video file in the current axis, the object can be tracked. In object tracking, once motion has been detected, the moving object is now to be extracted from its still background.

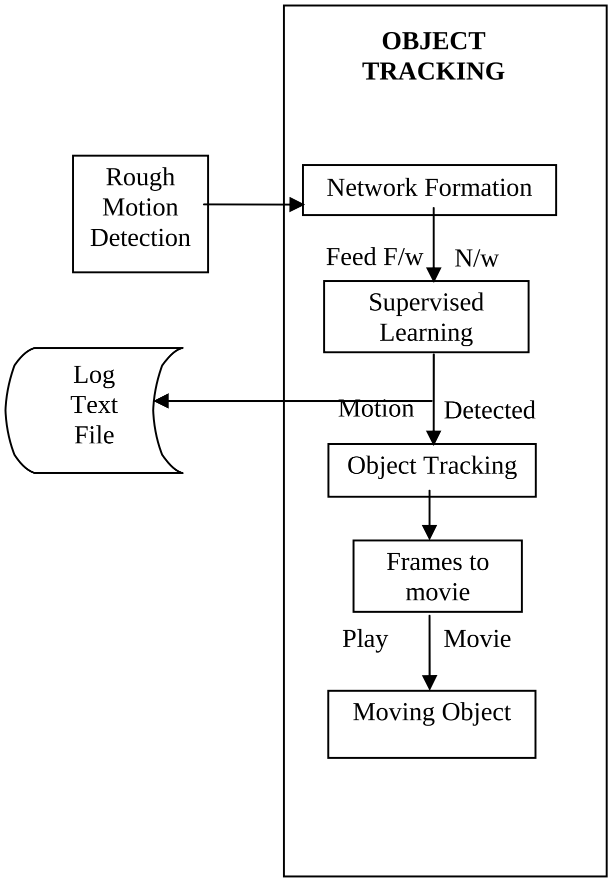

Fig. 6 Object Tracking

Here the intensity of the still background is increased so that it becomes bright and the edges of the moving object will be clearly visible. After object tracking is completed, the details such as nature of the moving object, size and direction of motion can be obtained.

VI. Observations and Results

To illustrate the efficiency and gain more insight to the achieved performance of this proposed method, the tracked MOVIE of this paper is presented in the following sequence.

```
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:32
In Frame Number 00002
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:33
In Frame Number 00003
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:35
In Frame Number 00004
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:37
In Frame Number 00005
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:38
In Frame Number 00006
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:38
In Frame Number 00007
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:40
In Frame Number 00008
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:41
In Frame Number 00009
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:42
In Frame Number 00010
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:44
In Frame Number 00011
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:45
In Frame Number 00012
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:47
In Frame Number 00013
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:48
In Frame Number 00014
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:49
In Frame Number 00015
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:50
In Frame Number 00016
Low level MOTION WAS DETECTED AT 09-Apr-2008 01:44:51
In Frame Number 00017
```

Fig. 7 Log text file for given input movie

The input video is taken with moving camera. It can be seen that the output, as shown in the Fig. 8, the tracked object is clearly visible (distinct). Thus the movement of the camera and the changes in the illumination has no effect on the efficiency of this algorithm and tracked output image is as shown in Fig. 8. Fig. 7 shows the LOG TEXT file for the given input image.

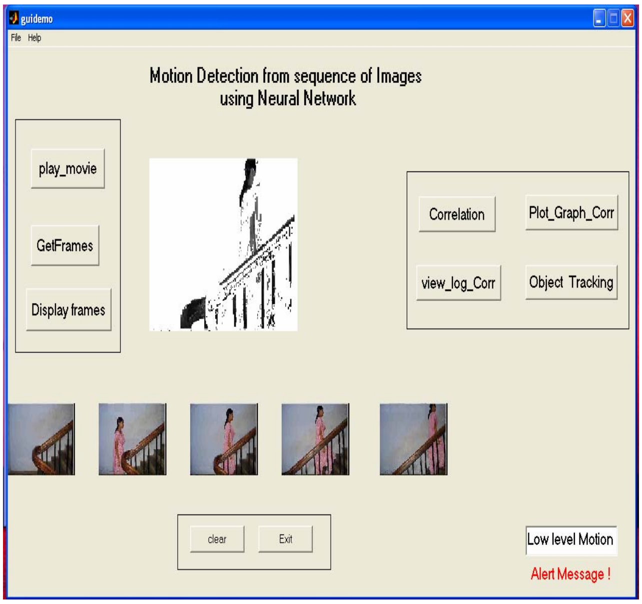

Fig. 8 Tracked output Image

VII. Conclusion

In this paper an effective algorithm to detect motion in video sequences has been presented. It has a very low computational cost & high efficiency in detecting sudden motion. The algorithm is able to detect moving blocks in every frame by detecting changes in pixel values in the quadrants.

The processing time of system depends on number of moving objects and is very less. This concludes that real time motion detection can be best implemented by the proposed approach based on correlation network and RBF. It can be mainly used in security systems and defense systems for detecting unauthorized motion and unwanted moving objects. This also has bio-medical applications where it can be used to detect the movement of cells and growth of tissues.

References

[1] 'Motion video sensor in the compressed domain'. SCS Euromedia Conf., Valencia, Spain, 2001.
[2] GieseandT.Poggio(2003) Neural mechanisms for the recognition of Biological movement sand action. Nature Reviews Neuro science 4:179–192.
[3] "Neural Networks for 3D Motion Detection From a Sequence of Image Frames," Chan Lai Wan and Yip Pak Ching, IEEE, computer Science Dept., The Chinese University of Hong Kong, 1991.
[4] Ferster and K.D. Miller. (2000). Neural mechanisms of orientation Selectivity in the visual cortex. Annual Review of Neuroscience, 23:441–471.
[5] Lippmann, R.P.,(1987), "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine.
[6] R. Brunelli and T. Poggio. Face recognition: Features versus templates. IEEE Transactions on Pattern Analysis and Machine Intelligence, 5(10):1042–1052, October 1993.
[7] D. Gavrila, "The visual analysis of human movement: A survey," Computer Vision and Image Understanding, vol. 73, pp. 82–98, 1999.
[8] Robert J .Schalkoff, (1997),"Artificial Neural Networks", McGraw-Hill, International Editions.
[9] Y. Song, A perceptual approach to human motion detection and labeling. PhD thesis,California Institute of Technology, 2003.