MVA'90 IAPR Workshop on Machine Vision Applications Nov. 28-30,1990, Tokyo

AN AUTOMATIC RECOGNITION SYSTEM OF ASSEMBLY DRAWINGS

Yong–Qing Cheng, Yu–Long Cao, Jing–Yu Yang

Dept. of Computer Science
East China Institute of Technology
Nanjing 210014, P.R.China

ABSTRACT

Automatic recognition of mechanical assembly drawings is a difficult problem in drawing recognition. In this paper, a heuristic recognition method "serial number–leader line–part symbol" is described to recognize assembly drawings. A tracking algorithm is proposed to recognize a variety of leader lines, find the corresponding part symbols. A tracking–matching algorithm is built to track and recognize assembly drawings in terms of a generating–testing process of primitives. Because algebra expression of the part symbol covers lots of semantic features about the part symbol, the tracking process is simultaneous with the recognition process. The experimental results show that our method is effective and isn't affected by complex degree of assembly drawings.

I. INTRODUCTION

Drawing recognition, which converts hand–drawn data into computer accessible codes, is one of the important branches in pattern recognition. Automatic assembly factories use "assembly drawings", which inform the species of parts and the interconnections among the parts, to accomplish mechanical assembly tasks. Because there are complicated connection relations of parts in mechanical assembly drawings, recognition of assembly drawings is very difficult.

So far, some methods have been developed in order to recognize equipment location maps, network diagrams, electronic circuit diagrams, chemical plant engineering drawings[1–4]. In [1], Kamei proposed a new template matching method which is very fast for binary images. In this method, the image is shifted by specified distances instead of moving the template. Among the shifted images,logical operations are performed on the entire image area. These operations can be performed very quickly with the raster operation in a workstation. In [2], Maeda developed the method of symbol recognition, according to the sequence of the following steps: a) key constituent detection; b) constituent detection; c) symbol determination; d) reclassification of remaining lines. In [3], Fahn presents a topology–based component extractor for the understanding of electronic circuit diagrams. These earlier systems don't utilize the semantic knowledge, what is expressed in the diagram. And this limits the possibility to improve recognition rate. It is not an easy task to make and modify software and symbol dictionaries, according to the types of the diagrams.

A more recent interest in drawing recognition is that of trying to design systems which embody knowledge about the basic structures of defferent kinds of graphic symbols and use this knowledge to analyze and identify the different components of a drawing. Such a system would tie together various aspects of drawing image analysis, like thinning, edge segmentation, filtering, etc. along with a high–level control structure that interprets the drawing image with the help of these image processing operations. A knowledge–based drawing recognition system involves the use of several types of knowledge including visual, spatial and linguistic. A knowledge–based system that directs the classification of the different entities on a drawing image and decides when an unambiguous classification of all the relevant entities has been achieved, is one of major goals in the field of drawing recognition. Here, an automatic recognition system ARSAD of assembly drawings is introduced, which can effectively utilize mechanical, graphic semantic knowledge to recognize assembly drawings.

A mechanical assembly drawing consists of six kinds of elements: serial numbers, leader lines, a list of parts, graphic part symbols, a outline, and character strings.

1) serial numbers: they indicate Nos. of part symbols, and are used to indicate which graphic symbols leader lines point to.

2) leader lines: they indicate the relationships between serial numbers and graphic symbols, or configuration of a outline. They are represented as two connected stragiht lines.

3) list of parts: it indicates the relationships between serial numbers and part names, quantity of parts, specification of parts, etc.

4) graphic part symbols: they indicate certain specific part symbols which stand for general mechanical parts, and are represented as fixed graphic symbols. In our ARSAD system, we can recognize all kinds of assembly drawings composed by 31 part symbols shown in Fig.1.

5) outline: it indicates the external form of the assembly drawing. There exists an unique outline in every assembly drawing.

6) character strings: they indicate comments in list of

parts.

Fig.2 shows an example of an assembly drawing to be recognized.

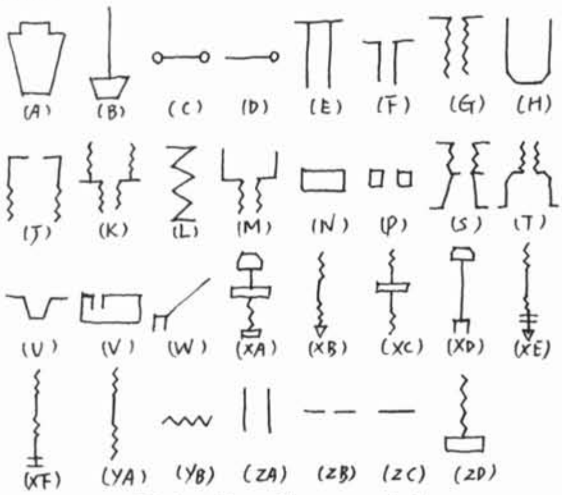

Fig.1 31 graphic part symbols

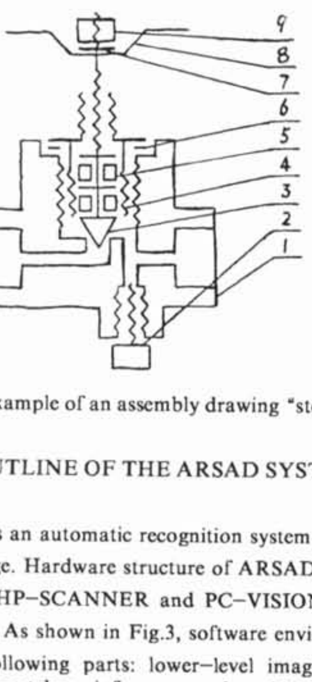

Fig.2 an example of an assembly drawing "stop valve"

II .OUTLINE OF THE ARSAD SYSTEM

ARSAD is an automatic recognition system based on domain knowledge. Hardware structure of ARSAD is composed of two parts: HP–SCANNER and PC–VISION image processing system. As shown in Fig.3, software environment consists of the following parts: lower–level image processing, segmentation, matcher, inference engine, blackboard, data base, and three knowledge sources. Assembly drawings are read by HP–SCANNER and converted to gray images. Lower–level image processing improves the gray images. Serial numbers, leader lines, part symbols are separated. Matcher is used to recognize basic primitives and part symbols. All knowlegde sources are controlled by the blackboard. Assembly drawings are represented by hierarchical model. This model consists of three parts: concept model, geometric model, physical model. An effective recognition sequence of assembly drawings is designed as follows: serial number–leader line–part symbol. First, ARSAD system recognizes serial numbers and finds the corresponding points $P_i(RE[i,1],RE[i,2])$, $i=1,2,\ldots,m$, P_i is a central coordiate of the serial number whose value is i.

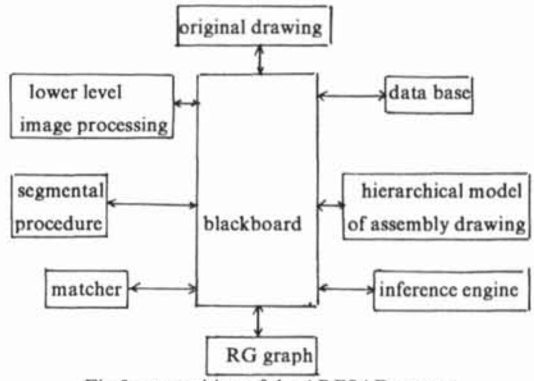

Fig.3 composition of the ARESAD system

According to these points $P_i(i=1,2,\ldots,m)$, starting points in leader lines can be found. Topology–based tracking algorithm for leader lines is used. This algorithm effectively tracks and recognizes all eight classes of leader lines and is independent of complex degree of assembly drawings; Next, matcher, which is composed of the generator–testing algorithm for basic primitives and heuristic tracking algorithm for assembly drawings, detects four categories of line segments, recognizes basic primitives and part symbols, and generates initial structure description CG–LG graph. CG is a connection graph, which describes a variety of connected relations. LG graph describes all hidden assembly relations. Inference engine, which is rule–based forward reasonning, infer their relations among part symbols, produces the corresponding relation graph RG, which can accurately describe assembly relations.

III .AUTOMATIC RECOGNITION OF ASSEMBLY DRAWINGS

1.Segmentation and recognition of leader lines

Usually, in mechanical assembly drawings, relationship among serial numbers, leader lines, and part symbols has the following features: (1) serial numbers must be written above horizontal straight lines of leader lines; (2) leader lines may be connected with a lot of graphic symbols encountered; (3) each graphic symbol must exist an unique leader line, and they are connected with each other; (4) If there exist disconnected branches in the graphic symbol, the leader line must be pointed to one of the branches; (5) there may be connected relations among graphic symbols. In order to find the true corresponding connected point between the leader line and the part symbol, the following steps are devised(Fig.4):

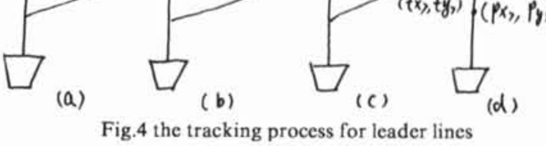

Fig.4 the tracking process for leader lines

(1) to segment the connected regions, first recognize the serial numbers, and find the central coordinate (x_i,y_i) of the i–th

serial number(Fig.4.b);

(2) according to the central point (x_i,y_i), to track down to the point LP_ion the horizontal line of the leader line(Fig.4.c);

(3) to track the i-th leader line, and find the true connected point between the i-th leader line and the corresponding part symbol(Fig.4.d).

2.Generator-testing recognition of primitives

Let p be an object pixel, we discuss eight neighbors of p as shown in Fig.5.a. When a pixel p is satisfies with one of the seven typical types as shown in Fig.5, the pixel p is called ZERO feature point. If $n_i(i=0,1, \ldots ,7)$ isn't abackground (i.e. $BV(n_i)=1$), then n_i= "true", else n_i= "false". Therefore, if the pixel p is ZERO, then it is expressed as follows:

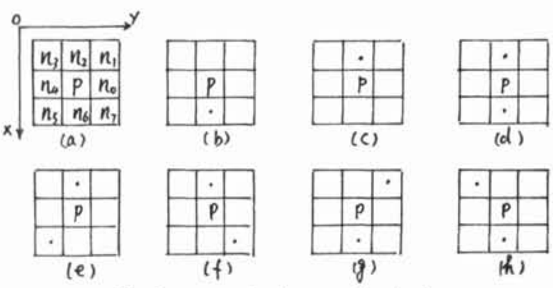

Fig.5 seven typical types of a pixel p

$$ZERO=\bar{n}_0 * \bar{n}_4 * (\bar{n}_1 * n_2 * \bar{n}_3 * \bar{n}_5 * n_6 * \bar{n}_7+\bar{n}_1 * n_2 * \bar{n}_3 * \bar{n}_5 * \bar{n}_6 * \bar{n}_7+\bar{n}_1 * \bar{n}_2 * \bar{n}_3 * \bar{n}_5 * n_6 * \bar{n}_7+\bar{n}_1 * \bar{n}_3 * \bar{n}_6 * (n_2 * \bar{n}_5 * n_7 +n_2 * n_5 * \bar{n}_7)+\bar{n}_2 * \bar{n}_5 * \bar{n}_7 * (n_6 * n_3 * \bar{n}_1+n_6 * n_1 * \bar{n}_3))$$

If ZERO is true, degree of the pixel p is called 0°. In addition, the following three boolean expressions are also gotten in the same way:

$$NT=\bar{n}_2 * \bar{n}_6 * (n_0 * \bar{n}_1 * \bar{n}_3 * n_4 * \bar{n}_5 * \bar{n}_7+n_0 * \bar{n}_1 * \bar{n}_3 * \bar{n}_4 * \bar{n}_5 * \bar{n}_7+\bar{n}_0 * \bar{n}_1 * \bar{n}_3 * n_4 * \bar{n}_5 * \bar{n}_7+\bar{n}_0 * \bar{n}_3 * \bar{n}_5 * (n_1 * n_4 * \bar{n}_7+\bar{n}_1 * n_4 * n_7)+\bar{n}_1 * \bar{n}_4 * \bar{n}_7 * (n_0 * n_3 * \bar{n}_5+n_0 * \bar{n}_3 * n_5))$$

$$FT=n_0 * \bar{n}_1 * \bar{n}_2 * n_3 * \bar{n}_4 * \bar{n}_5 * n_6 * \bar{n}_7+\bar{n}_0 * \bar{n}_1 * n_2 * \bar{n}_3 * n_4 * \bar{n}_5 * \bar{n}_6 * n_7+\bar{n}_0 * \bar{n}_1 * \bar{n}_2 * n_3 * \bar{n}_4 * \bar{n}_5 * \bar{n}_6 * n_7$$

$$OTF=\bar{n}_0 * n_1 * \bar{n}_2 * \bar{n}_3 * n_4 * \bar{n}_5 * n_6 * \bar{n}_7+n_0 * \bar{n}_1 * n_2 * \bar{n}_3 * \bar{n}_4 * n_5 * \bar{n}_6 * \bar{n}_7+\bar{n}_0 * n_1 * \bar{n}_2 * \bar{n}_3 * \bar{n}_4 * n_5 * \bar{n}_6 * \bar{n}_7$$

If NT of a pixel p is true, then the pixel p is called NT feature point and degree of the pixel p is called 90°. If FT of a pixel p is true, then the pixel p is called FT feature point and degree of the pixel p is 45°. If OTF of a pixel p is true, then the pixel p is called OTF feature point and degree of the pixel p is 135°. If ZERO or NT or FT or OTF is false, then the pixel p is set SGRY, which is a specific grayscale value.

In order to recognize primitives and track part symbols quickly, the above four boolean expressions ZERO, NT, FT, OTF are designed to describe line segments. During the tracking process of part symbols, ZERO, NT, FT, OTF are calculated. According to these ZERO, NT, FT, OTF feature points,the tracked line segments are diveded into four categories of initial primitives, and many candidite primitives are further generated. Fig.6 shows the generator-testing recognition process of primitives.

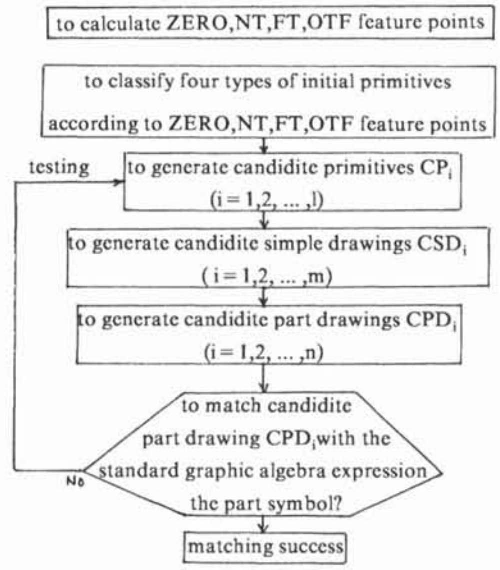

Fig.6 Generator-testing recognition of primitives

3. Recognition of assembly drawings

Each part symbol in a assembly drawing is composed of a set of picture segments which are delimited with ZERO,NT,FT,OTF feature points and properly tracked in the course of the tracking-matching process. Each picture segment is then described as a sequence of ZERO,NT,FT,OTF for primitive recognition.Because there are different complicated degrees in the part symbols, it is necessary to select a proper strategy to track the part symbols. Our experimental results show that it is more difficult to track the DG drawings than to track the others.

In this section, we will explain the heuristic recognition method of assembly drawings in detail. Let us first define the symbols as follows: Ncp(X): number of the cross points in the part symbol X; Nbp(X): number of the branch points in the part symbol X; Nep(X): number of the end points in the pact symbol X; Ndg(X): number of the DG drawings in the part symbol X; Ndp(X): number of the disconnected branches in the part symbol X; Nps(X): no. of the part symbol X.

Let PHF(X) be a heuristic function of the part symbol X, PHF is defined as:

$$PHF(X)=e_1 * Ndp(X)+e_2 * Ndg(X)+e_3 * Ncp(X)+e_4 * Nbp(X)+e_5 / (Nep(X)+1)+e_6 / Nps(X)$$

$e_i(i=1,2, \ldots ,6)$ are coefficients which can be selected according to practical requirements. The above symbols Ncp(X),Nbp(X),Nep(X),Ndg(X),Ndp(X) are automatically found by our automatical acquisition mechanism of standard part symbols. PHF function is used to calculate a complex degree for every part symbol. The complex degree is used to determine the tracking priority of the part symbol.

A heuristic recognition algorithm for assembly drawings is designed as follows:

(1) to recognize serial numbers $SN_1,SN_2, \ldots ,SN_m$in the assembly drawing;

(2) according to the recognized serial numbers $SN_1,SN_2, \ldots$

, SN_m, to segment, recognize the corresponding leader lines;

(3) to calculate the heuristic function PHF, and to obtainsorted part symbols $P_1, P_2, \dots, P_m$;

(4) for i–th part symbol P_i, to find its key point KP_i;

(5) from the key point KP_i, to track, recognize the i–th part symbol P_i, and record the connected relations and distribution regions;

(6) i: = i+1; if i< = m then goto step 4;

(7) end.

The above tracking–matching method applies a breadth–first search technique uniting a set of specific GAL algebra expressions during the traversal of an assembly drawing. Unlike other recognition methods, this method can fully utilize semantic knowledge and topological context in the GAL algebra expressions.

Ⅳ. EXPERIMENTAL RESULTS AND CONCLUSION

This section shows the actual results of automatic recognition of assembly drawings. We first tested the tracking–matching recognition algorithm, one of our first observations dealt with the algorithm's sensitivity to the choice of the key point and tracking priority. It is a very important factor to choose key point and tracking priority. For the tracking priority, the PHF function can deal with this problem better. For the part symbols in the assembly drawing, their standard GAL algebra expressions are automatically generated by our automatic acquisition mechanism.

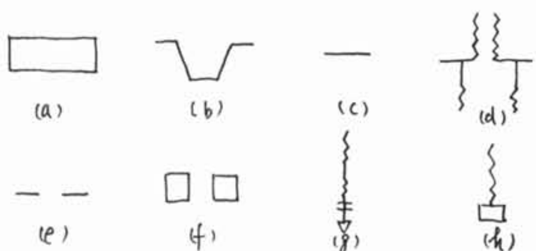

Fig.7 part symbols in "stop valve"

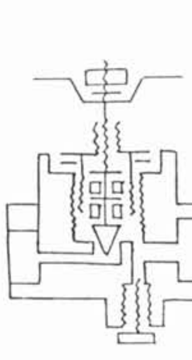

Fig.8 the assembly drawing obtained by segmentation and recognition of serial numbers and leader lines

Because the tracking–matching algorithm can fully utilize graphic, semantic knowledge in the GAL graph, it effectively deals with some ambiguous cases. The tracking process is simultaneous with the recognition process. The part symbol can be recognized as soon as the tracking–matching process is finished. Fig.2 shows an assembly drawing " stop valve" to be recognized. In Fig.7, there are recognized part symbols in the assembly drawing "stop valve". Fig.8 shows the assembly drawing obtained by segmentation and recognition of serial numbers and leader lines. The tracking order in the " stop valve" is 7,8,9,2,3,6,5,4,1.

REFERENCES

〔1〕 Katsuyuki Kamei, et al., "Symbol recognition of equipment location maps, with raster operation", Proc. of Int. Workshop on Industrial Applications of Machine Vision and Machine Intelligence, pp. 198–203, 1987.

〔2〕 Akira Maeda, et al., "A recognition method for network diagrams", proc. of Int. Workshop on Industrial Applications of Matchine Vision and Machine Intelligence, pp. 208–213, 1987.

〔3〕 Chin–Shyurng Fahn, et al., "A topolory–based component extractor for understanding electronic circuit diagrams", Computer Vision, Graphics and Image Processing, Vol. 44, No. 2, pp. 119–138, 1988.

〔4〕 H.Harada, Y.Itoh and M.Ishii, "recognition of freehand drawings in chemical plant engineering", IEEE Computer Society Workshop on CAPAIDM, pp. 146–153, 1985

〔5〕 Y.Q. Cheng and J.Y.Yang, "An expert system for understanding assembly drawings", to be presented in Int Conference "Signals & Systems", Oct. 1990.