High-speed Fuzzy Inference System in Integrated GUI Environment

Sang Gu Lee*

***Department of Computer Engineering, Hannam University**
133 Ojung-Dong, Daeduk-Gu, Daejeon 306-791, Korea

Abstract

We propose an intgrated GUI environment system having only integer fuzzy operations in the consequent part and the defuzzification stage. In this paper, we also propose an integrated GUI environment system with 4 parallel fuzzy processing units to be operated in parallel on the classification of the sensed image data. In this, we solve the problems of taking longer times as the fuzzy real computations of [0, 1] by using the integer pixel conversion algorithm to convert lines of each fuzzy linguistic term to the closest integer pixels. This procedure is performed automatically in the GUI application program. As a GUI environment, PCI transmission, image data pre-processing, integer pixel mapping and fuzzy membership tuning are considered. This system can be operated in parallel manner for MIMO or MISO systems.

Key Words : GUI environment, high-speed fuzzy inference, integer pixel mapping and parallel inference

1. Introduction

Since the fuzzy theory was proposed by Zadeh in 1965, and the fuzzy control logic was represented by Mamdani in 1974, the fuzzy systems have attracted a great deal of worldwide attention. Fuzzy logic models are successfully being used in many engineering applications involving the fuzzy control, data mining, expert system, pattern recognition, pattern clustering, prediction and medical diagnosis.[1] The pattern classification of multi-spectral image data obtained from satellites or aircrafts has become an important tool for generating ground cover maps.[2] In this paper, we use fuzzy theory for pattern classification of the remote sensing images with high-speed fuzzy inference. The digital images for experimentation are acquired with the airborne multi-spectral scanner sensor. The test images consist of 388 lines, with 388 pixels per line, one pixel size of about 3 * 3 meters, and the 3 visible and 1 near-infrared bands among the 6 bands. Using these 4-band images, the pattern classification is performed by applying the fuzzy inference to each pixel and the entire fuzzy rules. However, There needs very large fuzzy computation because of many fuzzy floating-point operations. To reduce the problem of the real valued operation in [0, 1], in this paper, we use an integer pixel conversion algorithm. In this, we select and plot the only integer pixel from 0 to 31 for fuzzy membership grades. Therefore, a fuzzy membership degree of [0, 1] is mapped into the closest integer from 0 to 31. The same procedure is applied to the universe of discourse (integer value 1 ~ 256 in the x-axis). This integer conversion mechanism can be performed automatically by the manipulation of the user in the GUI environment. In this paper, we design 4 fuzzy processing units to be operated in parallel by FPGA to each band.

We also use the PCI interface bus between PC and the high-speed fuzzy inference system to be proposed in order to transmit the large volumes of data rapidly. For the GUI environment for pre-processing of images and tuning of themembership functions, an integrated GUI application program is implemented by VC++.

Manuscrip received Apr. 7, 2004; revised Jun. 3, 2004.
This work was supported by grant No. 2000-1-30300-007-2 from the Basic Research Program of the Korea Science & Engineering Foundation.

2. Fuzzy inference system environment

2.1 GUI environment

In this paper, we implement an integrated GUI environment system by VC++ under the Window 2000. This application program transmits data from PC via PCI bus to the parallel fuzzy processors and receives the computed results of the parallel fuzzy inference system via PCI and shows the classified images on the screen. This application program of GUI environment for pattern classification of the remote sensing images consists of 3 stages as shown in the Fig. 1.

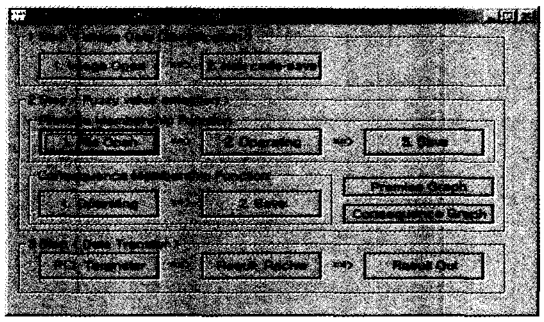

Fig. 1. GUI environment for fuzzy inference

2.2 Images analysis (Stage 1)

In the stage 1, this application program receives the remote sensing data (band2, band3, band5 and band7), and converts them to the format of integer values (from 1 to 256) of their digital images with gray levels. Each intensity value is used by single input value in the parallel fuzzy processor.

2.3 Membership function value (Stage 2)

The values of the fuzzy membership functions in the condition and consequent parts are extracted, and can be adjusted if needed in the stage 2. As showing in the Fig. 2, users can easily adjust the shapes of the fuzzy membership function by indicating points (vertexes) that construct a fuzzy linguistic term. The application program also converts these values of fuzzy membership functions to the formats of integer values by mid-point scan algorithm. As applying above, real values in [0, 1] of the fuzzy membership function are mapped into the integer values in [0, 31]. Hereafter, all fuzzy operations are performed by these integer values very rapidly.

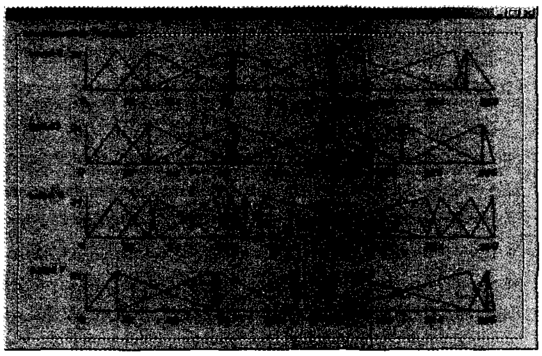

Fig. 2. Adjustment of membership function

In general, a line is represented as

$$F(x,y) = ax + by + c = 0.$$

If $dy = yk+1 - yk$, and $dx = xk+1 - xk$, this line can be written as

$$y = (dx/dy)x + B.$$

Therefore, $F(x,y) = dyx - dxy + Bdx = 0$.

Here, $a = dy$, $b = -dx$ and $c = Bdx$.

To apply the mid-point scan algorithm, we have to compute $F(xk+1,\ yk+1/2)$ to test its sign. Therefore, the decision variable is represented as

$$d = F(xk+1,\ yk+1/2) = a(xk+1) + b(yk+1/2) + c$$

Here, we select the next pixel E if $d \leq 0$.

When E is chosen, dnew and dold can be written as

$$d_{new} = F(xk+2, yk+1/2) = a(xk+2) + b(yk+1/2) + c$$

$$d_{old} = a(xk+1) + b(yk+1/2) + c$$

Therefore, $\Delta E = d_{new} - d_{old} = a + dy = 2dy$

If $d \langle 0$, then we choose the pixel NE. When NE is chosen, d_{new} and d_{old} can be written as

$$d_{new} = F(xk+2, yk+3/2) = a(xk+2) + b(yk+3/2) + c$$

$$d_{old} = a(xk+2) + b(yk+1/2) + c$$

Therefore,

$$\Delta NE = dnew - dold = a + d + dy - dx = 2(dy - dx)$$

The value of decision variable at the start point is

$$dstart = F(x0+1, y0+1/2) = a + b/2 = 2dy - dx \quad [3]$$

(example)

If the starting point is (5, 8) and ending point is (9, 11), we can easily find the next pixel, point by point, with only integer addition operations.

dx = 4

dy = 3

dstart = 2

ΔE = 6

ΔNE = -2

Using above values, the way to find y value as incrementing x by 1 is the followings;

1) from (5, 8), d = 2, d ≥ 0, so select *NE*
2) from (6, 9), d = 2 - 2 = 0, d ≥ 0, so select *NE*
3) from (7, 10), d = 0 - 2 = -2, d < 0, so select *E*
4) from (8, 10), d = -2 + 6 = 4, d ≥ 0, so select *NE*
5) from (9,11), ending point

We can select and plot the next integer pixel point by point whose fuzzy membership degree is closest to the line path. This efficient scan line conversion algorithm converts lines of fuzzy terms by using only incremental integer computations.

By making the slope of the line less than 1.0, the aliasing effect of line drawing is reduced. In this paper, we define the x coordinate as 1~256, and the y coordinate as 0~31, so the slope of each line of fuzzy terms is always set less than 1.0. We can compute the values for membership function of the condition and the consequent part by this method, and save values in integer type.

2.4 Data transmissions and display the results (Stage 3)

In the stage 3, input values from the stage 1, fuzzy membership functions in the condition and consequent parts, and 104 fuzzy rules are sent to the parallel fuzzy inference system via PCI bus. The application program receives the results computed in the parallel fuzzy inference system via PCI bus, displays the images of classified pattern on the screen.

3. Remote sensing images

3.1 Analysis of remote sensing data

The given test image is covered over Daeduk Science Complex Town, Daejeon, Korea and consists of 3 visible and 1 near infrared bands. Fig. 3 shows this images represented by gray-scale levels.

From the 3 (7-5-3 band and 7-3-2 band) among 4 bands, false color images are generated form their gray-scale levels. Using these images, we choose the 8 classes. (C1= Coniferous Tree, C2= Deciduous Tree, C3= Water, C4= Asphalt Road, C5= Cement Road, C6= Shadow, C7= Bare Soil, and C8= Dried Grass)

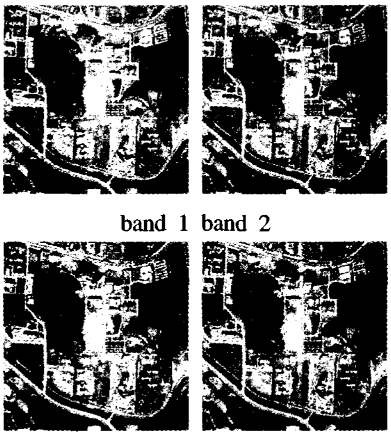

Fig. 3. Remote sensing images about each band

In each band, there are 12 fuzzy linguistic terms. Gray levels are classified into 256. To represent that, we use 3 points in the x-coordinate to display the fuzzy membership functions by using the GUI environment. Table 1 shows the x-coordinate values corresponding to band 2.

Table 1. x-coordinate values in the band 2

band 2			
x1	x2	x3	fuzzy set
0.000000	19.615385	39.230769	-6
19.615385	39.230769	39.230769	-5
39.230769	39.230769	90.690885	-4
39.230769	90.690885	92.657054	-3
90.690885	92.567054	92.680965	-2
92.657054	92.680965	152.515507	-1
92.680965	152.515507	155.374350	1
152.515507	155.374350	155.421446	2
155.374350	155.421446	230.331790	3
155.421446	230.331790	238.030905	4
230.331790	238.030905	238.661843	5
238.030905	238.661843	255.000000	6

Band 3, 5 and 7 are represented as the similar manners.

3.2 Fuzzy membership functions for pattern classification

We use 104 fuzzy rules for the pattern classification. There are 12 fuzzy linguistic terms in the condition part. The membership value of each fuzzy rule for a given pixel's image is computed by fuzzy implication. The output of the fuzzy controller is inferred from the input and the fuzzy rule by compositional rule of inference. The inferred fuzzy output is defuzzified by taking a deterministic value which represents one out of 8 classes. There are 8 fuzzy linguistic terms in the consequent part.

3.3 Fuzzy control rules

In this paper, we use only 104 fuzzy rules out of 778 fuzzy rules produced initially after classifying and selecting the test image from the training set. The form of the fuzzy rules is as following.

```
Rule 1: IF I1 IS -3 AND I2 IS -3 AND I3 IS -3
        AND I4 IS 3 THEN C1
Rule 2: IF I1 IS -3 AND I2 IS -3 AND I3 IS -3
        AND I4 IS 2 THEN C1
                    .
                    .
                    .
Rule 104: IF I1 IS 1 AND I2 IS 2 AND I3 IS 2
        AND I4 IS 4 THEN C8
```

3.4 Defuzzification

For defuzzification, there are many different methods, however, among them COA(center of area) is more common. The COA method has a good steady-state performance, and a defuzzification system based on the COA method generally yields a lower mean square error than those that are based on other methods. In this paper, the required data structure in the defuzzification stage is an integer array that can store the y-axis integer values corresponding to the 256 x-axis integer pixels in the consequent part. In this array, the y-axis integer value from 0 to 31 are stored. On computing the consequent part, this array must contain the maximum values in all fuzzy rules. The followings are the algorithm for this.

```
begin
     integer array defuzz[1: m]
     integer array max[1: m] ← 0

     for i = 1 step 1 until n do
     begin
        for j = 1 step 1 until m do
        begin
           operation of the integer pixel conversion
           return value ← defuzz[1:m]
         if max(j) < defuzz(j)
           then max(j) ← defuzz(j)
         end
      end
end.
```

Here, n is the number of the fuzzy rules, and m is the number of integer pixels in the x-coordinate.

In the defuzzification processing, in general, the part of above 55% in the x-coordinate have the "0" values. To overcome this problem, in this paper, we also propose an algorithm to find the start position and the ending position having the non-zero items. Using this concept, the unnecessary operations in the defuzzification process will be greatly reduced.

```
begin
   lower ← 256
   upper ← 1
   begin
      procedure left_line( )
      procedure right_line( )
      procedure middle_line( )
   end
   if (x1 < lower) then lower ← x1
   if (x3 > upper) then upper ← x3
end.
```

Here, procedures left_line, right_line, and middle_line process the integer pixel conversion operations of the triangular typed membership functions in the part of the left, right and middle lines, respectively. x1 and x3 are the leftmost point and rightmost point in each membership fuction.

4. High-speed fuzzy inference system

4.1 Organization of fuzzy inference system

4 gray-scale images of each band for remote sensing data are inputted to the high-speed fuzzy inference system. Fig. 4 illustrates the organization of the proposed high-speed fuzzy inference system.

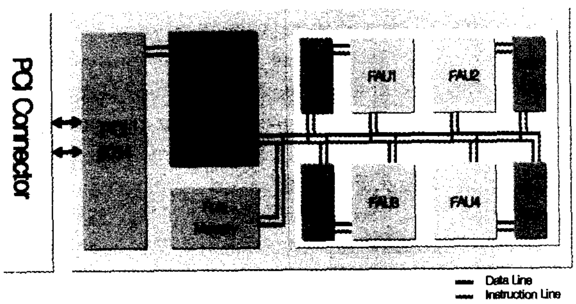

Fig. 4. Organization of high-speed fuzzy system

Each Component performs the following function.

(1) PCI 9054 Interface Chip

It controls the PCI bus and conducts the transmission of the instruction and data between application program and the PCI board.

(2) MC (Main Controller)

It saves the fuzzy rules and fuzzy membership functions sent from the application program via PCI bus to the corresponding memories, and controls 4 FAUs (Fuzzy Arithmetic Unit) simultaneously.

(3) FAU (Fuzzy Arithmetic Unit)

It performs the computations of the fuzzy MIN /MAX operations by using the fuzzy input values and fuzzy terms saved in the local memory. The proposed system can process the 4 FAU operations in parallel.

(4) MFiLM (Membership Function i Local Memory)

It saves the values of all fuzzy terms belong to the i-th condition and consequent part. In this paper, the number of the fuzzy terms in the condition part is 12 and consequent part is 8. It also has the t1 and t2 regions in order to save the intermediate values in computing of the consequent part.

(5) RM (Rule Memory)

It saves the entire fuzzy rules and each value of the degree of fulfillment in the condition part.

4.2 Organization of high-speed fuzzy inference system

Table 2 shows the registers in each unit.

Table 1. Registers

unit	name	bit	unit	name	bit
MC	c	3	FAU	in_f	8
	f	1		in_c	5
	s	1		v1~12	5
	pr1~4	4		t	5
	cr	4	RM	r1~r104	20
	pv1~4	5		pv_r1~r104	5
	t1~2	5	MFLM	p1~p12_(0~255)	5
FAU	f	1		c1~c8_(0~35)	5
	r	4		t1~t2(0~35)	5

4.2.1 MC(Main Controller)

c field selects the memory location to be saved for data among the 5 memories (RM, MF1LM, MF2LM, MF3LM and MF4LM). f flag shows the state representing the condition or consequent part. s flag means the start signal. pr and cr save the indexes of fuzzy rules in the condition and consequent part, respectively. pv value are used in finding the minimum values of the 4 α-level values in the condition part. t1 and t2 are temporary registers for magnitude comparison to the results in the consequent part.

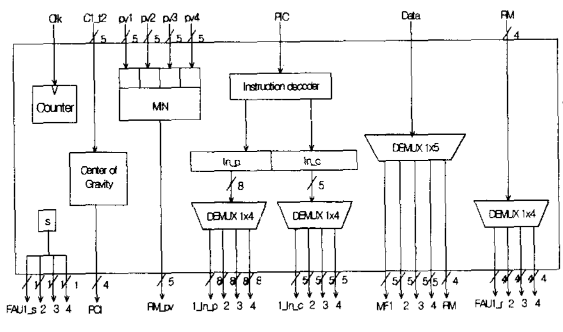

Fig. 5. Inside architecture of MC

To send the data from PCI to each memory module, sequence is controlled by a counter. Inside architecture of MC is illustrated in Fig. 5 and performs as follows.

(1) Incoming data from PCI bus are sent to the appropriate

memory module according to the c register.

(2) in_p, in_c sends input value of condition part and consequent part by relevant FAU's in_p and in_c according to control order of MC.

(3) Send the fuzzy control rules saved in RM to appropriate r in FAU.

(4) s signal start the fuzzy computations.

(5) In the case of the condition part, saves the computed value of in_p and r of each FAU to pv and finds the smallest value among the values of pv and saves it to RM_pv.

(6) Compute the center of gravity of the t2 in the MF memory, and transmit the controlled to the application program via PCI bus.

4.2.2 Fuzzy Arithmetic Unit

FAU(Fuzzy Arithmetic Unit) performs the fuzzy computation about the input value of each band.

f flag indicates the state whether the system is computing on the condition or consequent part. In-p and In-c have the input values for operation in the condition and consequent part, respectively. t is a temporary register used in comparing input value and fuzzy membership value in the consequent part. r register has the index value of fuzzy term coming from the current fuzzy rule. v register save the fuzzy membership degree.

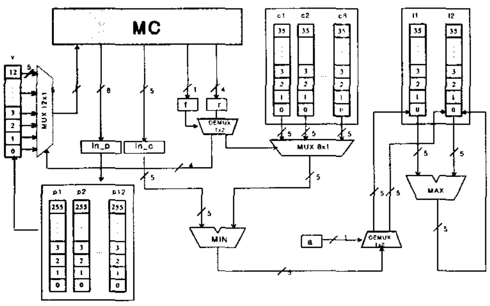

Fig. 6. Inside architecture of FAU

Inside architecture of FAU is illustrated in Fig. 6 and performs as follows.

(1) In the care of the condition part, find the appropriate value among the 12 fuzzy terms, according to in-p and r, and sent it to pv in the MC.

(2) In the care of the consequent part, after comparing in_c with the corresponding fuzzy term in C1 memory by the value of r, save the minimum value of them to register t2 on the first data of in_c.

(3) From the second data, save the minimum value to register t1. After comparing t1 and t2, save the maximum value of them to register t2.

4.2.3 Rule Memory

RM(Rule Memory) has the index numbers of fuzzy terms about entire fuzzy rules. Registersfrom r1 to r104 have fuzzy rules for the fuzzy operations. Find degrees of fulfillment in the condition part are saved in register pv.

4.2.4 MFiLM (Membership Function i Local Memory)

In each register from p1 to p12 and from c1 to c8, the membership values of the corresponding fuzzy term in the condition and consequent part are saved in the quantization form. t1 and t2 registers save the result values computed in the consequent part. Each FAU has its own MFiLM memory.

4.3 Sequence of high-speedfuzzy inference

In our system, the condition and consequence part have 12 and 8 fuzzy terms, respectively. Inferred results are come for the entire 104 rules. The sequence of the system is as followings as

(1) Save the membership values of each fuzzy term of the condition and consequent part to the MFiLM, respectively, according to the control of MC.

(2) Save the entire fuzzy rules to RM.

(3) Save the fuzzy singleton input (gray level) to the register (in_p) in the FAU, operated 4 units in parallel.

(4) Find the matched row of the local memory according to the in_p, and save them to v-register.

(5) MC transmits the partial fuzzy rules of the condition part to each FAU.

5. Experiments

The pattern classification on the large volumes of the multi-spectral image data obtained from the satellites or aircrafts has become an important tool in order to generate the ground cover maps. In this paper, we have implemented the integrated GUI environment system with 4 parallel fuzzy processing units to be operated in parallel on the classification of the sensed image data. In this, we have solved the problems of taking longer times as the fuzzy real computations of [0, 1] by using the integer pixel conversion algorithm. This gives the inferred results as faster as 20.7 times rather than conventional methods. We also proposed the new algorithm to reduce the non-zero items in the defuzzification stage. From this, the computations in the defuzzification stage are performed as about 11% as fast compared as the conventional methods. We used the PCI interface bus between PC and the proposed parallel fuzzy inference system in order to transmit the large volumes of data rapidly. Considering all the pixels from test image, it performs well with 92.57% correct classification. This is caused by the rounding operation of integer pixel mapping. However, only integer computations with 4 parallel inference units give much faster classification results.

6. Conclusions

We have implemented the intgrated GUI environment system having only integer fuzzy operations in the consequent part and the defuzzification stage. This system is also operated in parallel manner for MIMO(mutiple input and multiple output) or MISO(multiple input and single output) systems. There are several advantages and new possibilities found in the proposed GUI environment for the high-speed fuzzy inference system;

(1) The integrated GUI environment gives users easy approaches to the parallel fuzzy inference system, the fuzzy tuning procedure, and the monitoring of the classified images.

(2) This system brought a high-speed fuzzy computation time by using the integer pixel conversion algorithm.

(3) This system eliminated the unnecessary operations of the non-zero items on the membership function in the defuzzification stage.

(4) Several fuzzy rules are executed simultaneously.

(5) This system brought a relatively higher speedup, compared with the single processor system.

(6) The proposed architecture is very efficient to MIMO fuzzy control system, as well as MISO system.

(7) The proposed architecture featured not only its flexibility but also scalability because of the optimized parallelisms.

(8) PCI interface was used for the fast transmission of the large volumes of the sensed image data.

The proposed system can be applied to the ground cover map, an environmental control and GIS system that require the fast processing time with large volumes of data. And this system will be well applied to build powerful architectures for control applications, like robot control, with time-critical sensor integration.

References

[1] J. Yen and R. Langari, *Fuzzy Logic: Intelligence*, Pretice Hall, 1999.

[2] M. M. Hadhoud, "Multichannel Adaptive Filter for Remote Sensing Multi-spectral Images," ICECS'98, pp. 1214-1222, Dec. 1998.

[3] F. S. Hill, JR., *Computer Graphics*, 2nd ed., Prentice Hall, 2001.

[4] M. J. Patyra and et al., "Hardware implementations of digital fuzzy logic controllers," *Information Sciences*, Vol. 113, pp. 19-54, 1999.

[5] G. Ascia et al., "VLSI Hardware Architecture for Complex Fuzzy Systems," *IEEE Tr. on Fuzzy Systems*, Vol. 7, No. 5, pp. 553-570, Oct. 1999.

[6] S. Kagei, "Fuzzy Relational Equation with Defuzzification - Algorithm for the Largest Solution," *Fuzzy Set and Systems*, Vol. 123, pp. 119-127, 2001.

[7] PCI 9050-1 Data Book, PLX technology, 1998.

Sang Gu Lee

He received the B.S. degree in electronics engineering from Seoul National University, Seoul, Korea in 1978, and the M.S. degree in Computer Science from KAIST, Korea in 1981. He received the Ph.D. degree in Electrical Electronics and Computer Engineering from Waseda University, Tokyo, Japan. Since 1983, he has been a professor in division of Computer Engineering, Hannam University, Taejon, Korea. His current research interests are in parallel processing, parallel architecture and parallel neuro-fuzzy computing.

Phone : +82-42-629-7551
Fax : +82-42-487-9335
E-mail : sglee@mail.hannam.ac.kr