IMAGE SEGMENTATION USING FUZZY LOGIC AND GENETIC ALGORITHMS

M. Abdulghafour
Department of Computer Science, Sultan Qaboos University
P.O Box 36, Al-Khod 123
Muscat, Oman
E–mail: muhamad@squ.edu.om

ABSTRACT

Genetic algorithms (GAs) and fuzzy logic (FL) have been playing important roles in solving many problems in pattern recognition and image processing. This paper presents a hybrid approach of GAs and FL that is used to fuse (combine) extracted features from intensity and range images. GAs are used to help construct membership functions that are necessary to classify the strength of existence of image features through FL. Since range and intensity images provide different types of sensory modality, fusing the extracted features from these images reveals more accurate information about the scene. The extracted features are fused to generate a segmented image of the scene. The segmented image is compared with its ideal counterpart for the purpose of experimental evaluation.

Keywords: Edge detection; Fusion systems; Fuzzy set; GAs; Image segmentation.

1 INTRODUCTION

In a multisensory system, several sources of information can be used in a complementary fashion to gather different information which would yield more meaningful information otherwise unavailable or difficult to acquire by a single sensory modality. This method of reasoning is justified by the often incomplete, uncertain, and/or imprecise sensory reading [2, 3]. Numerous papers based on non–deterministic approaches have appeared in the literature dealing with combination of uncertain information. Bayesian approaches, the mathematical theory of evidence (*i.e.* Dempster's rule of combination), and FL are often used to deal with uncertain information. Bayesian approach and Dempester's rule of combination often fall short in their ability to provide stisfactory results when extreme conflicts among pieces of evidence are encountered [4, 8, 12]. Despite the success of FL in many real world applications, few limitations in their use must be pointed out. For a given problem, the construction of membership functions and the extraction of rule bases (RBs) have no unique solutions. Due to the nature of fuzziness, it is not expected that fuzzy engineers would perceive a subjective attribute identically. Often common sense and the experience of designers are important factors in the success of a given design [11]. It follows that the construction of membership functions has no specific rules. Researchers have recognized the shortcomings of FL-based schemes. GAs can serve as a searching algorithm to compute membership functions [13, 9]. Based on the concept of the best fitness value of a GA, optimal membership functions can easily be achieved. The selection process of best solutions is based on a test of their performances. Hence, a fraction of good solutions is selected, and other are eliminated (survival of the fittest.) The selected

Journal of WSCG, Vol.11,No.1, ISSN 1213–6972
WSCG'2003, Februry 3–7, 2003, Plzen, Czech Republic.

solutions are then used through the processes of reproduction, crossover, and mutation to generate a set of new solutions. These new solutions are expected to perform better than those generated in the previous evolution. The processes continue until a convergence within a generation of solutions is reached. For illustration, this hybrid approach is applied to extract different types of features from multisensory images. Features are used in a complementary fashion to generate complete scene information, which are necessary for segmentation of given images. Several images have been acquired using an Odetics Laser Range Scanner. Feature edge maps are extracted using the Gradient operator [7]. The edge maps from the range image report no information about texture features in the scene. Likewise, the edge maps from the intensity image have failed to provide complete information about roof edges. In the latter, the scanner is unable to sense low variations at the surface intersections of the object. This shows the necessity of integrating several feature maps.

In section 2, we describe an FL system, which has been used extensively in the literature. In section 3, we present a GA as a searching procedure. In section 4, we describe how the concept of GAs and FL systems are incorporated in a single system to solve an image processing problem. That is, we present a complete data fusion system that is based on FL and GAs to achieve image segmentation of real range and intensity images. In section 5, a summary of the performance of this work is presented.

2 A FUZZY LOGIC SYSTEM FOR DATA FUSION

Fuzzy sets theory allows us to characterize crisp measurements in terms of fuzzy concepts. This characterization is essential to manage the combination of measurements through their degrees of uncertainty [16, 14, 10]. Fuzzy values can be characterized by membership functions such as μ_{weak}, μ_{strong} etc. Formally, a fuzzy subset is characterized by membership functions, μ, over a set of elements of Θ: $\Theta = (\theta_1, \theta_2, \cdots, \theta_n)$ and $\mu : \Theta \longrightarrow [0,1]$. The fuzzy concept "weak" can be represented by a membership function that characterizes a weak portion of the universe of discourse. Suppose that we quantize the universe into portions such as weak, moderate, and strong. This will lead to three fuzzy-set values named as WE, MO, and ST. For each input in the universe of discourse, we assign a fit vector of length 3, where each element in the fit vector is assigned by the corresponding membership function. For two inputs θ_1 and θ_2, we have $\theta_1 \longrightarrow [\mu_{we}(\theta_1), \mu_{mo}(\theta_1), \mu_{st}(\theta_1)]$ and $\theta_2 \longrightarrow [\mu_{we}(\theta_2), \mu_{mo}(\theta_2), \mu_{st}(\theta_2)]$. Fuzzy reasoning is governed by rules that are sufficient to generate consensus among fuzzy entries, say θ_1 and θ_2. We construct a RB which is able to reflect subjective reasoning in particular when conditional statements are involved. Entries of the RB are found according to application domain and the nature of the problem. In general, RB asserted through mapping scheme that is able to map input values with different domains to an output value with only one domain. A mapping function $F : \theta_1 \oplus \theta_2 \longrightarrow \theta$ maps each element of the domain $\theta_1 \oplus \theta_2$ to only one element for θ. For instance, $\theta_1 \oplus \theta_2 = \{(a_1, a_2) \longrightarrow a\}$ =element of θ, where a_1, a_2, and a are prespecified linguistic values within the universe(s) of discourse [5]. In our case, they are the quantized segments *weak*, *strong*, etc. Note that $\oplus$ is defined as a combination operator such as logical or weighted average operators, where the selection of the operation depends on the formulation of the problem [14]. That is, $\theta_1 \oplus \theta_2 = \{(MO, ST) \longrightarrow ST\}$ = element of θ.

Outputs of fuzzy values are then defuzzified to generate a crisp value for the variable θ. Figure 1 shows an example of output from two possible rules. Often, the centroid method is used to recover crisp values for the output, which is given by

$$\theta_{ki} = \frac{\sum_{r=1}^{s} \mu^{(r)} \theta_{ki}^{(r)}(c)}{\sum_{r=1}^{s} \mu^{(r)}}. \qquad (1)$$

In this equation, r is the rule number, $\mu^{(r)} = \min\{\mu_{A_1}(v_{k1}), \ldots, \mu_{A_m}(v_{km})\}$, $A_j \subset \mathbf{V}_j$, $j = 1, \ldots, m$, $\theta_{ki}^{(r)}(c)$ is the center of the suggested output at rule r, and s is the number of rules involved in the computation.

3 GENERATING MEMBERSHIPS USING GAs

Given universes for inputs and outputs of a system and an RB governing the fuzzy reasoning between these inputs and outputs, one can assume the shape and the number of fuzzy values for each universe. To use GAs, we code the membership functions as some finite unsigned bit strings that are then concatenated to represent the entire universes. For illustration, consider the case of a single input(x)-single output(y) system with an RB and input-output values as shown in Fig. 2. In the RB, the symbols SM, MD, and LG represent the fuzzy values *small, medium*, and *large*, respectively. We assume that the universes for the input-output x

and y range from 0 to 60 and from 0 to 1, respectively. To simplify the illustrations, let membership functions be constrained at the left and right sides of the universes. Hence, the unknown variables are the lengths of the bases of the membership functions. That is, we have four unknown variables. For this problem, we will code each base as a binary unsigned integer of length 6. Strings are then concatenated to form a 24-bit string. The strings are mapped to values representing the lengthes of the bases of the assumed membership functions. This mapping process is computed through a simulation model

$$b^{(i)} = C^{(i)}_{min} + \frac{d}{2^L - 1}(C^{(i)}_{max} - C^{(i)}_{min}), \qquad (2)$$

where $C^{(i)}_{min}$ and $C^{(i)}_{max}$ are arbitrary constants; usually they are chosen as the minimum and the maximum of the universes ($C_{min} = 0$ and $C_{max} = 60$ for x and $C_{min} = 0$ and $C_{max} = 1$ for y), d is the decimal value of each sub-string, L is the length of the bit string of each base, and $b^{(i)}$ is the ith base of the membership functions. Details about this simulation model can be found in [13, 6].
A GA is implemented as follows. (1) After creating strings randomly for all variables, a set of parameters represented by these strings are passed through a numerical model such as the one suggested in Eq. 2. Solutions are evaluated with respect to the given values in the tables and then they are assigned fitness values. Those with the best fitness are selected and duplicated to substitute for those with low fitness values. Strings of the best solutions are considered for the second iteration of the GA. These solutions are expected to perform better than those which were eliminated. This part of the process in a GA referred to as *reproduction*. (2) In the second step of a GA, selected strings from the first step are mixed and matched in a random fashion. This step is important for solutions with high fitness values to share their properties among other strings that have relatively lower fitness values. This part of the process in a GA is referred to as *crossover*. (3) After mixing and matching, redundancy of 0 in the same locations of strings is not desirable. The reason is that the repetition could cause an early false convergence of the GA. The locations that are redundant must be checked and at least one of them is converted to 1. This part of the process in a GA referred to as *mutation*. Applying this procedure of GAs on the given example, the first iteration and the generated membership function are illustrated in Table 1 and in Fig. 3. The selected strings for the second iteration are
010101100101010110001101
010101100101010110001101
100110011010101111101011
100110011010101111101011.
Strings after crossover and mutation are
011101100101010110001011
010101100101011110011011
100110011010101111101101
100110011010101111101101.
This process of reproducing and evaluating strings continues until we get the optimal membership functions which have the best fitness for both x and y.

4 EXPERIMENTAL RESULTS

The implementation of this system is divided into five stages: detecting and extracting edge features, fuzzification or modeling of real data through GAs, fusion, defuzzification, and segmentation. In this experiment, two edge maps of intensity and range images are selected as inputs to the data fusion system. The combined output map is used as a reliable input for a segmentation procedure.

4.1 Edge Detection

In this work, the Gradient operator is used to detect and extract edges from range and intensity images. The gradient of an image function $f(x,y)$ is defined as a vector, which is given by

$$\mathbf{G}(x,y) = [\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y}]^T \qquad (3)$$

The magnitude of $\mathbf{G}(x,y)$ is equal to the maximum rate of increase of $f(x,y)$ per unit distance in the direction of the gradient $G(x,y)$ [7]. Hence, at a discontinuity in an image, the first derivative will give a peak response, which is an indication of edginess. In contrast, the derivative of a smooth region will give a low response, which is an indication of the absence of edginess. An approach for computing the gradient components G_x and G_y is based on the use of the Sobel operators. Assume a 3X3 neighborhood labeled as $x_1, x_2, \cdots, x_9$ around the point x_5. Using this mask, the components of the gradient at the center of the mask are given by $G_x(x,y) = (x_7 + 2\times x_8 + x_9) - (x_1 + 2\times x_2 + x_3)$ and $G_y(x,y) = (x_3 + 2\times x_6 + x_9) - (x_1 + 2\times x_4 + x_7)$. Computing G_x and G_y for each point in the image and substituting into Eq. 3 will result in a new image, which is often called the gradient image. Ideally, the gradient image highlights all edges and dims all smooth regions (see Fig. 6).

4.2 Fuzzification Using GAs

From a real robotics environment, intensity and range images are acquiring by using an Odetics

Laser Scanner. Based on the surface normal values of the range image, a new image can be derived [15]. Before fuzzification, edges are extracted from both image modalities. An edge in an image corresponds to a discontinuity in the scene. Through a successful edge detection, a scene can be partitioned into meaningful regions. The Sobel operator is used to detect edges [7]. The performance of a method to detect edges depends on the type of edge that needs to be extracted and on the type of data that makes up that edge. For example, roof edges (at the intersection of surfaces) can successfully be extracted from range images. The reason is that intersected surfaces in a range image introduce significant discontinuities in the scene. However, texture edges do not introduce any range discontinuities with respect to the scanner. Hence, intensity images are better suited for this type of edges. Since the domain of an edginess variable can have positive values on a gray scale, edginess strength can be normalized within the interval $[0, 1]$. For each edge map, two fuzzy values were chosen. The construction of the two membership functions and the percentage of overlapping between their clusters is determined by using GAs. During the implementation of the GAs, we would have two different procedures, each of which represents the system output when a single edge map is applied at the input. The input-output range and intensity edge data and their assumed RBs are shown in Fig. 4. In each case, the input-output unknown variables are the bases of four membership functions; two for each universe of discourse. Let us assign 6-bit string for each variable at random. Since we have four variables (the bases of membership functions), we would concatenate them to form a 24-bit string altogether. Assume that the strings that are used in the first iteration are
110101011001001101111011
110110111000110011010101
111100011110110011001100
110011110011110111001110.
Using the model given in Eq. 2 and evaluating each string on the basis of the fitness values, some of these strings are eliminated and others are considered for duplications. We repeat the procedure which was explained in section 3 to obtain the optimal membership functions for two separate sets of input-output cases, i.e., one set is for edge map from the range image and the other is for edge map from the intensity image. The GA of the input-output of the range edge map converges after 25 iterations. We repeat the same procedure for the intensity edge map, in which the GA converges after 19 iterations. In both cases, the number of iterations depends on the initial random strings, the cutoff value (the fitness value), and the given data. After convergence, optimal membership functions are obtained (see Fig. 5).

4.3 Fusion with Fuzzy Reasoning

The membership functions along with a new RB are used to generate a combined fuzzy value for both range and intensity edge maps. Reasoning is carried out through the following rules:
If x_1, is MO and x_2 is WE, then y is MO
If x_1 is ST and x_2 is WE, then y is MO
If x_1 is MO and x_2 is MO, then y is MO
If x_1 is ST and x_2 is MO, then y is ST.
At a location (α, β) in both maps, rules are chosen according to the pixel values with respect to its degree(s) of membership. Then, output fuzzy values are generated at the corresponding rules. If there is an overlap between the two membership functions when a pixel value is given, then more than one rule is needed to be included in the reasoning. This procedure is similar to the one in section 2. Finally, in order to recover crisp output values, the output of the fusion system is defuzzified. The defuzzification scheme is based on the centroid method as described in section 2.

4.4 Image Segmentation

Image segmentation is the process of dividing an image into regions, each of which corresponds to a "homogeneous" surface in a scene. Hence, our goal is to extract closed boundaries around surfaces. If complete edge information can be extracted, a reliable image segmentation can be achieved. Using different types of edge maps has the advantage of presenting most the information needed in the scene. After fusing the edge maps, distinct regions are identified. Figure 6 shows segmentation results of the two sets of range and intensity images. By inspecting the segmented image, all major regions that are isolated and labeled can be seen. In order to evaluate the segmentation process, results are tested and compared against an ideal segmented image which is synthetically generated. The evaluation criterion is based on the degree of similarity between actual and ideal segmented images. This degree of similarity may be quantified by computing a metric distance between the two segmented images [2, 1]. The degree of similarity between two sets of data x and y is given by

$$s_{xy} = (1 - \frac{d_{xy}}{d_{xw}})100\%, \tag{4}$$

where d_{xy} is the *Euclidean* distance between x and y and d_{xw} is a metric distance used as normalization factor taken with respect to the worst case, *i.e.*, the maximum distance between the worst case and the ideal case. Using this formula, the degree of similarity, s_{xy}, between two segmented cases is about 93%.

5 CONCLUSIONS

This research has resulted in the development of a segmentation approach to extract a complete edge map of an arbitrary scene. Real range and intensity images are acquired by using the Odetics Laser Range Scanner. Although images acquired by the Odetics are associated with random and systematic errors, the results from this approach have a high degree of similarity when compared to those obtained from an ideal case. Using the gradient operator, edge maps are detected and extracted from different image modalities. FL and GAs have been used in a complementary fashion to offset some of the undesirable properties that are inherited in the fuzzification of real data. GAs provide an adaptive searching approach in which a set of optimal solutions evolves over a sequence of solutions. These solutions are optimized through a test of fitness values that are associated with the generated solutions. The collaboration between FL and GAs would lead to take the human out of the loop and would simplify the design of membership functions. The simulation and implementation processes are carried out by using a 3000 Interprise Sun Microsystems machine. In closing, it is observed that if the search space of an optimization problem is quite large, then a parallel architecture for this hybrid approach would be more efficient.

6 ACKNOLEDGMENT

My thanks to DR. M. E. El–Gamal, Department of Engineering Mathematics and Physics, Cairo University, and Dr. K. Shihab, Department of Computer Science, Sultan Qaboos University, for their invaluable comments and suggestions with regard to the application of genetic algorithms.

References

[1] Abdulghafour, M., Chandra, T., and Abidi, M. Data fusion through fuzzy logic applied to feature extraction from multi–sensory images. *Proceedings of IEEE Int. Conference on Robotics and Automation 2* (Atlanta, Georgia, 1993), 359–366.

[2] Abdulghafour, M., Fellah, A., and Abidi, M. Fuzzy logic–based data integration: Theory and applications. *Proceedings of IEEE Int. Conference on Multisensor Fusion and Integration for Intelligent Systems* (Las Vegas, Nevada, 1994), 151–160.

[3] Abidi, M., Abdulghafour, M., and Chandra, T. Fusion of visual and range features using fuzzy logic. *The Journal of Control Engineering Practice (IFAC) 2, No. 5* (1994), 833–847.

[4] Berger, J. O. *Statical Decision Theory and Bayesian Analysis.* New York, NY: Spring–Verlag, 1985.

[5] El-Gamal, M., and Abdulghafour, M. Fault isolation in analog circuits using a fuzzy inference system. *The Journal of Computers and Electrical Engineering 29* (2002), 213–229.

[6] Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison–Wesley Publishing Company, 1989.

[7] Gonzalez, R., and Wood, R. E. *Digital Image Processing (3rd ed.).* Reading, MA: Addison–Wesley Publishing Company, 1992.

[8] Hau, H.-Y., and Kashyap, R. On the robustness of dempster's rule of combination. *IEEE Int. Workshop on Tools for Artificial Intelligence* (1989), 578–582.

[9] Jain, L. C., and Martin, N. M. *Fusion of Neural Networks, Fuzzy Sets, and Genetic Algorithms.* CRC Press, 1999.

[10] Klir, G. J., and Folger, T. A. *Fuzzy Sets, Uncertainty, and Information.* Prentice Hall, 1988.

[11] Kosko, B. *Neural Networks and Fuzzy Systems.* Englewood Cliffs, NJ: Prentice Hall, 1992.

[12] Pedrycz, W. *Fuzzy Sets Engineering.* CRC Press, Inc., 1995.

[13] Ross, T. J. *Fuzzy Logic With Engineering Applications.* McGraw-Hill, Inc., 1995.

[14] Silvert, W. Symmetric summation: A class of operation on fuzzy sets. *IEEE Trans. Syst., Man. Cybern. 9* (1979), 657–659.

[15] Sluder, J. C., and Abidi, M. Parallel range data processing: A real case study. *Proceedings SPIE On Intelligent Robots and Computer Vision X 1608* (1991), 248–261.

[16] Zadeh, L. A. *Fuzzy Sets and Applications: Selected Papers by L. A. Zadeh.* New York, NY: John Wiley and Sons, 1987.

string: (1)	b_1: (2)	b_2: (3)	b_3: (4)	b_4: (5)
10101001010101000101100111	42	21	5	39
01010110010101011000110 1	21	37	22	13
00111101110000100101011 0	15	28	9	22
10011001101010111110101 1	38	26	47	43
b_1: (6)	b_2: (7)	b_3: (8)	b_4: (9)	y_x: (10)
40	20	0.08	0.62	0
20	35.2	0.35	0.21	0
14.3	26.7	0.14	0.35	0
36.2	24.8	0.75	0.68	0
$y_x = 30$: (11)	$y_x = 60$: (12)	err: (13)	f: (14)	Count: (15)
0.06	1	0.81	0.94	0
0.82	1	0.9	1.04	2
0	1	0.75	0.87	0
0.62	1	0.98	1.15	2
$sum = 3.24$ $ave. = 0.86$ $max = 0.986$ $cutoff = 0.95$				

Table 1: The first iteration of the GA; $err = 1 - \sum(y_i - y_i\prime)^2, f = err/ave.$.

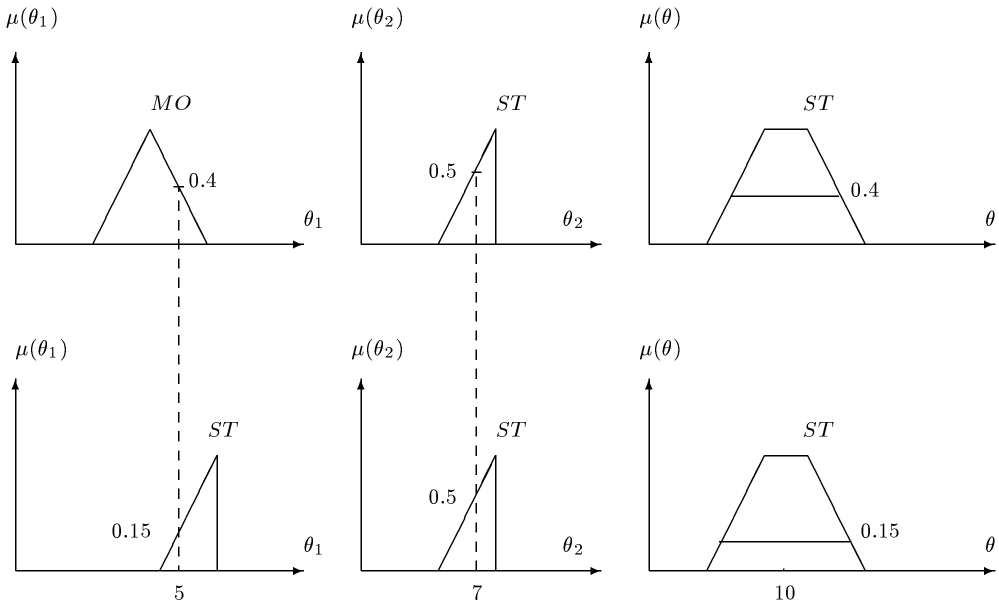

Figure 1: An example: Computing crisp output from fuzzy values.

x	0	30	60
y	0	0.5	1

(a)

x	SM	MO
y	0.5	1

(b)

Figure 2: Input–output data with their RBs.

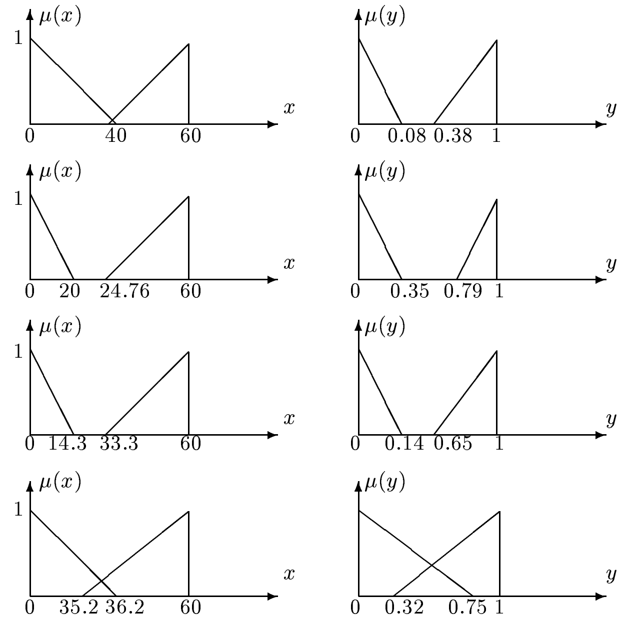

Figure 3: Membership functions generated by the first iteration of the GA.

x_2	0	0.45	0.72	1
y_2	0	0.45	0.75	1

(c)

x_2	MO	ST
y_2	MO	ST

(d)

x_1	0	0.4	0.7	1
y_1	0	0.5	0.8	1

(a)

x_2	MO	ST
y_2	WE	MO

(b)

Figure 4: Input-output values and their RBs: (a) and (b) for range map; (c) and (d) for intensity map.

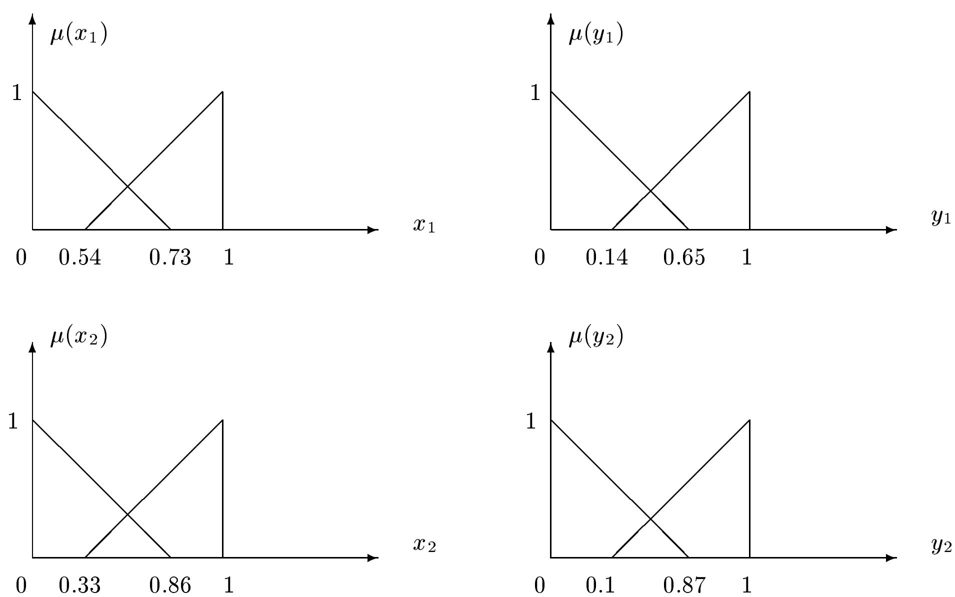

Figure 5: The membership functions generated by the GA.

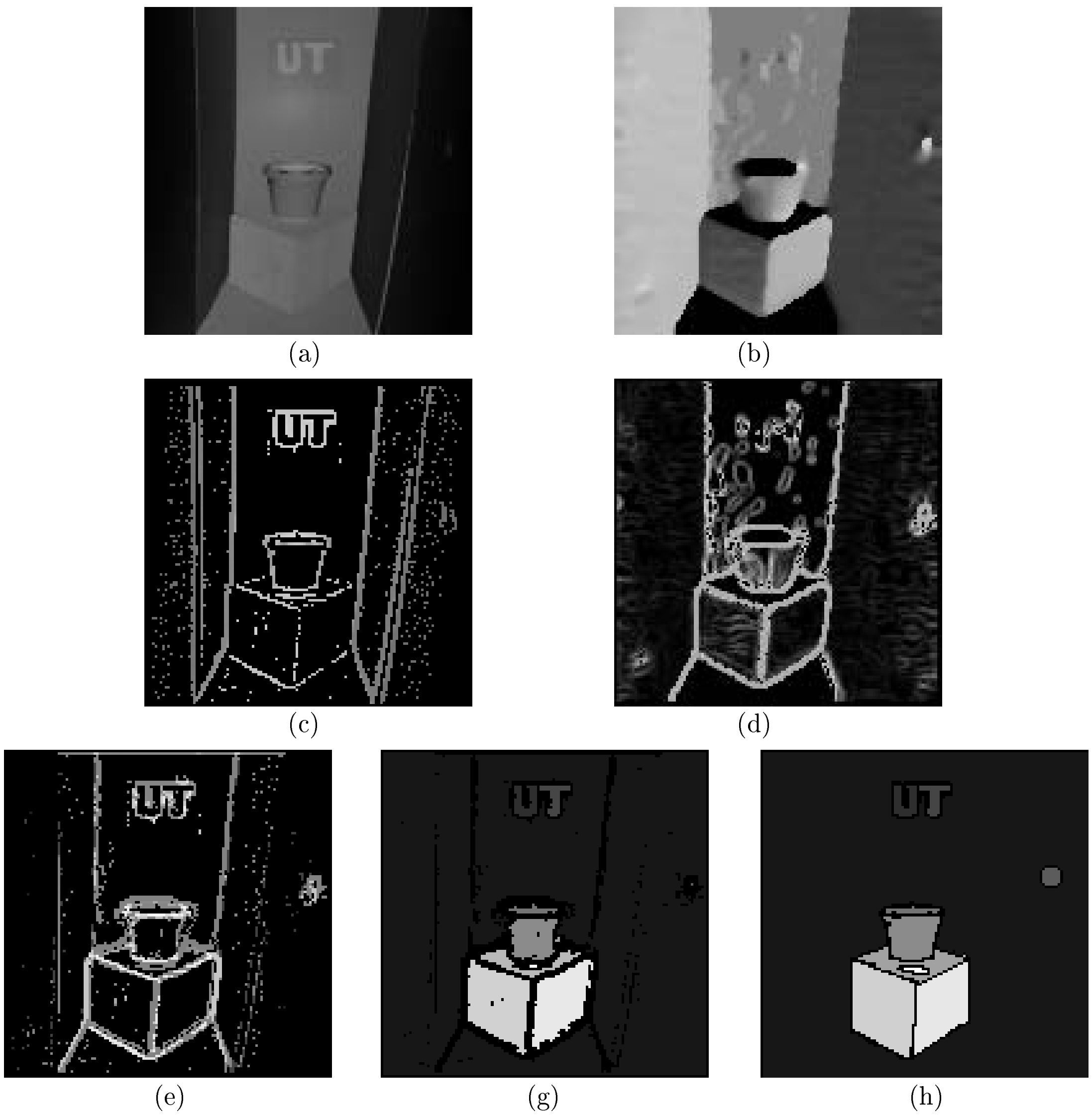

Figure 6: Image segmentation:(a) intensity image; (b) range image; (c) edge map from intensity image; (d) edge map from range image; (e) combined edge map from (c) and (d) using FL and GAs; (g) segmented image (isolating distinct regions); (h) an ideal segmented image of (e).