J. S. Gero and F. Sudweeks (eds), Artificial Intelligence in Design '98, 169-187.

Shape Pattern Recognition Using a Computable Pattern Representation

Myung Yeol Cha and John S. Gero
Key Centre of Design Computing
Department of Architectural and Design Science
University of Sydney, NSW 2006, Australia

Abstract. Properties of shapes and shape patterns are investigated in order to represent shape pattern knowledge for supporting shape pattern recognition. It is based on the notion that shape patterns are classified in terms of similarity of spatial relationships as well as physical properties. Methods for shape pattern recognition are explained and examples from an implementation are presented.

1. Introduction

Ever since humans have made artefacts, they have used design knowledge as design sources for new designs. This knowledge has been learned or abstracted from existing objects both natural and artificial using the human@s cognitive processes. Among design knowledge, shape pattern knowledge tends to be fundamental for aesthetic design. Recognition of this shape pattern knowledge will provide potentially important sources for design computation as formative ideas. Computer recognition of shape patterns has difficulties due to variations in shapes and shape properties. From a set of different shape elements, it is difficult to identify similar shapes and categorise them into a class. Applications of structural shape pattern representation can solve these difficulties. For example, drawings in Figure 1 are recognised differently in terms of their physical properties. The shape in Figure 1(a) is composed of ovals or curved lines, the shape in Figure 1(b) is composed of triangles or squares. Both shapes do not have any shared similarities in terms of physical properties for classification. However, a structural pattern recognition system can identify their congruency and classify them using their structural similarity. These two shapes can be described as 90^{o} rotations of four shape elements, thus they share the same shape pattern and can be categorised into a class. Structural shape pattern representation provides invariant knowledge for a pattern recognition system and makes the system more widely applicable.

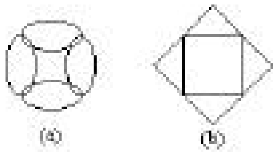


Figure 1. Two shapes are different in terms of their physical properties, but they share the same shape pattern that is a 90^{o} rotation of shapes.

The objective of this paper is to lay the foundations which can give the computer the ability to recognise shape patterns from existing drawings. A computable shape pattern representation is presented and methods for shape pattern recognition system hybridised with structural shape pattern representation are explained in Section 2. Then an implemented shape pattern recognition system is presented in Section 3.

2. Shape Pattern Representation

Shape knowledge in design disciplines has been represented variously in different contexts. Stiny (1975, 1976, 1980) developed shape grammars to generate shape designs using generative rules. Architectural morphology has been developed to describe syntactic arrangements of all possible plans using graph theory (Steadman, 1983). Computational models have been presented for shapes and shape semantic emergence (Gero, 1992; Gero and Yan, 1994; Gero and Jun, 1995; 1998). Rosenman (1995) introduced an edge vector representation for the construction of two-dimensional shapes. Also many methods of shape representations had been developed. Based on those shape representations and representation models, a computable structural shape pattern representation has been developed using knowledge representation methods from Artificial Intelligence (Cha and Gero, 1997). This representation is employed to support shape pattern recognition. Its focus is the representation of shape relationships as well as physical shape properties. Important features of shape pattern representation recognition are presented in this section.

According to Uhr (1973), pattern recognition is a cognitive process that identifies certain things using a many-to-one mapping from the set of all the variant instances. Some common properties that are necessary and useful are critical to identify similarity between objects. Many works in pattern recognition have produced hand-written letters, spoken speech and simple pictures successfully using template matching, feature matching and structural matching processes. In design, Gross and Do (Gross, 1994; Gross and Do, 1995) developed a system that can recognise, interpret and manage hand-drawn diagrams and sketches using shape analogy. Most pattern recognition systems depend on template, feature and first-level structural matching. Higher level patterns that are not recognised from physical properties but recognised from relationships or low-level patterns, are needed to be investigated to make them explicit for supporting design computation. In this section, methods of recognising structural patterns as well as feature patterns are presented.

2.1. PATTERNS

From pre-history, humans have decorated fabrics, pottery vessels, tools and buildings with patterns. Repeated usage of the same materials in decoration gives artefacts patterns (Rowland, 1964; Justema, 1976). The craftsmen@s understandings of materials have created interesting patterns. Natural orders surrounding us provide ways of creating patterns. Repeated similar relationships and organisations as well as materials and motifs create patterns. Relationships and organisations can loosely be referred to as patterns. Groups of patterns that are similar relationships and organisations as well as repetition of materials and motifs could be repeated as a design. These repeated groups of patterns may be organised in certain ways and specify higher patterns. Furthermore, sets of high-level patterns can create more high-level patterns recursively. Then, they are organised in the form of hierarchical tree structure. Patterns that are similar relationships and organisations, or repetitions of materials and motifs are regarded as low-level patterns in the hierarchical tree structure and working as units to specify high-level patterns.

Patterns have many characteristics in design. They often encapsulate design knowledge that appears in the design. Patterns, generalised from a set of objects, belong to a class. Related patterns can be composed in a hierarchical form and layered. Patterns expressed in the form of a hierarchical tree structure have variability at their lower levels. The application of invariant high-level patterns can generate many possible results under various contexts. Patterns encapsulate blocks of design knowledge. In complex objects, there may be many independent specific sets of sub-elements that are repeated and arranged in certain ways. Patterns are representations of these small blocks and clearly identify synthesis of primitives. They help to understand and interpret the design. Patterns as entities are recognised and work as parts for a larger whole. Lower level patterns that are hierarchically formed have variability, thus they can be turned into variables. In pattern recognition, a matching process that identifies the relationships among patterns disregards micro-patterns. This variability remains within the borders of higher-level constraints.

Patterns that are good solutions for certain problems and contexts can be applied elsewhere. The flexible application of one pattern for different problems and situations can occur through variable instantiations and analogy. Instantiation of pattern variables as well as physical element variables generates design objects that belong to a class specified by the high-level patterns. Patterns represent abstract formal knowledge for objects, thus it is possible to transfer patterns from one design domain to another.

2.2. SHAPE REPRESENTATION AND RECOGNITION

Shape pattern recognition starts from identification of shape congruency. According to Rowland (1964), a pattern is a design in which a certain shape is repeated many times. That is to say, a set of congruent shapes arranged in a certain way specifies a pattern. For the specification of shape patterns, congruency of shapes needs to be identified. Congruency between shapes is identified in terms of structural and physical properties. A shape is regarded as a unit in shape pattern recognition. Two shapes are considered as congruent shapes, if and only if structures of elements in one shape are equivalent to structural properties in another shape in terms of topology and geometry (Gero and Jun, 1997). For comparing structural properties, shapes are represented as bounded polyline shapes.

A bounded polyline shape is an enclosed polyline shape, for any point on the boundary of which there exists at least one circuit composed of line segments which starts from and ends at that point without covering any line segment more than once. A bounded polyline shape P is represented with a list of vertex coordinates with order and direction:

$P = \{(x_1, y_1), (x_2, y_2), @ , (x_n, y_n)\}$ (1)

An architectural drawing O is a set of bounded polyline shapes and represented as follows:

$O = \{P_1, P_2, @ , P_n\}$ (2)

A shape has the first and last vertex in its representation and is ordered anti-clockwise. Vertex order and direction are important in shape congruency recognition, even though two congruent shapes may have the different first vertices. All the first vertices of congruent shapes should be synchronised for congruency identification. The first vertices of different shapes can not be synchronised because they have different structures. The congruency between shapes is identified in terms of structural properties, such as the numbers of line segments, angles on vertices and ratios of two consecutive line segments.

The order of vertices is useful to identify reflected congruent shapes. The two shapes in Figure 2(a) are congruent and reflected. For identification of shape congruency, the first vertex of the second shape is moved to be synchronised with the first shape, Figure 2(b). Their directions of vertices are anti-clockwise in Figure 2(c), thus comparing structural properties for two shapes in terms of vertex order cannot confirm their congruency. But if the vertex order of the reflected shape is reversed, their structural properties could be equivalent, Figure 2(d).

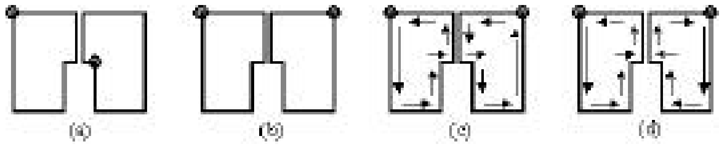


Figure 2. (a) Two reflected shapes with different first vertices, (b) the first vertex in the second shape is synchronised with the first shape, (c) the same direction of vertices in the reflected shape creates difficulties in identification of congruency between the reflected shapes and (d) two shapes have the same first vertices, directions of vertices and structural properties, but reflected.

All congruent shapes in a shape object are identified from all primitive shapes, and then congruent shapes are grouped together and specify a group shape. A shape object composed of group shapes is represented as follows:

$G_i = \{P_{i,1}, P_{i,2}, @ , P_{i,n}\}$ (3)

Where P: bounded polyline shape

G: group shape

There are many different shapes in terms of shape properties: subshapes, primitive shapes, group shapes. Subshapes are parts embedded in a shape and they may have spatial relationships between each other (Stiny, 1980). Primitive shapes are shapes that cannot be divided into subshapes. Group shapes are sets of shapes that are grouped together visually in terms of their properties and relationships, such as, proximity, similarity, closure, good continuation and symmetry (Arnheim, 1954; Kohler, 1930; Wertheimer, 1945). In addition, a set of group shapes is grouped together and specifies a shape object such as an architectural drawing. It is represented as follows:

$O = \{G_1, G_2, @ , G_n\}$ (4)

Where O: shape object

A group shape for constructing a high-level shape object is regarded as a unit or primitive so that its low-level properties are not important. Congruency between group shapes is considered for specifying high-level relationships and shapes. Congruency between these group shapes can be identified by comparing spatial relationships. These spatial relationships between two congruent shapes are described in the following section.

2.3. SHAPE RELATIONSHIP REPRESENTATION AND RECOGNITION

Spatial relationships can be specified from a set of shapes and a set of low-level relationships. A set of congruent or similar shapes organised in certain ways can be grouped together and specify relationships. Low-level relationships identified from a set of congruent shapes are clustered in terms of their congruency and identify high-level relationships. Furthermore, a set of similar relationships either constructed from different shape groups or different relationship groups can specify more high-level relationships recursively.

At first, spatial relationships are identified from a set of grouped congruent shapes, particularly isometric transformation relationships are identified. Isometric transformations are closed transformations that transform one shape into another shape without losing any properties. Isometric transformation relationships are the most fundamental spatial relationships upon which all shape representations, such as, topology, shape semantics and patterns, can be founded. These are relationships between congruent shapes. There are four kinds of isometric transformations: translation, reflection, rotation and scaling. They are described here in terms of the notation to be used later in the paper.

Translation : In a translation relationship denoted by τ_1, a shape element e_2 can be described with respect to the shape element e_1 using arguments a_1 and a_3, where a_1 is a translation distance along the axis a_3. The translation axis a_3 can be any free lines, either straight lines or curve lines.

$e_2 = \tau_1 \{e_1, (a_1, a_3)\}$ (5)

Congruent shapes, that are parallel, are considered to have a translation relationship as shown in Figure 3(a).

Rotation: In a rotation relationship denoted by τ_2, a shape element e_2 can be represented with respect to a shape element e_1 with a rotation angle a_2 and rotation centre a_5.

$e_2 = \tau_2\{e_1, (a_2, a_5)\}$ (6)

It is recognised by finding a rotation centre on the centreline (*cl*) that passes through a centre point of any two sychronised vertices and perpendicular to the line l which joins these two vertices. The rotation centre a_5 can be found by comparing two angles θ_1 and θ_2. If two angles θ_1 and θ_2 are equal and the intersection of two lines l_1 and l_2 is on the centerline (*cl*), this intersection is a rotation centre a_5 and the angle θ is a rotation angle a_2, Figure 3(b).

Reflection: The reflected shape element e_2 can be described with respect to a shape element e_1 with a reflection relationship τ_3 and a reflection axis a_3.

$e_2 = \tau_3 \{e_1, a_3\}$ (7)

It is identified from two sets of synchronised vertices. An axis that passes through a centre of two vertices v_1 and v_1@, and a centre of two vertices v_2 and v_2@ is the reflection axis a_3, Figure 3(c).

Scaling: The scale transformation represented by τ_4 changes the size of a shape e_1 by a scale factor a_4.

$e_2 = \tau_4\{e_1, a_4\}$ (8)

It is identified by comparing any synchronised line lengths from two shapes, Figure 3(d).

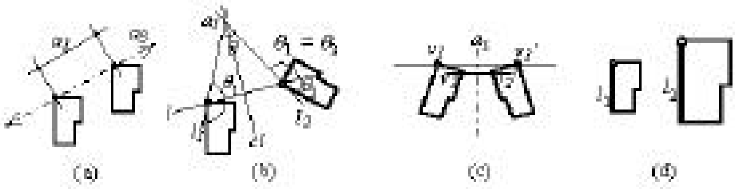


Figure 3. Isometric transformation relationship representation and recognition: (a) translation; (b) rotation; (c) reflection; (d) scaling.

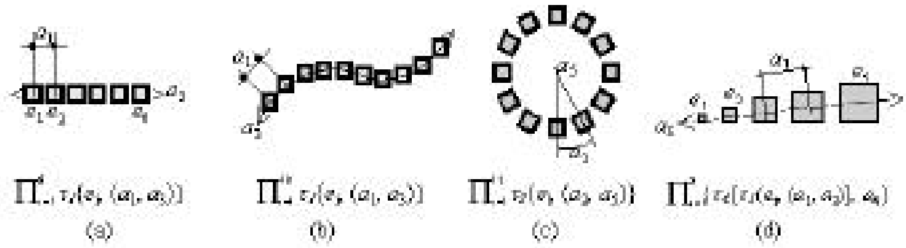
So far, well-known relationships between congruent shapes and their recognition methods have been described. In addition, sets of congruent relationships from congruent shapes can be grouped to specify higher level patterns. These shape pattern representations from congruent shape groups and their recognition methods are presented in the next section.

2.4. SHAPE PATTERN REPRESENTATION AND RECOGNITION

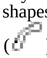
A group of shapes that are congruent or similar may be arranged in a pattern and one shape can be explained with respect to another shape recursively, then this group shape can be described using an isometric transformation relationship representation and a nesting operator ($\prod_{i=1}^{n}$). The nesting operator applies a transformation factor to elements recursively until all shapes in a pattern are described by another shape. It provides a description of how the given pattern can be constructed from the primitive shapes. Characteristics of low level elements are not included in the shape pattern representation. Thus shape patterns will be invariant regardless of their different element properties. The general representation of a relationship between two shapes is $e_i = \tau \{e_{i-1}, a\}$, and a shape pattern is an arrangement of a congruent shape group $\{e_1, e_2, @ , e_n\}$. The pattern in this shape group can be described with a nesting operator as:

$S = \prod_{i=1}^{n} \{e_i, a\}$ (9)

Some examples of shape pattern representations are shown in Figure 4. In addition, the pattern representation in Figure 4(d) has a complex pattern representation, two transformation relationships, a translation and a scaling relationship, are applied at the same time.



A set of patterns may be similar and arranged in a regular form. These patterns can specify higher patterns. Shape patterns can be constructed not only from shape congruency but also from congruency of spatial relationships or patterns.

According to Rips (1989), there are two main features to determine similarity: surface similarity and deep similarity (or structural similarity). Surface similarity is based on shape object attributes and structural similarity is similarity at the level of relational structure. Similarities in shapes are identified by attributes and physical structure (Gero and Jun, 1995), continuous transformations (March and Steadman, 1971; Mitchell, 1990; Steadman, 1983), or organising structure (Falkenhainer et al., 1989/90). Congruent shapes are identified in terms of structural properties between two shapes. Similarity of shape patterns can be specified in terms of structural similarity between two relationships or patterns. Even though their physical properties are different, they are regarded as similar shapes called analog shapes. Four shape groups in Figure 5 are different shapes in terms of their physical properties. The shape group in Figure 5(a) is composed of congruent four shapes () and the other shape groups in Figure 5(b), 5(c) and 5(d) are made up of different shapes. However, their patterns identified by a spatial relationship are congruent. All four shape patterns are 90^{o} rotations of four shape elements through the centre a_5. Thus they can be considered as congruent shape patterns in terms of their spatial relationships. The properties of the physical shape elements are disregarded.

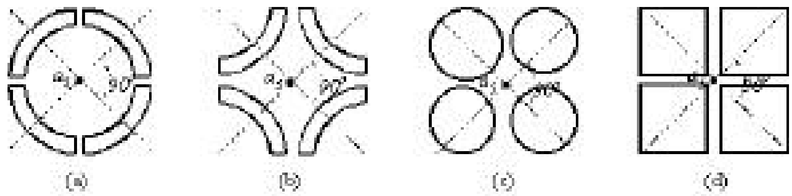
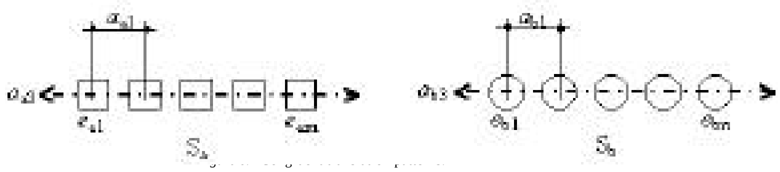


Figure 5. Four shapes have different physical properties, but considered as congruent shape patterns in terms of their structural similarity.

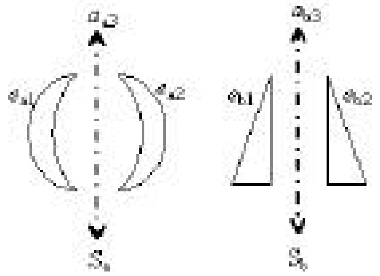
Based on structural similarity, similarity between shape patterns can be recognised from shape pattern representations. There are essentially two basic paradigms to classify a pattern into one of several different categories or pattern classes: statistical pattern recognition and structural pattern recognition (Jain, 1987). In the statistical approach, patterns are classified in terms of established decision boundaries in feature spaces. The choice of features and the form of the decision boundary are crucial to the performance of the recognition system. In complex patterns, the selection of features and establishment of decision boundary is very complicated and there are many invariant features in different complex objects that cannot be directly recognised using the statistical approach as surface similarity can not identify structural properties that are critical in the shape pattern recognition. A pattern could be viewed as being composed of simple subpatterns. A large set of complex patterns can be described by a small number of primitives, subpatterns and compositional rules using the structural approach. Thus the structural pattern recognition approach is mostly employed for shape pattern recognition. In shape pattern representation and recognition, a shape pattern is represented with symbols and numbers. Symbols classify objects into different structural domains and numbers specify dimensional spaces. At first, shape patterns are classified in terms of their structural properties, then a low-level classification can be used in dimensional spaces. Using the structural pattern recognition, a pattern recognition system for isometric transformation representation has been developed.

Each isometric transformation is recognised and classified in terms of its structural symbol described as τ . From a set of patterns which are the same isometric transformations, further classification is possible using dimensional properties.

Translation pattern: The translation pattern is represented using a translation distance a_1 and a translation axis a_3. The two shape patterns S_a and S_b in Figure 6 are translation patterns of squares and circles, and represented as $S_a = \prod_{i=1}^{m}{}_a \{e_{ai}, (a_{a1}, a_{a3})\}$ and $S_b = \prod_{i=1}^{n}{}_b \{e_{bi}, (a_{b1}, a_{b3})\}$. The properties of their low-level elements are different. A congruent translation pattern (S_a S_b) exists between two shape patterns S_a and S_b, if and only if two shape pattern descriptions have the same predicates ($\tau_a = \tau_b$), type of axes ($a_{a3} = a_{b3}$), translation distances ($a_{a1} = a_{b1}$) between sub-elements and the same number of sub-elements ($m = n$), Figure 6.



Reflection pattern: The reflection pattern is represented using the reflection axis a_3. The two shape patterns S_a and S_b in Figure 7 are reflection patterns of shapes and represented as $S_a = \prod_{i=1}^{m}{}_a \{e_{ai}, a_{a3}\}$ and $S_b = \prod_{i=1}^{n}{}_b \{e_{bi}, a_{b3}\}$. A congruent reflection pattern exists (S_a S_b) between two shape patterns S_a and S_b, if and only if two shape pattern descriptions have the same predicates ($\tau_a = \tau_b$) and the same type of axes ($a_{a3} = a_{b3}$), Figure 7.



Rotation pattern: The rotation pattern is represented using the rotation centre a_5 and the rotation angle a_2. The two shape patterns S_a and S_b in Figure 8 are rotation patterns of shapes and represented as $S_a = \prod_{i=1}^{m} \tau_a \{e_{ai}, (a_{a2}, a_{a5})\}$ and $S_b = \prod_{i=1}^{n} \tau_b \{e_{bi}, (a_{b2}, a_{b5})\}$. A congruent rotation pattern ($S_a \cong S_b$) exists between two shape patterns S_a and S_b, if and only if two shape pattern descriptions have the same predicates ($\tau_a = \tau_b$), the same number of sub-elements ($m = n$) and the same rotation angles ($a_{a2} = a_{b2}$), Figure 8.

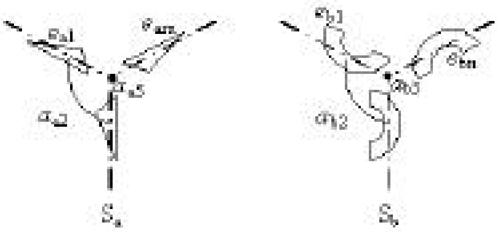


Figure 8. Congruent rotation patterns.

Similarity: Similarity between complex shape pattern descriptions represented in the form of a hierarchical tree structure can be identified by comparing the highest predicates and arguments, then moving down to lower predicates and arguments. If two shape pattern descriptions have the same predicates and arguments, they can be considered as similar shape patterns in terms of relationships, even though they have different lower predicates and arguments. Their lower shapes and patterns can be generalised.

Shapes in Figure 9 are composed of different subshapes and low level relationships, even though they share the same high-level patterns. Circles are arranged in a pattern, Figure 9(a) top, and L-shapes are arranged in a certain way, Figure 9(b) top. Three circles in Figure 9(a) specify a scale pattern in which sizes of circles are increased by the scale factor a_4 and this pattern is represented as $\prod_{i=1}^{3} \tau_4 \{e_i, a_4\}$. These four congruent scale patterns are rotated through the rotation centre a_5 by the rotation angle a_2 and the rotation pattern is represented as $\prod_{j=1}^{4} \tau_2 \{x_j, (a_2, a_5)\}$ where x_j are low-level elements ($\prod_{i=1}^{3} \tau_4 \{e_i, a_4\}$). Then three congruent rotation patterns are translated along the axis a_3 with the distance a_1 and the result is represented as $\prod_{k=1}^{3} \tau_1 \{x_k, (a_1, a_3)\}$. This shape pattern is in the form of hierarchical tree structure. In addition, the two L-shapes in Figure 9(b) are reflected around the reflection axis a_3 and this reflection pattern is represented as $\prod_{i=1}^{3} \tau_3 \{e_i, a_3\}$. These four reflected patterns are congruent and rotated through the rotation centre a_5 by the rotation angle a_2 and the rotation pattern is represented as $\prod_{j=1}^{4} \tau_2 \{x_j, (a_2, a_5)\}$. The three congruent rotated patterns are arranged on the axis a_3 with the translation distance a_1 ($\prod_{k=1}^{3} \tau_1 \{x_k, (a_1, a_3)\}$).

If we compare the high-level patterns, a congruent shape pattern can be identified even though their low-level shapes and patterns are different. The highest pattern in Figure 9(a) is a translation of three shape patterns. Also, the pattern in Figure 9(b) is a translation of three shape patterns. Thus, these two shape groups share the same highest shape pattern that is a translation of three shape pattern elements and represented as $\prod_{k=1}^{3} \tau_1 \{x_k, (a_1, a_3)\}$. The translated three shape patterns in these two congruent shape patterns are composed of four low-level shape patterns and specify a rotation pattern. They are congruent again and represented as $\prod_{j=1}^{4} \tau_2 \{x_j, (a_2, a_5)\}$, but their lowest shape patterns and shapes are different, the lowest shape patterns in Figure 9(a) are scale patterns of three circles ($\prod_{i=1}^{3} \tau_4 \{e_i, a_4\}$) and the lowest shape patterns in Figure 9(b) are reflection patterns of L-shapes ($\prod_{i=1}^{3} \tau_3 \{e_i, a_3\}$).

Even though each shape pattern has different low-level patterns and shapes, they can be considered as similar shape patterns in terms of their high-level pattern congruency.

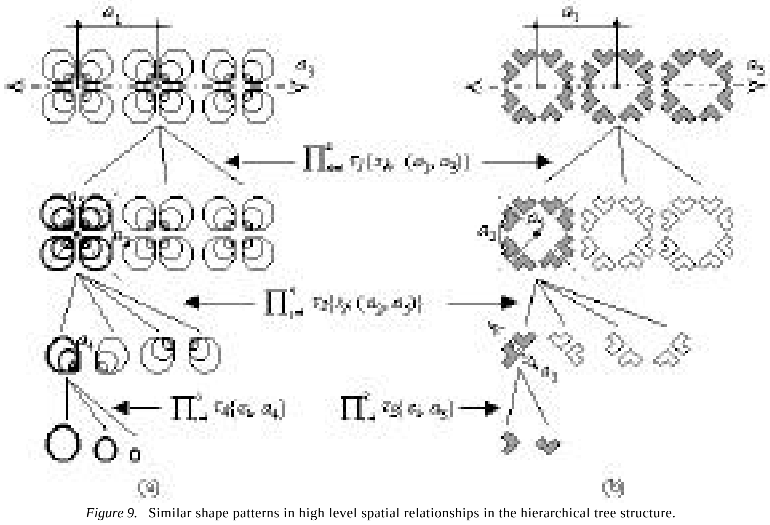


Figure 9. Similar shape patterns in high level spatial relationships in the hierarchical tree structure.

Some shape patterns may share the same relationships and arguments from the highest to the lowest in the hierarchical tree structure. The only differences may be at the lowest physical shapes. Their predicates and arguments in shape descriptions are the same, but the lowest shapes are different. These shape patterns are considered to be complete congruent shape patterns.

3. System Design and Implementation

3.1. SYSTEM DESIGN

The pattern recognition system is composed of the following modules: input module, output module, generalisation module and congruency identification module, Figure 10. The input module reads shape data from a file, then converts them into a data structure that is a list of polyline shapes and sends them into the generalisation and congruency identification modules. The output module describes all the pattern knowledge recognised from input data. The generalisation module groups and clusters shapes or shape patterns in terms of their congruency and then searches for high-level shape patterns. The congruency identification module identifies congruency between shapes, spatial relationships and shape patterns.

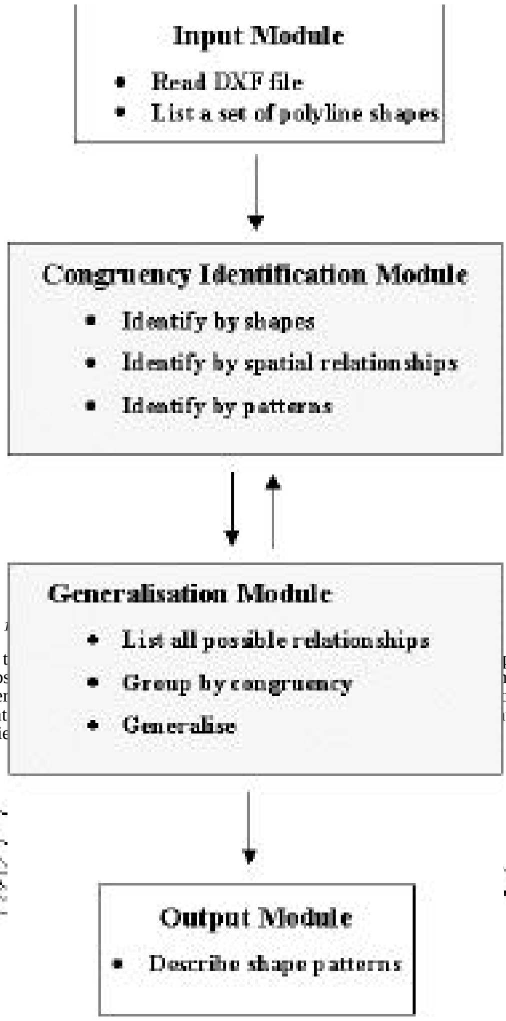
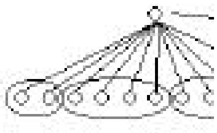


Figure 1

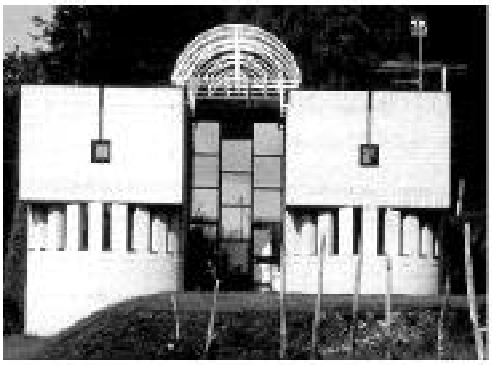
In the generalisation module data are grouped and clustered in terms of t pattern congruency. Shapes, relationships and patterns are represented as nodes in a hierarchical tree structure and encaps nilar so that they are grouped under a category. Then they are generalised and specify intermediate nodes between a super ract than subnodes, but more concrete than the supernode. Subnodes inherit properties of the intermediate node. The int tionship between shape elements or a pattern. Furthermore, a set of congruent intermediate nodes is grouped and specifie



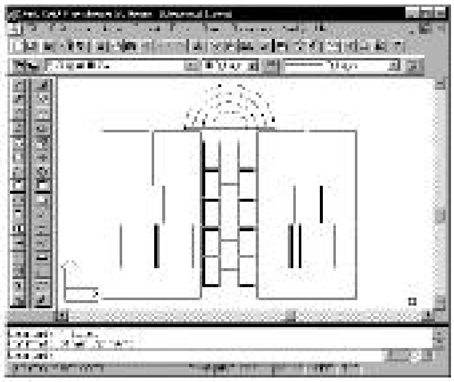
(a)

3.2. IMPLEMENTATION

The implemented shape pattern recognition system proceeds using a combination of the four modules. At first, this system reads a DXF data file exported from Autocad in the form of Ascii code, then converts it into a hierarchical tree data structure. The drawing in Figure 12(b) is drawn using Autocad from the picture of Mario Botta@s house in Origlio, Figure 12(a). This drawing is exported as a DXF data file. Then the shape recognition system reads the file.



(a)



(b)

The data structure in this system is a list of b he dot in the shape is the first vertex of the shape. This sl Section 2.2. Structural properties of all shapes are compared and congruent shapes are identified. During the structural property comparison process, vertex orders are synchronised, Figure 13(b). Group shapes are clustered and specify intermediate shape nodes. Figure 13(b) shows intermediate shape nodes with thick lines and their subnodes with dotted lines. Intermediate shape nodes have most of the properties that all congruent shapes have. In the congruency identification module, congruencies of shapes are confirmed in terms of structural properties, such as numbers of lines, inner angles of all vertices, ratios between the lengths of two consecutive lines. Congruency of relationships and patterns are determined in terms of comparing shape descriptions specified in Sections 2.3 and 2.4.

From a set of congruent shapes spatial relatic
two shapes on the right. All spatial relationsh
congruent shapes are compared and grouped a
module.

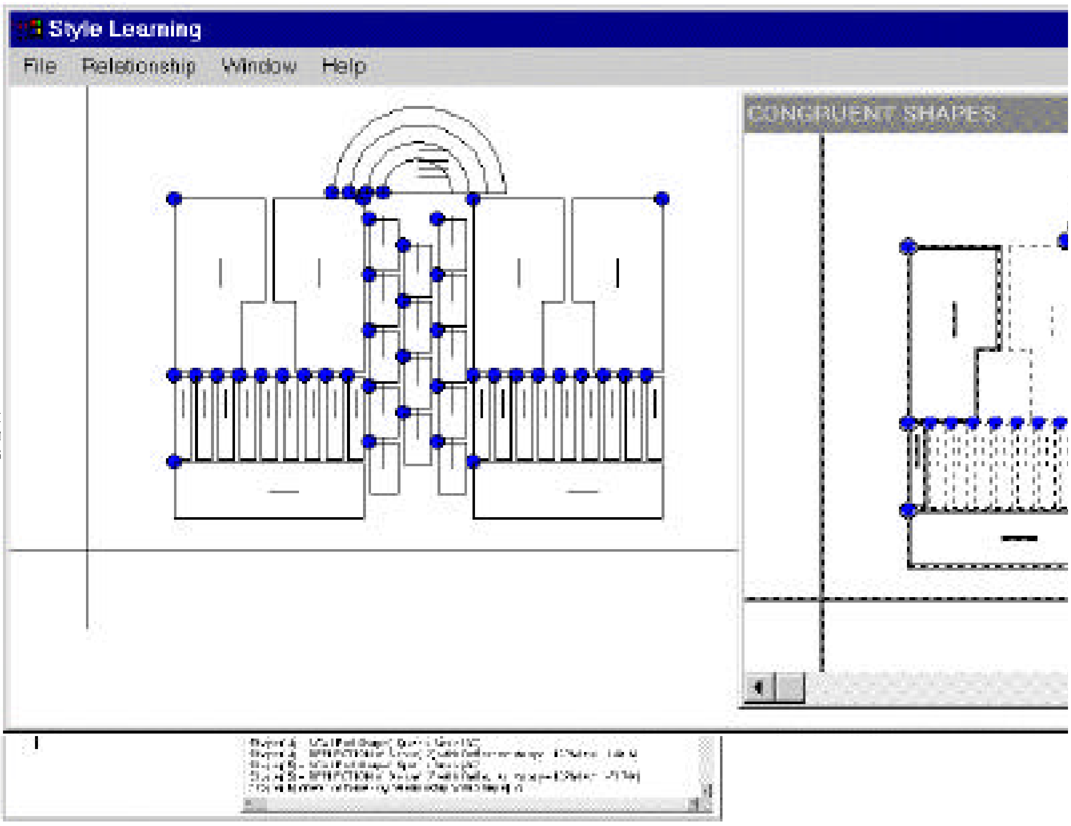


Figure 14. All possible relationships between two shapes from a set of congruent shapes.

Group nodes are generalised in terms of the values of arguments. Translation relationships that share the same type of translation axis and the same translation distance are grouped together and specify an intermediate node. Rotation relationships that have the same rotation centre and the same rotation angle are grouped. Scale relationships that have the same value for the scale factors are grouped. Useless relationship nodes are destroyed. With these processes, low-level shape patterns are recognised and presented as in Figure 15.

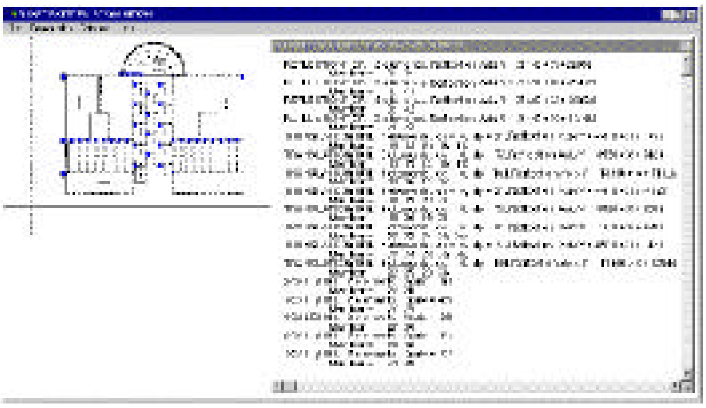


Figure 15. Low-level shape patterns recognised and represented from shapes.

From the lists of low-level shape patterns, congruent shape patterns are grouped again and constructed into a hierarchical tree structure. For example, four possible shape patterns are recognised from the drawing in Figure 16. Two congruent reflection shape patterns (a and b) are reflected again by the reflection axis a_3. Then they can be constructed into a hierarchical tree structure as shown in Figure 17.

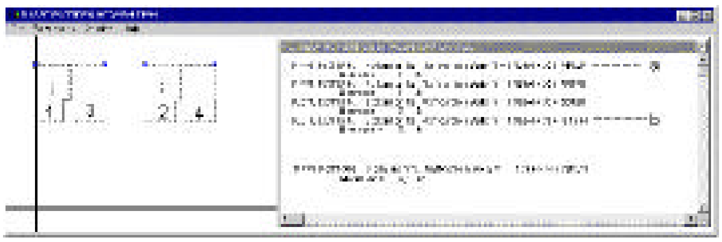


Figure 16. Shape pattern recognition and representation from congruent shape patterns.

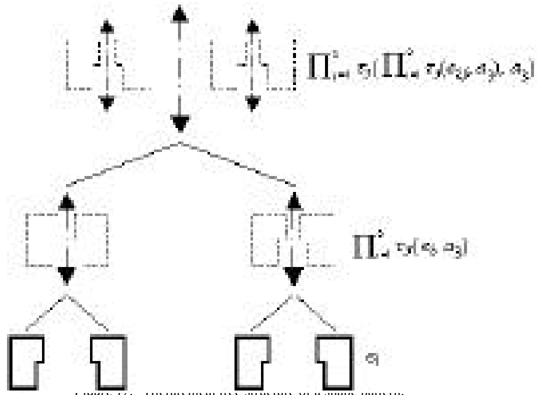


Figure 17: Hierarchical tree structure of a shape pattern.

4. Conclusion

A shape pattern recognition system is a system able to recognise pattern knowledge in shape drawings and describe it with symbols and numbers. Examples of successful recognition are given in Figures 15 and 16. Pattern knowledge is in the form of a hierarchical tree structure, Figure 17. The shape pattern recognition system was implemented using C++ under the Windows 95 environment. It can read any polyline drawings that are converted to DXF files. The two most important modules in this system are the generalisation module and congruency identification module. These two modules are interdependent. The sort algorithms used are above O(n) but are not fully polynomial implying that the system would scale up well.

A shape pattern recognition system using pattern representation supports many areas in design computation, such as style learning, analogical reasoning in shapes and measurement of shape complexity. Shape patterns can be recognised from a class of shape objects and common shape patterns among these are a learned style that characterises the class objects. Common shape patterns can be identified using inductive generalisation rules. There are two kinds of style: prototype and family style based on the notion of Wittgenstein@s (1960) family resemblance. Combinations of prototype and family styles and their instantiations produce design results that belong to the style. Learned style and pattern knowledge are abstract and high-level knowledge that can be used and applied either within that domain or between domains for analogical reasoning in shapes. According to Vosniadou (1989), the mechanism of analogical reasoning is the identification and transfer of structural information from a known system (the source) to a new and relatively unknown system (the target). Learned shape patterns stored in memory are retrieved if they have a similarity with the target design system in some way, then structural knowledge (pattern) is mapped onto the target. Furthermore complexity and simplicity of shapes can be measured in terms of normalising unity out of variety. Unity can be specified by the length of shape pattern descriptions and variety can be specified by the length of total shape descriptions. Measurement of the shape complexity would be useful for evaluating aesthetic value for potential results.

During the shape pattern recognition process some patterns that are unusual are recognised as emergent shape patterns. These provide design knowledge for humans and computers and potentially support creative design.

Acknowledgements

This work has been supported by the University of Sydney Postgraduate Award and in part by the Australian Research Council.

References

Arnheim, R.: 1954, *Art and Visual Perception*, University of California Press, Berkeley and Los Angeles.

Cha, M. Y. and Gero, J. S.: 1997, Shape pattern representation for design computation, *Working Paper*, Key Centre of Design Computing, University of Sydney, Sydney.

Do, E. Y. and Gross, M. D.: 1995, Drawing analogies: Finding visual references by sketching, *ACADIA @95.*

Falkenhainer, B., Forbus, K. D. and Gentner, D.: 1989/90, The structure-mapping engine: Algorithm and examples, *Artificial Intelligence*, **41**, 1-63

Gero, J. S.: 1992, Creativity, emergence and evolution in design, *in* J. S. Gero and F. Sudweeks (eds), *Preprints Computational Models of Creative Design*, Department of Architectural and Design Science, University of Sydney, Sydney, pp. 1-28

Gero, J. S and Jun, H. J.: 1995, Getting computer to read the architectural semantics of drawings, *ACADIA '95*, pp. 97-112

Gero, J. S. and Jun, H. J.: 1998, Shape semantics emergence of architectural shapes, *Environment and Planning B: Planning and Design* (to appear).

Gero, J. S. and Yan, M.: 1994, Shape emergence by symbolic reasoning, *Environment and Planning B: Planning and Design*, **21**, 191-212.

Gross, M. D.: 1994, Recognising and interpreting diagrams in design, *in* T. Catarci, M. Costabile, S. Levialdi and G. Santucci (eds), *Advanced Visual Interfaces*, ACM Press.

Jain, A. K.: 1987, Advances in statistical pattern recognition, *in* P. A. Devijver and J. Kittler (eds), *Pattern Recognition Theory and Applications*, Springer-Verlag, New York, pp. 1-19

Justema, W.: 1976, *Pattern: A Historical Panorama*, Paul Elek, London.

Kohler, W.: 1930, *Gestalt Psychology*, G. Bell and Sons, London.

March, L. and Steadman, P.: 1971, *The Geometry of Environment: Mathematics and Science in Design*, RIBA, London.

Mitchell, W. J.: 1990, *The Logic of Architecture: Design, Computation and Cognition*, MIT, Cambridge, MA.

Rips, L. J.: 1989, Similarity, typicality, and categorisation, *in* S. Vosniadou and A. Ortony (eds), *Similarity and Analogical Reasoning*, Cambridge University Press, Cambridge, pp. 21-59

Rosenman, M. A.: 1995, An edge vector representation for the construction of 2-dimensional shapes, *Environment and Planning B: Planning and Design*, **22**, 191-212.

Rowland, K. F.: 1964, *Pattern and Shape*, Ginn and Company, Oxford.

Steadman, P.: 1983, *Architectural Morphology: An Introduction to the Geometry of Building Plans*, Pion, London.

Stiny, G.: 1975, *Pictorial and Formal Aspects of Shape and Shape Grammars*, Birkhauser Verlag, Switzerland.

Stiny, G.: 1976, Two exercises in formal composition, *Environment and Planning B: Planning and Design*, **3**, 187-210

Stiny, G.: 1980, Introduction to shape and shape grammars, *Environmental and Planning B: Planning and Design*, **7**, 343-351

Uhr, L.: 1973, *Pattern Recognition, Learning and Thought: Computer-programmed Models of Higher Mental Processes*, Prentice-Hall, NJ.

Vosniadou, S. and Ortony, A.: 1989, Similarity and analogical reasoning: A synthesis, *in* S. Vosniadou and A. Ortony (eds), *Similarity and Analogical Reasoning*, Cambridge University Press, Cambridge, pp. 1-18

Wertheimer, M.: 1945, *Productive Thinking*, Harper, New York.

Wittgenstein, R.: 1960, *The Blue and Brown Books*, Basil Blackwell, Oxford.

This paper is a copy of: Cha, M. Y. and Gero, J. S. (1998). Shape pattern recognition using a computable shape pattern representation, *in* J. S. Gero and F. Sudweeks (eds), *Artificial Intelligence in Design '98*, Kluwer, Dordrecht, pp. 169-188.