N93-29522

Automatic Rule Generation for High-Level Vision

Frank Chung-Hoon Rhee and Raghu Krishnapuram
Department of Electrical and Computer Engineering
University of Missouri, Columbia, MO 65211

ABSTRACT

Many high-level vision systems use rule-based approaches to solve problems such as autonomous navigation and image understanding. The rules are usually elaborated by experts. However, this procedure may be rather tedious. In this paper, we propose a method to generate such rules automatically from training data. The proposed method is also capable of filtering out irrelevant features and criteria from the rules.

1. Introduction

High-level computer vision involves complex tasks such as image understanding and scene interpretation. In domains where the models of the objects in the image can be precisely defined, (such as the blocks world, or even the world of generalized cylinders) existing techniques for description and interpretation perform quite well. However, when this is not the case (such as the case of outdoor scenes or extra-terrestrial environments), traditional techniques do not work well. For this reason, we believe that the greatest contribution of fuzzy set theory to computer vision will be in the area of high-level vision. Unfortunately, very little work has been done in this highly promising area. Fuzzy set theoretic approaches to high-level vision have the following advantages over traditional techniques: i) they can easily deal with imprecise and vague properties, descriptions, and rules, ii) they degrade more gracefully when the input information is incomplete, iii) a given task can be achieved with a more compact set of rules, iv) the inferencing and the uncertainty (belief) maintenance can both be done in one consistent framework, v) they are sufficiently flexible to accommodate several types of rules other that just IF-THEN rules. Some examples of the types of rules that can be represented in a fuzzy framework are [1] possibility rules ("The more X is A, the more possible that B is the range for Y"), certainty rules ("The more X is A, the more certain Y lies in B"), gradual rules ("The more X is A, the more Y is B"), unless rules [2] ("if X is A, then Y is B unless Z is C").

The determination of properties and attributes of image regions and spatial relationships among regions is critical for higher level vision processes involved in tasks such as autonomous navigation, medical image analysis and scene interpretation. Many high-level systems have been designed using a rule-based approach [3,4]. In these systems, common-sense knowledge about the world is represented in terms of rules, and the rules are then used in an inference mechanism to arrive at a meaningful interpretation of the contents of the image. In a rule-based system to interpret outdoor scenes, typical rules may be

IF a REGION is **RATHER THIN** AND **SOMEWHAT STRAIGHT**
THEN it is a ROAD

IF a REGION is **RATHER GREEN** AND **HIGHLY TEXTURED** AND
        IF the REGION is **BELOW** a SKY REGION
THEN it is TREES

Attributes such as "THIN" and "NARROW", and properties such as "BRIGHT" and "TEXTURED" defy precise definitions, and they are best modeled by fuzzy sets. Similarly, spatial relationships such as "LEFT OF ", "ABOVE" and "BELOW" are difficult to model using the all-or-nothing traditional techniques [5]. We may interpret the attributes, properties and relationships as "criteria". Therefore, we believe that a fuzzy approach to high-level vision will yield more realistic results.

In most rule-based systems, the rules are usually enumerated by experts, although they may also be generated by a learning process. Several techniques have been suggested in the literature to generate rules for control problems [6-9], some of which use neural net methods to model the control system [7-12]. These systems convert a given set of inputs to an output by fuzzifying the inputs, performing fuzzy logic, and then finally defuzzifying the result of the inference to generate a crisp output [13]. Some of the methods also "tune" the membership functions that define the levels (such as "LOW", "MEDIUM" and "HIGH") of the input variables [10]. While these methods have been shown to be very effective in solving control problems, they cannot be directly used in high-level vision

applications. For example, in control systems, the fuzzy rules have consequents which are usually a desired level of a control signal whereas in high-level vision, the consequent clauses are usually fuzzy labels. Also, it is desirable that membership functions for levels of fuzzy attributes such as "THIN", and "NARROW", and properties such as "BRIGHT" be related to how humans perceive such attributes or properties. Hence they have very little to do with the decision making or reasoning process in which they are employed. In many reasoning systems for high-level vision, confidence (or importance) factors are associated with every rule since the confidence in the labeling may depend on the confidence of the rule itself. In this paper, we propose a new method to generate rules for high-level vision applications automatically. The rules so obtained may be combined with the rules given by the experts to complete the rule base.

In Section 2, we describe several fuzzy aggregation operators which can be used in hierarchical (multi-layer) aggregation networks for multi-criteria decision making. In Section 3, we describe how these aggregation networks can be used to filter out irrelevant attributes, properties, and relationships and at the same time generate a compact set of fuzzy rules (with associated confidence factors) that describes the decision making process. In Section 4 we present some experimental results on automatic rule generation. Finally Section 5 contains the summary and conclusions.

2. Fuzzy Aggregation Operators

Fuzzy set theory provides a host of very attractive aggregation connectives for integrating membership values representing uncertain and subjective information [14]. These connectives can be categorized into the following three classes based on their aggregation behavior: i) union connectives, ii) intersection connectives, and iii) compensative connectives. Union connectives produce a high output whenever any one of the input values representing different features or criteria is high. Intersection connectives produce a high output only when all of the inputs have high values. Compensative connectives are used when one might be willing to sacrifice a little on one factor, provided the loss is compensated by gain in another factor. Compensative connectives can be further classified into mean operators and hybrid operators. Mean operators are monotonic operators that satisfy the condition: $\min(a,b) \leq \text{mean}(a,b) \leq \max(a,b)$. The generalized mean operator [15] as given below is one of such operator.

$$g_p(x_1,\ldots,x_n;w_1,\ldots,w_n) = \left(\sum_{i=1}^{n} w_i x_i^p\right)^{1/p}, \text{ where } \sum_{i=1}^{n} w_i = 1. \tag{1}$$

The w_i's can be thought of as the relative importance factors for the different criteria. The generalized mean has several attractive properties. For example, the mean value always increases with an increase in p [15]. Thus, by varying the value of p between $-\infty$ and $+\infty$, we can obtain all values between min and max. Therefore, in the extreme cases, this operator can be used as union or intersection. The γ-model devised by Zimmermann and Zysno [16] is an example of hybrid operators, and it is defined by

$$y = \left(\prod_{i=1}^{n} x_i^{\delta_i}\right)^{1-\gamma} \left(1 - \prod_{i=1}^{n} (1-x_i)^{\delta_i}\right)^{\gamma}, \text{ where } \sum_{i=1}^{n} \delta_i = n \text{ and } 0 \leq \gamma \leq 1. \tag{2}$$

In general, hybrid operators are defined as the weighted arithmetic or geometric mean of a pair of fuzzy union and intersection operators as follows.

$$A \oplus_\gamma B = (1-\gamma)(A \cap B) + \gamma(A \cup B) \tag{3}$$

$$A \otimes_\gamma B = (A \cap B)^{(1-\gamma)}(A \cup B)^\gamma \tag{4}$$

The parameter γ in (3) and (4) controls the degree of compensation. The γ-model in (2) is a hybrid operator of the type in (4). The compensative connectives are very powerful and flexible in that by choosing correct parameters, one can not only control the nature (e.g. conjunctive, disjunctive, and compensative), but also the attitude (e.g. pessimistic and optimistic) of the aggregation.

One can formulate the problem of multicriteria decision making as follows. The support for a decision may depend on supports for (or degrees of satisfaction of) several different criteria, and the degree of satisfaction of each

criterion may in turn depend on degrees of satisfaction of other sub-criteria, and so on. Thus, the decision process can be viewed as a hierarchical network, where each node in the network "aggregates" the degree of satisfaction of a particular criterion from the observed support. The inputs to each node are the degrees of satisfaction of each of the sub-criteria, and the output is the aggregated degree of satisfaction of the criterion. Thus, the decision making problem reduces to i) selecting robust and useful criteria for the problem on hand, ii) finding ways to generate memberships (degrees of satisfaction of criteria) based on values of features (criteria) selected, and iii) determining the structure of the network and the nature of the connectives at each node of the network. This includes discarding irrelevant criteria to make the network simple and robust.

In our previous research, we have investigated the properties of several union and intersection operators, the generalized mean, and the γ-model [14,17]. We have shown that optimization procedures based on gradient descent and random search can be used to determine the proper type of aggregation connective and parameters at each node, given only an approximate structure of the network and given a set of training data that represent the inputs at the bottom-most level and the desired outputs at the top-most level [14,17]. In this paper, we extend this idea to the detection of irrelevant attributes and automatic rule generation.

3. Redundancy Analysis and Rule Generation

In the approach we propose, we first fuzzily partition the range of values that each criterion (property or an attribute or a relation) can take into several linguistic intervals such as LOW, MEDIUM and HIGH. The set of properties or an attribute or a relation which are used are the ones that may appear in the antecedent clause of a rule. As explained in Section 1, the membership function for each level needs to be determined according to how humans perceive such attributes, properties or relations. The membership values for an observed attribute, property or relationship value in each of the levels is calculated using such membership functions. (Methods to generate degrees of satisfaction of relationships such as "LEFT OF" may be found in [18]). The memberships are then aggregated in a fuzzy aggregation network of the type shown in Figure 1. The top nodes of the network represent the labels that may appear in the consequents of the rules. A suitable structure for the network, and suitable fuzzy aggregation operators for each node are chosen. The network is then trained with typical attribute, property or relationship data with the corresponding desired output values for the various labels to learn the aggregation connectives and connections that would best describe in input-output relationships. The learning may be implemented using a gradient descent approach similar to the backpropagation algorithm [14,17]. It is to be noted that there is a constraint on the weights.
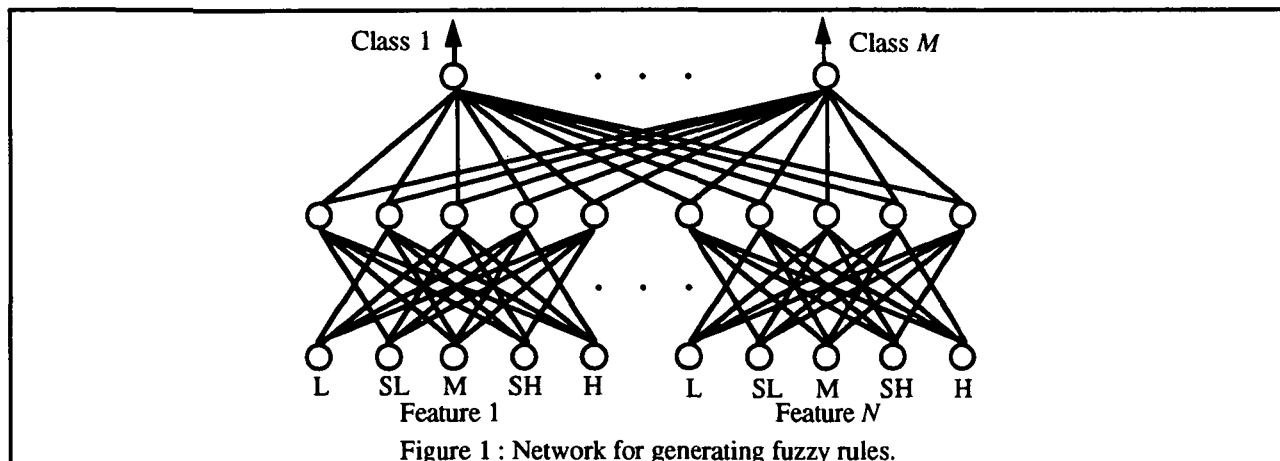

Figure 1 : Network for generating fuzzy rules.

Our experiments indicate that the choice of the network is not very critical. Also any compensative aggregation operator seems to yield good results. In all the results shown in this paper, we used the generalized mean operator as the aggregation operator. As indicated in Section 2, the generalized mean can closely approximate a union (intersection) operator for a large positive (negative) value of p. We start the training with the generalized mean aggregation function with $p=1$. If the training data is better described by a union (intersection) operator, then the value of p will keep increasing (decreasing) as the training proceeds, until the training is terminated when the error becomes acceptable. Also, the weights w_i in (1) may be interpreted as the relative importance factors for the different criteria. Initially we start the training with all the weights associated with a node being equal. As the training proceeds the weights automatically adjust so that the overall error decreases. Some of the weights eventually become

very small. Thus, the training procedure has the ability to detect certain types of redundancies in the network. In general, there are three types of redundancies (irrelevant criteria) that are encountered in decision making [17]. These correspond to uninformative, unreliable, and superfluous criteria.

Uninformative Criteria: These are criteria whose degrees of satisfaction are always approximately the same, regardless of the situation. Therefore, these criteria do not provide any information about the situation, thus contributing little to the decision-making process. For example, low texture content is a criterion that is always satisfied for both clear skies and roads, and hence it would be a uninformative criterion if one needs to distinguish between these two labels. Uninformative criteria do not contribute to the robustness of the decision making process, and therefore it is desirable that they be eliminated.

Unreliable Criteria: These correspond to criteria whose degrees of satisfaction do not affect the final decision. In other words, the final decision is the same for a wide range of degrees of satisfaction. For example, color would be an unreliable criterion for distinguishing a rose from a hibiscus because they both come in similar colors. Unreliable criteria do not contribute to the robustness of the decision making process, and therefore it is desirable that they be eliminated.

Superfluous Criteria: These are criteria which are strictly speaking not required to make the decision. The decisions made without considering such criteria may be as accurate or as reliable. For example, one may want to differentiate planar surfaces from spherical surfaces using Gaussian and mean curvatures, but the criteria are superfluous because either one of them is sufficient to distinguish between planar and spherical surfaces. However, redundancies of this type are not entirely without utility, since such redundancies make the decision making process more robust. If one criterion fails for some reason, we may still be able to arrive at the correct decision using the other. Hence such redundancies may be desirable to increase the robustness of the decision-making process.

Redundancy Detection and Estimation of Confidence Factors: A connection is considered redundant if the weight associated with it gradually approaches to zero (or a small threshold value) as the learning proceeds. A node (associated with a criterion) is considered redundant if all the connections from the output of this node to other nodes become redundant. Our simulations show that both in the case of uninformative criteria and unreliable criteria, the weights corresponding to all the output connections go to zero. Therefore such nodes (criteria) are eliminated from the structure. The examples in Section 4 illustrate this idea.

Rule Generation: The networks that finally result from this training process can be said to represent rules that may be used to make the decisions. If the final value of the parameter p at a given node is greater than one, the nature of the connective is disjunctive. If the value is less than one, it is conjunctive. Once the nature of the connective at each node is determined, we can easily construct the fuzzy rules that describe the input-output relations. In Section 4 we present some examples of this approach.

4. Experimental Results

In this section, we present some typical experimental results involving both synthetic and real data to show the effectiveness of the proposed automatic rule generation method. The method is shown to generate decision rules that best describe the decision criteria for the classes in each experiment. Figure 1 shows the general 3 layer aggregation network used to generate the rules. The input layer consists of nN number of input nodes where N is the number of fuzzy features or criteria (such as properties and relationships) and n is the number of linguistic levels used to partition each feature. For the hidden layer, there are nN hidden nodes where each node is connected to all but one (i.e., it is connected to n-1) input nodes representing levels within each feature. The top layer fully connects the hidden layer. In the experimental results shown here, we used 5 fuzzy linguistic levels to represent each feature, therefore, each hidden node has 4 connections. Other types of network structures were also tried, however the one described above produced the best results. The target values in the training data were chosen to be 1.0 for the class from which the training data was extracted, and 0.0 for the remaining classes. The feature values were always normalized so that they fall in the range [0,1]. Figure 2 depicts the trapezoidal fuzzy sets used to model the intuitive notions of the five linguistic levels **LOW (L), SOMEWHAT LOW (SL), MEDIUM (M), SOMEWHAT HIGH (SH),** and **HIGH (H).**

Figure 2 : Graphical representations of various fuzzy sets.
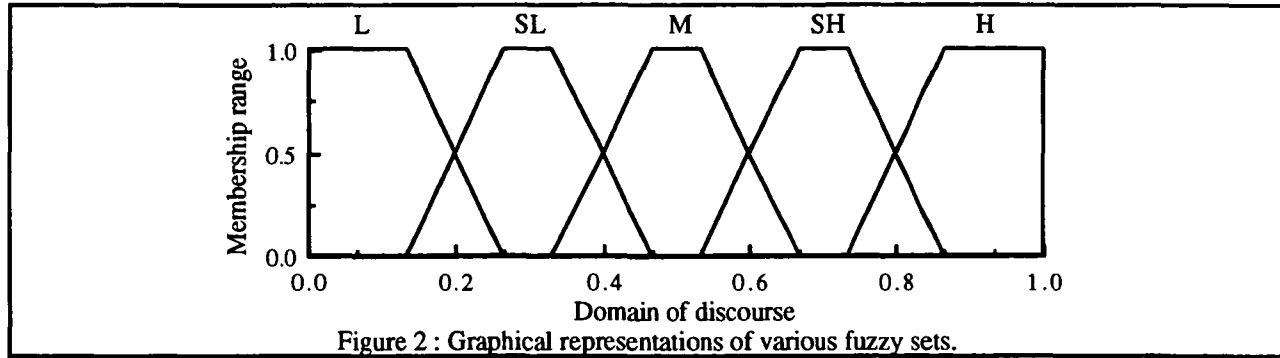
4.1 The ellipse problem

Figure 3 shows the scatter plot of the "ellipse data" after mapping each sample feature into the interval [0,1]. There are 50 samples in each class. The membership values in each linguistic level for each sample is computed using the membership functions shown in Figure 2, and these with the corresponding desired targets are used as training data in the training algorithm described in Section 3. Figure 4 shows the reduced network after training. All connections with weights below a value of 0.01 were considered as redundant. Table 1 shows the final weights (which determine the confidence factors of the rules and criteria) and the p parameter values (which determine the conjunctive or disjunctive nature of the connective) for the specified nodes in Figure 4. Using the properties for the p values obtained, the following rules are generated, as discussed in Section 3.

$$\text{Class 1} = \text{(Feature 1 SL} \vee \text{Feature 1 M} \vee \text{Feature 1 SH)} \wedge$$
$$\text{(Feature 2 SL} \vee \text{Feature 2 M} \vee \text{Feature 2 SH).} \tag{5}$$

In other words, the rule may be summarized as

R_1 : **IF** Feature 1 is **SL** or **M** or **SH** and Feature 2 is **SL** or **M** or **SH**
**THEN** the class is Class 1.

Similarly,

$$\text{Class 2} = \text{(Feature 1 L} \vee \text{Feature 1 H)} \vee \text{(Feature 2 L} \vee \text{Feature 2 H)} \tag{6}$$

and

R_2 : **IF** Feature 1 is **L** or **H** or Feature 2 is **L** or **H**
THEN the class is Class 2.

These rules make sense since the expansion (5) fuzzily covers the 9 inner cells and the expansion of (6) fuzzily covers the outer 16 cells of the plot shown in Figure 3.

Figure 3 : Scatter plot for ellipse data.

Figure 4: Reduced network for the ellipse data.
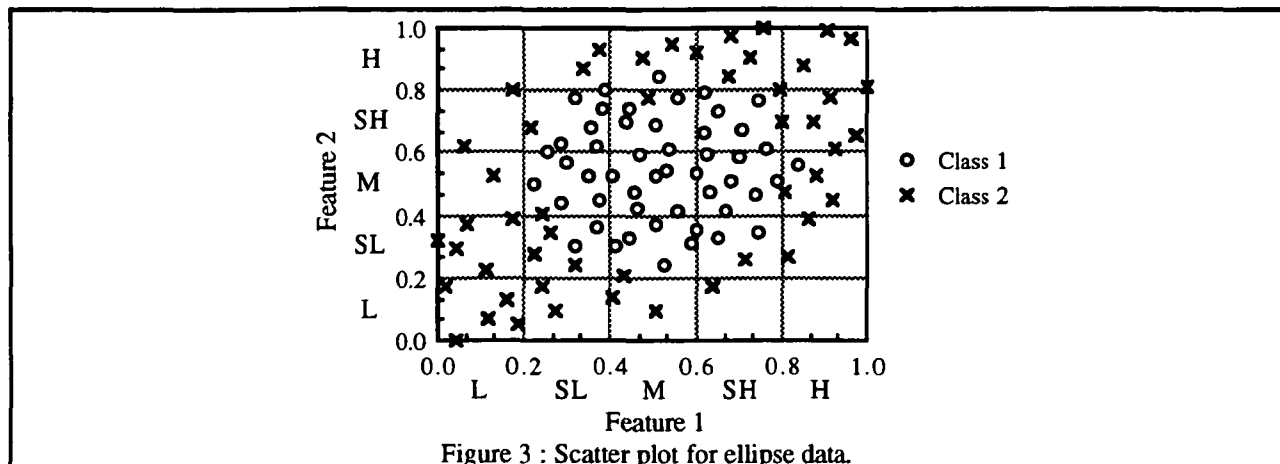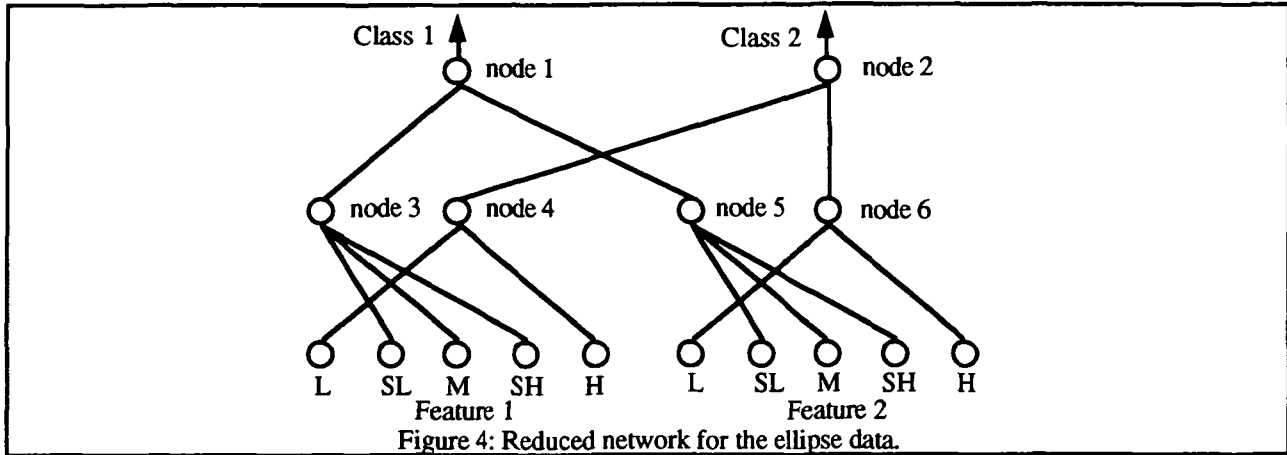
4.2 The natural scene problem

Figure 5(a) shows a 256×256 image of a natural scene and Figure 5(b) shows the scatter plot of the training samples extracted from three different regions (vegetation, sky, and road) in the image. The two features used were the intensity and the position (row number) of the pixels. We used 40 samples from each class. Figure 5(c) shows the reduced network after training. Table 1 shows the final weights and p parameter values for the specified nodes in Figure 5(c). The following rules may be generated from the reduced network.
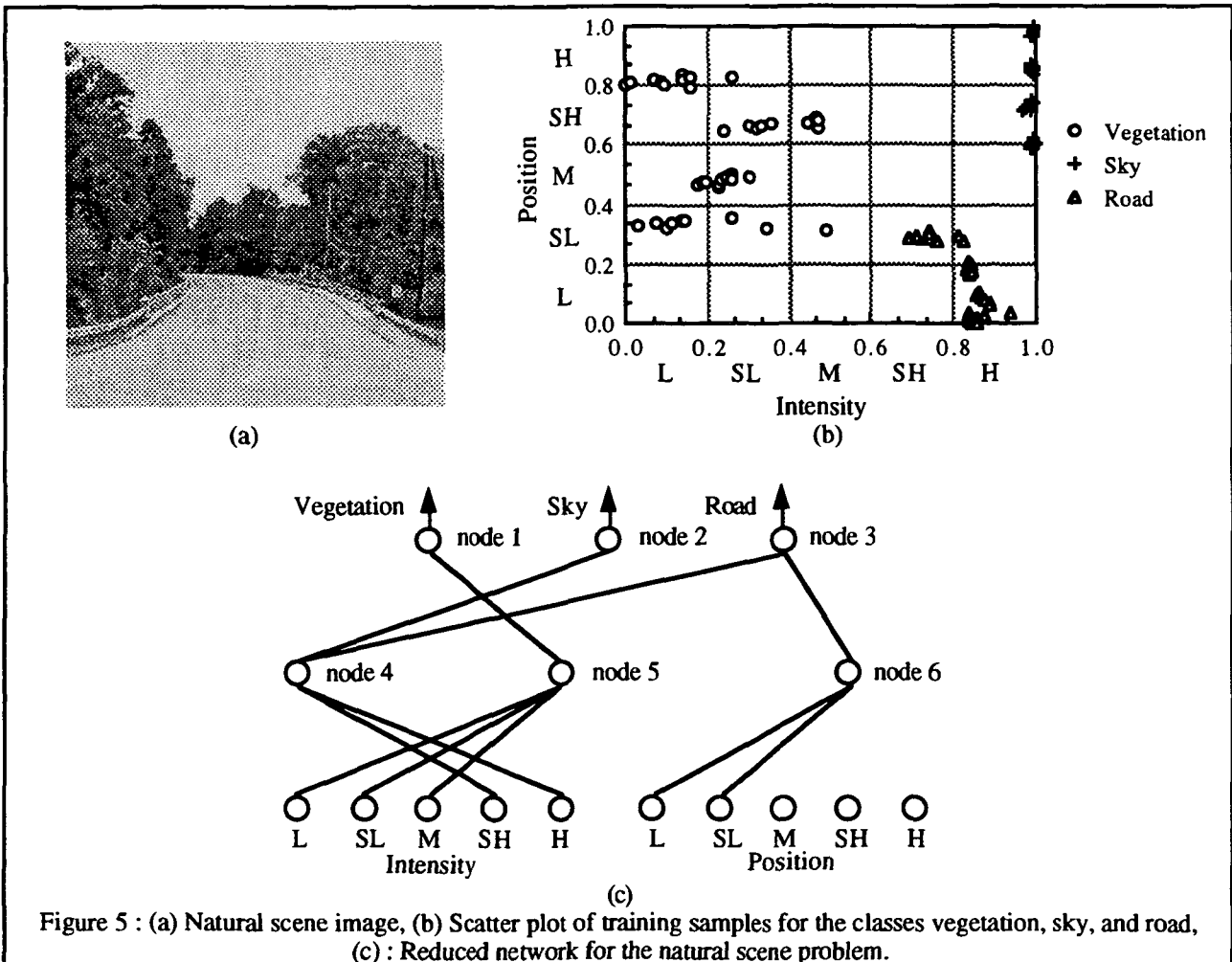

(a)

(b)

(c)

Figure 5 : (a) Natural scene image, (b) Scatter plot of training samples for the classes vegetation, sky, and road, (c) : Reduced network for the natural scene problem.

86

Table 1: Values of weights and parameter p for the ellipse and natural scene problems.

	ellipse problem		natural scene problem	
	weights	p	weights	p
node 1	0.55 0.45	-0.45	0.99	2.28
node 2	0.54 0.46	6.03	0.99	-0.30
node 3	0.19 0.52 0.28	6.15	0.23 0.77	-0.21
node 4	0.52 0.48	6.02	0.20 0.80	2.26
node 5	0.06 0.54 0.40	6.18	0.40 0.41 0.19	5.44
node 6	0.56 0.43	5.87	0.58 0.42	3.50

Class Vegetation = (Intensity **L** ∨ Intensity **SL** ∨ Intensity **M**). (7)

R_{VEG} : **IF** Intensity is **L** or **SL** or **M**
        THEN the class is Vegetation.

Class Sky = (Intensity **SH** ∨ Intensity **H**) (8)

R_{SKY} : **IF** Intensity is **SH** or **H**
        THEN the class is Sky.

Class Road = (Intensity **SH** ∨ Intensity **H**)∧(Position **L** ∨ Position **SL**) (9)

R_{ROAD} : **IF** Intensity is **SH** or **H** and Position is **L** or **SL**
        THEN the class is Road.

In the rule for vegetation, the position feature becomes redundant (i. e., all position weights connected to vegetation drop towards zero). The is reasonable, since the intensity feature clearly separates vegetation from the other classes and the position feature is "unreliable" according to the definition in Section 3. Also, in the rule for sky, the intensity of the sky is more or less uniform and so the intensity feature can clearly distinguish the sky from the other classes. The position feature is again "unreliable". In the rule for road, both position and intensity features play a role. This makes sense since when considering the road, the position feature clearly separates it from the sky and the intensity feature can separate it from the vegetation.

5. Summary and Conclusions

        In this paper, we introduced a new method for automatically generating rules for high level vision. The range of each feature is fuzzily partitioned into several linguistic intervals such as LOW, MEDIUM and HIGH. The membership function for each level is determined, and the membership values for an observed feature value in each of the linguistic levels is calculated using these membership functions. The memberships are then aggregated in a fuzzy aggregation network. The networks are trained with typical data to learn the aggregation connectives and connections that would give rise to the desired decisions. The learning process can also be made to discard redundant features. The networks that finally result from this training process can be said to represent rules that may be used to make the decisions. Riseman et al used similar rules for segmentation and labeling of outdoor scenes, but the weights used in the aggregation scheme were determined empirically [19]. The ability to generate rules that can be used in fuzzy logic

and rule-based systems directly from training data is a novel aspect of our approach. One of the issues that requires investigation is the choice of the number of linguistic levels and its effect on the decision making process.

6. References

1. D. Dubois and H. Prade, "Possibility Theory as a basis for preference propagation in automated reasoning", *Proceedings of the First IEEE Conference on Fuzzy Systems*, March 1992, San Diego, pp. 821-832.
2. D. Driankov, "Fuzzy logic with unless-rules", *Proceedings of the First IEEE Conference on Fuzzy Systems*, March 1992, San Diego, pp. 255-262.
3. T. O. Binford, "Survey of model-based image analysis systems", *International Journal of Robotics Research*, vol. 1, no. 1, Spring 1982, pp. 18-64.
4. D. M. McKeown, W. A. Harvey, and J. McDermott, "Rule-based interpretation of aerial imagery", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, no. 5, 1985, pp. 570-585.
5. J. Freeman, "The modeling of spatial relations", *Computer Graphics and Image Processing*, vol. 4, pp. 156-171, 1975.
6. B. Kosko, *Neural Networks and Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1992.
7. H. Takagi, "Fusion Technology of Fuzzy Theory and Neural Networks - Survey and Future Directions", *Proceedings of the International Conference on Fuzzy Theory and Neural Networks*, IIZUKA-90, pp. 13-17.
8. C. T. Lin, C. S. G. Lee, "Neural-network-based fuzzy logic control and decision system," *IEEE Trans. Computers*, vol. c-40, no. 12, Dec. 1991.
9. L. X. Wang, J. M. Mendel, "Fuzzy basis functions and orthogonal least squares learning," to appear in *IEEE Trans. Neural networks*, September 1992.
10. C. T. Lin, C. S. G. Lee, "Real-time Supervised Structure/Parameter Learning for fuzzy neural network," *Proceedings of the First IEEE International Conference on Fuzzy Systems*, San Diego, CA, March, 1992, pp. 1283-1291.
11. H. Nomura, I. Hayashi, and N. Wakami, "A learning method of fuzzy inference rules by descent method," *Proceedings of the First IEEE International Conference on Fuzzy Systems*, San Diego, CA, March, 1992, pp. 203-210.
12. D. G. Barkhart, P. P. Bonissone, "Automated fuzzy knowledge base generation and tuning," *Proceedings of the First IEEE International Conference on Fuzzy Systems*, San Diego, CA, March, 1992, pp. 179-188.
13. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller - parts I & II," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-20, no. 2, pp. 404-435, 1990.
14. R. Krishnapuram and J. Lee, "Fuzzy-connective-based hierarchical aggregation networks for decision making," *Fuzzy Sets Syst.*, vol. 46, no. 1, pp. 11-27, Feb. 1992.
15. H. Dyckhoff and W. Pedrycz, "Generalized means as a model of compensation connectives," *Fuzzy Sets Syst.*, vol. 14, no. 2, pp. 143-154, 1984.
16. H.-J. Zimmermann and P. Zysno, "Decisions and evaluations by hierarchical aggregation of information," *Fuzzy Sets Syst.*, vol. 10, no. 3, pp. 243-260, 1983.
17. R. Krishnapuram and J. Lee, "Fuzzy-set-based hierarchical networks for information fusion in computer vision," *The Journal of Neural Networks*, vol. 5, no. 2, pp. 335-350, March 1992.
18. R. Krishnapuram, J. M. Keller, and Y. Ma, "Quantitative Analysis of Properties and Spatial Relations of Fuzzy Image Regions", *Proceedings of NAFIPS*, Puerto Vallarta, Mexico, Dec. 1992.
19. E. M. Riseman and A. R. Hanson, "A Methodology for the Development of General Knowledge-Based Vision Systems", in *Vision Systems and Cooperative Computation*, M. A. Arbib (Editor), MIT Press, Cambridge Ma, 1988.