

Improve Robots Calibration Accuracy Using A Dynamic Online Interval Type-2 Fuzzy Interpolation System

Ying Bai

Dept of Computer Science and Engineering
Johnson C. Smith University
Charlotte, USA
ybai@jcsu.edu

Dali Wang

Dept of Physics and Computer Science
Christopher Newport University
Newport News, USA
dwang@pcs.cnu.edu

Abstract—This paper describes a novel technique for the position error compensations of the robot and manipulator calibration process based on an Interval Type-2 Fuzzy error interpolation (IT2FEI) method. Traditional robots calibration implements either model or modelless method. The compensation of position error in modelless method is to move the robot's end-effector to a target position in the robot workspace, and to find the target position error based on the measured neighboring 4-point errors around the target position. A camera or other measurement device is attached on the robot's end-effector to find and measure the neighboring position errors, and compensate the target position with the error interpolation results. By using the IT2FEI technique provided in this paper, the accuracy of the position error compensation can be greatly improved, which has been confirmed by the simulation results given in this paper. Compared with some other popular traditional interpolation methods, this IT2FEI technique is a better choice. The simulation results show that more accurate compensation result can be achieved using this technique compared with the type-1 fuzzy interpolation method.

Keywords—*Modelless robots calibrations; position error compensations; interval type-2 fuzzy interpolations; dynamic online fuzzy interpolation algorithm.*

I. INTRODUCTION

The prerequisite requirement of the robotic modelless calibration is the successful self-calibration of the camera [1, 2]. Both internal and external parameters of the camera need to be calibrated accurately [3, 4]. Then modelless robot calibration is divided into two steps [5]. The first step is to identify the position errors for all grid points on a standard calibration board, which is installed on the robot's workspace. A calibrated camera or other measurement device is attached on the robot's end-effector to find the neighboring position errors. This process can be considered as an identification process, which is shown in Fig. 1.

At each grid point, a calibrated camera is used to check the position errors of the end-effector of the robot. In Fig 1, the desired position of the grid point 0 is (x_0, y_0) , and the actual position of the robot end-effector is (x'_0, y'_0) . The position errors for this grid point are $e_x = x_0 - x'_0$, and $e_y = y_0 - y'_0$.

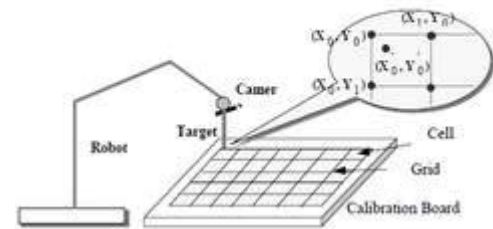


Fig.1 Setup of the robotic modelless calibration.

The robot will be moved to all grid points on the standard calibration board, and all position errors on these grid points will be measured and stored in the memory for future use.

In the second step, the robot's end-effector is moved to a target position that is located in the range of the workspace. The target position error could be found by an interpolation technique using the stored 4-neighboring grid position errors around the target position, which were obtained from the first step. Finally, the target position could be compensated with the interpolation results to obtain more accurate positions.

J. Triantafilis and D. Suzana et al reported approaches of using fuzzy interpolation methods to estimate the soil layer and geographical distributions for GIS database [6, 7]. F. Song et al described a fuzzy logic methodology for 4-dimensional (4D) systems with optimal global performance using enhanced cell state space [8]. The most popular interpolation techniques applied in the position compensations of the modelless robotic calibration include the bilinear interpolation and cubic spline interpolation methods; both methods can achieve satisfactory interpolation results for general calibration process [2]. Since the actual position errors are randomly distributed, and it is impossible to pinpoint a position on the error surface at any given moment, the traditional interpolation techniques are unable to provide an accurate estimation of the position errors. The IT2FEI technique utilizes an interval type-2 fuzzy inference system to estimate the position errors, which is consistent with the random distributed nature of position errors. The position errors can be considered as a fuzzy set at any given moment of the time. The fuzzification process takes into account of a range of error rather only a crisp error value. Therefore, the fuzzy error interpolation technique has the fundamentals to improve error estimation results.

The Interval Type-2 Fuzzy inference is the process of formulating the mapping from a given input to an output using interval type-2 fuzzy logic [9-10]. The mapping then provides a basis from which decisions can be made, or patterns discerned. Interval Type-2 Fuzzy inference systems have been successfully applied in fields such as automatic controls, data classification, decision analysis, expert systems, and computer vision [9]. S. Aminifar and A. Marzuki reported an analysis about the uncertainty between the type-1 and Interval Type-2 Fuzzy Logic System (IT2 FLS) [11]. Yu-Chuan Chang presented a new method for handling fuzzy rule interpolation in sparse fuzzy rule-based systems based on interval type-2 fuzzy sets [12]. Kashyap and Sudesh Kumar reported a research of using an interval type-2 fuzzy logic system to perform image fusion [13]. Qing Lu et al. reported to control a nonlinear system with the interval type-2 FLC [14]. Dongrui Wu and Mendel, J.M reported a new method to simplify the design process for the interval type-2 fuzzy system [15]. Schriber, M.D. and Biglarbegian, M reported to use an interval type-2 FLC to control the FPGA production process [16]. Ching-Chih Tsai et al. reported to use an interval type-2 FLC to control an intelligent bike [17]. Nurmaini, S. and Tutuko, B. described a motion coordination of swarm robots using Interval Type-2 Fuzzy Logic Controller (IT2FLC) to control swarm robots coordination to produce smooth trajectory without collision [18]. Mendel, J. et al. discussed and analyzed different structures of type-1 and type-2 fuzzy controllers [19]. Kumbasar, T examined the robust stability of a PD type Single input Interval Type-2 Fuzzy Logic Controller (SIT2-FLC) structure [20]. All of these updated applications provide a prospective future for interval type-2 fuzzy inference system.

This paper is organized in 5 sections. After this introduction section, the operational principles of the type-1 fuzzy interpolation technique are provided in section 2. Section 3 discusses the interval type-2 fuzzy error interpolation method. A simulation is given in section 4 to illustrate the effectiveness of the fuzzy error interpolation technique. Section 5 presents the conclusion.

II. TYPE-1 FUZZY ERROR INTERPOLATION SYSTEM

A. The on-line versus off-line fuzzy system

In order to improve the compensation accuracy, a dynamic on-line fuzzy error interpolation method is introduced. The traditional fuzzy inference system uses pre-defined membership functions and control rules to a construct lookup tables; then a control output is selected from the lookup table. This type of system is called off-line fuzzy inference system because all inputs and outputs have been defined prior to the application process. The off-line fuzzy system cannot meet our requirement for the several reasons. First, the position error of the target point is estimated based on 4 errors of the neighboring grid points, and these 4 neighboring errors are randomly distributed. The off-line fuzzy output membership functions are defined based on the range of errors, which is the neighboring errors' range. However, this range estimation is

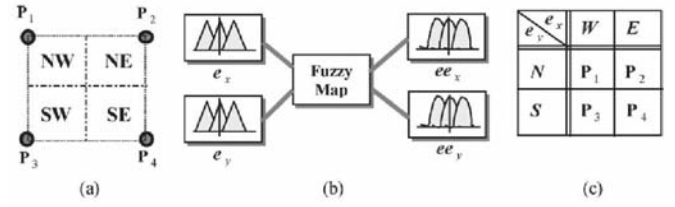


Fig. 2. Definition of the Fuzzy Error Interpolation System.

not as accurate as the real errors obtained on the grid points. Second, since each cell needs one lookup table for the off-line fuzzy system, it would require a huge memory space to save a large number of lookup tables. This results in demanding requirement in both space and time, and as a result, becomes not practical for real time processing. For example, in our study, 20×20 cells are utilized on the calibration board (each cell is 20×20 mm); this would require 400 lookup tables. By using an on-line dynamic fuzzy inference system, the target position error can be estimated by combining the output membership functions, which are defined based on the real errors on neighboring grid points and the control rules in real time. The output membership functions are not predetermined, and their definitions are based on the real errors on the grid points, not a range.

Fig. 2 shows the definition of the fuzzy error interpolation inference system.

Each square that is defined by 4 grid points is called a cell; and each cell is divided into 4 equal smaller cells, which are NW, NE, SW and SE, respectively (Fig. 2 (a)). The position error at each grid point is defined as P_1 , P_2 , P_3 and P_4 .

For the fuzzy inference system, we apply the fuzzy error interpolation method in two dimensions separately, so the inputs to the fuzzy inference system are e_x and e_y and the outputs are ee_x and ee_y (Fig. 2 (b)). The control rules are shown in Fig. 2 (c), and are discussed following the discussion of membership functions.

B. Membership functions

In this work, the distance between two neighboring grid points on the standard calibration board is 20 mm in both x and y directions, which is a standard value for a mid-size calibration workspace. The calibration board includes a total of 20 by 20 cells, which is equivalent to a 400 by 400 mm space.

The input membership functions for both x and y directions and the predefined output membership functions are shown in Fig. 3. The predefined output membership functions are used as default functions, and the final output membership function will be obtained by shifting the default one by the actual error values on the grid points.

The gaussian-bell waveforms are selected as the shape of the membership functions for both inputs (Fig. 3 (a)) in x and y directions. The ranges of inputs are between -10 mm and 10 mm (20 mm intervals). H. Zhuang and X. Wu reported a special

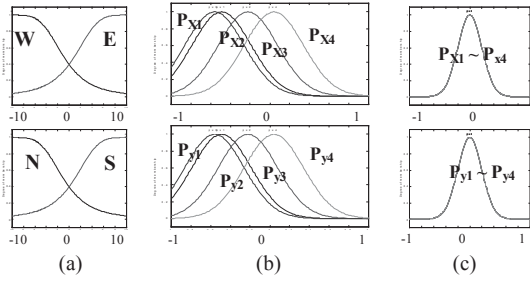


Fig. 3. Input and output membership functions.

histogram method to estimate the optimal membership function distribution [21]. However in our case, a gaussian-bell shape is selected due to the fact that most errors in real world match this distribution. We use W and E to represent the location of inputs in x direction, N and S to represent the location of inputs in y direction. Unlike the traditional fuzzy inference system, in which all membership functions should be determined in advance to produce the lookup table, the output membership functions will be determined during the application of the fuzzy inference system in real time. This is called dynamic fuzzy system.

Fig. 3 (b) shows an example of the output membership functions, which are related to the simulated random errors at neighboring grid points. Each P_{xi} and P_{yi} correspond to the position error at the i th grid point in x and y directions, respectively. During the design stage, all output membership functions are initialized to a gaussian waveform with a mean of 0 and a range between -0.5 and 0.5 mm, which is a typical error range for this workspace in robotic calibration (Fig. 3 (c)) [5]. These output membership functions will be determined based on the errors of the neighboring grid points around the target in the workspace as mentioned above. For example, during the compensation process if the input position in the x direction is in the NW area of a cell, the associated output membership function should be modified based on the position error in the NW grid point P_1 . This modification is equivalent to shift the P_{x1} Gaussian waveform (Fig. 3 (b)) and allow the center of that waveform to be located at x_0 = the position error value of the P_1 in the x direction. Similar modification should be performed for the position error in the y direction. It can be seen from Fig. 3 (b) that the performance loss would be significant if the default output membership function is utilized, which is shown in Fig. 3 (c), for the position compensation process.

C. Membership functions

The control rules shown in Fig. 2 (c) can be interpreted as follows after the output membership functions are determined:

- If e_x is W and e_y is N, ee_x is P_{x1} and ee_y is P_{y1} (NW).
- If e_x is W and e_y is S, ee_x is P_{x3} and ee_y is P_{y3} (SW).
- If e_x is E and e_y is N, ee_x is P_{x2} and ee_y is P_{y2} (NE).
- If e_x is E and e_y is S, ee_x is P_{x4} and ee_y is P_{y4} (SE).

Each P_i should be considered as a combination of two error components, P_{xi} and P_{yi} , which are corresponding to errors in both x and y directions. The error on NW grid point should take more weight if the target position (input) is located inside the NW area on a cell. Similar conclusion can be derived for errors on SW, NE and SE grid points.

III. INTERVAL TYPE 2 FUZZY INTERPOLATION SYSTEM

A. Overview of the interval type 2 fuzzy interpolation system

Similar to type 1 fuzzy inference system, the type 2 fuzzy inference system still uses the input and output membership functions, combined with the control rules, to derive the outputs [10, 15]. However, the fuzzy sets used in the type 2 fuzzy logic or the membership grades involved in each membership function are not crisp values, but another fuzzy sets. This means that the membership degrees for all membership functions used in the type 1 fuzzy system are fixed values and can be determined uniquely before the fuzzy inference system works. But the membership degrees for all membership functions used in the type 2 fuzzy system are fuzzy sets. The difference between the standard type-2 fuzzy system and the so-called interval type-2 fuzzy system is that in the former system, the membership degrees are pure fuzzy sets, but the membership degrees are a set of crisp values with a range of $0 \sim 1$ or an interval for the latter.

Fig. 4 shows the functional block diagram of an Interval Type-2 FLS [22]. It is similar to Typ-1 FLS, but the major difference is that at least one of the fuzzy sets in the rule base is an IT2 fuzzy set. The outputs of the inference engine are IT2 fuzzy sets, and a type-reducer is needed to convert them into a Typ-1 fuzzy set before defuzzification can be started.

Some fundamental operations in the type-2 fuzzy system are union (3.1), intersection (3.2) and complement (3.3) [23].

The union for interval type-2 fuzzy sets \tilde{A} and \tilde{B} is:

$$\tilde{A} \sqcup \tilde{B} = \left\{ \int_{x \in X} \mu_{\tilde{A}}(x) \sqcup \mu_{\tilde{B}}(x) / x \right\} = \left\{ \int_{x \in X} \left[\alpha \in [\mu_{\tilde{A}}(x) \vee \mu_{\tilde{B}}(x), \bar{\mu}_{\tilde{A}}(x) \vee \bar{\mu}_{\tilde{B}}(x)] \right] 1/\alpha / x \right\} \quad (3.1)$$

The intersection for interval type-2 fuzzy sets \tilde{A} and \tilde{B} is:

$$\tilde{A} \cap \tilde{B} = \left\{ \int_{x \in X} \mu_{\tilde{A}}(x) \cap \mu_{\tilde{B}}(x) / x \right\} = \left\{ \int_{x \in X} \left[\alpha \in [\mu_{\tilde{A}}(x) \wedge \mu_{\tilde{B}}(x), \bar{\mu}_{\tilde{A}}(x) \wedge \bar{\mu}_{\tilde{B}}(x)] \right] 1/\alpha / x \right\} \quad (3.2)$$

The complement for interval type-2 fuzzy sets \tilde{A} and \tilde{B} is:

$$\neg \tilde{A} = \left\{ \int_{x \in X} \mu_{\neg \tilde{A}}(x) / x \right\} = \left\{ \int_{x \in X} \left[\alpha \in [1 - \bar{\mu}_{\tilde{A}}(x), 1 - \mu_{\tilde{A}}(x)] \right] 1/\alpha / x \right\} \quad (3.3)$$

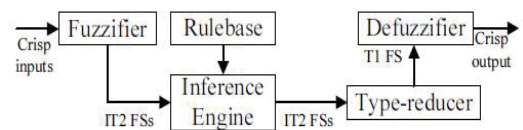


Fig. 4. A functional block diagram of the Interval Type-2 Fuzzy system.

In practice the computations in an IT2 FLS can be significantly simplified. Consider the rule base of an IT2 FLS consisting of N rules assuming the following form [22]:

R^n : if x_1 is X_1^n and ... and x_i is X_i^n , then y is Y^n ; $n=1,2,\dots,N$ where X_i^n ($i=1 \sim I$) are IT2 Fuzzy sets, and $Y^n=[y_1^n, y_2^n]$ is an interval, which can be understood as the centroid [9, 24] of a consequent Interval Type-2 fuzzy set, or the simplest TSK model, for its simplicity. In many applications we use $y_1^n = y_2^n$, i.e., each rule consequent is a crisp number.

Assume the input vector is $x' = (x'_1, x'_2, \dots, x'_I)$. Typical computations in an IT2 FLS involve the following steps:

- 1) Compute the membership of x'_i on each X_i^n
- 2) Compute the firing interval of the n^{th} rule, $F^n(x')$
- 3) Perform type-reduction to combine $F^n(x')$ and the corresponding rule consequents with the center-of-sets type-reducer [9]:

$$Y_{\text{cos}}(x') = \bigcup_{\substack{f^n \in F^n(x') \\ y^n \in Y^n}} \frac{\sum_{n=1}^N f^n y^n}{\sum_{n=1}^N f^n} = [y_l, y_r] \quad (3.4)$$

- 4) Compute the defuzzified output as:

$$y = \frac{y_l + y_r}{2} \quad (3.5)$$

B. Membership Functions

Similar to type-1 fuzzy interpolation system, the input membership functions for both x and y directions and the predefined output membership functions for IT2 FLS are shown in Fig. 5. The predefined output membership functions are used as default functions, and the final output membership function will be obtained by shifting the default those by the actual error values on the grid points.

We use W and E to represent the location of inputs in x direction, N and S to the location of inputs in y direction. For real outputs, 4 membership functions, $p_{x1} \sim p_{x4}$, should be designed for the x direction, and another 4 membership functions, $p_{y1} \sim p_{y4}$, are to be built for the y direction. These output functions should be located at the center position, which are defined as the default location, as the beginning and changed to the real location based on the actual position errors on each grid point. In Fig. 5, these functions are all displayed but not in the default locations.

As for the control rules, the identical control rules are used for this IT2 FLS, but the fuzzy sets are used as the degrees to replace those crisp values used in the type-1 FLS.

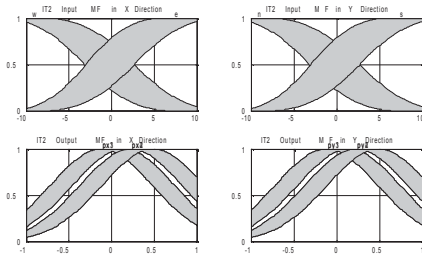


Fig. 5. The input and output membership functions for IT2 FLS.

IV. SIMULATION RESULT

Extensive simulation has been performed in order to illustrate the effectiveness of the proposed dynamic online IT2 fuzzy error interpolation technique in comparison to the type-1 FLS. Due to the random nature of the position errors, three different types of error are simulated in this study. These are:

- Normal distributed random error
- Uniform distributed random error
- Sinusoidal waveform error

Figs 6, 7 and 8 show the simulation results of the type-1 and the IT2 fuzzy error interpolation techniques for these three types of error [23].

In these figures, the simulated target (testing) positions on the standard calibration board are spaced from 1 mm to 20 mm within each cell being with a size of 1 mm.

Figs 9 to 11 show comparisons in mean error, maximum error and STD values between type-1 and IT2 fuzzy error interpolation techniques in the histograms.

It can be seen that both mean errors and maximum errors of the IT2 fuzzy error interpolation technique are smaller than

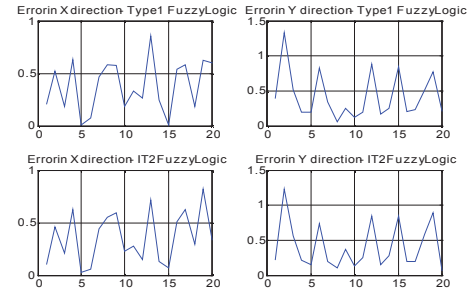


Fig 6. Normal Distributed Random Noise.

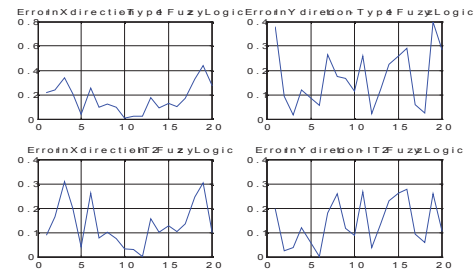


Fig 7. Uniform Distributed Noise.

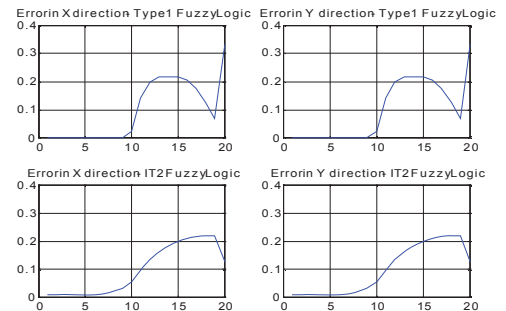


Fig 8. Sinusoidal Waveform Noise.

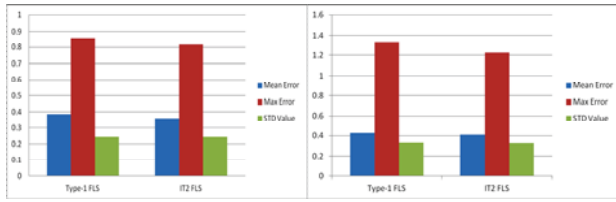


Fig 9. Normal distributed random errors in X and Y directions.

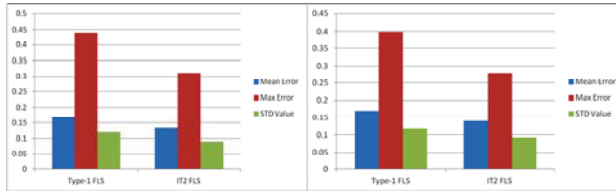


Fig 10. Uniform distributed random errors in X and Y directions.

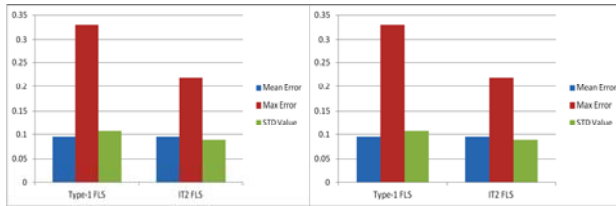


Fig 11. Sinusoid distributed random errors in X and Y directions.

those of type-1 FLS methods. For all three error distribution, the mean errors of the IT2 fuzzy error interpolation method are approximately 10% to 20% smaller compared with those of type-1 FLS method.

The maximum errors of the IT2 fuzzy error interpolation technique are about 10% to 30% smaller than those of the type-1 FLS method. In one case (normal distribution error in x and y direction), the maximum errors of the IT2 fuzzy interpolation method are about 4% smaller than those of the type-1 FLS method.

Figure 12 shows the IT2 fuzzy interpolation error surface.

The simulated results show the effectiveness of the dynamic on-line interval type-2 fuzzy error interpolation technique in reducing the position errors in the modelless robot compensation process.

To implement this interval type-2 fuzzy error interpolation technique as a real time application, an interface between the MATLAB[®] and high level programming languages C/C++ has been developed [25]. Although the most popular real time

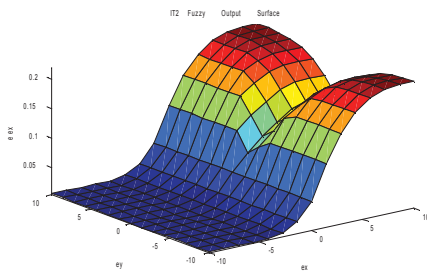


Fig 12. The IT2 fuzzy interpolation error surface.

control programming language is C/C++, the fuzzy error interpolation method is developed and implemented in MATLAB [23]. Using this interface, the measured position errors on the grid points can be passed from C/C++ to MATLAB functions that implements the interval type-2 fuzzy error interpolation functions; and the fuzzy error interpolation results can be sent back to C/C++ for the real time controller to operate on the next target position.

V. CONCLUSION AND SUMMARY

A dynamic on-line interval type-2 fuzzy error interpolation technique is presented in this paper. The compensated position errors in a modelless robot calibration can be greatly reduced by the proposed technique. Simulation results demonstrate the effectiveness of the proposed fuzzy error interpolation technique. Three typical error models are utilized for comparison and simulation; these include sinusoidal waveform, normally distributed and uniformly distributed errors. This fuzzy error interpolation technique is ideal for the modelless robot position compensation, especially the high accuracy robot calibration process.

ACKNOWLEDGMENT

Special thanks to Dr. Oscar Castillo and his group to provide me the Interval Type-2 Fuzzy Logic Toolbox to enable us to perform our simulations for this paper [26-27]. We could not develop and share this paper with all of you without his sincerely help and support.

REFERENCES

- [1] B. W. Mooring, Z. S. Roth and M. R. Driels, *Fundamentals of Manipulator Calibration*, John Wiley & Sons, Inc. 1991.
- [2] M. A. Penna, "Camera Calibration: A Quick and Easy Way to Determine the Scale Factor", *IEEE Trans. Pattern Anal. Machine Intell.* 13, 1991, pp. 1240-1245.
- [3] J. Weng et al, "Camera Calibration with Distortion Models and Accuracy Evaluation", *IEEE Trans, Pattern Anal. Machine Intell.* 14, 1992, pp. 965-980.
- [4] F. Du and M. Brady, "Self Calibration of the Intrinsic Parameters of Cameras for Active Vision Systems", In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, New York, NY June, 1993, pp. 477-482..
- [5] Hanqi Zhuang and Z. S. Roth, *Camera-Aided Robot Calibration*, CRC Press, Inc. 1996.
- [6] J. Triantafilis, W. T. Ward, I.O.A Odeh and A.B. McBratney, "Creation and Interpolation of Continuous Soil Layer Classes in the Lower Namoi Valley", *Soil Science Society of America Journal*, Mar/Apr 2001, Vol. 65, Issue 2, pp. 403-414.
- [7] D. Suzana, J.D. Marceau and M. Claude, "Space, time, and dynamics modeling in historical GIS databases: A Fuzzy Logic Approach", *Environment & Planning B: Planning & Design*, Jul 2001, Vol. 28 Issue 4, pp. 545-563.
- [8] F. Song, S. M. Smith, and C. G. Rizk, "A Fuzzy Logic Controller Design Methodology for 4D System with Optimal Global Performance Using Enhanced Cell State Space Based Best Estimate Directed Search Method", *IEEE Int. Conf. On Systems, Man and Cybernetics*, 1999.
- [9] Jerry M. Mendel. *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Prentice Hall PTR, 2001.

- [10] Oscar Castillo and Patricia Melin, Type-2 Fuzzy Logic: Theory and Applications, Studies in Fuzziness and Software Computing, Vol. 223, Springer, 2008.
- [11] S. Aminifar and A. Marzuki, "Uncertainty in Interval Type-2 Fuzzy Systems, Mathematical Problems in Engineering", Volume 2013 (2013), Article ID 452780, 16 pages.
- [12] Yu-Chuan Chang, "A new fuzzy interpolative reasoning method based on interval type-2 fuzzy sets", IEEE International Conference on Systems, Man and Cybernetics, 2008. SMC 2008, Singapore, Oct. 12-15, 2008, pp.82-87.
- [13] Kashyap and Sudesh Kumar, "IR and color image fusion using interval type 2 fuzzy logic system", 2015 International Conference on Cognitive Computing and Information Processing (CCIP), Noida, India, 3-4 March 2015, pp.1-4.
- [14] Qing Lu et al, "Interval Type-2 Fuzzy Model Predictive Control of Nonlinear Networked Control Systems", IEEE Transactions on Fuzzy Systems, Vol. PP, Issue 99, March 2015, pp. 1-8.
- [15] Dongrui Wu and Mendel, J.M, "Designing practical interval type-2 fuzzy logic systems made simple", 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Beijing, 6-11 July 2014, pp. 800-807.
- [16] Schrieber, M.D. and Biglarbegian, "Hardware implementation of a novel inference engine for interval type-2 fuzzy control on FPGA", 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 6-11 July 2014, Beijing, 6-11 July 2014, pp. 640-646.
- [17] Ching-Chih Tsai et al., "Interval type-2 fuzzy gear-changing control for intelligent bikes", 2014 International Conference on Machine Learning and Cybernetics (ICMLC), 13-16 July 2014, Lanzhou, pp. 741-747.
- [18] Nurmaini, S. and Tutuko, B., "Motion coordination for swarm robots", 2014 International Conference on ICT for Smart Society (ICISS), 24-25 Sept. 2014, Bandung, pp. 312-315.
- [19] Mendel, J. et al., Book Chapter, Analytical Structure of Various Interval Type-2 Fuzzy PI and PD Controllers, Wiley-IEEE Press, 2014.
- [20] Kumbasar, T., "Robust stability analysis of PD type single input interval type-2 fuzzy control systems", 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 6-11 July 2014, Beijing, pp. 634-639.
- [21] H. Zhuang and Xiaomin Wu, "Membership Function Modification of Fuzzy Logic Controllers with Histogram Equalization", IEEE Trans. On System, Man, and Cybernetics – Part B: Cybernetics, Vol. 31, No. 1, Feb. 2001.
- [22] Dongrui Wu, "A Tutorial on Interval Type-2 Fuzzy Sets and Systems", University of Southern California, USA2012.
- [23] ITT/UABC, Interval Type-2 Fuzzy Logic Toolbox For Use with MATLAB®, Tijuana Institute of Technology and Baja California Autonomous, University, Tijuana Campus, Mexico, 2008.
- [24] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," Information Sciences, vol. 132, pp. 195–220, 2001.
- [25] Ying Bai, Applications Interface Programming Using Multiple Languages, Prentice Hall, March 2003.
- [26] J.R. Castro, O. Castillo, P. Melin, L.G. Martínez, S. Escobar, I. Camacho. "Building Fuzzy Inference Systems with the Interval Type-2 Fuzzy Logic Toolbox", IFSA, 2007.
- [27] Juan R. Castro, Oscar Castillo, Patricia Melin. "An Interval Type-2 Fuzzy Logic Toolbox for Control Applications", FUZZ-IEEE, 2007.