International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 5 Issue 7, July-2016

Deciphering Data Fusion Rule by using Adaptive Neuro-Fuzzy Inference System

Ramachandran, A[1]., Raol, J. R[2].

[1]Professor, Dept. of Electronics and Instrumentation Engineering, MSRIT, Bangalore, and Research Scholar, VTU, Belgaum.
[2]Raol, J. R. Professor Emeritus, MSRIT, Bangalore.

ABSTRACT

In this paper a new application of fuzzy logic based adaptive neuro-fuzzy inference system (ANFIS) for deciphering a data fusion rule is presented. Image fusion is carried out and the data fusion rule is deciphered using the ANFIS, like an inverse mathematical modelling. This novel approach is validated using synthetic images generated by simulation in MATLAB. This novel approach would help in: a) analysis of vision-image-based robotic path and motion planning, b) building AI-based robotic systems, and c) analysis and design of AI based data fusion systems.

Keywords: Fuzzy logic, ANFIS, parameter estimation, data/image fusion, data fusion rule.

1. INTRODUCTION

Fuzzy logic (FL) as a soft computing (SC) paradigm differs from the conventional computing because it is tolerant of: a) imprecision, b) uncertainty, c) partial truth, and/or d) approximation, and the role model for such SC is the human mind, since the human mind does not always take hard decisions [1,2], it takes soft and vague decisions. Thus, the principle of FL-SC is to exploit and utilize: i) the tolerance for imprecision (of definitions/expressions of the views), ii) uncertainty in partial decisions/data/information, (available) partial truth, and iii) approximation (of computing new decisions/estimations) to achieve tractability, solvability and robustness of solutions. The FL paradigm in fact models our behavior and activities and also provides computational tool so that one can formally use the technique for solving difficult problems and even develop artificial intelligence (AI) systems that work like humans. Such AI-based systems (hardware, software, algorithms, and/or procedures) can also perform tasks of data/image processing, decision making, sensor data fusion, and parameter estimation. In the present application we utilize the so called ANFIS system that is inherently based on FL, for deciphering data fusion rule that might have been used initially for fusion of two images. We illustrate the successful application of this approach using MATALB based simulations in the context of image fusion. The presented approach is useful in studying various soft computing paradigms for applications to robotics, especially path/motion planning, and vision (images-) based obstacle avoidance.

2. FUZZY LOGIC AND ANFIS

FL is a multi-valued logic used for modeling vagueness (a kind of uncertainty) in data, and it allows to incorporate our acquired knowledge of environment and scenario using ‘If..., Then...’ rules into the fuzzy inference system (FIS) to analyze and design many FL-based control/engineering systems; the FL with its rule-based system and FIS is a powerful tool to design such systems based on AI. These techniques are becoming very important in the development of multi-sensor data fusion strategies: i) because of the inherent structures of ANNs and FL/S (system/sets), Figure 1, ii) their modeling abilities of linear as well as nonlinear dynamic systems, and iii) adaptive learning abilities (from experimental, and empirical data– e.g. adaptively determining the fuzzy rules by using ANNs). For FL, first the rules are defined or constructed; these rules can be learned from experts or can be devised from the data of the system that is under study; a fuzzy rule is ‘If u is A, Then v is B’; the ‘If’ part of the rule ‘If u is A’ is called the antecedent or premise, and the ‘Then’ part of the rule, ‘v is B’ is called the consequent part. The core aspect in the use of FL is FIS that via fuzzy implication functions (FIFs) defines mapping from input FSs into output FSs. In case the antecedent of a given rule has more than one clause, for example, ‘If u_1 is A_1 AND/OR u_2 is A_2, Then v is B’, the suitable fuzzy operators (T-norm/S-norm, respectively) are applied to obtain one value that represents the result of the antecedent for that rule [3].

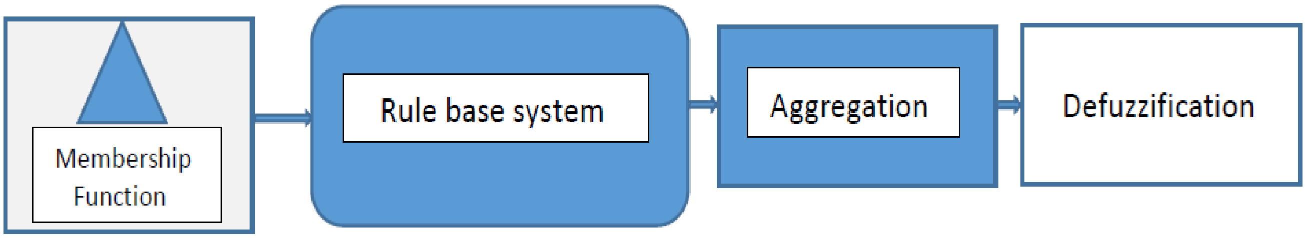

Figure 1 Fuzzy membership function in fuzzy inference system (FIS) ('→' indicates crisp input and crisp output)

ANFIS utilizes a rule based approach to represent the data-behavior in absence of a precise model of the system [4]. Assume, we have a collection of I/O data sets that we have obtained from a system under study, and we want to build a FIS that would approximate these data and use the assigned rule base and output the final result. This type of FIS should consist of some MFs and 'If…Then… rules' with adjustable parameters that define the MF, say triangular one; and these parameters can be chosen so as to adapt the MFs to the input data. Thus, ANFIS uses the I/O data to determine the MFs' parameters/shape, and these parameters are tuned using either a back propagation-steepest descent algorithm or its combination with well-known least squares LS method. The MFs are adaptively tuned by using artificial neural network (ANN) and the I/O data of the given system in an iterative fashion. For ANFIS we consider a rule base as [4]

i) If u_1 is A_1 and u_2 is B_1, then $y_1=c_{11}u_1 + c_{12}u_2 + c_{10}$
ii) If u_1 is A_2 and u_2 is B_2, then $y_2=c_{21}u_1 + c_{22}u_2 + c_{20}$; with u_1, u_2 as non-fuzzy inputs, and y as the desired output.

The ANFIS is layered based and its steps are specified as [4]

i) in layer 1 each ANN-neuron is adaptive with a parametric activation function, its output being the grade of MF to which the given input satisfies the MF → $\mu_{A,}\mu_B$. A generalized MF is used with its parameters as a, b, c to define the magnitude and shape of the MF

$$\mu(u) = \frac{1}{1+\left|\frac{u-c}{a}\right|^{2b}} \tag{1}$$

ii) every node in layer 2 is a fixed node with the output w_1 as the product of all incoming signals

$$w_1 = \mu_{A_i}(u_1)\mu_{B_i}(u_2), \; i=1,2 \tag{2}$$

iii) output of layer 3 for each node is the ratio of the i-th rule's firing strength relative to the sum of all rules' own firing strengths as given by

$$\overline{w}_i = \frac{w_i}{w_1+w_2} \tag{3}$$

iv) every node in layer 4 is an adaptive node with a node output

$$\overline{w}_i y_i = \overline{w}_i(c_{i1}u_1 + c_{i2}u_2 + c_{i0}), \; i=1,2. \tag{4}$$

In (4), 'c' are consequent parameters

v) every node in layer 5 is a fixed node which sums all incoming signals

$$y_p = \overline{w}_1 y_1 + \overline{w}_2 y_2 \tag{5}$$

In (5), y_p is the predicted output. When these premise parameters get fixed, the overall output would be a linear combination of the consequent parameters, and is written as

$$y_p = \overline{w}_1 y_1 + \overline{w}_2 y_2 = \overline{w}_1(c_{11}u_1 + c_{12}u_2 + c_{10}) + \overline{w}_2(c_{21}u_1 + c_{22}u_2 + c_{20})$$
$$= (\overline{w}_1 u_1)c_{11} + (\overline{w}_1 u_2)c_{12} + \overline{w}_1 c_{10} + (\overline{w}_2 u_1)c_{21} + (\overline{w}_2 u_2)c_{22} + \overline{w}_2 c_{20} \quad (6)$$

After the computations by the layers, a hybrid training-cum-estimation algorithm adjusts the consequent parameters in a forward pass and the premise parameters are updated in the backward pass

a) in the forward pass the network inputs propagate forward until layer 4, where the consequent parameters are identified by the LS method
b) in the backward pass, the errors propagate backward (while the time-computation cycle is always a forward phenomenon) and the premise parameters are updated by a gradient descent method.

The ANFIS can be used for parameter estimation and sensor data fusion, and for the present application it is used for parameter-determination-cum-deciphering data fusion rule.

3. PARAMETER ESTIMATION USING ANFIS

First, we study the procedure of parameter estimation using ANFIS: i) generation of initial FIS, and ii) training of FIS. In the step i) the set of training data is generated using 'genfis1' of the MATLAB ANFIS toolbox. The MATLAB-based ANFIS steps are [4]

a) generation of initial FIS by using INITFIS=genfis1(TRNDATA). The 'TRNDATA' is a matrix with N+1 columns where the first N columns contain data for each FIS, and the last column contains the output data - INITFIS is a single output FIS
b) training of FIS: [FIS,ERROR,STEPSIZE, CHKFIS, CHKERROR]=anfis(TRNDATA,INITFIS, TRNOPT, DISPOPT,CHKDATA). Vector TRNOPT specifies training options, vector DISPOPT specifies display options during training, CHKDATA prevents over fitting of the training data set, and CHKFIS is the final tuned FIS.

Then, the tuned FIS is used to predict the system output for new input data of the same class and the parameters are estimated using the so called delta method.

i) after the ANFIS is trained the perturbed data (with say, delta=0.01) are presented to the ANFIS again; while input1 perturbed data are presented, the input 2 data are kept at zero and vice versa for the input2 data
ii) the ratios of the averages of these respective output differences (with respect to the unperturbed data) to the perturbation gives the parameters a and b (assuming a simple algebraic mathematical model).

The ANFIS was tested, first for parameter estimation for an algebraic model by generating the simulated data using the equation

$$y=a+bx_1+cx_2 \quad (7)$$

with the known parameters as: a=1, b=2, and c=1. The simulated parameter were presented to the ANFIS and the delta-rule as discussed in the preceding paragraphs was used to determine the parameters a, b, and c. Very good estimation results were obtained using the ANFIS. Since, it has been tested for parameter estimation/determination, it can now be used for deciphering data fusion rule that might have been used in the image fusion, like an inverse modelling approach.

Deciphering a linear DF rule for images

For this we consider images generated by random numbers as well as real-life like images. We propose some linear fusion rule and fuse the two input images to get an output image. Then, we use these I/O images in the ANFIS and estimate the coefficients/weights of the proposed fusion rule.

Known rule and ANFIS determined rule

The two synthetic input images presented to the ANFIS system are the random images generated using random intensity matrices and are shown in Figure 2. The ANFIS requires the output for training for which we use the algebraic sum of the two input images as y = a * input image1 + b * input image2. The parameters are a =1, and b=0.5. The output image is also shown in Figure 2. These image matrices were converted to column (or row) vectors by concatenating the successive columns and the 2D images were stored as 1D images.

Figure 2 Input images and output image

These input 1D data (of the two random images) were given to the ANFIS system, likewise the training output image was also converted to 1D string of the intensity values. After the ANFIS was trained, the perturbed data (with delta=0.01) were presented to the ANFIS again; with this we obtain the perturbed output data, and the ratios of the averages of these respective differences (with respect to the unperturbed data) to the perturbation size gives the parameters a and b for respective input image1 and the input image 2. Entire exercise is done with 1D image strings. We saw that the estimation results were very good and we obtained the parameters as: a=1, and b=0.5. We see that ANFIS is effectively utilized to determine the image fusion rule.

Unknown rule and ANFIS determined rule

For this case, we consider an example of ANFIS system to decipher an unknown data fusion rule in the context of two random-data synthetic images. The two input images given to the ANFIS system are the random images generated using random intensity matrices as follows for the input images

image 1
rand('seed',1234);
im1=rand(128,128);

image 2
rand('seed',4321);
im2= rand (128,128);

the **output image**
randn('seed',2468);
imo=randn(128,128).

This means that the input random images are generated by uniformly distributed random numbers (intensities) while the output image is generated by using the normally distributed random numbers (intensities). At this stage we do not know the DF rule between the output and the two input images. When the ANFIS is run with these I/O images, the stabilization quickly occurs. This means that the ANFIS has determined the coefficients a, and b consistently. So, we choose a=0.7 and b=0.36 as the DF rule coefficients and determined the output image sequence using the formula: $y=0.7*x_1+0.36*x_2$. We ran the ANFIS once again to estimate these coefficients, and we got the exact results. The I/O images are shown in **Figure 3.** We have thus established the efficacy of the ANFIS system to really determine the coefficients of an unknown DF rule, which was subsequently verified by using the ANFIS. For the complex rules one needs to make further studies.

Figure 3 Images used for determining the unknown DF rule using ANFIS

Determination of DF rule using ANFIS and real blurred images

The two input images given to the ANFIS system are the images generated using: i) one real-true image, and ii) then partially blurring the same true image with 'imfilter' (Image processing-MATLAB tool box). The ANFIS requires the output for training for which we use the algebraic sum of the two input images as y = a * input image1 + b * input image2 for image fusion; the parameters being a =0.4, and b=0.6. These input 1D data (of the two random images) were given to the ANFIS system. After the ANFIS was trained the perturbed data (with delta=5 and 7) were presented to the ANFIS again (while input1 perturbed data were presented, the input2 data were kept at zero and vice versa for the input2 data). With this, we obtain the perturbed output data. We also obtained the ratios of the averages of the respective differences (with respect to the unperturbed data) to the perturbation delta. This gives the parameters 'a' and 'b' for respective input image 1 and the input image 2. We see that the estimation results are good and we obtained the estimated parameters as: a=0.4033, and b=0.5850. We see that ANFIS is effectively utilized to determine the image fusion rule for the real images, where in one input images was partially blurred. Various images as input/output of this ANFIS exercise are shown in **Figure 4.**

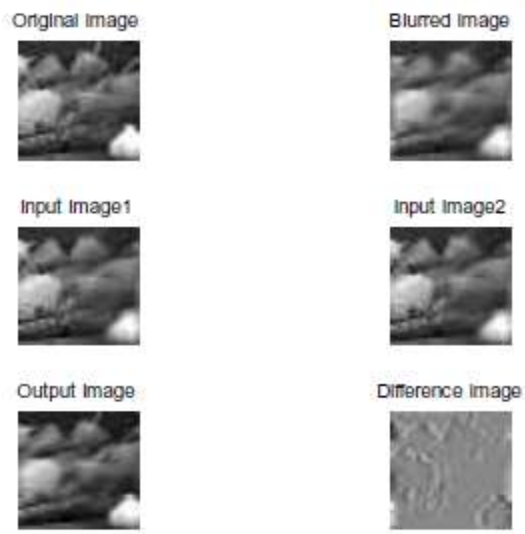

Figure 4 Blurred images (input to ANFIS/output image)

4. CONCLUDING REMARKS

Fuzzy logic based ANFIS has been successfully validated for: i) basic parameter estimation for a linear algebraic model, ii) determination of the data fusion rule when such a fusion rule is known, iii) the determination of data fusion rule when such a fusion rule is unknown, and finally iv) determination of the data fusion rule for a real image-fusion (with one input-image as a blurred image). In all the cases, it has been found that the ANFIS worked very well for determination of data fusion rules when the fused images were presented to the ANFIS. This approach is in the direction of inverse (mathematical) modeling and can be utilized for deciphering the inherent data fusion mechanism that might be operating when we receive some fused real images, and we would like to see how these component images might have been combined. This helps in: i) analysis of vision-image-based robotic path/motion planning, ii) building AI assisted robotic systems, and iii) analysis and design of AI based data fusion systems.

ACKNOWLEDGEMENT

The first author is grateful to his guide, Dr. D. Sheshachalam, Professor and Head, Dept. of E&CE, BMS College of Engineering, Bangalore, for encouraging the research in this area.

REFERENCES

[1]. Zadeh, L. A. A definition of soft computing. http://www.soft-computing.de/def.html, accessed April 2014.
[2]. Burkey, M., Paul, Pan, W., Kou, X., Marler, K. M., and Tsaptsinos, D. Soft Computing- Fuzzy logic is a part of soft computing (ppts). Dept. of Mathematics, Kingston University, http://www.kingston. ac.uk/~mas435@kingston.ac.uk. Also, www.ee.pdx.edu/~mperkows/class479/lectures479/fl002.pdf. Accessed June 2014.
[3]. Robert E. King., Computational Intelligence in Control Engineering, Marcel Dekker, INC., New York, USA, 1999.
[4]. Anon. Fuzzy logic tool box, ANFIS demo, MATLAB ®.