

Automatic Acquisition of Rules for 3D Reconstruction of Construction Components

Heng Li and Ling Yu

2 Department of Building and Real Estate, The Hong Kong Polytechnic University

ABSTRACT: Aiming at the rule-based recognition system for architectural structure drawings, this paper presents a sample-based method that will automatically extract the geometric features of architectural objects and convert the features into recognition rules. When users designate the first sample of one type of objects, this method automatically analyzes and extracts the features useful for drawing recognition. Aided by a little interactive operation, recognition rules for this type of objects are generated. When a new sample of one object is designated through automatically comparing the new features and the old, this method analyzes and modifies the features of this object, then generates new rules. Along with new object types or new representations of old types are processed, rules of the recognition system are perfected continuously without modifying the program. This method can bring to the drawing recognition system higher self-adaptability and practicability.

KEYWORDS: Rule acquisition, architectural structure drawing, self-adapting recognition, construction component

1. INTRODUCTION

Although from drawing production to structural analysis, many aspects of architecture field have been computerized, some very complicated and tedious works have low automaticity. The production of bills of quantities is such kind of work. For each construction project, a great deal of time and manpower must be spent by the quantity surveying profession on reading hundreds or thousands of drawings, calculating and checking the result repeatedly. We have designed and implemented an architectural structure drawing (ASD) interpretation system (ASDIS), by which bills of quantities are conducted automatically and the work efficiency can be improved greatly.

Similar with many existing technology, our system chose to convert knowledge into recognition rules manually at first. Though this kind of static-rule-based methods can achieve good results, it also brings some drawbacks. Firstly, manual rule analysis is time-consuming and the adaptability of the recognition algorithm isn't good [2-6]. Secondly, discussions are usually limited to symbol recognition. But for those complex graphic objects, the recognition ability is lower [4-6]. Thirdly, those methods based on object feature template are too closely related with the structure definition of templates. The recognition algorithm must be modified with the change of the template structure and the flexibility is low [2,4,6].

So some researchers had proposed ideas about automatic rule learning for drawing recognition [7,8]. But only a few actually implemented the learning procedure.

One class of these approaches uses decision tree. The values of one simple primitive attribute or dualistic relation (such as inclination) are collected through strict training of many samples. These values are the only basis for choosing cut points of the decision tree [9,10]. In addition, the user is allowed to decide which branches are more important and which can be deleted [11]. Thus, the tree structures are always very complex, lacking optimized path arrangement, and long traversal time is needed for recognition. Generally, they are only suitable for very simple graphics objects, such as "two cross lines", "triangle" etc.

The other class is devised for classifying non-self-intersecting shapes that comprise straight lines only [12]. It analyzes conjunctions of local properties of each shape, indexes all the shapes by the properties and matches the indexed shapes against the instance by calculating the sum of the weights of successfully matched properties. Weights are manually selected and modified till all the samples can be recognized correctly. This approach needs a lot of samples and several different training sequences for one object type. Furthermore, it cannot adapt to objects with complex graphics (self-intersecting, or containing arcs, circles, strings) that appear frequently in architectural drawings.

To solve above problems, this paper presents a new heuristic sample-based rule acquisition method for self-adapting recognition system. This method introduces the combination of *single-sample-analysis* and *multi-sample-comparison*. Based on the integrated calculation of primitive attributes and relationships, supported by heuristic principles derived according

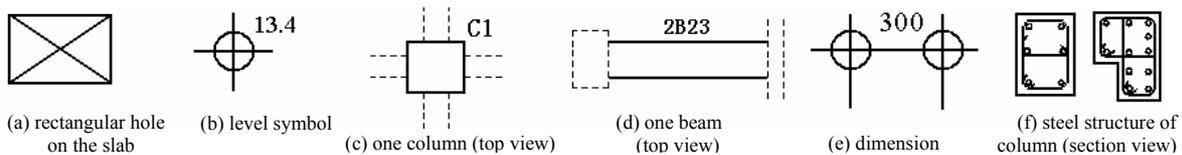
to general knowledge, feature extraction, reduction, synthesis and comparison are performed. Recognition rules are then converted from the features. Similar with human's learning procedure, while more new object representations and types are being processed, rules are perfected step by step and the recognition effect improves. This method can deal with complex objects and has no special requirements on sample amount or training sequence.

2. SELF-ADAPTING RECOGNITION MECHANISM

Graphic primitives (GP) in ASDs can be classified into two groups: lines (points, straight lines, circles and arcs) and strings. One group of GPs, which has certain architectural meaning, is called "*Construction Component*" (for short, using *object* in this paper), such as "level symbol", "column", "beam", "wall" etc.

2.1 Object Graphics and Corresponding Recognition Method

Different objects have different graphics representation characters, thus corresponding



Note: Dashed lines represent the GPs that don't belong to but are connected with the object graphics. They may not be dashed in real drawings.

Fig. 1 Examples of Architectural Objects

recognition methods are different too. Several typical examples are given as below.

- (1) **Topology matching:** Some objects' graphics have fixed topology and the topology matching is used to recognize them. Fig. 1(a) shows one rectangular hole one the slab. Its size and aspect ratio change greatly according to the reality, but it has the fixed topology: "one rectangle contains two catercorner lines intersecting each other".
- (2) **Leading GP matching:** Some objects contain one special GP fit for leading the recognition. Fig. 1(b) shows one level symbol, composed of one circle, two perpendicular lines and one string. Because the circle is usually a small part of all the GPs in a whole drawing, for improving the recognition efficiency, the circle in this symbol should be recognized firstly.
- (3) **Leading string matching:** Some object contain one string, which has particular

composition rule and so can be the lead of the recognition. Solid lines in Fig. 1(c) compose one rectangular column's boundary in the top view and dashed lines belong to the beams that are possibly connected with this column. String "C1" is column name, in which 'C' is the prefix, followed by one serial number. This is one of the naming rules of the column in ASDs in Hong Kong. String's amount is also relatively small in one drawing and content-based string matching is fast, so when recognizing such kind of objects, for improving the efficiency, search the special string to locate the object firstly, then search other lines beside the string.

- (4) **String content validation:** Some strings in objects have no regular contents, so the lines must be recognized firstly and then the existence of needed strings is examined. In Fig. 1(b), '13.4' is the string denoting the level value of the level symbol. This string must exist but the value cannot be forecasted, so it will be searched after the lines are recognized.

- (5) **Connectivity judgment:** Some objects are connected with non-object graphics by particular GPs. Such relationship must be

validated after recognizing the object graphics. Dashed lines in Fig. 1 (c) shows a kind of *flexible* relationship because the quantity and position of them are changeable. Dashed lines in Fig. 1 (d) represent two objects connected with the beam "2B23". They must appear around the beam and indicates a kind of *fixed* relationship.

- (6) **Exclusivity check:** If one object is not permitted to connect with any other objects or its graphics is a part of another object's, exclusivity check is needed. Fig. 1(e) shows one dimension. It can be looked as two level symbols, one of which there's no string. If one level symbol is found, additional examination should be done to confirm there're no another level symbol (without string) existed beside found one, viz it isn't a dimension.
- (7) **Group feature analysis:** Some objects contain GPs that can be classified into several groups. Each group embodies one kind of pattern –

quantity of GP is variable but arrangement is regular. The recognition method is the pattern matching aided by regularity checking. Fig. 1(f) shows steel bar structure in section view of two columns (strings are removed). One polygon of out layer represents the column boundary, within which the steel bars are represented by smaller polygons, short lines and small circles clung to them. This regularity then is used for recognizing such kind of objects though the quantity and position of these steel bars will change greatly with the column's size and shape.

2.2 Self-adapting Recognition Mechanism

Graphics representation characters of objects determine their recognition methods. But objects of the same type may have totally different shapes if applied drafting conventions or even only the draftsmen are different, so it's impossible to forecast what kinds of representations will appear. Therefore, predefined object templates are always inadequate and the recognition algorithm then needs to be modified manually when meeting new object representations. This situation reduces the practicability of the approaches.

representative features of T and converted into rules for recognizing T. when users designate another sample of T, the system extracts features from the new sample, then compare new features with old ones. Through analyzing the differences, the system can automatically (interactive operation can be done if needed) modifies the features and generates new rules that have better adaptability.

Along with more different object representations or types appear, the system modifies and perfects the recognition rules continuously, recognition module then can use new rules to achieve better recognition effect without modifying the program.

3. OBJECT ANALYSIS AND FEATURE EXTRACTION

Object feature, which in this paper means the geometric feature of the object, embodies a group of stable relationships among the GPs composing this object. Accordingly, recognition rules are the conditions judging whether the relations in object feature can be satisfied. Automatic feature extraction is therefore indispensable for later rule acquisition.

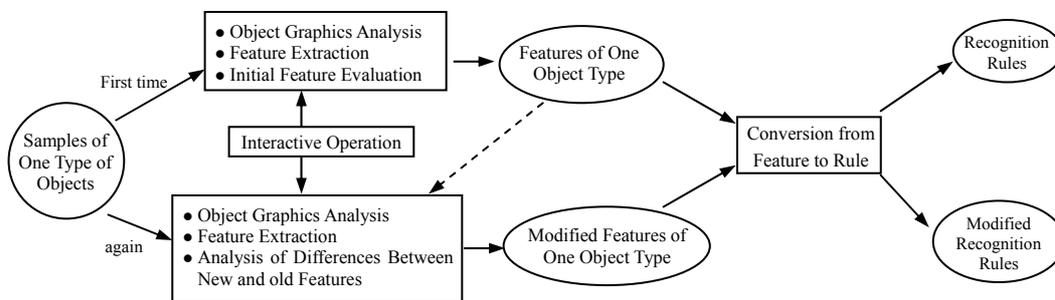


Fig. 2 Automatic Generation and modification of Object Recognition Rules

This paper presents a self-adapting architectural drawing recognition mechanism with a brand-new rule acquisition method. In this mechanism, the function of object graphics analysis and rule definition, which was manually done before, is moved into the recognition system. Fig. 2 shows the module that implements the generation and modification of the rules.

Assume we deal with the objects of type T. When users designate the first sample of T the system analyzes it to extract out the graphics features. These features are confirmed only or partially modified by users according to their experiences, then will be reserved as the

3.1 Designation of Object Graphics

Given the sample drawing of one object type, following steps are needed:

- (1) The user draws a box to designate the object graphics and give the object type.
- (2) The user designates which are *Connected Primitives (CP)*: the **lines** that don't belong to the object but are connected/intersected with the object.
- (3) The initial feature, including all the basic geometric attributes and relationships of primitives, is automatically calculated and a simple preprocess is performed:

- a) If CPs exist, then the object's initial *Connectivity* is *Fixed*, otherwise is *None*.
- b) If there're other primitives around the object except CPs, then the initial *Exclusivity* is *None*, otherwise is *All Exclusive*;

3.2 Leading Feature Analysis

Every object has one most distinctive characteristic, from which the recognition can begin to achieve faster speed and higher accuracy. So we called it the *leading feature*. The leading feature may be one of following three types:

(1) **Closed/Continuous Line Series (CLS)**, consisting of lines (arcs or a single circle) connected one by one, closed or open. Based on the initial feature data, all the lines are separated into one or several connected regions. Then in every region, all the CLSs (one primitive belongs to only one CLS) are extracted. Only four aspects of features are reserved for each CLS:

- Member primitives' types and attributes.
- Relationship of neighbor primitives.
- Relationship of non-neighbor but self-intersected primitives.
- Filling pattern of *Closed-LS*: whether and how it is filled with some primitives appearing regularly.

(2) **Object Name**: a string having special composition rules, which are related with the domain or drafting conventions and are very difficult to analyze automatically. Column name is an example, see §2.1(3). So we present a method to define the composition rule by inputting a format-control string composed of predefined special characters. For instances, '?' denotes digital numbers, '#' denotes spaces. This method is also used for analyzing other strings.

(3) **Critical Primitive (CrP)**: a special primitive except strings, whose size (e.g. line length) is the largest among the group of primitives of the same type. During recognition, if CrP is selected first, primitives smaller than it can be easily excluded.

Thereby, we analyze the leading feature as follows:

IF (*object name* was defined)
 THEN according to predefined composition rule, extract the string as **Leading String**.
 ELSE IF (*Closed-LSs* were found)

THEN classify them into several groups according to the following order: circle, polygon including arc(s), square, equilateral triangle, rectangular, ordinary polygon. The largest one in

the first group is defined as **Feature Line Series (FLS)**.

ELSE IF (*Continuous-LSs* were found)

THEN Assign higher priority to *Continuous-LSs* containing more arcs. Among those having the highest priority, the one that has the most member primitives is defined as the FLS.
 ELSE *CrP* of the whole object is **Feature Primitive (FP)**.

3.3 Object Feature Extraction

3.3.1 Definition of Key Feature Group

After further reduction of the initial feature data mainly by the principles listed below, the object feature is synthesized into six groups with descending priorities. Together they describe the key characteristics of one type of objects.

Principle 1. Relationships describing the filling pattern of *Closed-LS* should be preserved.

Principle 2. Such relationships as endpoint-connecting, vertically intersecting, parallel and concentric, should be preserved.

Principle 3. For preserved relationships, those relationships that can be derived from them are omitted.

Principle 4. other relationships between two primitives, both of which do not belong to the leading feature, are omitted.

(1) **Leading Feature(LF)**. It comprises four aspects:

- a) Leading body's type: can be *Leading String*, FLS or FP.
- b) Leading body's constraint: *Leading String*'s composition rules, FLS's feature or geometric attributes of FP.
- c) Matching constraint: describes how the leading body can be matched with the instance. Matching mode includes *Topology*, *Direct*, *Scaled* and *Rotated*. Valid scaling factor or rotation angle is also recorded.
- d) Leading mode: can be leader-line leading, position leading and distance leading.

(2) **Closure&Continuous Feature(CCF)**. If FLS is the only CLS in the object, then it is *None*, otherwise it comprises four aspects:

- a) Structural constraint: geometric structure and size of each CLS.
- b) Filling feature: *None*, or describes each CLS's filling pattern.

- c) Matching constraint: Similar to that of Leading Feature, describes how CLS is matched.
 - d) Leading primitive: each CLS's leading primitive is its CrP.
- (3) **Relationship Feature(RF)**. It includes the object primitives and the relationships except those already recorded in the former two groups.
- a) Relation mode: leader-line relation, distance relation, position relation, filling relation.
 - b) Relation constraint: detailed geometric relationships for a certain mode.
- (4) **Transformation Feature(TF)**. It is the matching constraint of the whole object graphics.
- (5) **Exclusivity Feature(EF)**.
- a) Exclusivity type: *None* (exclude nothing), *All Exclusive* (exclude everything), *Primitive Exclusive* (exclude some certain primitives).
 - b) Exclusivity constraint: only for the last type, includes attributes of excluded primitives and relationship between object and them.
- (6) **Connectivity Feature(ConF)**.
- a) Connectivity type: *None*, *Fixed* and *Flexible*.
 - b) Connectivity constraint: CPs' position and type for *Fixed* type, permitted quantity and attributes of CPs for *Flexible* type.

3.3.2 Single-sample-based Initial Feature Evaluation

Features that are extracted from only one sample may not be integrated enough, but the initial values can be set automatically by estimation according to the general knowledge. For an instance, system will select "rotated matching [0-360d] plus direct matching" as the default value of TF, but "topology matching" as that of the matching constraint in CCF. Other examples include: position relation is the default value of relation mode; thresholds in all kinds of relationships are required to be "exactly equal" initially; etc.

Through automatic evaluation like above, obtained features may have some limitations. Users can modify manually according to their experiences to make the features more adaptable, or designate more samples to let the system modify the features automatically.

3.4 Multi-sample-based feature synthesis and modification

Generally, new samples are obtained by three ways:

- Manually selecting new samples in advance.
- During recognition, manually selecting wrong/missing recognized object as new samples.
- Automatically processing the object that is being modified interactively.

After processing a new sample of an object type whose features have been extracted before, the system compares the old features with the new ones and automatically modifies (also reminds the users to check) the object features according to the differences. Some primary principles are listed below:

- (1) If the old exclusivity type is "*All Exclusive*" but the new one is "*None Exclusive*", then the latter one will be chosen.
- (2) If the old and new connectivity types are different, or both types are *Fixed* but CPs are different in type/amount, then the connectivity type is changed to "*Flexible*".
- (3) With the change of positional relationship between string and line, relevant relation mode can be changed from "position relation" to "distance relation", and the distance constraint value is changed. Similar changes apply to other constraints.
- (4) If the old *CCF* indicates one Closed-LS is *Direct Matching*, but in the new sample this Closed-LSz's size is changed but the shape unchanged, then its matching mode will be changed to *Scaled Matching*.
- (5) If the primitive composition of the new sample is the subset of the old, or vice versa, then only those features that are related with this subset of primitives are preserved.
- (6) If the new features are totally different from the old ones, both of them are preserved as coordinate features. For example, some CLSs appear in old samples but not in the new one, or the geometric structure is totally changed (Fig. 3). Then the common part (if exist) is extracted automatically and assigned with highest priority.



Fig. 3 Examples of Feature Modification – Coordinate Features

4. AUTOMATIC GENERATION OF RECOGNITION RULES

The recognition rules of each object type will be automatically generated or modified according to its current feature and recorded in the file with a predefined format.

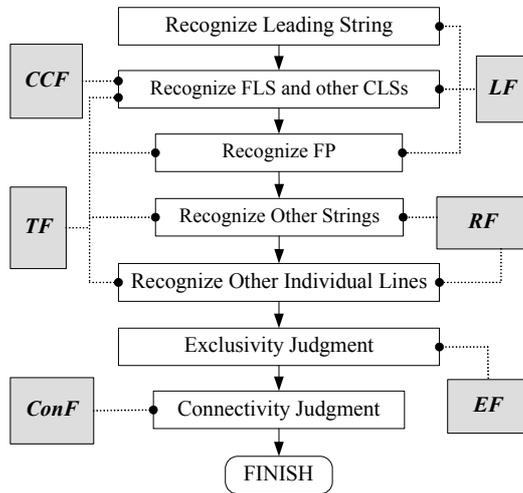


Fig. 4 Recognition Phases and Key Feature Group

4.1 Typical Object Recognition Process

Fig. 4 gives the typical recognition process and shows which group of features should be used for which recognition phase. If some features are lacked, corresponding phase can be omitted. That is, the recognition of one object depends on a group of ordered rules, each of which can be converted from some correlative object features.

4.2 Conversion from Features to Rules

Following are some examples.

- (1) Assume that leading type is *Leading String* whose composition rule is predefined as S (for example, S can be “?B?#{(?X?)}”), leading mode is down-lead leading: ‘LL’, then recognition rule of *Leading String* is:


```

      IF MATCH_LEAD_STRING(S, 'LL')
      THEN RETURN(Names,
      Leading-lines)
      ELSE FAIL
      
```
- (2) Assume CCF indicates: one CLS’s

structural constraint is described as the model (‘L’); filling feature is null (‘NF’); leading primitive is the longest line (*a*); matching constraint is *Direct Matching* (‘DM’). *TF* indicates “rotated matching” and permitted rotation angle is from 0 to 360: ‘T:R(0-360)’, then following rule is generated for recognizing this CLS:

```

  IF MATCH_LINES('L', 'NF', a,
  'DM', 'T:R(0-360)')
  THEN RETURN(Polygon)
  ELSE FAIL
  
```

- (3) Assume FP is the longest line: *a*; *RF* shows: there’s one string related with *a*, the relating mode is distance relating ‘DL’, relation constraint indicates that the string is parallel (‘P’) with *a* and the distance between them is less than *d*, then following rule is generated:

```

  IF MATCH_STRING(a, 'P;DL:≤d')
  THEN RETURN(String)
  ELSE FAIL
  
```

- (4) Assume the exclusivity type is *All Exclusive*: ‘ALL’, then generated rule is:


```

      IF (TEST_EXCLUDE('ALL') ≠
      NULL)
      THEN FAIL
      
```

5. EXPERIMENTAL RESULT

Our method was experimented in ASDIS. Fig. 6(a) shows a corner of one architectural framing plan. It contains 7 types of objects including grid symbol, column, beam, slab-mark, hole, level-mark and section-mark. In the initial file, processing only the samples shown in Fig. 5 (1)-(7) generated rules for these 7 object types. After the first round of recognition, Fig.5 (8)-(11) were chosen as new samples and processed using our method. As shown in Fig. 6 (b)-(c), the recognition effect using the initial database is not good, but becomes much better when the modified database is used.

For giving more detailed information, we take level-mark symbol as an example. Fig.7 shows 3 samples and Table 1 then gives the content of the symbol features after extraction and modification through processing these samples one by one .

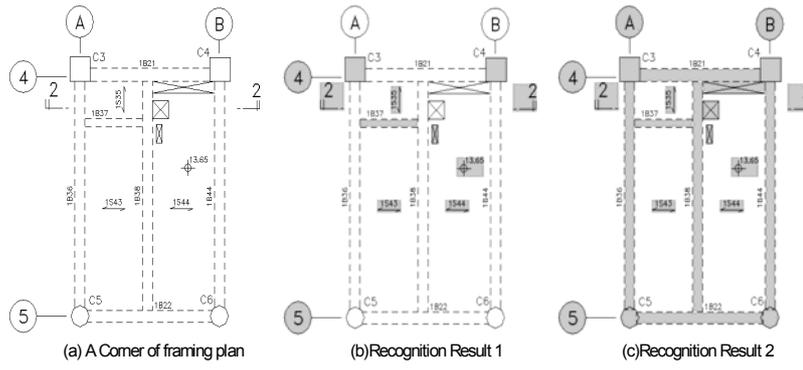


Fig. 6 Recognition effect

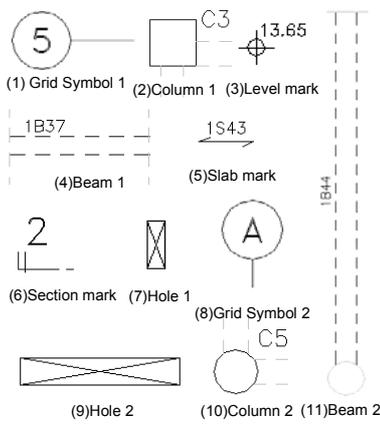


Fig. 5 Object samples

Table 1. Feature Analysis and Modification of Level-mark Symbol

	after processing Fig.5(a)	after processing Fig.5(b)	after processing Fig.5(c)
LF	FLS=Circle[1], radius[150,150], Direct +Rotated[0-360d], position leading.	FLS=Circle[1], radius[150,170], Direct +Rotated[0-360d], position+distance.	FLS=Circle[1], radius[110,170], Direct +Rotated[0-360d], position+distance.
RF	 Primitives: Relationships: 6 sets of relationships that can be described using above graphics	 String[1]+Line[2] Circle-String-Distance[0,radius*2.1] L1-ThroughCircle[at Circle Center] L1-L2-Vertical&Intersect[on middle segment of L1, on midpoint of L2]	 String[1]+Line[2] Circle-String-Distance[0,radius*2.4] L1(2)-ThroughCircle[midpoint of L1(2) is within the circle]

Unchanged features after processing Fig.5 (a): CCF is None, TF is Rotated[0-360d], EF is All Exclusive, and ConF is None.

6. CONCLUSION

We present a rule acquisition method that can adapt to complex object graphics, brings higher applicability and self-adaptability to architectural drawing recognition system and is very useful in improving the productivity of the quantity surveying work. This method performs the heuristic feature induction from single sample and stepwise perfecting of rules through multi-sample-based feature comparison and

modification. It need not be trained with a large number of samples and different sequences before the objects can be recognized. Our future research directions include investigating how to achieve feature reduction with higher flexibility.

7. REFERENCES

- [1] Yang RuoYu et al. "Concrete Reinforcement Quantity Survey System Based on Graphics Understanding", *Journal of Computer-Aided Design & Computer Graphics*, 2000, Vol.12, No.9 : 688-694
- [2] Prabhu B.S., Pande S.S., "Intelligent interpretation of CADD drawings", *Computers & Graphics* 1999(23): 25-44
- [3] Rangachar Kasturi, Sing T. Bow et al. A System for Interpretation of Line Drawings, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Oct. 1990, 12(10): 978-992
- [4] Ah-Soon Christian, "A Constraint Network for Symbol Detection in Architectural Drawings", *Graphics Recognition--Algorithms and Systems*, volume 1389 of *Lecture Notes in Computer Science*, 1998.4: 80-90
- [5] Josep Lladós et al. "A String Based Method to Recognize Symbols and Structural Textures in Architectural Plans.", *Graphics Recognition--Algorithms and Systems*, volume 1389 of *Lecture Notes in Computer Science*, 1998.4: 91-103
- [6] Yu Yuhong, Ashok Samal, "A System for Recognizing a Large Class of Engineering Drawings", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, August 1997, 19(8): 868-890
- [7] Samet, H.; Soffer, A. A legend-driven geographic symbol recognition system. In *Proceedings of 12th IAPR*, Vol.2, 1994 : 350-355
- [8] Satoh, S.; Mo, H.; Sakauchi, M. Drawing image understanding system with capability of rule learning. In *Proceedings of 2nd ICDAR*, 1993 : 119 -124
- [9] Lu, W.; Wu, W.; Sakauchi, M. A drawing recognition system with rule acquisition ability. In *Proceedings of 3rd ICDAR*, Vol.1, 1995 : 512 -515
- [10] Satoh, S.; Satou, T.; Sakauchi, M. One method of structural description rule extraction based on graphical and spatial relations. In *Proceedings of 11th IAPR International Conference on Pattern Recognition Methodology and Systems*, 1992: 281 -284
- [11] Kefalea, E. et al. An integrated object representation for recognition and grasping. In *Proceedings of 3rd International Conference on Knowledge-Based Intelligent Information Engineering Systems*, 1999 : 423 -426
- [12] Kyugon Cho and Stanley M. Dunn. Shape-based Object Recognition by Inductive Learning. In *Proceedings of ICPR(B)* : 681-684