NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

ISBN 0-315-56074-6

Canada

PATTERN RECOGNITION BASED ON ENTROPY ANALYSIS AND SHAPE TRANSFORMATIONS

Yuan Yan Tang

A Thesis
in
The Department
of
Computer Science

Presented in Partial Fulfillment of the Requirements
For the Degree of Doctor of Philosophy
Concordia University
Montreal, Quebec, Canada

February 1990

ABSTRACT

PATTERN RECOGNITION BASED ON ENTROPY ANALYSIS AND SHAPE TRANSFORMATIONS

Yuan Yan Tang, Ph. D.

Concordia University, 1990

Two new models, called multiple level information source (MLIS) and entropy-reduced transformation (ERT), are proposed to analyze systematically the changes in entropy which occur in the different phases of pattern recognition. Also they provide a systematic way to design a pattern recognition system.

Image transformation is one of the most important entropy transformations. It has two completely different characteristics: entropy increased and reduced characteristics. Both of these properties are frequently used in image processing and pattern recognition.

In this thesis, several new theorems on image transformation are presented and proved, and some new algorithms are also proposed. These algorithms can perform the mapping and filling at the same time while preserving the connectivities of the original image. As a result, the transformations become more consistent and accurate. The new algorithms can handle some problems which are

produced by other algorithms. A series of experiments are also conducted to verify the performance of the proposed algorithms.

Variance in size and orientation is one of the common intrinsic uncertainties in pattern recognition. To solve this problem, this thesis proposes a new method called Transformation-Ring-Projection (TRP). In this way, image transformation technique is employed to reduce the entropy which is produced by the variance in size and position. The Ring-Projection scheme is developed to cut down the orientation uncertainty. This method requires only simple and regular operations. Apart from being highly accurate, another merit of the proposed algorithm is that it can be implemented by parallel techniques. In this thesis, many experimental results obtained from a large data set including Chinese characters, Roman letters and numerals fully support the algorithm developed for TRP.

Nonlinear shape distortions, such as perspective projection distortions, produce a considerable uncertainty in computer vision, robot vision and recognition of motion. The correction of these distortions is a most difficult and challenging topic. This thesis presents an approximation of some perspective projection distortions using the shape transformation theory under certain conditions. In this way, bilinear, bi-quadratic and bi-cubic transformations are used to model the nonlinear shape distortion problems. Some useful algorithms are presented. Also inverse shape transformations are used in nonlinear restorations.

Dedicated to my parents

Mr. Yong Xi Tang

Ms. Shu Yu Wei

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor, Professor C. Y. Suen for his guidance, advice, and support during the research of this thesis. Without his inspiration and encouragement, this work could not be finished.

I wish to thank Professors T. D. Bui, Z. C. Li, R. Shinghal, T. Kasvand and H. D. Cheng, and Dr. J. Lu of the Civil Engineering Department for their assistance, suggestions and comments. I would like to express my appreciation to Professor Z. C. Li of the CRIM and Professor Y. H. Wang of the Mathematics Department for checking the mathematical formulas and proofs in this thesis. I am also thankful to C. L. Yu and W. Wong for their help in the experiments. I would acknowledge also those people in the Department of Computer Science and Computer Center of Concordia University, who provide me with the stimulating environment during my study and research. I am deeply grateful to my family for their immeasurable love, meticulous care and wholehearted support.

I take special pleasure in thanking Concordia University which awarded me the Concordia University Graduate Fellowship to enable me to study at this University.

Finally, we would like to acknowledge the research grants received from the Natural Sciences and Engineering Research Council of Canada and the Ministry of Education of Quebec.

Table of Contents

List of Figures and Tables

CHAPTER 1

INTRODUCTION

1.1 ENTROPY AND PATTERN RECOGNITION

Entropy is a basic principle in pattern recognition as well as in the thermodynamics and communication theory [Tou74, Watana69]. The concept of entropy was originally used to formulate the Second Thermodynamical Principle by Clausius in 1865 [Clausi87]. The definition of entropy was given purely in terms of thermodynamical quantities. The idea is briefly described below:

The increment of entropy dH_Q is defined as

$$dH_Q = \frac{\Delta Q}{T}$$

where ΔQ is the amount of heat energy given to the system at absolute temperature T.

Towards the end of the century, efforts were made by such physicists as Boltzmann, Gibbs, and Maxwell to reinterpret the entropy from the standpoint of the atomistic view of the physical world [Boltzm96, Gibbs02, Maxwel67]. As a result, entropy is defined as the following:

$$H = \log (\textit{probability})$$

or

$$H = -\sum p_i \log p_i \ ,$$

where p_i is probability. But, it is only used in thermodynamics.

Until 1932 this concept gained independence from thermodynamics owing to Von Neumann's effort to use so called microscopic entropy to demonstrate the irreversibility of the process of the physical observation [Von32]. The first paper which used this quantity as a measure of the structure of a system was written in 1939 by S. Watanabe to discuss the properties of nuclear matter [Watana39]. In 1948, C. E. Shannon successfully set up the modern communication theory by using the concept of information entropy [Shanno48a, b], in which the entropy function was considered as a measure of uncertainty.

However, this concept was not widely used in pattern recognition until the last quarter of this century [Guiasu71, Landa62a, b, Watana64, 69a, b].

(1) The measure of interdependence in terms of the finite discrete entropy was introduced and studied by S. Watanabe [Watana69]. According to this idea, a strategy of classification was established. The basic principle is presented briefly in the following paragraph.

Pattern recognition starts with an observation of selected, say n, variables and ends with a single binary variable which decides whether or not a particular sample belongs to a certain class. There exist many steps in which a set of patterns is decomposed into several subsets to achieve the goal set above. Each such step reduces the interdependence between the samples which belong to different classes. The measure of interdependence can be expressed as

$$\Theta(W_S{}^i ; W_{S_1}{}^i, W_{S_2}{}^i, \cdots, W_{S_r}{}^i) = \sum_{i=1}^{r} H(W_{S_i}{}^i) - H(W_S{}^i)$$

where

(a) $W_S{}^i$ is a target set of patterns, $W_S{}^i = \{ W_{S_1}{}^i, W_{S_2}{}^i, \cdots, W_{S_r}{}^i \}$. It has been decomposed into some subsets, i.e. $W_{S_1}{}^i, W_{S_2}{}^i, \cdots, W_{S_r}{}^i$;

(b) $\sum_{i=1}^{r} H(W_{S_i}{}^i)$ is the sum of entropies of each subsets, and $H(W_S{}^i)$ is the total entropy of target set of patterns.

(2) L. N. Landa proposed a strategy of recognition giving the "most rational" algorithm of recognition and applying it to the problem of recognition of sentences in Russian syntax [Landa62a]. A similar approach was independently given by P. M. Lewis in the same year [Lewis62, Guiasu68, 71]. The main concept of this entropic algorithm will be described below.

According to the entropic algorithm of recognition, it is necessary at every moment to choose and to observe such a feature F_ξ which carries the largest amount of, from the features extracted. In other words, this process eliminates the largest degree of uncertainty. The amount of uncertainty which is removed by the observation of the feature F_ξ will be given by the entropy

$$\begin{aligned} H_\xi &= H(F_\xi) \\ &= - \sum_{i=1}^{r_\xi} P_\xi (i) \log P_\xi (i) \end{aligned}$$

In the light of the entropic algorithm, we shall choose and observe the feature F_{ξ_0}

such that

$$H_{\xi_0} = \max_{1 \leq \xi \leq N} H_\xi ,$$

where N is the number of possible characteristics we choose.

(3) The minimum-entropy approach to feature selection given in [Tou74, 67, Watana64, 67] is due to J. T. Tou, S. Watanabe and R. P. Heydorn. The entropy concept was used as a suitable criterion in the selection of optimum features. Features which reduce the uncertainty of a given situation are considered more informative than those which have the opposite effect. If one views entropy as a measure of uncertainty, a meaningful feature selection criterion is to choose the feature which minimizes the entropy of the pattern classes under consideration. Since this criterion is equivalent to minimizing the dispersion of the various pattern populations, it is reasonable to expect that the resulting procedure has clustering properties. In [Tou74], based on the assumption that the pattern classes under consideration are normally distributed, a linear transformation matrix T_L which operates on the pattern vectors to yield new vectors of lower dimensionality was determined such that

$$F^m = T_L \, F^n$$

where F^n is an n-vector, F^m is an m-vector of lower dimensionality than F^n.

When the assumption of normal distribution is not valid, the method of orthogonal expansion such as Fourier, Karhunen-Loeve expansions offered an

alternative approach to select the features. [Watana64, 67] made use of the Karhunen-Loeve (K-L) expansion in carrying out feature selection. The principal advantage of this expansion is that it does not need to know the various probability densities. In addition, the K-L expansion possesses a couple of optimal properties which can be used as meaningful criteria for feature extraction.

(4) The minimum entropy principle used as one of the general heuristics for pattern recognition given in [Watana81] is due to S. Watanabe (1981). His comprehensive textbook on pattern recognition published in 1985 was written with the spirit of this heuristic principle [Watana85]. In 1988, S. Watanabe and T. Kaminuma conducted a survey on different methods based on the heuristic principle of entropy minimization, and introduced a brand new algorithm IMPRL which decides the class-affiliation of a new input pattern when multi-class paradigms are given in advance [Watana88].

(5) In 1984, C. Y. Suen and Q. R. Wang first designed successfully a decision tree based on entropic theory to recognize Chinese characters [Wangq84]. A new clustering algorithm called ISOETRP has been developed. Several new objectives have been introduced to make ISOETRP particularly suitable to hierarchical pattern classification. These objectives are :

(a) Minimizing overlap between the various groups of pattern classes;

(b) Maximizing entropy reduction;

(c) Keeping balance between these groups.

The overall objective to be optimized is

$$M_G = \frac{H_R}{F\ (\ \sigma_H\ )}$$

where M_G is the " Gain ", H_R presents entropy reduction, and $F\ (\ \sigma_H\ )$ denotes a function of the overlap.

This tree-like discriminator has been used in the recognition of Chinese characters printed in different fonts by Y. Y. Tang, C. Y. Suen and Q. R. Wang in 1984 and 1986 [Tang84, Suen86].

(6) 1988, the first paper which proposed entropy-reduced transformation model for pattern recognition was published by Y. Y. Tang et al [Tang88] and subsequently in [Tang89a, Qu88] which are covered in this thesis.

This thesis presents a new theory of pattern recognition dealing with the principle of entropy and information theory. In order to explain these ideas clearly and conveniently, the elementary concept of entropy will be introduced.

1.2 UNCERTAINTY AND ITS MEASUREMENT - ENTROPY

One of the most important contributions of Shannon's theory is a quantitative measure of the amount of information supplied by a probabilistic experiment [Jeline68, Khinch57, McMill62]. In this concept the amount of information is inversely proportional to the amount of uncertainty, i.e. the information obtained is equal to the removed uncertainty. [Shanno48a, b] made the first consistent attempt towards the measurement of such difficult and yet abstract notions as information and uncertainty.

Definition 1.1

Let (Ω , β , P) be a finite probability space. Where Ω is a set, and some of the subsets of Ω construct a Borel field (σ field), denoted by β. The ordered pair (Ω , β) is called the measure space. P is a set function defined on the Borel set β such that $\beta \rightarrow [0,1]$, i.e. the domain of P is $D_P = \beta$ and codomain of P is $G_P = [0,1]$. This finite probability space must hold the following conditions:

(1) $P (\Omega) = 1$;

(2) $\forall_{A \in \beta} \ (0 \leq P (A) \leq 1)$;

(3) For $\forall_{A_i \in \beta} \ ((A_i \cap A_j = \Phi) \cap (i \neq j) \cap (i,j = 1,2, \cdots))$ we have

$$P \left(\sum_{i=1}^{n} A_i \right) = \sum_{i=1}^{n} P(A_i) .$$

The P is called probability measure defined on the β ; The elements A_1 ,

$A_2, \cdots A_i, \cdots A_j, \cdots$ of β are called events; The $P(A_i)$ is the probability of event A_i [Stark86] .

Let us consider a set of events A having n possible events $A_1, A_2, \cdots, A_n$ with the respective probabilities $p_1, p_2, \cdots, p_n$, satisfying the conditions

$$p_i \geq 0 \qquad (i = 1,2,\ldots,n),$$

$$\sum_{i=1}^{n} p_i = 1 .$$

We will denote also the probability of the event A_i by $p(A_i)$. The set of events A can be represented in the following forms

$$A = \begin{bmatrix} A_1 & A_2 & \cdots & A_n \\ p_1 & p_2 & \cdots & p_n \end{bmatrix}$$

$$= \begin{bmatrix} A_1 & A_2 & \cdots & A_n \\ p(A_1) & p(A_2) & \cdots & p(A_n) \end{bmatrix}. \tag{1.1}$$

Of course, such representation contains an amount of uncertainty about the particular event which will occur if we perform the experiment. We can see that this amount of uncertainty essentially depends on the probabilities of the possible events of the experiment. For example, we consider two simple forms

1) $$\begin{bmatrix} A_1 & A_2 & A_3 & A_4 \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

and

2) $$\begin{bmatrix} A_1 & A_2 & A_3 & A_4 \\ \frac{5}{8} & \frac{1}{8} & \frac{1}{8} & \frac{1}{8} \end{bmatrix}.$$

In the first case we can not predict which particular event will occur, while in the second case we may say that event A_1 may possibly occur. Therefore, it is obvious that the first case contains more uncertainty than the second one. How to measure the uncertainty? We will see that the entropy can serve as good measure of the uncertainty of the form represented by Eq. (1.1).

Definition 1.2

Let us consider a finite probability distribution

$$p_i \geq 0 \qquad (i = 1,2,\ldots,n),$$

$$\sum_{i=1}^{n} p_i = 1 .$$

The corresponding **entropy** [Shanno48a, b] is the quantity

$$H_n = H_n(p_1, p_2, \cdots, p_n) = -\sum_{i=1}^{n} p_i \log p_i . \tag{1.2}$$

The logarithms can be taken with respect to an arbitrary base greater than unity. If we take 2 as the base we will write $\log_2$. Then the unit of uncertainty will be " bit ". If we take e as the base, we will write $\log_e$, and the unit of uncertainty will be " nat ". Similarly, $\log_3$ and " tet " for the base 3, $\log_{10}$ and " det " for the base 10, etc. As the logarithm with base 2 is commonly used in informa-

tion systems, in this thesis we take the base 2, and simplify $\log_2$ by log.

The entropy has a number of properties which we might expect from a reasonable measure of uncertainty in a probabilistic experiment. The quantity $H_n (p_1, p_2, \cdots, p_n)$ is interpreted either as a measure of uncertainty or as a measure of information. Both interpretations are justified [Shanno48a, b]. In fact, the difference between these two interpretations is whether we imagine ourselves at the moment before carrying out an experiment whose n possible events have the probabilities $p_1, p_2, \cdots, p_n$, in which case the entropy $H_n (p_1, p_2, \cdots, p_n)$ measures our uncertainty concerning the event of the experiment. Or we imagine ourselves after the experiment has been carried out, in this case the entropy $H_n (p_1, p_2, \cdots, p_n)$ measures the amount of information we got from the experiment.

It is necessary to discuss the properties which make entropy a measure of uncertainty. The basic properties are listed in Section 1.3.

1.3 BASIC PROPERTIES OF ENTROPY

Property 1 - Symmetry [Zhou83]

$$\begin{aligned} H_n\,(\,p_1,\,p_2,\,p_3,\,\cdots,\,p_n\,) &= H_n\,(\,p_2,\,p_1,\,p_3,\,\cdots,\,p_n\,) \\ &= H_n\,(\,p_3,\,p_2,\,p_1,\,\cdots,\,p_n\,) \\ &= \cdots\cdots \\ &= H_n\,(\,p_n,\,p_{n-1},\,p_{n-2},\,\cdots,\,p_1\,)\,; \end{aligned} \tag{1.3}$$

This means that the change of the order of elements p_i does not affect the entropy.

Property 2 - Non-negative [Zhou83]

$$H_n\,(\,p_1,\,p_2,\,\cdots,\,p_n\,) \geq 0\,; \tag{1.4}$$

Property 3 - Certainty [Zhou83]

If $p_{i_0} = 1$ and $p_i = 0$ $(\,1 \leq i \leq n\;;\; i \neq i_0\,)$ then

$$H_n\,(\,p_1,\,p_2,\,\cdots,\,p_n\,) = 0\,; \tag{1.5}$$

This is just the case where the event of the experiment can be predicted beforehand with complete certainty, so that there is no uncertainty on the events.

Property 4 - Extension [Zhou83]

$$\lim_{\epsilon \to 0} H_{n+1}\,(\,p_1,\,p_2,\,\cdots,\,p_n,\,\epsilon\,) = H_n\,(\,p_1,\,p_2,\,\cdots,\,p_n\,) \tag{1.6}$$

The extension of entropy illustrates that the number of possible events can be increased, but the events with small probabilities could be omitted.

Property 5 - Additive [Zhou83]

$$\begin{aligned} H_{n.n} & (p_1 p_{11}, p_1 p_{21}, \cdots, p_1 p_{m1}; \\ & p_2 p_{12}, p_2 p_{22}, \cdots, p_2 p_{m2}; \\ & p_n p_{1n}, p_n p_{2n}, \cdots, p_n p_{mn}) \\ = & H_n (p_1, p_2, \cdots, p_n) \\ + & \sum_{i=1}^{n} p_i \cdot H_{m\ i} (p_{1i}, p_{2i}, \cdots, p_{m\ i}) \end{aligned} \tag{1.7a}$$

where

$$\sum_{i=1}^{n} n_i = 1, \qquad p_i \geq 0$$

$$\forall_i \sum_{j=1}^{m} p_{ji} = 1, \qquad p_{ji} \geq 0$$

Particularly, let (X, Y) be two dimensional random vector, and X and Y have probabilities p_i and q_i respectively,

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ p_1 & p_2 & \cdots & p_n \end{bmatrix},$$

$$Y = \begin{bmatrix} y_1 & y_2 & \cdots & y_m \\ q_1 & q_2 & \cdots & q_m \end{bmatrix}.$$

And suppose that they are independent each other. We have

$$H (X, Y) = H (X) + H (Y) \tag{1.7b}$$

or

$$H_{mn} = H_n (p_1, p_2, \cdots, p_n) + H_m (q_1, q_2, \cdots, q_m) \qquad (1.7c)$$

where

$$\sum_{i=1}^{n} p_i = 1, \qquad p_i \geq 0$$

$$\forall_i \sum_{i=1}^{m} q_i = 1, \qquad q_i \geq 0$$

This property is very important, since it makes the information which we received at different times can be added simply.

Property 6 - Limitation [Zhou83]

For any probability distribution

$$p_i \geq 0 \quad (i = 1, 2, \cdots, n), \quad \sum_{i=1}^{n} p_i = 1$$

we have

$$H_n (p_1, p_2, \cdots, p_n) \leq H_n (\frac{1}{n}, \frac{1}{n}, \cdots, \frac{1}{n}). \qquad (1.8)$$

Property 6 shows us that Shannon's entropy assumes its largest value for just the uniform probability distribution.

1.4 PRIMARY PRINCIPLE OF INFORMATION SYSTEM

There are a lot of common characteristics in both pattern recognition systems and communication systems. In this thesis, new theories will be developed using the mathematical theory of communication systems. In order to explain conveniently the essential and basic ideas in the following chapters, this section will present the primary principles of communication systems.

A typical communication system consists of essentially five parts:

(1) An **information source** which produces a message or sequences of messages to be communicated to the receiving terminal. Each message is a sequence of letters or signals. Of course, the word " letter " must be understood in the abstract form.

(2) A **transmitter** which operates on the message in some way to produce a signal suitable for transmission over the channel.

(3) The **channel** is merely the medium used to transmit the signal from the transmitter to the receiver.

(4) The **receiver** ordinarily performs the inverse operation of that done by the transmitter, it reconstructs the message from the sequence of signals.

(5) The **destination** is the person, or object, or entity, for whom the message is

intended.

The information source selects a desired message out of a set of messages. In the process of transmitting the signals, it is an unfortunate characteristic that certain things not intended by the information source are added to the signals, having as consequences the appearance of the error in transmission. All these changes of the signals may be called " noise ".

A communication system is symbolically represented in Fig. 1.1 (a).

In comparison with a pattern recognition system shown in Fig. 1.1 (b), we might find some similarities between these two systems as shown roughly in Fig. 1.1.

Now we pay our attention to information source. The information source can be classified into two types : (1) discrete information source, and (2) continuous one. Referring to the pattern recognition system, we discuss the discrete case here. In probability theory of the transmission of information, the output of the information source is regarded as a random process. The statistical structure of this process constitutes the mathematical characterization of the respective information source. In the discrete case, the information source corresponds to a random process with a discrete set which is either a function of the time or the space or both of them. For example,

a) For speech signals, the information source is regarded as a function of the time I_S (t);

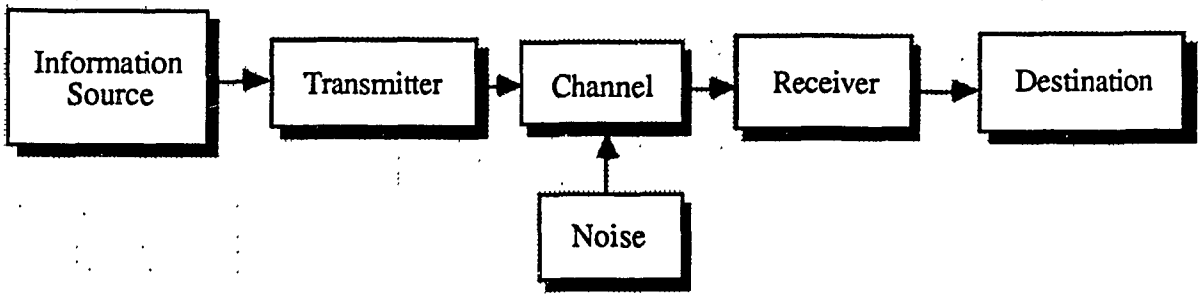
b) For two-dimensional images such as picture, character, etc. the information source is considered as a function of the space I_S (x_0, y_0);

c) Similarly, I_S (x_0, y_0, z_0) can be used for 3-dimensional image;

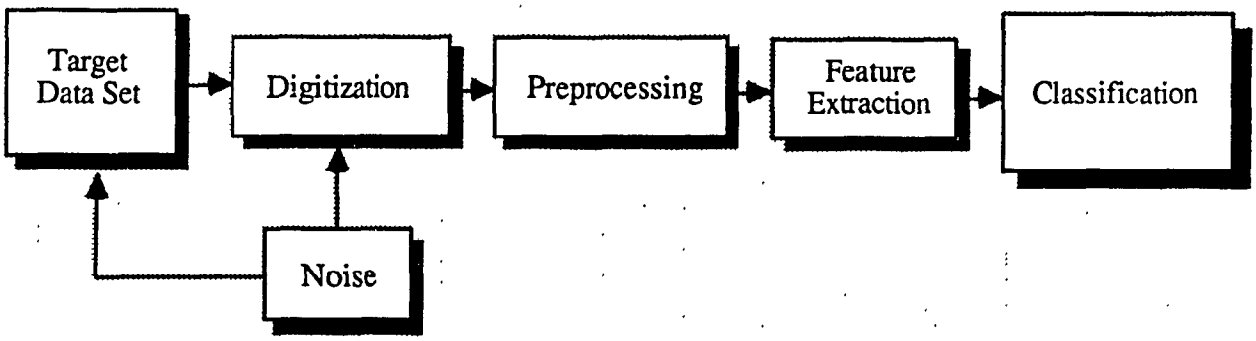
d) For moving 2-dimensional patterns, the information source is said a function of both time and space I_S (x_0, y_0, t) ;

e) Similarly, For moving 3-dimensional objects, the information source is I_S (x_0, y_0, z_0, t) .

For more details about information theory and communication system, please refer to references [Bell62, Dobrus72, Feinst54, McMill54].



(a) Communication System



(b) Pattern Recognition System

Fig. 1.1 Communication System and Pattern Recognition System

1.5 OBJECTIVES AND ORGANIZATION OF THE THESIS

As mentioned above, a lot of researchers have already investigated the use of entropy in pattern recognition for more than 2 decades, and they have developed many theories and methods to explain and solve problems in this area. But neither of them did analyze systematically the uncertainty which exists in each part of the recognition system.

In recent years many new topics such as computer vision, robot vision, the recognition of motion, etc. have evolved. They bring along more and more complex and difficult pattern recognition problems. Also the pattern recognition system itself becomes more and more complicated. In order to investigate efficiently such difficult problems and such complicated systems, this thesis proposes a systematic way of analyzing the uncertainty in the different stages of a pattern recognizer. It proposes a systematic scheme to design an appropriate pattern recognition system based on the proposed new models.

Since this thesis is concerned with the principle of entropy and information theory, in order to explain clearly the proposed ideas, the elementary concepts about entropy and information theories are introduced in Chapter 1.

Based on Shannon's information theory, in Section 2, we first build a new theoretical model called Multiple-level Information Source (MLIS) model to analyze the internal structures of various information sources. Although the number of levels in an MLIS formed by a pattern recognition system is problem-

oriented, for a typical recognition system there are four levels in this information source, i. e. IS1 to IS4, and they can be divided into two categories : entropy-reduced and entropy-increased. Also a new theorem is presented and proved to describe the necessary and sufficient conditions to determine the category of an MLIS. Any practical pattern recognition system can be considered as an MLIS by distinguishing the characteristics of different factors. Chapter 2 also analyzes the various factors in the process of pattern recognition and presents the basic ideas on the mapping of a pattern recognition system into an MLIS. In general, an increase in the entropy in the different parts of a pattern recognition system can be attributed to three factors : a) intrinsic distortion arising from the characteristics of the target pattern set, b) distortions due to the transducer and digitization, and c) information loss attributed to the feature extraction process. This chapter shows that a theoretical analysis of a pattern recognition system can be modeled by an MLIS, and in MLIS the entropy in the different parts can be addressed. Consequently, the task of a pattern recognition system can be regarded as a conversion of an entropy-increased MLIS into an entropy-reduced one.

To perform this conversion, a theoretical framework called Entropy-Reduced Transformation (ERT) model is developed and described in Chapter 3. Two important properties of the ERT i. e. the cascade and parallel properties are presented and proved in Theorems 3.1 and 3.2. Based on these properties an algorithm is proposed to build a practical pattern recognizer in a systematic manner.

Image transformation as one of the most important entropy transformations

has two completely different characteristics: entropy increased and reduced properties which have been used in many image processing and pattern recognition problems. In Chapter 4, both linear and nonlinear transformations are presented, some new algorithms are developed. These algorithms can perform the mapping and filling at the same time while preserving the connectivities of the original image. In the proposed algorithms, complicated operations are not needed, resulting in not only a speed up in computation, but also a more meaningful and accurate filling process.

Chapters 5 and 6 are devoted to the removal of uncertainty from the first level and the second level of the multiple-level information source respectively.

One of the algorithms used to remove uncertainty from the first level of MLIS, which is called Transformation-Ring-Projection (TRP), is developed in Chapter 5 to handle the size-rotation problem.

Nonlinear shape distortions, such as some perspective projection distortions, produce a considerable uncertainty in the second level of the information source in the area of computer vision, robot vision and motion. Chapter 6 provides an approximation to handle the perspective projection distortions using the shape transformation models. Several algorithms eliminating uncertainty from the second level of MLIS are presented in Chapter 6 to restore the original shape which has undergone these nonlinear distortions.

In order to illustrate the process of converting an entropy-increased MLIS into an entropy-reduced one by the ERT, Chapter 7 presents two applications of

designing the integrated discriminator, which includes many interesting practical engineering algorithms. The experimental results obtained from a large set of data and the restoration of distorted patterns support the theory developed for MLIS and ERT and indicate that they provide an efficient way to solve many difficult problems.

Conclusions of this thesis are given in the last Chapter. It indicates that, similar to other information processing systems, entropy reduction plays a major role in every step of a pattern recognition system. Based on this principle and the use of MLIS and ERT models we may develop other new and efficient methods to tackle more complicated problems.

Finally, it is worth mentioning that some materials from most of the chapters of this thesis have appeared in our earlier works, and in the following publications:

Chapter 2 and 3:

1. Tang, Y. Y., Y. Z. Qu and C. Y. Suen "Entropy-reduced transformation approaches to pattern recognition of complex data set," *Proc. Int. Workshop on Computer Vision (IAPR),* pp. 347 - 350, Tokyo, Japan, Oct. 1988.

2. Qu, Y. Z., Y. Y. Tang and C. Y. Suen, "Entropy-reduced transformation (I) : Theoretical analysis and application in pattern recognition," *Proc. IEEE Int. Computer Science Conf.,* pp. 486 - 493, Hong Kong, Dec. 1988.

Chapter 4:

1. Li, Z. C., Y. Y. Tang, T. D. Bui and C. Y. Suen, "Shape transformation models and their applications in patterns recognition," in press, *Int. Journal of Pattern Recognition and Artificial Intelligence.*

2. Li, Z. C., T. D. Bui, Y. Y. Tang and C. Y. Suen, *Computer Transformation of Digital Images and Patterns,* World Scientific Publishing Co. Pte, Ltd., Singapore, 1989.

3. Li, Z. C., T. D. Bui, C. Y. Suen and Y. Y. Tang, "Nonlinear transformations of digitized pattern," *Proc. 9th Int, Conf. on Pattern Recognition,* pp. 134 - 136, Rome, Italy, Nov. 1988.

4. Cheng, H. D., Y. Y. Tang, and C. Y. Suen, "VLSI Architecture for image transformation," *Proc. IEEE 1989 COMPEURO Conf., VLSI and Computer Peripherals,* pp. 2-124 - 2-126, Hamburg, May 1989.

5. Cheng, H. D., Y. Y. Tang, and C. Y. Suen, "Parallel image transformation and its VLSI implementation," in press, *Pattern Recognition.*

Chapter 5:

1. Tang, Y. Y., H. D. Cheng and C. Y. Suen, "Size-rotation-invariant character recognition" *Proc. Int. Conf, on Computer Processing of Chinese and Oriental Languages,* pp. 161 - 165, Toronto, Canada, Aug. 1988.

2. Wang, K., Y. Y. Tang and C. Y. Suen, "Multi-layer projections for the classification of similar Chinese characters," *Proc. 9th Int, Conf. on Pattern*

Recognition, pp. 842 - 844, Rome, Italy, Nov. 1988.

Chapter 6:

1. Tang, Y. Y., Z. C. Li, C. Y. Suen and T. D. Bui, "Conversion of Chinese characters by transformation models," *Proc. Int. Conf. on Computer processing of Chinese and Oriental Languages,* pp. 293 - 297, Toronto, Canada, Aug. 1, 1988.

Chapter 7:

1. Tang, Y. Y. and C. Y. Suen, "Nonlinear shape restoration by transformation models," in press, *Proc. 10th Int, Conf. on Pattern Recognition,* Atlantic City, New Jersey, June 1990.

2. Tang, Y. Y., C. Y. Suen and Q. R. Wang, "Chinese character classification by globally trained tree classifier and Fourier descriptors of condensed patterns," *Proceedings The IEEE 1-st Int. Conf. on Computers and Applications,* pp. 215-220, Beijing, China, June 1984.

3. Q. R. Wang, C. Y. Suen and Y. Y. Tang, "Application of a statistical equivalent block classifier in the recognition of Chinese characters printed in different fonts," *Proceedings Int. Conf. Chinese Computing,* pp. H2.1-H2.13, San Francisco, U. S. A., Feb. 1985.

CHAPTER 2

MULTIPLE - LEVEL INFORMATION SOURCE MODEL

2.1 INTRODUCTION

In recent years many new topics such as computer vision, robot vision, the recognition of motion, etc. have evolved. They bring along more and more complex and difficult pattern recognition problems. Also the pattern recognition system itself becomes more and more complicated. In order to investigate efficiently such difficult problems and such complicated systems, it is necessary to analyze systematically the changes in entropy which occur in the different stages of a pattern recognizer. In this way, the different factors which affect the uncertainty in different phases in the process of pattern recognition can be found. Furthermore, corresponding methods might be found to handle the different problems and reduce the entropy in the different stages of a recognition system.

In order to study the effects of different factors in the process of pattern recognition, we build a new theoretical model called Multiple-level Information Source Model (MLIS) [Tang89a] to analyze the internal structures of various information sources based on Shannon's information theory [Guiasu77, Jones79, Rubbe63, Shanno48a, b]. To explain these developments clearly and conveniently,

Section 2.2 gives primary concept of ordinary information source first.

Section 2.3 presents the definition of the MLIS. For a typical recognition system, there are four levels, IS1 to IS4, in this information source, and they can be divided into two categories : entropy-reduced and entropy-increased. Section 2.4 describes the necessary and sufficient conditions to determine the category of an MLIS. Any practical pattern recognition system can be considered as an MLIS by distinguishing the characteristics of different factors.

Section 2.5 analyzes the various factors in the process of pattern recognition and presents the basic ideas on the mapping of a pattern recognition system into an MLIS. In general, an increase in the entropy in the different parts of a pattern recognition system can be attributed to three factors : a) intrinsic distortion arising from the characteristics of the target pattern set, b) distortions due to the transducer and digitization, and c) information loss which is dependent on the feature extraction method. Through theoretical analysis, a pattern recognition system can be modeled by an MLIS, so that the entropy in the different parts can be addressed. Consequently, a theoretical analysis of the entropy distribution in the MLIS indicates that in order to improve the performance of a pattern recognition system, the entropy of the MLIS must be reduced in all the different levels. Specifically, IS1, IS2 and IS4 must be converted.

2.2 PRIMARY CONCEPT OF INFORMATION SOURCE

In Shannon's information theory [Shanno48a, b], an information source (IS) is described by a probability space $\{ W, \beta_W, P \}$, where W is the set of letters or signals coming out of the IS; β_W is the σ - algebra generated by all the well-defined subset of W ; and P is a probability measure defined on β_W [Davenp70, Eisen69, Parzen60, Stark86]. In general, the output of an IS is regarded as a random process, which may be either a function of time, or space, or both. Therefore IS's can be categorized as either discrete or continuous by their time or space characteristics. In the case of discrete IS, from [Zhou83] we know that the output is a random sequence, that is

$$w_1, w_2, \cdots, w_i, \cdots, w_n$$

where $n \in I$.

If w_i is a discrete variable, and let

$$w_i \in A = \{ a_1, a_2, \cdots, a_L \} \qquad i = 1,2,\ldots,n$$

then the IS can be represented by the random vector

$$W = (w_1, w_2, \cdots, w_n) \in A^n .$$

Its probability distribution is

$$\begin{aligned} & P (w_1, w_2, \cdots, w_n) \\ = \; & P \{ w_1 = a_{s1}, w_2 = a_{s2}, \cdots, w_n = a_{sn} \} \\ = \; & P \{ W = \bar{a} \} , \end{aligned}$$

where $\bar{a} = \{ a_{s1}, a_{s2}, \cdots, a_{sn} \}$,

or denoted by

$$P(\bar{a}) = P(a_{s1}, a_{s2}, \cdots, a_{sn}) = P_{s1,s2,\ldots,sn}$$

If w_i 's are uncorrelated, then

$$P(\bar{a}) = \prod_{i=1}^{n} P(w_i = w_{si}) = \prod_{i=1}^{n} P_{si}$$

Obviously, the following condition is true

$$\sum_{a_1 \in A, \cdots, a_n \in A} P(w_i) = P(W) = P(A^n) = 1.$$

where the multiple sum is taken over the elements belonging to A .

To describe the information uncertainty of an IS, an entropy function is defined as

$$H = - \sum_{a_1 \in A, \cdots, a_n \in A} P(w_i) \log P(w_i).$$

In order to analyze the internal structure of IS's, in this work, we will adopt the following definition wherever we mention an IS.

Definition 2.1

An information source (IS) is defined by an entropy space

$$\Omega = (W, P_W, H_W)$$

where

$$W = \{ w_i \mid w_i \in A = \{ a_1, a_2, \cdots, a_L \}, \quad i = 1,2, \cdots, n, \ n \in I \};$$

P_W is the apriori probability defined on W such that

(i) $P_W (w_i) = p_i$;

(ii) $0 < p_i < 1$;

(iii) $\sum_{i=1}^{n} p_i = 1$.

H_W is the uncertainty measure of Ω

$$H_W = H (p_1, p_2, \cdots, p_n) = - \sum_{i=1}^{n} p_i \log p_i .$$

In the real world, it is often true that an IS can be considered as a combination of several subcomponents each with distinct characteristics, that is given an Ω, then

$$\Omega = \bigcup_{i=1}^{m} \Omega_i ,$$

we call Ω_i 's sub-entropy space or sub-IS's. See [Guiasu77, Jones79, Raisbe63, Young71] for details about ordinary information source.

2.3 MULTIPLE-LEVEL IS (MLIS)

Of most interest is to find the relationship among Ω_i 's. In this work we focus on one special case : all Ω_i 's form a chain, (i.e. the output of the Ω_1 is the input of Ω_2 , and so on). We call this kind of IS's multiple-level information sources (MLIS) which is formally defined as follows.

Definition 2.2

Given an IS $\Omega = (W, P_W, H_W)$. If there is no such IS, as $\Omega' = (W', P_{W'}, H_{W'})$, such that

$$\forall_{i,j} (w_i \rightarrow w_j, w_i \in W', w_j \in W),$$

where $w_i \rightarrow w_j$ denotes a mapping of W' onto W , then Ω is called a single-level IS, otherwise Ω is called a multiple-level IS.

Definition 2.3

Given two IS's $\Omega_1 = (W^1, P_{W^1}, H_{W^1})$, and $\Omega_2 = (W^2, P_{W^2}, H_{W^2})$. If the following condition holds,

$$\forall_{i,j} (w_i \rightarrow w_j, w_i \in W^1, w_j \in W^2),$$

where $w_i \rightarrow w_j$ denotes a mapping of W^1 onto W^2 , then Ω_2 is called the superlevel IS of Ω_1. If Ω_1 is a single-level IS, Ω_2 is called a 2-level IS. If Ω_1 is a k-level IS, Ω_2 is called a ($K + 1$) - level IS as shown in Fig. 2.1.

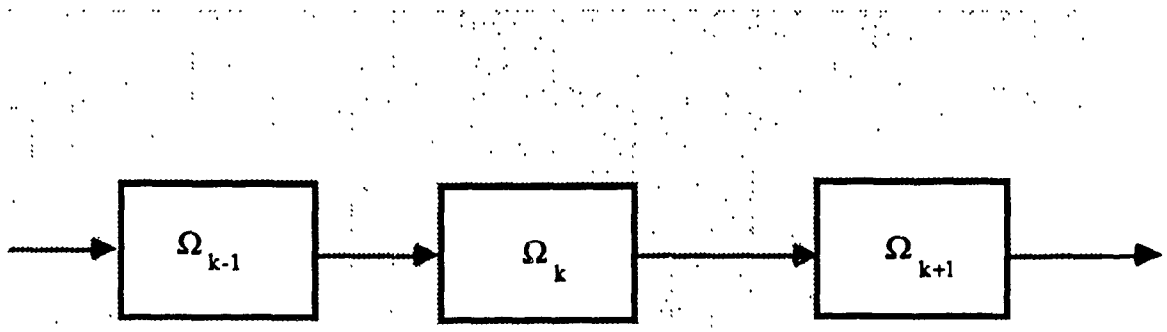


Fig. 2.1 (k + 1) - Level IS

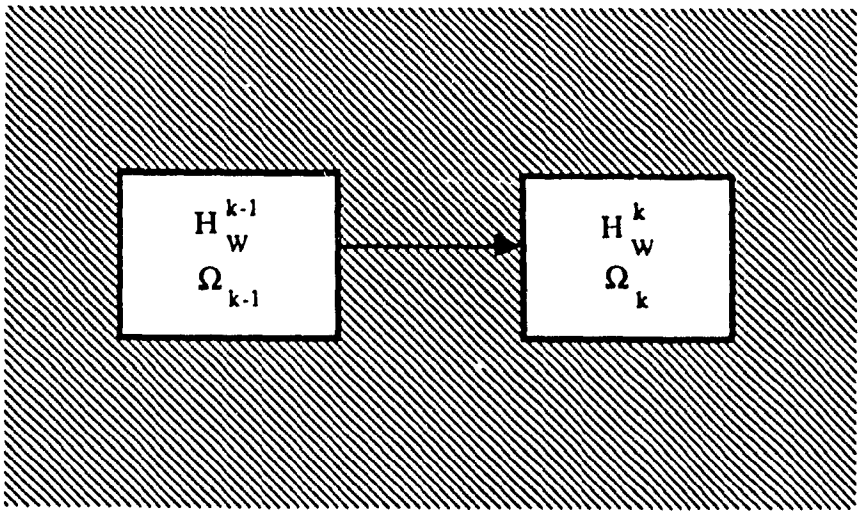


Fig. 2.2 Entropy-Increased or Entropy-Reduced IS

2.4 ENTROPY-INCREASED AND REDUCED IS'S

Considering the entropy function, the relationship between two IS's of two adjacent levels can be divided into two categories : entropy-reduced and entropy-increased.

Definition 2.4

Given two IS's $\Omega_{k-1} = (W^{k-1}, P_{W^{k-1}}, H_{W^{k-1}})$ and $\Omega_k = (W^k, P_{W^k}, H_{W^k})$. If $H_{W^k} > H_{W^{k-1}}$, we call Ω_k an entropy-increased IS; if $H_{W^k} < H_{W^{k-1}}$, we call Ω_k an entropy-reduced IS (Fig. 2.2).

The following theorem describes the necessary and sufficient conditions to judge if an MLIS is entropy-reduced or entropy-increased. This theoretical model will be used to describe the pattern recognition system in the next section.

Theorem 2.1

Given $\Omega_{k-1} = (W^{k-1}, P_{W^{k-1}}, H_{W^{k-1}})$, $\Omega_k = (W^k, P_{W^k}, H_{W^k})$, $| W^{k-1} | = m_{k-1}$, $| W^k | = m_k$,

(i) Ω_k is an entropy-reduced IS,

iff there is a mapping

$$I : W^{k-1} \rightarrow W^k$$

such that

$$\forall_j \exists_{l=1,2,\ ,m_j} (w_i^{j_l} \rightarrow w_j , w_i^{j_l} \in W^{k-1} , w_j \in W^k)$$

and

$$\forall_{j,l} \; (P(w_j \mid w_i^{j_l}) = 1) ,$$
$$j = 1, 2, \ldots, m_k .$$
$$\sum_{j=1}^{m_k} m_j = m_{k-1} ,$$
$$\forall_j \; m_j \geq 2 ,$$

where $w_i^{j_l}$ represents the i-th element of W^{k-1} whose image is $w_j \in W^k$ and it is also the l-th element in the j-th group in which all the elements have the same image w_j . $P(. \mid .)$ represents a conditional probability, m_j denotes the number of elements in the j-th group.

(ii) Ω_k is an entropy-increased IS,

iff there is a mapping

$$I : W^{k-1} \rightarrow W^k$$

such that

$$\forall_i \exists_{l=1,2, \ldots, m_i} (w_i \rightarrow w_j^{i_l} , w_i \in W^{k-1} , w_j^{i_l} \in W^k)$$

and

$$\forall_{j,l,m_i} (P(w_j^{i_l} \mid w_i) > 0, \; \sum_{l=1}^{m_i} P(w_j^{i_l} \mid w_i) = 1),$$
$$i = 1, 2, \ldots, m_{k-1} ,$$
$$\sum_{i=1}^{m_{k-1}} m_i = m_k ,$$

where $w_i \in W^{k-1}$ has a group of m_i images in W^k . $w_j^{i_l}$ represents the l-th image of w_i in the i-th image group and it is also the j-th element of W^k

Proof

(i)

(a) If =>

The given conditions mean that each $w_j \in W^k$ is the image of a group of $w_i \in W^{k-1}$, i represents the i-th element of W^{k-1} , w_i 's in the same group are denoted as $w_i^{j_l}$, l represents the l -th element in the j -th group. Therefore we have

$$P(w_i^{j_l}) = P_{W^{k-1}} (w_i^{j_l})$$
$$= P_{W^{k-1}} (w_i)$$

Since it is known that $P(w_j \mid w_i^{j_l}) = 1$ (which means that each w_i belongs to only one group and has only one image).

Whereby

$$P_{W^k} (w_j) = P (w_j)$$
$$= \sum_{l=1}^{m_j} P(w_j \mid w_i^{j_l}) P(w_i^{j_l})$$
$$= \sum_{l=1}^{m_j} P(w_i^{j_l})$$
$$= \sum_{l=1}^{m_j} P_{W^{k-1}} (w_i^{j_l}) . \qquad (2.1)$$

This means that

$$P_{W^k} (w_j) > P_{W^{k-1}} (w_i^{j_l}) \qquad \text{for all } l$$

so that

$$-\log P_{W^k}(w_j) < -\log P_{W^{k-1}}(w_i^{j_l}) \qquad \text{for all } l \ .$$

Let

$$P_{W^{k-1}}(w_i^{'}) = \max P_{W^{k-1}}(w_i^{j_l}) \qquad \text{for all } l$$

whereby

$$-\log P_{W^{k-1}}(w_i^{'}) \leq -\log P_{W^{k-1}}(w_i^{j_l}) \qquad \text{for all } l$$

then

$$-\sum_{l=1}^{m_j} P_{W^{k-1}}(w_i^{j_l}) \log P_{W^{k-1}}(w_i^{j_l})$$

$$\geq -\sum_{l=1}^{m_j} P_{W^{k-1}}(w_i^{j_l}) \log P_{W^{k-1}}(w_i^{'})$$

and

$$-\sum_{l=1}^{m_j} P_{W^{k-1}}(w_i^{j_l}) \log P_{W^{k-1}}(w_i^{'})$$

$$= -\left[\sum_{l=1}^{m_j} P_{W^{k-1}}(w_i^{j_l})\right] \log P_{W^{k-1}}(w_i^{'}) \ .$$

From Eq. (2.1) we have

$$-\left[\sum_{l=1}^{m_j} P_{W^{k-1}}(w_i^{j_l})\right] \log P_{W^{k-1}}(w_i^{'})$$

$$= -P_{W^k}(w_j) \log P_{W^{k-1}}(w_i^{'})$$

$$> -P_{W^k}(w_j) \log P_{W^k}(w_j) \ . \qquad (2.2)$$

Since m_k is the number of elements in W^k, then from Eqs.(2.1) and (2.2) we have

$$\sum_{i=1}^{m_k} -\sum_{l=1}^{m_j} P_{W^{k-1}}(w_i^{j_l}) \log P_{W^{k-1}}(w_i^{j_l})$$

$$> - \sum_{j=1}^{m_k} P_{W^k}(w_j) \log P_{W^k}(w_j).$$

That is

$$H_{W^{k-1}} > H_{W^k}$$

so that Ω_k is an entropy-reduced IS.

(b) Only if <=

(Proof by using *Contradiction Method*)

From definition 2.4, we know that if Ω_k is an entropy-reduced IS, then we have

$$H_{W^{k-1}} > H_{W^k}. \tag{2.3}$$

Now we change the given condition to

$$\forall_j \exists_{l=1,2,\ldots m}, \ (w_i^{j_l} \rightarrow w_j, w_i^{j_l} \in W^{k-1}, w_j \in W^k)$$

and

$$\exists_{j,l} (P(w_j \mid w_i^{j_l}) < 1),$$
$$j = 1, 2, \cdots, m_k,$$
$$\sum_{j=1}^{m_k} m_j = m_{k-1},$$
$$\forall_j \ m_j \geq 2.$$

That is some w_i may have more than one image in W^k. For example, there are Ω_k, Ω_{k-1} as shown in the following and Fig. 2.3.

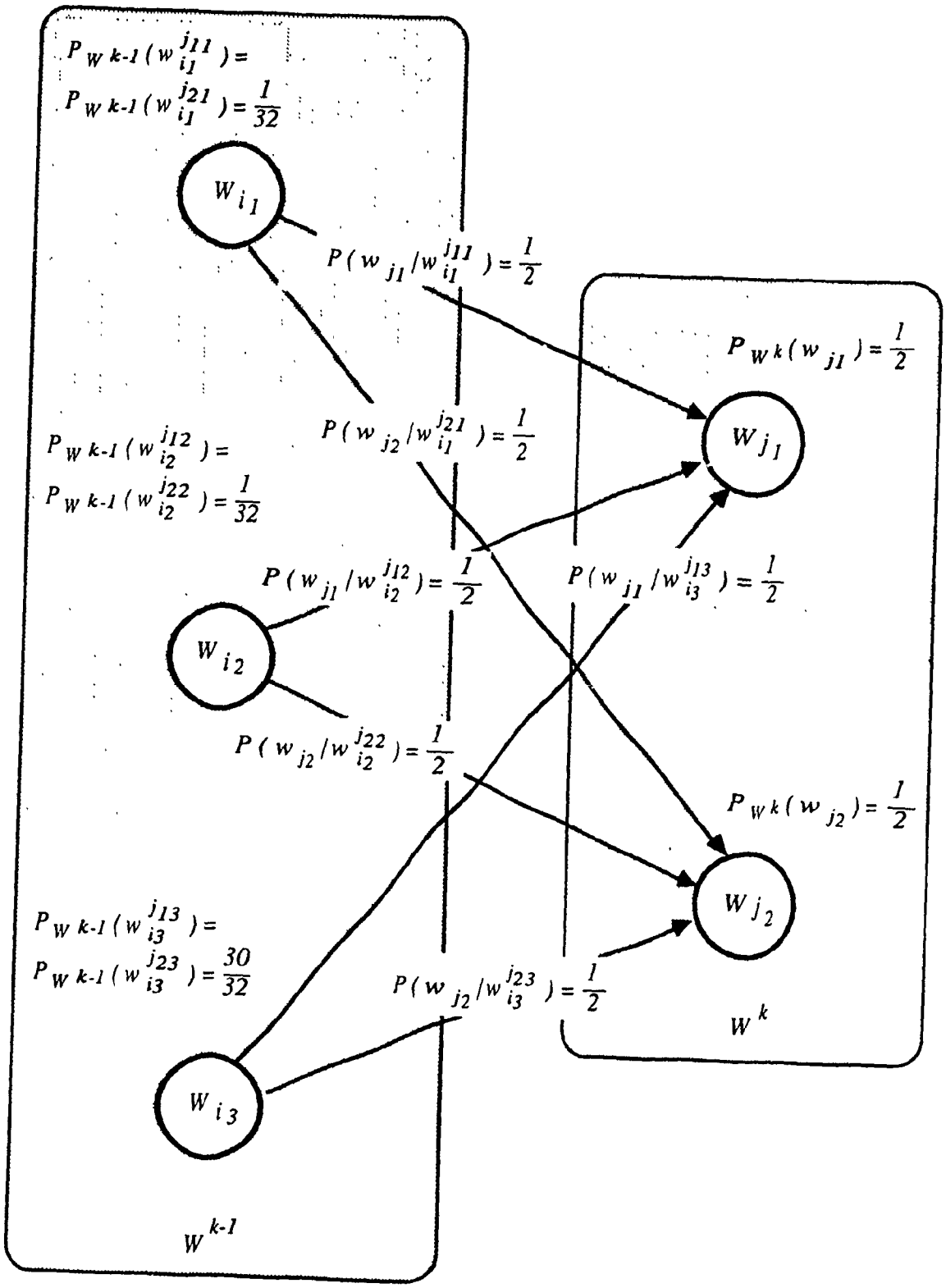


Fig. 2.3 An Example for the Theorem 2.1

$$W^{k-1} = \{ w_{i_1}, w_{i_2}, w_{i_3} \}.$$
$$W^{k} = \{ w_{j_1}, w_{j_2} \}.$$

$$P_{W^{k-1}}(w_{i_1}) = P_{W^{k-1}}(w_{i_1}^{j_{1_1}}) = P_{W^{k-1}}(w_{i_1}^{j_{2_1}}) = \frac{1}{32},$$
$$P_{W^{k-1}}(w_{i_2}) = P_{W^{k-1}}(w_{i_2}^{j_{1_2}}) = P_{W^{k-1}}(w_{i_2}^{j_{2_2}}) = \frac{1}{32},$$
$$P_{W^{k-1}}(w_{i_3}) = P_{W^{k-1}}(w_{i_3}^{j_{1_3}}) = P_{W^{k-1}}(w_{i_3}^{j_{2_3}}) = \frac{30}{32},$$

$$P(w_{j_1} \mid w_{i_1}^{j_{1_1}}) = P(w_{j_2} \mid w_{i_1}^{j_{2_1}}) = \frac{1}{2},$$
$$P(w_{j_1} \mid w_{i_2}^{j_{1_2}}) = P(w_{j_2} \mid w_{i_2}^{j_{2_2}}) = \frac{1}{2},$$
$$P(w_{j_1} \mid w_{i_3}^{j_{1_3}}) = P(w_{j_2} \mid w_{i_3}^{j_{2_3}}) = \frac{1}{2},$$

$$P_{W^k}(w_{j_1}) = \sum_{l=1}^{3} P_{W^{k-1}}(w_{i_l}^{j_{1_l}}) P(w_{j_1} \mid w_{i_l}^{j_{1_l}})$$
$$= \frac{1}{32} \times \frac{1}{2} + \frac{1}{32} \times \frac{1}{2} + \frac{30}{32} \times \frac{1}{2} = \frac{1}{2}.$$
$$P_{W^k}(w_{j_2}) = \sum_{l=1}^{3} P_{W^{k-1}}(w_{i_l}^{j_{2_l}}) P(w_{j_2} \mid w_{i_l}^{j_{2_l}})$$
$$= \frac{1}{32} \times \frac{1}{2} + \frac{1}{32} \times \frac{1}{2} + \frac{30}{32} \times \frac{1}{2} = \frac{1}{2}.$$

$$H_{W^k} = -(P_{W^k}(w_{j_1}) \log P_{W^k}(w_{j_1}) + P_{W^k}(w_{j_2}) \log P_{W^k}(w_{j_2}))$$
$$= -(\frac{1}{2} \log \frac{1}{2} + \frac{1}{2} \log \frac{1}{2})$$
$$= 1.000 \text{ bit}.$$

$$H_{W^{t-1}} = -\sum_{l=1}^{3} P_{W^{t-1}}(w_{i_l}) \log P_{W^t}(w_{i_l})$$

$$= -\left(\frac{1}{32}\log\frac{1}{32} + \frac{1}{32}\log\frac{1}{32} + \frac{30}{32}\log\frac{30}{32}\right)$$

$$= 0.340 \text{ bit} .$$

That is

$$H_{W^t} > H_{W^{t-1}} .$$

This contradicts to Eq.(2.3). Therefore we have to keep the given conditions. The proof of part (i) is completed.

(ii)

(a) if $=>$

$$P_{W^t}(w_j) = P_{W^t}(w_j^{i_l})$$

$$= P(w_i \cap w_j^{i_l})$$

$$= P(w_j^{i_l} \mid w_i) P(w_i)$$

$$= P(w_j^{i_l} \mid w_i) P_{W^{t-1}}(w_i)$$

whereby

$$-\sum_{l=1}^{m'} P_{W^t}(w_j^{i_l}) \log P_{W^t}(w_j^{i_l})$$

$$= -\sum_{l=1}^{m'} \left(P(w_j^{i_l} \mid w_i) P_{W^{t-1}}(w_i) \left(\log\left(P(w_j^{i_l} \mid w_i) P_{W^{t-1}}(w_i)\right)\right)\right)$$

$$= -\sum_{l=1}^{m'} P(w_j^{i_l} \mid w_i) P_{W^{t-1}}(w_i) \left(\log P(w_j^{i_l} \mid w_i) + \log P_{W^{t-1}}(w_i)\right)$$

$$= -P_{W^{t-1}}(w_i) \sum_{l=1}^{m'} P(w_j^{i_l} \mid w_i) \log P(w_j^{i_l} \mid w_i)$$

$$-\left(\sum_{l=1}^{m'} P(w_j^{i_l} \mid w_i)\right)\left(P_{W^{t-1}}(w_i) \log P_{W^{t-1}}(w_i)\right) .$$

Since

$$-P_{W^{k-1}}(w_i)\sum_{l=1}^{m_i'}P(w_j^{i_l}\mid w_i)\log P(w_j^{i_l}\mid w_i)>0,$$

and

$$\sum_{l=1}^{m_i'}P(w_j^{i_l}\mid w_i)=1\quad(*\ \textit{given condition}\ *).$$

Therefore we have

$$-\sum_{l=1}^{m_i'}P_{W^k}(w_j^{i_l})\log P_{W^k}(w_j^{i_l})>-P_{W^{k-1}}(w_i)\log P_{W^{k-1}}(w_i)$$

whereby

$$\sum_{i=1}^{m_{k-1}}\left(-\sum_{l=1}^{m_i'}P_{W^k}(w_j^{i_l})\log P_{W^k}(w_j^{i_l})\right)$$
$$>-\sum_{i=1}^{m_{k-1}}P_{W^{k-1}}(w_i)\log P_{W^{k-1}}(w_i),$$

or

$$-\sum_{j=1}^{m_k}\left(P_{W^k}(w_j)\log P_{W^k}(w_j)\right)$$
$$>-\sum_{i=1}^{m_{k-1}}P_{W^{k-1}}(w_i)\log P_{W^{k-1}}(w_i).$$

That is

$$H_{W^k}>H_{W^{k-1}}$$

so that

Ω_k is an entropy-increased IS .

(b) Only if <=

(Proof by using *Contradiction Method*)

Since Ω_k is an entropy-increased IS, we have

$$H_{W^k} > H_{W^{k-1}} .$$

This means that

$$\forall_j \, (P_{W^k} (w_j) > 0)$$

and

$$\sum_{j=1}^{m_k} P_{W^k} (w_j) = 1 . \tag{2.4}$$

Now we change the given condition to

$$\exists_{j,l,m} , \, (P (w_j^{i_l} \mid w_i) = 0 , \; \sum_{l=1}^{m_i} (P (w_j^{i_l} \mid w_i) < 0) ,$$
$$i = 1, 2, \cdots, m_{k-1} ,$$
$$\sum_{i=1}^{m_{k-1}} m_i = m_k ,$$

This implies

$$\exists_j \, (P_{W^k} (w_j) = P (w_j^{i_l} \mid w_i) \, P_{W^{k-1}} (w_i) = 0) ,$$

therefore

$$\sum_{j=1}^{m_k} P_{W^k} (w_j) < 1 .$$

This contradicts to Eq.(2.4). Thereby we have proved the only-if part. The proof of (ii) is completed.

2.5 UNCERTAINTY IN PATTERN RECOGNITION

2.5.1 MLIS in Pattern Recognition System

A pattern recognition system is illustrated in Fig. 2.4. The initial input is a target pattern set composed of physical variables. A data acquisition subsystem captures the analog data from the physical world by a transducer, and converts them to digital information called measured data. The measured data are then input into the data preprocessing and feature extraction subsystems and grouped into a set of characteristic features called feature vectors as output. The last subsystem is a classifier which puts the unknown input pattern into the identified class based on its feature vector. Essentially, the process of pattern recognition is one which removes uncertainty hidden in the input pattern. Normally, information theory treats uncertainty as if it comes from only a single information source. However a pattern recognition problem is better treated as one with a multiple information source. This can be seen from Fig. 2.4:

(1) Of course the target pattern set is the primary information source already denoted by IS1.

(2) The data acquisition subsystem combined with IS1 is considered as the second level of information source and is denoted by IS2.

(3) The third level of information source consists of IS2 and the data preprocess-

ing subsystem and is denoted by IS3.

(4) Furthermore, IS3 is combined with the feature extraction subsystems to construct the fourth level of information source denoted by IS4.

Distinguishing the information sources will help to analyze the factors resulting in the addition of uncertainty or entropy, and find solutions to reduce the uncertainty.

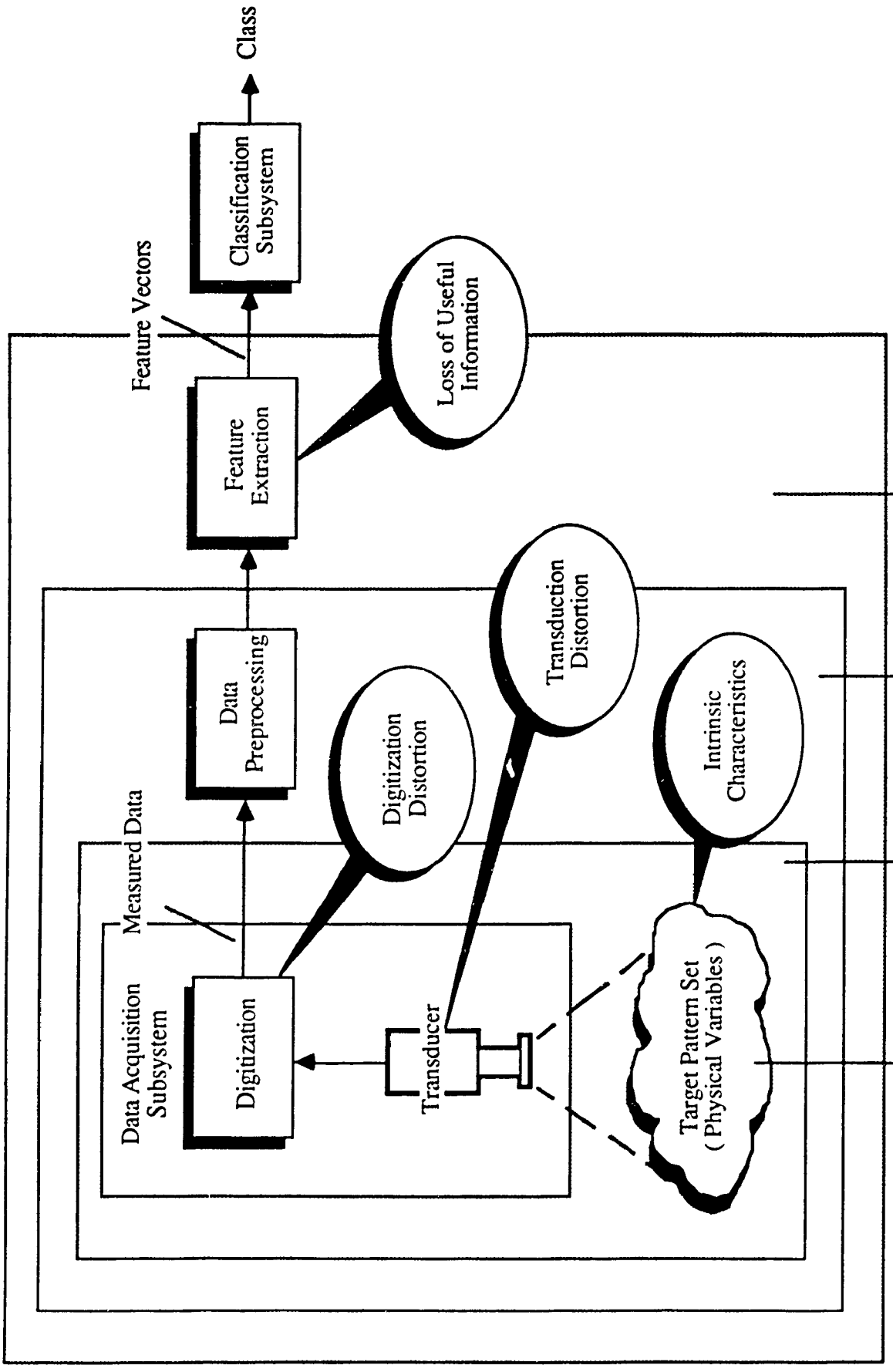


Fig. 2.4 MLIS in Pattern Recognition System

2.5.2 Uncertainty in MLIS for Pattern Recognition

IS1 :

Obviously the uncertainty comes primarily from the IS1, i.e. it is determined by the intrinsic characteristics of the target pattern set, such as the categories and numbers of patterns; sizes, fonts and directions of characters; and other types of intrinsic noise shown in Fig. 2.5(a) - Fig. 2.5(d).

Fig. 2.5(a) illustrates several examples of the uncertainty in IS1 - categories of pattern including (i) characters, (ii) medical patterns, (iii) concept cars, (iv) fingerprints, and (v) parts. Figs. 2.5(b)-(d) give some examples of the uncertainty in IS1 including different fonts, rotations and sizes of characters respectively.

For example, the entropy of a set of only 52 standard English letters without any size, font, and rotation variations is certainly much smaller than that of a set which allows every letter to have ten sizes, five fonts and any degree of rotations.

IS2 :

In IS2, the uncertainty is introduced by transducer and digitization distortions. Compared with digitization distortion, which may confuse pattern samples due to scanning noises, transducer distortion may create severe problems. An obvious example is the photograph taken by a camera at an oblique angle. In the camera scanning system shown in Fig. 2.6, if the angle α between the plane which contains the object to be scanned and the direction of the camera is 90°, a standard digitized image as shown in Fig. 2.6(a) is produced. Obviously a standard

image is easily identifiable by the recognition system. However, this ideal condition is not guaranteed. Consequently, nonlinear distortion arises [Tang88a]. In other practical examples, the documents with nonlinear shape distortion produced by a camera scanning system are illustrated in Fig. 2.7.

Suppose the original target pattern set consists of both English letters and Chinese characters and every sample has only one font, two size and one direction. This means that our IS1 has a finite entropy. However, because of the transducer problem, every standard sample can produce an infinite number of distorted copies. Correspondingly, the entropy of IS2 becomes infinite.

IS3 :

In the IS3, some preprocessing techniques such as filling, thinning etc. have been employed to remove noise from the measured data. The entropy in this level is reduced.

IS4 :

As for IS4, although it is well known that a feature extraction subsystem [Tou74] generally reduces the dimensions of the measured data, and hence the entropy, yet on the other hand, the entropy may also increase for several reasons : (1) sometimes it is difficult for the selected dimensions to contain the main features which can distinguish the pattern samples; (2) because one orthogonal transformation technique may not be able to take care of multiple categories of intrinsic characteristics. The net result is that it may lose useful information

leading to confusions of some patterns. In other words, losing useful information is equivalent to changing the intrinsic characteristics of the target pattern set so that it will add to the entropy from another aspect.

The factors which increase the uncertainty are summarized as the follows:

(1) IS1 : Intrinsic characteristics of the target pattern set.

(2) IS2 : Distortions due to the transducer and digitization.

(3) IS4 : Information loss due to unsuitable feature extraction methods.

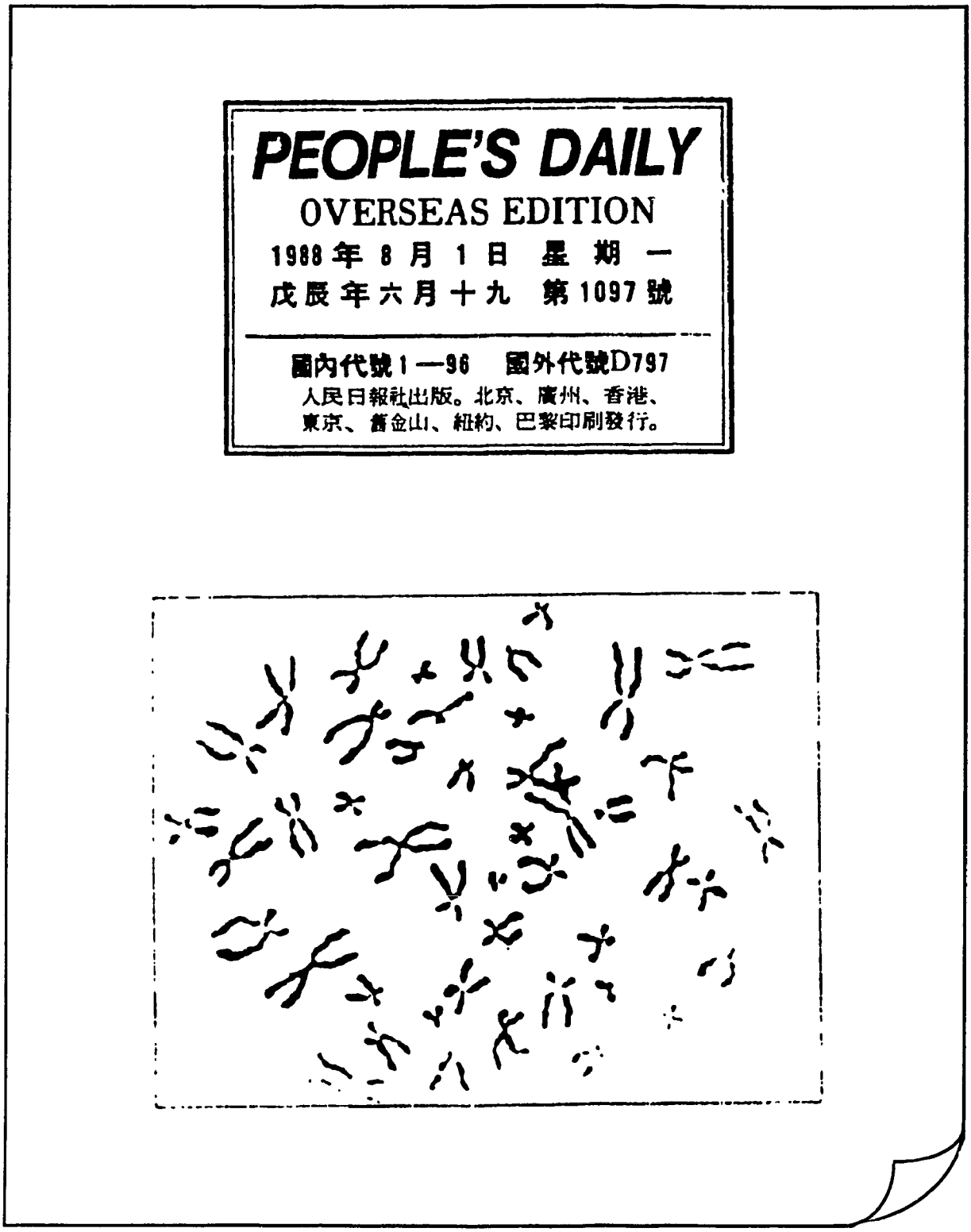


Fig. 2.5 Examples of Uncertainty in IS1 (a) Categories of Patterns

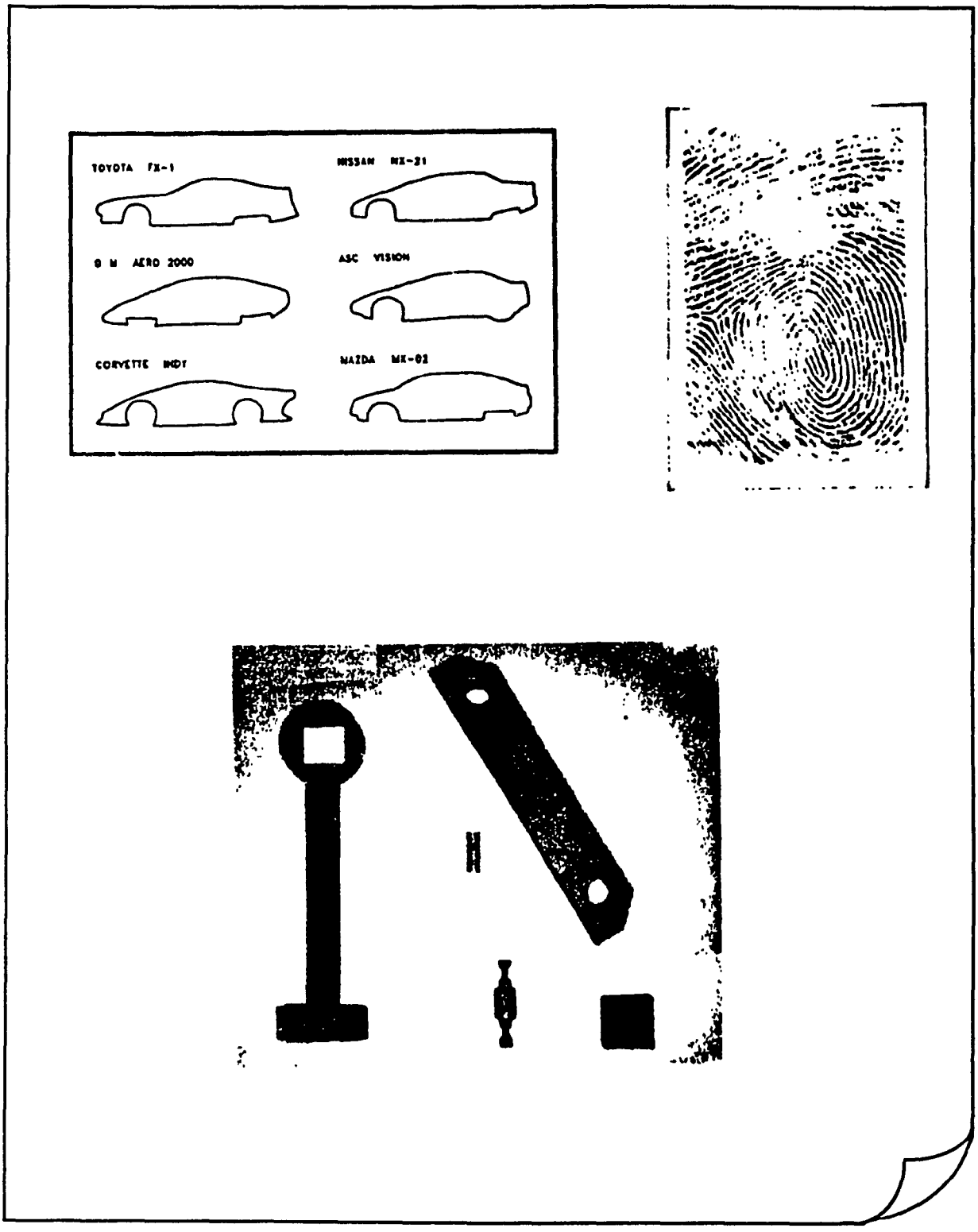


Fig. 2.5 Examples of Uncertainty in IS1 (a) Categories of Pattern (Cont.)

Fig. 2.5 Examples of Uncertainty in IS1 (b) Fonts of Characters

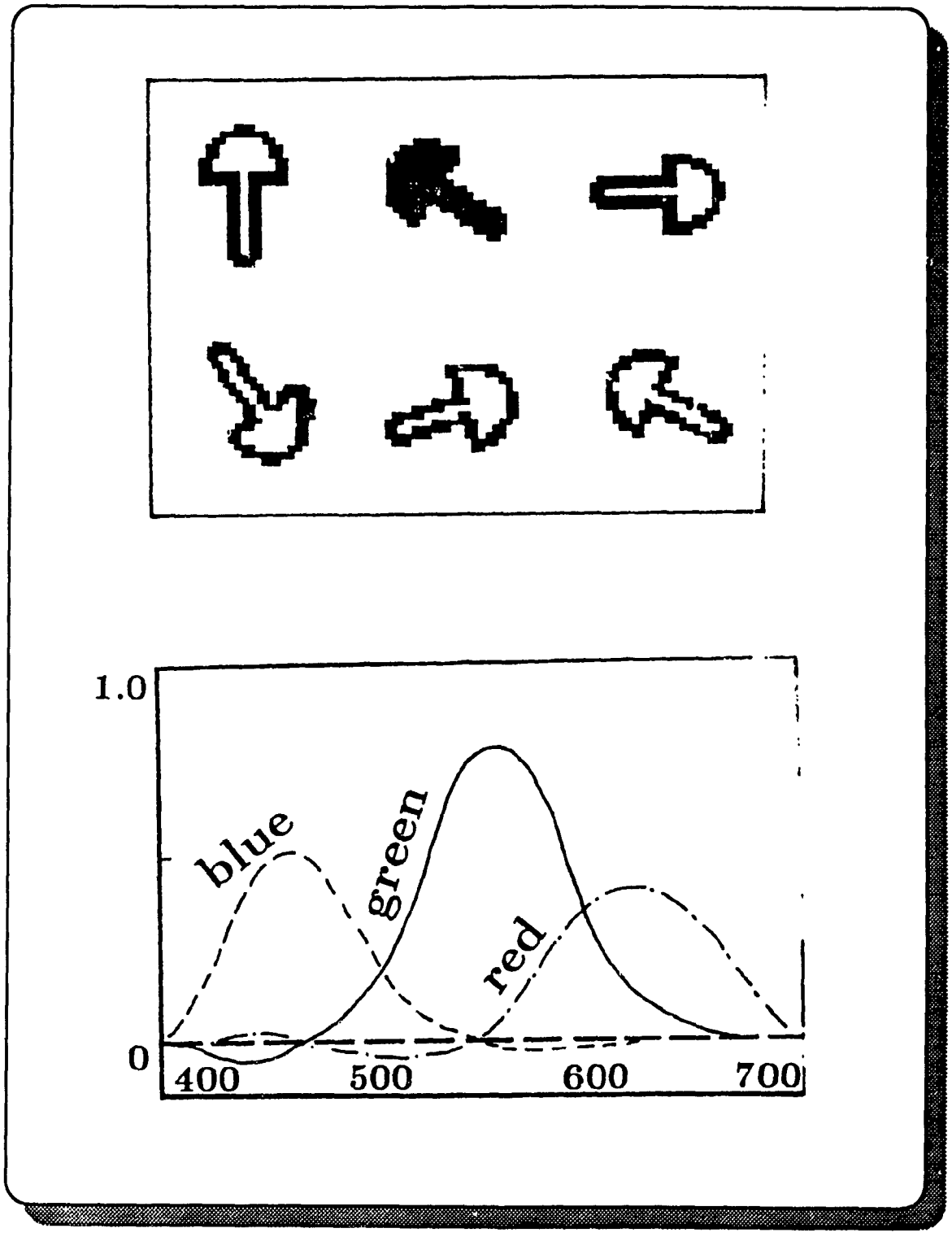


Fig. 2.5 Examples of Uncertainty in IS1 (c) Rotations of Pattern

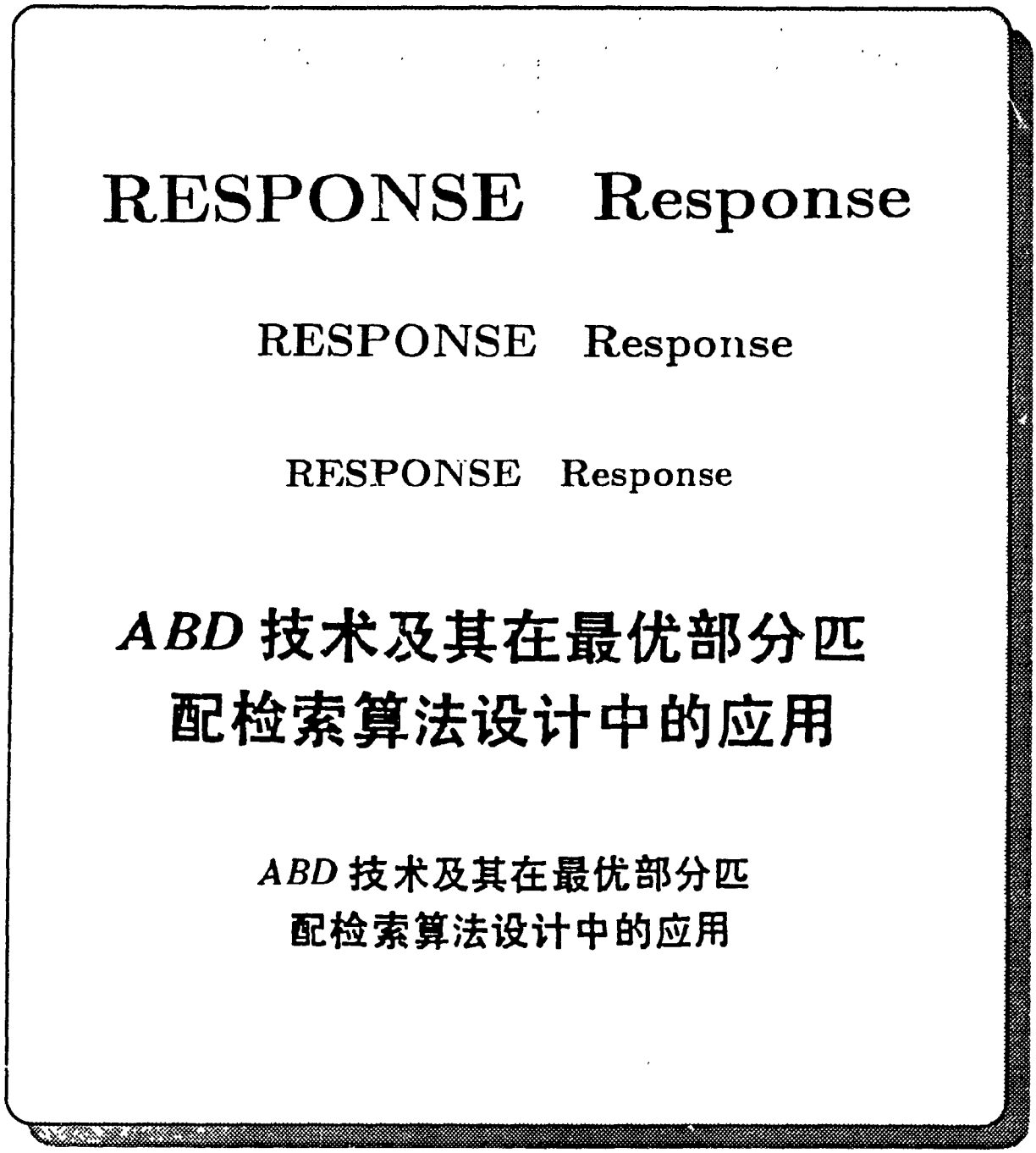


Fig. 2.5 Examples of Uncertainty in IS1 (d) Sizes of pattern

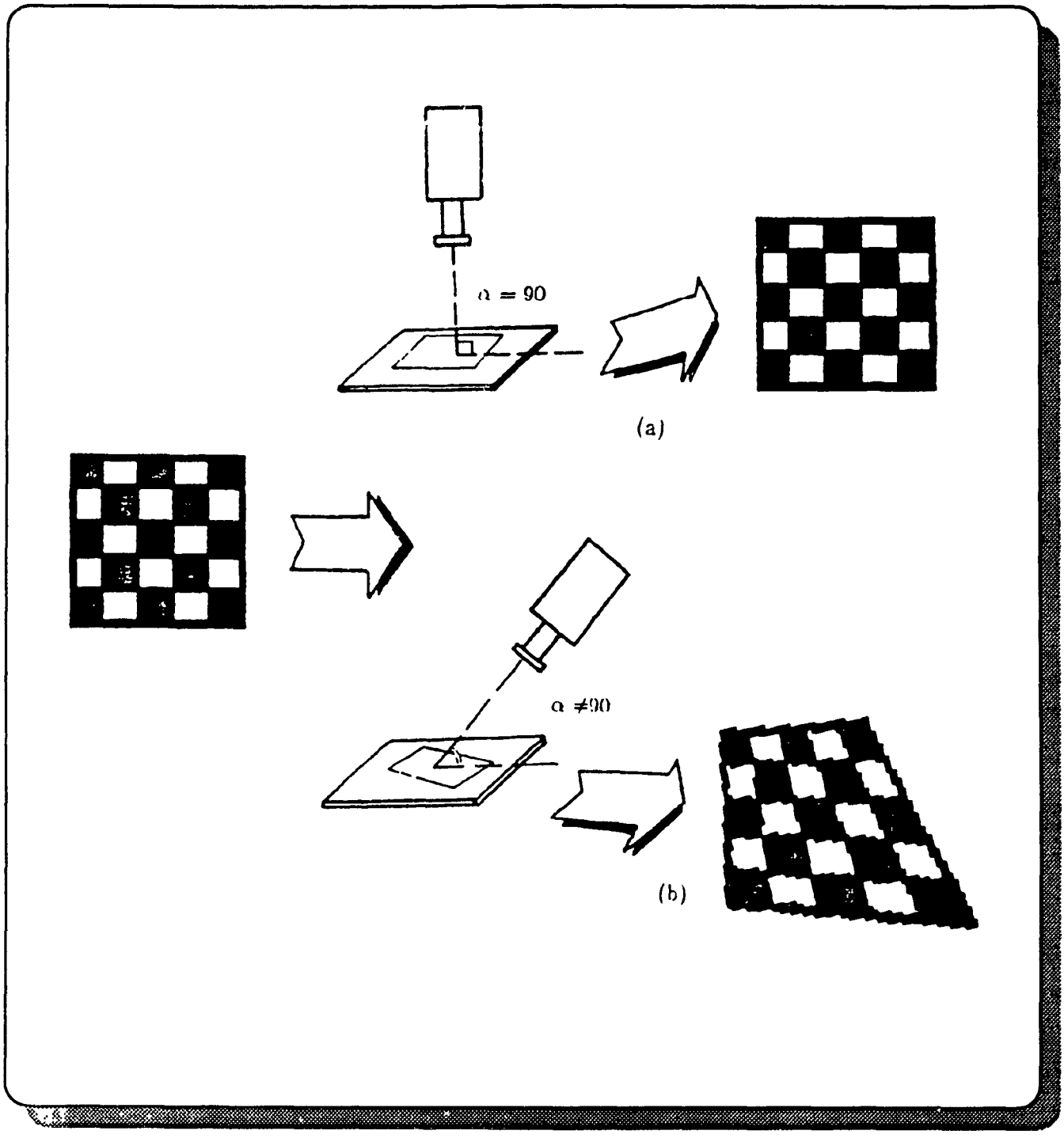


Fig. 2.6 An Example of Uncertainty in IS2

Fig. 2.7 An Example of the Document with Distortion

2.5.3 Examples of Uncertainty in MLIS

Clarifying the analysis of different factors of increasing entropy will help to handle them separately. Furthermore, we find that all these factors can be divided into two categories of variations: linear and nonlinear. For instance, in character recognition, we assume the standard pattern sample set only has one size, one orientation and one font, and does not have any geometric shape distortion. Then we can say that size, and rotation variations are linear, and some kinds of geometric shape distortions are nonlinear [Li89c]. Now it is natural that we can use various kinds of transformation theory to model these factors mathematically.

Two examples are presented below:

Example 2.1 :

Size variation can be treated as a linear transformation of the standard size. This is shown in the following formula [tang88b]:

$$\begin{bmatrix} X_j \\ Y_j \end{bmatrix} = \begin{bmatrix} D_j/d & 0 \\ 0 & D_j/d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2.5}$$

where (x,y) are the coordinates of a point of a pattern sample with the standard size, (X_j,Y_j) are the coordinates of a point of a distorted pattern sample with size j, d is the standard size, D_j stands for size j.

This example indicates that size variation increases the uncertainty of pattern set.

Example 2.2 :

Under certain conditions, the perspective projection distortion produced by scanning the data at an oblique angle, can be approximated by a simple shape model, bilinear transformation model which is described by the following formula

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1-x & x \end{bmatrix} & \begin{bmatrix} 0 & \\ 1-x & x \end{bmatrix} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} X_A & X_D \\ X_B & X_C \end{bmatrix} \\ \begin{bmatrix} Y_A & Y_D \\ Y_B & Y_C \end{bmatrix} \end{bmatrix} \begin{bmatrix} 1-y \\ y \end{bmatrix} \qquad (2.6)$$

where (X, Y) are the coordinates of a point of a distorted pattern sample. (x , y) are the coordinates of a point of the standard pattern sample. The eight parameters of X_A , X_B , X_C , X_D , Y_A , Y_B , Y_C and Y_D are used to determine four convex vertices of the distorted shape.

Therefore the inverse transformations (if they exist) naturally become the candidates of the tools to reduce the entropy which occurs at different parts of the recognition system. Based on the above analysis, several transformation techniques including shape transformation theory (linear and nonlinear) [Foley82, Li89c, Tang88a, b] have been developed and many of them aimed at reducing entropy. In [Tang88c, Qu88] we have set up a consistent theory framework named as Entropy-reduced Transformation (ERT) which will be introduced in the next chapter.

CHAPTER 3

ENTROPY-REDUCED TRANSFORMATION MODEL

3.1 INTRODUCTION

As mentioned above, the task of a pattern recognition system can be regarded as a conversion of an entropy-increased MLIS into an entropy-reduced one. To perform this conversion, a theoretical framework called Entropy-Reduced Transformation (ERT) model [Tang88c, 89a, Qu88] has been developed and described in Section 3.2.

In Section 3.3, Two important properties of the ERT i. e. the cascade and parallel properties have been presented and proved in Theorems 3.1 and 3.2. Based on these properties an algorithm has been proposed in Section 3.4 to build a practical pattern recognizer in a systematic manner.

3.2 ENTROPY-REDUCED TRANSFORMATION (ERT)

First let us define the notations to be used. For a given data set $W = \{ w_i \mid i = 1,2,...,n \}$, we can define an entropy space $\Omega = (W, P_W, H_W)$, where P_W is the apriori probability defined on W such that

$$\begin{aligned} &(i) \quad P_W (w_i) = p_i, \\ &(ii) \quad 0 < p_i < 1, \\ &(iii) \quad \sum_{i=1}^{n} p_i = 1. \end{aligned} \tag{3.1}$$

H_W is the uncertainty measure defined on W according to Shannon's entropy theory [Guiasu77, Shanno48a, b, Jones79, Young71]

$$H_W = H (p_1, p_2, \cdots, p_n) = -\sum_{i=1}^{n} p_i \log p_i . \tag{3.2}$$

Generally a target pattern set can be represented as

$$U = \bigcup_{i=1}^{m} W^i . \tag{3.3}$$

In its entropy space, P_U is determined by the relationships among the subsets W^i's (i = 1,2,...,m), and the uncertainty of U will have an upperbound $H_U \leq \sum_{i=1}^{m} H_{W^i}$.

The concept of entropy-reduced transformation is defined as follows.

Definition 3.1 :

Let $\Omega_i = (W^i, P_{W^i}, H_{W^i})$, $\Omega_j = (W^j, P_{W^j}, H_{W^j})$, $i \neq j$. $|W^i|$ and

$|W^j|$ stand for the number of elements (i.e. pattern samples) contained in W^i and W^j respectively. Assume that $|W^i| < |W^j|$. If a function F_{ji} can be defined such that

$$\begin{aligned} &(i) \quad |F_{ji}(W^j)| = |W^i|, \\ &(ii) \quad H_{F_{ji}(W^j)} = H_{W^i}, \end{aligned} \tag{3.4}$$

then F_{ji} is called an entropy-reduced transformation (ERT). We say that Ω_j is normalized to Ω_i by F_{ji}. W^i is called the reference set.

Example 3.1 :

Given a $\Omega_1 = (W^1, P_{W^1}, H_{W^1})$, such that

$$\begin{aligned} &W^1 = \{ x_j^i \mid i = 1,2;\ j = 1,2,\ldots,8 \}, \\ &P_{W^1}(x_j^i) = 1 / (2 \times 8) = 1 / 2^4, \quad i = 1,2;\ j = 1,2,\ldots,8, \\ &H_{W^1} = -\sum_{k=1}^{16} (1 / 2^4) \log (1 / 2^4) = 4 \text{ bits}. \end{aligned}$$

where the superscript i represents the size of the pattern samples and the subscript j represents the class the pattern samples belong among. That is x_1^1 is similar to x_1^2 except in size. Denote size 1 as d_1 and size 2 as d_2. Now we select all the elements in W^1 with size 1 as the reference set W^0. That is

$$\begin{aligned} &W^0 = \{ x_j^1 \mid j = 1,2,\ldots,8 \}, \\ &P_{W^0}(x_j^1) = 1 / 8 = 1 / 2^3, \quad j = 1,2,\ldots,8, \\ &H_{W^0} = -\sum_{j=1}^{8} (1 / 2^3) \log (1 / 2^3) = 3 \text{ bits}. \end{aligned}$$

Let d_k be the size of an element in W^1. F_{10} is defined as

$$F_{10}(d_k) = \begin{cases} d_k & \text{if } d_k = d_1; \\ d_k \times (d_1 / d_2) & \text{if } d_k = d_2. \end{cases}$$

This means that F_{10} groups two elements in W^1 into one cluster which corresponds to one element in W^0, whereby

$$| F_{10}(W^1) | = | W^0 |.$$

On the other hand,

$$P_{F_{10}(W^1)}(F_{10}^j(W^1)) = 2 \times \frac{1}{2^4} = \frac{1}{2^3},$$

$$H_{F_{10}(W^1)} = H_{W_0}.$$

Therefore F_{10} is an ERT.

Obviously the essential characteristics of ERT are:

(a) dimension reduction and

(b) the invariance of the entropy of the reference set.

This presents a consistent criterion to judge if a transformation technique is an ERT or not. For example if an orthogonal transform F can reduce the dimensions of W^j to that of W^i but cannot reduce H_{W^j} to H_{W^i} then F is not an ERT. This is why we cannot simply equate dimension reduction to entropy reduction. It is worth while to note that the reference set is a relative concept. Between any two or among more data sets generally we select one with minimum elements as the reference set. Sometimes the reference set is a union of several

sets its elements are still minimum when comparing with other sets.

According to the definition of ERT we can get a significant conclusion which can be treated as a lemma as follows.

Lemma 3.1 :

Given $\Omega_i = (W^i, P_W{}^i, H_W{}^i)$, $\Omega_j = (W^j, P_W{}^j, H_W{}^j)$, $i \neq j$ and $| W^i | < | W^j |$. Ω_j is normalized to Ω_i, iff there is an F_{ji} such that

$$\sum_{w_l \in F_{ji}^t(W^j)} p_l = p_t, \quad t = 1,2,\ldots, |W^i|, \tag{3.5}$$

where $F_{ji}^t(W^j)$ stands for the t-th cluster of $F_{ji}(W^j)$, $p_l \in P_{W^j}$, and $p_t \in P_{W^i}$.

The proof of the lemma is straight forward so that it is omitted here.

3.3 BASIC PROPERTIES OF ERT

Two important properties of the entropy-reduced transformation are given by the following theorems:

Theorem 3.1 :

Given $\Omega_i = (W^i, P_W{}^i, H_W{}^i)$, $\Omega_k = (W^k, P_W{}^k, H_W{}^k)$, and $\Omega_j = (W^j, P_W{}^j, H_W{}^j)$. If F_{ik} normalizes Ω_i to Ω_k, and F_{kj} normalizes Ω_k to Ω_j, then we have

$$F_{ij}(W^i) = F_{kj}(F_{ik}(W^i)). \tag{3.6}$$

Proof:

Since F_{kj} normalizes Ω_k into Ω_j, we have

$$| F_{kj}(W^k) | = |W^j| \tag{3.7}$$

that is W^k is divided into $|W^j|$ clusters by F_{kj} . Denote the r-th cluster of $F_{kj}(W^k)$ as $F_{kj}^r(W^k)$.

According to the Lemma above we have

$$\sum_{w_i \in F_{kj}^r(W^k)} p_i = p_r, \quad r = 1,2,\ldots,|W^j|, \tag{3.8}$$

This means that F_{kj} is a one-to-one mapping between sets $\{F_{kj}^r(W^k) | r = 1,2,\ldots,|W^j|\}$ and $\{w_r | r = 1,2,\ldots,|W^j|\}$. Therefore we call $F_{kj}^r(W^k)$ and w_r as probability-equivalent images, and define them as

$$F_{kj}^r(W^k) \leftarrow p \rightarrow w_r, \tag{3.9}$$

They can substitute each other in the sense of probability equivalence.

Similarly, we have

$$\sum_{w_l \in F_{kj}^t(W^i)} p_l = p_t, \quad t = 1,2,\ldots, |W^k|, \tag{3.10}$$

and

$$F_{ik}^t(W^i) \leftarrow p \rightarrow w_t, \tag{3.11}$$

where $F_{ik}^t(W^i)$ is the t-th cluster of $F_{ik}(W^i)$, $w_t \in W^k$.

On the other hand, we know that

$$\begin{aligned} F_{kj}^r(W^k) &= \{w_t\}, \\ F_{ik}^t(W^i) &= \{w_l\}, \end{aligned} \tag{3.12}$$

where $r = 1,2,\ldots, |W^j|$, $t = 1,2,\ldots, |W^k|$, $l = 1,2,\ldots, |W^i|$.

Therefore, we have

$$\begin{aligned} w_r \leftarrow p \rightarrow F_{kj}^r(W^k) &\models w_r \leftarrow p \rightarrow \{w_t\} \\ &\models w_r \leftarrow p \rightarrow \{F_{ik}^t(W^i)\} \\ &\models w_r \leftarrow p \rightarrow \{\{w_l\}\} \\ &\models w_r \leftarrow p \rightarrow F_{kj}^r(F_{ik}^t(W^i)), \end{aligned} \tag{3.13}$$

whereby

$$|F_{kj}^r(F_{ik}^t(W^i))| = |W^j|. \tag{3.14}$$

Because of (3.8) and (3.10) we can have

$$\begin{aligned} H_{W^j} &= H_{F_{kj}(W^k)} \\ &= -\sum_{r=1}^{|W^j|} p_r \log p_r \end{aligned}$$

$$= -\sum_{r=1}^{|W^j|}\Big(\sum_{t=1}^{|W^k|} p_t\Big)\log\Big(\sum_{t=1}^{|W^i|} p_t\Big)$$

$$= -\sum_{r=1}^{|W^j|}\Big(\sum_{t=1}^{|W^k|}\Big(\sum_{l=1}^{|W^i|} p_l\Big)\Big)\log\Big(\sum_{t=1}^{|W^k|}\Big(\sum_{l=1}^{|W^i|} p_l\Big)\Big)$$

$$= H_{F_{kj}(F_{ik}(W^i))}. \tag{3.15}$$

Based on the definition of $F_{ij}(W^i)$, we know that

$$F_{ij}(W^i) = F_{kj}(F_{ik}(W^i)). \tag{3.16}$$

That is the theorem has been proved.

Example 3.2 :

An example which illustrates Theorem 3.1 is shown in Fig. 3.1. To normalize Ω_i to Ω_j, we use F_{ik} first normalizing W^i to a smaller set W^k, which will be then normalized to a much more smaller set W^j by F_{kj}.

Theorem 3.2 :

Given $\Omega_U = \bigcup_{i=1}^{m}\Omega_i$. $\Omega_i = (W^i, P_{W^i}, H_{W^i})$ and the reference $\Omega_j = (W^j, P_{W^j}, H_{W^j})$ are known , and $|W^j| \leq \min(|W^i|)$, where i = 1, 2, ..., m. If for all Ω_i we can find an entropy-reduced transformation which normalizes Ω_i to Ω_j , then we can find an entropy-reduced transformation F which normalizes Ω_U to Ω_j .

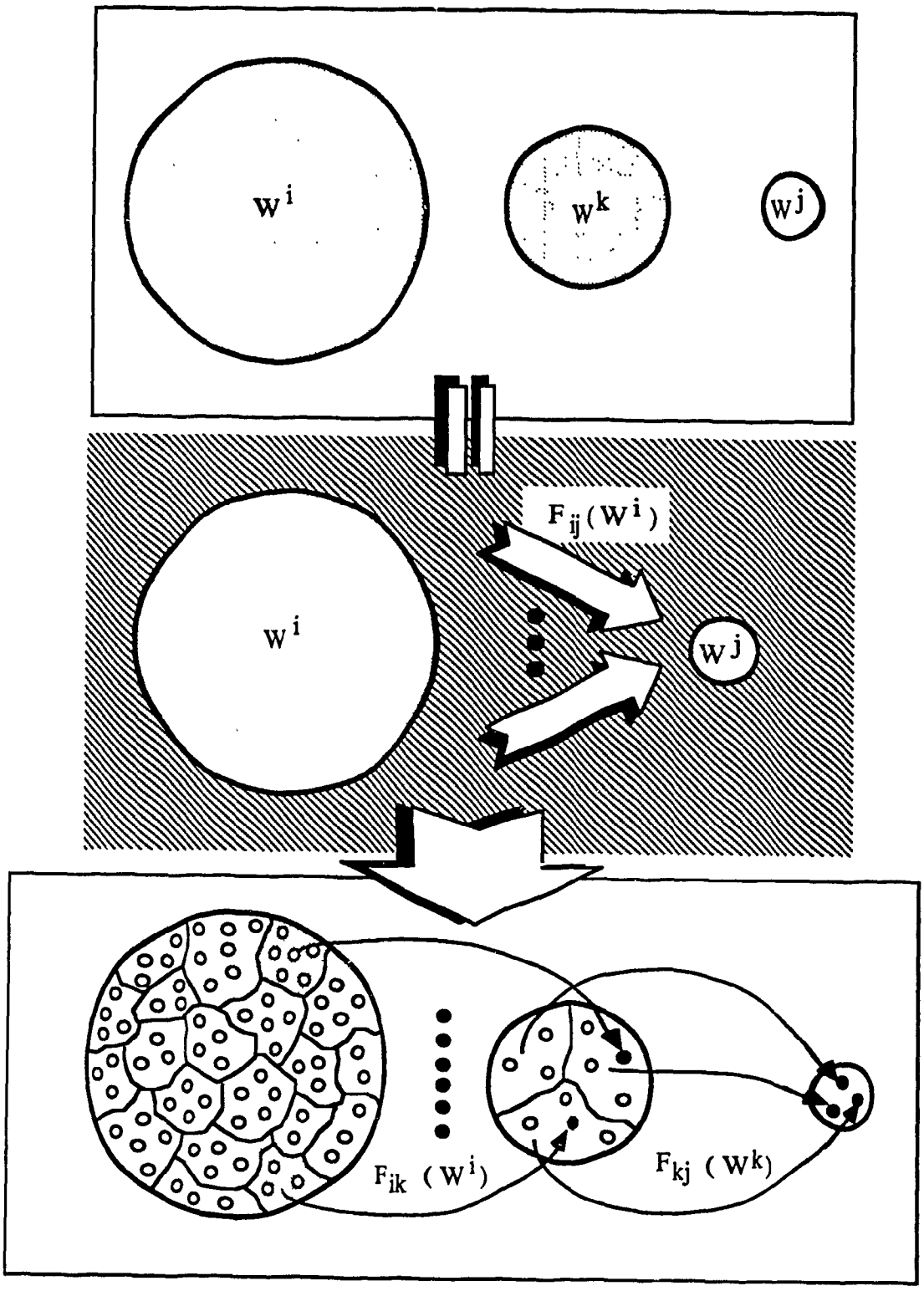


Fig. 3.1 Illustration of the Theorem 3.1

Proof:

According to the Lemma above we know that for the Ω_i, i = 1, 2, ..., m, we have

$$p_{ik} = \sum_{w_v \in F_{ij}^t(W^i)} p_{il} = p_{jk}, \quad k = 1,2,\ldots, |W^j|, \tag{3.17}$$

where p_{ik} stands for the apriori probability of an element of W^i falling into the k-th cluster of $F_{ij}(W^i)$, p_{il} is the apriori probability of the l—th element of W^i, p_{jk} stands for the apriori probability of the k-th element of W^j.

Then every element of U is examined from the global point of view. Because after normalization U can be considered as a set of $|W^j|$ elements and the apriori probability of the t-th element is determined by the following equation

$$p_t = \sum_{i=1}^{m} p_{it}\ p_i, \tag{3.18}$$

where p_i stands for the probability of an element falling into W^i, $\sum_{i=1}^{m} p_i = 1$, p_{it} stands for the apriori probability of an element falling into the t-th cluster of $F_{ij}(W^i)$, which is determined by equation (3.17).

Therefore we can have

$$\begin{aligned} H_U &= -\sum_{t=1}^{|W^j|} p_t \log p_t \\ &= -\sum_{t=1}^{|W^j|} \left(\sum_{i=1}^{m} p_{it}\ p_i\right) \log\left(\sum_{i=1}^{m} p_{it}\ p_i\right) \\ &= -\sum_{t=1}^{|W^j|} \left(\sum_{i=1}^{m} p_{jt}\ p_i\right) \log\left(\sum_{i=1}^{m} p_{jt}\ p_i\right) \end{aligned}$$

$$= -\sum_{t=1}^{|W^j|}(\sum_{i=1}^{m} p_i)p_{jt}\log(\sum_{i=1}^{m} p_i)p_{jt}$$

$$= -\sum_{t=1}^{|W^j|} p_{jt}\log p_{jt}$$

$$= H_{W^j}. \tag{3.19}$$

Obviously we can define F as a multiple branch complex function such that

$$F = \{ F_{ij} \mid i = 1,2,\ldots,m \}. \tag{3.20}$$

That is the theorem has been proved.

Example 3.3 :

An example which illustrates the principle of Theorem 3.2 is shown in Fig. 3.2. To normalize set $\{ W^1, W^2, W^3, W^4 \}$ to a smaller set W^0, we use different transformations $F_{10}, F_{20}, F_{30}, F_{40}$ for the different subset W^1, W^2, W^3, W^4.

We can call Theorem 3.1 the cascade principle and Theorem 3.2 the parallel principle. They bring to light two basic ways to reduce the uncertainty of a complex target pattern set. Obviously, if both of them are used together, more complicated problems can be solved. Although these theorems impose some crucial principle to design an efficient discriminator to recognize a complex target pattern set, here we would like to stress another important consequence implied by these theorems as follows.

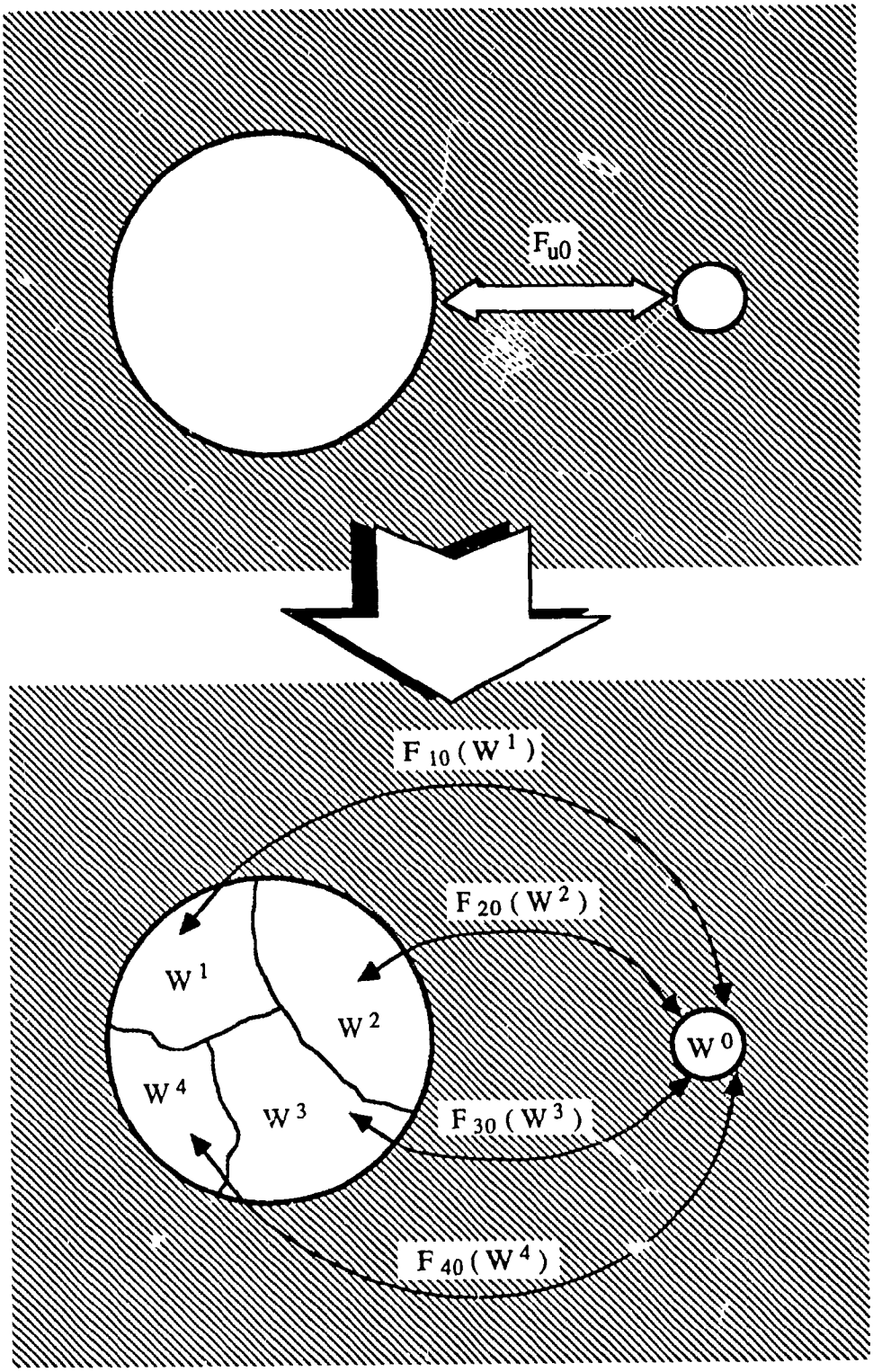


Fig. 3.2 Illustration of the Theorem 3.2

Theorem 3.3 :

Let $U = \bigcup_{i=1}^{m} W^i$. D is a well-designed discriminator for $\Omega_j = (W^j, P_W{}^j, H_W{}^j)$, where $W^j \subset U$ and $| W^j | = \min(| W^i |)$, i, j = 1, 2, ..., m. If Ω_U can be normalized to Ω_j by entropy-reduced transformation according to either Theorem 3.1 or Theorem 3.2 or both of them, then Ω_U can be discriminated by D.

3.4 ALGORITHM FOR CONSTRUCTING A PATTERN RECOGNITION SYSTEM

According to the theorems listed above, a pattern recognition system in practice, can be constructed by two kinds of basic operations : cascade-union denoted by ∇_c and parallel-union denoted by ∇_p. The functions of them are shown in Fig. 3.3.

Two ERT's can form a cascade component by using cascade-union if they obey Theorem 3.1. Two or more ERT's can form a parallel component by using parallel-union if they obey Theorem 3.2. It is useful to stress that every parallel component has a part called "switch" which is an algorithm determined by the practical application. According to Theorem 3.3, cascade-union and parallel-union can be used recursively and mixed.

Generally speaking, corresponding to the MLIS, there are three basic components in a pattern recognition system : F_I is the component consisting of the ERT's for intrinsic characteristics (i.e. IS1), F_D is the component consisting of the ERT's for various kinds of digitization distortions and transducing distortions (i.e. IS2), F_E is the component consisting of the ERT's for feature extraction (i.e. IS4). All of them can be formed by using the operations above. If we denote the classifier by F_C, the entire pattern recognition system for a given target set U, which is denoted by F_U, can be represented as follows :

$$F_U = F_I \nabla_c F_D \nabla_c F_E \nabla_c F_C$$

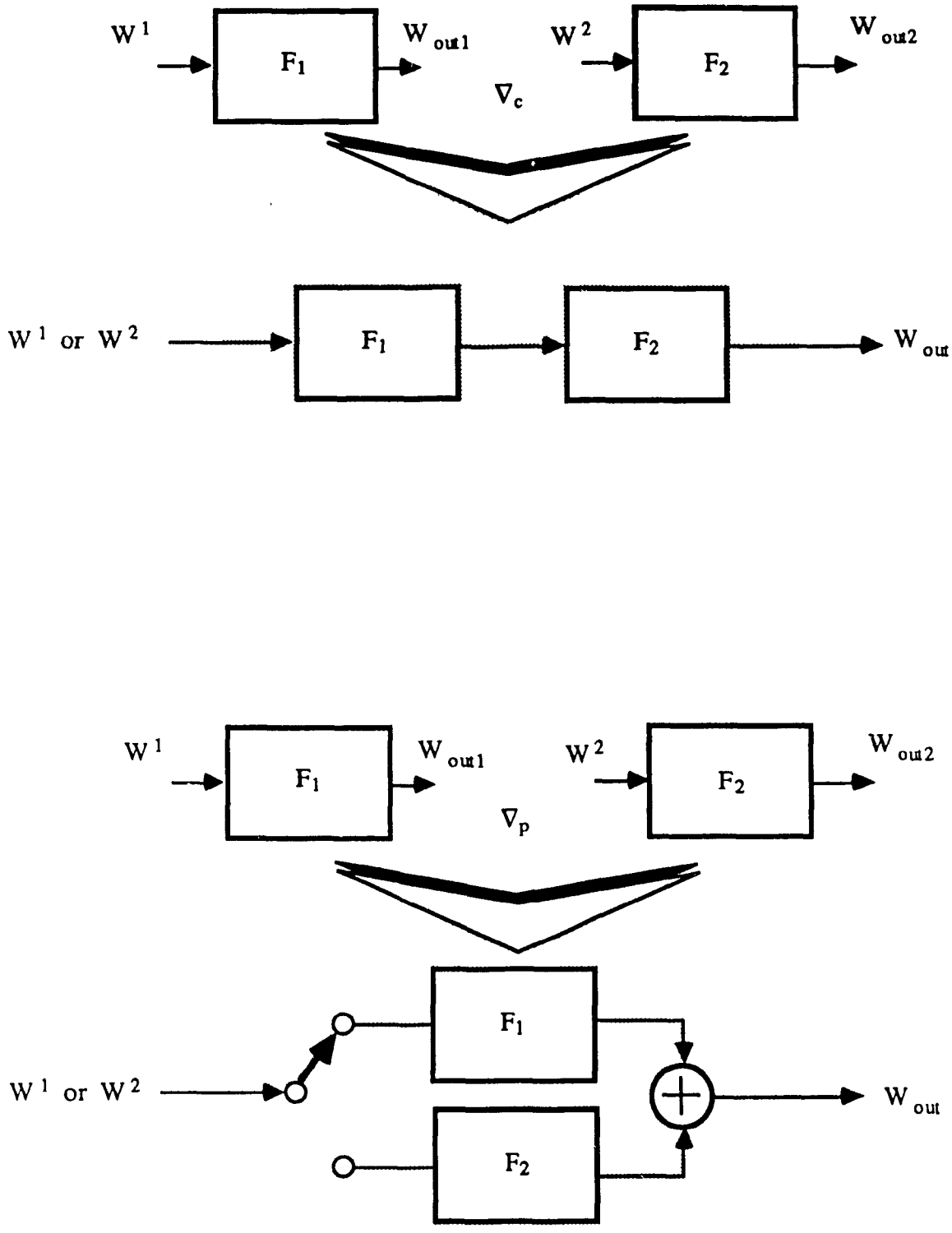


Fig. 3.3 Definition of ∇_c and ∇_p

In practice it is possible to have multiple choices to decompose a given target set in order to have multiple possibilities to combine ERT's. Also it is possible to have different ERT's for solving the same problem. Therefore a computer-aided constructing algorithm is preferable. The following is a practical algorithm for this purpose which is based on the principle of optimization through " maximum entropy reduction ".

Algorithm 3.1 :

{* Input U : a target pattern set .

Output F_U: a pattern recognition system architecture. *}

BEGIN

Step-1

: If necessary decompose U into two. For each of them do the following steps.

Step-2

: If possible then choose a reference set W^0 else Goto Step-1 to redecompose U.

Step-3

: Distinguish U into m subsets if U has m kinds of intrinsic characteristics. That is $U = \bigcup_{i=1}^{m} W^i$, where W^i is a subset corresponding to the i-th kind of intrinsic characteristics. Then start to find F_{i0}'s for W^i's.

For i = 1 to m Do

Begin

If it can find an F_{i0}

Then Goto label

Else

If it can find a chain of F_{i0^1} , $F_{0^1 0^1}$, ..., $F_{0^* 0}$

Then $F_{i0} \leftarrow F_{0^1 0^1} \nabla_c \cdots \nabla_c F_{0^* 0}$

Else

If there is a $W^{0'} \supset W^0$ and $F_{i0'}$ exists

Then

Goto Step-2 to reselect W^0;

label: End.

Step-4

: Begin

For j = 2 To m Do

Begin

If $W^1 \cap W^j = \varnothing$

Then $F_{10} \leftarrow F_{10} \nabla_p F_{j0}$

Else $F_{10} \leftarrow F_{1j} \nabla_c F_{j0}$;

$W^1 \leftarrow W^1 \cup W^j$;

End ;

$U \leftarrow W^1$;

$F_I \leftarrow F_{10}$

End.

Step-5

: Find the appropriate ERT's to resist noise denoted by F_N.

Step-6

: Find ERT's to recover distortions denoted by F_D.

Step-7

: Similarly find suitable ERT's to extract features denoted as F_E.

Step-8

: Design a classifier F_C which is suitable for W^0.

Step-9

: $F_{U^1} \leftarrow F_D \ \nabla_c \ F_I \ \nabla_c \ F_E \ \nabla_c \ F_C$

Step-10

: $F_U \leftarrow \forall k(\nabla_p \ F_{U^1})$.

END { Algorithm }

CHAPTER 4

AN IMPORTANT ENTROPY TRANSFORMATION : IMAGE TRANSFORMATION

4.1 INTRODUCTION

Image transformation is one of the most important entropy transformations. It has two completely different characteristics: entropy increased and reduced characteristics which have been used in many image processing and pattern recognition problems [Cappel86, Devijv82, Fu80, Gans69, Gonzal87, Stark82, Yarosl79, Young86]. According to these properties it performs at least two different functions. One of which uses the entropy increased property to produce a variety of samples from a given image for different purposes, e.g. it can produce test images to measure the performance of a proposed or existing image processing system, and to train and test a classification system. By using an appropriate transformation, we can produce a huge number of test samples in a very short time. This can not be done using type settings or hand drawings without a great deal of effort. For example, we have used this method to produce more than sixty thousand test samples in few days. Another function of image transformation is the use of the entropy reduced property to convert a given image into another one which can be processed more easily or with a better result. An example of this is

image normalization which plays a major role in pattern matching, and is frequently used in image processing and pattern recognition.

Image transformation may also be applied, potentially, to a great variety of topics, such as

1) Computer graphics and animation [Foley82, Hearn86, Reicha72, Salmon87];
2) Computer Aided Design / Computer Aided Manufacturing (CAD/CAM) [Barnhi74, Machov80, Prince71];
3) Computer vision and robot vision [Ballar82, Brady81, Critch85, Hall82];
4) Other applications such as digitization, preprocessing, filtering, restoration, reconstruction, segmentation, feature description, approximation of lines, curves and surfaces, etc [Li89c].

The basic concepts of entropy increased and reduced properties are given in the next sections. Section 4.3 presents linear image transformation and its new algorithms which have been done in our earlier work [Cheng89]. Five important nonlinear transformations, bilinear, quadratic, bi-quadratic, cubic and bi-cubic, are described in Sections 4.4, 4.5 and 4.6 respectively, and also refer to [Li89c, Tang88a].

4.2 ENTROPY INCREASED AND REDUCED TRANSFORMATIONS

Let us consider the triple $\Omega = (W, P_W, H_W)$, an entropy space, where $W = \{ w_i \mid i = 1,2,...,n \}$, is a given data set, P_W is the apriori probability defined on W such that

$$P_W(w_i) = p_i, \qquad 0 < p_i < 1, \qquad \sum_{i=1}^{n} p_i = 1. \tag{4.1}$$

H_W is the uncertainty measure defined on W [Shanno48a, b]

$$H_W = H\ (\ p_1, p_2, \cdots p_n\ ) = -\sum_{i=1}^{n} p_i \log p_i. \tag{4.2}$$

A pattern set can be represented as

$$U = \bigcup_{i=1}^{m} W^i. \tag{4.3}$$

In its entropy space, P_U is determined by the relationships among the subsets W^i's (i = 1,2,...,m).

Definition 4.1

Let $\Omega_i = (W^i, P_{W^i}, H_{W^i})$, $\Omega_j = (W^j, P_{W^j}, H_{W^j})$, $i \neq j$. $|W^i|$ and $|W^j|$ stand for the numbers of elements (i.e. pattern samples) contained in W^i and W^j respectively. F_{ji} is a mapping of W^j into W^i such that

$$(i) \quad \forall_{W_l{}^i, W_k{}^i \subset W^i, W_l{}^j, W_k{}^j \subset W^j, l \neq k} \left[\begin{array}{l} W_l{}^i \cap W_k{}^i = \varnothing \\ W_l{}^j \cap W_k{}^j = \varnothing \end{array} \right]$$

$$(ii) \quad |F_{ji}(W^j)| = |W^i|,$$

$$(iii) \quad H_{F_{ji}(W^j)} = H_{W^i}. \tag{4.4a}$$

If

$$|W^i| < |W^j| \tag{4.4b}$$

then F_{ji} is called an entropy-reduced transformation (ERT). We say that Ω_j is normalized to Ω_i by F_{ji}. W^i is called the reference set. Otherwise, if

$$|W^i| > |W^j| \tag{4.4c}$$

then F_{ji} is called an entropy-increased transformation (EIT).

An important property of these transformations is presented below.

Theorem 4.1

Let F be an entropy-increased transformation from W^j to W^i, and assume F is injective and surjective, i.e.

$$FW_r{}^j \neq FW_s{}^j \qquad \text{for } r \neq s$$

and

$$\forall_{W_k \in W^i,\ W_l \in W^j} (F\ W_l = W_k) .$$

If its inverse transformation F^{-1} exists, then F^{-1} will be an entropy-reduced one.

Proof

Since F is an entropy-increased transformation, so Eqs. (4.4a) and (4.4c) are satisfied. Suppose F maps W^j to W^i, that is

$$W^i = F\ W^j$$

If F^{-1} exists, then we have

$$W^j = F^{-1} \; W^i. \tag{4.5}$$

According to Eq. (4.4c), $| W^i | > | W^j |$. Therefore F^{-1} is an entropy-reduced transformation in Eq. (4.5).

This significant property will be employed to remove the uncertainty from a recognition system which will be illustrated in the next two chapters.

4.3 LINEAR IMAGE TRANSFORMATION

Translation, scaling, rotation and their combinations are the most common types of transformations. Mathematically, they can be represented in the following form:

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} = \begin{bmatrix}A & B\\ C & D\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix} + \begin{bmatrix}E\\ F\end{bmatrix} \tag{4.6}$$

where:

$x, y, \xi, \eta \in R(real\ set)$ and $A, B, C, D, E, F \in R(real\ set)$

x, y - original coordinates of the $x0y$ plane

ξ, η - new coordinates of the $\xi\ 0\ \eta$ plane.

It will be one-to-one mapping, if

$$|T| = \begin{vmatrix}A & B\\ C & D\end{vmatrix} \neq 0$$

In order to obtain the desired transformation, we have to find the parameters A, B, C, D, E and F by solving six linear equations. It means that we have to find three pairs of corresponding points:

$$\left\{(x_1\ y_1)\ (x_2\ y_2)\ (x_3\ y_3)\right\} \subset \text{xoy} \quad \text{and} \quad \left\{(\xi_1\ \eta_1)\ (\xi_2\ \eta_2)\ (\xi_3\ \eta_3)\right\} \subset \xi\ O\ \eta$$

We will have

$$\begin{bmatrix}\xi_1\\ \eta_1\\ \xi_2\\ \eta_2\\ \xi_3\\ \eta_3\end{bmatrix} = \begin{bmatrix}x_1 & y_1 & 0 & 0 & 1 & 0\\ 0 & 0 & x_1 & y_1 & 0 & 1\\ x_2 & y_2 & 0 & 0 & 1 & 0\\ 0 & 0 & x_2 & y_2 & 0 & 1\\ x_3 & y_3 & 0 & 0 & 1 & 0\\ 0 & 0 & x_3 & y_3 & 0 & 1\end{bmatrix}\begin{bmatrix}A\\ B\\ C\\ D\\ E\\ F\end{bmatrix} = T^{*}\begin{bmatrix}A\\ B\\ C\\ D\\ E\\ F\end{bmatrix} \tag{4.7}$$

and

$$\begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \end{bmatrix} = T^{*-1} \begin{bmatrix} \xi_1 \\ \eta_1 \\ \xi_2 \\ \eta_2 \\ \xi_3 \\ \eta_3 \end{bmatrix} \tag{4.8}$$

The necessary and sufficient condition for Eq. (4.8) to have a non-trivial solution is $|T^*| \neq 0$. By choosing some special point pairs, we can solve Eq. (4.8) more easily by first simplifying T^*.

In Eq. (4.6), if $A = D = 1$ and $B = C = 0$, it will become a translation mapping as shown in Fig. 4.1. If $A = D = S$ and $B = C = E = F = 0$, then Eq. (4.6) will perform a scaling mapping as shown in Fig. 4.2 and Eq. (4.6) becomes

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

In this case, either x(ξ) or y(η) will enable us to find the parameter s. $s > 1$ and $s < 1$ correspond to the magnification and reduction of the given image respectively. In Eq. (4.6), if $A = D = \cos\theta$, $B = -\sin\theta$, $c = \sin\theta$ and $E = F = 0$, then Eq. (4.6) represents a rotation mapping as shown in Fig. 4.3.

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

If we wish to find the parameter θ, one pair of corresponding points (x y) and (ξ

η) will be enough.

As indicated earlier, all of the transformations will be one-to-one mapping in the real number domain provided $|T| \neq 0$. This will no longer be true when these transformations are applied to digitized images due to the discrete nature of their representations. In this case, we have to switch from the real number domain to the integer domain. Many researchers have worked on this subject [Lee87, Lipkin70, Li89c, Pavlid82]. In this Chapter, we propose a new transformation approach which can be performed in the integer domain with much better result and much lower time complexity. The details will be presented in the next section.

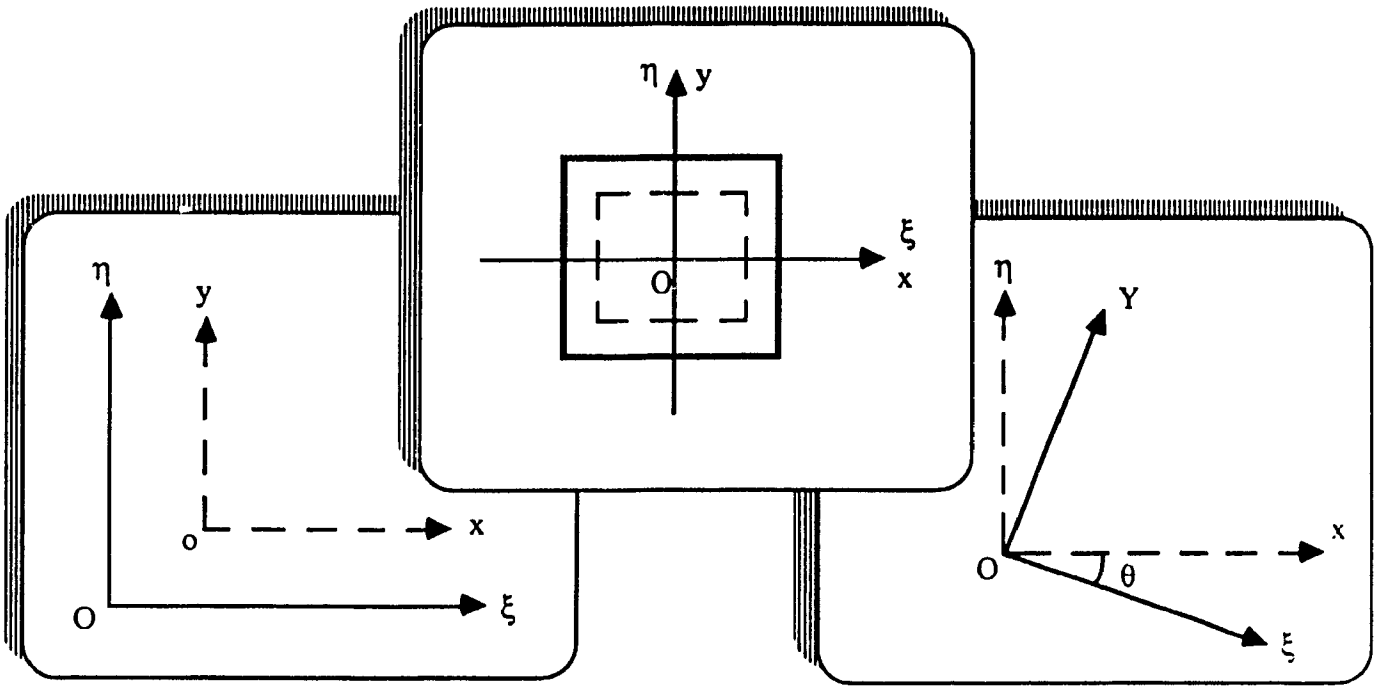


Fig. 4.1 Translation

Fig. 4.2 Scaling

Fig. 4.3 Rotation

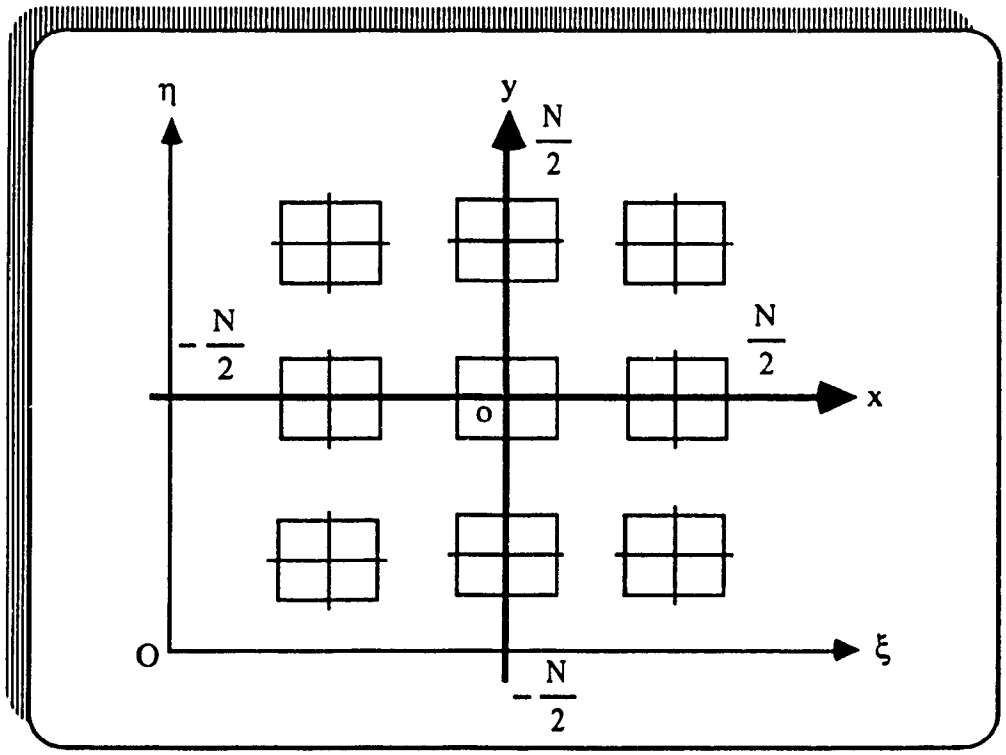


Fig. 4.4 Image Plane

4.3.1 New Algorithms

4.3.1.1 Preliminary Discussion

Let us define an image plane as shown in Fig. 4.4. Before presenting the new algorithms, we first give the following definitions to facilitate our discussions.

Definition 4.2

For a binary image P and a pixel (x y) of the image plane, where x, y $\in$ I, let G(x y) be the grey level of the pixel, then

$$G(x\ y) = \begin{cases} 1 & \text{if } (x\ y) \in P \quad (image) \\ 0 & \text{otherwise} \qquad (background) \end{cases}$$

Definition 4.3

The 4-neighbors of a pixel (x y) will be defined by the set of pixels {(x+1 y) (x-1 y) (x y+1) (x y-1)} and denoted by N_4(x y).

Definition 4.4

The 4-diagonal-neighbors of a pixels (x y) will be defined by the set of pixels {(x+1 y+1) (x+1 y-1) (x-1 y+1) (x-1 y-1)} and denoted by N_D(x y).

Definition 4.5

The adjacent vector of a given pixel (x y), is described by V(x y) which contains the gray levels of its neighbors. V(x y) = (G(x-1 y) G(x-1 y+1) G(x y+1) G(x+1 y+1)).

For the digitized images,

$$\xi, \eta, x, y \in I \text{ (integer set)}$$

$$A, B, C, D, E, F \in R \text{ (real set)}$$

1) Translation

$$A = D = 1$$

$$B = C = 0$$

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} E \\ F \end{bmatrix} + \begin{bmatrix} \frac{N+2}{2} \\ \frac{N+2}{2} \end{bmatrix} \tag{4.9}$$

It is one-to-one mapping and causes no problem at all.

Where:

N is the size of the image,

$\begin{bmatrix} \frac{N+2}{2} \\ \frac{N+2}{2} \end{bmatrix}$ means that after mapping the image is moved to the origin of the image plane which can be shown in Fig. 4.4.

2) Scaling

(s is the scaling factor)

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{N+2}{2} \\ \frac{N+2}{2} \end{bmatrix} \tag{4.10}$$

When s < 1, it will become many-to-one mapping and will not produce any problem. There is no need to do anything other than mapping the pixels according to Eq. (4.10). The same for $s = 1$. However, when s > 1, it will be one-to-one mapping with the possibility of breaking the connectivity of the pixels. This will produce undesirable "measles" [Lee87, Lipkin70, Li89c, Pavlid82] in the resulting image which should be removed.

3) Rotation

$(0 \leq \theta \leq 2\pi)$

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \left\lfloor \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{N+2}{2} \\ \frac{N+2}{2} \end{bmatrix} \right\rfloor \tag{4.11}$$

Depending on the values of x, y and θ, it could be many-to-one mapping or one-to-one mapping. It may also destroy the connectivity of the originally connected pixels and create the "measles" problem.

4.3.1.2 New Transformation Algorithms

1) Translation Algorithm

Translation can be done easily by using the following algorithm:

Algorithm 4.1

Translation Algorithm

For all (x y) {(x y) | G(x y) = 1} do

begin

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} = \left\lfloor \begin{bmatrix}x\\ y\end{bmatrix} - \begin{bmatrix}E\\ F\end{bmatrix} + \begin{bmatrix}\frac{N+2}{2}\\ \frac{N+2}{2}\end{bmatrix} \right\rfloor$$

$$G(\xi\ \eta) = G(x\ y)$$

end

2) Scaling Algorithm

As mentioned earlier, if $s < 1$, mapping is simply done according to Eq. (4.10). If $s > 1$, the "measles" have to be eliminated by using a one-to-many mapping approach, i.e., G(x y) should be assigned to more than one pixel to fill up the positions occupied by the measles. The algorithm consists of sixteen subroutines invoked by the value of V(x y) which can be shown in Fig. 4.5. Details are described below.

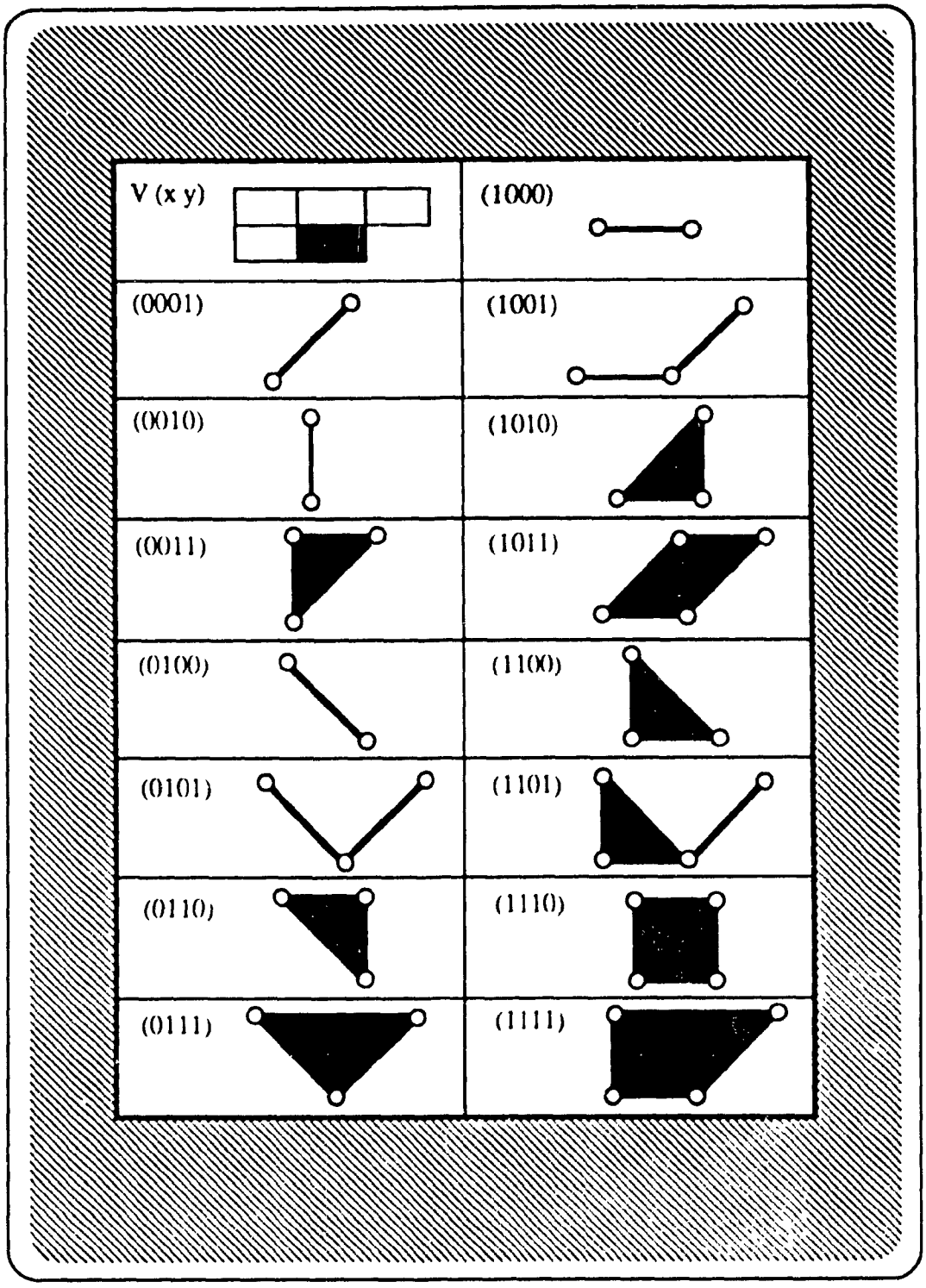


Fig. 4.5 Subroutines for Scaling Mapping

Algorithm 4.2

Scaling Algorithm

For all pixels $\in$ P, we examine their adjacent vectors and perform mapping and filling at the same time.

For all (x y) {(x y) | G(x y) = 1} do

begin

/ compute the adjacent vector /

compute V(x y) = (G(x-1 v) G(x-1 y+1) G(x y+1) G(x+1 y+1))

/ perform mapping /

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \left\lfloor \begin{bmatrix} s\ x + \frac{N+2}{2} \\ s\ y + \frac{N+2}{2} \end{bmatrix} \right\rfloor$$

/ Assign the gray level /

G(ξ η) = G(x y)

/ subroutine 1 - S_1 invoked by V(x y) = (0001) /

/ mapping and filling the line as shown in Fig. 4.5 /

If V(x y) = (0001) then

begin

$$\xi_D = \left\lfloor s\ (x+1) + \frac{N+2}{2} \right\rfloor$$

for K = 0 to ξ_D - ξ do

$$G(\xi + K \quad \eta + K) = G(xy)$$

end

/ subroutine 2 - S_2 invoked by V(x y) = (0010) /

/ mapping and filling the line as shown in Fig. 4.5 /

If V(x y) = (0010) then

begin

$$\eta_C = \left\lfloor s\,(y+1) + \frac{N+2}{2} \right\rfloor$$

for K = 0 to $\eta_C - \eta$ do

$$G(\xi \quad \eta + K) = G(xy)$$

end

/ subroutine 3 - S_3 invoked by V(x y) = (0011) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (0011) then

begin

$$\begin{bmatrix} \xi_D \\ \eta_D \end{bmatrix} = \left\lfloor \begin{bmatrix} s\,(x+1) + \frac{N+2}{2} \\ s\,(y+1) + \frac{N+2}{2} \end{bmatrix} \right\rfloor$$

for I = 0 to $\xi_D - \xi$ do

for J = $\eta + I$ to η_D do

/ Note $\eta_D - \eta = \xi_D - \xi$ /

$$G(\xi + I \quad J) = G(xy)$$

end

/ subroutine 4 - S_4 invoked by V(x y) = (0100) /

/ mapping and filling the line as shown in Fig. 4.5 /

If V(x y) = (0100) then

begin

$$\xi_B = \left\lfloor s\,(x-1) + \frac{N+2}{2} \right\rfloor$$

for K = 0 to ξ - ξ_B do

G (ξ − K η + K) = G (xy)

end

/ subroutine 5 - S_5 invoked by V(x y) = (0101) /

/ mapping and filling the line as shown in Fig. 4.5 /

If V(x y) = (0101) then

begin

$$\begin{bmatrix} \xi_B \\ \eta_B \end{bmatrix} = \left\lfloor \begin{bmatrix} s\,(x+1) + \frac{N+2}{2} \\ s\,(y+1) + \frac{N+2}{2} \end{bmatrix} \right\rfloor$$

for I = 0 to ξ - ξ_B do

or

for I = 0 to η_B - η do

begin

G (ξ − I η + I) = G (xy)

$$G(\xi + I \quad \eta + I) = G(xy)$$

end

end

/ subroutine 6 - S_6 invoked by V(x y) = (0110)/

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (0110) then

begin

$$\begin{bmatrix} \xi_B \\ \eta_B \end{bmatrix} = \left\lfloor \begin{bmatrix} s\,(x-1) + \dfrac{N+2}{2} \\ s\,(y+1) + \dfrac{N+2}{2} \end{bmatrix} \right\rfloor$$

for I = 0 to $\xi - \xi_B$ do

for J = η + I to η_B do

$$G(\xi - I \quad J) = G(xy)$$

end

/ subroutine 7 - S_7 invoked by V(x y) = (0111) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (0111) then

begin

$$\begin{bmatrix} \xi_B \\ \eta_B \end{bmatrix} = \left\lfloor \begin{bmatrix} s\,(x-1) + \dfrac{N+2}{2} \\ s\,(y+1) + \dfrac{N+2}{2} \end{bmatrix} \right\rfloor$$

$$\begin{bmatrix} \xi_D \\ \eta_D \end{bmatrix} = \left\lfloor \begin{bmatrix} s\,(x+1) + \dfrac{N+2}{2} \\ s\,(y+1) + \dfrac{N+2}{2} \end{bmatrix} \right\rfloor$$

for I = 0 to $\xi - \xi_B$ do

for J = η + I to η_B do

G (ξ − I J) = G (xy)

for I = 0 to $\xi_D - \xi$ do

for J = η + I to η_D do

G (ξ + I J) = G (xy)

end

/ subroutine 8 - S_8 invoked by V(x y) = (1000) /

/ mapping and filling the line as shown in Fig. 4.5 /

If V(x y) = (1000) then

begin

$$\xi_A = \left\lfloor s(x-1) + \frac{N+2}{2} \right\rfloor$$

for I = $\xi_A - \xi$ to 0 do

G (ξ + I η) = G (xy)

end

/ subroutine 9 - S_9 invoked by V(x y) = (1001) /

/ mapping and filling the line as shown in Fig. 4.5 /

If V(x y) = (1001) then

begin

$$\xi_A = \left\lfloor s(x-1) + \frac{N+2}{2} \right\rfloor$$

$$\xi_D = \left\lfloor s(x+1) + \frac{N+2}{2} \right\rfloor$$

for K = 0 to ξ_D - ξ do

G (ξ + K η + K) = G (xy)

for I = ξ_A - ξ to 0 do

G (ξ + I η) = G (xy)

end

/ subroutine 10 - S_{10} invoked by V(x y) = (1010) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (1010) then

begin

$$\xi_A = \left\lfloor s(x-1) + \frac{N+2}{2} \right\rfloor$$

$$\eta_C = \left\lfloor s(y+1) + \frac{N+2}{2} \right\rfloor$$

for I = 0 to ξ - ξ_A do

for J = η to η_C - I do

G (ξ − I J) = G (xy)

end

/ subroutine 11 - S_{11} invoked by V(x y) = (1011) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (1011) then

begin

$$\xi_A = \left\lfloor s(x-1) + \frac{N+2}{2} \right\rfloor$$

$$\xi_D = \left\lfloor s(x+1) + \frac{N+2}{2} \right\rfloor$$

$$\eta_C = \left\lfloor s(y+1) + \frac{N+2}{2} \right\rfloor$$

$$\eta_D = \left\lfloor s(y+1) + \frac{N+2}{2} \right\rfloor$$

for I = 0 to ξ - ξ_A do

for J = η to η_C - I do

G (ξ − I J) = G (xy)

for M = 0 to ξ_D - ξ do

for N = η + M to η_D do

G (ξ + M N) = G (xy)

end

/ subroutine 12 - S_{12} invoked by V(x y) = (1100) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (1100) then

begin

$$\xi_B = \left\lfloor s(x-1) + \frac{N+2}{2} \right\rfloor$$

for I = ξ_B - ξ to 0 do

for J = η to η + | I | do

G (ξ + I J) = G (xy)

end

/ subroutine 13 - S_{13} invoked by V(x y) = (1101) /

/ mapping and filling the line and area as shown in Fig. 4.5 /

If V(x y) = (1101) then

begin

$$\xi_B = \left\lfloor s\ (x-1) + \frac{N+2}{2} \right\rfloor$$

$$\xi_D = \left\lfloor s\ (x+1) + \frac{N+2}{2} \right\rfloor$$

for I = ξ_B - ξ to 0 do

for J = η to η + | I | do

G (ξ + I J) = G (xy)

for K = 0 to ξ_D - ξ do

G (ξ + K η + K) = G (xy)

end

/ subroutine 14 - S_{14} invoked by V(x y) = (1110) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (1110) then

begin

$$\begin{bmatrix} \xi_B \\ \eta_B \end{bmatrix} = \left\lfloor \left[\begin{matrix} s(x-1) + \frac{N+2}{2} \\ s(y+1) + \frac{N+2}{2} \end{matrix} \right] \right\rfloor$$

for I = ξ_B to ξ do

for J = η to η_B do

G (I J) = G (xy)

end

/ subroutine 15 - S_{15} invoked by V(x y) = (1111) /

/ mapping and filling the area as shown in Fig. 4.5 /

If V(x y) = (1111) then

begin

$$\begin{bmatrix} \xi_B \\ \eta_B \end{bmatrix} = \left\lfloor \begin{bmatrix} s\ (x-1) + \dfrac{N+2}{2} \\ s\ (y+1) + \dfrac{N+2}{2} \end{bmatrix} \right\rfloor$$

for I = ξ_B to ξ do

for J = η to η_B do

G (I J) = G (xy)

for M = 0 to $\xi - \xi_B$ do

for N = η + I to η_B do

G (ξ + M N) = G (xy)

end

end

The basic idea of Algorithm 4.2 can be explained more clearly by referring to Fig. 4.5. There we can figure out the following features of the above algorithm.

1) For any V(x, y), at most one subroutine is invoked.

2) All of the subroutines are very simple and the number of operations (N) is quite small.

3) The algorithm will perform mapping and filling at the same time according to the adjacent vector. It makes the filling more meaningful and the computing faster.

4) Each subroutine can be implemented by hardware. The computations are local and independent. Hence the algorithm is suitable for parallel processing and VLSI implementation [Cheng 89]. The more details about VLSI architecture for image processing and pattern recognition refer to [Bowen82, Charot85, Cheng86, Fu84, Nudd85, Offern85, Parker85, Siegel82].

3) Rotation ($0 \leq \theta < 2\pi$) Algorithm

As mentioned before, the rotation transformation is quite complicated. Some pixels may perform many-to-one mapping while others may perform one-to-one mapping. Depending on the values of x, y and θ, it may produce "measles" under some circumstances. Before discussing the rotation algorithm, we need to prove the following theorem.

Theorem 4.2

For any pixel (x y) $\in$ P and its four neighbors N_4(x y) = {(x+1 y) (x-1 y) (x y+1) (x y-1)}, after rotation, the upper bound of distance between (ξ η) and the corresponding mapped neighbor pixels will be $2^{\frac{1}{2}}$.

Proof.

1) Consider (x y) and its neighbor (x+1 y).

According to Eq. (4.11), we have

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} = \left\lfloor \begin{bmatrix} x\cos\theta - y\sin\theta + \frac{N+2}{2} \\ x\sin\theta + y\cos\theta + \frac{N+2}{2}\end{bmatrix} \right\rfloor$$

$$\begin{bmatrix}\xi^*\\ \eta^*\end{bmatrix} = \left\lfloor \begin{bmatrix} (x+1)\cos\theta - y\sin\theta + \frac{N+2}{2} \\ (x+1)\sin\theta + y\cos\theta + \frac{N+2}{2}\end{bmatrix} \right\rfloor$$

Let

$$x\cos\theta - y\sin\theta + \frac{N+2}{2} = I_1 + \alpha \qquad (4.12)$$

and

$$x\sin\theta + y\cos\theta + \frac{N+2}{2} = I_2 + \beta \tag{4.13}$$

where I_1 and I_2 are integers, $0 \leq \alpha < 1$ and $0 \leq \beta < 1$ are real numbers. Then

$$\xi^* - \xi = \left\lfloor I_1 + \alpha + \cos\theta \right\rfloor - \left\lfloor I_1 + \alpha \right\rfloor = \lfloor \alpha + \cos\theta \rfloor$$

$$\eta^* - \eta = \left\lfloor I_2 + \beta + \sin\theta \right\rfloor - \left\lfloor I_2 + \beta \right\rfloor = \lfloor \beta + \sin\theta \rfloor$$

Since $-1 < \alpha + \cos\theta < 2$, so $-1 \leq \lfloor \alpha + \cos\theta \rfloor \leq 1$, and

$-1 < \beta + \sin\theta < 2$, so $-1 \leq \lfloor \beta + \sin\theta \rfloor \leq 1$. We have

$$d = ((\xi^* - \xi)^2 + (\eta^* - \eta)^2)^{\frac{1}{2}} \leq (1 + 1)^{\frac{1}{2}} = 2^{\frac{1}{2}}$$

2) Similarly we can prove the other three pixels of N_4(x y) have the same property.

Q.E.D.

From the theorem, we can conclude that the connectivity of the four neighbors N_4(x y) and (x y) will not be changed in the sense of the eight-neighbor connectivity. Therefore, in rotation transformation, we only consider the 4-diagonal-neighbors N_D(x y) which may create measles.

Algorithm 4.4

Rotation Algorithm

For all pixels $\in$ P. We compute the adjacent vectors and only consider the pixels $\in N_D$(x y) which may create measles.

For all (x y) {(x y) | G(x y) = 1} do

begin

V(x y) = {G(x-1 y) G(x-1 y+1) G(x y+1) G(x+1 y+1)}

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} = \left\lfloor \begin{bmatrix} x\cos\theta - y\sin\theta + \dfrac{N+2}{2} \\ x\sin\theta + y\cos\theta + \dfrac{N+2}{2}\end{bmatrix} \right\rfloor$$

G(ξ η) = G(x y)

/ subroutine 1 - S_{r1} invoked by V(x y) = (0100) /

/ mapping and filling will be performed /

If V(x y) = (0100) then

begin

$$\begin{bmatrix}\xi_B\\ \eta_B\end{bmatrix} = \left\lfloor \begin{bmatrix} (x-1)\cos\theta - (y+1)\sin\theta + \dfrac{N+2}{2} \\ (x-1)\sin\theta + (y+1)\cos\theta + \dfrac{N+2}{2}\end{bmatrix} \right\rfloor$$

CASE of ξ , η , ξ_B, η_B

1) $\xi_B > \xi$ and $\eta_B > \eta$:

for I = 0 to ξ_B - ξ do

G (ξ + I η + I) = G(x y)

2) $\xi_B > \xi$ and $\eta_B < \eta$:

for I = 0 to ξ_B - ξ do

G (ξ + I η - I) = G(x y)

3) $\xi_B < \xi$ and $\eta_B < \eta$:

for I = 0 to $\xi - \xi_B$ do

G (ξ - I η - I) = G(x y)

4) $\xi_B < \xi$ and $\eta_B > \eta$:

for I = 0 to $\xi - \xi_B$ do

G (ξ - I η + I) = G(x y)

5) $\xi_B = \xi$ and $\eta_B > \eta$:

for I = 0 to $\eta_B - \eta$ do

G (ξ η + I) = G(x y)

6) $\xi_B = \xi$ and $\eta_B < \eta$:

for I = 0 to $\eta - \eta_B$ do

G (ξ η - I) = G(x y)

7) $\xi_B > \xi$ and $\eta_B = \eta$:

for I = 0 to $\xi_B - \xi$ do

G (ξ + I η) = G(x y)

8) $\xi_B < \xi$ and $\eta_B = \eta$:

for I = 0 to $\xi - \xi_B$ do

G (ξ - I η) = G(x y)

end

/ subroutine 2 - S_{r2} invoked by V(x y) = (0001) /

/ mapping and filling will be performed /

If V(x y) = (0001) then

begin

$$\begin{bmatrix}\xi_D \\ \eta_D\end{bmatrix} = \left\lfloor \begin{bmatrix}(x+1)cos\theta - (y+1)sin\theta + \frac{N+2}{2} \\ (x+1)sin\theta + (y+1)cos\theta + \frac{N+2}{2}\end{bmatrix} \right\rfloor$$

CASE of ξ , η , ξ_D, η_D

1) $\xi_D > \xi$ and $\eta_D > \eta$:

for I = 0 to ξ_D - ξ do

G (ξ + I η + I) = G(x y)

2) $\xi_D > \xi$ and $\eta_D < \eta$:

for I = 0 to ξ_D - ξ do

G (ξ + I η - I) = G(x y)

3) $\xi_D < \xi$ and $\eta_D < \eta$:

for I = 0 to ξ - ξ_D do

G (ξ - I η - I) = G(x y)

4) $\xi_D < \xi$ and $\eta_D > \eta$:

for I = 0 to ξ - ξ_D do

G (ξ - I η + I) = G(x y)

5) $\xi_D = \xi$ and $\eta_D > \eta$:

for I = 0 to η_D - η do

G (ξ η + I) = G(x y)

6) $\xi_D = \xi$ and $\eta_D < \eta$:

for I = 0 to $\eta - \eta_D$ do

G (ξ η - I) = G(x y)

7) $\xi_D > \xi$ and $\eta_D = \eta$:

for I = 0 to $\xi_D - \xi$ do

G (ξ + I η) = G(x y)

8) $\xi_D < \xi$ and $\eta_D = \eta$:

for I = 0 to $\xi - \xi_D$ do

G (ξ - I η) = G(x y)

end

/ subroutine 3 - S_{r3} invoked by V(x y) = (0101) /

/ mapping and filling will be performed /

If V(x y) = (0101) then

begin

S_{r1}

S_{r2}

end

end

Now let us proceed to discuss parallel transformation algorithms.

Algorithm 4.5

Parallel Translation Algorithm

parfor all (x y) {(x y) | G(x y) = 1} do

begin

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} = \left\lfloor \begin{bmatrix}x\\ y\end{bmatrix} - \begin{bmatrix}E\\ F\end{bmatrix} + \begin{bmatrix}\frac{N+2}{2}\\ \frac{N+2}{2}\end{bmatrix} \right\rfloor$$

$$G(\xi\ \eta) = G(x\ y)$$

end

Algorithm 4.6

Parallel Scaling Algorithm

parfor all (x y) {(x y) | G(x y) = 1} do

parbegin

S_1;

S_2;

S_3;

S_4;

S_5;

S_6;

S_7;

S_8;

S_9;

S_{10};

S_{11};

S_{12};

S_{13};

S_{14};

S_{15}

parend

Algorithm 4.7 Parallel Rotation Algorithm

parfor all (x y) {(x y) | G(x y) = 1} do

parbegin

S_{r1};

S_{r2};

S_{r3};

parend

The advantages of the proposed algorithms are listed below:

1. All of the operations depend only on the local information - adjacent vector.
2. All of the pixels of the entire image plane can perform independent operations in parallel.

3. They do not need to distinguish the boundary pixels and the inside pixels and can perform mapping and filling at the same time. Since finding the boundary and the direction of boundary is very time-consuming [Lee87], the new algorithms will speed up the computation considerably. Also it makes the filling more meaningful. Further details will be discussed later.

4. The results obtained by a) performing scaling first and then rotation and b) by performing rotation first and then scaling, are almost identical. This means that the performance of the proposed algorithms is better than the one described in [Lee87].

5. The proposed algorithms have no difficulty in processing multi-boundary and complicated images, and in handling large scaling factors.

6. The proposed algorithms can be implemented using VLSI technology. This will be discussed in [Cheng89, 90].

4.3.2 Experiments and Results

A series of experiments have been conducted to test the proposed algorithms. By performing the proposed transformations on a variety of images including the alphabetic character, Chinese character, space shuttle and panda pictures, it can be concluded that the algorithms proposed in this section have much better performance compared with others. Details are described below.

With different scaling factors $s = 1.4$, $s = 1.8$, $s = 2.0$, $s = 0.8$ and $s = 0.6$ respectively, Fig. 4.6(a) shows the results by applying the proposed algorithm to the panda picture. Fig. 4.6(b) shows the results by applying the ordinary scaling algorithm to the same picture. Fig. 4.6(c) shows the results of filling the scaled images of Fig. 4.6(b) using the filling algorithm described in [Doyle60, Unger59]. From Fig. 4.6(c), we can conclude that the ordinary scaling and filling algorithms fail to remove the unwanted measles when s becomes larger than 1, and when s becomes smaller ($s < 1$), errors occur also. For instance, when $s = 0.8$ and $s = 0.6$, the hands and eyes of the panda are much distorted from the original.

Fig. 4.7(a) shows the results of applying the proposed algorithm to a circle with the scaling factors $s = 1.5$, $s = 2.0$, $s = 3.0$, $s = 0.8$ and $s = 0.6$ respectively. Fig. 4.7(b) shows the results of applying the ordinary scaling algorithm without filling. Fig. 4.7(c) shows the results after filling. The measles problem can not be solved when $s > 1$.

Fig. 4.8(a) shows the results of applying the proposed algorithm to a Chinese character using the same scales of $s = 1.5$, $s = 2.0$, $s = 3.0$, $s = 0.8$, and $s =$

0.6. Again, Fig. 4.8(b) shows the results of the ordinary scaling algorithm and Fig. 4.8(c) shows the results after filling. As before, it shows that when $s > 1$, the filling algorithm will not work. On the other hand, when $s < 1$, the filling algorithm can produce serious errors as indicated by the arrows.

Following the above procedure and using the same scaling factors, similar results have been obtained in Fig. 4.9.

Fig. 4.10 shows the results after applying the proposed algorithm to the space shuttle image with $s = 1.4$ and $s = 2.0$.

To test the rotation algorithm, the Chinese Character shown in Fig. 4.11 with $\theta = 15°$, $\theta = 30°$, $\theta = 45°$, $\theta = 90°$ and $\theta = -45°$, we have obtained Fig. 4.11(a). Fig. 4.11(b) shows the results by applying the ordinary rotation algorithm to the Chinese character.

In order to observe the combination of rotation and scaling transformations, the proposed algorithms have been applied to the the Chinese character with $s = 2.0$ and $\theta = 15°$, $\theta = 45°$ and $\theta = 90°$ respectively as shown in Fig. 4.12.

The effect of the order in which the rotation and scaling are applied was also investigated. The panda picture was first scaled with $s = 1.5$ and then rotated $30°$. Similarly, the reverse order was made, i.e. rotation first and scaling next. The results are shown in Fig. 4.13. Scaling the Chinese character with $s = 2$ and rotating it by $30°$, and vice versa produced the results displayed in Fig. 4.14. From Figs. 4.13 and 4.14, it can be easily observed that the errors due to the order in which rotation and scaling are applied are much smaller than those

reported in [Lee87].

Hence we can conclude from these experiments that the proposed algorithms will perform more consistent and accurate mappings than those obtained by other known approaches. The better results produced by the new algorithms come primarily from the operations which always check the connectivity of the original image and preserve it by performing mapping and filling at the same time. This makes the filling more meaningful. Also, since the new algorithms do not have to go through the very time-consuming process of finding the boundary and its direction, they are much simpler and faster than those described in [Lee87].

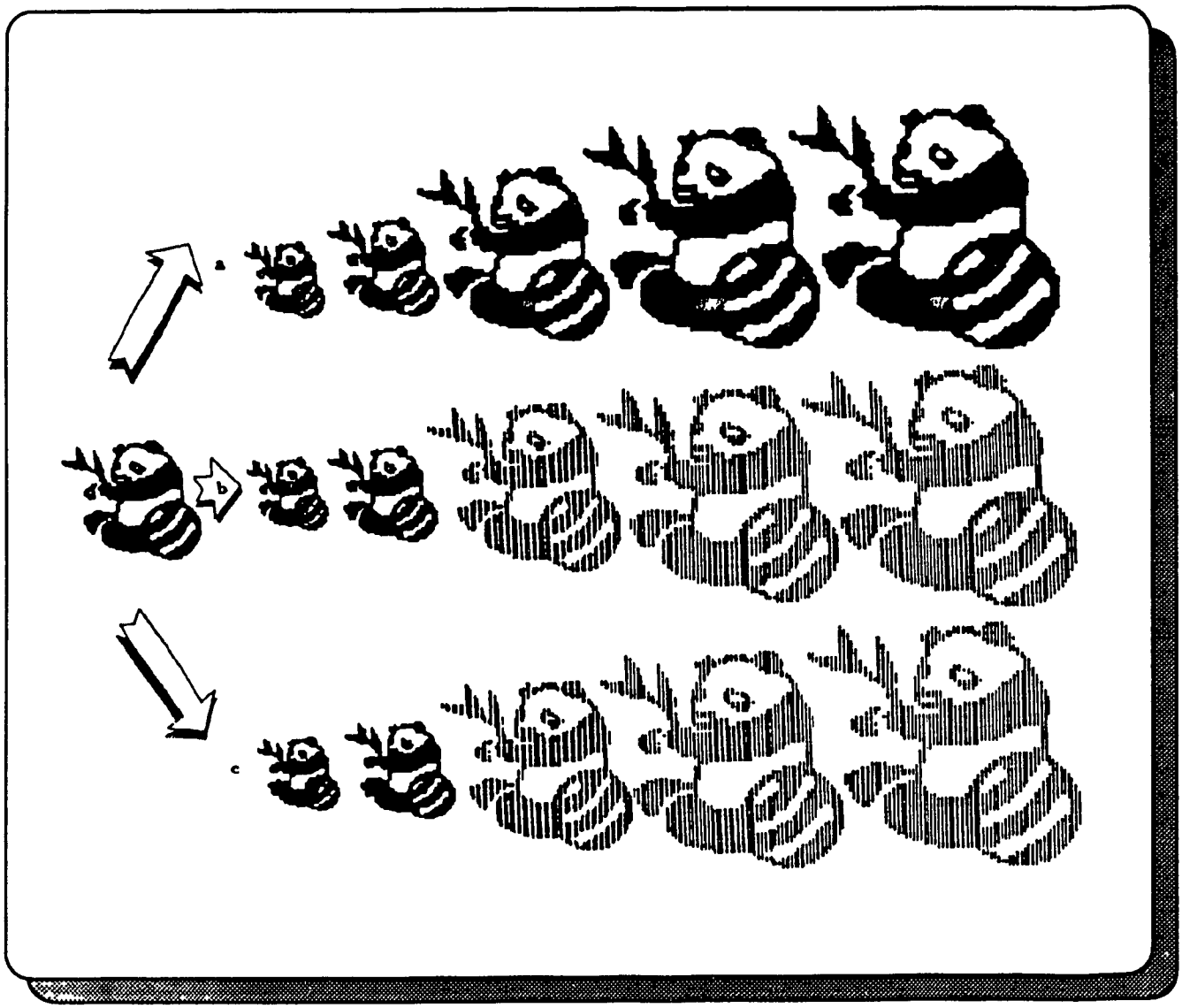


Fig. 4.6 (a) Scaling Results Obtained by the Proposed Algorithm
(b) Scaling Results Obtained by the Ordinary Algorithm
(c) Scaling Results Obtained by Filling Results in (b)

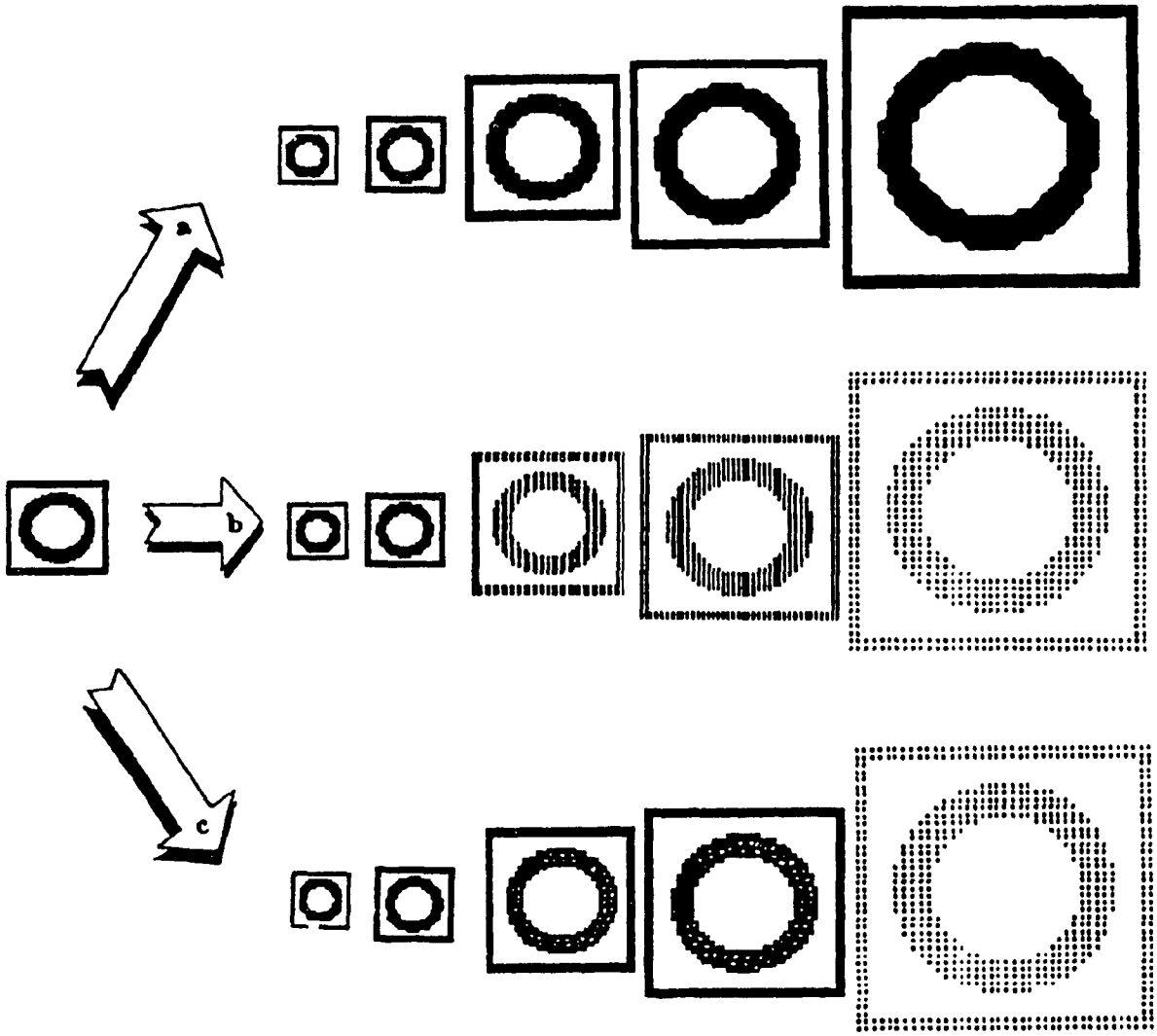


Fig. 4.7 (a) Scaling Results Obtained by the Proposed Algorithm
(b) Scaling Results Obtained by the Ordinary Algorithm
(c) Scaling Results Obtained by Filling Results in (b)

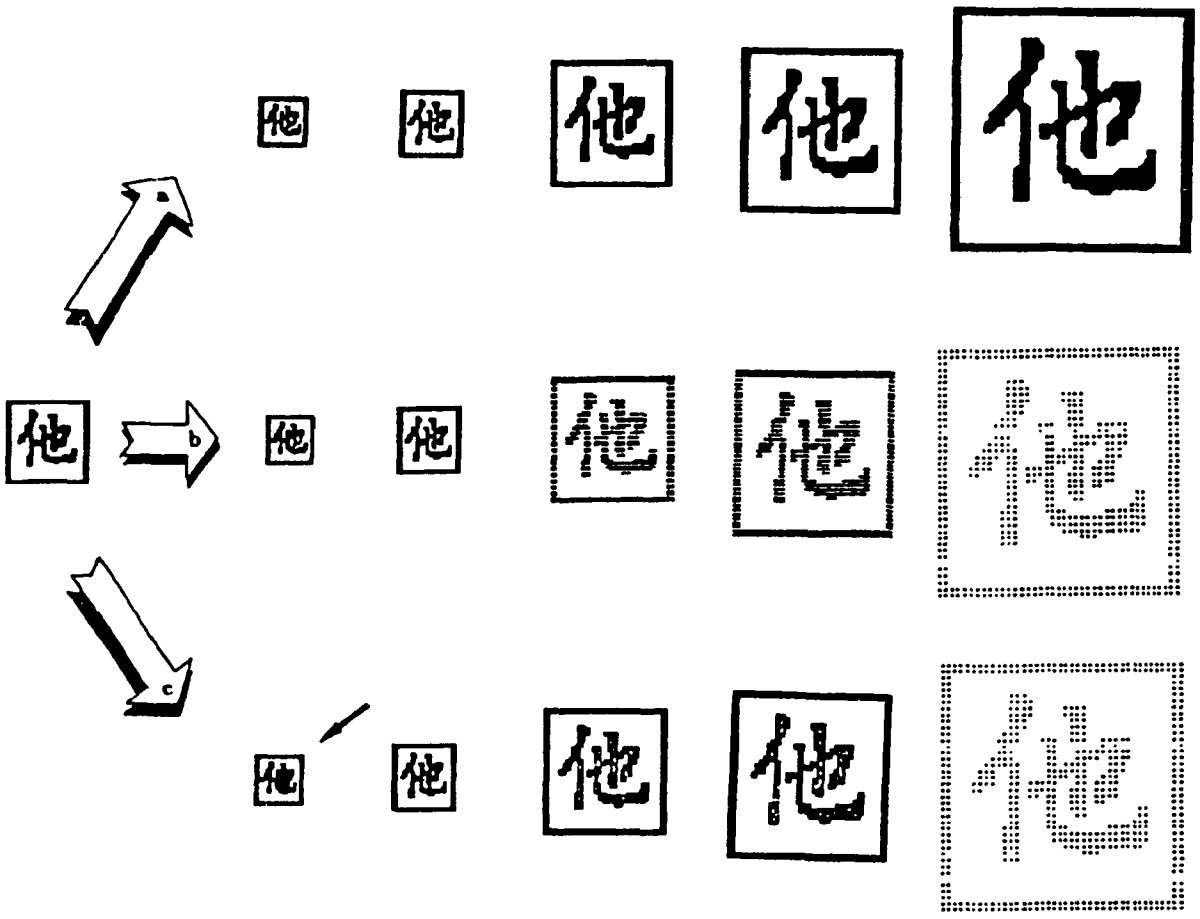


Fig. 4.8 (a) Scaling Results Obtained by the Proposed Algorithm
(b) Scaling Results Obtained by the Ordinary Algorithm
(c) Scaling Results Obtained by Filling Results in (b)

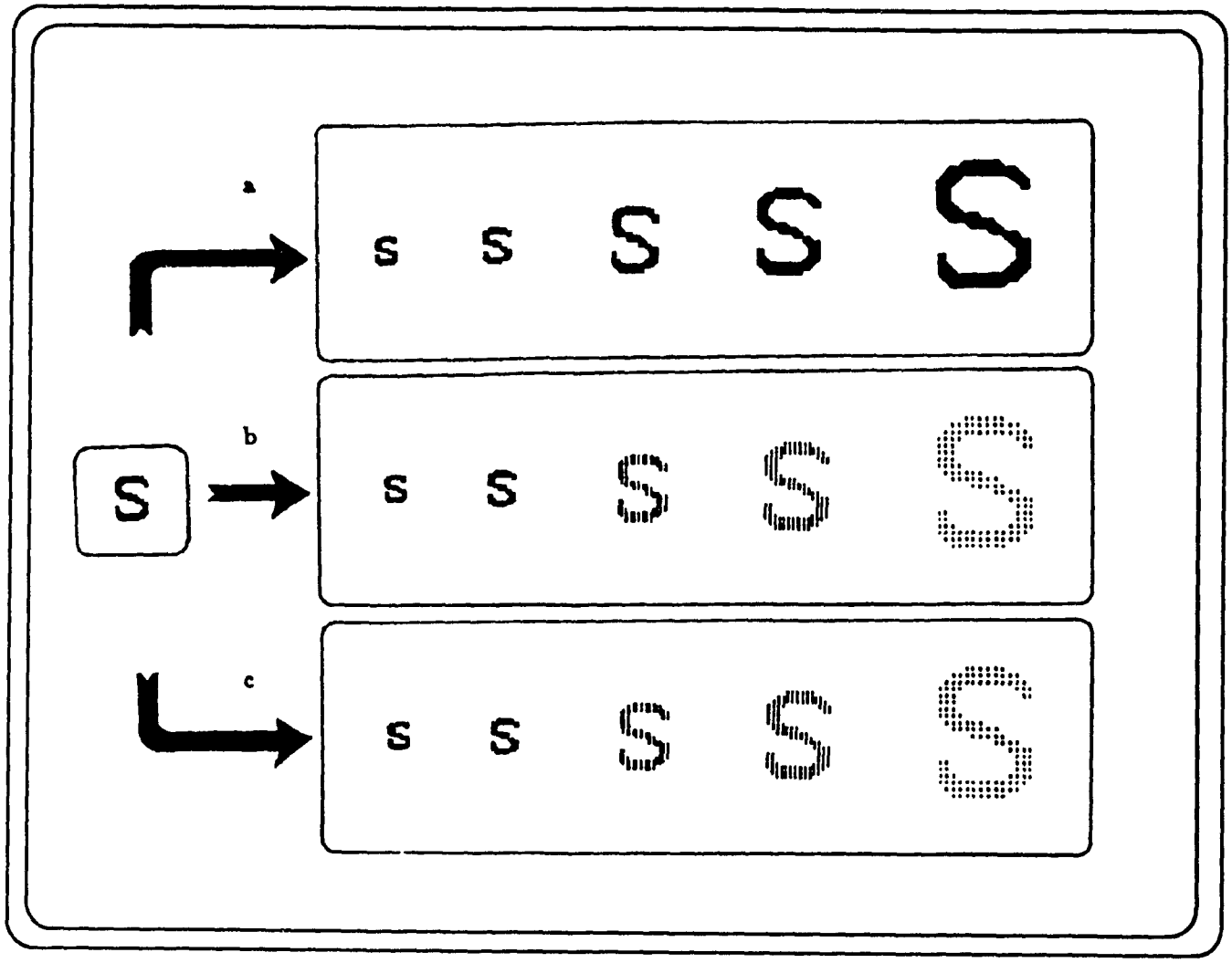


Fig. 4.9 (a) Scaling Results Obtained by the Proposed Algorithm
(b) Scaling Results Obtained by the Ordinary Algorithm
(c) Scaling Results Obtained by Filling Results in (b)

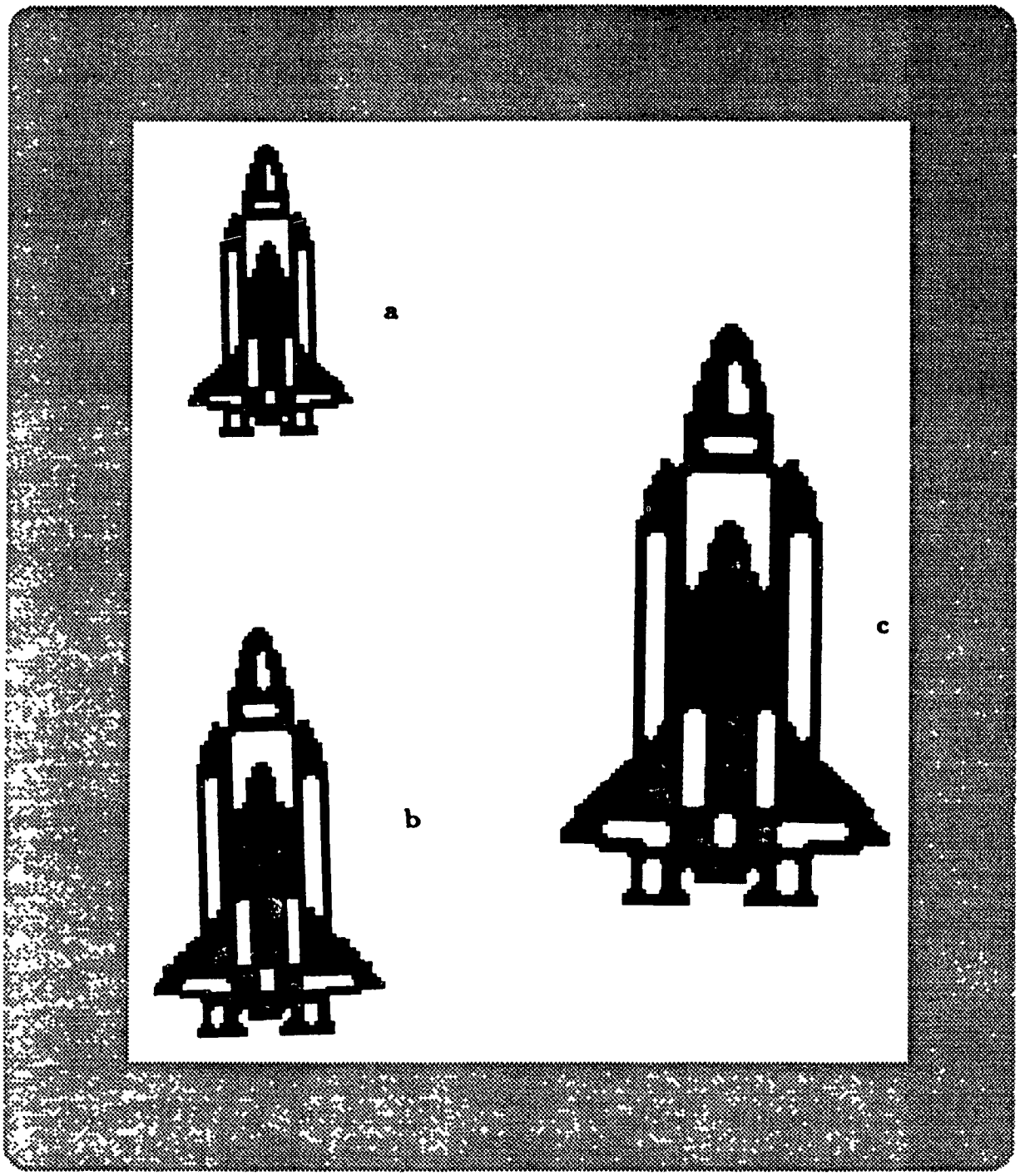


Fig. 4.10 (a) Original Image of the Space Shuttle
(b) Result Obtained by Scaling 1.4
(c) Result Obtained by Scaling 2.0

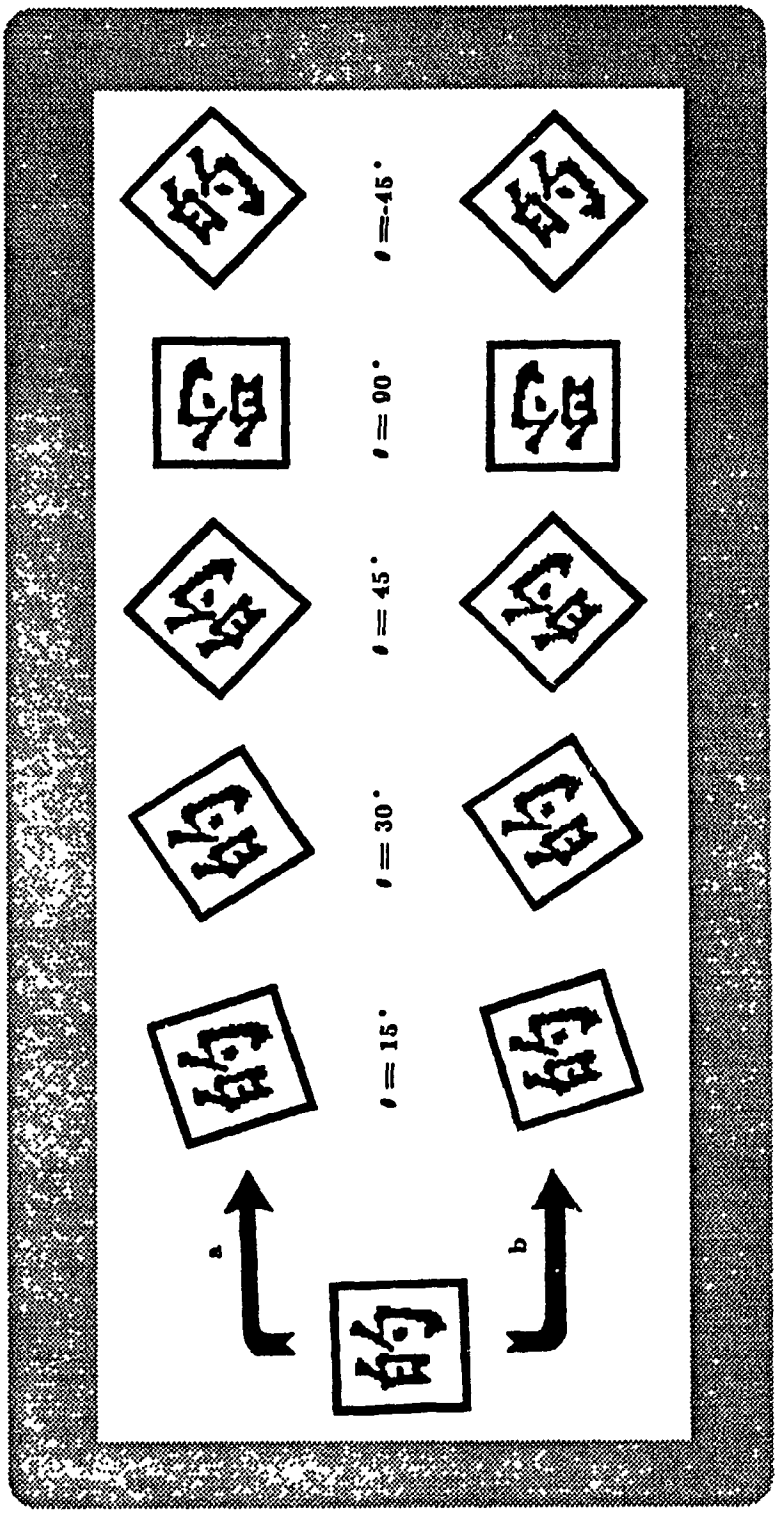


Fig. 4.11 (a) Rotation Results Obtained by the Proposed Algorithm
(b) Rotation Results Obtained by the Ordinary Algorithm

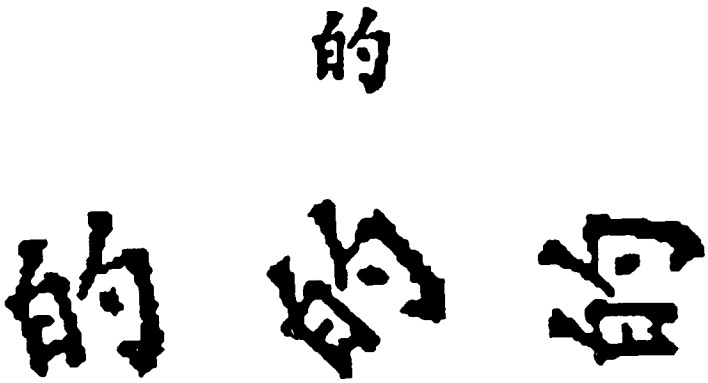


Fig. 4.12 Chinese Character with $\theta = 15^{o}$, $\theta = 45^{o}$ and $\theta = 90^{o}$ respectively first, and $S = 2.0$ next

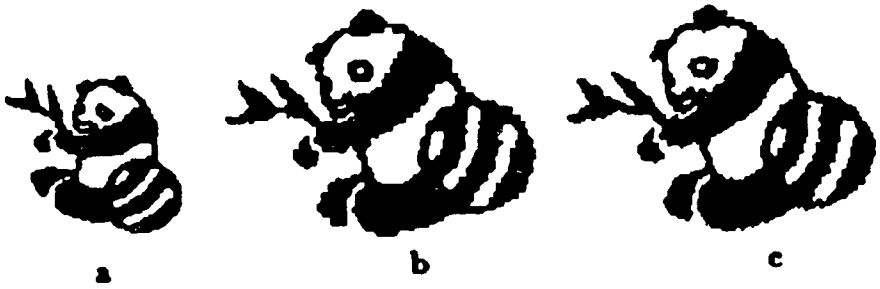


Fig. 4.13 (a) Original Image of the Panda
(b) Result for $S = 1.5$ First and $\theta = 30^{o}$ Next
(c) Result for $\theta = 30^{o}$ First and $S = 1.5$ Next

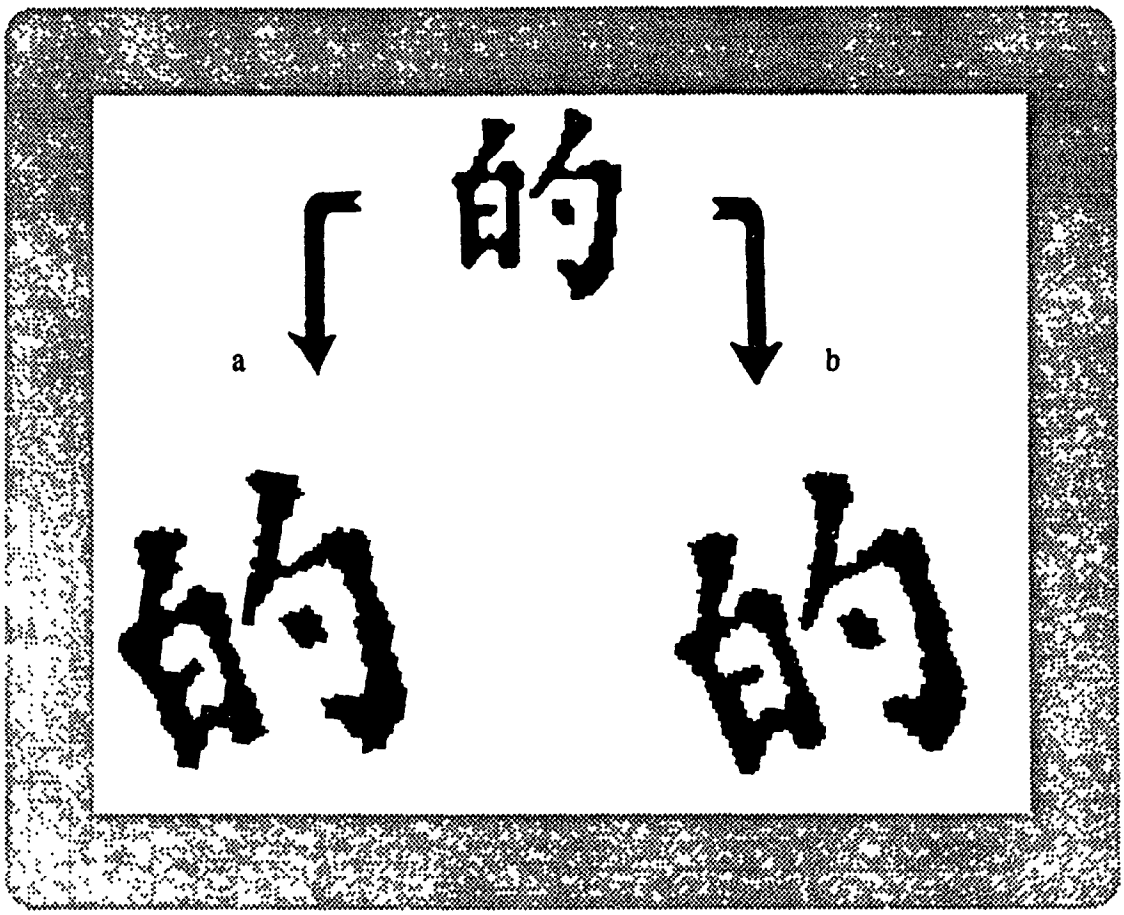


Fig. 4.14 (a) Result for $S = 2.0$ First and $\theta = 30^{o}$ Next

(b) Result for $\theta = 30^{o}$ First and $S = 2.0$ Next

4.4 NONLINEAR IMAGE TRANSFORMATION

In last section, we have discussed the linear image transformation. However, sometimes nonlinear image transformation is required for patterns such as the one shown in Fig. 4.15, where nonlinear distortion has been produced by the bending of the surface on which the letter " A " is written. Once the image in Fig. 4.15a has been normalized into the standard one shown in Fig. 4.15b, the recognition process can be carried out easily. Consequently, it is necessary to study nonlinear image transformations

$$T : (\xi, \eta) \rightarrow (x, y) , \tag{4.14}$$

where

$$x = f (\xi, \eta) , \qquad y = g (\xi, \eta) .$$

Assume that the functions f and g have continuous derivatives with respect to ξ and η respectively, then the Jacobian matrix exists in the form

$$\frac{D (f, g)}{D (\xi, \eta)} = \begin{bmatrix} \dfrac{\partial f}{\partial \xi} & \dfrac{\partial f}{\partial \eta} \\ \dfrac{\partial g}{\partial \xi} & \dfrac{\partial g}{\partial \eta} \end{bmatrix} . \tag{4.15}$$

We then have [Lang (1987)]

Lemma 4.1

The nonlinear transformation $T : (\xi, \eta) \rightarrow (x, y)$ is one-to-one, if and if the determinant of the Jacobian matrix, Eq. (4.15), is nonzero:

$$J (\xi, \eta) = \left| \frac{D (f, g)}{D (\xi, \eta)} \right|$$

$$= \begin{vmatrix} \frac{\partial X}{\partial \xi} & \frac{\partial X}{\partial \eta} \\ \frac{\partial Y}{\partial \xi} & \frac{\partial Y}{\partial \eta} \end{vmatrix} \neq 0 , \tag{4.16}$$

for all (ξ, η) through the transformation T.

While the theoretical analysis of the one-to-one property is difficult for general nonlinear transformation, we suggest that condition (4.16) be tested by a computer, if necessary, from time to time during the transformation.

In this thesis, we will discuss the basic concepts for the following nonlinear image transformations:

(i) Bilinear image transformation,

(ii) Quadratic image transformation,

(iii) Bi-quadratic image transformation,

(iv) Cubic image transformation,

(v) Bi-cubic image transformation.

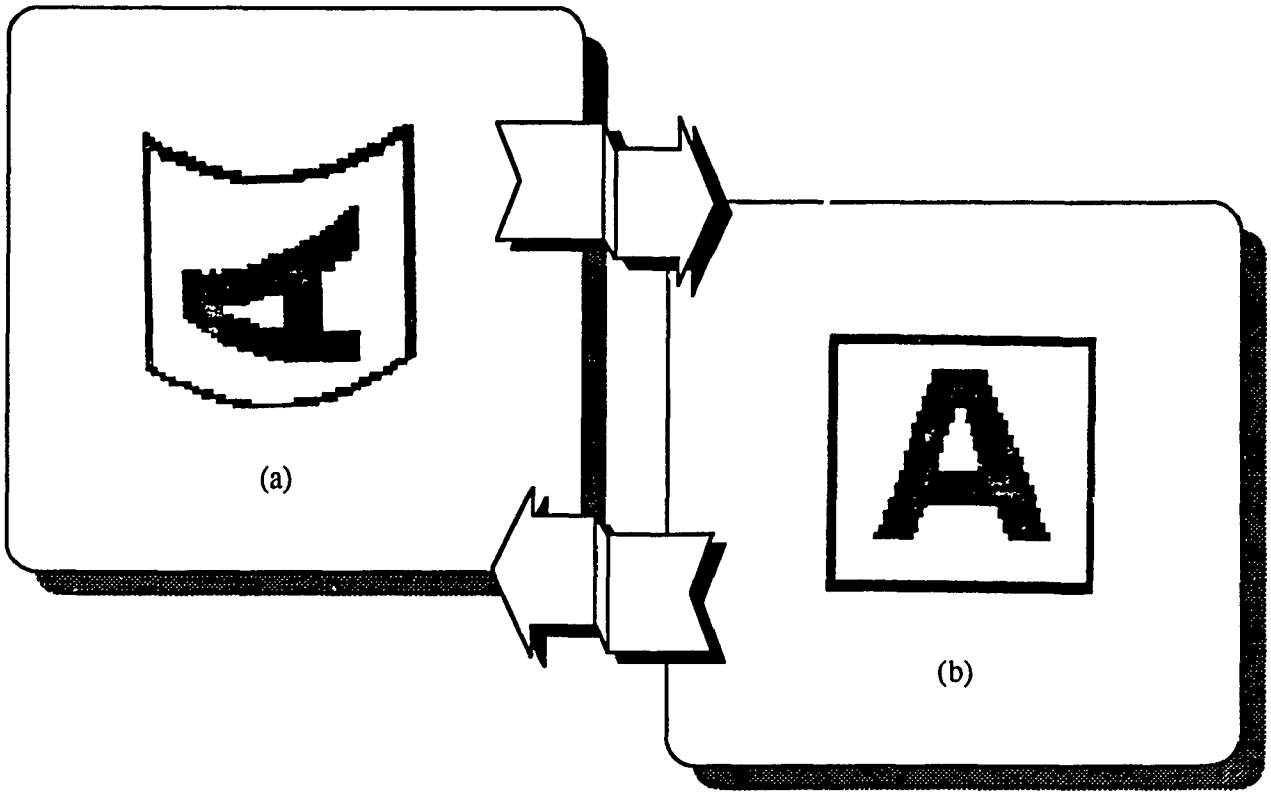


Fig. 4.15 Nolinear Transformation

4.4.1 Bilinear Transformation

Bilinear geometric transformation is the simplest type of nonlinear transformations. The definition of bilinear transformation using homogeneous coordinate representation [Maxwel46, 61, Robert65] can be denoted as follows:

Definition 4.6

Let $\vec{W} = \begin{bmatrix} \xi & \eta & 1 \end{bmatrix}$, $\vec{Z} = \begin{bmatrix} X & Y & 1 \end{bmatrix}$, the bilinear transformation T: $\vec{W} \rightarrow \vec{Z}$ is defined as

$$\vec{Z} = \vec{W}\ T. \tag{4.17a}$$

$$T = \begin{bmatrix} a_{11} + a_{12}\,\eta & b_{11} + b_{12}\,\eta & 0 \\ a_{22} & b_{22} & 0 \\ r & s & 1 \end{bmatrix} \tag{4.17b}$$

where a_{11}, a_{12}, a_{22}, b_{11}, b_{12}, b_{22}, r and s are constants.

Eqs. (4.17a-b) define a bilinear transformation T , which applies to the images in the standard coordinate system $\xi O \eta$ to produce the images in another coordinate system XOY .

The Jacobian matrix of Eq. (4.17 b) is

$$J = \begin{bmatrix} \dfrac{\partial X}{\partial \xi} & \dfrac{\partial X}{\partial \eta} \\ \dfrac{\partial Y}{\partial \xi} & \dfrac{\partial Y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} a_{11} + a_{12}\eta & a_{22} + a_{12}\xi \\ b_{11} + b_{12}\eta & b_{22} + b_{12}\xi \end{bmatrix} \tag{4.17c}$$

The necessary and sufficient conditions of one-to-one correspondence for T is

that the determinant of J is nonzero during the whole transformation procedure.

Graphic examples are given in Fig. 4.16 to illustrate shape distortion produced by bilinear transformation. In this figure, different bilinear shapes are obtained due to the transformations T_1, T_2, T_3 and T_4 which correspond with the choice of parameters a_{11}, a_{12}, a_{22}, b_{11}, b_{12}, b_{22}, r and s.

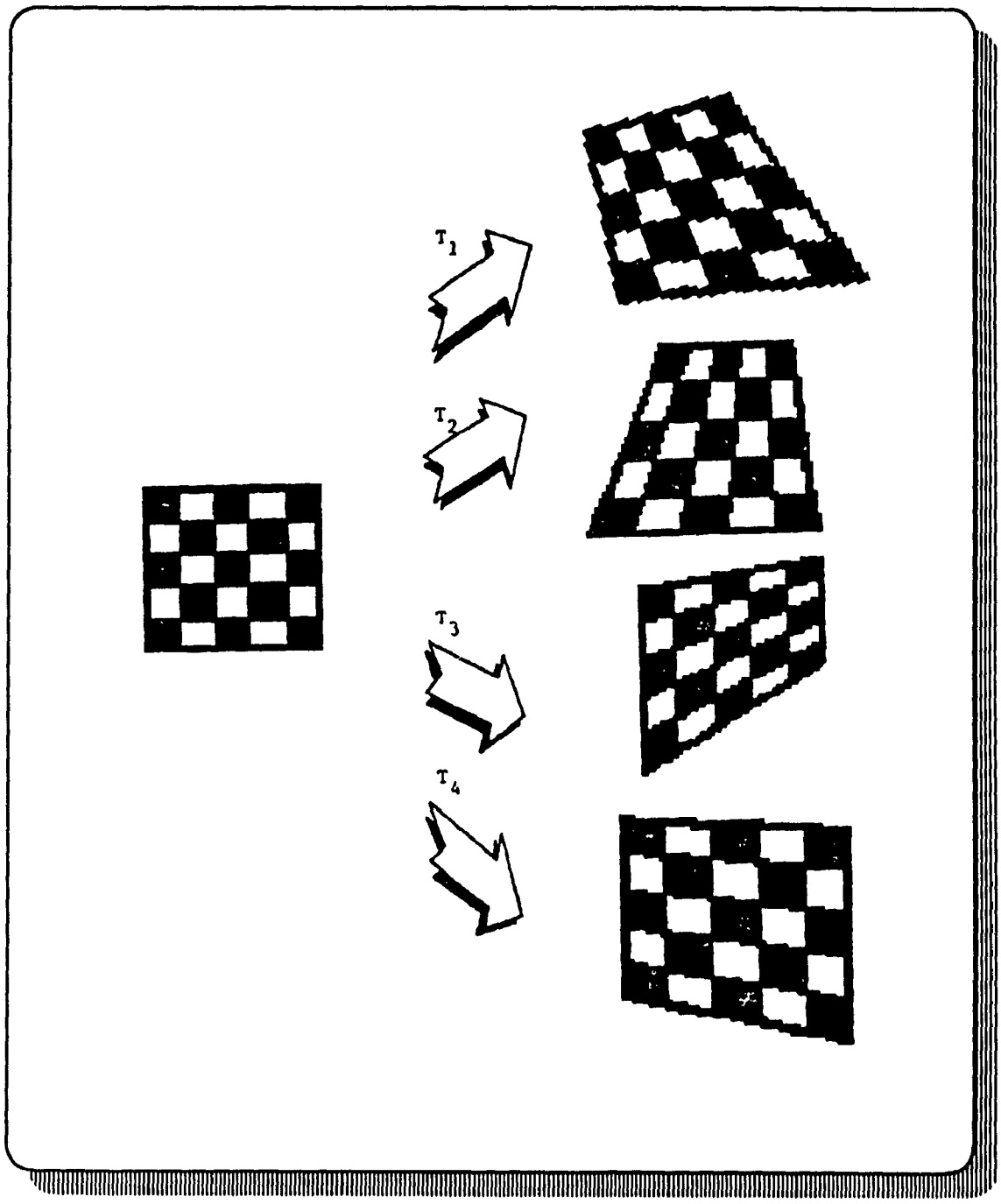


Fig. 4.16 Examples of Bilinear Transformation

4.4.2 Quadratic and Bi-quadratic Transformations

Quadratic and bi-quadratic image transformations belong to the nonlinear transformations. The definition of quadratic transformation using homogeneous coordinate representation can be denoted as follows:

Definition 4.7

Let $\vec{W}^T = \left[\xi\ \eta\ 1\right]$, $\vec{Z}^T = \left[X\ Y\right]$, the quadratic transformation T: $\vec{W} \rightarrow \vec{Z}$ is defined as

$$\vec{Z} = T\ \vec{W}. \tag{4.18a}$$

$$T = \begin{bmatrix} \vec{W}^T & 0 \\ 0 & \vec{W}^T \end{bmatrix} T^*, \tag{4.18b}$$

$$T^* = \begin{bmatrix} \begin{bmatrix} a_{11} & \frac{1}{2}a_{12} & \frac{1}{2}a_{10} \\ \frac{1}{2}a_{12} & a_{22} & \frac{1}{2}a_{20} \\ \frac{1}{2}a_{10} & \frac{1}{2}a_{20} & r \end{bmatrix} \\ \begin{bmatrix} b_{11} & \frac{1}{2}b_{12} & \frac{1}{2}b_{10} \\ \frac{1}{2}b_{12} & b_{22} & \frac{1}{2}b_{20} \\ \frac{1}{2}b_{10} & \frac{1}{2}b_{20} & s \end{bmatrix} \end{bmatrix}. \tag{4.18c}$$

where $a_{10}, a_{20}, a_{11}, a_{12}, a_{22}, b_{10}, b_{20}, b_{11}, b_{12}, b_{22}$, r and s are constants.

Eq. (4.18a-c) define a quadratic transformation, which applies to the images

in the standard coordinate system $\xi O \eta$ to produce the images in another coordinate system XOY . The graphic examples are given in Fig. 4.17 to illustrate the quadratic distortion formulated by quadratic transformation.

The details for bi-quadratic transformation model will present in Chapter 6. Here, we only give some graphic examples in Fig. 4.18 to illustrate the bi-quadratic distortion produced by bi-quadratic transformation.

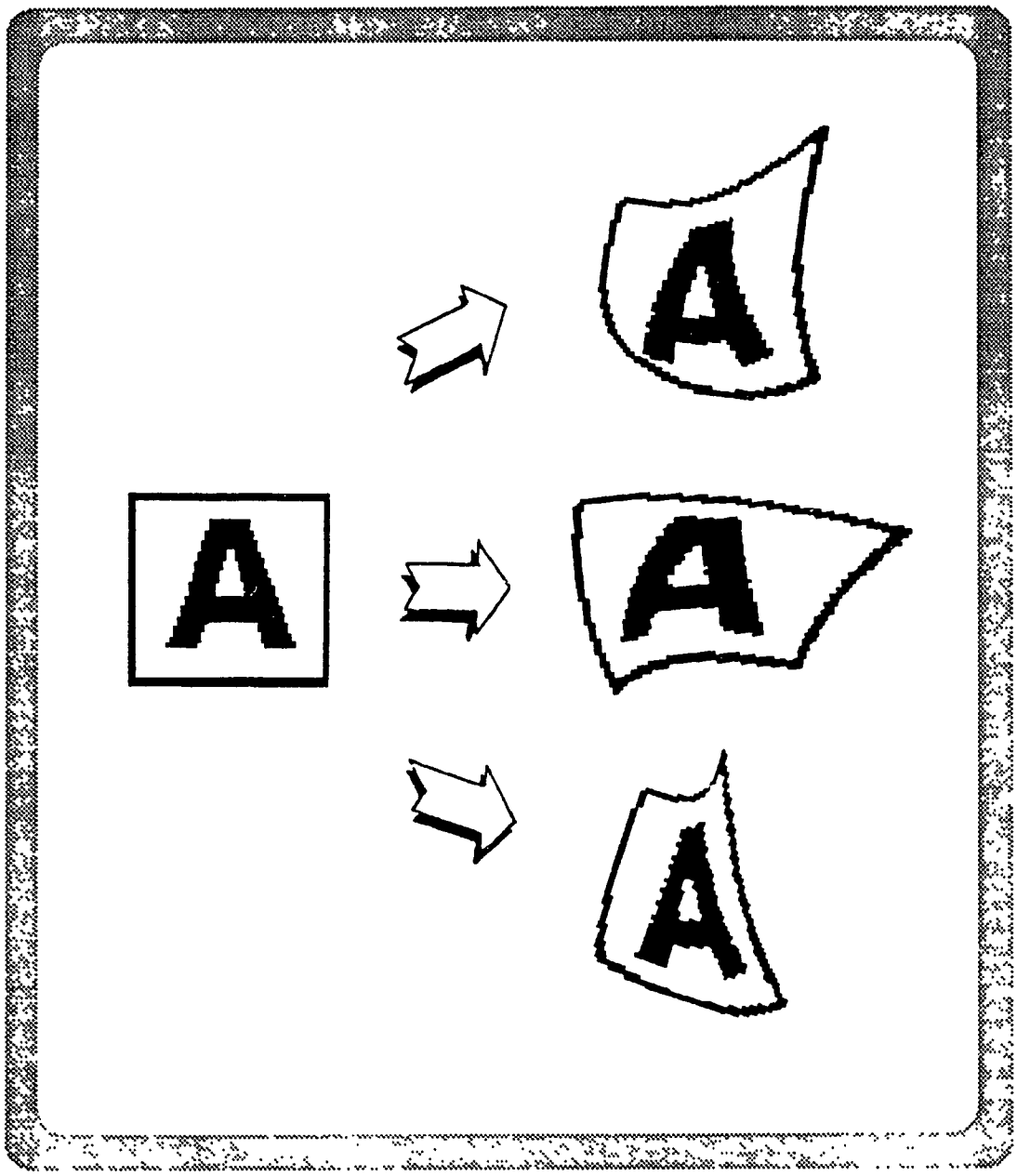


Fig. 4.17 Examples of Quadratic Transformation

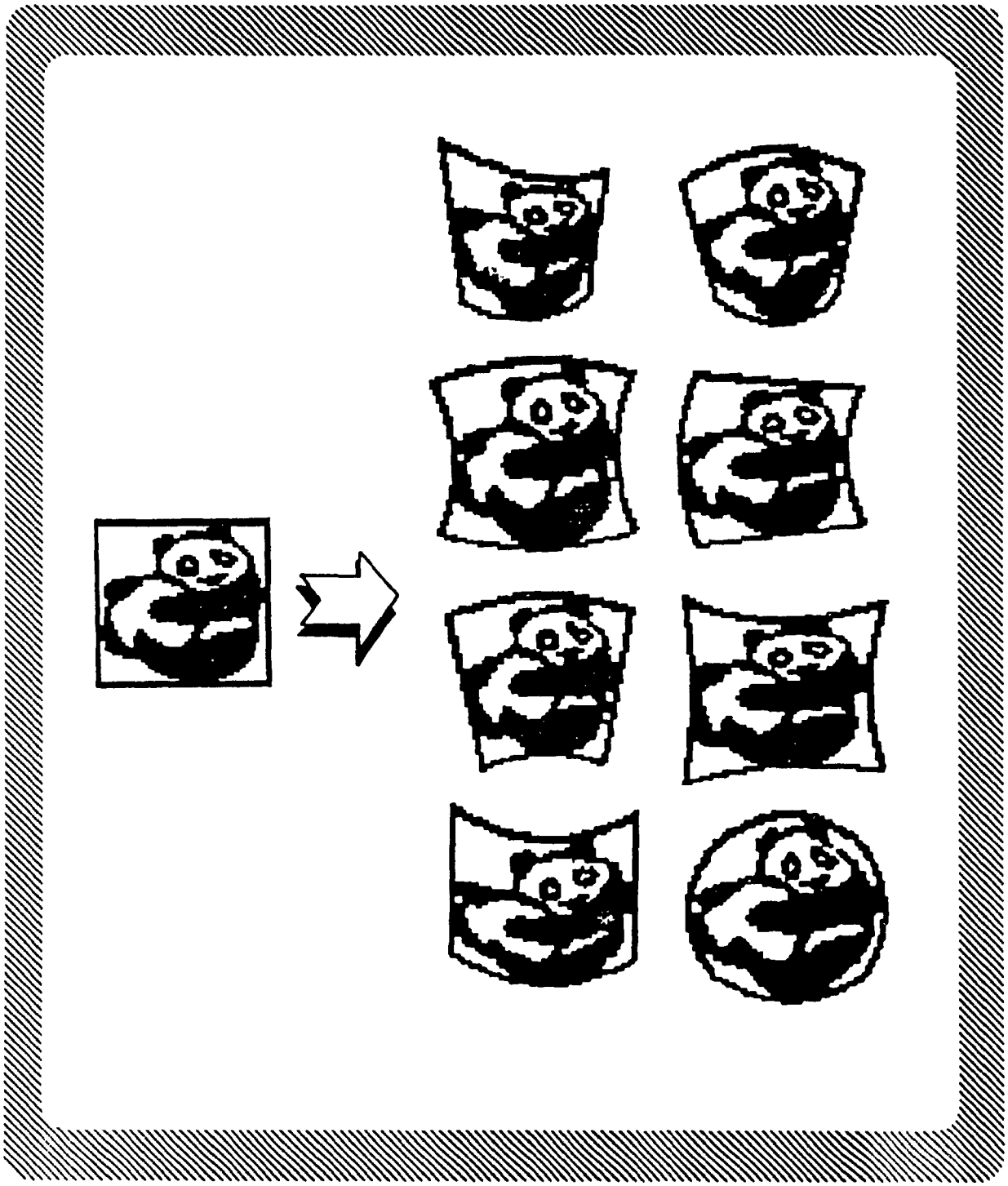


Fig. 4.18 Examples of Bi-quadratic Transformation

4.4.3 Cubic and Bi-cubic Transformations

The definition of cubic transformation can be denoted below:

Definition 4.8

Let $\vec{W}^T = \begin{bmatrix} \xi & \eta & 1 \end{bmatrix}$, $\vec{Z}^T = \begin{bmatrix} X & Y \end{bmatrix}$, the cubic transformation T: $\vec{W} \rightarrow \vec{Z}$ is defined as

$$\vec{Z} = T\ \vec{W}\ , \tag{4.19a}$$

$$T = \begin{bmatrix} \Theta & 0 \\ 0 & \Theta \end{bmatrix} T^*\ , \tag{4.19b}$$

$$\Theta = \begin{bmatrix} \xi^2 + \xi\eta + \eta & \eta^2 + \xi\eta + \xi & 1 \end{bmatrix}, \tag{4.19c}$$

$$T^* = \begin{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \\ a_7 & a_8 & r \end{bmatrix} \\ \begin{bmatrix} b_1 & b_2 & b_3 \\ b_4 & b_5 & b_6 \\ b_7 & b_8 & s \end{bmatrix} \end{bmatrix}. \tag{4.19d}$$

where $a_1 - a_8$, $b_1 - b_8$, r and s are constants.

Explicit equations are

$$\begin{aligned} X = {} & a_1\xi^3 + a_5\eta^3 + (a_1 + a_2 + a_4)\,\xi^2\eta + (a_2 + a_4 + a_5)\,\eta^2\xi + \\ & (a_3 + a_4)\,\xi^2 + (a_2 + a_6)\,\eta^2 + (a_1 + a_3 + a_5)\,\eta\xi + \\ & (a_6 + a_7)\,\xi + (a_3 + a_8)\,\eta + r\ , \end{aligned} \tag{4.20a}$$

$$Y = b_1\xi^3 + b_5\eta^3 + (b_1 + b_2 + b_4) \xi^2\eta + (b_2 + b_4 + b_5) \eta^2\xi + (b_3 + b_4) \xi^2 + (b_2 + b_6) \eta^2 + (b_1 + b_3 + b_5) \eta\xi + (b_6 + b_7) \xi + (b_3 + b_8) \eta + s \ , \qquad (4.20b)$$

Eqs. (4.19) and (4.20) define a cubic transformation, which applies to the images in the standard coordinate system $\xi O \eta$ to produce the images in another coordinate system XOY .

The details about bi-cubic transformation model will discuss in Chapter 6. In order to give a basic idea, this section only shows some graphic examples in Fig. 4.19 to illustrate bi-cubic shape distortion formed by the bi-cubic transformation model.

As mentioned in the beginning of this section,

$$T : (\xi, \eta) \rightarrow (x, y) , \qquad (4.14)$$
$$x = f (\xi, \eta) , \qquad y = g (\xi, \eta) .$$

provides variant image transformations. All those transformation models are continuous; but the image pixels are discrete. How can we property apply the continuous transformations (4.14) to discrete image transformations? One trouble is that some superfluous holes and blanks occur, this is called "measles" problem. In Section 4.3, we developed a method solving linear image transformation, but it will be fail in nonlinear cases. To handle the measles problems in nonlinear image transformations, Li et al. have developed Splitting-Shooting and Splitting-Integrating methods. The key ideas in these methods are presented below:

(1) Establish the corresponding relations between a pixel image and a matrix. In fact, any image can be represented by numerical quantities. Therefore, the image transformations can be performed through mathematical operations.

(2) Calculate transformed functions, based on numerical integrations. Since the current methods in numerical analysis [Burden81] are invalid for the images produced by an optical scanner, [Li88a, b, 89c, 90] developed the splitting-shooting method for the transformation T, and the splitting-integrating method for the inverse transformation T^{-1}.

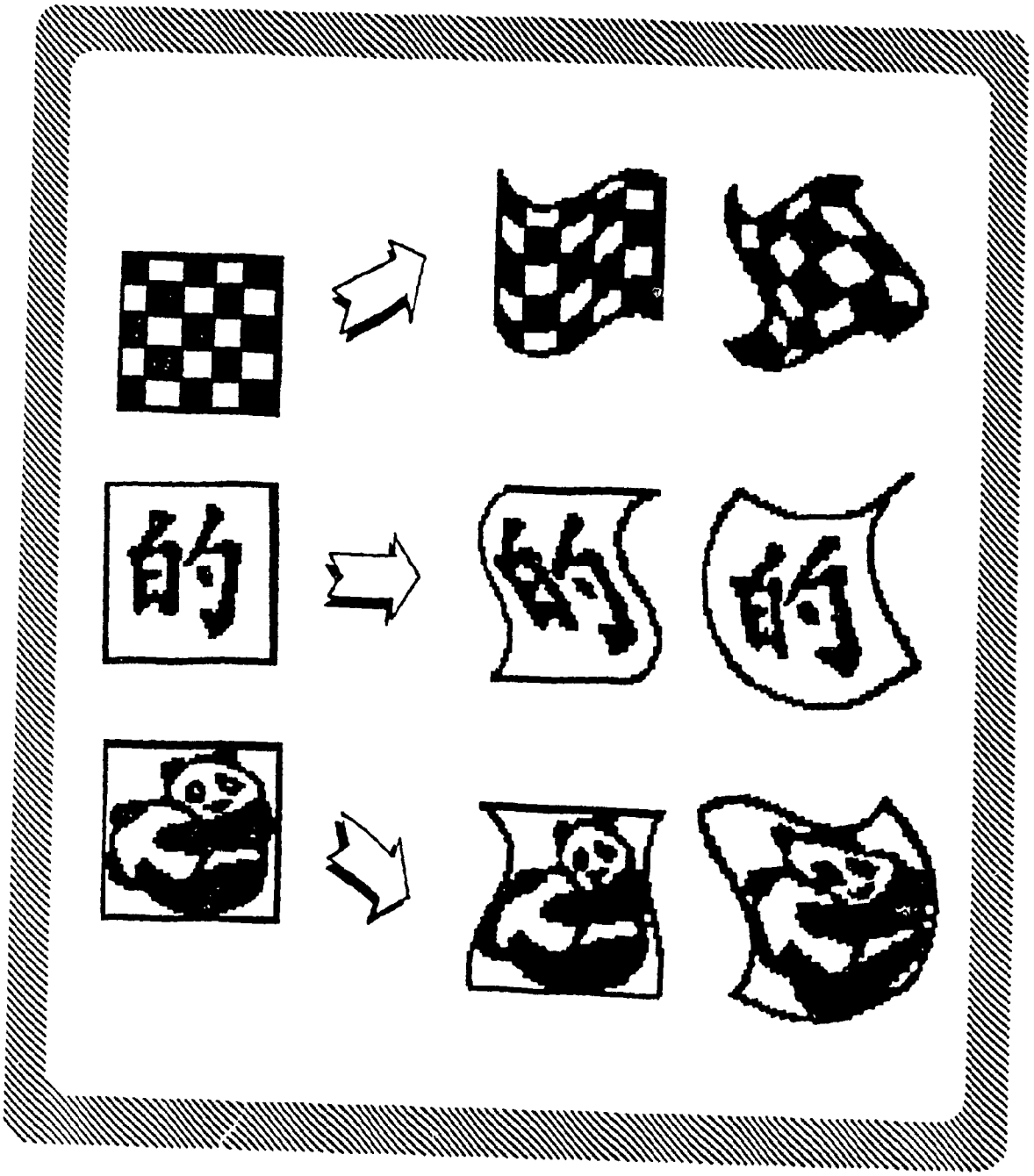


Fig. 4.19 Examples of Bi-cubic Transformation

CHAPTER 5

REMOVAL OF UNCERTAINTY FROM IS1: SIZE-ROTATION INVARIANT ALGORITHM

5.1 INTRODUCTION

As mentioned in Chapter 2, the intrinsic characteristics of the target pattern set, such as the categories of patterns, number of classes, fonts of characters, variance of size and orientation etc., exist in the first level of MLIS. All of them belong to the primary uncertainty in a pattern recognition system which encourage a lot of researchers to develop a variety of methods to handle such intrinsic problems. All these methods can be regarded as ERT. Its function is to remove the uncertainty from the IS1.

In this chapter, we will deal with one of the common intrinsic uncertainties in the IS1, the variance of size and orientation. In order to eliminate this uncertainty, our earlier works [Tang88b, 89b] have presented an algorithm called Transformation-Ring-Projection (TRP). The TRP will be discussed briefly in this chapter.

Invariance in size and rotation will greatly facilitate character recognition especially for texts mixed with graphics. Character readers with such capabilities would find many applications in office automation, computer aided design,

electronic publication, etc. However, this is one of the most challenging problems in pattern recognition. Although some commercial OCR machines are available which perform very well when the characters are fixed in size and rotation, the recognition rate would drop drastically if the size is varied, and would be much worse if the orientation is also changed. Many papers have been written on solving this problem. For instance, in [Kahan87], an approach has been described to recognize printed text of various fonts and sizes for the Roman alphabet. Several techniques, such as thinning, shape extraction, line adjacency graph, and shape mapping etc., have been combined in order to improve the overall recognition rate. An algorithm has been proposed by T. Antoine and C. Y. Suen [Antoin89] in which a descriptor is obtained. This descriptor is independent of the position, orientation and size. References [Cormar63, 64, Hansen81, Hsu82, Merser86, Psalti77, Wu86] describe a rotation-invariant operation based on circular-harmonic function (CHF). And some scale-invariant operations based on Mellin transformation and Broadband dispersion-compensation technique have been presented in [Merser86, Wu86]. Most of them are optical methods and quite complicated.

In this chapter, a new method is proposed. The basic operations have been carefully designed so that the proposed algorithm is very simple and regular. As a result, the parallel processing and VLSI technology can be used to speed up the computation [Tang89b]. This method is called the Transformation-Ring-Projection (TRP) method. In this method, image transformation technique as

mentioned in Chapter 4 is employed to center the image and normalize its size. This brings a reduction in entropy. The Ring-Projection scheme is used to handle the orientation problem, and it reduces the entropy further.

In this chapter, Section 5.2 gives the basic concepts of the Ring- Extraction Panel. Section 5.3 describes the proposed algorithm (TRP). Finally, experimental results and discussions are presented in Section 5.4.

5.2 RING - EXTRACTION PANEL

It is well known that pattern recognition is a process to reduce the uncertainty of the target set to be recognized. An important source of the uncertainty is the intrinsic characteristics of the target pattern set. For example the intrinsic characteristics of a Chinese character set include categories, sizes, fonts, directions etc.. The uncertainty from the intrinsic characteristics can be reduced considerably based on the relative knowledge about the relationships among various kinds of characteristics. Again taking the Chinese character set as an example, suppose we have a set of 5000 Chinese characters. If we treat each a Chinese character as a pattern class we will have 5000 classes and each class has only one pattern sample. Then if we consider the more complex situations such as each character may have several sizes, say 10 sizes, and allow 360 degree rotations, there will be two solutions to handle them. The first solution is to treat them as single pattern sample classes, i.e. the same character but with different sizes or directions treated as different classes. This means the uncertainty will increase considerably. The second solution is that we only consider the categories of characters and don't care about the sizes or directions, i.e. the same character with different sizes or directions will be grouped into the same class. This means the uncertainty will remain as large as that of the set with only 5000 classes. Obviously the second solution is what we need. Now the question is whether we can find a method for the machine to know which characters should be grouped into the same class? The answer is positive. In this chapter we will present a method to pursue size-

rotation-invariant character recognition.

To describe the principle of this method we define the following notations.

Definition 5.1 :

A target pattern set W is denoted as follows

$$W = \{ w_1, w_2, \cdots, w_i, \cdots, w_N \}$$

where $N \in I$, and w_i is the i-th class which may contain finite pattern samples, i.e.

$$w_i \in A = \{ a_1, a_2, \cdots, a_L \} \qquad i = 1, 2, \cdots, N$$

If a set of W, in which all w_i are single pattern sample classes, we call W a standard set.

One important aspect of our method is that we use a ring-extraction panel defined as follows to extract the features of the target pattern samples.

Definition 5.2 :

A ring-extraction panel consists of n concentric rings and m spokes, as shown in the Fig. 5.1, where n and m are integers. r_i is the radius of the i-th ring (R_i), s_j represents the j-th spoke ($i = 1, 2, \cdots n$; $j = 1, 2, \cdots m$). Each cross point between a ring and a spoke is called a sample-point, $p(i,j)$, represents the cross point between the i-th ring and the j-th spoke.

After having been extracted by the ring-extraction panel, a pattern sample is represented by a vector called ring-projection vector, defined as follows.

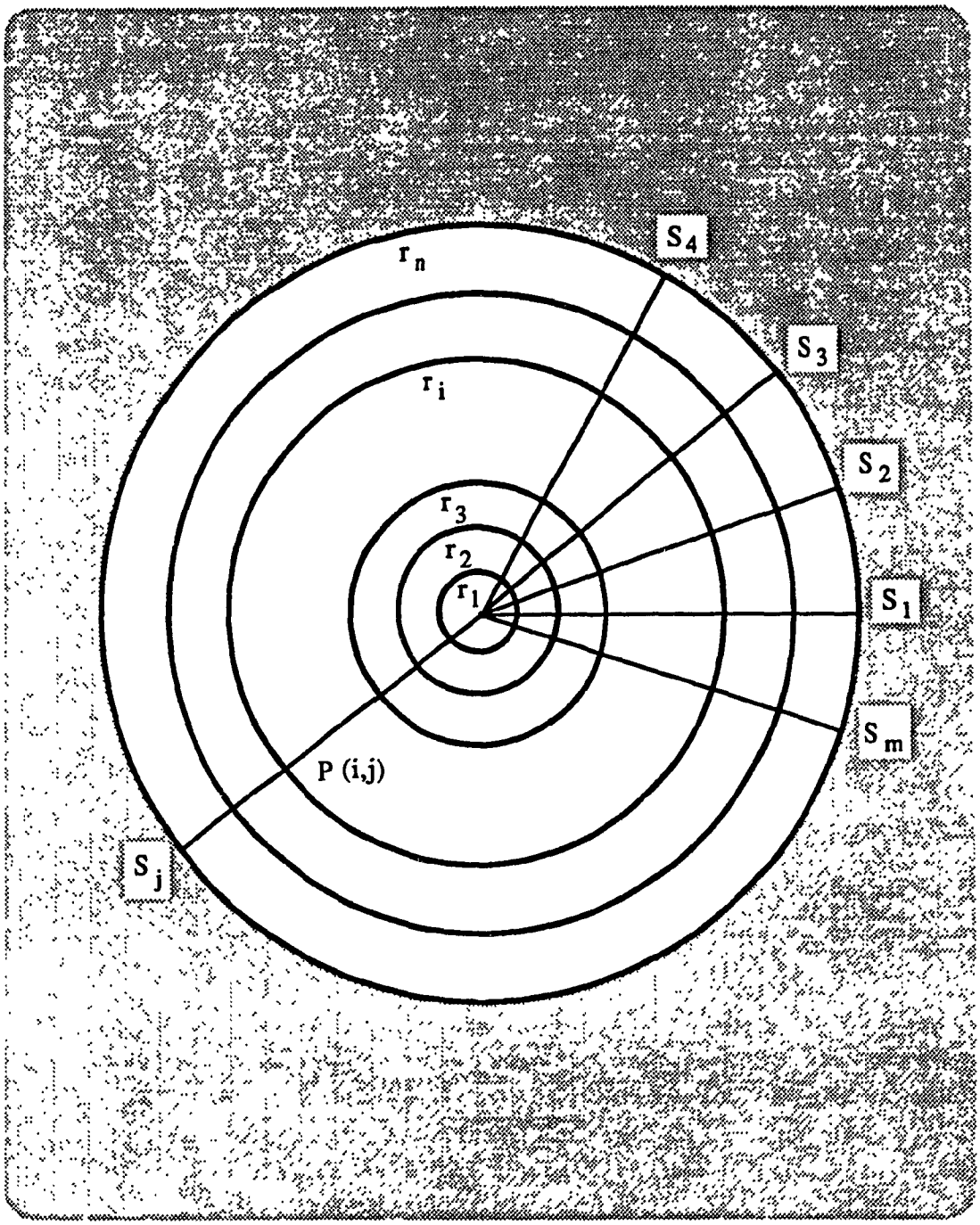


Fig. 5.1 A Ring-Extraction Panel

Definition 5.3 :

A ring-projection vector is

$$\overline{V} = (p_{r_1}, p_{r_2}, \cdots, p_{r_n})$$

where

$$p_{r_i} = \sum_{j=1}^{m} p(i,j), \qquad i = 1, 2, \cdots, n$$

and

$$p(i,j) = \begin{cases} 1 & \textit{overlaps with the pattern sample} \\ 0 & \textit{otherwise} \end{cases}$$

An apparent result of using ring-extraction panel is that all characters with the same category but different directions will be automatically grouped into the same pattern class because they have the same ring-projection vectors. Furthermore we can use a simple linear transformation to normalize all different sizes into the one of ring-extraction panel to unify the sizes of characters. Concrete algorithms are presented in the next section.

5.3 TRP Algorithm

The following processing steps are included in the algorithm:

5.3.1 Position Invariant

Removing the shift of the position of the input character :

Once the character has entered the system, its center of gravity will be found then be moved to the origin (0,0) of the image plane.

For a binary character in the 2-dimensional Cartesian system:

$$f(x,y) = \begin{cases} 1 & \text{if}(x,y) \in P \ \ (image) \\ 0 & otherwise \ \ (background) \end{cases} \qquad (5.1)$$
$$(\ 1 \leq x,\ y \leq M\ ),$$

where M is the size of the input image.

The center of gravity $(\overline{X},\overline{Y})$ for the character is given by

$$\begin{cases} \overline{X} = m_{10} \ / \ m_{00}; \\ \overline{Y} = m_{01} \ / \ m_{00}; \end{cases} \qquad (5.2)$$

where

$$m_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{M} x^p y^q f(x,y)$$

denotes the geometrical moments of the character [Wang179].

5.3.2 Size Invariant

1). Finding the largest size of the image :

The pixel of the input character image farthest away from the center will be found. (Note there may be more than one pixel which have the same distance value. Since we are only concerned with the distance, only one pixel value will be used). Also there are several approaches available for finding the distance. To be consistent with step 4, we will use the method that divides the image plane into several rings. Take the distance between each adjacent pair of rings as 1 unit, i.e. ($r_i - r_{i-1} = 1$). The largest non-zero ring distance r_K will be assigned by the value d.

2). Normalizing the size of the input character :

Let D be the standard size, then the size of the input character is normalized by a factor of D/d_j.

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} D/d_j & 0 \\ 0 & D/d_j \end{bmatrix} \begin{bmatrix} x_j \\ y_j \end{bmatrix} \tag{5.3}$$

where (X, Y) are the new coordinates of a point for a pattern sample in a standard size, (x_j, y_j) are the coordinates of a point for a pattern sample with size j, D is the standard size, and d_j stands for size j.

5.3.3 Rotation Invariant

Rotation-invariant operation - Ring Projection :

After size normalization, the system will convert the Cartesian coordinate system into the polar coordinate system. Then the pattern is transformed from the image domain to the area domain using ring projection. If we consider the area of rings, it would be invariant to rotation, i.e. rotation-invariant. In order to reduce the rotation error and to emphasize the difference between characters, we use the rings with the radii r_i starting from the center with an increment 1 pixel. Let p_i be the number of pixels (pixels $\in$ picture and pixels $\in p_{r_i}$). p_i will be used as the feature to describe ring R_i of the input character, as shown in Fig. 5.2.

Using $(\overline{X}, \overline{Y})$ as the new origin of a polar coordinate system, i.e. $x_0 = \overline{X}$, $y_0 = \overline{Y}$, then the 2-dimensional character can be represented by

$$f^*(r,\theta) = \begin{cases} 1 & (for\ the\ black\ pixels) \\ 0 & (for\ the\ white\ pixels) \end{cases} \qquad (5.4)$$
$$(0 \le r \le n\ ,\ \ 0 \le \theta \le 2\pi).$$

where

$$n = (x_1^2 + y_1^2)^{\frac{1}{2}}$$

and

$$x_1 = Max\{\overline{X}\ ,\ M - \overline{X}\};$$
$$y_1 = Max\{\overline{Y}\ ,\ M - \overline{Y}\}.$$

Partitioning the character and projecting the black pixels onto annular multi-layers of character give a projection value p_i as follows:

$$p_i = \sum_{\theta=0}^{2\pi} f^*(i,\theta), \quad (i = 0,1,2,\ldots,n), \tag{5.5a}$$

in order to reduce the error produced by the shift of the center of gravity due to the noise, we take

$$P_i = \sum_{i=0}^{i} p_i, \quad (i = 0,1,2,\ldots,n), \tag{5.5b}$$

Then we can have an n-dimensional vector

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ P_n \end{bmatrix} \tag{5.6a}$$

or

$$\begin{bmatrix} p_1 \\ p_1 + p_2 \\ p_1 + p_2 + p_3 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ p_1 + p_2 + \cdots + p_{n-1} + p_n \end{bmatrix} \tag{5.6b}$$

as the feature vector to characterize size-normalized input characters.

In this method, because of the sum of projections P_i instead of the individual projection of a ring p_i, the system is more stable than that proposed in [Taza89],

the details can be found in [Tang89b].

Now we can summarize the algorithm as follows:

Step 1: Find the center of gravity and translate it to the origin of the image plane;

Step 2: Find the largest distance d;

Step 3: Scale the input image by D/d ;

Step 4: Find the feature vector using ring projection:

$$\left[P_1 \quad P_2 \quad \cdots \quad P_n\right]^T ,$$

where

$P_1 = p_1$;

$P_2 = p_1 + p_2$;

$P_n = p_1 + p_2 + \cdots + p_{n-1} + p_n$.

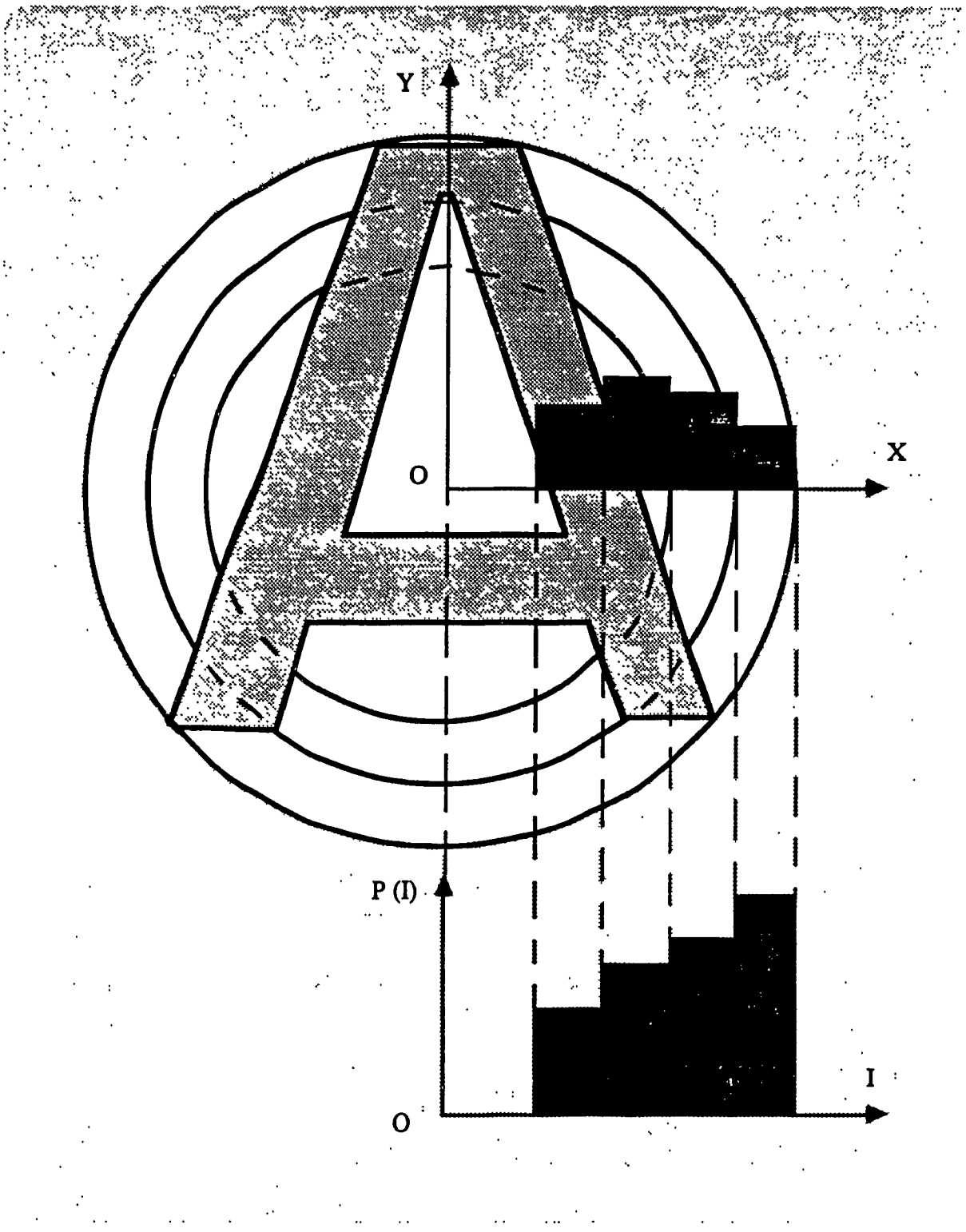


Fig. 5.2 Ring Projection

5.4 EXPERIMENTAL RESULTS

A series of experiments have been conducted to verify the proposed algorithms.

Fig. 5.3(a) shows the Chinese character "Of" with

scaling 1.2 and rotation 15 ° ;

scaling 1.4 and rotation 30 ° ;

scaling 1.6 and rotation 45 ° ;

scaling 1.8 and rotation 90 ° ;

scaling 2.0 and rotation -45 ° respectively.

Fig. 5.3(b) shows the results after normalizing the size of the characters by the proposed algorithm.

Fig. 5.3(c) shows the results after applying step 4 to process the characters of Fig. 5.3(b) oriented in 6 different directions. We plot

$$P(i) = \sum_{i=0}^{i} \sum_{\theta=0}^{2\pi} f^{*}(i, \theta), \quad (i = 0,1,2,...,n),$$

along the vertical axis and i along the horizontal.

From Fig. 5.3(c) , it can be seen that different sizes and rotations of the given character produce very similar feature vectors when processed by the proposed algorithm. Hence it can be used for the purpose of recognizing characters.

Fig. 5.3 (a) Chinese Character "Of" with Scalings and Rotations

Fig. 5.3 (b) Results Obtained by Normalizing the Size in (a)

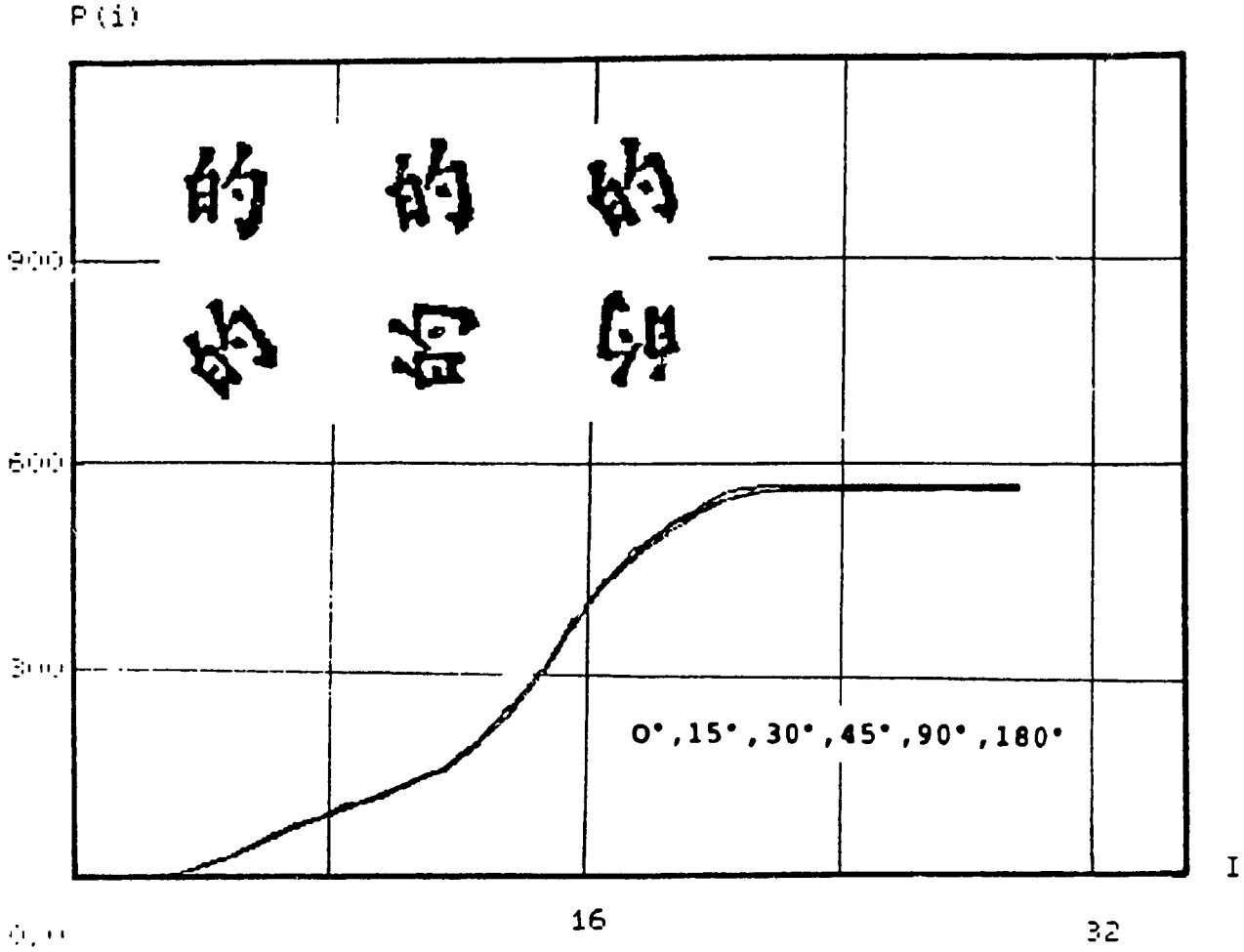


Fig. 5.3 (c) Results Obtained by the Ring Projection in (b)

Fig. 5.4 describes the results on two similar Chinese characters "Nine" and "Ball" rotated 0°, 15°, 30°, 45°, 90° and 180°, respectively. From the figure we can see:

1) For any given character, the curves representing different rotation lie within a very small range.

2) At least for some p_i's, the two families of the curves for two similar characters are far apart from each other, and these features can be used for recognition.

Fig. 5.5 shows the results after applying the proposed algorithm to two similar Chinese characters "End" and "No". The same conclusions as before can be reached.

The features of 26 English alphabetic letters are found in Fig. 5.6. And the results for several pairs of similar letters (U,V), (W,M), (B,8), (Z,2) and (S,5) are also described in Figs. 5.7, 5.8, 5.9, 5.10, and 5.11.

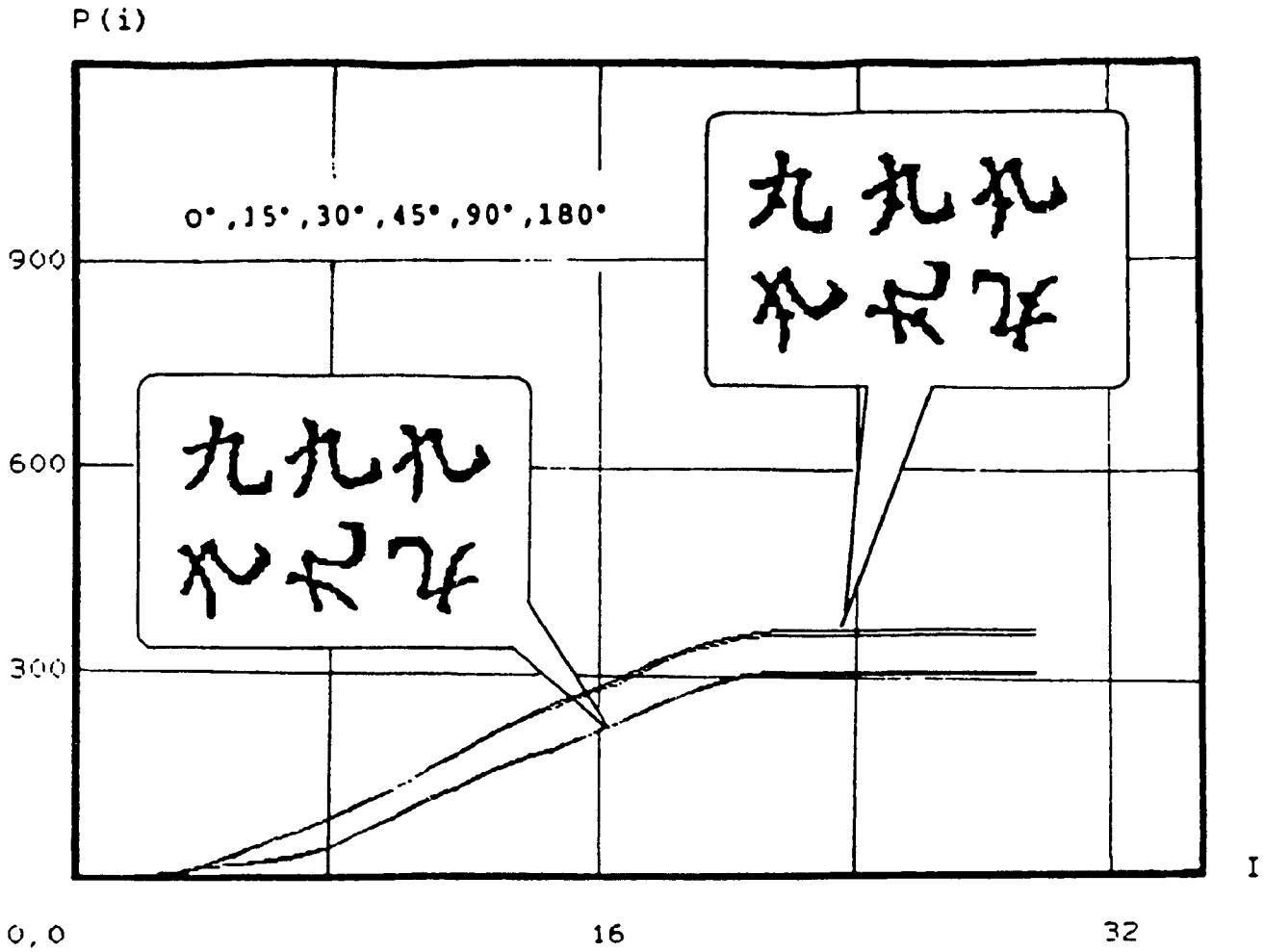


Fig. 5.4 Results on Two Similar Chinese Characters "Nine" and "Ball" Rotated 0°, 15°, 30°, 45°, 90° and 180°, Respectively

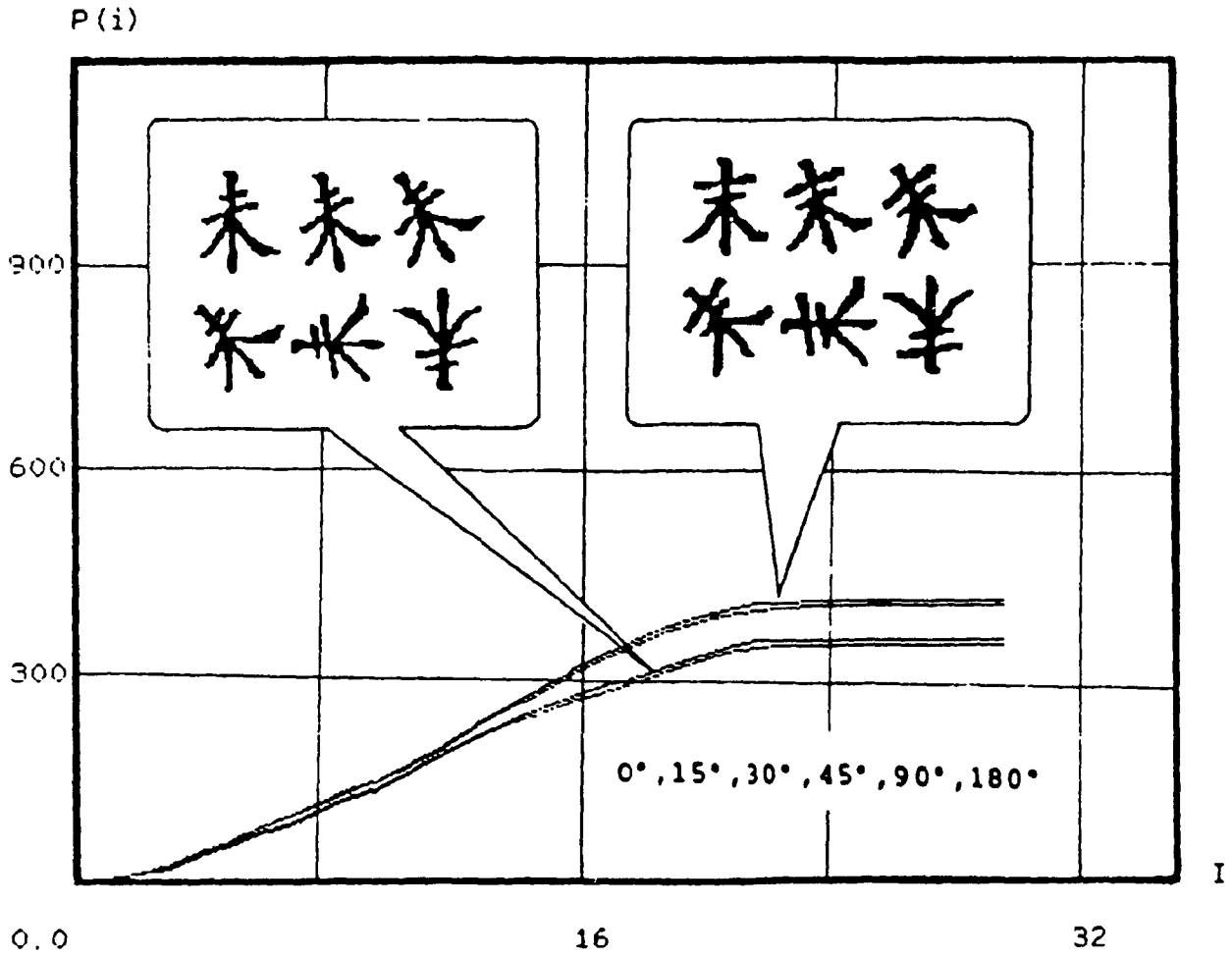


Fig. 5.5 Results on Two Similar Chinese Characters "End" and "No" Rotated 0°, 15°, 30°, 45°, 90°, and 180°, Respectively

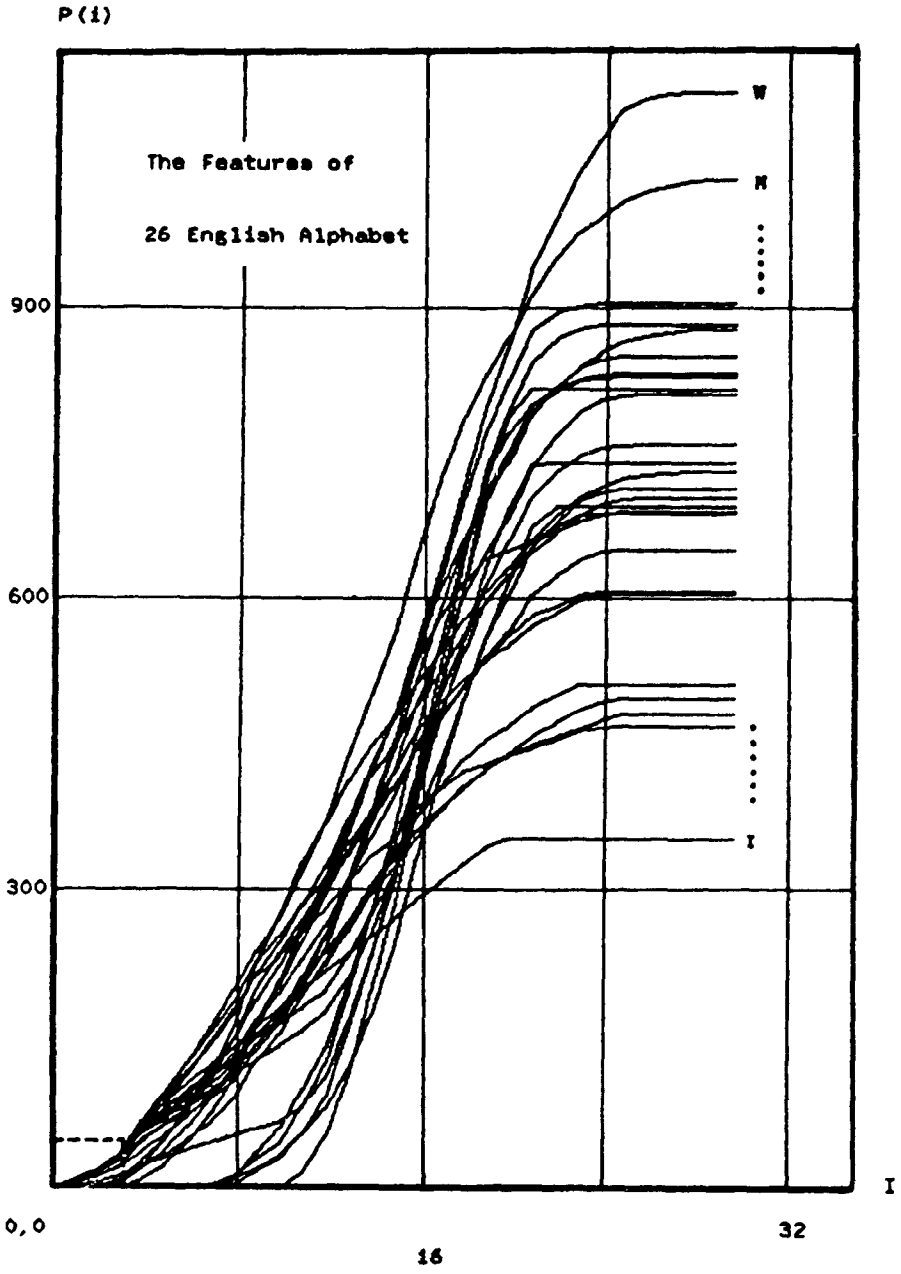


Fig. 5.6 Features of 26 English Alphabetic Letters Extracted by Proposed Algorithm

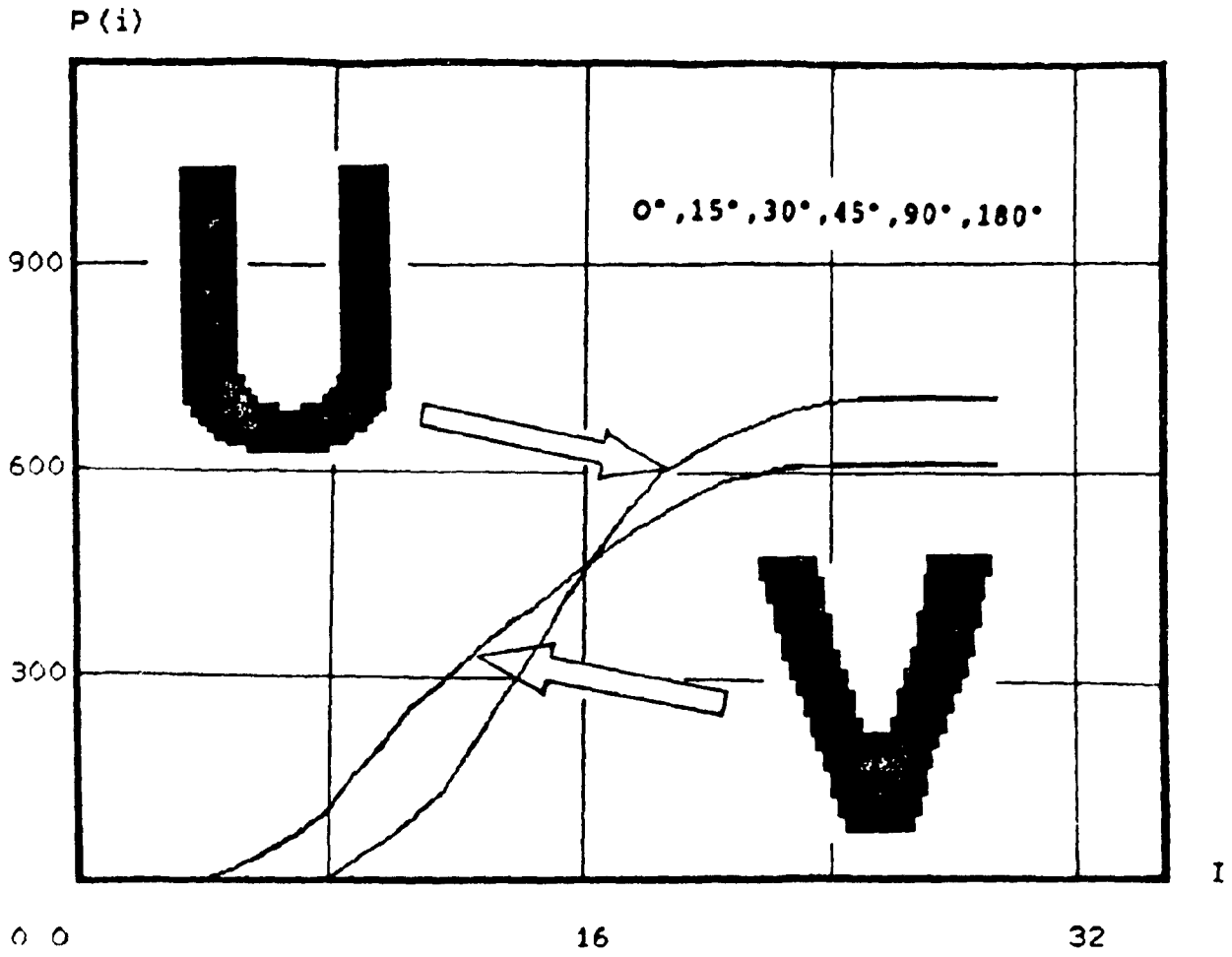


Fig. 5.7 Results on Two Similar Letters (U, V) Rotated Through Different Degrees

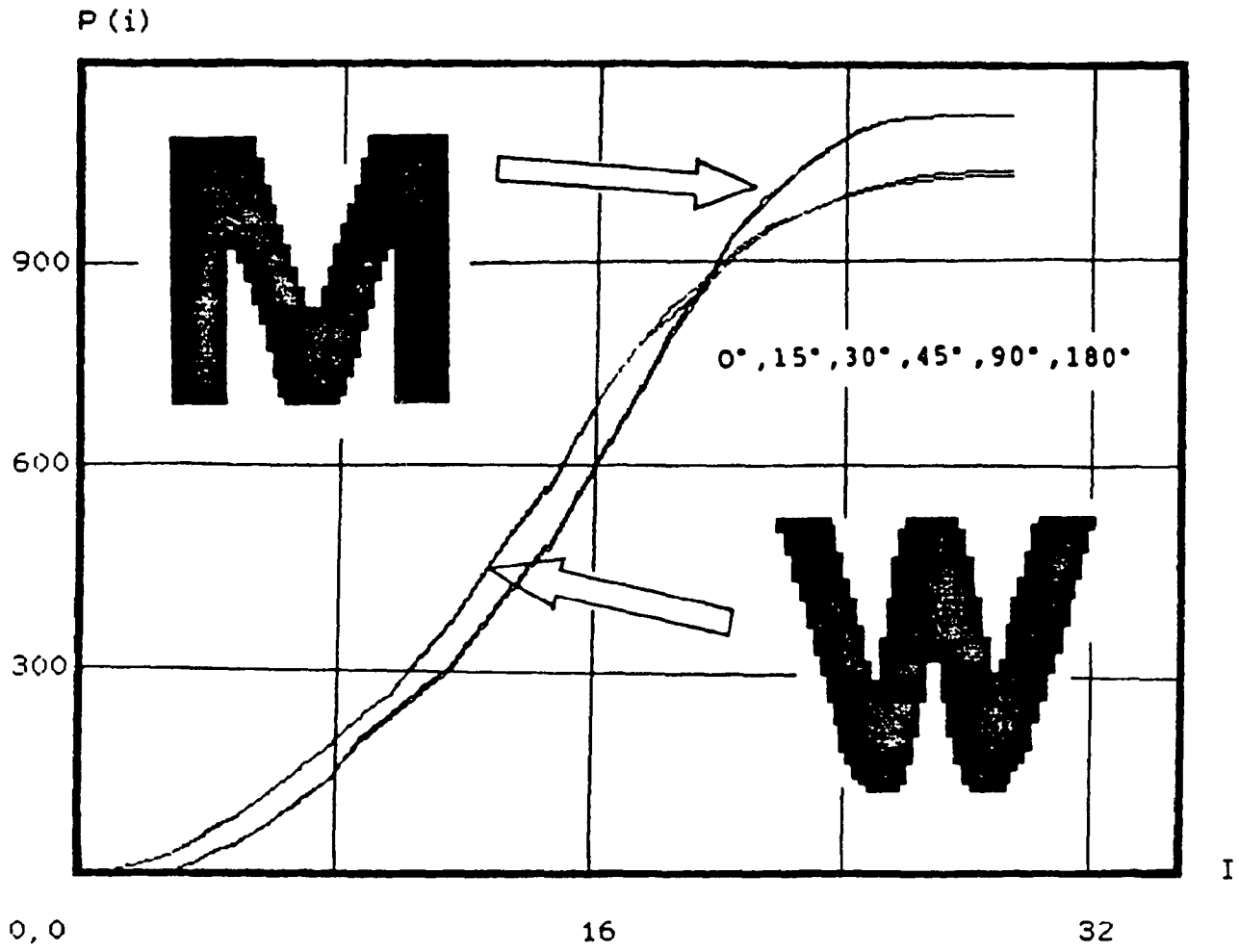


Fig. 5.8 Results on Two Similar Letters (W, M) Rotated Through Different Degrees

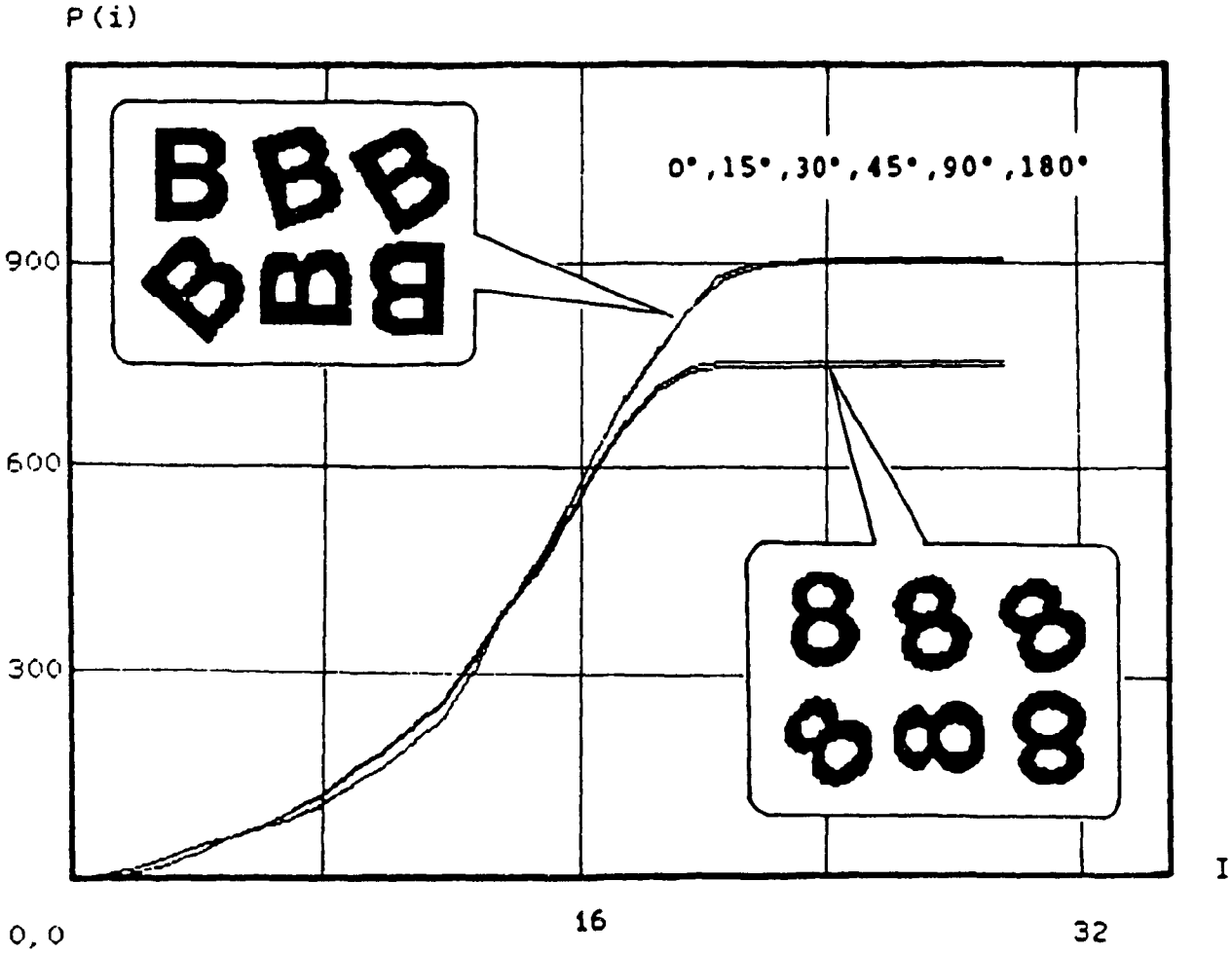


Fig. 5.9 Results on Two Similar Letters (B, 8) Rotated Through Different Degrees

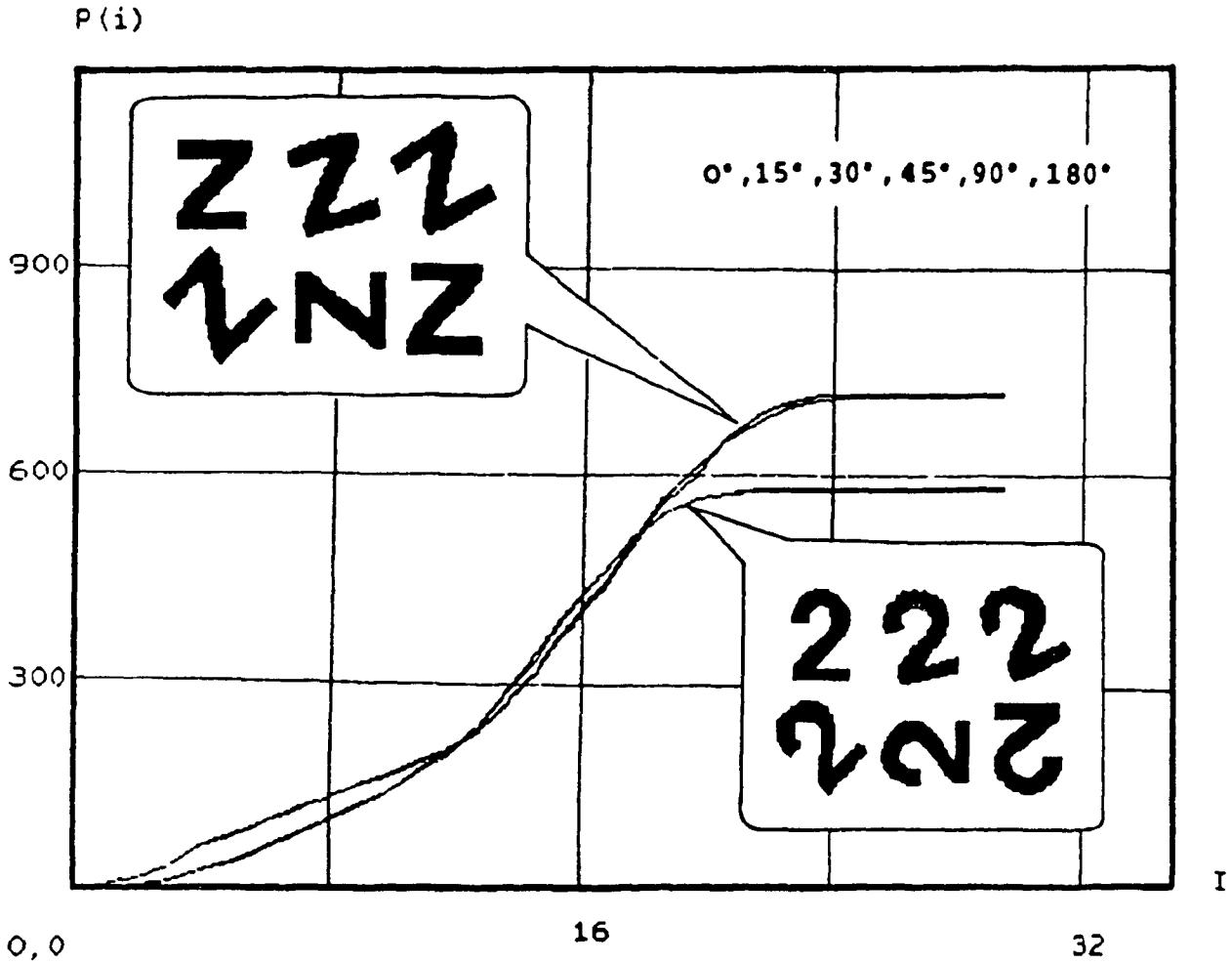


Fig. 5.10 Results on Two Similar Letters (Z, 2) Rotated Through Different Degrees

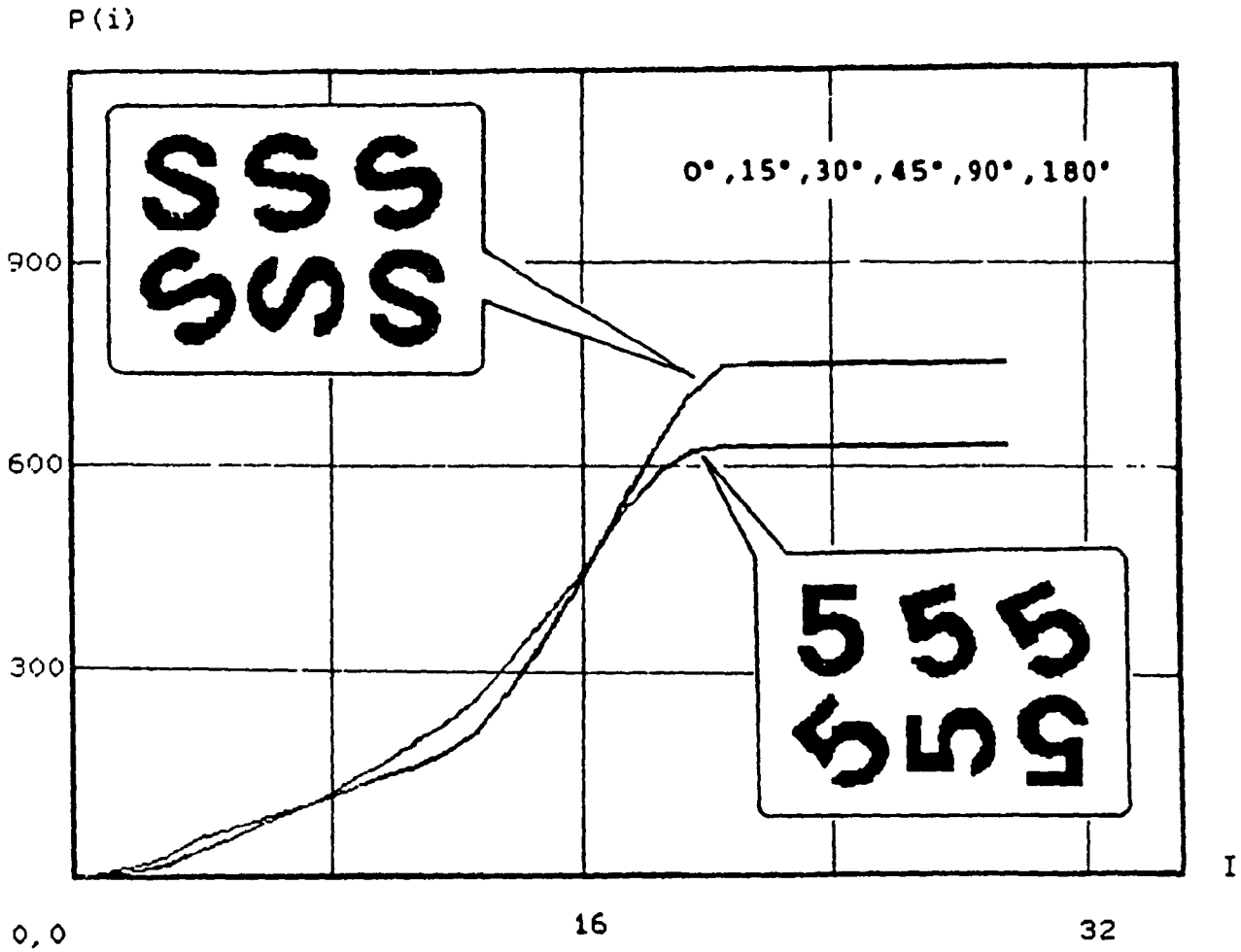


Fig. 5.11 Results on Two Similar Letters (S, 5) Rotated Through Different Degrees

CHAPTER 6

REMOVAL OF UNCERTAINTY FROM IS2 : NONLINEAR SHAPE INVARIANT ALGORITHM

6.1 INTRODUCTION

As mentioned in Chapter 2, one of the common information source uncertainty is shape distortion which exists in the second level of MLIS. Shape distortion uncertainty can be divided into two types : linear and nonlinear [Tang88b, 89b, Sawchu72, Pavlid82, Lee87, Li89c, Gonzal87, Hearn86, Rogers76, Rosenf82, Foley82]. Linear shape distortions such as size and orientation variances have been solved by the previous chapter and other works [Suen80, Kahan87, Psalti77, Wu86, Cormar63, 64, Merser86, Hansen81, Hsu82]. The solution to correct nonlinear shape distortions is also a most difficult, significant and challenging topic in the area of computer vision, robot vision and moving pattern recognition. It is the purpose of this chapter to discuss these distortions.

Nonlinear shape distortions may introduce uncertainty to the pattern set, i.e., the entropy of pattern set is increased, and may confuse the design of the recognition system. These distortions will been regarded as entropy-increased transformation (EIT) described in section 6.2. Several algorithms will been presented in Sections 6.3 - 6.5, which were derived from the finite element method [Haber78,

Strang73, Zienki77, Martin73, Norrie73], to perform the nonlinear shape transformations including bilinear, quadratic, cubic, bi-quadratic and bi-cubic models.

Finally, the proposed entropy-reduced transformation, i.e. nonlinear shape restoration using the inverse nonlinear shape transformation algorithms will be described in section 6.6.

The material of this chapter will appear in [Tang90].

6.2 NONLINEAR DISTORTION

Definition 6.1

Let $\Omega_s = (W^s, P_W{}^s, H_W{}^s)$ represent an entropy space of the standard data set and $\Omega_d = (W^d, P_W{}^d, H_W{}^d)$ stand for an entropy space of the distorted data set. Let $W_i{}^s$ be a standard pattern in W^s, and $W_i{}^d$ be a distorted pattern of $W_i{}^s$ in W^d. This means that $W_i{}^d$ is a mapping of pattern $W_i{}^s$ from space Ω_s into Ω_d:

$$W_i{}^d = F_{sd} \quad W_i{}^s. \tag{6.1}$$

We will call F_{sd} the distortion transformation if the conditions of entropy-increased transformation according to Eqs.(4.4a) and (4.4c) are satisfied in F_{sd} .

Definition 6.2

In Eq. (6.1), the distortion on the pattern $W_i{}^d$ will be called bilinear distortion if F_{sd} is a bilinear transformation.

Definition 6.3

In Eq. (6.1), the distortion on the pattern $W_i{}^d$ will be called quadratic distortion if F_{sd} is a quadratic transformation.

Definition 6.4

In Eq. (6.1), the distortion on the pattern $W_i{}^d$ will be called bi-quadratic distortion if F_{sd} is a bi-quadratic transformation.

Definition 6.5

In Eq. (6.1), the distortion on the pattern $W_i^{\,d}$ will be called cubic distortion if F_{sd} is a cubic transformation.

Definition 6.6

In Eq. (6.1), the distortion on the pattern $W_i^{\,d}$ will be called bi-cubic distortion if F_{sd} is a bi-cubic transformation.

In order to overcome nonlinear distortion, we have to discuss the corresponding transformation. In chapter 4, the basic concept of bilinear, quadratic, cubic, bi-quadratic and bi-cubic transformations has been introduced. Here we will describe several algorithms with details. They can be employed in the area of computer vision, robot vision and the recognition of moving patterns to remove the uncertainty produced by nonlinear shape distortions.

6.3 BILINEAR TRANSFORMATION ALGORITHMS

First, we will define some notations and terminologies commonly used in the following sections.

The mathematical symbols are defined in the following list :

D^e	Element or cell;
P_i	The i^{th} node in element D^e ;
P_{ij}	The middle node between the i^{th} and j^{th} nodes in element D^e ;
(ξ, η)	Coordinates representing the normalized image;
(X, Y)	Coordinates representing the distorted image;
(X_{P_i}, Y_{P_i})	Coordinate value of node P_i;
$F_{XY \rightarrow \xi\eta}$	Entropy-reduced filter;
$\Phi_i (P)$	Shape function which will be given in definition 6.7;
$\Delta P_i P_j P_k$	Reference triangle which will be described by definition 6.8.
$\square P_i P_j P_k P_l$	Reference quadrilateral which will be described by definition 6.9.

Now, let us define the most important function, that is shape function $\Phi (P)$ of element D^e :

Definition 6.7

Let $\Phi_i\ (P_j)$ be a polynomial in element D^e, $i = 1, 2, \cdots, p$. We call $\Phi_i\ (P)$ the shape function of element D^e, if the following conditions are satisfied:

(i) $\sum_{i=1}^{p} \Phi_i\ (P_j) = 1;$

(ii) Any polynomial $\Phi_i\ (P)$ in element D^e has the value 1 in node P_i but takes the value 0 in the remaining $p-1$ nodes, i.e.

$$\Phi_i\ (P_j) = \delta_{ij}\ . \qquad (i, j = 1, 2, \cdots, p)$$

The shape function $\Phi_i\ (P)$ can represent those functions related to coordinates (X, Y) or (ξ, η).

With a view to discussing this method conveniently we will introduce several useful terminologies : reference triangle, reference quadrilateral and reference point.

Definition 6.8

Let $\Delta P_1P_2P_3$ and $\Delta P_1'P_2'P_3'$ be two triangles in XOY and $\xi O\eta$ respectively as shown in Fig. 6.1. The following three pairs of points are one-to-one correspondences in transformations:

$$P_1 \longleftrightarrow P_1',\quad P_2 \longleftrightarrow P_2',\quad P_3 \longleftrightarrow P_3'$$

We define $\Delta P_1P_2P_3$ and $\Delta P_1'P_2'P_3'$ as reference triangles.

The corner points $P_1, P_2, P_3, P_1', P_2', P_3'$ of the two triangles are called the reference points.

Definition 6.9

Let $\square P_1P_2P_3P_4$ and $\square P_1'P_2'P_3'P_4'$ be two quadrilaterals in XOY and $\xi O\eta$ respectively as shown in Fig. 6.2. Four pairs of points are one-to-one correspondences in transformations.

$$P_1 \longleftrightarrow P_1', \quad P_2 \longleftrightarrow P_2', \quad P_3 \longleftrightarrow P_3', \quad P_4 \longleftrightarrow P_4'$$

We define $\square P_1P_2P_3P_4$ and $\square P_1'P_2'P_3'P_4'$ as reference quadrilaterals. The corner points $P_1, P_2, P_3, P_4, P_1', P_2', P_3', P_4'$ of the two quadrilaterals are called the reference points.

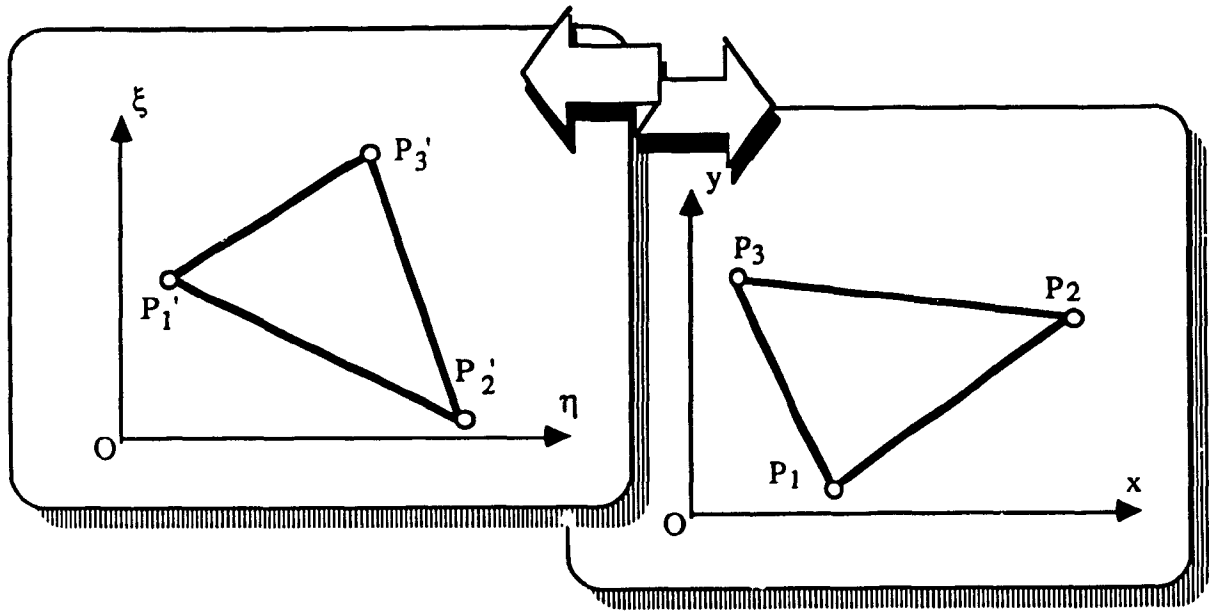


Fig. 6.1 Reference Triangles and Points

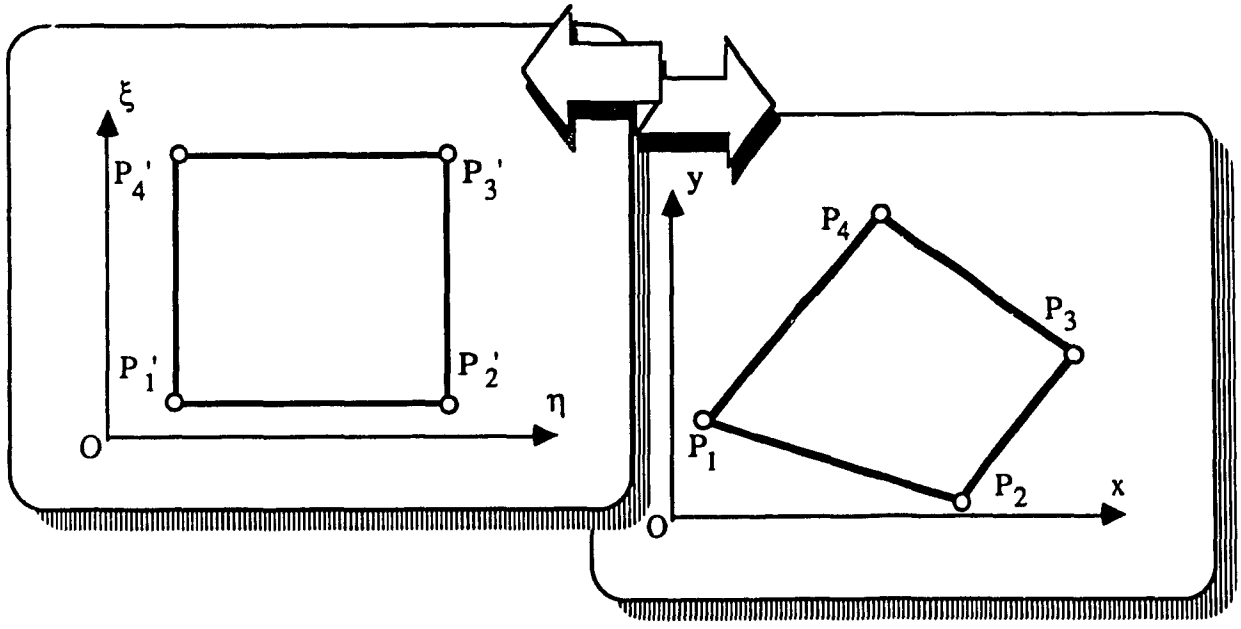


Fig. 6.2 Reference Quadrangles and Points

6.3.1 8-Coefficient Algorithm

In order to obtain the desired bilinear transformation

$$T : (\xi, \eta) \rightarrow (x, y) ,$$

$$x = f (\xi, \eta) ,$$
$$y = g (\xi, \eta) ,$$

we have to find the eight coefficients $a_{11}, a_{12}, \cdots r, s$ in the following equations

$$\begin{aligned} x &= f (\xi, \eta) \\ &= a_{11} \xi + a_{12} \xi\eta + a_{22} \eta + r , \\ y &= g (\xi, \eta) \\ &= b_{11} \xi + b_{12} \xi\eta + b_{22} \eta + s . \end{aligned} \tag{6.2a}$$

The easy way is to find eight special points in the two coordinate systems $\xi 0 \eta$ and $X0Y$. Here we choose four pairs of special one-to-one mapping points in $\xi 0 \eta$ and $X0Y$.

To simplify the computation without loss of generality, we let the reference quadrilateral $\square P_1' P_2' P_3' P_4'$ be a unit square in the standard coordinate system $\xi O \eta$, and the coordinates of their reference points be:

$$P_1' = (0,0), \ P_2' = (1,0), \ P_3' = (1,1), \ P_4' = (0,1) . \tag{6.2b}$$

Based on Eqs. (6.2a) and (6.2b), we can easily obtain the constants in the bilinear transformation :

$$r = X_{P_1} ,$$
$$a_{11} = X_{P_2} - X_{P_1} ,$$

$$
\begin{aligned}
a_{22} &= X_{P_4} - X_{P_1}, \\
a_{12} &= X_{P_1} + X_{P_3} - X_{P_2} - X_{P_4},
\end{aligned}
\tag{6.3a}
$$

and

$$
\begin{aligned}
s &= Y_{P_1}, \\
b_{11} &= Y_{P_2} - Y_{P_1}, \\
b_{22} &= Y_{P_4} - Y_{P_1}, \\
b_{12} &= Y_{P_1} + Y_{P_3} - Y_{P_2} - Y_{P_4},
\end{aligned}
\tag{6.3b}
$$

In terms of Eqs. (6.2a,b) and (6.3a,b), we have now established a bilinear transformation

$$T : (\xi, \eta) \rightarrow (x, y),$$

6.3.2 4-Node Quadrangle Algorithm

In the above algorithm, the eight coefficients must be found to establish a bilinear transformation. Here another algorithm is proposed, it is based on the finite element principle [Haber78, Strang73, Zienki77, Martin73, Norrie73] which does not need to solve for the coefficients.

The shape function $\Phi_i (P)$ can be taken in the following formulas in the case of unit reference quadrangle.

$$
\begin{aligned}
\Phi_1 (\xi, \eta) &= (1 - \xi)(1 - \eta) \\
\Phi_2 (\xi, \eta) &= \xi (1 - \eta) \\
\Phi_3 (\xi, \eta) &= \xi \eta \\
\Phi_4 (\xi, \eta) &= (1 - \xi) \eta
\end{aligned}
\tag{6.4}
$$

Because the coordinates ξ and η of (ξ_i, η_i) at node P_i in standard coordinate system have values 1 or 0, the shape function can be generalized to produce the following formula :

$$\Phi_i = (-1)^{\xi_i + \eta_i} (\xi + \xi_i - 1)(\eta + \eta_i - 1) \qquad (6.5)$$
$$(i = 1, 2, 3, 4).$$

From shape function $\Phi_i (\xi, \eta)$ we can get the relation between the coordinates of the distorted pattern and those of the standard one.

$$X = \sum_{i=1}^{4} \Phi_i (\xi, \eta) X_{P_i}$$
$$Y = \sum_{i=1}^{4} \Phi_i (\xi, \eta) Y_{P_i} \qquad (6.6a)$$

or

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & X_{P_2} & X_{P_3} & X_{P_4} \\ Y_{P_1} & Y_{P_2} & Y_{P_3} & Y_{P_4} \end{bmatrix} \begin{bmatrix} \Phi_1(\xi,\eta) \\ \Phi_2(\xi,\eta) \\ \Phi_3(\xi,\eta) \\ \Phi_4(\xi,\eta) \end{bmatrix} \qquad (6.6b)$$

Substituting Eq. (6.4) into Eq. (6.6b) results the following formula :

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} 1-\xi & \xi \end{bmatrix} & \begin{bmatrix} 0 & \\ & \end{bmatrix} \\ \begin{bmatrix} 0 & \\ & \end{bmatrix} & \begin{bmatrix} 1-\xi & \xi \end{bmatrix} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} X_{P_1} & X_{P_4} \\ X_{P_2} & X_{P_3} \end{bmatrix} \\ \begin{bmatrix} Y_{P_1} & Y_{P_4} \\ Y_{P_2} & Y_{P_3} \end{bmatrix} \end{bmatrix} \begin{bmatrix} 1-\eta \\ \eta \end{bmatrix} \qquad (6.7)$$

6.4 QUADRATIC AND BI-QUADRATIC ALGORITHMS

6.4.1 Quadratic Model (6-Node Triangle Algorithm)

Let the standard reference triangle $\Delta P_1' P_2' P_3'$ in $\xi 0 \eta$ be non-degenerate, and any point P_i' be the coordinates (ξ, η). We can find three area coordinates λ_1, λ_2, λ_3 (see Fig. 6.3) such that [Wangl79, Li89c]

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \xi_{P_1} & \xi_{P_2} & \xi_{P_3} \\ \eta_{P_1} & \eta_{P_2} & \eta_{P_3} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix}, \tag{6.8a}$$

and

$$\lambda_1 + \lambda_2 + \lambda_3 = 1 . \tag{6.8b}$$

When $\Delta P_1' P_2' P_3'$ is non-degenerate, the determinant

$$\Delta = \begin{vmatrix} \xi_{P_1} & \xi_{P_2} & \xi_{P_3} \\ \eta_{P_1} & \eta_{P_2} & \eta_{P_3} \\ 1 & 1 & 1 \end{vmatrix} = 2\, Area\ \Delta P_1' P_2' P_3' \neq 0 . \tag{6.9}$$

Consequently, we can see

$$\lambda_1 = \frac{1}{\Delta} \begin{vmatrix} \xi & \xi_{P_2} & \xi_{P_3} \\ \eta & \eta_{P_2} & \eta_{P_3} \\ 1 & 1 & 1 \end{vmatrix} = \frac{Area\ \Delta P_i' P_2' P_3'}{Area\ \Delta P_1' P_2' P_3'} , \tag{6.10a}$$

$$\lambda_2 = \frac{1}{\Delta} \begin{vmatrix} \xi_{P_1} & \xi & \xi_{P_3} \\ \eta_{P_1} & \eta & \eta_{P_3} \\ 1 & 1 & 1 \end{vmatrix} = \frac{Area\ \Delta P_1' P_i' P_3'}{Area\ \Delta P_1' P_2' P_3'} , \tag{6.10b}$$

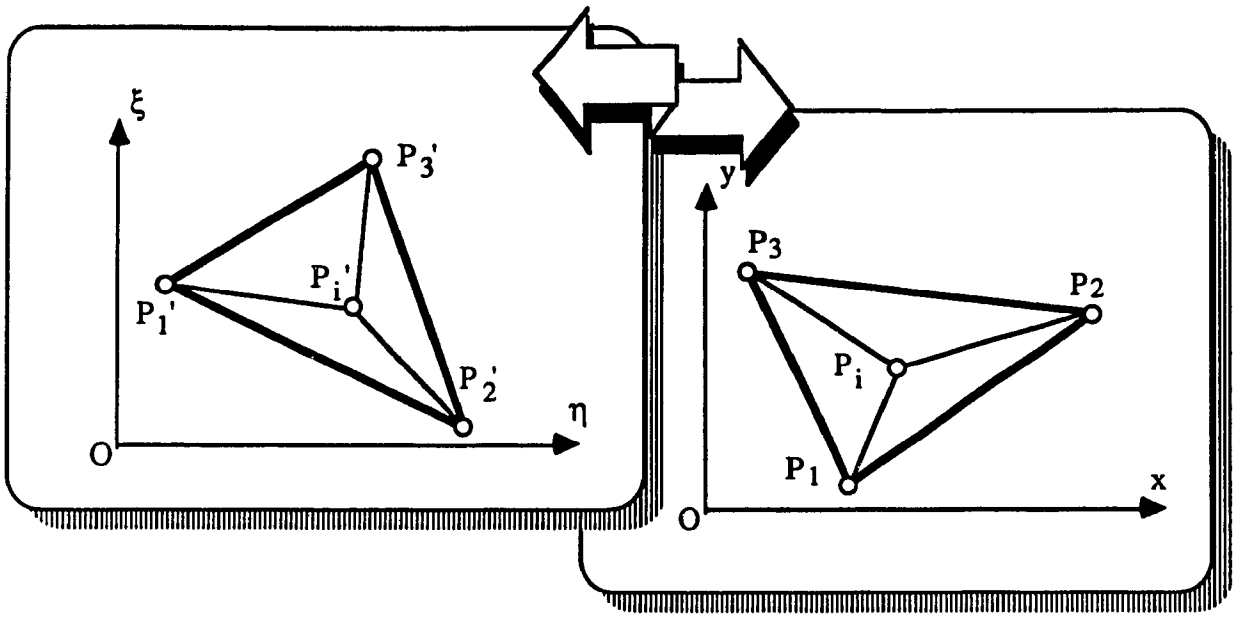


Fig. 6.3 Reference Triangles $\Delta P_1' P_2' P_3'$ and $\Delta P_1 P_2 P_3$

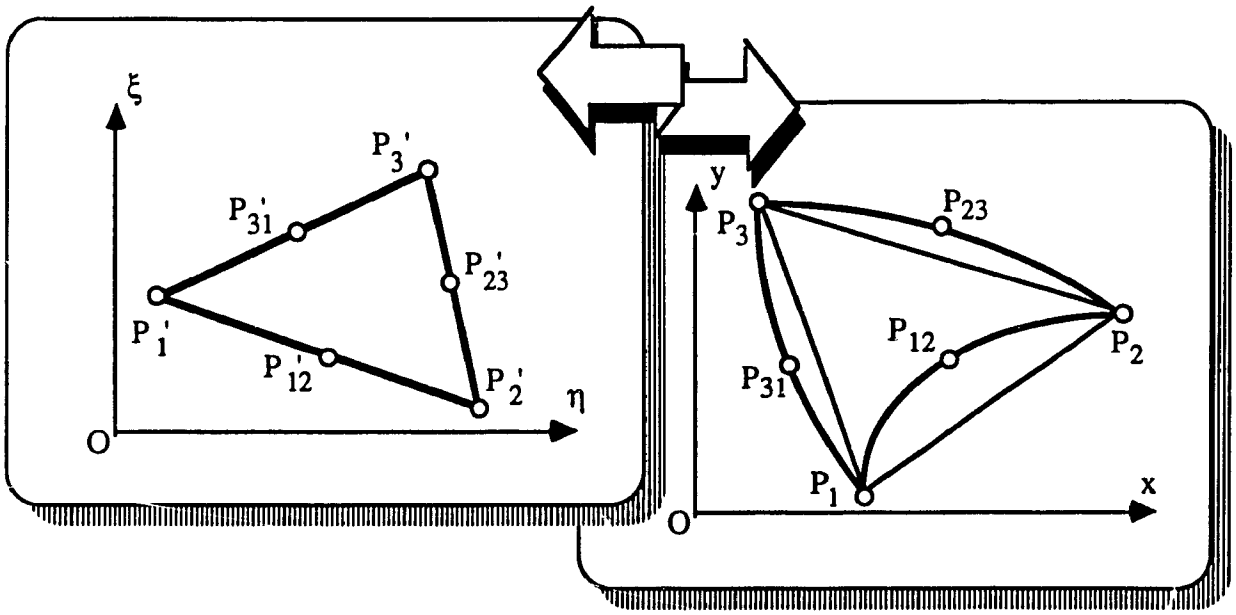


Fig. 6.4 6 - Node Triangle

$$\lambda_3 = \frac{1}{\Delta}\begin{vmatrix} \xi_{P_1} & \xi_{P_2} & \xi \\ \eta_{P_1} & \eta_{P_2} & \eta \\ 1 & 1 & 1 \end{vmatrix} = \frac{Area\ \Delta P_i' P_1' P_2'}{Area\ \Delta P_1' P_2' P_3'}, \tag{6.10c}$$

Suppose that $\Delta P_1 P_2 P_3$ is another reference triangle in XOY, whose vertices (i.e., the reference points) correspond to those of the standard triangle $\Delta P_1' P_2' P_3'$ under the transformation T, that is

$$P_1 \longleftrightarrow P_1' \quad P_2 \longleftrightarrow P_2' \quad P_3 \longleftrightarrow P_3' \tag{6.11}$$

Also we assume that $\Delta P_1 P_2 P_3$ is non-degenerate, then we can establish the follow ng transformation [Wangl79] :

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & X_{P_2} & X_{P_3} \\ Y_{P_1} & Y_{P_2} & Y_{P_3} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{bmatrix}, \tag{6.12}$$

Besides Eq. (6.12), we can establish the quadratic models based on the principle of finite elements [Haber78, Strang73, Zienki77, Martin73, Norrie73]. Denote the midpoints of the boundaries $P_1' P_2'$, $P_2' P_3'$, , $P_3' P_1'$ by the reference points P_{12}', P_{23}', P_{31}'. Accompanied with Eq. (6.11), we also assume the one-to-one relations (Fig. 6.4)

$$P_{12} \longleftrightarrow P_{12}' \quad P_{23} \longleftrightarrow P_{23}' \quad P_{31} \longleftrightarrow P_{31}' \tag{6.13}$$

where P_{12}, P_{23}, P_{31} are the reference midpoints on the boundaries $P_1 P_2$, $P_2 P_3$, $P_3 P_1$. Hence, the shape function Φ_i (P) can be represented in the following formulas from the isoparameters element.

$$\Phi_i = 2\,\lambda_i\,(\,\lambda_i - \frac{1}{2}\,) \qquad (\,i = 1, 2, 3\,)\,;$$

$$\Phi_{ij} = 4\,\lambda_i\,\lambda_j \qquad (\,i \neq j\,)\,. \tag{6.14}$$

From shape function $\Phi_i\,(\,\xi, \eta\,)$ we can get the relation between the coordinates of the distorted pattern and those of the standard one.

$$X = \sum_{i=1}^{3} \Phi_i\,(\,\xi, \eta\,)\,X_{P_i} + \sum_{ij=12,23,31} \Phi_{ij}\,(\,\xi, \eta\,)\,X_{P_{ij}}$$

$$Y = \sum_{i=1}^{3} \Phi_i\,(\,\xi, \eta\,)\,Y_{P_i} + \sum_{ij=12,23,31} \Phi_{ij}\,(\,\xi, \eta\,)\,Y_{P_{ij}} \tag{6.15a}$$

or

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & X_{P_2} & X_{P_3} & X_{P_{12}} & X_{P_{23}} & X_{P_{31}} \\ Y_{P_1} & Y_{P_2} & Y_{P_3} & Y_{P_{12}} & Y_{P_{23}} & Y_{P_{31}} \end{bmatrix} \begin{bmatrix} \Phi_1(\xi,\eta) \\ \Phi_2(\xi,\eta) \\ \Phi_3(\xi,\eta) \\ \Phi_{12}(\xi,\eta) \\ \Phi_{23}(\xi,\eta) \\ \Phi_{31}(\xi,\eta) \end{bmatrix} \tag{6.15b}$$

Substituting Eq. (6.14) into Eq. (6.15b) results the following formula :

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & Y_{P_1} \\ X_{P_2} & Y_{P_2} \\ X_{P_3} & Y_{P_3} \\ X_{P_{12}} & Y_{P_{12}} \\ X_{P_{23}} & Y_{P_{23}} \\ X_{P_{31}} & Y_{P_{31}} \end{bmatrix}^T \begin{bmatrix} \lambda_1\,(\,2\lambda_1 - 1\,) \\ \lambda_2\,(\,2\lambda_2 - 1\,) \\ \lambda_3\,(\,2\lambda_3 - 1\,) \\ 4\lambda_1\lambda_2 \\ 4\lambda_2\lambda_3 \\ 4\lambda_3\lambda_1 \end{bmatrix} \tag{6.16}$$

where λ_1, λ_2, λ_3 are given by Eqs. (6.10a - 6.10c).

6.4.2 Bi-quadratic Model (8-node Quadrangle Algorithm)

In 8-node method, eight nodes must be considered in each reference quadrangle. Apart from the four vertices P_1, P_2, P_3 and P_4 of reference quadrangle $\square P_1P_2P_3P_4$, the middle points of each side, P_{12}, P_{23}, P_{34} and P_{41} shown in Fig. 6.5 are chosen. The choice of nodes in reference quadrangle of standard pattern $\square P_1'P_2'P_3'P_4'$ is similar to reference quadrangle $\square P_1P_2P_3P_4$.

Between the reference quadrangles $\square P_1P_2P_3P_4$ in coordinate system $X0Y$ shown in Fig. 6.5 (a) and $\square P_1'P_2'P_3'P_4'$ in coordinate system $\xi 0\eta$ shown in Fig. 6.5 (b) there exists one-to-one mapping relation. To simplify the computation, the standard reference quadrangle $\square P_1'P_2'P_3'P_4'$ is defined to satisfy the following condition :

$$0 \leq \xi \leq 1 \ , \ 0 \leq \eta \leq 1$$

shown in Fig.6.5(b).

The shape function Φ_i (P) can be represented in the following formulas in the case of unit reference quadrangle.

$$\begin{aligned}
\Phi_1(\xi,\eta) &= (1-\xi)(1-\eta)(1-2\xi-2\eta) \\
\Phi_2(\xi,\eta) &= \xi(1-\eta)(2\xi-2\eta-1) \\
\Phi_3(\xi,\eta) &= \xi\eta(2\xi+2\eta-3) \\
\Phi_4(\xi,\eta) &= \eta(1-\xi)(2\eta-2\xi-1) \\
\Phi_{12}(\xi,\eta) &= 4\xi(1-\xi)(1-\eta) \\
\Phi_{23}(\xi,\eta) &= 4\xi\eta(1-\eta) \\
\Phi_{34}(\xi,\eta) &= 4\xi\eta(1-\xi) \\
\Phi_{41}(\xi,\eta) &= 4\eta(1-\xi)(1-\eta)
\end{aligned} \tag{6.17}$$

From shape function $\Phi_i\ (\xi, \eta)$ we can get the relation between the coordinates of the distorted pattern and those of the standard one.

$$X = \sum_{i=1}^{4} \Phi_i\ (\xi, \eta)\ X_{P_i} + \sum_{ij=12,23,34,41} \Phi_{ij}\ (\xi, \eta)\ X_{P_{ij}}$$
$$Y = \sum_{i=1}^{4} \Phi_i\ (\xi, \eta)\ Y_{P_i} + \sum_{ij=12,23,34,41} \Phi_{ij}\ (\xi, \eta)\ Y_{P_{ij}} \qquad (6.18a)$$

or

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & X_{P_2} & X_{P_3} & X_{P_4} & X_{P_{12}} & X_{P_{23}} & X_{P_{34}} & X_{P_{41}} \\ Y_{P_1} & Y_{P_2} & Y_{P_3} & Y_{P_4} & Y_{P_{12}} & Y_{P_{23}} & Y_{P_{34}} & Y_{P_{41}} \end{bmatrix} \begin{bmatrix} \Phi_1(\xi,\eta) \\ \Phi_2(\xi,\eta) \\ \Phi_3(\xi,\eta) \\ \Phi_4(\xi,\eta) \\ \Phi_{12}(\xi,\eta) \\ \Phi_{23}(\xi,\eta) \\ \Phi_{34}(\xi,\eta) \\ \Phi_{41}(\xi,\eta) \end{bmatrix} \qquad (6.18b)$$

Substituting Eq. (6.17) into Eq. (6.18b) results the following formula :

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & Y_{P_1} \\ X_{P_2} & Y_{P_2} \\ X_{P_3} & Y_{P_3} \\ X_{P_4} & Y_{P_4} \\ X_{P_{12}} & Y_{P_{12}} \\ X_{P_{23}} & Y_{P_{23}} \\ X_{P_{34}} & Y_{P_{34}} \\ X_{P_{41}} & Y_{P_{41}} \end{bmatrix}^T \begin{bmatrix} (1-\xi)(1-\eta)(1-2\xi-2\eta) \\ \xi(1-\eta)(2\xi-2\eta-1) \\ \xi\eta(2\xi+2\eta-3) \\ \eta(1-\xi)(2\eta-2\xi-1) \\ 4\xi(1-\xi)(1-\eta) \\ 4\xi\eta(1-\eta) \\ 4\xi\eta(1-\xi) \\ 4\eta(1-\xi)(1-\eta) \end{bmatrix} \qquad (6.19)$$

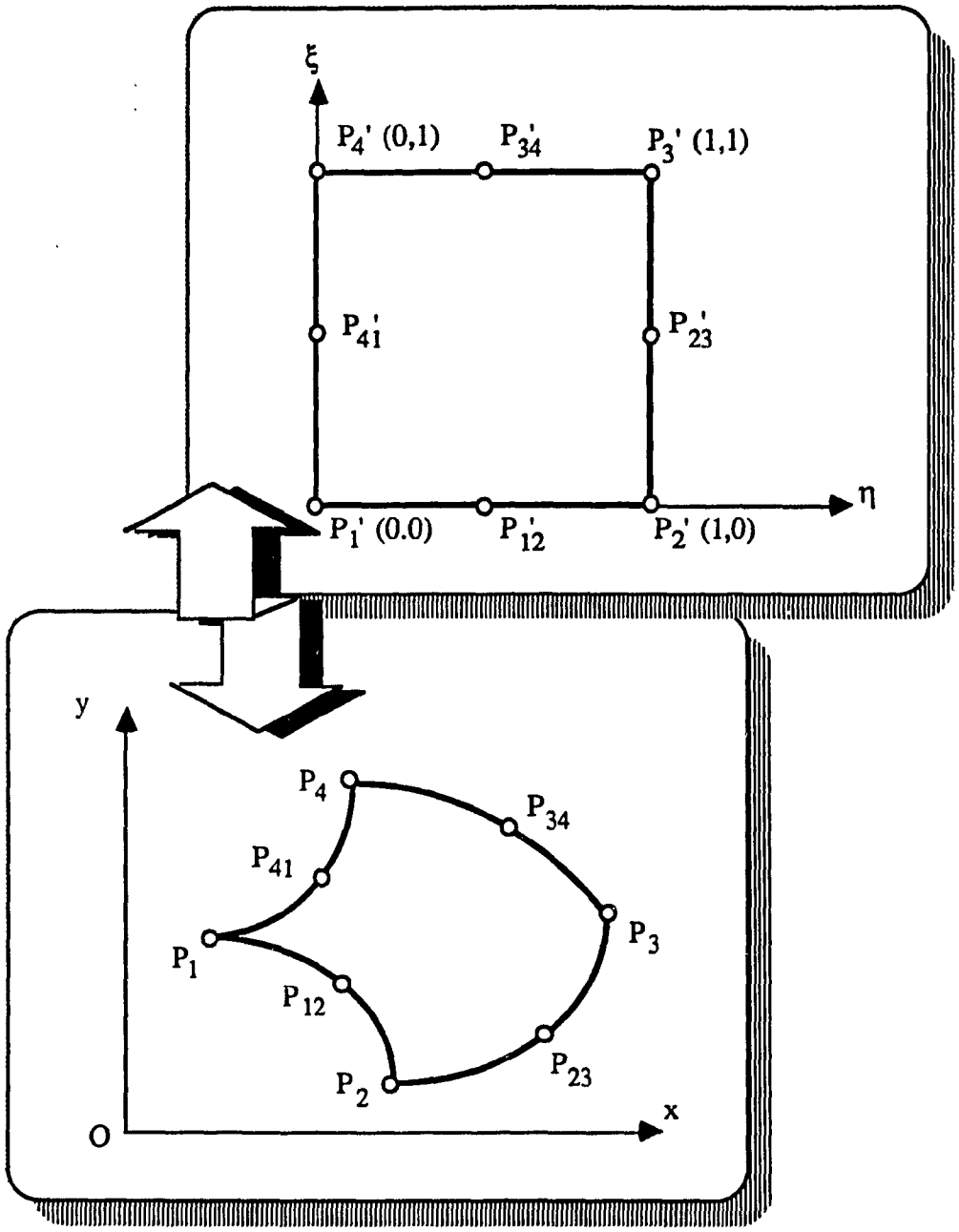


Fig. 6.5 8 - Node Quadrangle

6.5 CUBIC AND BI-CUBIC ALGORITHMS

6.5.1 Cubic Model (9-Node Triangle Algorithm)

The standard reference triangle is given in Fig. 6.6 (a), where P'_{12a} P'_{12b} are located at one, two thirds on the side of $P'_1P'_2$, and so are P'_{23a}, P'_{23b} and P'_{31a} P'_{31b} on the other two sides. Let another reference element be given in Fig. 6.6 (b), and the relations of reference points be

$$\begin{aligned} &P_1 \longleftrightarrow P'_1, \quad P_2 \longleftrightarrow P'_2, \quad P_2 \longleftrightarrow P'_3, \\ &P_{12a} \longleftrightarrow P'_{12a}, \quad P_{23a} \longleftrightarrow P'_{23a}, \quad P_{31a} \longleftrightarrow P'_{31a}, \\ &P_{12b} \longleftrightarrow P'_{12b}, \quad P_{23b} \longleftrightarrow P'_{23b}, \quad P_{31b} \longleftrightarrow P'_{31b}, \end{aligned} \tag{6.20}$$

We can get the shape function Φ_i (P) from isoparameter elements [Wang179]

$$\begin{aligned} \Phi_i &= \frac{9}{2}\lambda_i\left(\lambda_i - \frac{1}{3}\right)\left(\lambda_i - \frac{2}{3}\right) - \frac{9}{2}\lambda_1\lambda_2\lambda_3 \\ &\quad (i = 1, 2, 3); \\ \Phi_{ij} &= \frac{27}{2}\lambda_i\lambda_j\left(\lambda_j - \frac{1}{3}\right) + \frac{27}{4}\lambda_1\lambda_2\lambda_3 \\ &\quad (i \neq j); \end{aligned} \tag{6.21}$$

where λ_i are defined in Eqs. (6.10a-c).

From shape function Φ_i (ξ, η) we can again get the relation between the coordinates of the distorted pattern and those of the standard one.

$$X = \sum_{i=1}^{3} \Phi_i(\xi, \eta) X_{P_i} +$$

$$\sum_{ij=12,23,31} \Phi_{ij}\,(\xi,\eta)\,X_{P_{ij_a}} +$$

$$\sum_{ij=12,23,31} \Phi_{ij}\,(\xi,\eta)\,X_{P_{ij_b}}$$

$$Y = \sum_{i=1}^{3} \Phi_i\,(\xi,\eta)\,Y_{P_i} + \tag{6.22}$$

$$\sum_{ij=12,23,31} \Phi_{ij}\,(\xi,\eta)\,Y_{P_{ij_a}} +$$

$$\sum_{ij=12,23,31} \Phi_{ij}\,(\xi,\eta)\,Y_{P_{ij_b}}$$

Substituting Eq. (6.21) into Eq. (6.22) results the following formula :

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & Y_{P_1} \\ X_{P_2} & Y_{P_2} \\ X_{P_3} & Y_{P_3} \\ X_{P_{12a}} & Y_{P_{12a}} \\ X_{P_{12b}} & Y_{P_{12b}} \\ X_{P_{23a}} & Y_{P_{23a}} \\ X_{P_{23b}} & Y_{P_{23b}} \\ X_{P_{31a}} & Y_{P_{31a}} \\ X_{P_{31b}} & Y_{P_{31b}} \end{bmatrix}^T \begin{bmatrix} \frac{1}{2}\lambda_1(\lambda_1 - 1/3)(\lambda_1 - 2/3) - A \\ \frac{1}{2}\lambda_2(\lambda_2 - 1/3)(\lambda_2 - 2/3) - A \\ \frac{1}{2}\lambda_3(\lambda_3 - 1/3)(\lambda_3 - 2/3) - A \\ \frac{27}{2}\lambda_1\lambda_2(\lambda_1 - 1/3) + B \\ \frac{27}{2}\lambda_1\lambda_2(\lambda_2 - 1/3) + B \\ \frac{27}{2}\lambda_2\lambda_3(\lambda_2 - 1/3) + B \\ \frac{27}{2}\lambda_2\lambda_3(\lambda_3 - 1/3) + B \\ \frac{27}{2}\lambda_3\lambda_1(\lambda_3 - 1/3) + B \\ \frac{27}{2}\lambda_3\lambda_1(\lambda_1 - 1/3) + B \end{bmatrix} \tag{6.23}$$

where

$$A = \frac{9}{2}\lambda_1\lambda_{23}\,, \qquad B = \frac{27}{4}\lambda_1\lambda_{23}\,.$$

6.5.2 Bi-cubic Model (12-Node Quadrangle Algorithm)

Let the bi-cubic transformation be shown in Fig. 6.7, where P_{12}, P_{12}, etc. are located as one, two thirds of the square boundaries. Then the transformation formulas are as follows

$$Q = \frac{1}{8}\left\{ -10 - 9(2\xi - 1)^2 - (2\eta - 1)^2 \right\}$$

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_{P_1} & Y_{P_1} \\ X_{P_2} & Y_{P_2} \\ X_{P_3} & Y_{P_3} \\ X_{P_4} & Y_{P_4} \\ X_{P_{12}} & Y_{P_{12}} \\ X_{P_{21}} & Y_{P_{21}} \\ X_{P_{34}} & Y_{P_{34}} \\ X_{P_{43}} & Y_{P_{43}} \\ X_{P_{13}} & Y_{P_{13}} \\ X_{P_{24}} & Y_{P_{24}} \\ X_{P_{31}} & Y_{P_{31}} \\ X_{P_{42}} & Y_{P_{42}} \end{bmatrix}^T \begin{bmatrix} (1-\xi)(1-\eta)Q \\ (1-\eta)Q \\ \xi\eta Q \\ (1-\xi)\eta Q \\ \frac{9}{2}\xi(1-\xi)(1-\eta)(2-3\xi) \\ \frac{9}{2}\xi(1-\xi)(1-\eta)(3\xi-1) \\ \frac{9}{2}\xi\eta(1-\eta)(2-3\eta) \\ \frac{9}{2}\xi\eta(1-\eta)(3\eta-1) \\ \frac{9}{2}\xi\eta(1-\xi)(3\xi-1) \\ \frac{9}{2}\xi\eta(1-\xi)(2-3\xi) \\ \frac{9}{2}\eta(1-\xi)(1-\eta)(3\eta-1) \\ \frac{9}{2}\eta(1-\xi)(1-\eta)(2-3\eta) \end{bmatrix} \quad \cdots (24)$$

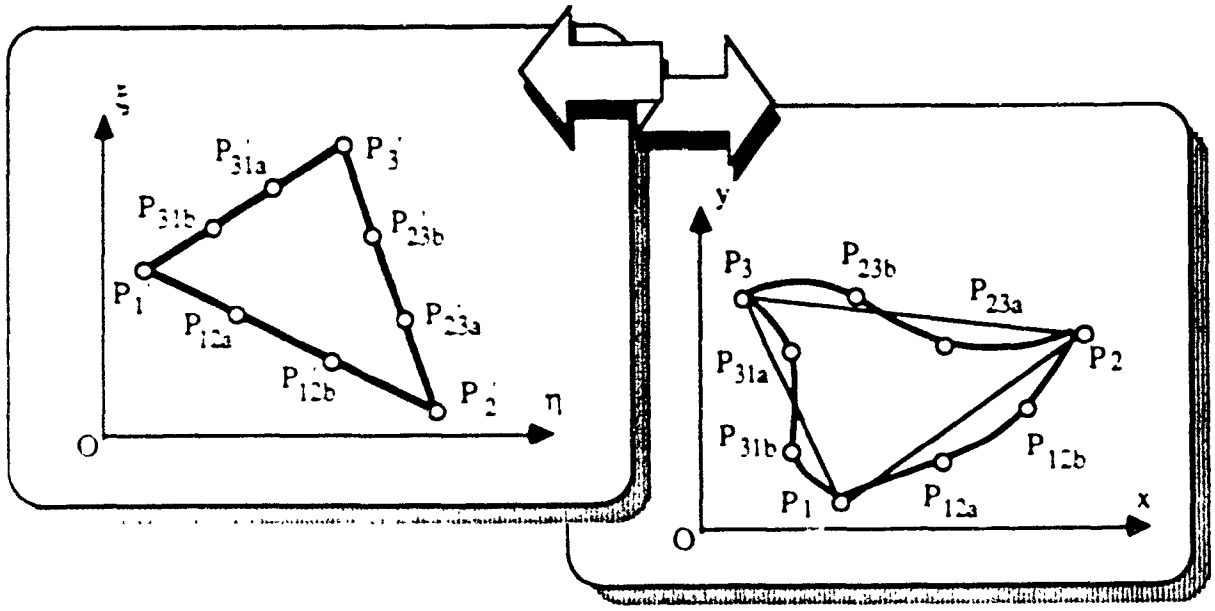


Fig 6 6 9 - Node Triangle

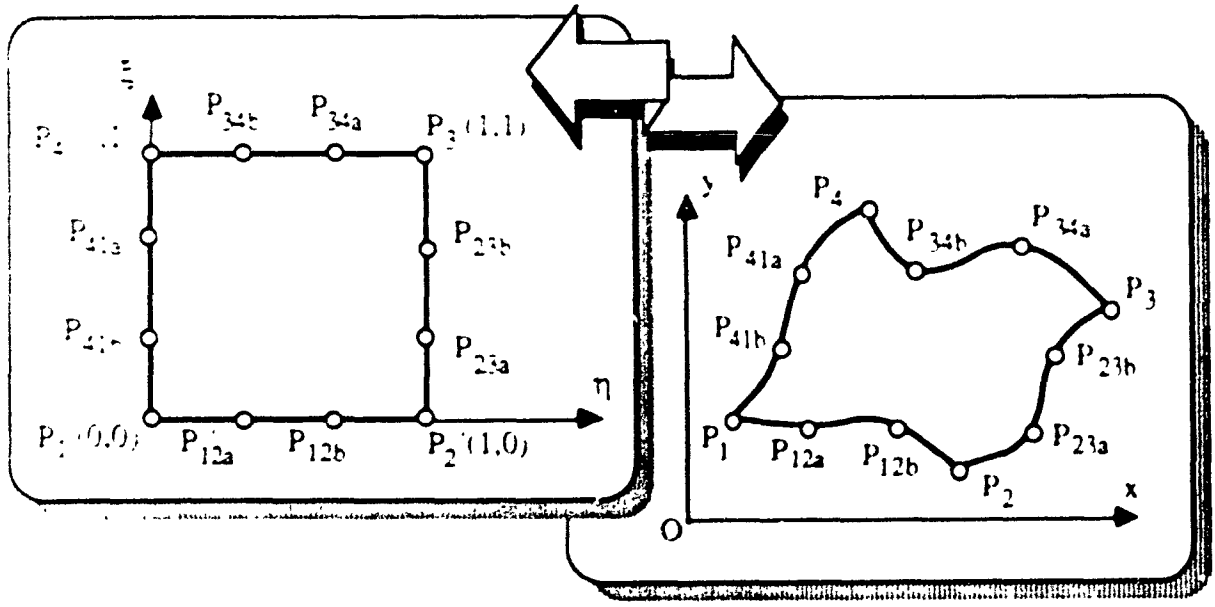


Fig 6 7 12 - Node Quadrangle

6.6 REMOVAL OF NONLINEAR DISTORTIONS

To overcome nonlinear shape distortions including bilinear, quadratic, cubic, bi-quadratic and bi-cubic distortions produced in image processing in the areas of moving pattern recognition, computer vision and robot vision, we have come up with a transformation based on entropy reduction. It uses the inverse nonlinear shape transformation which will be described in this section. In the previous sections, we have mathematically formulated the nonlinear shape distortions in the nonlinear shape transformations. Due to nonlinear shape transformation, a standard image is distorted and produces a variety of images with different shapes. This means that the uncertainty of patterns to be recognized is increased. On this ground, the nonlinear shape transformation is called a kind of entropy-increased transformation. According to theorem 4.1, their inverse transformation can be used as an entropy-reduced transformation. Hence the inverse shape transformation is a suitable candidate for removing the nonlinear shape distortions.

6.6.1 Inverse Shape Transformation

Once the transformation T has been established, its inverse transformation T^{-1} can be obtained easily from the Newton iteration method provided that the transformation T is one-to-one correspondence, i.e., $|J| \neq 0$ is satisfied. The Newton method converges quickly for a good initial approximation [Ortega70, Hagema81]. On the other hand, if the Newton method is convergent, $|J| \neq 0$ must hold true. Besides, the signs of $|J|$ can be evaluated easily during the itera-

tion procedure.

As an example of inverse transformation, we discuss the bilinear shape transformation in some details in this chapter. The Newton iteration method is not only a powerful tool for bilinear inverse transformation but also for other nonlinear inverse transformations as well, such as quadratic, cubic, bi-quadratic and bi-cubic inverse transformations. In case of the bilinear one, there is also a simpler method - the quadratic equation method. We will discuss these two methods below.

Several interesting experimental results of bilinear, bi-quadratic and bi-cubic inverse shape transformations will be given in this section.

6.6.1.1 Newton Iteration Algorithm

The inverse algorithm of bilinear transformation using the Newton iteration method can be realized in three steps:

Step 1: Choose the reference quadrangle $\square P_1P_2P_3P_4$ as in Fig. 6.2, and $\square P_1'P_2'P_3'P_4'$, convex and non-degenerate, their $\square P_1'P_2'P_3'P_4'$ vertices being one-to-one;

Step 2: The positive transformation T is established by the constants given in Eq. (6.3);

Step 3: For any point (X,Y) in XOY, the corresponding point (ξ,η) in (6.3) can be obtained by the Newton Iteration Method:

$$Z^{n+1} = Z^n - J^{-1}(Z^n)F^n, \quad n = 0,1,2, \qquad (6.25)$$

where

$$\vec{Z} = \begin{bmatrix} \xi \\ \eta \end{bmatrix}$$

and $\vec{R}^{(n)}$ is the residual error vector :

$$\vec{R}^{(n)} = \vec{Z}^{(n)} - T\ (\ \xi^{(n)}, \eta^{(n)}\ )\ , \tag{6.26a}$$

and

$$\vec{R}^{(n)} = \begin{bmatrix} r_1^{(n)} \\ r_2^{(n)} \end{bmatrix} = \begin{bmatrix} f\ (\ \xi^{(n)}, \eta^{(n)}\ ) - x \\ g(\ \xi^{(n)}, \eta^{(n)}\ ) - y \end{bmatrix} \tag{6.26b}$$

Choose an initial approximation $Z^{(0)}$, and perform the iteration (6.25) until the errors

$$|\ |\vec{\delta}^{(n+1)} - \vec{\delta}^{(n)}\ |\ | \leq \epsilon$$

where the norm $\|\vec{\delta}\| = (\xi^2 + \eta^2)^{1/2}$, and ϵ is a small number, e.g., $\epsilon = 10^{-8}$.

More concretely, take the bilinear model as an example, we have

$$J\ (\ \vec{Z}^{(n)}\ ) = \begin{bmatrix} a_{11} + a_{12}\eta^{(n)} & a_{22} + a_{12}\xi^{(n)} \\ b_{11} + b_{12}\eta^{(n)} & b_{22} + b_{12}\xi^{(n)} \end{bmatrix}, \tag{6.27}$$

$$\vec{R}^{(n)} = \begin{bmatrix} r_1^{(n)} \\ r_2^{(n)} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{22} \\ b_{11} & b_{12} & b_{22} \end{bmatrix} \begin{bmatrix} \xi^{(n)} \\ \xi^{(n)}\ \eta^{(n)} \\ \eta^{(n)} \end{bmatrix} + \begin{bmatrix} r - x \\ s - y \end{bmatrix}. \tag{6.28}$$

Substituting Eqs. (6.27) and (6.28) into Eq. (6.25) results the following formula:

$$\begin{bmatrix} \xi^{(n+1)} \\ \eta^{(n+1)} \end{bmatrix} = \begin{bmatrix} \xi^{(n)} \\ \eta^{(n)} \end{bmatrix} - \begin{bmatrix} a_{11} + a_{12}\eta^{(n)} & a_{22} + a_{12}\xi^{(n)} \\ b_{11} + b_{12}\eta^{(n)} & b_{22} + b_{12}\xi^{(n)} \end{bmatrix}^{-1} \left(\begin{bmatrix} a_{11} & a_{12} & a_{22} \\ b_{11} & b_{12} & b_{22} \end{bmatrix} \begin{bmatrix} \xi^{(n)} \\ \xi^{(n)}\eta^{(n)} \\ \eta^{(n)} \end{bmatrix} + \begin{bmatrix} r - x \\ s - y \end{bmatrix} \right) \tag{6.29}$$

It is known from [Ortega70, Hagema81-1] that when the determinant of J is not zero and the initial value $\xi^{(0)}$, $\eta^{(0)}$ is approximate, the Newton Iteration Method (6.25 or 6.29) is convergent.

6.6.1.2 Quadratic Equation Algorithm

The Newton method is valid for all nonlinear models described in this chapter. In this section, we will introduce a simpler method : the quadratic equation method. It will be suitable for the bilinear model.

In this method, the first and second steps are the same as the Newton iteration method; but the third step is different in solving a quadratic equation. Let us introduce step 3 by the substeps given below.

Substep 1 :

We have from Eq. (6.2a)

$$\xi = \frac{N - (a_{22}\eta + r)}{a_{11} + a_{12}\eta} \qquad \text{if } a_{11} + a_{12}\eta \neq 0 \tag{6.30}$$

Substep 2 :

Substituting (6.30) into (6.2a) yields a quadratic equation

$$a\ \eta^2 + b\ \eta + c\ = 0 \tag{6.31a}$$

where the coefficients are

$$a\ = \begin{vmatrix} b_{22} & b_{12} \\ a_{22} & a_{12} \end{vmatrix}, \tag{6.31b}$$

$$b\ = \begin{vmatrix} b_{22} & b_{11} \\ a_{22} & a_{11} \end{vmatrix} + \begin{vmatrix} s - Y & b_{12} \\ r - X & a_{12} \end{vmatrix}; \tag{6.31c}$$

$$c\ = \begin{vmatrix} s - Y & b_{11} \\ r - X & a_{11} \end{vmatrix}, \tag{6.31d}$$

Substep 3 :

If $a\ = 0$, we have

$$\eta = -\frac{c}{b}\ ; \tag{6.32a}$$

otherwise

$$\eta_{\pm} = \left[-b \pm (b^2 - 4ac)^{1/2} \right] / 2a\ . \tag{6.32b}$$

When Eq. (4.17) holds, the one-to-one property of T guarantees the existence of the solutions $\eta_{\pm}$ in Eq. (6.32). We note that only one of η_+ and η_- is acceptable. Then the solution of ξ is obtained from Eq. (6.30). The solutions of neighbour pixels (ξ^*, η^*) can be employed to choose either η_+ or η_- as the solution.

6.6.1.3 Experimental Results

Fig. 6.8 illustrates the results of inverse bilinear shape transformation which normalizes the distorted images with different shape functions into a standard one. The results for both the bi-quadratic and bi-cubic inverse shape transformations are presented in Figs. 6.9 and 6.10 respectively.

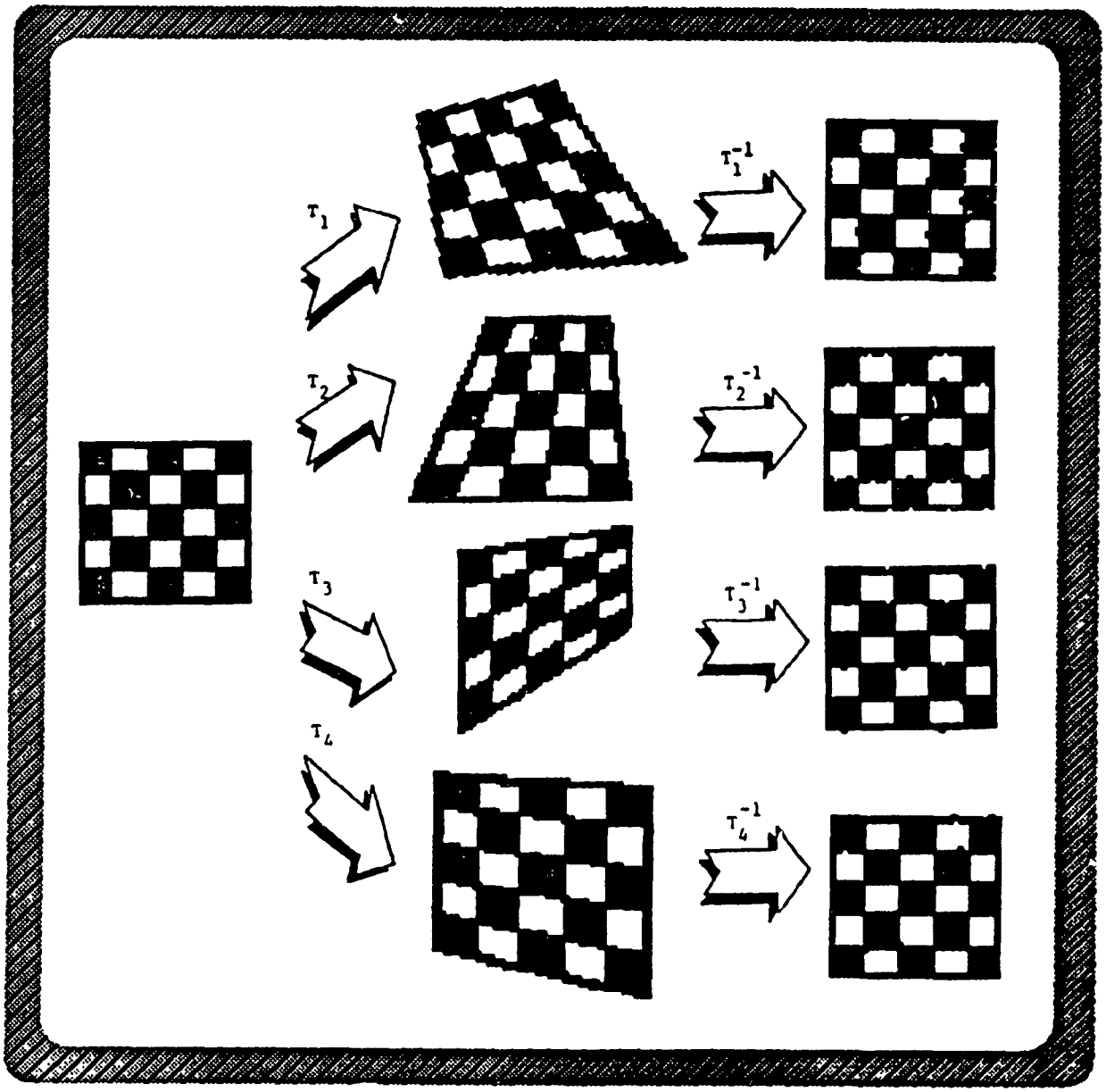


Fig. 6.8 Examples of Inverse Bilinear Shape Transformation

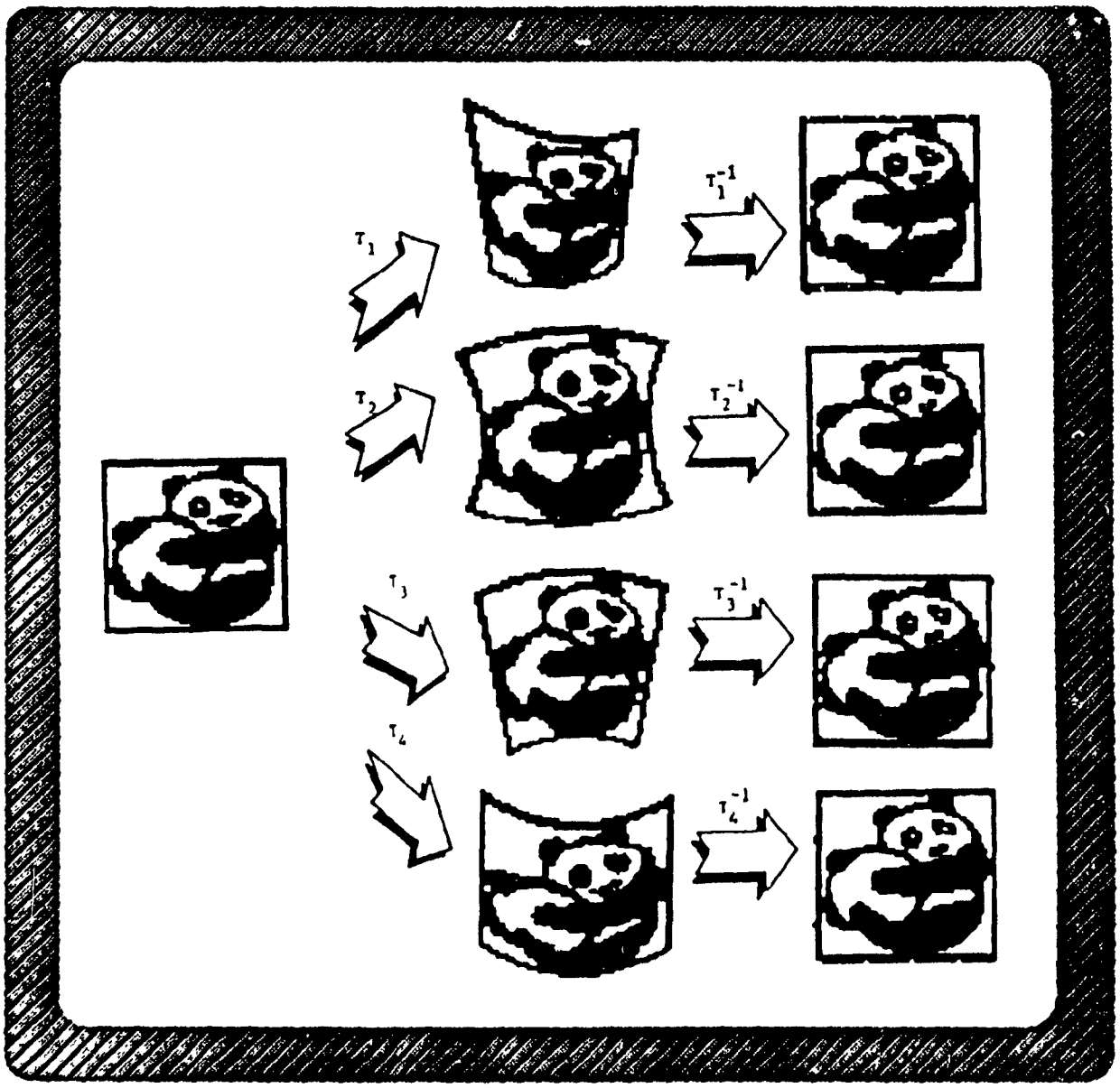


Fig. 6.9 Examples of Inverse Bi-quadratic Shape Transformation

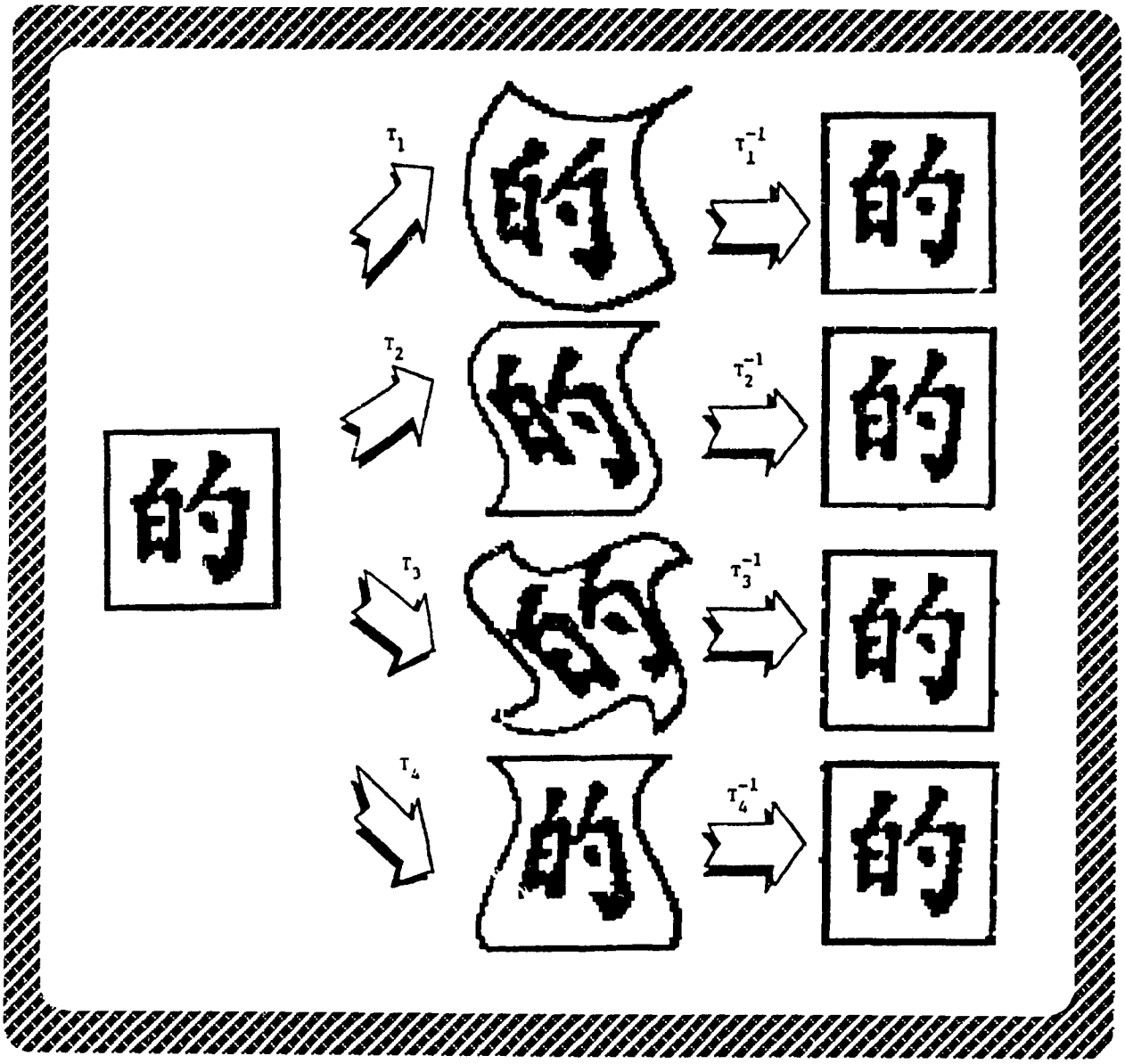


Fig. 6.10 Examples of Inverse Bi-cubic Shape Transformation

6.6.2 Entropy-Reduced Transformation for Nonlinear Shape Distortion

From Eqs. (6.2), (6.25) or (6.30 - 6.32), an inverse bilinear transformation has been achieved. We call it entropy-reduced transformation or entropy-reduced filter, specifically bilinear distortion filter $F^{BL}_{XY\to\xi\eta}$, such that :

$$\begin{bmatrix}\xi\\ \eta\end{bmatrix} = F^{BL}_{XY\to\xi\eta}\begin{bmatrix}X\\ Y\end{bmatrix} \tag{6.33}$$

From Eq. (6.33) we can see that a distorted image passes through the filter $F^{BL}_{XY\to\xi\eta}$, its bilinear distortion will be filtered away to produce a standard image. The images with various bilinear distortions can be normalized into a standard one by use of the bilinear filter with different parameters. That means the entropy has fallen down. For example, given a $\Omega_1 = (W^1, P_{W^1}, H_{W^1})$, such that

$$W^1 = \{ x_j^i \mid i = 1,2,3,4; \quad j = 1,2,\ldots,8 \},$$
$$P_{W^1}(x_j^i) = 1 / (4 \times 8) = 1 / 2^5, \quad i = 1,2,3,4; \quad j = 1,2,\ldots,8,$$
$$H_{W^1} = -\sum_{k=1}^{32} (1 / 2^5) \log (1 / 2^5)$$

where the superscript i represents the different shapes due to the bilinear distortion of the pattern samples and the subscript j represents the class of the pattern samples. That means x_1^1 is similar to x_1^2 except the distortion. Next, let us denote x_j^1 as a standard pattern sample. Now we select all the elements x_j^1, $(j = 1,2,\ldots,8)$ in W^1 as the reference set W^0. That is

$$W^0 = \{ x_j^1 \mid j = 1,2,\ldots,8 \},$$
$$P_{W^0}(x_j^1) = 1 / 8 = 1 / 2^3, \quad j = 1,2,\ldots,8,$$
$$H_{W^0} = -\sum_{j=1}^{8} (1 / 2^3) \log (1 / 2^3) .$$

We define F_{10} as follows

$$F_{10}(x_j^i) = \begin{cases} x_j^i & \text{if } x_j^i = x_j^1; \\ F^{BL}_{XY \to \xi\eta}\ x_j^i & \text{if } x_j^i \neq x_j^1. \end{cases}$$

This means that F_{10} groups four elements in the W^1 into one cluster which corresponds to one element in W^0, whereby

$$| F_{10}(W^1) | = | W^0 |.$$

On the other hand,

$$P_{F_{10}(W^1)}(F_{10}{}^j(W^1)) = 4 \times \frac{1}{2^5} = \frac{1}{2^3},$$

$$H_{F_{10}(W^1)} = H_{W_0}.$$

Therefore F_{10} is an ERT. From this point of view we can see this filter belongs to one kind of entropy-reduced transformation.

CHAPTER 7

DESIGNING PATTERN RECOGNITION SYSTEM BY MLIS AND ERT MODELS

7.1 INTRODUCTION

To demonstrate the application of the multiple-level information source (MLIS) and the entropy-reduced transformation (ERT) models for the design of practical pattern recognition system, two examples are presented in this chapter.

In the first example, a recognition system identifying the complex data set which includes Chinese characters and English letters with size and orientation problems has been designed. Some experimental results have been given [Tang88c, 89a, Qu88].

A design for the system to recognize the object with perspective projection distortion in the computer vision, robot vision, and motion has been presented in the second example.

A approximate method which treats the perspective projection distortion as a bilinear or bi-quadratic or bi-cubic distortion has been presented in this chapter also.

7.2 RECOGNITION SYSTEM FOR COMPLEX DATA SET

7.2.1 Description

Our task is to design an integrated character discriminator. The major user's requirements are as follows. The target data set to be recognized includes 3200 Chinese characters, 52 Roman letters (26 uppercase plus 26 lowercase letters), and 10 numerals. Due to the necessity to process real life samples like those shown in Figs. 7.1 and 7.2, all Chinese characters, Roman letters, and numerals are allowed to have 10 different sizes plus a rotation of α degrees, where $\alpha = 1^\circ$, 2°, ..., 360°. The Chinese character set is also allowed to have the 3 major fonts, Kai, Song and Bold. The input transducer is a MICROTEX - MS - 200 data capture system.

7.2.2 Analysis

(1) Intrinsic characteristics in the first level of information source IS1 :

(i) 10 sizes : for the j-th size, j = 1, 2, ..., 10 we have $| W^{sj} | = (3200 \times 3 + 52 + 10) \times 360$.

(ii) 360 rotation directions : for the j-th direction, j = 1, 2, ..., 360 , we have $| W^{rj} | = (3200 \times 3 + 52 + 10) \times 10$.

(iii) 3 fonts for Chinese characters : for the j-th font, j = 1, 2, 3, we have $| W^{fj} | = (3200 + 52 + 10) \times 10 \times 360$.

If each size or direction variation is treated as a variable, this target pattern set will be a huge data set totalling more than ten million pattern samples ($(3200 \times 3 + 62) \times 10 \times 360 = 34{,}783{,}200 = 3.5 \times 10^7$). The uncertainty of this set comes up to about 25 bits if all the pattern samples with apriori probability are counted. Such variations will make all the existing methods suitable for single category data set either out of function or very inefficient.

(2) Feature Extraction IS4 :

It is hard to find a method suitable for both size variation and rotation variation. These variations have to be handled separately.

(3) Noise : can be omitted.

7.2.3 Design

Based on the analysis above, the architecture of the discriminator has been designed as presented in Fig. 7.3. Here, a block diagram of the whole system is shown, detailed design of each subsystem are presented in references [Li89c, Tang88a, b, Wangq84, 85].

By using Algorithm 3.1.

Step-1 We don't need to decompose U.

Step-2 We select W^0 such that it contains only 3200+62 samples with a standard size (one of 10 sizes is selected as the standard size).

Step-3 First we consider the complex data set to be composed of 10 distinct sub-

sets. Each subset contains all the samples with the same size. For each subset W^{sj}, we define a linear transformation as follows

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} D/d_j & 0 \\ 0 & D/d_j \end{bmatrix} \begin{bmatrix} x_j \\ y_j \end{bmatrix}$$

where (X, Y) are the new coordinates of a point for a pattern sample with the standard size, (x_j,y_j) are the coordinates of a point for a pattern sample with size j, D is the standard size, d_j stands for size j, j = 1,2,...,10. This is our F_{j0}. After this the uncertainty of the original complex data set has been reduced to the level of that of a set having $(3200 + 62) \times 360$ possible pattern samples.

Step-4 Because all the W^{sj}'s are disjoint we use the operation ∇p to connect all F_{j0}'s. That is our $F_I = F_{10}\nabla pF_{20}\nabla p \cdots \nabla pF_{100}$. These operations have reduced the uncertainty by about 3.32 bits.

Step-5 A font selector F_F, which is based on the use of a Statistical Equivalent Block (SEB) classifier shown in Fig. 7.4, is employed to recognize different fonts F_1, F_2 and F_3 [Wangq85]. This further reduces the entropy by about 1.59 bits.

Step-6 Then a rotation-invariant transformation (RIT) defined in [Tang88b] is applied to cluster all 360 rotations (an increment of 1 degree each time) of a pattern sample into a unique reference pattern sample which belongs to the reference set W^0. That is our F_E is a rotation-invariant transformation. In this subsystem, the entropy has been decreased by another

8.48 bits. Hence by now the uncertainty of the complex data set has been reduced to the level of that of W^0, i.e. about 11.7 bits.

Step-7 After all these 4 operations listed above, we can now apply the tree-like discriminator F_C [Tang84, Wangq84], which is suitable only for the recognition of a data set of standard size, single font and without rotation.

Simulation of the above design has been tried on a CYBER-835 computer. The experimental results shown in the Table 7.1 fully support our theoretical predictions.

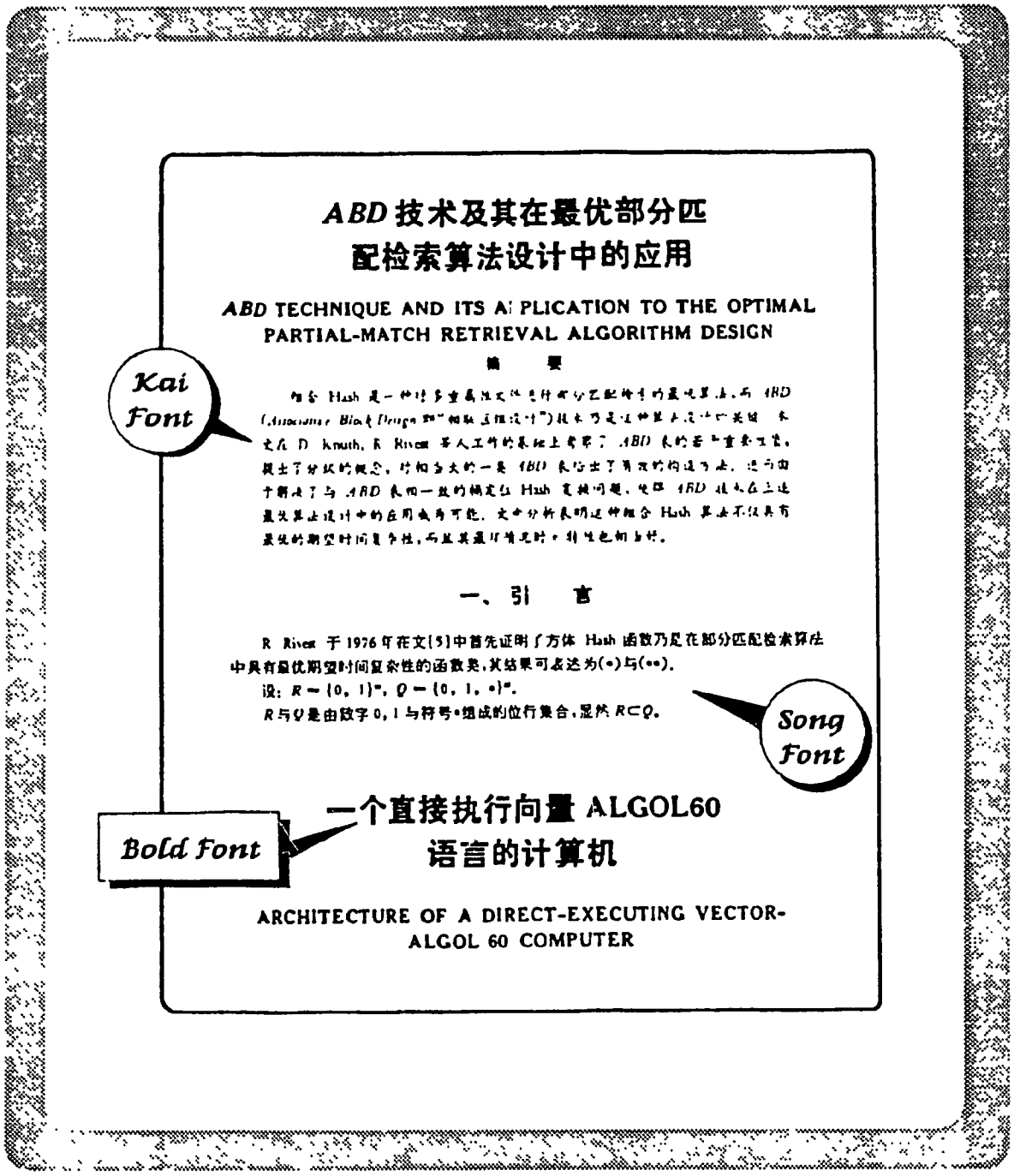


Fig. 7.1 An Example of Complex Data Set with Chinese and Roman Letters

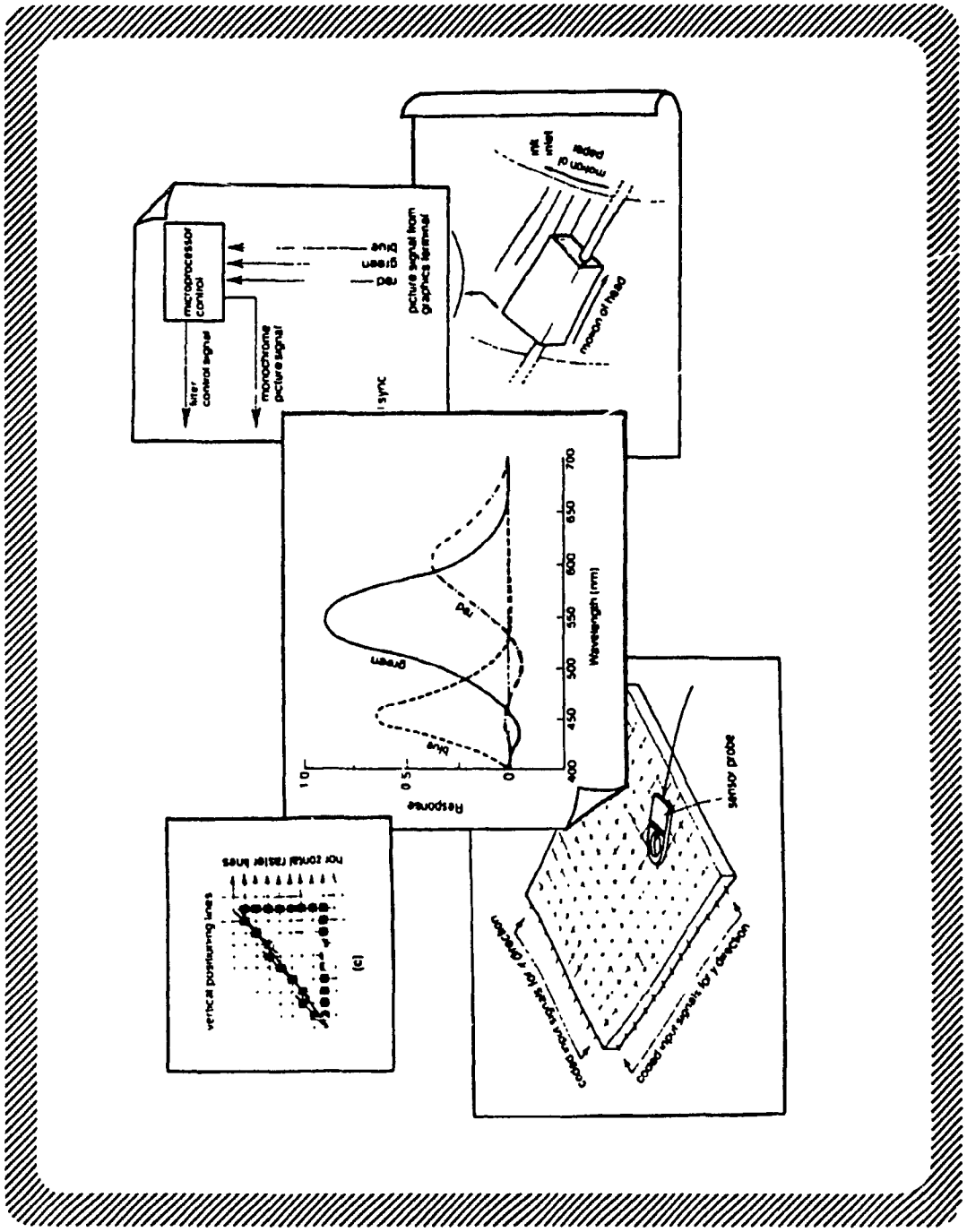


Fig. 7.2 An Example of Complex Data Set with Different Rotations

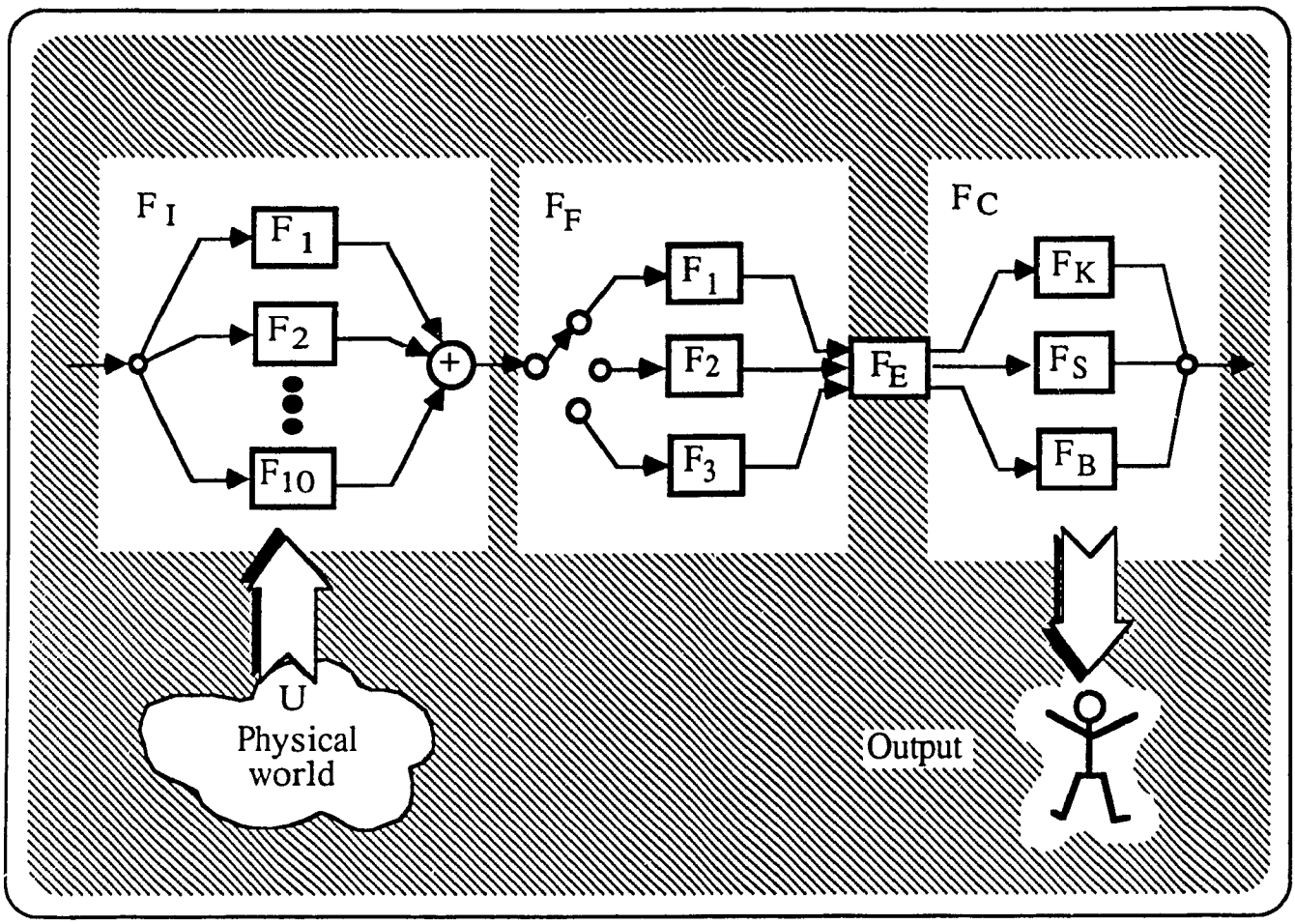


Fig. 7.3 An Architecture of Recognition System for Complex Data Set

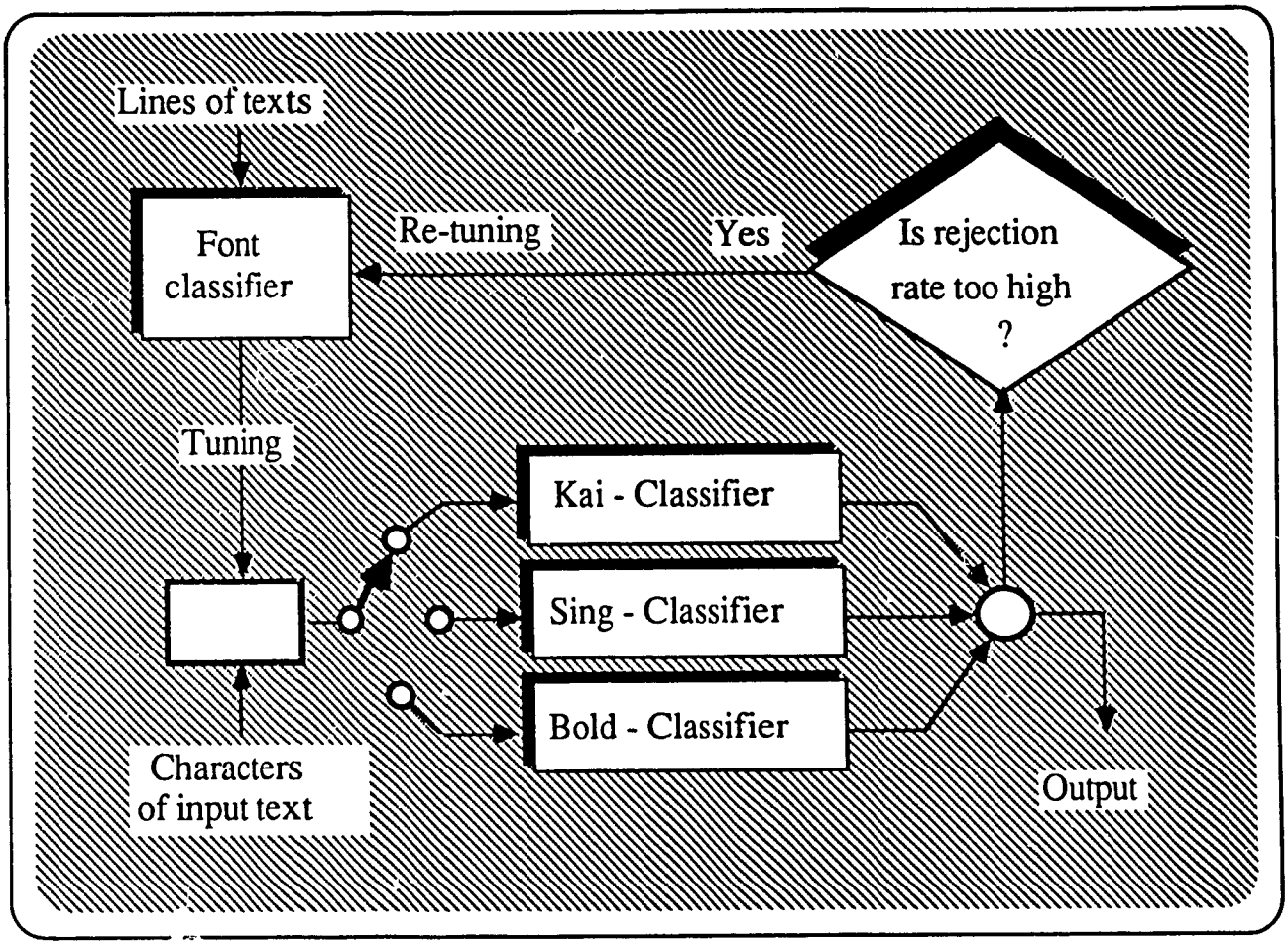


Fig. 7.4 Font Selection System

Table 7.1 Recognition Results

Tree Models		3262 Characters with Different Fonts, Sizes and Rotations	3262 Characters with Single Font, Size, without Rotation
Search I*	Error Rate	1.5 %	1.4 %
	Recognition Rate	97.9 %	98.6 %
	Speed	945 / sec.	958 / sec.
Search II*	Error Rate	0.140 %	0.113 %
	Rejection Rate	0.010 %	0.006 %
	Recognition Rate	99.85 %	99.88 %
	Speed	868 / sec.	873 / sec.

* The details of Search I and II are in reference [Wangq84].

7.3 RECOGNITION SYSTEM FOR COMPUTER VISION

This task is to design a recognition system for a distorted pattern set in a computer vision system. The video camera is the most common producer to capture the input data in computer vision system. One of the common uncertainties is shape distortion produced by the data acquisition system. These perspective projections can be regarded as nonlinear shape distortion, and under some conditions the perspective distortion can be either bilinear or bi-quadratic or bi-cubic, etc.

7.3.1 Perspective Projection Distortion in Computer vision system

An obvious example is the problem associated with the scanning of an object at an oblique angle. In the scanning system shown in Fig. 7.5, if the angle α between the camera and that of the direction of object to be scanned is 90°, it produces a standard digitized image as shown in Fig. 7.5(a), which can be easily identified by computer. However it is often a practical situation that this ideal set up condition can not be guaranteed (i.e. usually the angle α is not 90°). Consequently nonlinear distortion occurs. As a practical example, a document scanned by a camera at an oblique angle is illustrated in Fig. 7.5(b).

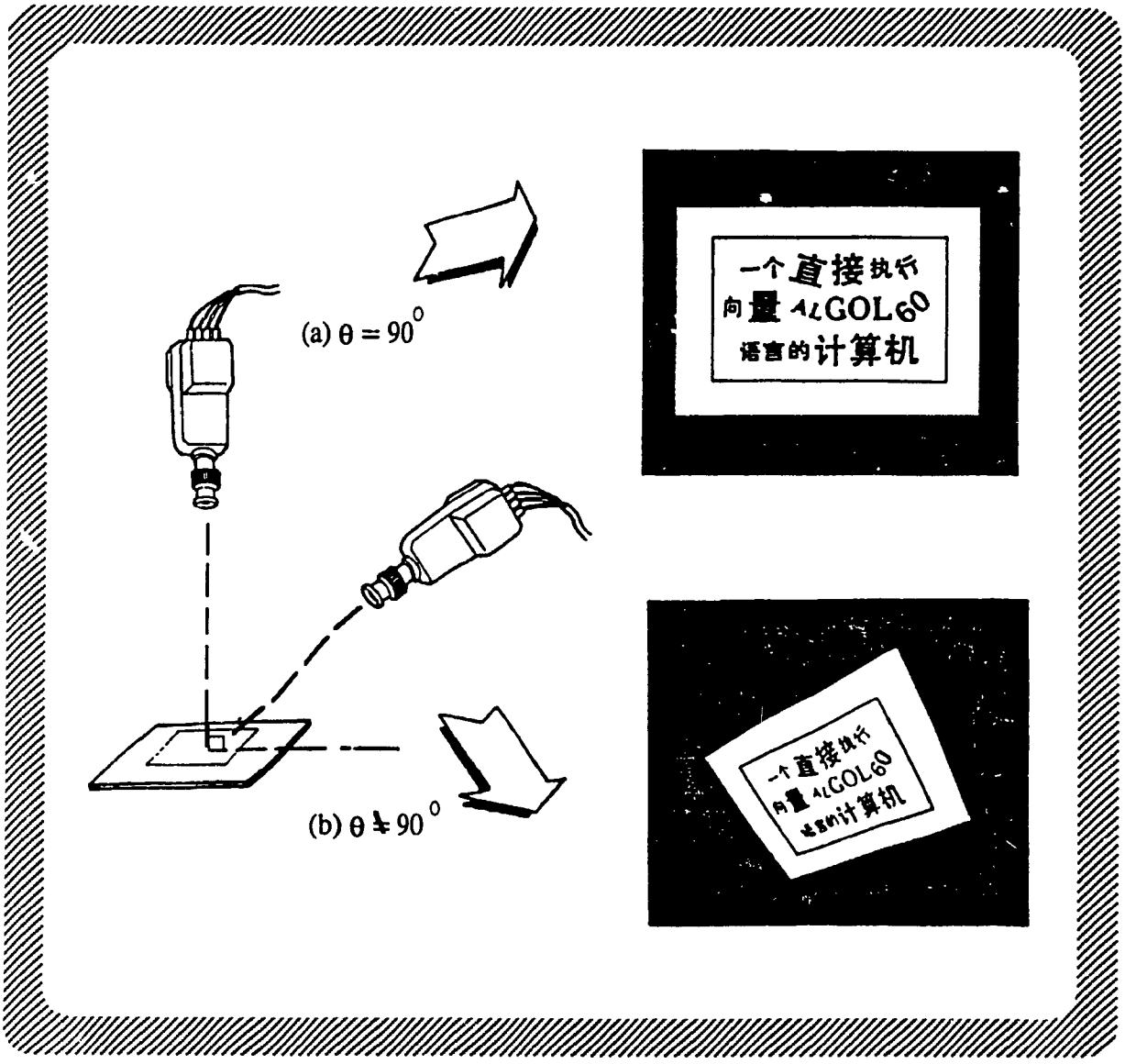


Fig. 7.5 An Example of Perspective Projection Distortion

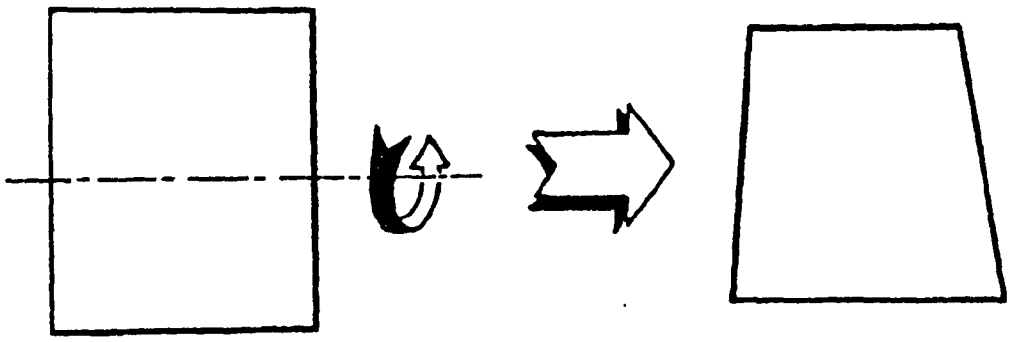


Fig. 7.6 An Example of Motion Pattern

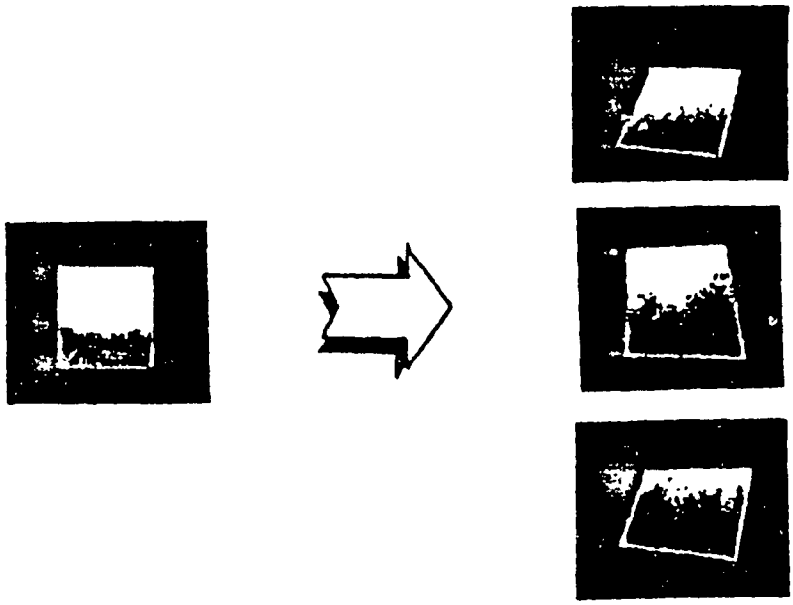


Fig. 7.7 Space-Variant Motion Degradation

Another example is problems associated with the motion patterns. When the object, for instance, a plate with square shape, revolves around an axis shown in Fig. 7.6, the image of the object is degraded, it is called space-variant motion degradation [Sawchu72]. The results are shown in Fig. 7.7. The distortions in the above examples are called perspective projection shape distortions.

7.3.1.1 Perspective Projection

Let us consider a perspective projection as shown in Fig. 7.8, we project points along projection lines that meet at the center of projection. In the case of Fig. 7.8, the center of projection is on the negative Z axis at a distance D behind the projection plane. Any position can be selected for the center of projection, but choosing a position along the Z axis simplifies the calculations.

The transformation equation for a perspective projection from the parametric equations describing the projection line from point $P(X,Y,Z)$ to the center of projection can be obtained [Hearn86]. The parametric form for the projection line in Fig. 7.8 can be expressed below:

$$\begin{aligned} \alpha &= X - X\delta \\ \beta &= Y - Y\delta \\ \gamma &= Z - (Z + D)\,\delta \end{aligned} \tag{7.1}$$

where parameter δ takes values $0 \sim 1$, and coordinates (α, β, γ) represent any position along the projection line. When $\delta = 0$, Eqs. (7.1) yield a point at the object coordinates (X, Y, Z). At the other end of the value $\delta = 1$, and Eqs.

(7.1) yield a point at the center of projection of coordinates (0, 0, $-D$). To get the coordinates on the projection plane, let $\gamma = 0$ and solve for parameter δ:

$$\delta = \frac{Z}{Z + D} \tag{7.2}$$

This value for parameter δ produces the intersection of the projection line with the projection plane at (X_P, Y_P, 0). Substituting Eq. (7.2) into Eqs. (7.1), we obtain the perspective transformation equations

$$\begin{aligned} X_P &= X \left(\frac{1}{\frac{Z}{D} + 1} \right) \\ Y_P &= Y \left(\frac{1}{\frac{Z}{D} + 1} \right) \\ Z_P &= 0 \,. \end{aligned} \tag{7.3}$$

For further discussions of transformations based on the viewing angle see references [Blinn78, Cyrus78, Foley82, Liang83, Michen80, Pavlid82, Salmon87].

7.3.1.2 Approximation of Perspective Transformation

Even though the perspective transformations are not bilinear, or bi-quadratic, or bi-cubic models, under some circumstances they can be regarded as approximations of the latter ones.

Theorem 7.1

The perspective transformation according to Eq. (7.3) is a bilinear model if

the following condition is satisfied:

$$Z \ll D \ . \tag{7.4}$$

The proof of this theorem can be found in reference [Tang90].

According to this approximation, we may tackle perspective distortion as a bilinear one, and restore it using an inverse bilinear algorithm.

Apart from bilinear shape distortion, both the bi-quadratic and bi-cubic shape distortions are encountered frequently. As shown in Fig. 7.9, the trade mark " Coke " is printed on its cylindrical surface of the Coca-cola bottle. Due to the cylindrical shape of the bottle, the trade mark has been changed from a square into a quadrangle with quadratic curve. This kind of shape distortion can be regarded as bi-quadratic distortion if Eq. (7.4) is satisfied. An example of bi-cubic shape distortion is given in Fig. 7.10. The shape of the pages in an open book have undergone distortion with cubic curve. See reference [Li89c] for details.

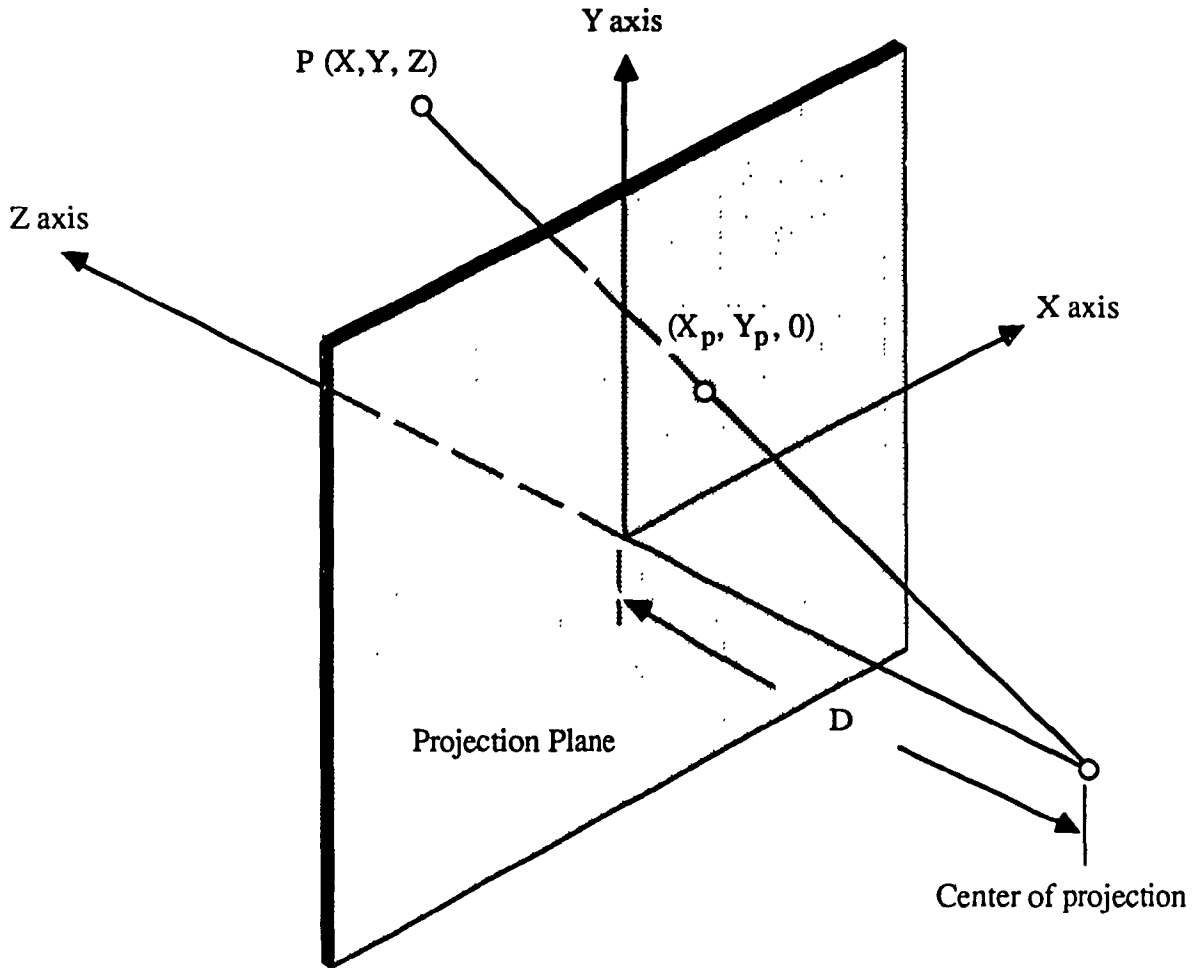


Fig. 7.8 Perspective Projection

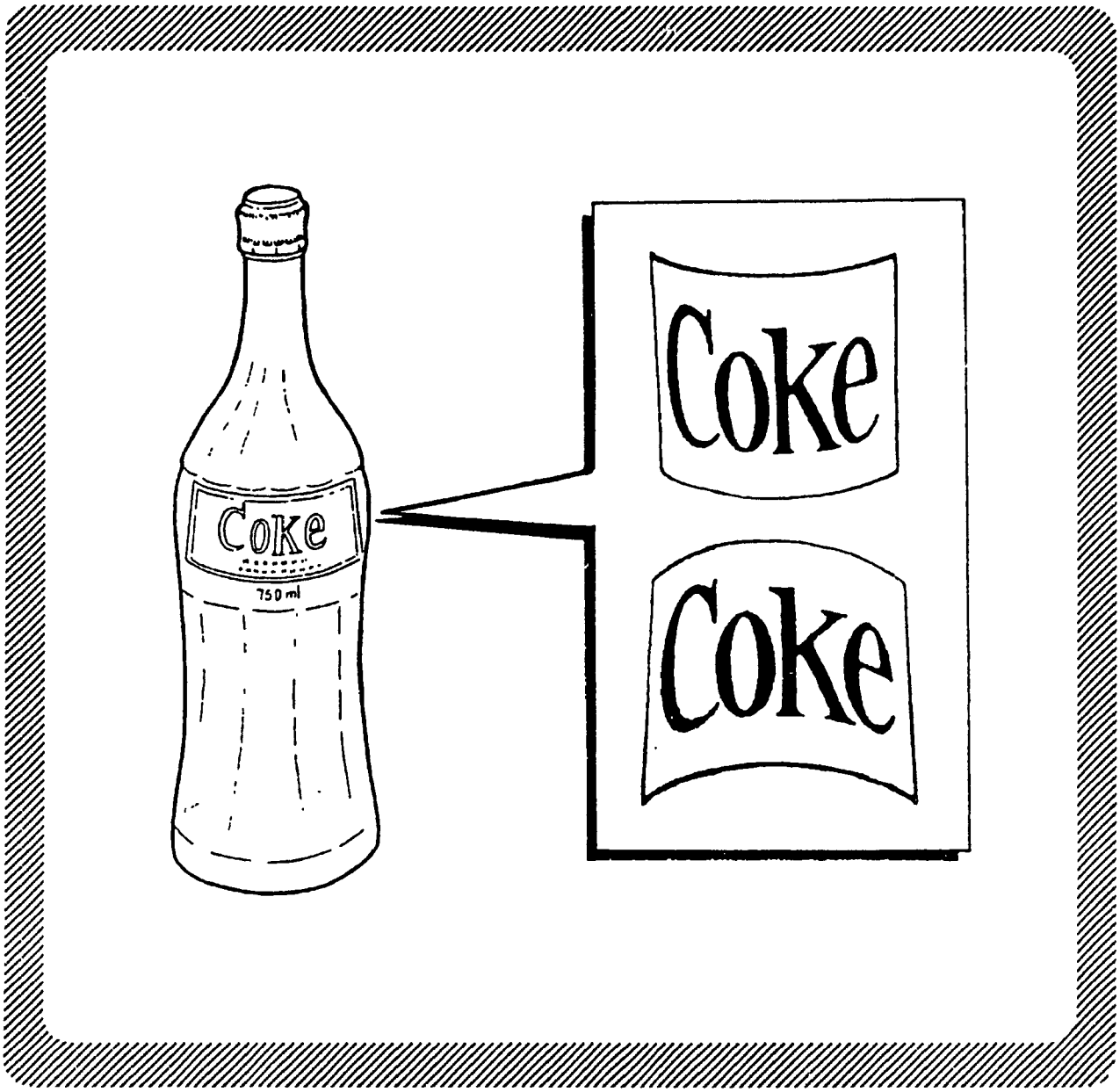


Fig. 7.9 Bi-quadratic Model Approach for Perspective Projection

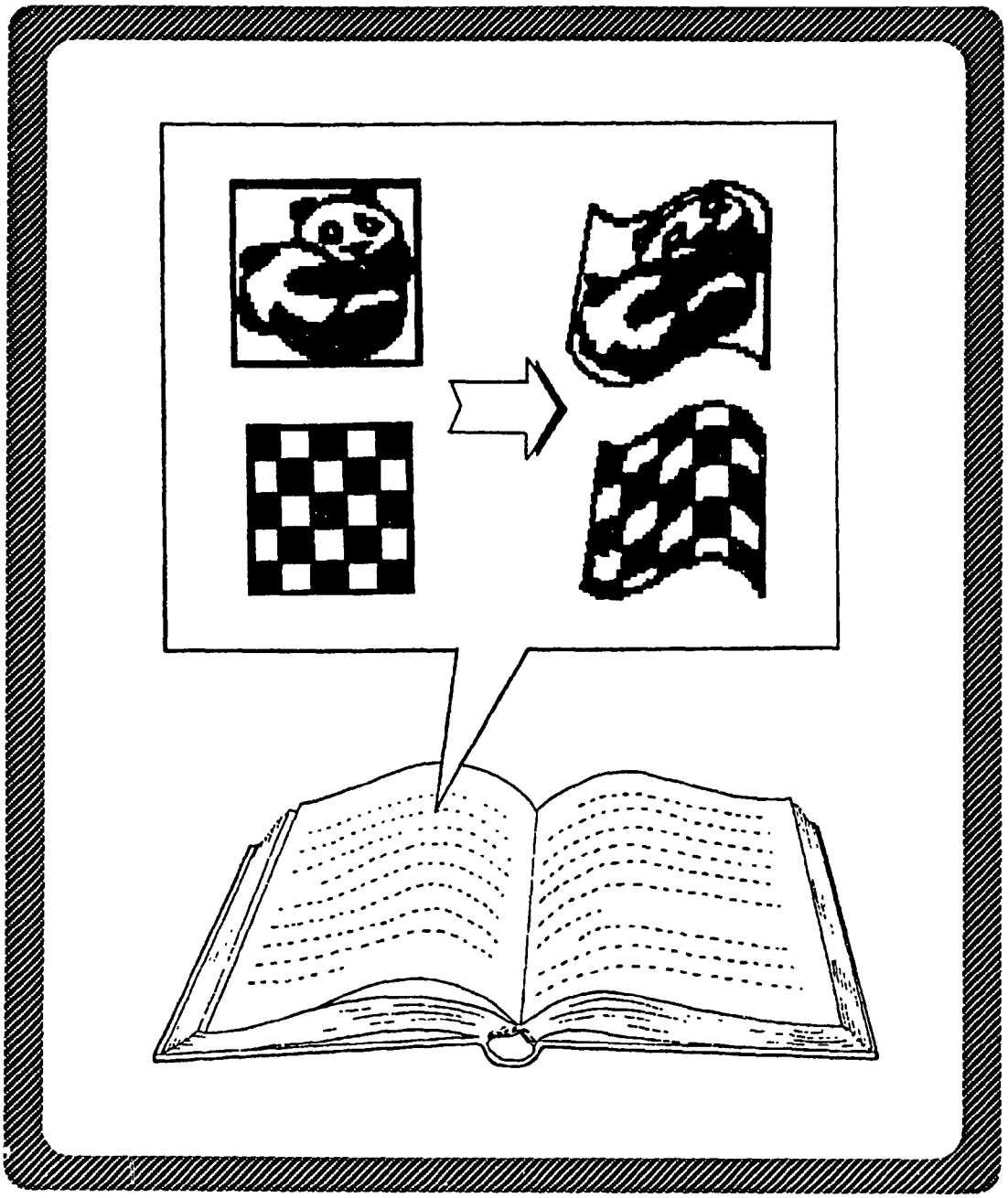


Fig. 7.10 Bi-cubic Model Approach for Perspective Projection

7.3.2 Description

As shown in Fig. 7.11, the target data set to be recognized is similar to the previous example. Additionally, a video camera data capture system CHORUS DATA SYSTEM CA-1600U has been employed in our experiment (Fig. 7.12), it is required that the angle of photograph capture has no restriction. Another assumption is that there are two kinds of physical geometric entities on which the characters appear : planar and cylindrical surfaces and the transducer distortion occurs as shown in Fig. 7.13.

7.3.3 Analysis

The uncertainty in IS1 and IS4 is as same as Section 7.2.3. Additionally, it is necessary to consider the photograph scanning distortion in IS2. As mentioned in the previous chapters the capture of a picture taken at an oblique angle can produce infinite uncertainty.

7.3.4 Design

The architecture of this system has been designed as shown in Fig. 7.14. This is a cascade system. The first cascade is the camera system F_S which captures the physical samples. The second cascade F_D is a parallel component which is used to filter out various distortions due to the camera being placed at an oblique angle in the first level of MLIS, i.e. IS1. Consequently, $F_D = F_{bilinear}\ \nabla p\ F_{bi-quadratic}\ \nabla p\ F_{pass}$. The details of the principle of $F_{bilinear}$ and $F_{bi-quadratic}$ have been presented in the above chapter. Here we only describe

the working principle of the switch S_D, which is presented as an algorithm as follows.

Algorithm 7.1

Step-1 Choose eight reference points as indicated in Fig. 7.15. The i-th point is denoted by (X_i, Y_i), i = 1, 2, ..., 8.

Step-2 Initialize the line equations :

$$L_i \ : \quad y - y_i = k_i(x - x_i),$$

where

$$k_1 = (y_5 - y_1) \,/\, (x_5 - x_1),$$
$$k_2 = (y_6 - y_2) \,/\, (x_6 - x_2),$$
$$k_3 = (y_7 - y_3) \,/\, (x_7 - x_3),$$
$$k_4 = (y_8 - y_4) \,/\, (x_8 - x_4)$$
$$k_5 = (y_2 - y_5) \,/\, (x_2 - x_5),$$
$$k_6 = (y_3 - y_6) \,/\, (x_3 - x_6),$$
$$k_7 = (y_4 - y_7) \,/\, (x_4 - x_7),$$
$$k_8 = (y_1 - y_8) \,/\, (x_1 - x_8).$$

Step-3 **if** $(k_1 = k_5)$ **and** $(k_2 = k_6)$ **and** $(k_3 = k_7)$ **and** $(k_4 = k_8)$

then if $(k_1 k_2 = -1)$ **and** $(k_3 k_4 = -1)$

then goto F_{pass}

else goto $F_{bilinear}$

else goto $F_{bi-quadratic}$

According to algorithm 7.1 the switch S_D selects automatically an appropriate

entropy-reduced filter either $F_{bilinear}$ or $F_{bi-quadratic}$ or just pass through F_{pass} doing nothing. Fig. 7.16 shows the experimental results of passing entropy-reduced filters $F_{bilinear}$ and $F_{bi-quadratic}$ respectively.

After second cascade F_D, we arrive at the position to deal with the intrinsic characteristics which appear in the first level of MLIS (IS1). This is completed by the third cascade F_I. To solve the orientation problem, the fourth cascade F_E in our system is used. This is a rotation-invariant feature extraction approach. It clusters all 360 kinds (an increment of one degree per kind) of rotations of a pattern sample into a unique reference pattern sample, which belongs to the reference set W^0. This data set has about 11.7 bits of uncertainty.

After all the above operations, we can now apply the tree-like discriminator F_C. At last we achieve the required system design as shown in the Fig. 7.14.

Fig. 7.11 An Example of Pattern Set Mixed by English Alphabets and Chinese Characters with Different Sizes and Rotations

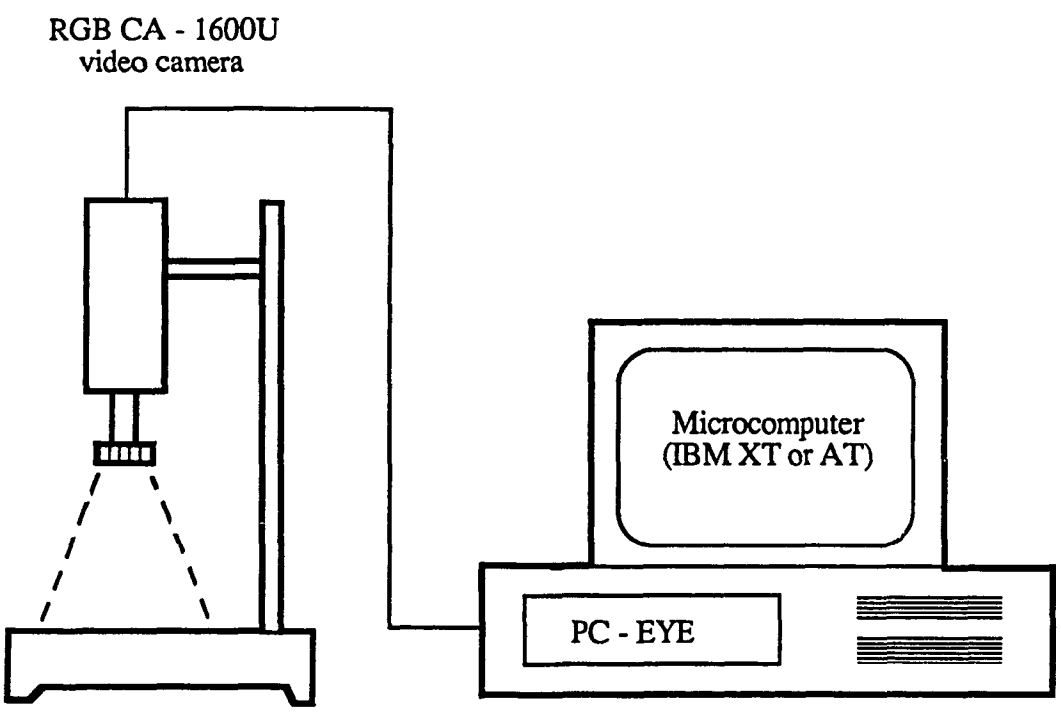


Fig. 7.12 CHORUS DATA SYSTEM CA - 1600U Photographing System

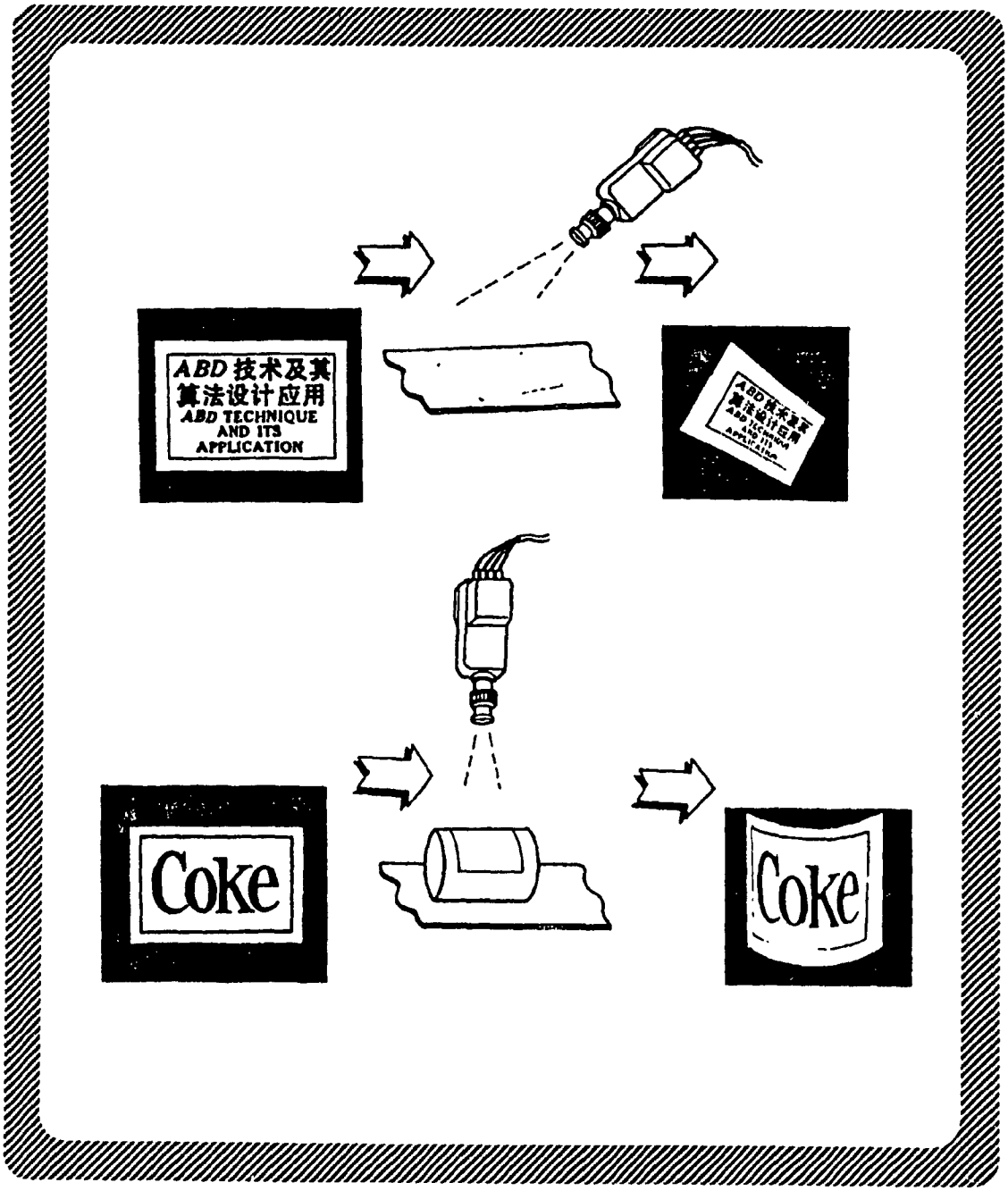


Fig. 7.13 Examples of the Transducer Distortion in Photographing System

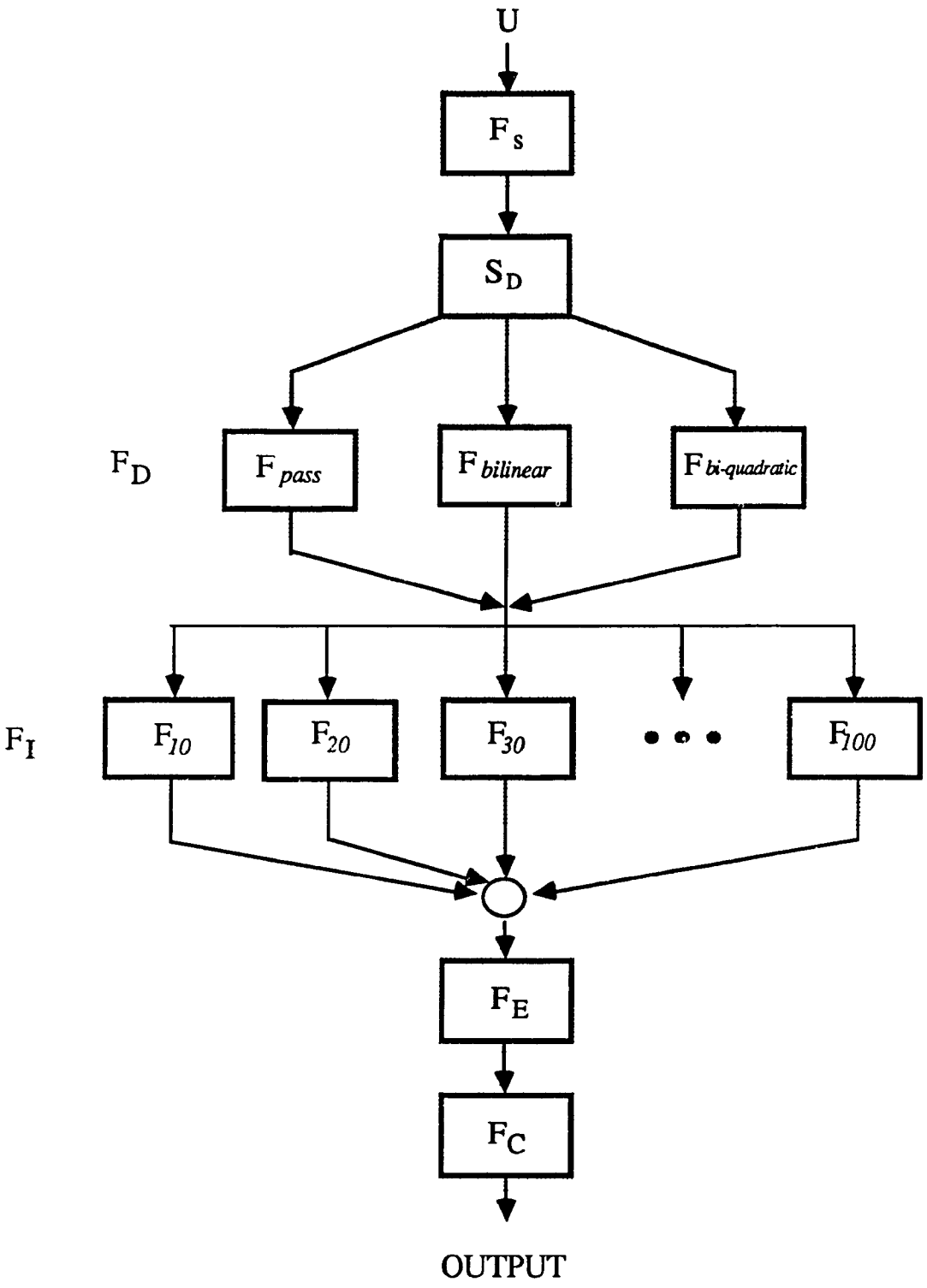


Fig. 7.14 An Architecture of Recognition System for Computer Vision

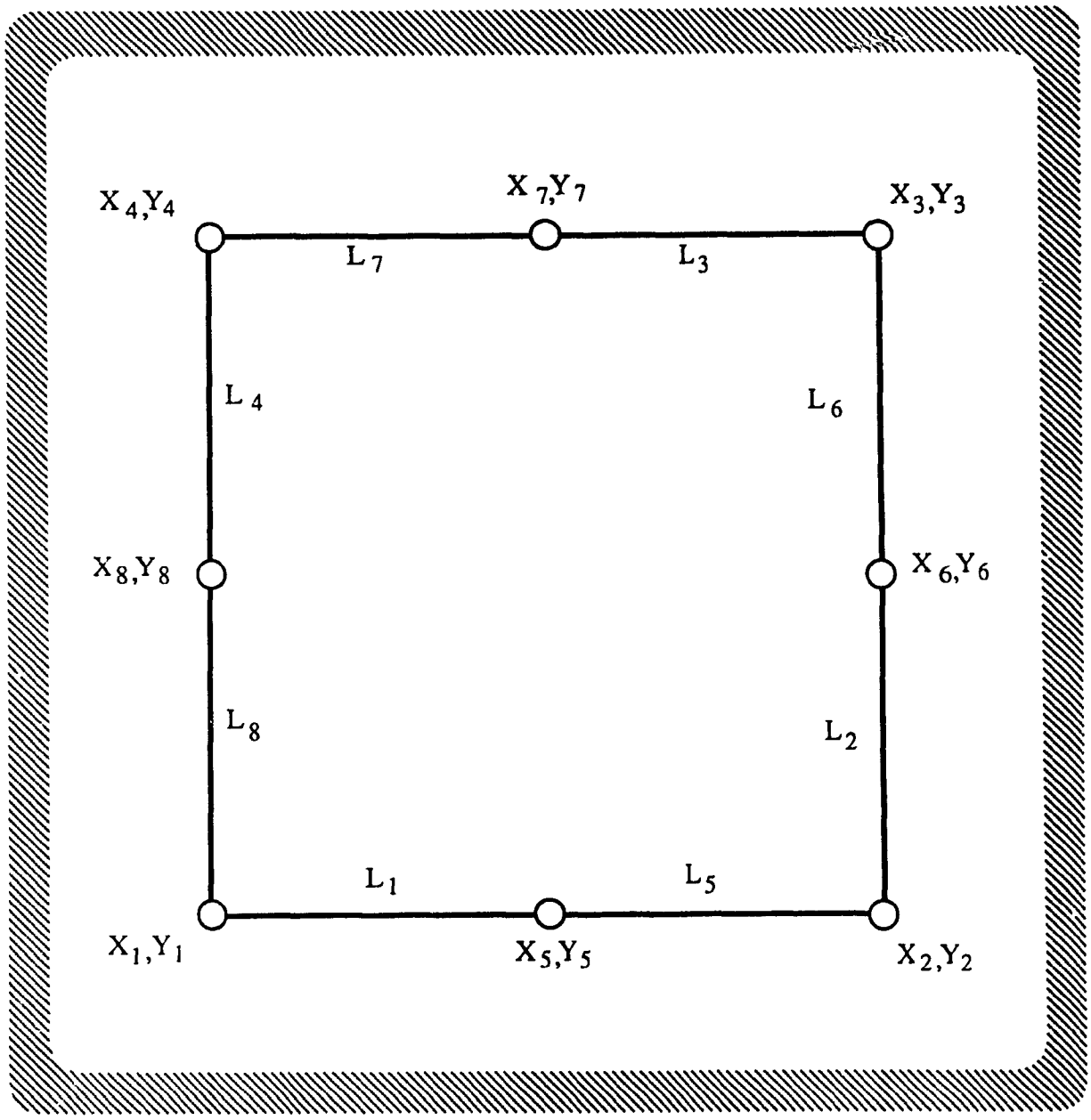


Fig. 7.15 Reference Points for Algorithm 7.1

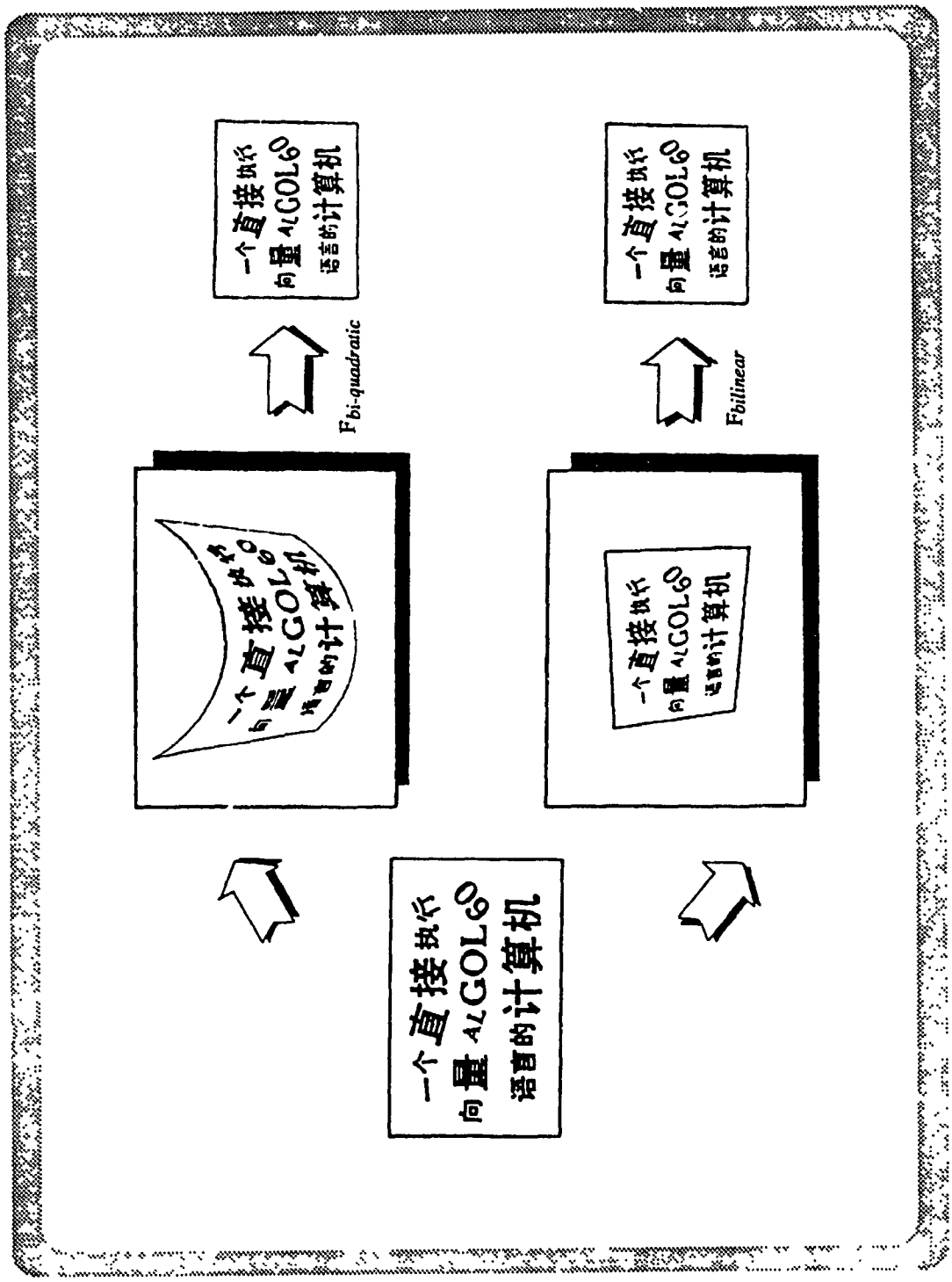


Fig. 7.16 Experimental Results of Passing Entropy-Reduced Filters $F_{bilinear}$ and $F_{bi\text{-}quadratic}$

CHAPTER 8

CONCLUSIONS

In contrast with the single-level information source model of classical information theory, MLIS allows us to address all the different factors which increase the entropy in the entire pattern recognition system. As a result, a complete and practical model for a pattern recognition system can be developed efficiently. Generally speaking, although the number of levels in an MLIS formed by a pattern recognition system is problem-oriented, the intrinsic distortions and those due to other factors should preferably be handled at different levels, because the strategies needed to tackle them are different. For example, by using some known knowledge about the relationships among the elements of the target pattern set, e.g. the size, it has been shown that the use of MLIS can reduce quite efficiently a lot of entropy due to the intrinsic characteristics. Then the orthogonal transformation theory, which is generally thought to have no direct relationship with pattern recognition, can be used to solve some severe entropy addition problems due to either the transducer or its movements.

The process of pattern recognition system can be considered as a task which transforms an entropy-increased MLIS into an entropy-reduced one. Although the methods to implement the entropy-reduced transformations at the different levels may be totally different, they can be combined by using the cascade and parallel

properties of the ERT model, a complete system can be built. It acts as a guide to provide a set of problem-oriented solutions for the design of an application-oriented system. This thesis shows that MLIS provides a systematic way to design a pattern recognition system based on the ERT approach.

In this thesis we present two examples of designing the integrated discriminator, which includes many interesting practical engineering algorithms. The experimental results support the theory developed for MLIS and ERT and indicate that they provide an efficient way to solve many difficult problems. The results also indicate that, similar to other information processing systems, entropy reduction plays a major role in every stage of a pattern recognition system. Based on this principle and the use of MLIS and ERT models we may develop other new and efficient methods to tackle more complicated problems.

Image transformation is one of the most important entropy transformations. It has two completely different characteristics: entropy increased and reduced characteristics. Both of these properties are commonly used in image processing and pattern recognition. One of which uses the entropy increased property to produce a variety of samples from a given pattern for different purposes, e.g. it can produce test samples to measure the performance of a proposed or existing image processing system, and to train and test a classification system. By using an appropriate transformation, we can produce a huge number of test samples in a very short time. This can not be done using hand drawings without a great deal of effort. The other one is the use of the entropy reduced property to per-

form image normalization which plays a major role in pattern matching.

In this thesis, some new algorithms have been proposed. These algorithms can perform the mapping and filling at the same time while preserving the connectivities of the original image. In the proposed algorithms, there is no need to distinguish the boundary and the inside pixels and to compute the direction of the boundary, resulting in not only a speed up in computation, but also a more meaningful and accurate filling process. The proposed algorithms have no difficulty in dealing with some long narrow objects. Also the order in which rotation and scaling are done has minimal effect on the transformation results. Accordingly, they can handle multi-boundary and complicated images. The principal idea of the proposed algorithms can also be extended to process three-dimensional and gray level images.

The essential parallelism of the proposed algorithms makes them easy to be implemented by using VLSI architectures. The structure of each PE has been discussed in [Cheng89, 90]. The time complexity will be O(N) for the proposed VLSI architectures and it will be $O(N^2)$ for the uniprocessor, where N is the dimension of the image. The proposed algorithms and their VLSI implementation can be applied in real-time image processing, pattern recognition and related areas.

It is necessary to emphasize that the image transformation presented in this thesis is our primary work, and that the our recent development of this topic appears in references [Li89a, b, c, 90a]. [Li89c] deals with the subject of geometric transformations of digitized images and patterns by computers to

develop systematically various shape models and discrete techniques, in particular for picture processing and pattern recognition. Transformations discussed in this book include linear, quadratic, cubic, bilinear, bi-quadratic, bi-cubic, Coons models and other nonlinear forms such as harmonic and perspective transformations. They can be used in computer graphics, character recognition, pattern recognition, computer vision and image processing. In [Li90a], a new discrete technique for the realization of nonlinear shape transformation is developed. The Splitting-Shooting method is developed to avoid superfluous holes or "measles" in the transformed images. Error analysis and graphical experiments are presented.

Variance of size and orientation is one of the common intrinsic uncertainties in the pattern recognition system. To solve this most challenging problem in pattern recognition, this thesis proposes a new method called Transformation-Ring-Projection (TRP). The basic operations have been carefully designed so that the proposed algorithm is very simple and regular. In this method, image transformation technique and the Ring-Projection scheme have been employed. As a result, the parallel processing and VLSI technology can be used to speed up the computation [Tang89b]. This work is highly significant because invariance in size and rotation will greatly facilitate pattern recognition especially for texts which are embedded in graphics. Character readers with such capabilities would find many applications in office automation, computer aided design, electronic publishing, etc.

Nonlinear shape distortions produce a considerable uncertainty in computer

vision, robot vision and recognition of motion. The correction of these distortions is one of the most difficult and challenging topics. This thesis uses the shape transformation theory [Li89c] which includes bilinear, quadratic and cubic transformations to model the nonlinear shape distortion problems. Some useful algorithms have been presented. It is worth pointing out that inverse shape transformations are powerful and useful tools which can be used in nonlinear restorations. Nevertheless, by using all existing approaches, inverse transformations are difficult to realize due to the necessity of solving nonlinear equations which often produce multiple solutions. In references [Li88b, 89c], we provide new efficient methods to carry out inverse transformations which bypass the need to solve nonlinear equations. A method called the Splitting-Integrating method is developed for this purpose. Furthermore, a combination CSIM in particular, is designed for a study of the cycle conversion $T^{-1}\,T$ of images and patterns.

REFERENCES

[Ballar82] Ballard, D. H. and C. M. Brown, *Computer Vision,* Prentice-Hall, Inc,. Englewood Cliffs, New jersey, 1982.

[Barnhi74] Barnhill, R. E. and R. F. Riesenfeld, *Computer Aided Geometric Design,* Academic Press, New York, 1974.

[Bell62] Bell, D. A., *Information Theory and its Engineering Applications,* Pitman, New York, 1962.

[Blinn78] Blinn, J. F., and M. E. Newell, "Clipping using homogeneous coordinates," SIGGRAPH '78 Proceedings, *Computer Graphics,* Vol. 11(2), pp. 245 - 251, 1978.

[Boltzm96] Boltzmann, L., *Vorlesungen über Gastheorie,* J. A. Barth, Leipzig, 1896.

[Bowen82] Bowen, B. A. and W. R. Brown, *VLSI Systems Design for Digital Signal Paocessing,* Vol. 1 of *Design for Digital Signal Processing.* Prentice Hall. N. J., 1982.

[Brady81] Brady, J. M. (Ed.), *Computer Vision.* North-Holland Publishing Company, Amsterdam, 1981.

[Burden81] Burden, R. L., J. D. Faires and A. C. Reynolds, *Numerical Analysis (Second Ed.),* Prindle, Weber & Schmidt. Boston, 1981.

[Cappel86] Cappellini, V. and R. Marconi (Eds.), *Advances in Image Processing and Pattern Recognition,* Elsevier Science Publishers, North-Holland, Amsterdam, 1986.

[Charot86] Charot, F., P. Frison and P. Quinton, "Systolic architectures for connected speech recognition," *IEEE Trans. on ASSP, ASSP - 34,* pp. 765 - 779, Aug. 1986.

[Cheng86] Cheng, H. D., and K. S. Fu, "Algorithm partition and parallel recognition of general context-free languages using fixed-size VLSI architecture," Pattern Recognition, Vol. 19, No. 5, 1986.

[Cheng89] Cheng, H. D., Y. Y. Tang, and C. Y. Suen, "VLSI Architecture for image transformation," *Proc. IEEE 1989 COMPEURO Conf., VLSI and Computer Peripherals,* pp. 2-124 - 2-126, Hamburg, May 1989.

[Cheng90] Cheng, H. D., Y. Y. Tang, and C. Y. Suen, "Parallel image transformation and its VLSI implementation," in press, *Pattern Recognition,* 1990.

[Clausi87] Clausius, R. J., *Die Mechanische Wärmetheorie,* Friedrich Vieweg and Sohn, Braunschweig, 1887.

[Cormar63] Cormarck, A. M., "Representation of a function by its line integrals, with some radiological applications," *Journal of Applied Physics* , Vol. 34, No. 9, pp. 2722-2727, Sept. 1963.

[Cormar64] Cormarck, A. M., "Representation of a function by its line integrals,

with some radiological applications II," *Journal of Applied Physics* , Vol. 35, No. 10, pp. 2908-2919, Oct. 1964.

[Critch85] Critchlow, A. J., *Introduction to Robotics,* Macmillan Publishing Company, New York, 1985.

[Cyrus78] Cyrus, M., and J. Beck, "Generalized two- and three-dimensional clipping," *Computer and Graphics,* Vol. 3(1), pp. 23 - 28, 1978.

[Davenp70] Davenport, W. B., *Probability and Random Processes,* McGraw-Hill, New York, 1970.

[Devijv82] Devijver, P. A. and J. Kittler, *Pattern Recognition: A Statistical Approach,* Prentice-Hall, Englewood Cliffs, New Jersey, 1982.

[Dobrus72] Dobrushin, R. L., "A survey of Soviet research in information theory," *IEEE Trans. Inform. Theory,* Vol. IT - 18, pp. 703 - 727, Nov. 1972.

[Doyle60] Doyle, W., "Recognition of sloppy hand-printed characters", *Proc. Western Joint Computer Conference,* Vol. 17, 133-142, May 1960.

[Eisen69] Eisen, M., *Introduction to Mathematical Probability Theory,* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1969.

[Feinst54] Feinstein, A., "A new basic theorem of information theory," *IRE Trans. Inform. Theory,* Vol. IT - 4, pp. 2 - 22, Sept. 1954.

[Foley82] Foley, J. D., and A. Van Dam, *Fundamentals of Interactive Computer Graphics,* Addison-Wesley Publishing Company, New York, 1982.

[Fu80] Fu, K. S. (Ed.), *Digital Pattern Recognition,* Springer-Verlag, Berlin,

1980.

[Fu84] Fu, K. S., *VLSI for Pattern Recognition and Image Processing,* Springer-Verlag, Berlin, Heidelberg, 1984.

[Gans69] Gans, D., *Transformations and Geometries,* Appleton-Century-Crofts, New York, 1969.

[Gibbs02] Gibbs, J. W., *Elementary Principles in Statistical Mechanics,* Yale University Press, New Haven, 1902.

[Gonzal77] Gonzalez, R. C., P. Wintz, *Digital Image Processing,* Addison-Wesley Publishing Company, New York, 1977.

[Gonzal87] Gonzalez, R. C., P. Wintz, *Digital Image Processing,* (Second Ed.), Addison-Wesley Publishing Company, New York, 1987.

[Gudese76] Gudesen, R., "Quantitative analysis of preprocessing techniques for the recognition of handprinted characters," *Pattern Recognition,* Vol. 8, 219-227, 1976.

[Guiasu68] Guiasu, S., "On the most rational algorithm of recognition," *Kybernetik,* Vol. 5, pp. 109 - 113, 1968.

[Guiasu71] Guiasu, S., "On an algorithm for recognition," In D. G. Kendell, D. Hobson and P. Tautu (Eds.), *Mathematics in The Archiological and Historical Sciences,* pp. 96 - 102, Edinburgh University Press, Edinburgh, 1971.

[Guiasu77] Guiasu, S., *Information Theory with Applications,* McGraw-Hill, New

York, 1977.

[Haber77] Haber, R. M., J. Shephorrd, R. Abel, and D. Greenberg, "A generalized graphic preprocessor for two-dimensional finite element analysis," *Computer Graphics,* Vol. 12, pp. 323-329, 1978.

[Hageme81] Hagemen, L. A. and D. M. Young, *Applied Iterative Methods,* Academic Press, New York, 1981.

[Hall82] Hall, E. L., J. B. K. Tio, C. A. Mcpherson and F. A. Sadjaki, "Measuring curved surfacesfor robot vision," *Computer IEEE,* pp. 42 - 54, Dec. 1982.

[Hansen81] Hansen, E. W., "Circular harmonic image recognition: experiments," *Applied Optics* , Vol. 20, No. 13, pp. 2266-2274, July 1981.

[Hearn86] Hearn, D. and M. P. Baker, *Computer Graphics,* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1986.

[Hsu82] Hsu, Yuan-Neng, H. H. Arsenault, and G. April, "Rotation-invariant digital pattern recognition using circular harmonic expansion," *Applied Optics* , Vol. 21, No. 22, pp. 4012-4015, Nov. 1982.

[Ingard62] Ingarden, R. S., and K. Urbanik, "Information without probability," *Colloquium Mathematicum,* Vol. 9, pp. 131 - 150, 1962.

[Jeline68] Jelinek, F., *Probabilistic Information Theory,* McGraw-hill, New York, 1968.

[Jones79] Jones, D. S., *Elementary Information Theory,* Clarendon Press,

Oxford, 1979.

[Klinch57] Klinchin, A. Y., *Mathematical Foundations of Information Theory,* Dover, New York, 1957.

[Kahan87] Kahan, S., T. Pavlidis and H. S. Baird, "On the recognition of printed characters of any font and size," *IEEE Trans. on Pattern Anal. and Machine Intell.,* Vol. 9, No. 2, pp. 274-288, March 1987.

[Landa62a] Landa, L. N., "Logical - informational algorithm for learning theory" *Psychological Journal (in Russian),* Vol. 2, pp. 19 - 40, 1962.

[Landa62b] Landa, L. N., "The characteristic selection problem in recognition systems," *IRE Trans. Infor. Theory,* Vol. IT-8, pp. 171 - 178, 1962.

[Lang87] Lang, S., *Calculus of Several Variables (Third Ed.),* Springer-Verlag, New York, 1987.

[Lee87] Lee, S. Y., S. Yalamanchili and J. K. Aggarwal, "Parallel Image normalization on a mesh connected array processor," *Pattern Recognition,* Vol. 20, No. 1, 1987.

[Lewis62] Lewis, P. M., "The characteristic selection problem in recognition system," *IRE Trans. Inform. Theory,* Vol. IT-8, pp. 171 - 178, 1962.

[Li88a] Li, Z. C., T. D. Bui, C. Y. Suen and Y. Y. Tang, "Nonlinear Transformation of digitized patterns," *Proc. Int. Conf. Pattern Recognition,* pp. 134 - 136, Rome, 1988.

[Li88b] Li, Z. C., T. D. Bui, C. Y. Suen, Y. Y. Tang and Q. L. Qu, "Splitting-

integrating methods for nonlinear images by inverse transformations," *Technical Report,* Department of Computer Science, Concordia University, 1988.

[Li89a] Li, Z. C., C. Y. Suen, T. D. Bui, Q. L. Gu and Y. Y. Tang, "Digital images and patterns of perspective transformations," *Proc. the 5th Int. Conf. on Image Analysis and Processing,* Positano, Italy, Sept. 1989 (in press).

[Li89b] Li, Z. C., C. Y. Suen, T. D. Bui, Y. Y. Tang and Q. L. Gu, "Discrete approaches for digital images and patterns of transformations," *Proc. the 5th Int. Conf. on Image Analysis and Processing,* Positano, Italy, Sept. 1989 (in press).

[Li89c] Li, Z. C., T. D. Bui, Y. Y. Tang and C. Y. Suen, *Computer Transformation of Digital Images and Patterns,* World Scientific Publishing Co. Pte, Ltd., Singapore, 1989.

[Li90a] Li, Z. C., T. D. Bui, C. Y. Suen and Y. Y. Tang, "Splitting-shooting method for nonlinear transformations of digitized pattern," in press, *IEEE Trans. Pattern Anal. and Machine Intell.*

[Li90b] Li, Z. C., Y. Y. Tang, C. Y. Suen and T. D. Bui, "Shape transformation models and their applications in pattern recognition," in press, *Int. Journal Pattern Recognition and Artificial Intelligence,* 1990.

[Liang83] Liang, Y. D., and B. A. Barsky, "An analysis and algorithm for polygon clipping," *Communications of the ACM,* Vol. 26(11), pp. 868

- 877, 1983.

[Lipkin70] Lipkin, B. S., and A. Rosenfeld, *Picture Processing and Psychopictorics* , Academic Press, New York, 1970.

[Machov80] Machover, C. and R. E. Blauth (Eds.), *The CAD / CAM Handbook,* Computer Vision Corporation, Bedford, MA, 1980.

[Martin73] Martin, H. C., and G. F. Carey, *Introduction to Finite Element Analysis, Theory and Application,* McGraw-Hill, New York, 1973.

[Maxwel46] Maxwell, E. A., *The Methods of Plane Projective Geometry Based on The Use of General Homogeneous Coordinates,* Cambridge University Press, Cambridge, 1946.

[Maxwel61] Maxwell, E. A., *General Homogeneous Coordinates in Space of Three Dimensions,* Cambridge University Press, Cambridge, 1961.

[Maxwel67] Maxwell, J. C. *Philosophical Magazine,* Vol. 19, pp. 1860, Transaction Royal Society, London, Vol. 157, pp. 49, 1867.

[McMill53] McMillan, B., "The basic theorems of information theory," *Ann. Math. Stat., Vol. 24, pp. 196 - 219, June 1953.*

[McMill62] McMillan, B. and D. Slepian, "Information theory," *Proc. IRE,* Vol. 50, pp. 1151 - 1157, May 1962.

[Merser86] Mersereau, K., and G. M. Morris, "Scale, rotation, and shift invariant image recognition," *Applied Optics* , Vol. 25, No. 14, pp. 2338-2342, July 1986.

[Michen80] Michener, J. C., and I. B. Caplbom, "Natural and efficient viewing parameters," SIGGRAPH '80 proceedings, *Computer Graphics,* Vol. 14(3), pp. 238 - 245, 1980.

[Norrie73] Norrie, D. H., and Gerard de Vries, *The Finite Element Method, Fundamentals and Applications,* Academic Press, New York, 1973.

[Nudd85] Nudd, G. R. and J. G. Nash, "Application of concurrent VLSI system to two-dimensional signal processing," In *VLSI and Modern Signal Processing,* Chapter 17, pp. 307 - 325, Prentice Hall, 1985.

[Offern85] Offern, R. T., VLSI Image Processing, William Collins Sons and Co. Ltd., 1985.

[Ortega70] Ortega, J., and W. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables,* Academic Press, New York, 1970.

[Parker85] Parker, I. N., "VLSI architecture," in *VLSI Image Processing,* R. J. Offen, editor, Chapter 3, pp. 99 - 127, 1985.

[Parzen60] Parzen, E., *Modern Probability Theory and Its Applications,* John Wiley, New York, 1960.

[Pavlid82] Pavlidis, T., Algorithms for Graphics and Image Processing , Computer Science Press, Rockville, MD, 1982.

[Prince71] Prince, M. D., *Interactive Graghics for Computer-Aided Design,* Addison-Wesley, Reading, MA, 1971.

[Psalti77] Psaltis, D., and D. Casasent, "Deformation invariant optical processors

using coordinate transformations," *Applied Optics,* Vol. 16, No. 8, pp. 2288-2292, Aug. 1977.

[Qu88] Qu, Y. Z., Y. Y. Tang and C. Y. Suen, "Entropy-reduced transformation (I) : Theoretical analysis and application in pattern recognition," *Proc. Int. Computer Science Conf.,* pp. 486 - 493, Hong Kong, Dec. 1988.

[Raisbe63] Raisbeck, G. *Information Theory an Introduction for Science and Engineers,* The MIT Press, Cambridge, 1963.

[Reicha72] Reichardt, J., *The Computer in Art,* Van Nostrand Reinhold, New York, 1972.

[Robert65] Roberts, L. B., "Homogeneous matrix representation and manipulation of N-dimensional constructs," *The Computer Display Review,* Adams Associates, May 1965.

[Rogers76] Rogers, D. F., and J. A. Adams, *Mathematical Elements for Computer Graphics,* McGraw-Hill Book Company, New York, 1976.

[Rosenf82] Rosenfeld, A., and A. C. Kak, Digital Picture Processing, Vol. 2, Second Edition, Academic Press, 1982.

[Salmon87] Salmon, R. and M. Slater, *Computer Graphics System and Concepts,* Addison-Wesley Publishing Company, Wokingham, England, 1987.

[Sawchu72] Sawchuk, A. A., "Space-variant image motion degradation and restoration," *Proc. IEEE,* pp. 854-861, July 1972.

[Shanno48a]Shannon, C. E., "A Mathematical Theory of Communication," *Bell Syst. Tech. J.,* Vol. 27, pp. 379 - 423, 1948.

[Shanno48b]Shannon, C. E., "A Mathematical Theory of Communication," *Bell Syst. Tech. J.,* Vol. 27, pp. 623 - 656, 1948.

[Siegel82] Siegel, L. S., H. J. Siegel and A. E. Feather, "Parallel processing approaches to image correlation," *IEEE Trans. on Computers,* C - 31(3), pp. 208 - 218, March 1982.

[Stark82] Stark, H. (Ed.), *Applications of Optical Fourier Transforms,* Academic Press, New York, 1982.

[Stark86] Stark, H., and J. W. Woods, *Probability, Random Processes, and Estimation Theory for Engineers,* Prentice-Hall, Englewood Cliffs, New Jersey, 1986.

[Strang73] Strang, G., and G. J. Fix, *An Analysis of the Finite Element Method,* Prentice-Hall, Englewood Cliffs, New Jersey, 1973.

[Suen80] Suen, C. Y., M. Berthod and S. Mori, "Automatic recognition of hand-printed characters - the state of the art," *Proc. IEEE,* Vol. 68, No. 4, pp. 469-487, April 1980.

[Suen86] Suen, C. Y., Y. Y. Tang and Q. R. Wang, "Feature extraction in the recognition of Chinese characters printed in different fonts," *Proc. Int. Conf. Chinese Computing,* pp. 136 - 143, Singapore, Aug. 1986.

[Tang84] Tang, Y. Y., C. Y. Suen and Q. R. Wang, "Chinese character

classification by globally trained tree classifier and Fourier descriptors of condensed pattern," *Proc. IEEE 1-st Int. Conf. On Computers and Applications,* pp. 215 - 220, Beijing, China, June 1984.

[Tang88a] Tang, Y. Y., Z. C. Li, C. Y. Suen and T. D. Bui, "Conversion of Chinese characters by transformation models," *Proc. Int. Conf. on Computer Processing of Chinese and Oriental Languages,* pp. 293 - 297, Toronto, Canada, Aug. 1988.

[Tang88b] Tang, Y. Y., H. D. Cheng and C. Y. Suen, "Size-rotation-invariant character recognition," *Proc. Int. Conf. on Computer Processing of Chinese and Oriental Languages,* pp. 161 - 165, Toronto, Canada, Aug. 1988.

[Tang88c] Tang, Y. Y., Y. Z. Qu and C. Y. Suen, "Entropy-reduced transformation approaches to pattern recognition of complex data set," *Proc. Int. Workshop on Computer Vision (IAPR),* pp. 347 - 350, Tokyo, Japan, Oct. 1988.

[Tang89a] Tang, Y. Y., Y. Z. Qu and C. Y. Suen, "Multiple-level information source and entropy-reduced transformation models : Theoretical analysis and application in pattern recognition," Technical Report, Department of Computer Science, Concordia University, 1989.

[Tang89b] Tang, Y. Y., H. D. Cheng and C. Y. Suen, "Size-rotation-invariant character recognition and its VLSI implementation" Technical Report, Department of Computer Science, Concordia University, 1989.

[Tang90] Tang, Y. Y. and C. Y. Suen, "Nonlinear shape restoration by transformation models," in press, *Proc. 10th Int, Conf. on Pattern Recognition,* Atlantic City, New Jersey, June 1990.

[Taza89] Taza, A. and C. Y. Suen, "Discrimination of planar shapes using shape matrices," *IEEE Trans. on Systems, Man, and Cybernetics, Vol. 19, No. 5, pp. 1281 - 1289, Sept. 1989.*

[Tou74] Tou, J. T. and R. C. Gonzalez, *Pattern Recognition Principles,* Addison-Wesley Publishing Company, New York, 1974.

[Tou67] Tou, J. T. and R. P. Heydorn, "Some approach to optimum feature extraction," J. T. Tou (Ed.), *Computer and Information Science - II,* Academic Press, New Youk, 1967.

[Unger59] Unger, S. H., "Pattern detection and recognition," *Proc. IRE,* 1737-1752, 1959.

[Von32] Von Neumann, J., *Mathematische grundluger der Quantenmechanik,* J. Springer, Berlin, 1932.

[Wangl79] Wang, L. X., et al., *Mathematics Handbook,* High Education Publishing House, Beijing, China, 1979.

[Wangk88] Wang, K., Y. Y. Tang and C. Y. Suen, "Multi-layer projections for the classification of similar Chinese characters," *Proc. 9th Int. Conf. on Pattern Recognition,* pp. 842 - 844, Rome, Italy, Nov. 1988.

[Wangq84] Wang, Q. R. and C. Y. Suen, "Analysis and Design of a Decision Tree

Based on Entropy Reduction and its Application to Large Character Set Recognition," *IEEE Trans. Pattern Anal. Machine Intell.,* Vol. PAMI-6, No. 4, pp. 406-417, 1984.

[Wangq85] Wang, Q. R. C. Y. Suen and Y. Y. Tang, "Application of a statistical equivalent block classifier in the recognition of Chinese characters printed in different fonts," *Proc. Int. Conf. on Chinese Computing,* pp. H-2.1 - H-2.13, San Francisco, Feb. 1985.

[Watana39]Watanabe, S., "Uber die Anwendung Thermodynamischer Begriffe auf den normalznstand des Atomkerns," *Zeitschrift für Physik,* Vol. 113, pp. 482, 1939.

[Watana64]Watanabe, S., "The Loeve-Karhunen expansion as a means of information compression for classification of continuous signals," *Technical Documentary Report to Biomedical Research Laboratory,* Denver, 1964.

[Watana67]Watanabe, S., "Karhunen-Loeve expansion and factor analysis - Theoretical remarks and applications," *Transactions of the 4-th Prague Conf. on Information Theory, Decision Functions, Random Processes,* 1965, pp. 635, Publishing House of the Czechoslovak Academy of Science, Prague, 1967.

[Watana69a]Watanabe, S., *Knowing and Guessing,* John Wiley & Sons, Inc., New York, 1969.

[Watana69b]

Watanabe, S., (Ed.) *Methodologies of Pattern Recognition,* Academic Press, New York, 1969.

[Watana81]Watanabe, S., "Pattern Recognition as a Quest for Minimum Entropy,"

Pattern Recognition, Vol. 13, No. 5, pp. 381 - 387, 1981.

[Watana85]Watanabe, S., *Pattern Recognition : Human and Mechanical,* John Wiley & Sons, Inc., Toronto, 1985.

[Watana88]Watanabe, S., and T. Kaminuma, "Recent developments of the minimum entropy algorithm," *Proc. 9th Int. Conf. on Pattern Recognition,* pp. 536 - 540, Rome, Italy, Nov. 1988.

[Wu86] Wu, R. H., and H. Stark, "Rotation and scale invariant recognition of image," *Proceedings of the 8-th International Conference on Pattern Recognition,* pp. 92-94, October 1986.

[Yarosl79] Yaroslavsky, L. R., *Digital Picture Processing: An Introduction,* Springer-Verlag, Berlin, 1979.

[Young71] Young, J. F., *Information Theory,* Butterworth & Co Ltd., London, 1971.

[Young86] Young, T. Y. and K. S. Fu (Eds.), *Handbook of Pattern Recognition and Image Processing,* Academic Press, Orlando, Florida, 1986.

[Zhou83] Zhou, J. P., *Fundamentals of Information Theory,* P. P. T. Publish-

ing House, Beijing, China, 1983.

[Zienki77] Zienkiewicz, O. C., *The Finite Element Method in Engineerinr Science* (Third Ed.) McGraw-Hill, London, 1977.