IJCSN International Journal of Computer Science and Network, Volume 6, Issue 2, April 2017
ISSN (Online) : 2277-5420
www.IJCSN.org
Impact Factor: 1.5

Motion Clustering Estimation on Video Sequences Using Kohonen's Self Organizing Map (SOM) Neural Networks

[1] Krishna Kumar; [2] Dr. Krishan Kumar; [3] Dr. S.S. Bedi; [4] Dr. Rahul Mishra

[1] Department of Computer Application, IFTM University, Moradabad,
Uttar Pradesh, India

[2] Department of Computer Science, Faculty of Technology, Gurukul Kangri University, Haridwar,
Uttarakhand, India

[3] Department of CSIT, IET, MJP Rohilkhand University, Bareilly,
Uttar Pradesh, India

[4] Department of Computer Application,IFTM University, Moradabad,
Uttar Pradesh, India

Abstract – Motion estimation is a very important and interesting area of research. It has become the necessity of many fields such as agriculture, security, medicine, traffic, and sports, the growth of a plant, tracking the movement of a vehicle within traffic, or observing the movements of a runner's hands or legs. Traditional methods for motion estimation estimate the motion field between a pair of images as the one that minimizes a predesigned cost function. An unsupervised learning method from the family of artificial neural networks i.e. Kohonen;s Self-organizing Map or SOM, a popular clustering way, based on Euclidian distance, when at test time, is given a pair of images as input it produces a dense motion field as its output layer. In the absence of large datasets with ground truth motion that would allow classical supervised training methods, the network in an unsupervised manner using the Self-organizing map is used for the training and hence for clustering to find the dense active regions in multiple frames of a video sequence. SOM used in this paper is also compared with other methods of clustering like k-Means algorithm, Nearest neighbour algorithm, Image subdivision algorithm, and Competitive learning network. Consequently It is observed that the Self-organizing map provides more accurate results and less error.

Keywords - *Motion Estimation, Video sequence, clustering, Self-organizing map neural Network*

1. Introduction

Basically, motion analysis is the technique for the evaluation of movement occurring in an environment. This evaluation process includes the determination of the direction, speed, size, and density of movement. Moving objects on video sequences obtained by a video camera can either be a human being or other living thing, or could be a moving vehicle. Motion analysis is used in various fields, such as agriculture, security, medicine, traffic, and sports. Observing the growth of a plant, tracking the movement of a vehicle within traffic, or observing the movements of a runner's hands or legs depend on the processes of movement analysis. It is the "royal league" task when it comes to reconstruction and motion estimation and provides an important basis for numerous higher-level challenges such as advanced driver assistance and autonomous systems. Motion analysis can also be used in observing the movements of crowds [1]. Various techniques have been so far used for the implementation of motion analysis. Artificial neural network (ANN), a popular and classical technique used for most of area of research like pattern recognition and classification can also be used for promising results in motion analysis of a video sequence. Perhaps, supervised, unsupervised, and reinforcement learning might be used for the motion estimation. However some methods which are based on unsupervised learning are more popular.

2. Literature Survey

A lot of research work has already been done in the area of motion estimation so far. Soft computing techniques like neural networks have given wonderful results. Many researchers used many models to handle the issues regarding the motion estimation. Moreover different-2 modelling and simulations have been done. Some researchers examined the simulation for one person and some examined for more than two persons. This situation of more than two persons in same directions is shown in figure 1 and in opposite direction is shown in figure 2.

Fig.1. (A) Screenshots displaying the motion of more than two persons in same direction at some position.

Fig.1. (B) Screenshots displaying the motion of more than two persons in same direction at other some position.

Fig. 2: Screenshots displaying the motion of more than two persons in opposite direction. [29]

Many academic studies have been implemented on motion analysis. Some examined only the motion of single person and other has been implemented on more than two persons. A single person was analysed using indoor environment and the image segmentation for the different human actions by two different techniques based on colour intensity and motion [2]. Some studies searched groups of people and evaluated the motion of people in groups. The authors tracked groups of people and the tracking process was realized in both indoor and outdoor environments using different-2 approaches of motion estimation [3].

In addition, there are findings that deal with the estimation of traffic density by watching the motion of vehicles in traffic. They find regions where the traffic density is higher than other regions. In a study using adjusted images, the vehicle's speed and traffic intensity were estimated [4]. Moreover, in another study the vehicles in real-time traffic flux were considered to determine the traffic intensity. Traffic intensity is estimated due to the presence of a vehicle in the sequence [5],[6].

Various research studies are carried out using artificial neural networks (ANN). ANN itself proved as a good tool to analyse the motion estimation. In [7], a special node-splitting criterion was presented based on the competitive learning network (CNN), which is a self-creating model, and this model gives better clustering or quantization results. Moreover, the competitive learning network was used in various studies. These studies deals with network intrusion detection [8], adaptive feedback controller [9], and feature maps integrations [10].

Furthermore, basically neural networks are divided in two important categories i.e. single layer perceptron (SLP) and multi-layer perceptron (MLP.) Multilayer perceptron was used to determine different team tactics in volleyball. The team tactic patterns were used to train multilayer perceptron neural network [11].

Another study dealt with a neural network of motion perception and speed discrimination [12].There were some other studies also that used different types of cameras, as catadioptric [13] and multicamera systems [14,15]. A technique based on m-mediods was used in [16] to classify motion and anomaly detection. In this study, object motion trajectories were modeled as a time series and used in the algorithm.

Another study was about the estimation of dominant motion direction on video sequences, and the k-means algorithm was used to classify motion vectors [17].

Dosovitskiy et al. showed that optical flow estimation can be posed as a supervised learning problem and can be solved with a large network. For training their network, they created a simple synthetic 2D dataset of flying chairs, which proved to be sufficient to predict accurate optical flow in general videos. These results suggest that also disparities and scene flow can be estimated via a convolutional network, ideally jointly, efficiently, and in real-time [18].

3. Motion analysis

The motion analysis process mainly consists of three basic stages: pre-processes, the determination of motion vectors using the full search algorithm, and clustering utilizing the competitive learning network. At first, video images are subjected to specific pre-processing operations in the motion analysis process. Motion vectors occurring in the moving regions in video images are determined after these processes. Finally, the motion vectors are grouped and the motion analysis process is implemented, which is shown in Figure 1.

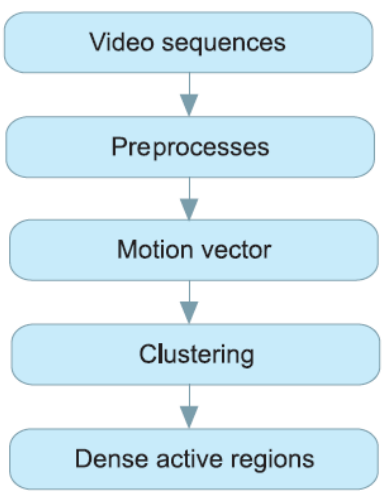

Fig. 3 Process of motion analysis

3.1. Pre-processing operations

Basically video sequences consist of frames, and consecutive frames are compared for motion estimation. Two frames are selected for comparison by leaving 5-6 frames apart from each other, since there must be a definite interval between the frames for formation of the movement. Various preprocessing operations are applied to the frames that usually will be compared after determination. The first preprocessing operation is finding the RGB values of the frames' pixels. R denotes the level of red brightness, G denotes the level of green brightness, and B denotes the level of blue brightness. The frames are converted into gray scale using the RGB values. The process of converting to gray scale is done by summation after multiplication of RGB values singly with certain coefficients. The process of converting gray scale is done given according to Eq. (1), where $g(x; y)$ represents the pixel value.

$$g(x; y) = 0{:}299R + 0{:}587G + 0{:}114B \qquad (1)$$

Moreover, the frames are separated into 4 regions within this process and the average values of the pixels converted into gray scale in each region are calculated as given in Eq. (2), where M and N represent the sizes of each region.

$$\text{Average value} = \frac{1}{MN}\sum_{u=0}^{N}\sum_{u=0}^{M} g(x, y) \qquad (2)$$

Later, the edges of the objects are found using the Sobel fillter. The neighbourhoods of the pixels and operators of the Sobel filter are given in Figure 4. Sobel operators are also used to determine the direction of a pixel or the local direction of a block in an image [19]. The pixel's gradient values of the x and y directions are calculated as given in Eqs. (3) and (4). The pixel value to be assigned is calculated as given in Eq. (5).

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

(a)

1	0	−1
2	0	−2
1	0	−1

(b)

1	2	1
0	0	0
−1	−2	−1

(c)

Fig. 4. a) Neighbourhoods of the pixels, b) horizontal operator, c) vertical operator of the Sobel filter.

$$G_x = Z_1 + 2Z_4 + Z_7 - Z_3 - 2Z_6 - Z_9 \qquad (3)$$

$$G_y = Z_1 + 2Z_2 + Z_3 - Z_7 - 2Z_8 - Z_9 \qquad (4)$$

$$G = \sqrt{G_x^2 + G_y^2} \qquad (5)$$

Frames are converted into black-and-white form by utilizing the average value obtained from 4 separated regions. The thresholding process is performed according to Eq. (6).

$$g(x; y) = \begin{cases} 255, if\ (g(x,y) >= averagevalue \\ 0, otherwise \end{cases} \qquad (6)$$

After these processes, 2 frames are compared with each other in order to find the active region. A new image is formed; in this form, the regions are white with a white-coloured first frame and a black-coloured second frame, and the rest of them are black. Regions of the white pixel in the new image show the active region [20]. The process for finding the active region is carried out according to Eq. (7). Finally, noises are cleaned on the image by a filter.

$$g(x; y) = \begin{cases} 255, if\ (g1(x,y) >= 255\ and\ g1(x,y) >= 0 \\ 0, otherwise \end{cases} \qquad (7)$$

3.2 Full search algorithm

The full search algorithm is one of the block-matching algorithms. Block-matching algorithms are generally used in motion estimation and motion analysis. Frames are divided into macroblocks in block-matching algorithms. Motion vectors are formed via comparing macroblocks situated in the first frame with macroblocks in the search region located in the second frame. Macroblocks are compared for each probability in the search region in the full search algorithm, which is the most accurate one among the block-matching algorithms [21],[22]. Motion vectors are found using the full search algorithm after the end of the preprocessing operations. Initially, the frames are separated into W x W pixel macroblocks for the full search algorithm. The macroblocks of the first frame's active region are matched on a 3 x 3 macroblock region of the second frame by scanning.

The sum of absolute differences (SAD) is computed at all points in a 3 x 3 macroblock region [21]. The W x W pixel regions that have the lowest SAD are matched. The SAD is computed as follows:

$$SAD = \sum_{u=0}^{w} \sum_{v=0}^{w} |MB_1(x+u, y+v) - MB_2(x'+u, y'+v)| . \qquad (8)$$

Here, *MB*1 denotes the macroblock that will be compared in the first frame and *MB*2 denotes the macroblock in the search region in the second frame of the equation. (*x; y*) represent the coordinates of the macroblock pixels. In many academic studies, the solutions of the problems or the algorithms are given as pseudocodes [22,23,24]. The pseudocode of the full search algorithm that we follow in this study is shown in Figures 5 and 6.

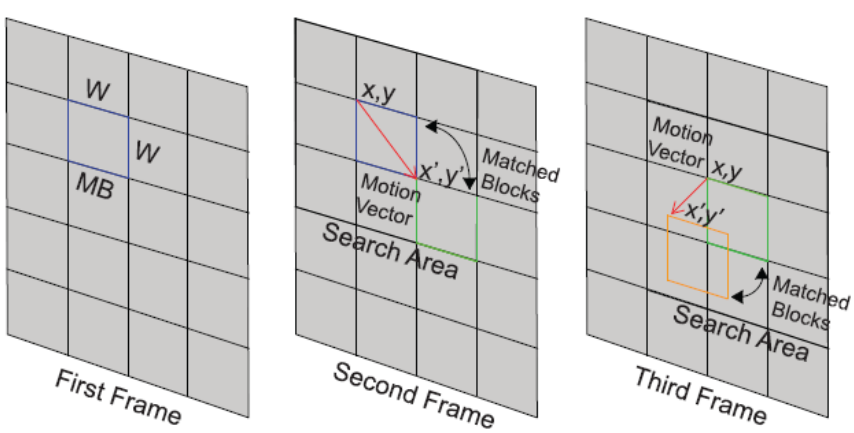

Fig. 5 Steps of Full search Algorithm

```
Step 1 : Start
Step 2 : Divide frames that will be compared into macroblocks having W×W dimension
Step 3 : Define search region as 3×3 macroblock
Step 4 : For each MB1 (x,y) in the first frame
  {
      Step 4.1 : For each i ∈{ x - W, x - ( W − 1 ),..., x + W }
      {
      For each j ∈{ y - W, y - ( W − 1 ),..., y + W}
        {
         Calculate SAD value between MB2(i,j) in the second frame and MB1(x,y)
         If i = x - W and j = y - W The lowest value = SAD
         If SAD < The lowest value { The lowest value = SAD, x' = i, y' = j }
        }
      }
      Step 4.2 : Form motion vector between (x,y) and (x', y') points
  }
Step 5 : End
```

Fig. 6 Full search Algorithm's pseudocode

3.3 Methodology Used

3.3.1 Kohonen's Self-Organizing map Neural Networks

Artificial Neural Networks or simply ANNs are the generalization of biological neural networks which have been developed after the capability and usefulness of human brain. Human brain is a network of massively parallel neurons distributed throughout. In other words, it can be said that neural networks mimic the behaviour of human brain. Neural networks were basically developed after the theory of Mclloch Pitts model given in 1943. They designed the neural network that time using the electrical circuits of that time. The first model of the ANN was known as the first artificial practical neuron. This basically took the concept of human nervous system. Millions of neurons constitute to form the biological neural networks connected by means of synapses a medium. ANNs are quite popular today because of their characteristics like: learning capability, non-linear behaviour, adaptiveness, fast real time operation, generalization, and simple implementation features like only input output mapping [25], [26].

Moreover; all ANNs are categorised in two main categories: 1) feedforward neural newtorks and 2) feedback neural networks. Feedforward networks allow information flow in only in forward direction while feedback networks send output back towards the input as to form the loops. Hence feedforward networks are known as static networks or non-recurrent networks while feedback networks are known as dynamic or recurrent neural networks. Furthermore; each artificial neural network has a different number of inputs and outputs, which varies according to the type of the problem. All the neural networks learn from experiences or examples. Hence learning in ANN plays a very important role in validation/testing procedure ultimately. Learning is also known as training and are of three types i.e. supervised (teacher based), unsupervised (without a teacher), and reinforcement learning (trial and error based). Supervised learning is most commonly used in many applications like face recognition etc. Unsupervised learning is the advancement over supervised learning and does not require supervisor/teacher.

Kohonen Self-Organizing Maps (or just Self-Organizing Maps, or or SOMs) and ART-Map neural networks are

the two popular methods of unsupervised learning in ANNs. Kohonen Self-Organizing Maps or SOMs, are a type of neural network which were developed in 1982 by Tuevo Kohonen, a professor emeritus of the Academy of Finland. Self-Organizing Maps are aptly named. "Self-Organizing" is because no supervision is required. SOMs learn by their own unsupervised competitive learning. "Maps" is because they attempt to *map* their weights to conform to the given input data. The nodes in this network attempt to become like the inputs presented to them. In this sense, this is how they learn. They can also be called "Feature Maps", as in Self-Organizing Feature Maps. Retaining principle 'features' of the input data is a fundamental principle of SOMs, and one of the things that makes them so valuable. Specifically, the topological relationships between input data are preserved when mapped to a SOM network [25], [27],[28].

In this study, the SOM network is used to cluster motion vectors. The number of samples is equal to the number of the motion vectors occurring in the network. The coordinate values of the motion vectors are given to the network as input values; therefore, the number of inputs is 2. A maximum of 3 different moving regions (maximum number of outputs can be 3) can be determined with a 3-output network (or three output clusters) in this structure. The clusters are formed as the number of output weights that are updated; that is, if the weights of 2 outputs are updated, then the number of the cluster is 2.

3.3.2 Architecture

Each node in the SOM is mapped to neuron in the neural network. The architecture of SOM is shown in the "Fig. 7".

The neighborhood of the radii R=2, 1 and 0 are shown in the "Fig.8" for a rectangular grid and in "Fig. 9" for hexagonal grid. In each illustration, the winning unit is indicated by the symbol "#" and the other units are denoted by "*". Note that each unit has eight nearest neighbors in the rectangular grid, but only six in the hexagonal grid. Winning units that are close to the edge of the grid will have some neighbourhoods that have fewer units than that shown in the respective figure [28].

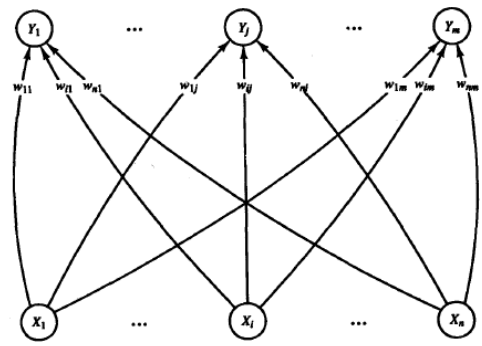

Fig. 7. Kohonen self-organizing map [28]

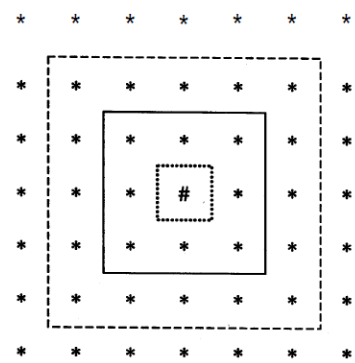

Fig 8. Neighborhood for rectangular grid [28]

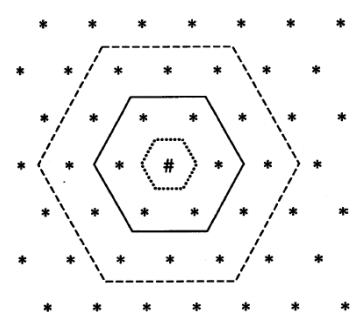

Fig.9. Neighborhoods for hexagonal grid [28]

3.3.3 Algorithm

Step 0. Initialize weights w_{ij}. (possible choices are discussed below). Set topological neighboring parameters. Set learning rate parameters.

Step 1. While stopping condition is false, do step 2 to 8.

Step 2. For each input vector x, do Steps 3 to 5.

Step 3. For each j, compute: D(j) = Σ(wij – xi)2

Step 4. Find index J such that D (J) is a minimum.
Step 5. For all units j within a specified neighborhood of J, and for all I:
Wij (new) = wij (old) + α [xi – wij(old)].
Step 6. Update learning rate.
Step 7. Reduce radius of topological neighborhood at specified times.
Step8. Usually the stopping criterion is a fixed number Of iterations or till radius becomes zero or the weight matrix reduces to a very negligible value.

4. Result and Discussion

In order to perform motion analysis the program is developed using advance java programming which can be applied for any video sequence.

Fig.10. Example of a simple video sequence (One object)

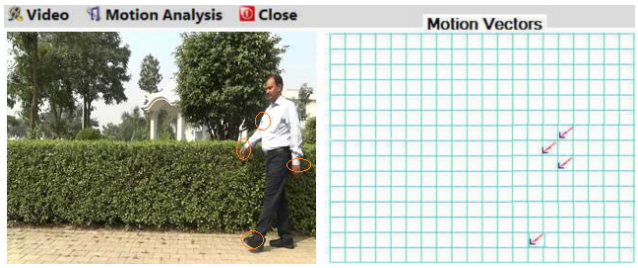

Fig. 11 Program Interface for motion estimation (Position-1, at t sec)

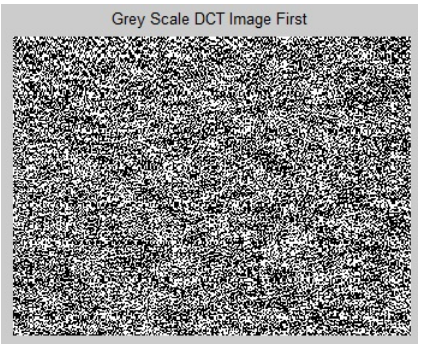

Fig. 12. Gray scale image of 9

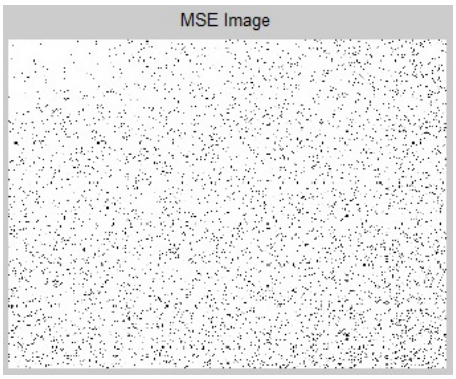

Fig. 13 Mean square error of original image is 262.05 (PSNR = 23.98)

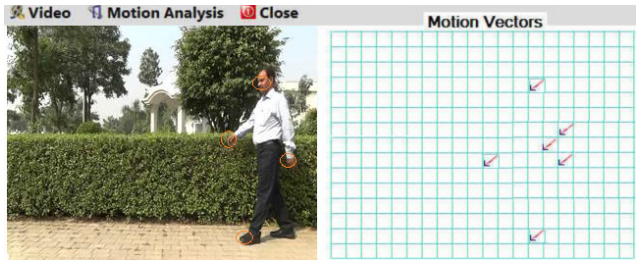

Fig. 14: Program Interface for motion estimation (Position-2, after t+3 sec)

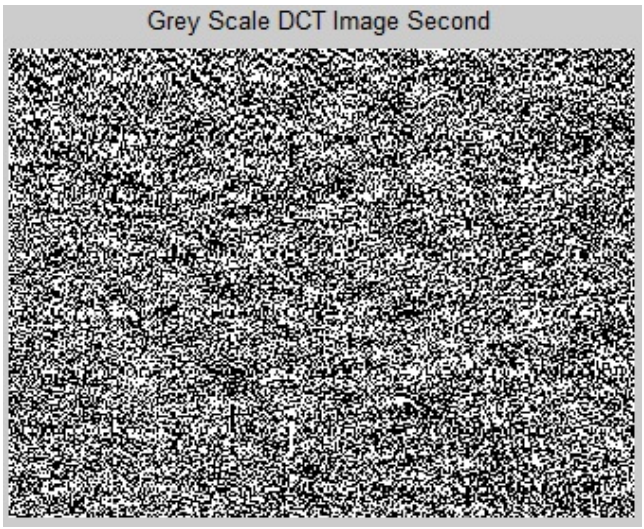

Fig. 15: Gray scale image of 10

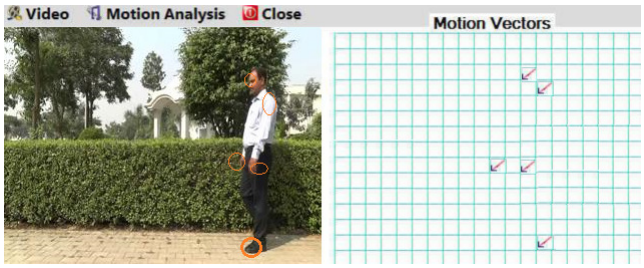

Fig. 16 Program Interface for motion estimation (Position-3, after t+6 sec)

Fig. 17 Program Interface for motion estimation (Position-4, after t+9 sec)

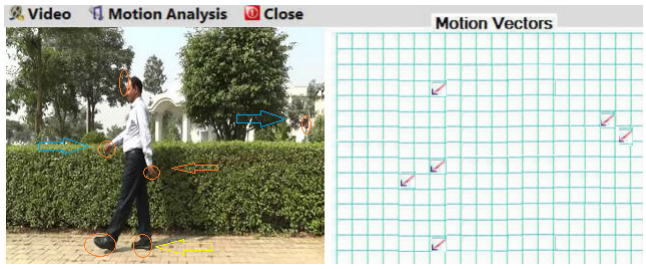

Fig. 18. Program Interface for motion estimation (Position-5, after 12 sec)

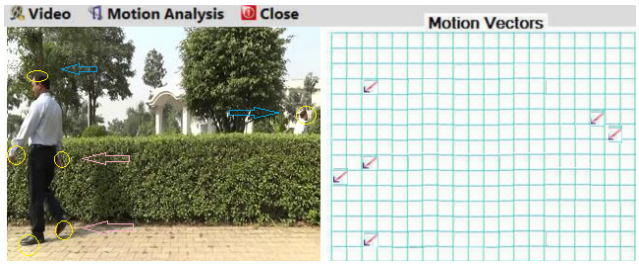

Fig. 19 Program Interface for motion estimation (Position-6, after 15 sec)

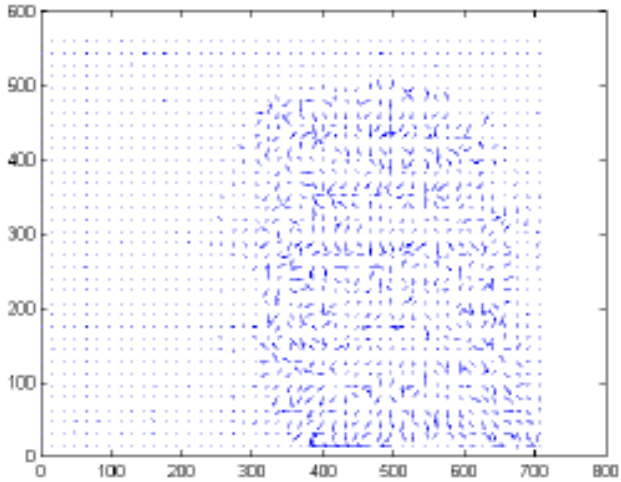

Fig. 20. Motion vector in detail

Table 1: Processing Time (in milliseconds)

S.No.	Activity	Time (in ms)
1	Time for two frames	2000 ms
2	Obtaining RGB values	0ms
3	Converting grayscale	56 ms
4	Finding edges	200ms
5	Threshold and finding active regions	67 ms
6	Full Search algorithm	500 ms
7	Competitive learning networks	50 ms
8	SOM	35 ms

The motion vectors and centers of the clusters can be seen in the screen by the users and the user can determine the clustering method, number of iterations, learning rate of the self-organizing map (SOM) learning network, block

size, and number of groups. Furthermore, the user can observe the execution time of the algorithm used. A screenshot of the software developed is shown in "fig. 11" to "fig. 19". In the video scene that is analyzed, the motion vectors formed and the processing times of the algorithms can be seen in the "Table1".

Screenshots, and the detection and classification of the direction of the motion in these screenshots obtained in the experimental studies, are displayed in "Fig. 21" Self-organizing map (SOM) neural network is used as a clustering method. The video sequences utilized are at pixel dimensions of 320 x 240. The blue and red circles displayed by the program indicate the centers of the clusters, which denote the moving regions. The red circle displays the center of the most intensive motion. The absolute errors obtained as a result of 30 different measurements of the clustering methods are given in graph obtained in "fig.22".

Fig. 21 An example of video sequence

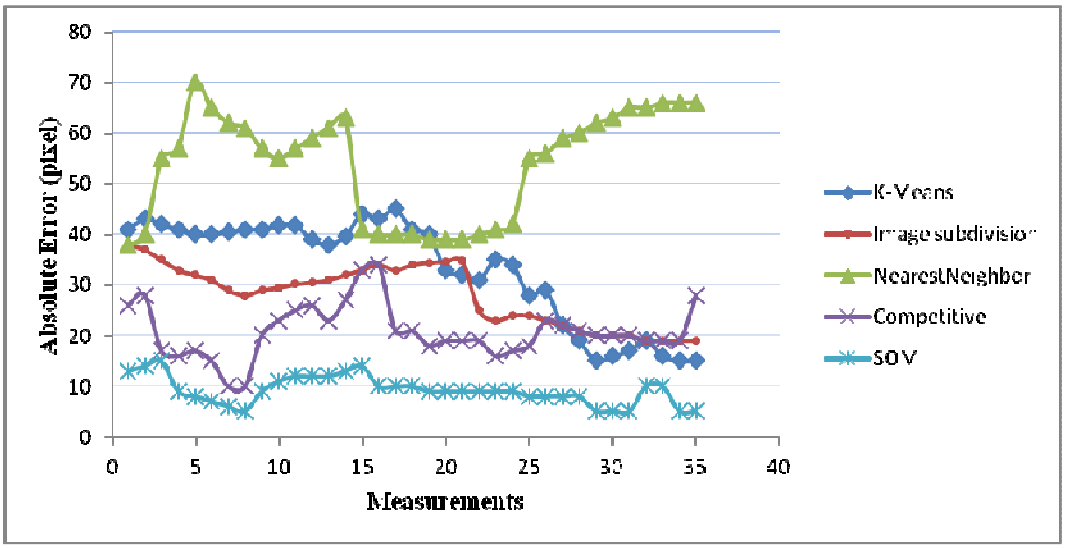

Fig. 22. Error of different methods

The average durations and mean absolute errors of the algorithms (eg k-means, competitive learning, SOM) used in the study are given in the Table-2. The average durations are calculated by taking the average value of 50 different values, taken from six different video sequences, into consideration. The sizes of the macro-blocks utilized within the study are set to 10 x 10, 16 x 16, and 20 x 20 pixels. The number of moving macro-blocks is limited to 10. The iteration count is taken as 30. Since pre-processing algorithms are carried out in terms of pixels, a change in the size of the macro-block does not affect the processing period.

Moreover, as the macro-block size in the full search algorithm increases, the search region and process time increase as well. In order to determine the success of the algorithms, the absolute error is obtained according to 20 different measurements, which are calculated with the help of Equation below [29]:

$$\text{error} = |x\text{-}xapprox| + |y\text{-}yapprox|$$

In this equation, x and y denote the true values, and x_{approx} and y_{approx} denote the approximate values. It is seen that the SOM has the smallest error value out of the 5 algorithms discussed above. On the other hand, the processing time of the SOM lasts longer compared to other algorithms like k-means, competitive learning etc. But the error is less compared to these algorithms. The average durations and absolute errors of the algorithms are obtained from a computer with configurations: Intel(R) Pentium(R) corei3, 2.16 GHz processor, 1 GB RAM, and Windows 7 operating system. In addition, the camera frame rate used within the system is 25.

Table 2: Average duration and absolute errors of the algorithms.

Process		Time			Error
		10x10 pixel block	16x16 pixel block	20x20 pixel block	
Preprocessing operations	Time for 2 frames	240 ms	240 ms	240 ms	
	Obtaining RGB values	1.75 ms	1.75 ms	1.75 ms	-
	Converting gray scale	7.36 ms	7.36 ms	7.36 ms	-
	Finding edges	55.66 ms	55.66 ms	55.66 ms	-
	Cleaning noises	21.66 ms	21.66 ms	21.66 ms	-
Full search algorithm		16.88	101.83	238.74	
k-Means algorithm		16.07 μs	9.86 μs	7.61 μs	32.74
Nearest neighbor algorithm		58.62 μs	44.45 μs	37.47 μs	57.09
Image subdivision algorithm		319.23 μs	170.18 μs	131.13 μs	37.47
Competitive learning network		319.23 μs	170.18	131.13	10.72
Self-organizing map (SOM)		***400.23µs***	***355.89 µs***	***243.45 µs***	**5.66**

5. Conclusion

This paper basically has been used to find the active dense regions of multiple frames obtained from the motion vectors in video sequences. The process to find the active regions can be divided in some steps like: pre-process, finding the motion vectors, clustering and finally active dense regions. Motion vectors are find out using a software program developed in Java language and the clustering has been carried out using Self-organizing map (SOM) neural network. Moreover, it is also compared with the other methods used so far for the clustering like k-Means algorithm, nearest neighbour algorithm, Image subdivision algorithm, and Competitive learning network. Consequently It observed that the competitive Self-organizing map (SOM) provides more accurate results and the clusters are obtained automatically according to the motion vectors. Furthermore, more motion centres can be determined by increasing the number of classes or output in environment that have scattered motion. It is also seen from the graph and table that the error is also reduced. On the other hand, the total duration of the SOM is considerably important when the moving objects' speeds are very high. Therefore, the duration performances of the algorithms are also compared. Adaptive resonance theory may also be used for the classification purpose in future. Convolution neural network, an advance area of neural network based on visual learning capability of

References

[1] Motion clustering on video sequences using a competitive learning network, Salih G ORG UNO_GLU1;, S_afak ALTAY2

[2] L. Shao, L. Ji, Y. Liu, J. Zhang, \Human action segmentation and recognition via motion and shape analysis", Pattern Recognition Letters, Vol. 33, pp. 438{445, 2012.

[3] S.J. McKenna, S. Jabri, Z. Duric, H. Wechsler, A. Rosenfeld, \Tracking groups of people", Computer Vision and Image Understanding, Vol. 80, pp. 42{56, 2000.

[4] C. Maduro, K. Batista, P. Peixoto, J. Batista, \Estimation of vehicle velocity and traffic intensity using rectified images", 15th IEEE International Conference on Image Processing, pp. 777{780, 2008.

[5] H.S. Mohana, M. Ashwathakumar, G. Shivakumar, "Vehicle detection and counting by using real time traffic flux through differential technique and performance evaluation", International Conference on Advanced Computer Control, pp. 791{795, 2009.

[6] C. Ozkurt, F. Camci, "Automatic traffic density estimation and vehicle classification for traffic surveillance systems using neural networks", Mathematical and Computational Applications, Vol. 14, pp. 187{196, 2009.

[7] H. Xiong, M.N.S. Swamy, M.O. Ahmad, I. King, "Branching competitive learning network: a novel self-creating model", IEEE Transactions on Neural Network, Vol. 15, pp. 417{429, 2004.

[8] J.Z. Lei, A. Ghorbani, "Network intrusion detection using an improved competitive learning neural network", Communication Networks and Services Research, pp. 190{197, 2004.

[9] H. Gu, "Adaptive synchronization for competitive neural networks with different time scales and stochastic perturbation", Neurocomputing, Vol. 73, pp. 350{356, 2009.

[10] J. Ste_en, M. Pardowitz, J.J. Steil, H. Ritter, "Integrating feature maps and competitive layer architectures for motion segmentation", Neurocomputing, Vol. 74, pp. 1372{1381, 2011.

[11] J.M. Jager, W.I. Schoollhorn, "Identifying individuality and variability in team tactics by means of statistical shape analysis and multilayer perceptrons", Human Movement Science, Vol. 31, pp. 303{317, 2012.

[12] J. Chey, S. Grossberg, E. Mingolla, "Neural dynamics of motion processing and speed discrimination", Vision Research, Vol. 38, pp. 2769{2786, 1998.

[13] J.C. Bazin, C. Demonceaux, P. Vasseur, I.S. Kweon, "Motion estimation by decoupling rotation and translation in catadioptric vision", Computer Vision and Image Understanding, Vol. 114, pp. 254{273, 2010.

[14] J.H. Kim, M.J. Chung, B.T. Choi, "Recursive estimation of motion and a scene model with a two camera system of divergent view", Pattern Recognition, Vol. 43, pp. 2265{2280, 2010.

[15] J.H. Kim, M.J. Chung, B.T. Choi, "Recursive estimation of motion and a scene model with a two camera system of divergent view", Pattern Recognition, Vol. 43, pp. 2265{2280, 2010.

[16] S. Khalid, "Activity classification and anomaly detection using m-mediods based modelling of motion patterns", Pattern Recognition, Vol. 43, pp. 3636{3647, 2010.

[17] S. Gorgunoglu, S. Altay, B. Sen, "Estimation of dominant motion direction on video sequences", 2nd International Symposium on Computing in Science & Engineering, pp. 9{15, 2011.

[18] A. Dosovitskiy, P. Fischer, E. Ilg, P. H¨ausser, C. Hazırbas¸, V. Golkov, P. van der Smagt, D. Cremers, and T. Brox. FlowNet: Learning optical flow with convolutional networks. In ICCV, 2015.

[19] A. Cavusoglu, S. Gorgunoglu, "A fast fingerprint image enhancement algorithm using a parabolic mask", Computers & Electrical Engineering, Vol. 34, pp. 250{256, 2008.

[20] Z. Telatar, "Video dizilerinde ayrit sezmeye dayali hareket kestirimi", Gazi Universitesi Muhendislik Mimarlik Dergisi, Vol. 24, pp. 245{255, 2009 (article in Turkish).

[21] R.T.N. Rajaram, V. Vasudevan, \Optimisation of the one-dimensional full search algorithm and implementation using an EPLD", 13th International Conference on VLSI Design, pp. 336{341, 2000.

[22] Z. Jing, J. Liangbao, C. Xuehong, \Implementation of parallel full search algorithm for motion estimation on multi-core processors", 2nd International Conference on Next Generation Information Technology, pp. 31{35, 2011.

[23] E. Arcaklio_glu, A. Cavusglu, A. Erisen, "Thermodynamic analysis of refrigerant mixtures for possible replacements for CFCs by an algorithm compiling property data", Applied Thermal Engineering, Vol. 26, pp. 430{439, 2006.

[24] E. Arcaklioglu, A. Cavusoglu, A. Erisen, "An algorithmic approach towards finding better refrigerant substitutes of CFCs in terms of the second law of thermodynamics", Energy Conversion and Management, Vol. 46, pp. 1595{1611, 2005.

[25] Krishan Kumar, "Self Organizing Map (SOM) Neural Networks For Air Space Sectoring", in IEEE International Conference Computational Intelligence & Communication Networks, Udaipur, India, 17-18, Nov'2014.

[26] S. Haykin (1994),'Neural Networks: A Comprehensive Foundation', *Prentice Hall*, Prentice Hall PTR Upper Saddle River, NJ, USA

[27] T. Kohonen, "Generalizations of the self-organizing map," Neural Networks. IJCNN '93-Nagoya. Proceedings of 1993 International Joint Conference on, page(s): 457 - 462 vol.1 Volume: 1, 25-29 Oct. 1993.

[28] Laurene Fausett, Fundamentals of Neural networks: Architecture, Algorithm and Applications, Pearson Education, ISBN 978-81-317-0053, 1994**.**

[29] Salih GORGUNOGLU*;*, Safak ALTAY, "Motion clustering on video sequences using a competitive learning network", Turkish Journal of Electrical Engineering & Computer Sciences, pp 400-411, Published Online: 17.01.14.