Nonlinear autoregressive neural network with exogenous inputs based solution for local minimum problem of agent tracking using quad rotor

D. Sayfeddine

Extrapolation is the fact of predicting beyond the observation range. This terminology is used frequently in path planning tasks. This is due to the importance of “early alarm” factor in avoiding surprising changes in an unknown place.

This paper offers an improvement for machine vision based tracking algorithm using NARX neural network. This is achieved by predicting the next position of a mobile agent based on previous movement described by time-series functions. The algorithm remains passive and does not interfere with the flight path until detection of multiple agent-candidates, when the tracking algorithm is not able to identify or locate the assigned agent. Meanwhile the NARX training set are updated along which each successful tracking cycle. This process is done to eliminate the overfitting problem of BPTT training.

This paper is concerned about predicting the next position of a mobile agent tracked by a miniature rotorcraft, a quadrotor. The agent is recognized by its color using HSV algorithm [1]. Although being a straightforward method mathematically, the HSV based machine vision recognition and tracking is directly influenced by the type and nature of light in the operating field. Per instance, a blue car is recognized differently with reference to the time of observation. This is due by the light temperature. Figure.1. shows set of crayons under different color temperature.

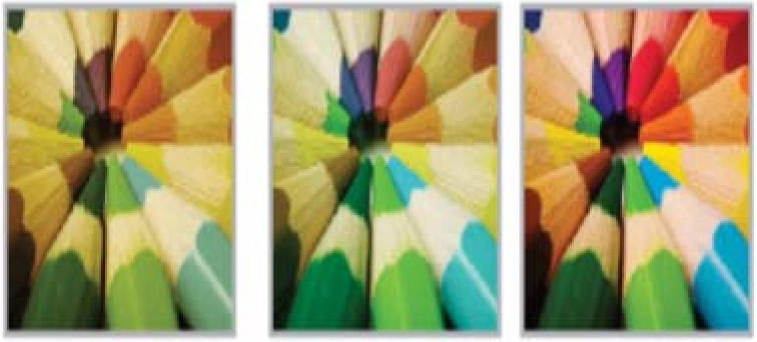

Figure 1. Influence of light temperature on observed object

The change in the color of the observed agent can lead to its unrecognition. The same can happen if in the field of tracking appears another identical object. This fact will cause uncertainty for the tracking algorithm and the task may fail. Another reason that focuses on the importance of position prediction in tracking tasks is the possibility that the agent can disappear for a moment due to environmental circumstances, such as trees, facades, buildings and so on [2].

Description of NARX

As path generator is possibly dealing with dynamic and changing surrounding atmosphere [3], it is of a great importance to design adaptable system able to track, define or detect agents based on the flight mission. Uncertainties as described earlier can be caused from different sources: optical noises, homogeneity of agents, light sources color temperature etc…To overcome these uncertainties and inconvenience of machine vision based tracking algorithms, we suggest to use a nonlinear autoregressive exogenous neural network. NARX is a model of nonlinear neural network, which can accept dynamic inputs represented by time-series sets. This is the main advantage of the NARX over feed forward Back propagation neural networks [2, 4]. NARX can deal also with discrete and continuous inputs [5].

NARX as any recurrent neural network can be trained using one of the following algorithms

- Backpropagation through time or BPTT. It is similar to Backpropagation training held for feed forward neural networks. It is widely used in the literature [4, 6];
- Real-time recurrent learning or RTRL is the straight mathematically but requires computation power and time;
- Algorithms based on extended Kalman filters or EKF. This method has shown success in prediction but it puts constraints on the dynamics of the quadrotor [7].

The aim of the prediction using NARX is to be passive mostly during successful recognition of the agent [8]. During this time NARX can be trained by the updated positions of the mobile agent represented by data pairs (input- Output). This vector is described in (1)

$$u(n) = (u_1(n), \dots u_K(n))', \ d(n) = (d_1(n), \dots d_L(n))' \quad n = 1..T \qquad (1)$$

Where $u(n) -$ is the input set, $d(n) -$ is the output set, *n* represents the time, K is the number of perceptron in the input layer, L is the number of perceptron is the output layer.

The training of the NARX is achieved using BPTT because the output of the network is not fed into the tracking algorithm. As discussed earlier, the training is done passively following three steps.

1- The first step consists of calculating and discovering the status of the activation functions $x(n)$ of each perceptron starting from $u(n)$, $x(n-1)$ and $y(n-1)$ or the activation of the output layer if it is fed into a certain perceptron;

2- The second step includes the calculation of the backpropagation error of each perceptron starting from $n = T..1$, $x(n)$ and $y(n)$ for each instance of time n. this is achieved using the following system of equations (2)

$$\delta_j(T) = (d_j(T) - y_j(T))(\partial f(u))/\partial u|_{u=z_j(T)}$$

$$\delta_i(T) = [\sum_{j=1}^{L} \delta_j(T)\, w_{ji}^{out}](\partial f(u))/\partial u|_{u=z_i(n)}$$

$$\delta_j(n) = [\left(d_j(n) - y_j(n)\right)\sum_{i=1}^{N} \delta_i(n+1) w_{ji}^{back}]\,(\partial f(u))/\partial u|_{u=z_j(n)} \tag{2}$$

$$\delta_i(n) = [\sum_{i=1}^{N} \delta_i(n+1) w_{ji} + \sum_{j=1}^{L} \delta_j(n) w_{ji}^{out}](\partial f(u))/\partial u|_{u=z_i(n)}$$

Where $\delta_j(T) -$ is the backpropagation error of the output perceptron, $\delta_i(T) -$ is the backpropagation error of the perceptron located in the hidden layer with activation $x_i(T)$, $\delta_j(n)$ and $\delta_i(n)$ are consequently the backpropagation error of the output perceptron and the one located in the hidden layer in an earlier time T layer and $z_i(n) -$ is the potential of each perceptron.

3- After finding the backpropagation error, the weights connecting different perceptron are calculated using the following system (3)

$$w_{ij} = w_{ij} + \gamma \sum_{n=1}^{T} \delta_i(n)\, x_j(n-1)$$

$$w_{ij}^{in} = w_{ij}^{in} + \gamma \sum_{n=1}^{T} \delta_i(n)\, u_j(n)$$

$$w_{ij}^{out} = w_{ij}^{out}\gamma \begin{cases} \sum_{n=1}^{T} \delta_i(n)\, u_j(n)\ if\ j - is\ an\ output\ perceptron \\ \sum_{n=1}^{T} \delta_i(n)\, x_j(n)\ if\ j\ is\ a\ hiden\ perceptron \end{cases} \tag{3}$$

$$w_{ij}^{back} = w_{ij}^{back} + +\gamma \sum_{n=1}^{T} \delta_i(n)\, y_j(n-1)$$

Where $w_{ij} -$ is the weight connecting the hidden perceptron, w_{ij}^{in}, w_{ij}^{out} and w_{ij}^{back} are the input, output and feedback weights consequently, $\gamma -$ is an incremental small value that is used during the minimization of the squared error.

Simulation and results

The NARX structure consists of a single time-series data pairs as was described earlier, which contains all the historical data about the movement of the agent. The hidden layer includes 10 perceptrons and a single output is generated thereafter predicting the possible location of the agent. The structure model of the used NARX is shown is figure.2.

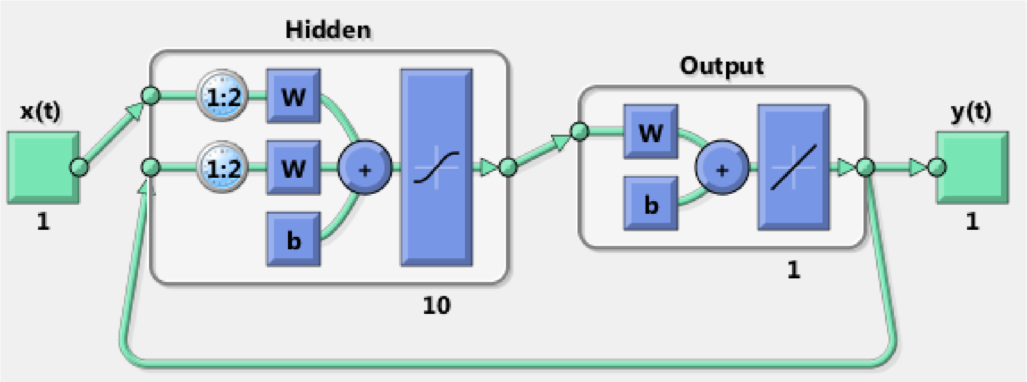

Figure 2. NARX structure model used to predict position

The training is achieved based on 1000 epochs and was concluded in one minute and eight seconds. the result is obtained based on minimal gradient 1^{-10} .

It is important to mention that the NARX acts like recurrent neural network with embedded memory. It allows the NARX to “remember” the output of the perceptrons located in any layer by unfolding the dependencies of the forecasted series far longer than a conventional recurrent neural network. This shows an impact when forecasting nonlinear, aperiodic and unknown data sets.

In light of this, the prediction can be tested on different nonlinear functions conserving the periodicity, the damping factor and chaotic movement of the tracked agent [9].

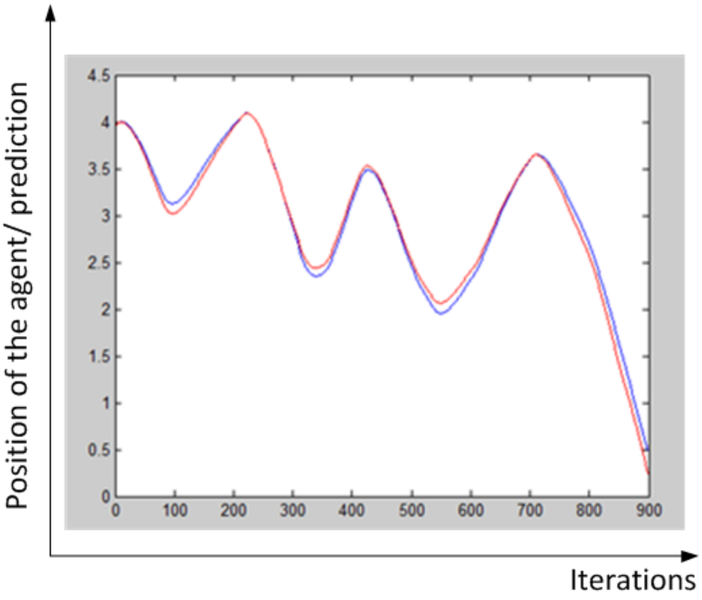

Figure 3. Simulation results

Blue curve- real position of the agent, red curve- predicted position

Simulation results of the predicting the next position of the agent are shown in figure.3. It is clear from the curves that the estimation (red) was very close to the real values presented in the time series (blue).

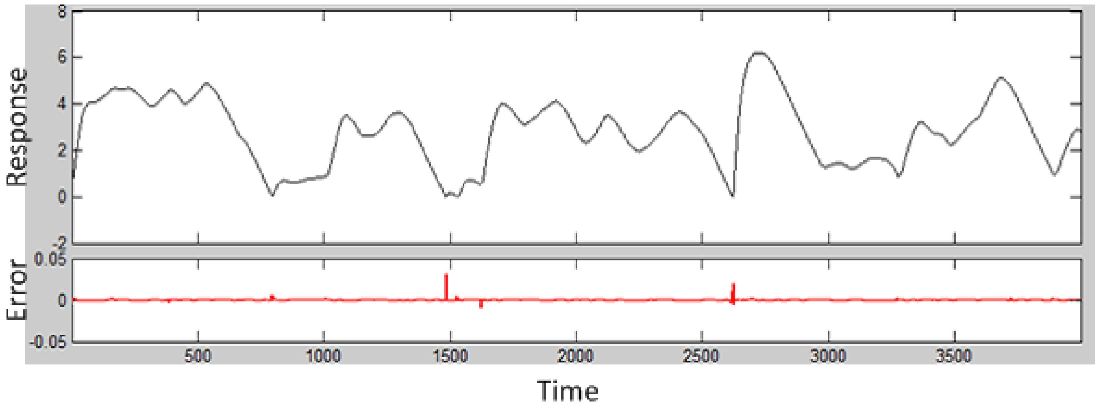

Figure 4. Response of the NARX with reference to the time series (black) and the backpropagation error (red) over time

We can figure out that the NARX was successful in estimating the next position of the dynamic agent, movement of which was described randomly using predesigned time series in Matlab based on magnitude and phase form (input output coupled pairs).

We recommend using the NARX predictor to improve machine vision based path-planning algorithms. The results of this paper are used to guide an autonomous quadrotor facing a local minimum situation [10] performing local planning tasks such as tracking or patrolling tasks.

Literature:

1. Sayfeddine D. Real-time agent tracking based on geometric approach using quadrotor. Material mezhdunarodnoy nauchnoy konferentsii, tom 2. // Severodonetsk, Ukraina, 2013, pp. 109-112. (lang.- Rus.)
2. Sayfeddine D., Bulgakov A.G., Kruglova T.N. Neural Network Position tracking of dynamic agent using quadrotor. [elektronnyy resurs] / Puchkov E.V. // «Inženernyj vestnik Dona (Rus)», 2014, №1 – Rezhim dostupa: http://www.ivdon.ru/magazine/archive/n1y2014/2293 (dostup svobodnyy) – Zagl. s ekrana. – (lang.- Rus.)
3. Tseligorova E.N. Modern Information technologies and their usage to study automatic system . [elektronnyy resurs] / Tseligorova E.N // «Inženernyj vestnik Dona (Rus)», 2010, №3 – Rezhim dostupa: http://ivdon.ru/magazine/archive/n3y2010/222 (dostup svobodnyy) – Zagl. s ekrana. – (lang.- Rus.).
4. Herbert Jaeger. A tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the echo state network approach. Fraunhoffer institute of Autonomous Intelligent Systems (AIS). Fourth Revision. July 2013. pp. 46.

5. Tsungnan Ling, Bill G. Horne, Peter Tino, C. Lee Giles. Learning long-term dependencies in NARX recurrent neural network. IEEE transactions on Neural Networks, vol.7, no.6, 1996, pp. 1329.
6. L.R. Medesker. Neural Networks design and applications. CRC press. 2001. pp. 389.
7. Feldkamp, L. A., Prokhorov, D., Eagen, C.F., and F. Yuan (1998) Enhanced multi-stream Kalman filter training for recurrent networks. In: XXX (ed.), Nonlinear modeling: advanced black-box techniques, 1998, pp. 29-53. Boston: Kluwer.
8. Jaraslaw Bednarz, Tomasz Barszcz, Tadeusz Uhl. NARX model in rotating machinery diagnostics. Mechanics and Control. Vol. 30. No 2, 2011. pp. 41-52.
9. Eugen Diaconescu. The use of NARX Neural Networks to predict Chaotic Time Series. WSEAS TRANSACTIONS on COMPUTER RESEARCH. Issue 3, Volume 3, 2008. pp. 183-191.
10. D. Sayfeddine. T.N. Kruglova. Path Planning algorithm of quadrotor flight using potential fields method. Iskusstvennyy intellekt. Intellektual'nye sistemy II-2013. Materialy Mezhdunarodnoy nauchno-tekhnicheskoy konferentsii Krym, 2013. pp. 206-209. (lang.- Rus.)

Literature:

1. Сайфеддин Д. Геометрический алгоритм отслеживания мобильного агента с помощью квадрокоптера в режиме реального времени полета. Материал международной научной конференции, том 2. // Северодонецк, Украина, 2013, с. 109-112.
2. Д. Сайфеддин, А.Г. Булгаков, Круглова Т.Н. Нейросетевая система отслеживания местоположения динамического агента на базе квадрокоптера. [электронный ресурс] / Пучков Е.В. // «Инженерный вестник Дона», 2014, №1 – Режим доступа:

http://www.ivdon.ru/magazine/archive/n1y2014/2293. (доступ свободный) – Загл. с экрана. – Яз. рус.

3. Целигорова Е.Н. Современные информационные технологии и их использование для исследования систем автоматического управления. [электронный ресурс] / Целигорова Е.Н // «Инженерный вестник Дона», 2010, №3 – Режим доступа: http://ivdon.ru/magazine/archive/n3y2010/222. (доступ свободный) – Загл. с экрана. – Яз. Рус.
4. Herbert Jaeger. A tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the echo state network approach. Fraunhoffer institute of Autonomous Intelligent Systems (AIS). Fourth Revision. July 2013. pp. 46.
5. Tsungnan Ling, Bill G. Horne, Peter Tino, C. Lee Giles. Learning long-term dependencies in NARX recurrent neural network. IEEE transactions on Neural Networks, vol.7, no.6, 1996, pp. 1329.
6. L.R. Medesker. Neural Networks design and applications. CRC press. 2001. pp. 389.
7. Feldkamp, L. A., Prokhorov, D., Eagen, C.F., and F. Yuan (1998) Enhanced multi-stream Kalman filter training for recurrent networks. In: XXX (ed.), Nonlinear modeling: advanced black-box techniques, 1998, pp.29-53. Boston: Kluwer.
8. Jaraslaw Bednarz, Tomasz Barszcz, Tadeusz Uhl. NARX model in rotating machinery diagnostics. Mechanics and Control. Vol. 30. No 2, 2011. pp. 41-52.
9. Eugen Diaconescu. The use of NARX Neural Networks to predict Chaotic Time Series. WSEAS TRANSACTIONS on COMPUTER RESEARCH. Issue 3, Volume 3, 2008. pp. 183-191
10. Д. Сайфеддин, Т.Н. Круглова. Алгоритм планирования траектории перемещения квадрокоптера методом потенциальных полей.

Искусственный интеллект. Интеллектуальные системы ИИ-2013.// Материалы Международной научно-технической конференции Крым, 2013. с. 206-209