Jurnal Teknik Elektro Vol. 3, No. 2, September 2003: 84 - 87

Handwriting Digit Recognition With Fuzzy Logic

Hany Ferdinando
Electrical Engineering Depaartment, Faculty of Industrial Technology, Petra Christian University
Control Engineering, Faculty of EE-Math-CS, University of Twente, P.O.Box 217, NL-7500 AE Enschede
the Netherlands
Email: hanyf@petra.ac.id, h.ferdinando@student.utwente.nl

Abstract

To recognize handwriting digit is not a difficult task for human, but for a computer, it could be very difficult. This project implements the Fuzzy Logic system to recognize the handwriting digit. There are 3 constraints need to be considered here, they are: the real data were written with the same pen; the real data will be scanned into image data and then converted to BW mode with other software outside this project; program will read image data file instead of capturing with special device such as camera. Software is implemented in matlab. The design of fuzzy logic will use fuzzy logic editor. Before processing with Fuzzy algorithm, it needs to process the image then to get its features. Only simple image processing technique will be used. Feature extraction was made with a vertical and two horizontal lines. The position of crossing point between these lines with the image data will be a feature. These pre-processed data will be an input parameter for the fuzzy system. The fuzzy system has 7 inputs and 1 output with 57 rules. The average result of recognizing process is 80% after membership functions tuning.

Keywords : fuzzy logic, character recognetion, handwriting recognition.

Abstrak

Bukan masalah yang rumit bagi manusia untuk mengenali angka yang ditulis oleh orang lain, tetapi tidak untuk computer. Paper ini akan membahas bagaimana menggunakan Fuzzy Logic untuk mengenali tulisan angka. Ruang lingkup paper ini dibatasi oleh 3 hal berikut: angka akan ditulis dengan menggunakan pena yang sama; hasil tulisan tersebut akan discan menjadi data mentah dan diubah menjadi gambar BW dengan menggunakan software di luar proyek ini, data ini akan langsung dibaca oleh program untuk dikenali (program tidak membaca data melalui kamera). Program dibuat dalam matlab dengan dibantu fuzzy logic editor. Untuk dapat mengenali data gambar tadi, harus dilakukan pengambilan informasi yang mewakili data tersebut (feature extraction). Cara yang dilakukan sangat sederhana, yaitu dengan menggunakan 2 garis horisontal dan 1 garis vertikal. Posisi titik potong antara garis-garis tersebut dengan angka merupakan data yang menjadi input untuk fuzzy logic. Fuzzy logic diimplementasikan dengan menggunakan 7 input, 1 output dan 57 aturan. Dari hasil percobaan, didapat bahwa kemampuan rata-rata program untuk mengenali angka adalah 80%. Hasil ini didapat setelah dilakukan tuning terhadap fuzzy logic.

***Kata kunci** : fuzzy logic, pengenalan karakter, pengenalan tulisan tangan.*

Introduction

To recognize handwriting digit is not a difficult task for human, but for a computer, it could be very difficult. The problem is how to program a computer to do that kind of task. With conventional approach, it is very difficult to formulate the solution, but the intelligent approach has been developed for this kind of job. The development of Fuzzy Logic system, Artificial Neural Network, Genetic Algorithm, etc. have helped human to program computer to decide something that imitate organism behavior, although in a very basic level.

This idea can be applied for example to convert some kind of image of book into real text document instead of doing it manually. Off course it needs not only digit recognizing, but also character. This project is limited to the digit only. The result will be compared to the same problem but with Neural Network [3].

Fuzzy Logic System

Prof. Lotfi Zadeh discovered fuzzy logic in 1965 from his idea about fuzzy sets. In fuzzy system, all input will be converted into certain variable through fuzzification process. This kind of variable is defined such that people can understand easily. For example: poor, good and excellent [1] are used to represent the satisfaction

Note: Discussion is expected before November, 1st 2003. The proper discussion will be published in "Jurnal Teknik Elektro" Volume 4 Number 1 March 2004.

level of service in a snack bar. Those are called label in the input membership function (mf). People cannot define that criteria exactly, but it is the real situation around us. Figure 1 shows general block diagram of a Fuzzy system. At the output side, there also label for its parameter. Both input and output are connected by inference mechanism. The result of fuzzy logic is still in fuzzy variable, to set it back to real condition value, the de-fuzzification process is used.

Fuzzy Logic Toolbox in Matlab [1]

In matlab, there is a fuzzy toolbox. This toolbox is equipped with FIS editor, which enable user to design fuzzy system easily. With this editor, people can design input and output mf, write the rules and even make some test with certain input value to see its behavior. The result is saved in .fis file.

To combine this design inside matlab code, the readfis.m and evalfis.m can be used. First, user has to read the fuzzy configuration (readfis) and then to evaluate the input signal (evalfis) to get the result.

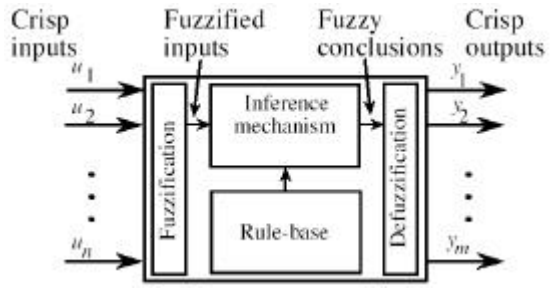


Figure 1. Block diagram Fuzzy system [2]

Constraints

Before continuing, it needs to define several constraints for this project, i.e.:

- The real data were written with the same pen.
- A scanner will transfer those real data into digital image data.
- The digital image data will be converted to BW mode with other software outside this project to be ready image data.
- Program will only read the ready image data.

Image Pre-processing

The raw image data is in black pixel with white background. For the position of the interested object inside the real image data is varying, it needs to get only the image, without any border. Cropping the image is the solution. The algorithm for cropping is to detect the black pixel line by line then cut the raw image data according to the first detected black pixel. Figure 2 shows the image pre-processing step from original image until ready image.

Cropping will result an image with different size. For the two horizontal lines, it will be a problem, because their position should be fixed. But the vertical line has no problem. So, the resizing is applied to the height only, resulting image with the same height but different width with the same height, it is easy to place the two horizontal lines.

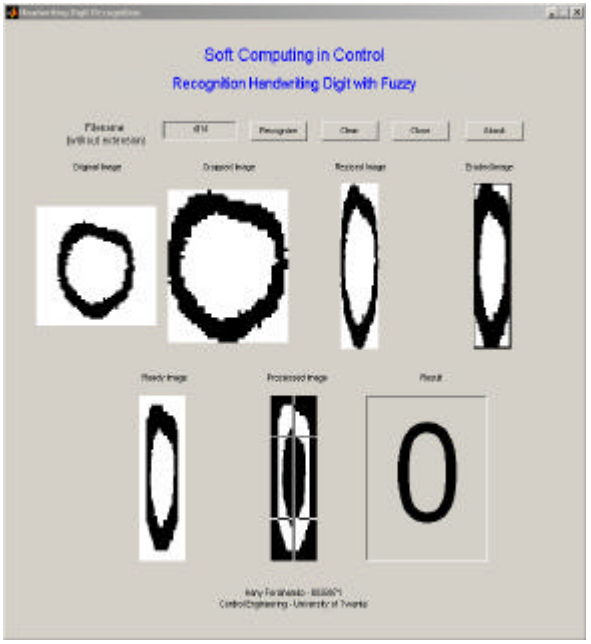


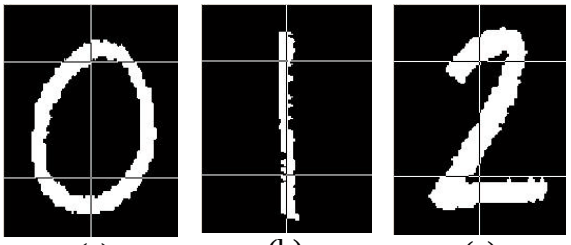
Figure 2. GUI of this project

To resize something means to resize everything inside that object. On this project, all images will be enlarged. It makes the image not smooth anymore. This can disturb the feature extraction process. To make the enlarged image smooth, erode process is done. This process will remove unnecessary pixel from and fill some pixels to the image. The effect can be used as a simple smoothing process. For this process results a black border at the edge of the image, at the end of the process this border has to be eliminated, then white border is added in order to be a ready image (see figure 2). This ready image will be an input for the fuzzy logic system after changing black to white pixels and vice versa.

Feature Extraction

It is impossible to process image directly with fuzzy logic or other system, because the number of input will be very large. For this purpose, the feature extraction is used. Feature extraction means to get the most significant information to represent the whole data in small number of new data.

The idea is using a vertical and two horizontal lines inside the image data (see figure 3). The feature is the crossing position between these lines and the white pixel. To get the single number of position instead of its range, the median value will be taken. Figure 3(a) shows that number '0' has two crossing position for each line. So, there will be six positions for its feature. All image number will be treated with the same manner. From the experiment, it is known that the maximum number of crossing position is seven. So, the number of input is seven.



(a) (b) (c)

Figure 3. Feature extraction using 3 lines.

Implementation

As discussed above, the feature extraction process is done with 3 lines. The features from horizontal lines need to be normalized into relative position instead of absolute one. It means it should be divided by the width of the image. So, feature from wide and narrow image will have the same relative position.

GUI (Graphical User's Interface) was made for this project in order to get simplicity in running the system. For a demo system, GUI is a good representation. This GUI will enable user to input the filename then ask computer to recognize this image as a number. User can also see the original and processed image. The processed image contains the vertical and horizontal line, so user can also know the position of the crossing point. Figure 3 shows this GUI when it is running to recognize number '0'.

User only enters the filename (without extension) then click 'Recognize' button. Software will crop, resize and erode the image, get the features from it and then process them with fuzzy system. The original and processed images are displayed also in order to give the visualization of process inside the system.

To clear all representation, user can click the 'Clear' button. The brief information is shown when user click 'About' button. Such an abstract will explain this project briefly.

Experimental Result

First, this project applied fuzzy without image pre-processing (except for the resizing technique). The result will be compared to that with complete image pre-processing as describe before. Off course the first experiment will have 'more beautiful' representation after pre-processing (see figure 2) than the second implementation (see figure 3).

Table 1 shows all experimental result. The image processing technique is needed since this project relates to image. Without that process, the system performance cannot be increased.

At the first experiment, the system performance for each number is increased or remains the same. Membership function tuning can increase the average performance from 66% to 73.33%.

Table 1. Summary of experimental result

Number	Without image pre-processing		With image pre-processing	
	Before tuning (%)	After tuning (%)	Before tuning (%)	After tuning (%)
0	80.00	93.33	100	100
1	86.67	100	66.67	86.67
2	53.33	73.33	53.33	66.67
3	40.00	46.67	66.67	66.67
4	40.00	53.33	66.67	66.67
5	66.67	73.33	93.33	93.33
6	86.67	86.67	66.67	93.33
7	86.67	86.67	53.33	73.33
8	53.33	53.33	73.33	73.33
9	66.67	66.67	86.67	80.00
Average	**66.00**	**73.33**	**72.67**	**80.00**

When the image pre-processing technique is used, the average performance (before tuning) is close to the final result of system without image pre-processing. After tuning, only performance of number '9' gets lower value, but in general, the tuning process makes this system better.

Conclusion

The average capability is 80% after tuning. It cannot also be conclude that this is the best result since there are no specific rules to tune fuzzy system. Changing one parameter can influence the other parameters. The idea is to use some algorithm to optimize it. People can use Neural Network or Genetic Algorithm for that.

Compare to [3], this project is not so good, because it's only 80% compare to 98%. Unfortunately, [3] used sophisticated image processing. But it is interesting to see inside the process, since this project only use simple image processing technique. Both projects have the same goal but different algorithm, since [3] used Artificial Neural Network.

Acknowledgements

I would like to acknowledge prof Ariën J. van der Wal for his kindness discussion and idea when this project was being done, also to Ir. Berend Jan van der Zwaag for his help in finding a simple feature extraction method for this project. Last but no least, I would like to thank to all peoples in Control Engineering Group who have given their 'nice' handwriting digit as samples to me.

References

[1] _____, Fuzzy Logic Toolbox User's Guide, The MathWorks Inc., 2001
[2] Passino, Kevin M. and Stephen Yurkovich, Fuzzy Control, Addison Wesley Longman, 1998
[3] van der Zwaag, Berend Jan, Handwriting Digit Recognition: a Neural Network Demo.