Training of the Freehand Curve Identifier FSCI Using a Fuzzy Neural Network

その他（別言語等）のタイトル	ファジィニューラルネットワークを用いた手書き曲線同定法FSCIの学習最適化
著者	SAGA Sato, MORI Saori, YAMAGUCHI Toru
journal or publication title	Memoirs of the Muroran Institute of Technology
volume	50
page range	1-7
year	2000-11-30
URL	http://hdl.handle.net/10258/129

Training of the Freehand Curve Identifier FSCI Using a Fuzzy Neural Network

その他（別言語等）のタイトル	ファジィニューラルネットワークを用いた手書き曲線同定法FSCIの学習最適化
著者	SAGA　Sato, MORI　Saori, YAMAGUCHI　Toru
journal or publication title	Memoirs of the Muroran Institute of Technology
volume	50
page range	1-7
year	2000-11-30
URL	http://hdl.handle.net/10258/129

Mem. Muroran Inst. Tech.,50(2000)1~7

Training of the Freehand Curve Identifier FSCI Using a Fuzzy Neural Network

Sato SAGA*, Saori MORI* and Toru YAMAGUCHI*

(Accepted 31 August 2000)

This paper demonstrates effectiveness of training of Fuzzy Spline Curve Identifier (FSCI) using a fuzzy neural network. FSCI was proposed as a primitive curve identification system designed to establish a general-purpose freehand interface for computer aided drawing (CAD) systems. It succeeded in distinguishing a freehand drawing into seven kinds of primitive curves which are indispensable for use in CAD. The key was the introduction of a fuzzy reasoning which embodied a strategy to try to find the simplest primitive curves in drawing. A trainable version of FSCI was then proposed, by introducing a structured fuzzy neural network, in order that it would acquire learning ability to adapt itself to individual drawing manner. This paper sets up some experiment on FSCI and demonstrates the effectiveness of the training by evaluating curve class recognition rates. Furthermore, through some considerations on a concrete example of the training, it shows that the introduced fuzzy neural network is informative for us to analyze users' drawing manner and also the identification characteristics of FSCI.

Keywords: Freehand Drawing, Pattern Recognition, Human Interface, Fuzzy System, Neural Network

1 INTRODUCTION

Usual CAD entities are drawn as combinations of seven classes of primitive curves: line, circle, circular arc, ellipse, elliptic arc, closed free curve and open free curve. Accordingly, a general-purpose curve identifier should be required to have a capability to classify a freehand drawing into the seven kinds of primitive curves. However, the shape of a freehand drawing is not enough information to determine curve classes due to the inclusion relations among the primitive curve classes shown in Figure 1: line is a kind of circular arc, circular arc is a kind of elliptic arc, and so on.

The Fuzzy Spline Curve Identifier (FSCI)[1),(3] has overcome the difficulty by utilizing user's drawing manner as well as the curve shape. FSCI was designed to tend to classify roughly drawn curves as simple primitive curves, but carefully drawn curves as complex ones. This implies that a user can intend to draw a rather simple curve by drawing roughly but a rather complex curve by drawing carefully. Experimental results in (4) and (5) showed that the strategy was effective for expert users. However, since the strategy was realized as a fuzzy reasoning with a fixed fuzzy rule set, new users needed quite a little drawing practice to master the characteristics of FSCI. A trainable FSCI was then proposed to adapt itself to each user's characteristics and reduce new user's burden in practice[6]. This was actualized by replacing the fixed fuzzy reasoning in the original FSCI with a common feedforward 3-layer neural network. The learning of neural network carried plasticity into FSCI to improve the curve class recognition rates for the experienced but non-expert users. However, it lost FSCI the explicit representation of the original strategy. In (7), a new version of trainable FSCI was finally proposed by introducing a structured fuzzy neural network into the original FSCI in order that it would acquire learning ability while it would

* Department of Computer Science and Systems Engineering

preserve the original strategy; and its fundamental function was confirmed.

This paper sets up some experiment on FSCI proposed in (7) and demonstrates the effectiveness of the training by evaluating curve class recognition rates. Then, through some considerations on a concrete example of the training, it shows that the introduced fuzzy neural network is informative for us to analyze both users' drawing manner and the identification characteristics of FSCI.

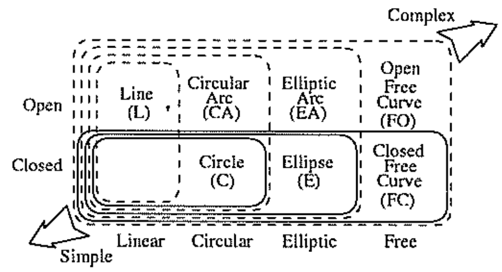


Fig. 1. Inclusion relations among primitive curve classes.

2 OUTLINE OF TRAINABLE FSCI

First of all, this section gives the outline of the trainable version of FSCI proposed in (7).

Given a freehand curve drawn by a user, FSCI performs a sort of fuzzy reasoning to try to identify it as one of the seven kinds of primitive curves, and outputs a fuzzy primitive curve. The fuzzy primitive curve is concretely composed of seven membership grades (which are $\mu(L)$, $\mu(C)$, $\mu(CA)$, $\mu(E)$, $\mu(EA)$, $\mu(FC)$ and $\mu(FO)$) and seven sets of curve shape parameters which are associated with the seven primitive curve classes. It can be also regarded as seven different classes of primitive curve candidates ordered according to the grades.

The introduction of the fuzzy reasoning is essential for FSCI to tell the difference among the seven curve classes. The shape of a freehand curve is not enough information to determine the curve classes because of the inclusion relations shown in Figure 1: strictly speaking, all freehand curves should be categorized into open free curve as long as only the shape is taken into account.

In order to overcome the problem, FSCI utilizes the drawing manner as well as the curve shape. So far as the membership grades are concerned, the schematic process of FSCI is illustrated as shown in Figure 2. First, FSCI performs *fuzzy spline interpolation* and models a freehand curve as a *fuzzy spline curve* which involves vagueness (associated with roughness in drawing) in their positional information. Secondly, it performs *possibility evaluation*, where it estimates linearity, circularity, ellipticity and closedness*2 of the fuzzy spline curve taking account of the vagueness, and outputs four possibility values: p^{Linear}, $p^{Circular}$, $p^{Elliptic}$ and p^{Closed}.

*2We use a term "closedness" to express the degree to which the fuzzy spline curve is closed.

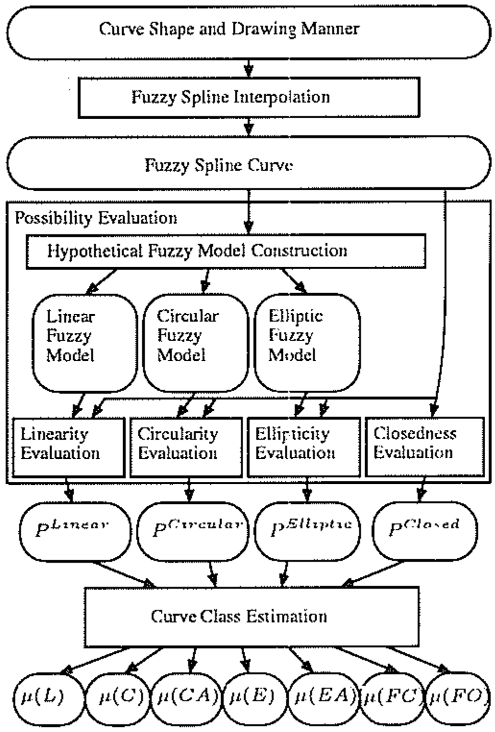


Fig. 2. Schematic process flow by FSCI.

Thirdly, it performs *curve class estimation*, a sort of fuzzy reasoning, where it tries to find the simplest possible primitive curves based on the four possibility values, and outputs seven membership grades: $\mu(L)$, $\mu(C)$, $\mu(CA)$, $\mu(E)$, $\mu(EA)$, $\mu(FC)$ and $\mu(FO)$.

Because even a simple primitive curve can be possibly found in the fuzzy spline curve when it is vague enough, a user is now given a way to let FSCI identify a simple primitive curve. This implies that a user can intend to draw a rather simple curve by drawing roughly but rather complex curve by drawing carefully (see Figure 3).

In the trainable version of FSCI, the curve class estimation process is realized as a fuzzy neural network so that it may be trained and, as a result, FSCI may adapt itself to user's drawing manner.

2.1 Fuzzy Spline Interpolation

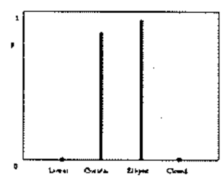
A drawn curve is given to the system as a sequence of a certain number of sampled points p_k and time stamps t_k. However, the sampled points are not always considered to have accurate positional information exactly reflecting the intention of the drawer. In general, the more roughly a curve is drawn, the more vague its positional information will be. From this observation, each sampled point p_k is replaced by a conical fuzzy point model $\tilde{p}_k =< p_k, r_{p_k} >$ shown in Figure 4 (a), where the fuzziness r_{p_k} is generated according to the roughness in drawing. In FSCI, the value of r_{p_k} is simply set as $r_{p_k} = Q \times a_{p_k}$, where a_{p_k} is the acceleration at p_k and Q is a constant value. Then, the fuzzy spline curve that interpolates to the fuzzy points $\tilde{p}_k$ is generated by the method proposed in (1) and (2). The fuzzy spline curve

(a) A rough drawing.

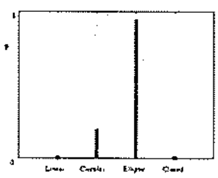
(b) A careful drawing.

(c) Fuzzy spline curve from (a).

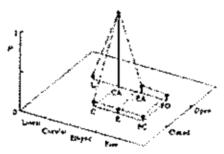
(d) Fuzzy spline curve from (b).



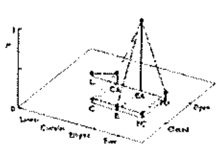
(e) Four possibility values: p^{Linear}, $p^{Circular}$, $p^{Elliptic}$ and p^{Closed} from (c).



(f) Four possibility values: p^{Linear}, $p^{Circular}$, $p^{Elliptic}$ and p^{Closed} from (d).

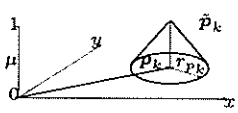


(g) Seven grades: $\mu(L)$, $\mu(C)$, $\mu(CA)$, $\mu(E)$, $\mu(EA)$, $\mu(FC)$ and $\mu(FO)$ from (e).



(h) Seven grades: $\mu(L)$, $\mu(C)$, $\mu(CA)$, $\mu(E)$, $\mu(EA)$, $\mu(FC)$ and $\mu(FO)$ from (f).

Fig. 3. Examples of identification by FSCI.



(a) Conical fuzzy point.

(b) Fuzzy spline curve.

Fig. 4. Fuzzy spline interpolation.

is defined as an extension of an ordinary spline curve and illustrated as a locus of a fuzzy point which travels while changing its vagueness according to the roughness in drawing, as shown in Figure 4 (b). It is utilized as a fuzzy model of the drawing which may involves vagueness.

2.2 Possibility Evaluation[1],[3]

First, FSCI constructs three hypothetical fuzzy models: the linear fuzzy model, the circular fuzzy model and the elliptic fuzzy model. They are obtained as fuzzy Bézier curves[*3] whose parameters are adjusted so that they fit the given fuzzy spline curve as well as possible.

[*3] A fuzzy Bézier curve is defined as a special case of the fuzzy spline curve.

Secondly, each hypothetical fuzzy model is compared with the original fuzzy spline curve and its validity is evaluated by a possibility value: P^{Linear}, $P^{Circular}$ or $P^{Elliptic}$ based on the possibility measure[9]. In other words, the degrees of linearity, circularity and ellipticity of the drawn curve are evaluated by P^{Linear}, $P^{Circular}$ and $P^{Elliptic}$ respectively. Thirdly, the accordance between the fuzzy end points of the fuzzy spline curve is checked and the closedness is evaluated by another possibility value P^{Closed}.

Now, it must be noted that three of the possibility values obtained in this process are always in a fixed order. Namely, P^{Linear} is always less than or equal to $P^{Circular}$, and $P^{Circular}$ is always less than or equal to $P^{Elliptic}$, as shown in Figure 3 (e) and (f). This is because of the inclusion relations among the primitive curve classes.

2.3 Curve Class Estimation

Due to the fixed order among the three possibility values, it is inconclusive to determine the curve class by simply comparing them. In addition, the closedness should be taken into account for FSCI to distinguish between closed primitive curves and open primitive curves (for example, between circle and circular arc). Therefore, FSCI performs the curve class estimation process that is embodied as a fuzzy neural network shown in Figure 5; and calculates the seven membership grades: $\mu(L), \mu(C), \mu(CA), \mu(E), \mu(EA), \mu(FC)$ and $\mu(FO)$ from the four possibility values: P^{Linear}, $P^{Circular}$, $P^{Elliptic}$ and P^{Closed}. In the fuzzy neural network, each min-unit performs min operation that outputs the minimum value. On the other hand, both T_s-units and F_s-units are sigmoid units each of which has a function $S(x) = 1/(1 + e^{-x})$, and the i^{th} sigmoid unit outputs $S(w_i P_i + \theta_i)$, where P_i ($\in$ { P^{Linear}, $P^{Circular}$, $P^{Elliptic}$, P^{Closed} }) is the input to the unit, w_i is the weight factor to the input, and θ_i is the bias term.

Let us see how this fuzzy neural network plays a role of fuzzy reasoning that tries to find the simplest possible curve class. Let us set w_{T_s} (= 6.6) and θ_{T_s} (= −3.3) to w_i and θ_i respectively for all T_s-units as shown in Figure 6 (a); and set w_{F_s} (= −6.6) and θ_{F_s} (= 3.3), for all F_s-units as shown in Figure 6 (b). Then, with this setting, each T_s-unit acts as a fuzzy proposition "P is T_s," where T_s is *fuzzy true* shown in Figure 6 (c); and each F_s-unit acts as a fuzzy proposition "P is F_s," where F_s is *fuzzy false* shown in Figure 6 (d). Considering that the min-unit can be regarded as a logical operator *and*, the fuzzy neural network can be translated into the fuzzy rule set which consists of the seven expressions shown in Figure 7, where $\wedge$ denotes the logical multiplication or the min-operator and the fuzzy truth values shown as membership functions are T_s or F_s respectively. Because the fuzzy rules regarding rather complex curve classes are severer than the ones regarding rather simple curve classes, it is now understood that the the fuzzy neural network embodies the fuzzy reasoning that tries to find

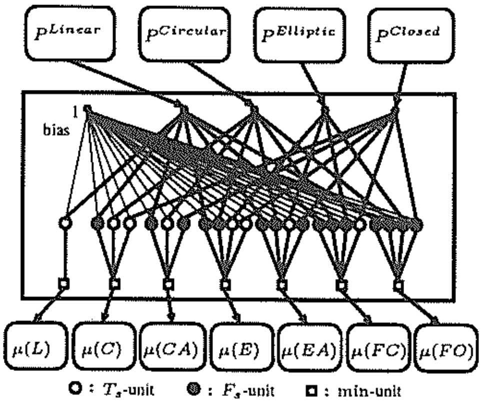
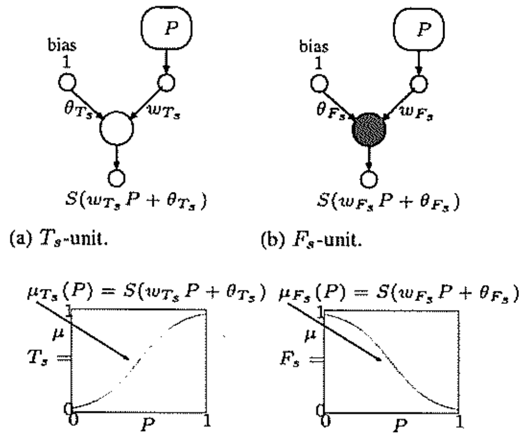


Fig. 5. Curve class éstimation by a fuzzy neural network.



(a) T_s-unit. (b) F_s-unit.

(c) *Fuzzy true* T_s by T_s-unit. (d) *Fuzzy false* F_s by F_s-unit.

Fig. 6. Linguistic truth value by a sigmoid unit.

the simplest possible curve class.

The structure of the fuzzy neural network lets FSCI preserve the basic strategy: "Try to find the simplest possible primitive curves." On the other hand, the learning ability of the neural network makes FSCI trainable, as we discuss in the following section.

3 TRAINING OF FSCI

Given drawings and drawer's intentions about curve classes, the parameters w_i and θ_i of the fuzzy neural network presented in Figure 5 are adjusted so as to adapt FSCI's identification results to the drawer's intentions as much as possible. The inputs to the neural network: p^{Linear}, $p^{Circular}$, $p^{Elliptic}$ and p^{Closed} are calculated from each of the given drawings by the possibility evaluation process following the fuzzy spline interpolation process shown in Figure 2. On the other hand, the desired outputs from the neural network: $\mu(L)$, $\mu(C)$, $\mu(CA)$, $\mu(E)$, $\mu(EA)$, $\mu(FC)$ and $\mu(FO)$ are directly set based on the drawer's intention. (For example, when the drawer's intention is CA, we set 1 to $\mu(CA)$ and 0 to all other grades.) Therefore, the commonly used back-propagation learning algorithm can be simply applied to train the network.

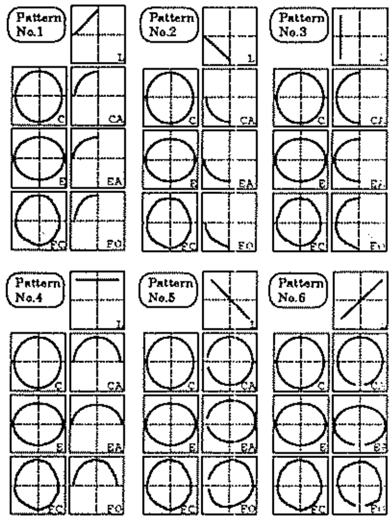


Fig. 8. Presented patterns.

4 EXPERIMENTAL RESULTS OF TRAINING

This section demonstrates the effectiveness of the training of FSCI by evaluating curve class recognition rates, and then examines a concrete example of training in order to show how the fuzzy rule set is adjusted.

4.1 Experimental Conditions

For the experiment, we gathered 840 drawing samples from each of six different users (named A, B, C, D, E and F). Each user was presented with the six kinds of patterns (each of which have seven curve shapes and their corresponding curve classes) shown in Figure 8 in turn; and requested to draw primitive curves similar to the ones in the patterns intending to let FSCI recognize the indicated curve classes. A set of presentation consisted of the six patterns of small size and the ones of large size (that is 12 patterns in total) and ten sets were presented to each user. Out of the ten sets of presentation (that is 840 drawing samples) to each user, seven sets (that is 588 drawing samples) were used for training and the other three sets (that is 252 drawing samples) were used for testing. Because the fuzzy neural network has the explicit representation as a fuzzy rule set, all the training could start with the meaningful initial setting shown in Figure 7.

4.2 Improvement of Curve Class Recognition Rates by Training

Table 1 shows the curve class recognition rates by FSCI with the initial fuzzy rule set in Figure 7 and ones by FSCI with fuzzy rule sets obtained after the training. In the table, the column labeled "1st Candidate" shows the recognition rates regarding the curve classes given the highest grades; "1st–2nd Candidates," the first and second highest grades; and "1st–3rd Candidates," the first through third highest grades. Although we evaluated the curve class recognition rates using the testing samples

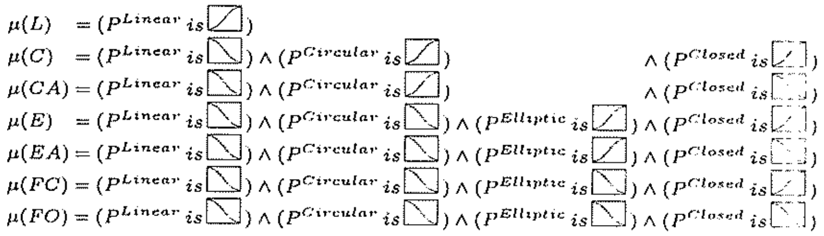


Fig. 7. Initial fuzzy rule set.

Table 1. Curve class recognition rates.

User	Fuzzy Rule Set	Recognition Rates(%) 1st Candidate	1st-2nd Candidates	1st-3rd Candidates
A	Initial	78.869	91.369	94.345
	Trained	80.952	92.560	95.238
B	Initial	78.869	92.262	96.429
	Trained	79.762	92.857	96.726
C	Initial	62.500	80.060	88.095
	Trained	68.155	86.012	92.857
D	Initial	76.786	94.048	97.917
	Trained	83.613	95.536	99.405
E	Initial	91.071	98.512	100.000
	Trained	93.451	99.405	99.702
F	Initial	70.833	87.500	94.048
	Trained	72.024	91.071	96.429

(without using the samples used for training), the results, from all of the six users, demonstrates the improvement of the curve class recognition rates after the training.

4.3 Considerations on a Concrete Example of Training

Let us look at the case of the user D in detail. Figure 9 shows the fuzzy rule set obtained after the training for the user D. Table 2 shows the curve class recognition map by the trained fuzzy rule set, comparing it with the one by the initial fuzzy rule set.

Now, Figure 9 tells us how the fuzzy rule set was adjusted so that it would adapt FSCI to the drawing manner of the user D. When we pay attention to the propositions with P^{Closed} for free curves (that is FC and FO), the *fuzzy true* got milder than the initial *fuzzy true* T_s while the *fuzzy false* got severer than the initial *fuzzy false* F_s. This implies that FSCI was trained so that the user D would easily close free curves. Indeed, it was difficult for the user D to get FC when FC was his intention as shown in Table 2 (a). However, it was improved after the training as shown in Table 2 (b). The drawing samples labeled (a) or (b) in Figure 10 are concrete examples in the case.

On the other hand, when we regard the propositions with $P^{Circular}$ in Figure 9, we find that the *fuzzy true*'s got severer but the *fuzzy false*'s got milder after the training. This means that the fuzzy rule set was adjusted to get severer to C and CA; and, as a result, it came to tend to recognize E or EA rather than C or CA. This tendency obtained after the training considerably improved the curve class recognition rates for E and EA as shown in Table 2, although the tendency slightly disimproved the ones for C. The drawing samples labeled (c) or (d)

Table 2. Curve class recognition maps for user D.

(a) By initial fuzzy rule set.

Intentional Curve Class	Recognized Results (Number of Drawings): L	CA	EA	FO	C	E	FC
L	48	0	0	0	0	0	0
C	0	0	0	0	46	2	0
CA	5	34	1	1	6	1	0
E	0	0	0	0	15	29	4
EA	0	13	24	7	1	3	0
FC	0	0	0	8	4	3	33
FO	0	3	1	44	0	0	0

(b) By trained fuzzy rule set.

Intentional Curve Class	Recognized Results (Number of Drawings): L	CA	EA	FO	C	E	FC
L	48	0	0	0	0	0	0
C	0	0	0	0	42	4	0
CA	2	39	1	1	4	1	0
E	0	0	0	0	9	35	4
EA	0	6	30	9	0	3	0
FC	0	0	0	1	0	4	43
FO	0	2	1	44	0	0	1

in Figure 10 are the cases improved by this effect; (e) or (f), the cases disimproved.

It will be noticed from these examples that the curve class estimation process realized as the fuzzy neural network tells us what was difficult for a specific user to deal with and how the fuzzy rules were adjusted to relieve the difficulties; and this will not only help a user to change one's drawing manner, but will also give us hints for further improvement in the algorithms of FSCI.

5 CONCLUSIONS

This paper gave an outline of a trainable version of FSCI, in which a fuzzy neural network was embedded as the curve class estimation process. Then, experimental results, from six different users, demonstrated that the training of fuzzy neural network improved FSCI in terms of curve class recognition rates. Furthermore, through some considerations on a concrete example of the training, we showed that the fuzzy neural network (which has an explicit expression as a fuzzy rule set) is informative for us to analyze both users' drawing manner and the identification characteristics of FSCI. This is expected

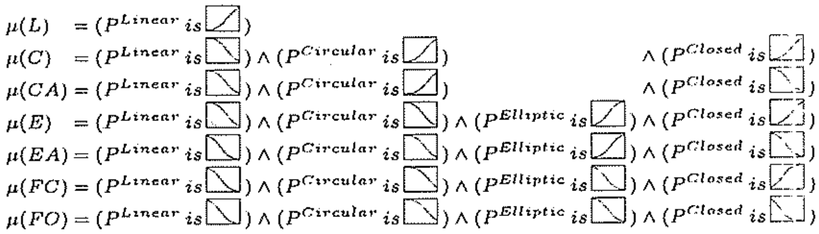


Fig. 9. Trained fuzzy rule set for the user D.

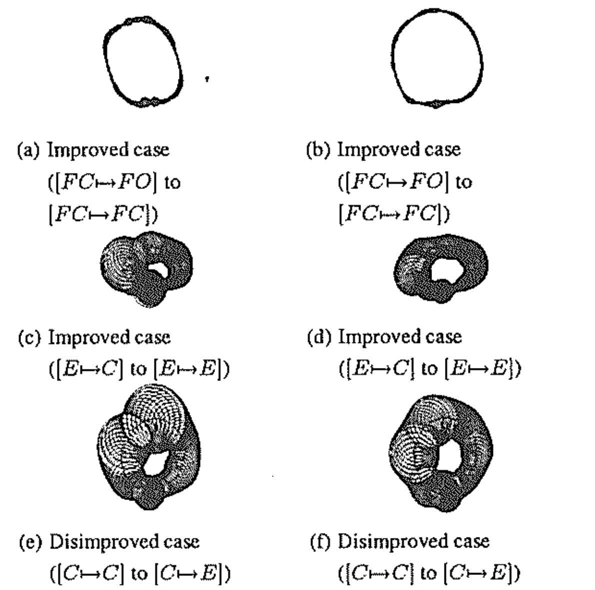


Fig. 10. Drawing samples shown as fuzzy spline curves.

to be helpful for further improvement in FSCI.

REFERENCES

(1) Saga, S. and Makino, H., Fuzzy Spline Interpolation and Its Application to On-Line Freehand Curve Identification, Proc. of 2nd IEEE Int. Conf. on Fuzzy Systems, San Francisco, USA (1993), pp. 1183–1190.

(2) Saga, S., Makino, H. and Sasaki, J., A Method for Modeling Freehand Curves — the Fuzzy Spline Interpolation —, IEICE Trans. (in Japanese), Vol.J77-D-II, 8 (1994), pp. 1610–1619.

(3) Saga, S., Makino, H. and Sasaki, J., The Fuzzy Spline Curve Identifier, IEICE Trans. (in Japanese), Vol.J77-D-II, 8 (1994), pp. 1620–1629.

(4) Saga, S. and Sasaki, J., A Freehand CAD Drawing Interface Based on the Fuzzy Spline Curve Identifier, IPSJ Trans. (in Japanese), Vol.36, 2 (1995), pp. 338–350.

(5) Saga, S., A Freehand Interface for Computer Aided Drawing Systems Based on the Fuzzy Spline Curve Identifier, Proc. of 1995 IEEE Int. Conf. on Systems, Man and Cybernetics, Vancouver, Canada (1995), pp. 2754–2759.

(6) Saga, S. and Seino, N., Trainable Fuzzy Spline Curve Identifier Using a Neural Network, Proc. of International Workshop on Soft Computing in Industry '96, Muroran, Japan (1996), pp. 41–46.

(7) Saga, S. and Mori, S., Trainable Freehand Curve Identifier with a Fuzzy Neural Network, Proc. of 5th European Congress on Intelligent Techniques and Soft Computing, Aachen, Germany (1997), pp. 127–131.

(8) Farin, G., Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide, Academic Press, New York (1998).

(9) Zadeh, L. A., Fuzzy Sets As a Basis for a Theory of Possibility, Fuzzy Sets and Systems, Vol.1, 1 (1978), pp. 3–28.

ファジィニューラルネットワークを用いた手書き曲線同定法 FSCI の学習最適化

佐賀 聡人*, 森 さおり*, 山口 徹*

概要
本論文は，ファジィニューラルネットワークを用いたファジィスプライン曲線同定法 (FSCI) 学習最適化法の効果を実証的に示す．FSCI は CAD システムの汎用的手書きインタフェースを確立すること目的とするプリミティブ曲線同定システムとして提案された．これは，手書き曲線を，CAD に不可欠な 7 種類のプリミティブ曲線のいずれかとして識別し分類することを可能にした．その鍵となったのは描画曲線の中から最も単純なプリミティブ曲線を見出そうとする戦略を実現するファジィ推論の導入であった．その後，FSCI にユーザ個人毎の描画動作の違いに適応する学習能力を付加することを目的として，構造化ファジィニューラルネットワークを導入した最適化可能な FSCI が提案された．本論文ではこの FSCI に対する実験を行い，その曲線クラスの認識率を評価することにより学習最適化の効果を実証的に示す．さらに，学習最適化の具体的な結果例についての考察を通して，ファジィニューラルネットワークの導入がユーザの描画動作の解析はもとより FSCI の同定特性を解析するためにも有益な情報をもたらすことを示す．

キーワード：手書き描画、パターン認識、ヒューマンインタフェース、ファジィシステム、ニューラルネットワーク

* 情報工学科