Video Target Tracking by using Competitive Neural Networks

ERNESTO ARAUJO[1,2,3]
[1]Integration and Testing Laboratory – LIT
Space Technologies and Engineering – ETE
Instituto Nacional de Pesquisas Espaciais – INPE
Av. Astronautas, 1758, 12.227-010, São José dos Campos

[2]Health Informatics Dept. – DIS
Universidade Federal de São Paulo – UNIFESP
Botucatu, 862, 04023-062, São Paulo

[3]Hospital Municipal Dr. José de Carvalho Florence
Av. Saigiro Nakamura, 800, 04023-062, São José dos Campos
BRAZIL
ernesto.araujo@{lit.inpe.br}{unifesp.br}

CASSIANO R. SILVA, DANIEL J. B. S. SAMPAIO
Electrical Engineering Department
Universidade Estadual Paulista – UNESP
Av. Dr. Ariberto Pereira da Cunha, 333,
12.516-410, Guaratinguetá
BRAZIL
cassiano_rs@unesp.br, dsampaio@feg.unesp.br

Abstract: A target tracking algorithm able to identify the position and to pursuit moving targets in video digital sequences is proposed in this paper. The proposed approach aims to track moving targets inside the vision field of a digital camera. The position and trajectory of the target are identified by using a neural network presenting competitive learning technique. The winning neuron is trained to approximate to the target and, then, pursuit it. A digital camera provides a sequence of images and the algorithm process those frames in real time tracking the moving target. The algorithm is performed both with black and white and multi-colored images to simulate real world situations. Results show the effectiveness of the proposed algorithm, since the neurons tracked the moving targets even if there is no pre-processing image analysis. Single and multiple moving targets are followed in real time.

Key–Words: Image motion, Target Tracking, Neural Network, Video Digital Camera, Computational Intelligence.

1 Introduction

Through biological vision system it is possible, for instance, to obtain the position and properties of objects and their relationship with themselves and with the environment in which they are inserted. Computer vision approach is related to the study of computational elements of vision and, in special, to emulate vision mechanisms by using artificial systems. In this sense, this paper presents an artificial vision system for emulating human mechanisms when interested in target tracking as well as position and trajectory identification.

Target tracking has been of great interest for application in diverse areas, such as surveillance system, radar, defense, entertainment, automation and manufacturing, only to mention few [1, 2]. The proposed artificial vision system is based on computational intelligent techniques inspired by biological neural model of human beings. Known as Artificial Neural Networks (ANN), these techniques are mainly characterized by its ability to learn through experiences, to adapt to adverse conditions, and to be tolerant to noise [3, 4, 5]. These features make the ANN a great field of study and drive it to reach success in solving real problems such as identification, classification, digital image processing, and control.

Kohonen Neural Network (KNN) is the technique within artificial neural network designed, here, to find the position and to track single and multiple moving targets. Based on a previous approach for image motion analysis [6], this technique is applied to video digital monochromatic and multicolor image sequence. This kind of Kohonen neural network differs from the classical Kohonen self-organizing map (SOM). In the latter the winner neuron and its neighbor neurons update their weight by using a smooth function. The approach employed here, the winner neuron is adjusted in a step forward direction while the loser neurons move by using a step backward function contrary to the winner neuron movement or may stay fixed, depending on the problem. The advantage of using such a system is related to its ability for offline and on-line patter recognition in vision analysis.

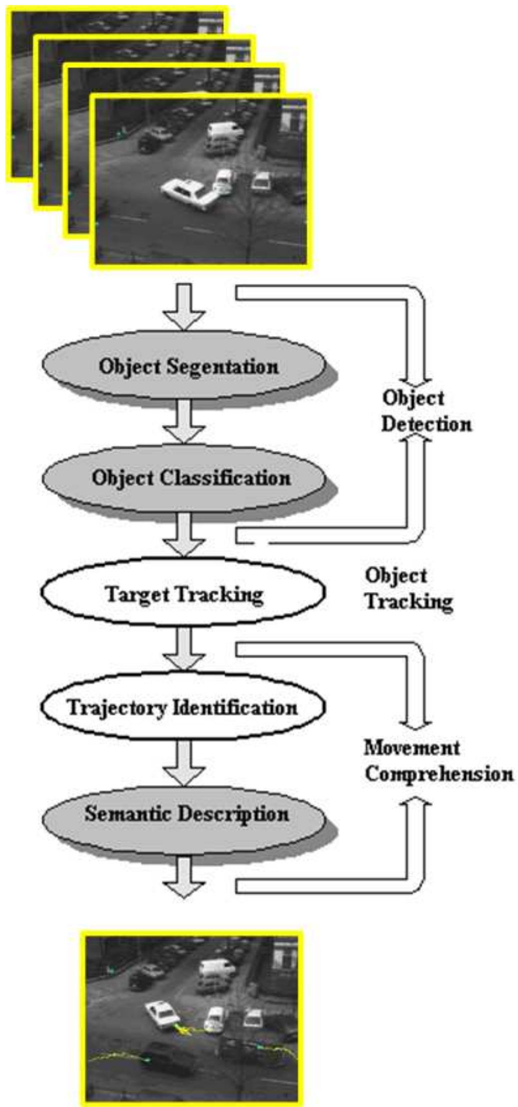


Figure 1: Moving Target Tracking Diagram [2].

The use of computational/artificial intelligence techniques for tracking and to pursuit a moving target is not limited to Kohonen artificial neural network. Different techniques have been used to model motion [7], specially when dealing with video image analysis [8]. Among many, computational/artificial intelligence techniques are alternatives to build up such vision motion systems. For example, a tracker using fuzzy logic is implemented in [5]. Multiple tracking targets in an unstructured environment by using fuzzy approach is accomplished in [9]. An image tracking by using fuzzy logic is developed in [10]. Fuzzy logic and Kalman filter are employed for target tracking in [11]. There also are hybrid systems that combines neural networks and fuzzy logic to make the tracker [12] where the position of the target is estimated by angle and distance estimators assuming that moving target radiates narrow band waves. In [13] a multiple elastic modules model based on Self-Organizing neural network incorporating a mechanisms analogous to thermodynamic temperature value is described to escape of poor local minima when used for passive tracking problem. A hybrid fuzzy neural network approach based on Kohonen feature map is described in [14]. A neural network directed conditional probability generator and sequential classifier based on Bayes decision rule is found in [15] for target dynamic behavior and target classification. The dynamics is determined by using velocity/acceleration and curvature sequences from each track. A new biological neural network inspired in the Hodgkin and Huxleys biological membrane model is presented in [16] to dynamic collision-free trajectory generation in a non stationary environment. When dealing with visual information, the approach in [17] uses an artificial neural network that receives as input the image for supplying the direction, distance of the scenario and the position of the moving target. Inspired by the fly visual system of Diptera males for conducting fast-moving aerial pursuits and interceptions of females a novel network for determining the motion of objects is proposed in [18] that uses as input the light intensity variation. Also inspired in animals, a neural network structure based on bat is used to face recognition and velocity of a dynamic target moving toward the camera [19]. A feedforward neural network is used in [20] for static and moving identification for robot control. This approach also predicts the position of the target and of the manipulator by using time derivatives of the position of the object. In [21], a non-supervised neural network was employed in the development of a data composition system in a radar data multichannel mechanism for target identification. The characteristics of these data from three distinct radar channels are extracted from digital signal processing. Hierarchical Artificial Neural Network (HANN) to detect and track moving targets in video sequences is presented in [22]. The use of neural network working in conjunction with Kalman filter for target tracking is accomplished in [23]. In turn, Kalman filter with a self-constructing neural fuzzy inference network algorithm for target tracking is presented in [24]. Multiagent and neural network are employed to detect in air image characteristics, shapes and objects [25]. Backpropagation neural network and learning vector quantization neural network are employed for data association when addressing multitarget tracking [26]. A Hopfield neural network based on data association technique is implemented in [27]. Data association and Hopfield neural network for multiple target tracking is presented in [28]. Multitarget Tracking Data Association also using Hopfield neural network is described in [29]. An adaptive neural network retrain-

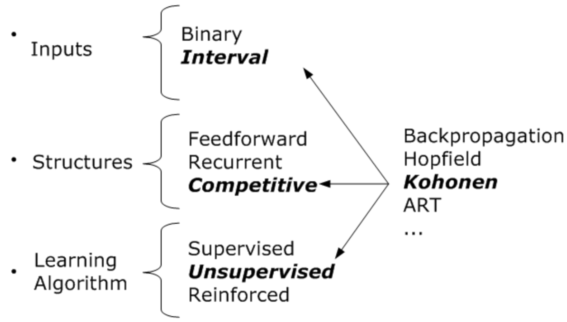


Figure 2: Kohonen Neural Network Features.

ing for unsupervised video object segmentation and tracking of stereoscopic video sequence is proposed in [30]. A moving target radar detection based on convolutional neural networks dealing with highly complex and nonstationary cluttered environments is proposed [31]. A second order recurrent artificial neural network for target tracking in passive multistatic radar application is developed in [32]. There are among these techniques limitations as the possibility of tracking just near targets while in other techniques the system is not fast, like those that use back-propagation neural network, or fail in tracking in presence of crossing targets when the sample rate of the target is relatively low.

In general, moving target tracking activities can be represented as shown in Fig. 1 [2]. The focus of this paper is related to the object tracking (target tracking) and movement comprehension (trajectory identification) as detached. In this work, the proposed approach is extended to real world, day-to-day problems in order to find the position and to pursuit moving targets. The modified KNN approach is an alternative when dealing with video digital image sequence by using, for instance, digital camera or other electronic devices.

The identification of a target presence is achieved by the ground subtraction technique. It is designed both to find and to track single and multiple monochromatic or color moving targets. Black and white images at first, and colored images after, are employed to demonstrate the effectiveness of the algorithm. This algorithm is applied to each one of the situations to track any moving target that appears in the camera field of vision. With the combination of neural networks and ground subtraction technique, it is possible to notice that the proposed tracking system in both video digital images shows an efficient accomplishment.

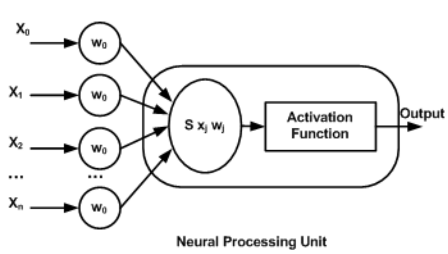


Figure 3: Neural Processing Unit (neuron).

2 Neural Networks for Target Position Identification and Tracking

Artificial neural networks are techniques of artificial intelligence that uses as a model, the human brain, where neurons are trained to answer to input patterns. The main features of the Kohonen Neural Network (KNN) are summarized in Fig. 2.

Kohonen Neural Network is characterized as possessing *multi-valued* inputs and is denominated *unsupervised* net since it does not require a waited output for a given input pattern. Contrary to supervised training, which is accomplished when a desired output corresponding to a given input is furnished, the KNN learning is obtained just presenting inputs to the network. In so doing, the network adapts their weights in such a way to follow data distributed in groups, i.e., clusters with similar features. The KNN is, then, an approach able to recognize patterns (clusterings) and to group similarities in the input data with the advantage of not requiring previous training or learning. The network updates its parameters while in use [3, 12, 15]. The neurons compete among themselves for the learning (training) of their synapses (weights) showing a *competitive* structure. Also termed competitive learning, the best neuron related to a given input is activated as the winner. Due to that, this competitive characteristic gives the neural network also the classification of “winner-takes-all” neural network.

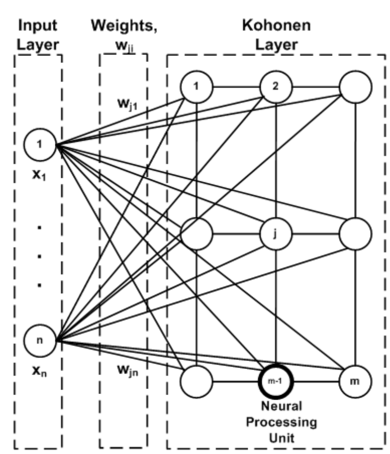
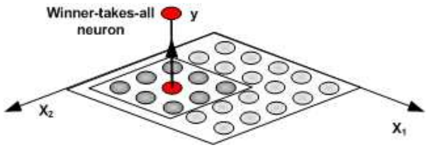
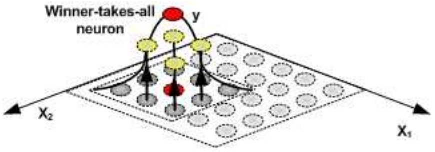


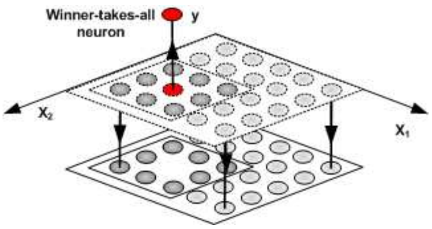
Figure 4: Kohonen Neural Network Basic Structure.



(a) KNN – Step forward function.



(b) SOM KNN – Gaussian training function.



(c) KNN – Step forward and backward training function.

Figure 5: Training in a 2D Kohonen Neural Network.

A neural processing unit (neuron) receives the input values and generates an output (Fig. 3). Each output of the neuron in the input layer is connected to all the neuron in the output layer [3], also denominate Kohonen layer (Fig. 4). The dimension of the input space is associated to each output neuron.

The training is given by the neighboring between the patterns and the weight of the neurons. Kohonen neural network identifies and classifies input patterns in classes using the geometric proximity. During the competitive learning, the neurons fight for training their own weights. For each input presented to the KNN only one neuron is chosen to be trained. The winner neuron assumes this condition due to the fact that it is the closest to the input clustering computed and grouped (pattern) from previous inputs.

Learning Mechanism in Unsupervised Algorithm

The learning mechanism of the KNN that determines by itself the winner neuron and its training according to the input patterns detects groups by geometric closeness of similar characteristics of input data. The learning mechanism is given by:

$$\|x - w_j\| = \|x - w_i\| \, \forall i \neq j \;, \tag{1}$$

where the input, x, corresponds to an output y in such a way that only one neuron is active, $y = +1$, while the other neurons are considered to be in inhibition. The winner neuron, the best to be trained, is, then stepped forward to a recognized pattern and grouped according to previous inputs (Fig. 5(a)). At each input data presented to the algorithm the winner neuron tends to approach the group. The Kohonen neural network algorithm is presented in Fig. 6.

The proximity of the neurons and the pattern, here representing the target, is determined by the Euclidean distance between the input vector, x, (position) and the synaptic weights, w_j, in all the dimensions that characterize the target:

$$D = \|x - w_j\| = \left[\sum_{i=1}^{n} (x_i - w_{ij})^2\right]^{1/2} \tag{2}$$

where "j" is the number of output neurons, and "i" is the number of elements in the input vector x and w. The proposed approach may be used with any dimensional problem and the states of the system of interest are represented as $x^k = \left[x_1^k, x_2^k, \ldots, x_n^k\right]$ that represent the n clusters, in such a way that k is related to the amount of elements of the input pattern. The number of clusters or classes, n, is assumed to be previously known [33].

The winner neuron, x_j, is the one that is closest of the vector x. Thus, the best neuron is the one who presents the smaller Euclidean norm:

$$x_j = min_j \|x - w_j\| \tag{3}$$

The learning mechanism is, then, described by the similarity matching:

$$\|x - \widehat{w}_j\| = min_{1 \leq i \leq n} \{\|x - \widehat{w}_i\|\} \tag{4}$$

Self Organizing Map in Unsupervised Learning

In the Kohonen neural network, also termed as Self Organizing Maps (SOM), all the neurons are trained by using an activation function, $A_{win.c}$ in the form:

$$\begin{cases} \widehat{w}_{j(new)} = \widehat{w}_{j(old)} + \beta A_{win.c} \left(x - \widehat{w}_{j(old)}\right) \\ \widehat{w}_{i(new)} = \widehat{w}_{i(old)}, \forall i = 1, 2, \ldots, n, i \neq j \;, \end{cases} \tag{5}$$

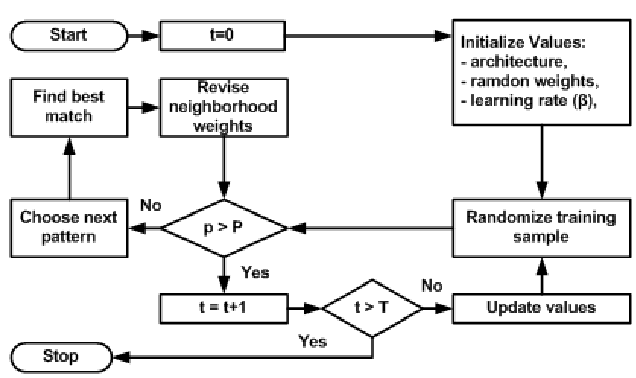
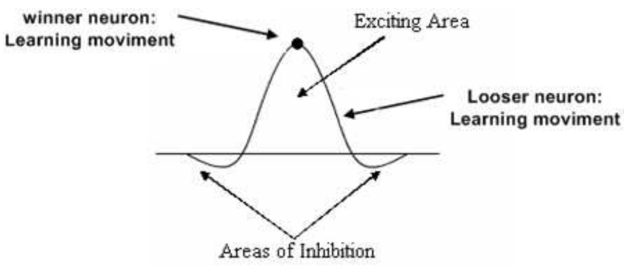
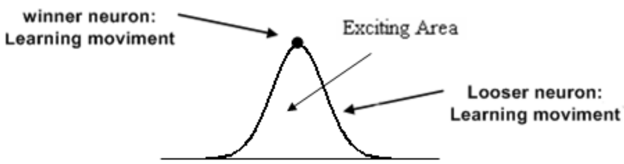


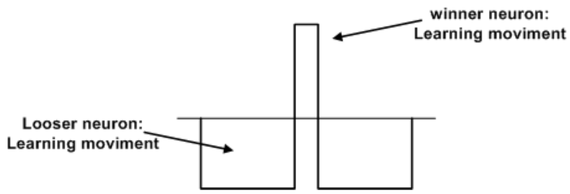
Figure 6: Kohonen Neural Network algorithm [34].



(a) KNN SOM – Gaussian training function.



(b) KNN – Step forward training function.



(c) KNN – Step forward and backward training function.

Figure 7: Training in 1D Kohonen Neural Network.

where β is the *learning rate*. The input patterns and the weights of the neurons must be normalized. The normalized vector, $\widehat{w}_j$, is given by:

$$\widehat{w}_j = \frac{w_j}{\|w_j\|} \ . \qquad (6)$$

The activation function is most of time accomplished by using the Mexican hat or the Gaussian, respectively, presented in Fig. 7(a) and 7(b). When using this activation function all neurons in the neighborhood of the winner neuron are also trained. Nevertheless, they are stepped forward with lower intensity determined by the activation function (Fig. 5(b)).

Deterministic Unsupervised Learning

The updating stage in which the KNN teaches the winner neuron by driven it closer to the inputs as possible:

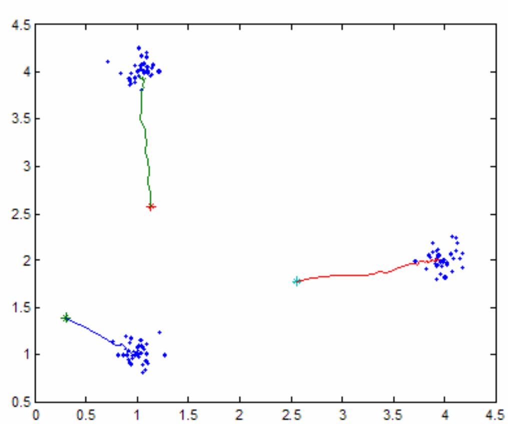
$$\begin{cases} \widehat{w}_{j(new)} = \widehat{w}_{j(old)} + \beta \left(x - \widehat{w}_{j(old)} \right) \\ \widehat{w}_{i(new)} = \widehat{w}_{i(old)}, \forall i = 1, 2, \ldots, n, i \neq j \ , \end{cases} \qquad (7)$$

where j-th winner neuron is stepped forward while the loser neurons remains the same.

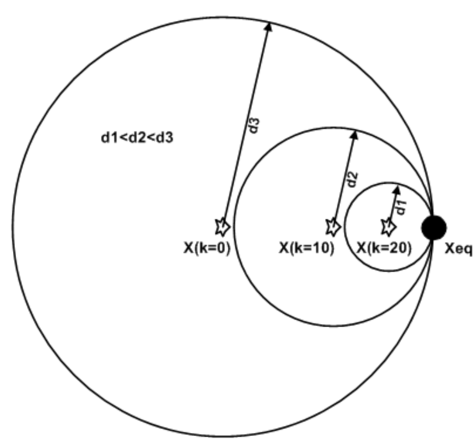
With the input data, the neural network trains the winner neuron w_i approximating it from the inputs when using eq. (7). The net perform it for each input. The weights to the neuron approximate of that of the inputs. In the tracking problem it means that, geometrically, the neuron approximates to the desired target to be pursuit.

Different of the Kohonen SOM in which all neurons are trained, in this work, only the winner neuron is trained in a step forward manner or it is trained in a step forward manner and the loser neurons are trained in a step backward manner represented, respectively, in 2D and 1D KNN training as Fig. 5(c) and 7(c).

These characteristics motivate the use o KNN in identifying the position and trajectory of fixed and moving targets, since there is no exigencies of an anticipate training and could be carried out in a fast manner. In order to illustrate the algorithm, consider the clusters in Fig. 8(a). These clusters represent pixels



(a) Position identification of fixed clusters.



(b) Neuron approximation of static target.

Figure 8: Example of KNN training for target position identification.

of targets to be pursuit or the position identification in the image. Three neurons are randomly generated. It is possible to verify that they converge to each of the cluster. The lines are the trajectory the neurons follow to reach the target in a convergent manner. When the target is not static but moving, the neurons track the targets in the same manner. The difference is concerned to faster time convergence after initial position identification. Since the initial condition for the next iteration is the final condition of the latter iteration, the neuron is closer to the target than it was at the beginning of the target tracking.

3 Video Target Tracking Application

Examples of sampled digital image sequences are employed to demonstrate the effectiveness of the proposed algorithm when applied in video target tracking. Images are presented sequentially to the algorithm that is in charge of identifying pixels associated to the target and follows it onwards. The neural network is trained in real time to identify the target of each image.

The technique of ground subtraction is used in this paper to verify the feasibility of KNN in tracking multiple target problems. When each new image is presented to the algorithm, a ground subtraction is carried out between the current image and the background image.

An image is composed by pixels that are related to numbers, which have the information of cell positions and colors of the pixels, to compose a matrix. The system stores an image in the form of number in matrices with the information of position and color of each pixel. As the images are obtained from a camera and shown to the algorithm, they are compared to a standard image previously determined – background – to be subtracted. When the subtraction between the matrices is carried out, the result is a new matrix which contains only the pixels that define the target. After that, the resulting matrix may have elements different of zero indicating that the pixel belongs to the target, or not.

A subtraction between the images will produce values close to zero for representing the background or scenery, and values different of zero for the target. This matrix is, then, supplied to the Kohonen neural network in order to find the position of the target in the field of vision determined by the digital camera. Finally, these pixels are used by the neural network to fulfill the target tracking.

As real world examples, the algorithm was employed to two distinct applications. The first one deals with white and black images and the second example utilize colored images. In both examples, there is no pre-processing image analysis in order to verify the influence of environmental effects upon the effectiveness of the algorithm.

3.1 Black and White Video Target Tracking

The first example of target tracking is performed by using black and white images (Fig. 9). Cars and a person are moving in the image. Since there are four targets, a minimal of four neurons is necessary in the neural network target tracking. When the targets appeared in the camera field of vision, immediately the neurons identified their presence and the track starts. The algorithm identifies the targets and the neuron that is nearest of each target.

There are three cars moving in the image. Dark cars dislocate faster than the white one, so with the dark cars, it is better to see that the neurons follow appropriately the targets. The neuron identified the presence of the target in the camera field of vision, and reached the target at the first image of the sequence.

Each image in a sequence is presented to the algorithm, and the algorithm trains each neuron through modifying their synapses to track the moving target. Looking sequentially to the images, it can be seen that the multiple targets move through the image. The neurons are initially distributed randomly all over the first image. It can be noticed that the neurons identify the targets and move to reach it.

As new images are presented, the targets move to other positions and the neurons follow it. The images are presented to the algorithm one by one and, in each of them, the algorithm identifies the valid clusters and tracks the moving target.

The yellow lines are the trajectory of the cars and represent the current positions of each neuron, which is following the target. There is another neuron on the left top of the image because there is a person moving along the space. Even being a small target, the neuron recognizes it as a moving and follows it.

The *Learning Rate* assumes an important whole in the training of the neural network since it defines the size steps the KNN follows the target. When to this parameter is assigned a low value, for instance $\beta = 0.01$, the track is not good; the neuron becomes slow and does not reach the target. In contrary, attributed to it higher values, the neuron can easily approximate to the target but may jump from one point to the other with out achieving the central point of the target. In so doing, there is no convergence of the algorithm and, so, of the target tracking. This parameter setting may be solved by different manners.

An adaptive variable learning rate is an alternative where the step size to up-to-date is also computed in

(a) From initial condition to target position 1.

(b) From target position 1 to position 2.

(c) From target position 2 to position 3.

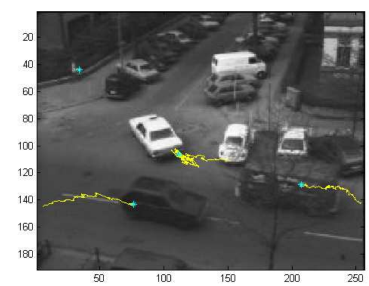
(d) From target position 3 to position 4.

(e) From target position 4 to position 5.

(f) From target position 5 to position 6.

(g) From target position 6 to position 7.

(h) From target position 7 to position 8.



(i) From target position 8 to position 9.

Figure 9: Black and White: Neuron departs from randomly initial position and identificates the target at the position i.

function of the error obtained from the difference of the target and the current neuron. Selecting a medium step size value is also another option. A fixed medium step size value is chosen in this paper.

In order to illustrate the effect of the subtraction technique, the colored final target tracking is shown in Fig. 10. When the subtraction is achieved, all the background color value comes close to zero. Every pixel that has color different about zero is related to the target and will be presented as input value to the neural network. The blue pixels are the background, and points with other colors go to the neural network to be trained. The car is on the left side of the image, and there are some points along the image that are different of blue. It means that the algorithm sees some

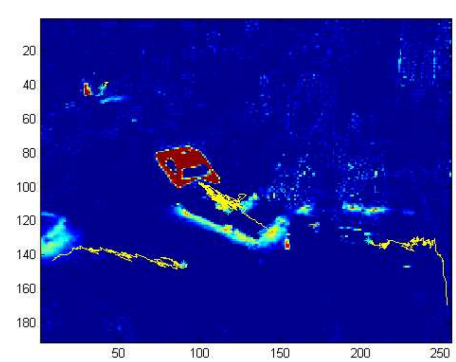


Figure 10: Background subtraction, example Fig. 9.

points of the background as points that belong to the target.

3.2 Color Video Target Tracking

The last simulation is performed by using color images. The video image sequence shows a street, where initially there is no car moving on the camera field of vision, but the car appears on the right side and starts crossing the image from right to left side. The neuron is initially on the right side of the image, stopped. When the car appears, the neuron perceives its presence and starts tracking the car. Looking at the figure, it can be seen that the car is on the left side of the image, and it is there because this is the last image.

In this example, ground subtraction technique is also employed to find the target. The result of the tracking is shown in sequence of images that compose the video (Fig. 11). Selected images are disposal to exemplify the final position of the neuron as emphasized (Fig.12).

The video shows a street, where initially there is nothing, but a car appears and moves along the camera field of vision. There are several images in a sequence, composing a video, but here, is shown only the final image. Initially, there is no car moving on the camera field of vision. A car appears on the right side and starts crossing the image from right to left side moving on the street. The neuron is initially on the right side of the image, stopped. When the car appears, the neuron perceives its presence and immediately starts

(a) From initial condition to target position 1.

(b) From target position 1 to position 2.

(c) From target position 2 to position 3.

(d) From target position 3 to position 4.

(e) From target position 4 to position 5.

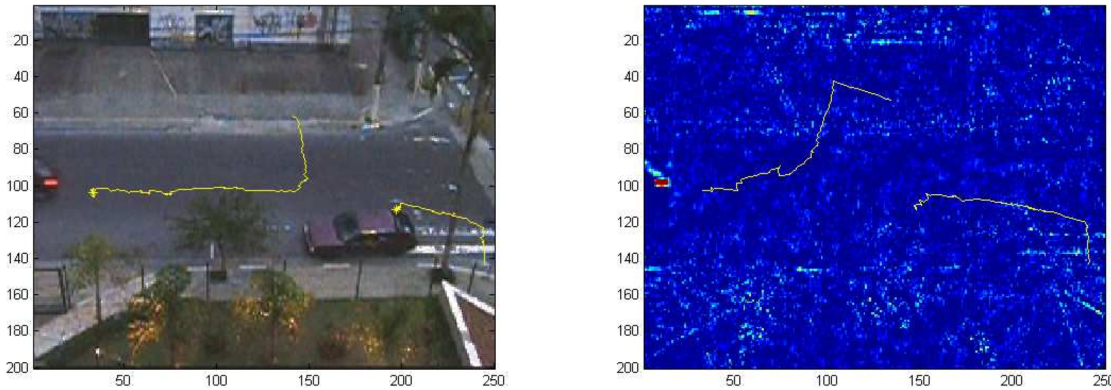
(f) From target position 5 to position 6.

(g) From target position 6 to position 7.

(h) From target position 7 to position 8.

(i) From target position 8 to position 9.

Figure 11: Color Image: Neuron departs from randomly initial position and identificates the target at the position i.



(a) Resulting tracking for original image, referring to Fig. 11.

(b) Resulting tracking for subtract image, other training.

Figure 12: Approximation and convergence of the neuron to target.

tracking the car.

It can be noticed in Fig. 11 that the car is on the left side of the image, and it is there because this is the last image. The neuron followed the target through all the pursuit, and identified its trajectory. The yellow line shows the trajectory of the target, and it can be seen that the car moved from right to left side of the image. With this simulation, it can be seen that this algorithm can be used in digital images to track objects moving in the camera field of vision.

3.3 Step Size Influence and Synaptic Convergence

Consider again the color video target tracking but, now, with only one initial condition. In order to illustrate the influence of step size distinct values ranging from 0.01 to 0.3 sizes are employed for training phase. The initial condition is randomly generated (and latter approximate to round number) but once determined is maintained for all the various step size. The graphics concerning the tracking trajectory from diverse analysis are depicted in Fig. 13

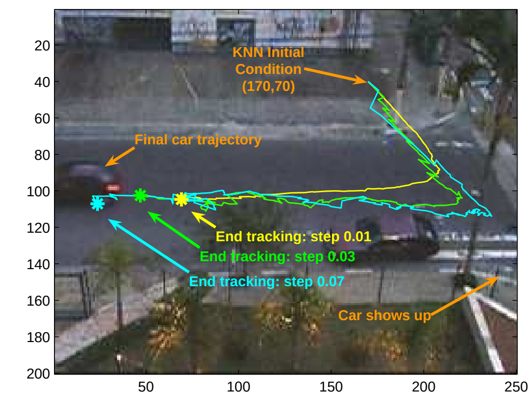
The tracking trajectories for a reduced number of initial conditions (0.01, 0.03, 0.07) are selected, without loss of generality, to be plotted upon image, as shown in Fig. 13(a). This option is for making clear the identification of the distinct trajectories.

The tracking trajectory set for all the before mentioned step sizes in a 3D representation is presented in Fig. 13(b). This graph traces the trajectory of the target of synaptic trajectories, $w1$ and $w2$ on time. The Kohonen neural network initial condition is set up to $w1_{initial} = 170$ and $w2_{initial} = 70$.

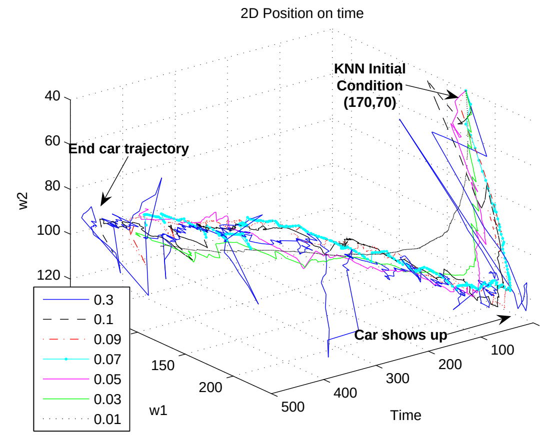
The synaptic trajectories of $w1$ and $w2$ are, respectively, represented in Fig. 13(c) and Fig. 13(d). The continuous convergence of the neuron to the target may immediately be realized. These neuron approximations are similar to the conceptual example presented in Fig. 8(b). The fact that, here, the target is not fixed but it is moving does not alter the convergence, except that the convergence is longer. Additionally, there are oscillations mainly due to two reasons. The first one is that the target moves not in perfect trajectory carrying by itself oscillations. The other reason is the step size as discussed onwards.

Finally, the number of iterations for each step size is illustrate in Fig. 13(e). They represent mean values for diverse video target tracking operation by using the same initial conditions and parameters.

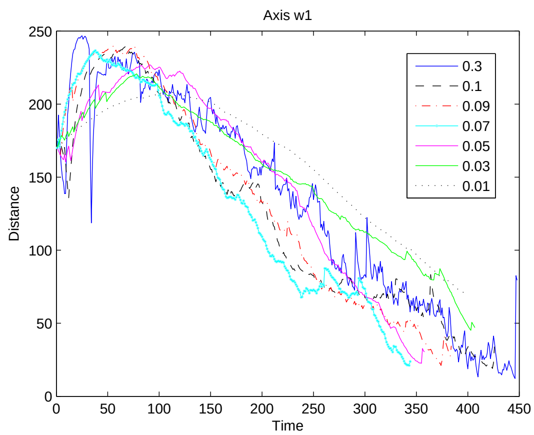
Through these graphics it is possible to notice the different trajectories obtained are strongly influenced by the step size. If from one standpoint the trajectory with 0.01 step size presents no oscillation, for the other hand it is not big enough to reach the target in the current image. The neuron approximate to the target but without ever achieving it. It is emphasized mainly in the beginning of the tracking trajectory. The target is very far and the neuron is not able to arrive to it, as are the other step sizes. Despite that, it is able to follow the target with the advantage of being robust to variations on the image that is presented to the algorithm. In the other extremity, there is a 0.3 step size. When using this step value the neuron reaches very fast the target but the tracking presents some problem. The convergence become not stable since variations in the image became attractive targets for the neuron. If these oscillations are acceptable, the algorithm allows to approximate and to converge being able to track to the target. The middle term may be an interesting alternative. For instance, the 0.07 step size allows to approximate to the target very fast meanwhile presents



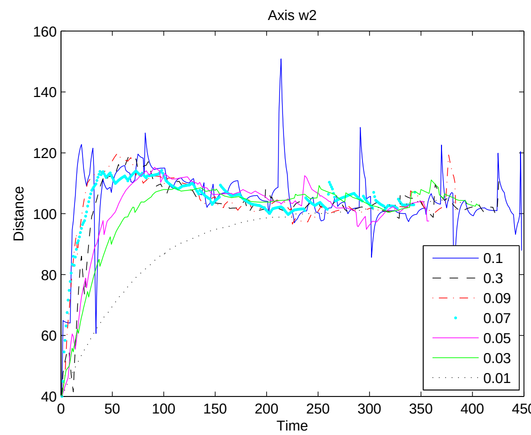
(a) Resulting tracking for diverse step sizes.



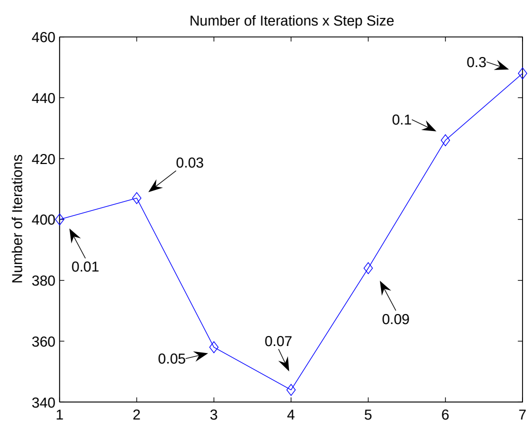
(b) Resulting 3D tracking for diverse step sizes.



(c) Step (synapses) convergence for w_1 with diverse step sizes.



(d) Step (synapses) convergence for w_2 with diverse step sizes.



(e) Number of iterations compared to diverse step sizes.

Figure 13: Step Size Influence and Synaptic Convergence of the neuron to target.

reduced oscillation. Moreover, the number of iterations when compared to diverse step sizes are reduced, as indicated in Fig. 13(e).

4 Conclusions

The objective of this paper is to show an algorithm for target tracking in digital image sequences using a modified Kohonen neural network. At first, the algorithm is applied to monochromatic images and with color images onwards.

This paper combines Kohonen neural network with ground subtraction approach for the identification of the target's pixels in the image and the tracking of the moving target.

Results show that the proposed approach can be applied to digital image processing for position identification and target tracking moving inside the camera field of vision.

References:

[1] J. I. Agbinya and D. Rees, "Multi-object tracking in video," *Real-Time Imaging*, vol. 5, pp. 295–304, 1999.

[2] L. Wang, W. Hu, and T. Tan, "Recent developments in human motion analysis," *Pattern Recognition*, vol. 36, pp. 585–6001, 2003.

[3] B. Krose and P. V. der Smagt, *An Introduction to Neural Networks*. New York: Academic Press, 1977, 8th ed.

[4] C. L. Nascimento and T. Yoneyama, *Artificial Intelligence in Control and Automation*. São Paulo: Ed. Edgard Blucher Ltda., 2000, (in Portuguese).

[5] S. J. Wu and C. T. Lin, "Global optimal fuzzy tracker design based on local concept approach," *IEEE Transaction On Fuzzy Systems*, vol. 10, pp. 128–143, 2002.

[6] F. Garcia and E. Araujo, "Visual multi-target tracking by using modified kohonen neural networks," in *International Joint Conference on Neural Network*. IEEE, 2008, pp. 4162–4167.

[7] P. K. Agarwal and et. al, "Algorithmic issues in modeling motion," *ACM Computing Surveys*, vol. 44, pp. 550–572, 2002.

[8] C.-W. Chang and S.-Y. LEE, "A video information system for sport motion analysis," *Journal of Visual Languages and Computing*, vol. 8, 1997.

[9] N. Rahman, A. R. Jafri, and M. U. Keerio, "Fuzzy behaviour based navigation of a mobile robot for tracking multiple targets in an unstructured environment," *WSEAS Transactions on Systems*, vol. 5, no. 6, pp. 1425–1430, 2006.

[10] H.-C. Kuo and L.-J. Wu, "An image tracking system for welded seams using fuzzy logic," *Journal of Materials Processing Technology*, vol. 120, 2002.

[11] H. Lazoff, "Target tracking using fuzzy logic association," in *Proc. IEEE*, 1998, pp. 2477–2460.

[12] C. W. Ma and C. C. Teng, "Tracking a near-field moving target using fuzzy neural networks," *Fuzzy Sets And Systems*, vol. 110, pp. 365–377, 2000.

[13] S. Shams, "Neural network optimization for multi-target multi-sensor passive tracking," *Proc. IEEE*, vol. 84, pp. 1442–1457, 1996.

[14] R. J. Kuo, S. C. Chi, and B. W. Den, "A fuzzy kohonens feature map neural network with application to group technology," in *Proc. IEEE*, 1999, pp. 3098–3101.

[15] X. Yu and M. R. Azimi-Sadjadi, "Neural network directed bayes decision rule for moving target classification," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 36, pp. 176–188, 2000.

[16] S. X. Yang and M. Meng, "Neural network approaches to dynamic collision-free trajectory generation," *IEEE Transaction on System Man and Cybernetic Part B-Cybernetics*, vol. 31, pp. 302–318, 2001.

[17] S. G. S. Cameron and F. H. Guenther, "A self-organizing neural network architecture for navigation using optic flow," *Neural Computation*, vol. 10, pp. 313–352, 1998.

[18] J. M. Missler and F. A. Kamangar, "A neural-network for pursuit tracking inspired by the fly visual-system," *Neural Networks*, vol. 8, pp. 463–480, 1995.

[19] I. E. Droer, F. L. Florer, D. Rios, and M. Zagaeski, "Using artificial bat sonar neural networks for complex pattern recognition: Recognizing faces and the speed of a moving target," *Biological Cybernetics*, vol. 74, pp. 331–338, 1996.

[20] B. J. A. K. G. Schram, F. X. van der Linden and F. C. A. Groen, "Visual tracking of moving objects using a neural network controller," *Robotics and Autonomous Systems*, vol. 18, pp. 293–299, 1996.

[21] C. D. Wann and S. C. A. Thomopoulos, "Application of self-organizing neural networks to multiradar data fusion," *Optical Engineering*, vol. 36, pp. 799–813, 1997.

[22] M. Benvenuti, S. Colantonio, M. G. Di Bono, G. Pieri, and O. Salvetti, "Tracking of moving targets in video sequences," *WSEAS Transactions on Systems*, vol. 4, no. 4, pp. 359–364, 2005.

[23] F. Amoozegar and S. H. Sadati, "Target tracking by neural network maneuver detection and input estimation," in *International Radar Conference*. IEEE, 1995, pp. 143–138.

[24] F.-B. Duh and C.-T. Lin, "Tracking a maneuvering target using neural fuzzy network," *IEEE Transaction on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 34, pp. 16–33, 2004.

[25] R. Cozien, C. Rosenberger, P. Eyherabide, J. Rossettini, and A. Ceyrolle, "Target detection and identification using neural networks and multi-agents systems," in *Proceedings of ISIF*, 2000, pp. MoD1–10–Mod1–17.

[26] M. Daneva, "Multitarget tracking in clutter: Two algorithms for data association using bpnn and lvqnn," in *International Conference on Intelligent Systems*. IEEE, 2004, pp. 92–97.

[27] M. Farook and T. K. Roob, "A neural network based data association technique for tracking," in *Proc. IEEE*, 2002, pp. 631–634.

[28] H. Leung and M. Blanchette, "Data association for multiple target tracking using hopfield neural network," in *International Symposium on Speech, Image Processing and Neural Networks*. Hong Kong: IEEE, 1994, pp. 280–283.

[29] S. Mao, P. Lin, and W. YU, "A new insight to the ann-based multitarget tracking data association," in *Proc. IEEE*, 1993, pp. 2423–2426.

[30] N. D. Doulamis, A. D. Doulamis, and K. Ntalianis, "Adaptive classification-based articulation and tracking of video objects employing neural network retraining," in *Proc. DSP*. IEEE, 2002, pp. 575–578.

[31] G. L.-R. no, J. Graial, and R. Díaz-Oliver, "Target detection convolutional in sea clutter using neural networks," in *IEEE Radar Conference*, 2003, pp. 321–328.

[32] N. J. Farsaris, T. D. Xenos, and P. P. Stavroulakis, "Time difference of arrival method of target tracking using a second order recurrent artificial neural network for passive multistatic radar application," *WSEAS Transactions and Communications*, vol. 4, no. 7, pp. 439–448, 2005.

[33] C.-T. Lin and C. S. G. Lee, *Neural Fuzzy Systems : A Neuro-Fuzzy Synergism to Intelligent Systems*. New Jersey: Prentice-Hall, Inc., 1996.

[34] A. F. Shapiro and R. P. Gorman, "Implementing adaptive nonlinear models," *Insurance Mathematics and Economics*, vol. 26, pp. 289–307, 2000.