FEATURE BASED QUALITATIVE REPRESENTATION OF ARCHITECTURAL PLANS

Information contained in 2-dimensional design drawings.

JOHN S. GERO AND JULIE R. JUPP
Key Centre of Design Computing and Cognition
University of Sydney NSW 2006 Australia
Email address: {john, jupp_j}@arch.usyd.edu.au

Abstract. This paper develops an approach to the qualitative representation of architectural plan drawings. We describe a schema for representing the internal shapes features and their associated spatial relations using syntactic pattern and contour specifications. This schema uses a qualitative symbolic representation to detect features. An example application of this representation is presented.

1. Introduction

This paper describes a qualitative approach to the modelling of 2-dimensional drawings applicable to representing architectural plans. The aim of our approach is to produce a canonical representation that captures information relating to the qualitative character of the drawing and to computationally analyse and compare a corpus of architectural plans.

We draw on the work of Gero and Park's (1997) feature-based qualitative modelling of shapes known as Q-codes to develop a representation of shape and spatial relations. In a previous paper (Gero and Jupp, 2002) we identified the need to represent not only the 1-dimensional shape features in the Q-code schema but to include 2D spatial features, that is, the design drawing's internal topology. We expand the existing Q-code schema to describe information derived from both shape and spatial characteristics whilst maintaining the feature-based approach and the equivalent analogy with language. The extension represents classes of shape features that include symbolic descriptions of their organization.

1.1. BACKGROUND AND MOTIVATION

In designing representation plays an important role, as the design drawing is not only a description of what is but it is also an exploration of what it might be (Mitchell, 1990). Any representation therefore poses restrictions on the ways in which the designer can reason about a drawing, especially during the early stages of designing and analysis tasks. Imposing one structure on a drawing may therefore not adequately capture the meaning of it. Since the designer is already restricted in a planning sense to a geometrical discipline on a 2D canvas it is important to maximize the potential of a drawing's representation and develop a computationally manipulable symbolic representation.

The current approach to the representation of shapes in CAD systems is to be explicit about both the geometry and to a lesser extent, shape topology. This is important for CAD since shapes need to be represented on the screen and so are made available as mathematical representations and manipulations. Such numerically based representation is not adequate for capturing design knowledge related to qualitative characteristics. Shape topology is central to representing spatial relations, since cognitively humans recognize and identify space not only through complex forms (by registering their characteristic features), but by also their configurations (Treisman and Gelade, 1980). Topology is therefore essential to any formal system of drawing representation because of its ability to structure many of our reasoning capabilities.

In developing a representation of aspects of both geometry and topology we utilise concepts derived from feature-based modelling. This approach allows us to capture design knowledge related to the qualitative character of a 2D drawing and produce classes of shape features rather than simply instances of them. Gero and Damski (1999), in their paper on feature-based modelling of objects, enumerated the benefit of using features as being able to lend themselves to semantic interpretation that have meaningful labels for designers. From this regard, qualitative shape features add a level to what a system can reason about an architectural plan drawing.

The remainder of the paper outlines a qualitative schema for representing shape features and their spatial relations. Section 2 develops representations based on syntactic pattern and contour representation methods. Section 3 describes how shape semantics may be derived and turns qualitative descriptions of shape and its organization into 2D semantics. The section ends with a discussion of how we may develop further design knowledge relating to qualitative spatial characteristics in architectural planning and derive their spatial semantics.

2. 2D Shape Representation

In 2D design drawings there exists an interrelation between the whole and its parts, as well as the hierarchic scale of importance by which some structural features are more dominant than others. Although we can perceive an organised structure in shapes their arrangement may limit what is directly apparent in the perception of an individual shape. Experiments in perception show that the mind organizes visual patterns spontaneously in such a way that the simplest available structure results (Zifp, 1949; Arnheim, 1969). If a figure can be seen as a combination of one large and one small square it is more readily apprehended than the combination of one square and four “L” shapes as illustrated in Figure 1(e).

Arrangements in the layout of 2D plans are called orderly when an observer can perceive their overall arrangement and the consequence of the individual structures (Arnheim, 1969). Order makes it possible to focus on what belongs together and what is segregated. However arrangements outside the shape do not always reflect a shape’s inner structure. Geometrical elements used to describe a shape transform due to the different relationships that are created by a shape’s connections with other shapes. In other words, the shape structure remains constant yet the corresponding elements that previously defined the shape are transformed by the addition of another shape. Figure 1 illustrates some possible combinations for different types of connectivity for Shape **A**.

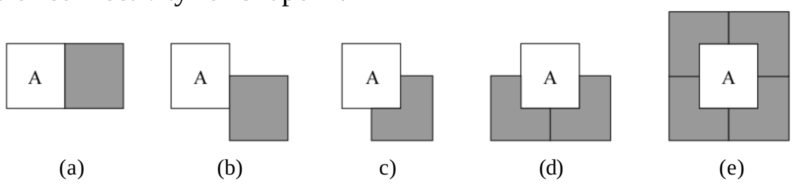


Figure 1. (a) Adjacent, (b) adjacent offset, (c) and (d) indentation or protrusion, and (e) surrounded

Shape **A** in Figure 1 maintains the same structural description of four adjacent right-angles. However when shape **A** is combined with one, two, four or more other shapes (in a finite number of ways) it produces: (a) a new description at the intersection and (b) additional intersections.

Representing and evaluating shapes in isolation does not carry sufficient information. It is not adequate to represent a shape only in terms of its structure, it needs to be represented in relation to the organisation which it is a part of. The shape structure may be explicit and yet misleading, because its structure does not correspond to the arrangements embedded in its contours. In this paper we address this lack of correspondence between outer and inner order where the spatial layout of floor plans reflect the distribution and interconnections of various shapes.

2.1. QUALITATIVE SYMBOLIC REPRESENTATION OF 2D DRAWINGS

Whatever is expressed within an architectural plan, whatever practical function it may serve and however it is constructed the choice of form in designing is constrained by what is geometrically and topologically possible (Steadman, 1983). For this reason, the majority of shape and spatial descriptors have been concerned with geometry and to a lesser extent topology.

We restrict ourselves to rectilinear shapes and their spatial relations. Our objective is to establish shape features as classes derivable from the intersection of line segments under the following conditions:

i) *bounded rectilinear polyline shape* – a shape that is composed of a set of only perpendicular straight lines where for any point on its contour there exists a circuit that starts from and ends at any vertex without covering any vertex more than once. These shapes are closed, without holes and are oriented vertically and horizontally;

ii) *primitive shape* – a shape that satisfies the conditions in (i) and is also represented explicitly, initially and thus can be input and manipulated by specifying the behaviours of its vertices; and

iii) *shape union* – a shape that satisfies the conditions in (i) and exists as a composite of two or more bounded rectilinear polyline shapes.

2.2. 2D DRAWINGS AS GRAPHS WITH CONTOUR SPECIFIED FEATURES

We take the standard graph theoretic representation of shape contours and add intersection semantics to its vertices and arcs. This specification method provides a description for shape features that are represented in terms of contour and position. This method is applied to plan graphs. A *plan graph* is a diagrammatic graph version of the plan itself. Elements associated with this graph are defined as follows:

Definition 1: In a plan graph let $\boldsymbol{v}_q^{\boldsymbol{a}}$ be a vertex, $\boldsymbol{v}$, where $\boldsymbol{a}$ is the list of arcs that intersect at $\boldsymbol{v}$ and $\boldsymbol{q}$ the Q-code that describes its intersection type. $\boldsymbol{v}_q^{\boldsymbol{a}}$ carries a minimum of two arcs $\boldsymbol{a}$, and includes all arcs within the plan graphs including the arcs that describe the external walls of the plan.

For example, the symbol $\boldsymbol{v}_q^{12}$ denotes the intersection of the two arcs $\boldsymbol{a}^1$, $\boldsymbol{a}^2$, where the order of the superscripts is significant. An enclosed shape contour in plan graph is represented by at least four arcs $\boldsymbol{a}^1$, $\boldsymbol{a}^2$, $\boldsymbol{a}^3$, $\boldsymbol{a}^4$ and can be represented at the vertices as:

$$(\ \boldsymbol{v}_q^{12}, \boldsymbol{v}_q^{23}, \boldsymbol{v}_q^{34}, \boldsymbol{v}_q^{41}\ ) \tag{1}$$

where (1) represents a bounded rectilinear polyline shape.

2.3. Q-CODE REPRESENTATION

Vertices can have one of five qualitative values representing the intersection type. Each intersection is defined through the following:

Definition 2: (convex) Let **L** be the label for the vertex produced by two arcs intersecting when viewed from inside the acute angle produced by the intersection of the walls they represent.

Definition 3: (concave, complement of convex) Let **G** be the label for the vertex produced by two arcs intersecting when viewed from the complementary angle.

Definition 4: (straight/right-angle) Let **T** be the label for the vertex produced by three arcs intersecting when viewed from inside either of the acute angles.

Definition 5: (complement of straight/right-angle) Let ^ be the label for the vertex produced by three arcs intersecting when viewed from the complementary angle.

Definition 6: (four right-angles is its own complement) Let **+** be the label for the vertex produced by four arcs intersecting when viewed from inside any of its acute angles.

The Q-codes L, G, T, ⊥ , and + define labels for five kinds of intersection of arcs for rectilinear shapes and their surrounding spaces. Figure 2 describes these values pictorially and illustrates the distinguished arcs as well as the viewpoint.

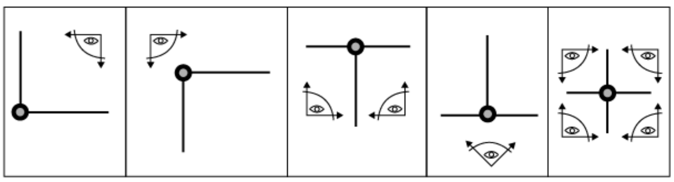


Figure 2. (a) and (b) two arcs, (c) and (d) three arcs, and (e) four arcs.

2.4. SHAPE EXTRACTION AND EMBEDDING

At the contact of two or more shape contours, there are extraction and embedding relationships for the qualitative values of intersection. Where this occurs there is a transformation in representation.

Shape extraction is the process of removing a shape from its surroundings. There are five rules of shape extraction, which reduce the description of shape connectivity to a structural representation only. For each process if we isolate an individual shape contour the following rules apply:

For a boundary intersection, Ø may be extracted from the vertex: $\boldsymbol{v}_{\wedge}$

$$\boldsymbol{v}_{\wedge} \Rightarrow Ø \tag{2}$$

Vertex $\boldsymbol{v}_{\mathsf{L}}$ may be extracted from the vertex: $\boldsymbol{v}_{\mathsf{L}}$

$$\boldsymbol{v}_{\mathsf{L}} \Rightarrow \boldsymbol{v}_{\mathsf{L}} \tag{3}$$

Vertex $\boldsymbol{v}_{\mathsf{G}}$ may be extracted from the vertex: $\boldsymbol{v}_{\mathsf{G}}$

$$\boldsymbol{v}_{\mathsf{G}} \Rightarrow \boldsymbol{v}_{\mathsf{G}} \tag{4}$$

Vertex $\boldsymbol{v}_{\mathsf{L}}$ may be extracted from the vertex: $\boldsymbol{v}_{\mathsf{T}}$

$$\boldsymbol{v}_{\mathsf{T}} \Rightarrow \boldsymbol{v}_{\mathsf{L}} \tag{5}$$

Vertex $\boldsymbol{v}_{\mathsf{L}}$ may be extracted from the vertex: $\boldsymbol{v}_{+}$

$$\boldsymbol{v}_{+} \Rightarrow \boldsymbol{v}_{\mathsf{L}} \tag{6}$$

Shape embedding is the process of adding a shape to its surroundings. There are three conditions of embedding shapes that describe information at the vertex to define its connectivity. The conditions of shape embedding are:

Vertex $\boldsymbol{v}_{\mathsf{L}}$ may be embedded in each of the following vertices: $\boldsymbol{v}_{\mathsf{G}}$, $\boldsymbol{v}_{\mathsf{T}}$, $\boldsymbol{v}_{+}$

$$\boldsymbol{v}_{\mathsf{L}} \Rightarrow \{ \boldsymbol{v}_{\mathsf{L}} , \boldsymbol{v}_{\mathsf{T}} , \boldsymbol{v}_{+} \} \tag{7}$$

Vertex $\boldsymbol{v}_{\mathsf{G}}$ may be embedded in each of the following vertices: $\boldsymbol{v}_{\mathsf{L}}$, $\boldsymbol{v}_{\mathsf{T}}$, $\boldsymbol{v}_{+}$

$$\boldsymbol{v}_{\mathsf{G}} \Rightarrow \{ \boldsymbol{v}_{\mathsf{G}} , \boldsymbol{v}_{\mathsf{T}} , \boldsymbol{v}_{+} \} \tag{8}$$

For a boundary intersection Ø that describes two shape contours, Ø may be embedded in vertex $\boldsymbol{v}_{\wedge}$.

$$Ø \Rightarrow \{\boldsymbol{v}_{\wedge}\} \tag{9}$$

2.5. POSITION CONSTRAINTS

The position of vertices in relation to the overall shape contour and plan graph are described in terms of the order in which they appear in the graph. Position constraints concern the structures within which vertices are organized and are represented as groups of vertices. There are three kinds of intersection groups that have been enumerated by Gero and Yan (1994): ordinary groups, adjacent groups and enclosed groups, which respectively specify three kinds of topological structures by organizing vertices, shape contours and the overall plan graph in different ways:

a) An ordinary group is represented a pair of parentheses: "(" and ")", in which any two vertices in the group represent an arc.

b) An adjacent group is represented by a pair of angled brackets, “<” and “>”, in which only two adjacent vertices can represent an arc. The order of intersections in an adjacent group is significant.

c) An enclosed group is represented by pairs of square brackets “[” and “]”, and is a set of vertices which must satisfy (2), and the first and the last intersections in the group are adjacent to each other.

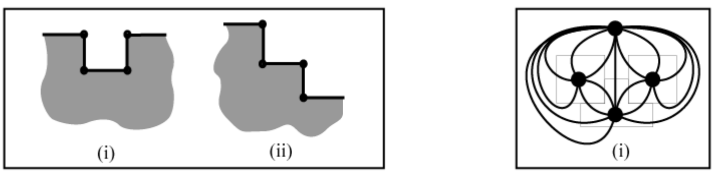
3. Q-code Graph Representation Analysis

Syntactic patterns for shape and spatial relations can be identified using these definitions of vertices. The process of discovering visual patterns from drawings is called shape semantics and plays an important role in organising and providing order.

3.1. Q-CODE SHAPE SEMANTICS AND SEMANTIC GRAPHS

Patterns that reflect basic shape features in terms of repetition and convexity, have been selected as they best distinguish feature characteristics in syntactic patterns. There are five shape semantics identified: indentation, protrusion, iteration, alternation and symmetry. Iteration refers to a repetition of patterns with no interval; alternation refers to a repetition of patterns with regular or irregular intervals; and symmetry refers to a reflective arrangement of patterns (not necessarily expressed as visual symmetry).

We differentiate two classes of shape semantics. The first is a pattern of intersection relationships, which are represented explicitly and includes three of the five semantics: indentation, protrusion and iteration. The second is a pattern abstracted from the combination of shape contours that exists only implicitly in the relationships of a shape’s connectivity derivable from the *vertex graph*[1] and include all five semantics. Figure 3 shows examples of each class of shape semantic.



(a) (b)

Figure 3. (a) Class 1 semantics: (i) indentation and (ii) protrusion, and (b) class 2 shape semantics: (i) spatial connectivity

[1] A *vertex graph* is a graph where spaces or 2D *planes* are represented by vertices and adjacent spaces connected by arcs.

Definition 7: (*Protrusion*) Let P be the symbol for protrusion where n an integer.

$$P = \mathsf{G}\,n(\mathsf{L})\,\mathsf{G} \tag{5}$$

Definition 8: (*Indentation*) Let I be the symbol for indentation where n an integer.

$$I = \mathsf{L}\ n(\mathsf{G})\ \mathsf{L} \tag{6}$$

Definition 9: (*Iteration*) Let E be the symbol for iteration where n an integer.

$$E = n(\mathsf{L}) \wedge n(\mathsf{G}) \wedge n(\mathsf{T}) \wedge n(\perp) \wedge n(+)\ . \tag{7}$$

Definition 10: (*Alternation*) Let A be the symbol for alternation where n an integer.

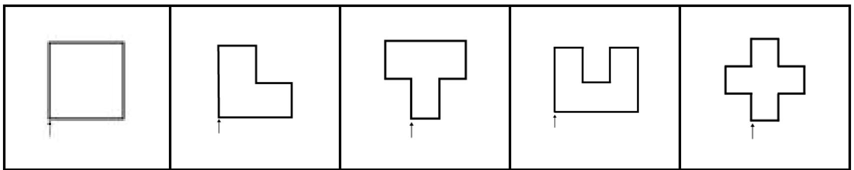
$$\mathrm{A} = n(\mathsf{L}) \wedge \vee\ \mathrm{n}(\mathsf{G}) \wedge \vee\ \mathrm{n}(\mathsf{T}) \wedge \vee\ \mathrm{n}(\perp) \wedge \vee\ \mathrm{n}(+). \tag{8}$$

Definition 11: (*Symmetry*) Let S be the symbol for symmetry where n an integer.

$$\mathrm{S} = \{\ \mathrm{n}\ Q \vee\ Q^{\,c}\ \} \tag{9}$$

Where $Q^{\,c}$ is the complement of Q, where Q is the class descriptor.

Figure 4 illustrates some syntactic pattern encoding for five primitive shapes.



Rectangle	*Saw-Tooth*	*Tee Shape*	*U Shape*	*Cruciform*
$E =\ _{4*\mathrm{L}}$	$E =\ _{3*\mathrm{L}}$, $I =\ _{\mathrm{G}}$	$I =\ _{\mathrm{G}}$, $I =\ _{\mathrm{G}}$ $E =\ _{2*\mathrm{L}}$	$E =\ _{4*\mathrm{L}}$, $I =\ _{2*\,\mathrm{G}}$	$P =\ _{2*\mathrm{L}}$, $E =\ _{2*\mathrm{L}}$, $P =\ _{2*\mathrm{L}}$, $E =\ _{2*\mathrm{L}}$

Figure 4. Q-code semantics for primitive shapes.

3.2. SEMANTIC NETWORKS

Semantic graphs are used to represent design knowledge in 2-dimensional drawings because different design-related features can be abstracted at each level of the encoding procedure. By taking the graph theoretic representation and its intersection semantics we can produce a semantic network as the qualitative representation of the 2D plan drawing. From this representation we can then reason about the structures in the semantic network and their features.

The topology of 2D shapes can be represented as a secondary set of vertices and arcs. This is implemented through a semantic net graph and the arcs denote the connection between adjacent spaces (Mantyla, 1988) and all spaces on the boundary are connected to an external vertex. For each arc in the new semantic graph we assign a label “L”, “G’, “T”, “⊥”, or “+”corresponding to the labels given to the plan graph’s vertices. Figure 5 illustrates this graph for the plan drawing in Figure 5(a). Figure 5(b) shows four vertices, r, s, t and u (r is an external vertex), 16 arcs and 13 regions. Each arc is labelled by the Q-codes of the two vertices of the arc they cross in the plan graph in Figure 5(a). We use this semantic net graph to represent the sequence of Q-code encodings of a 2D design drawing, and to extract qualitative information, such as features by parsing this representation.

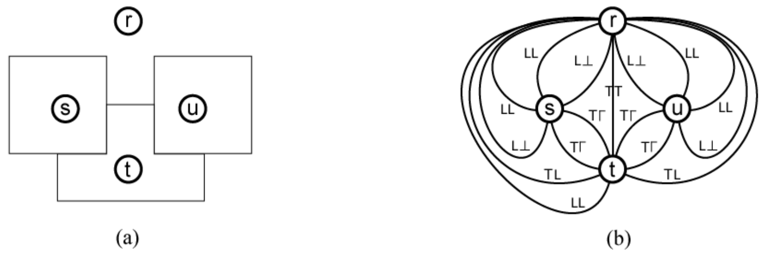


Figure 5. (a) Original drawing, and (b) semantic graph representation.

Let us restrict ourselves to plan graphs where only three arcs intersect at each vertex. Using this restriction there are 14 variants of intersection for “L”, “G’, “T” and “⊥” vertices illustrated in Figure 6, and can be generalised within a hierarchy of 10 states illustrated in Figure 7.

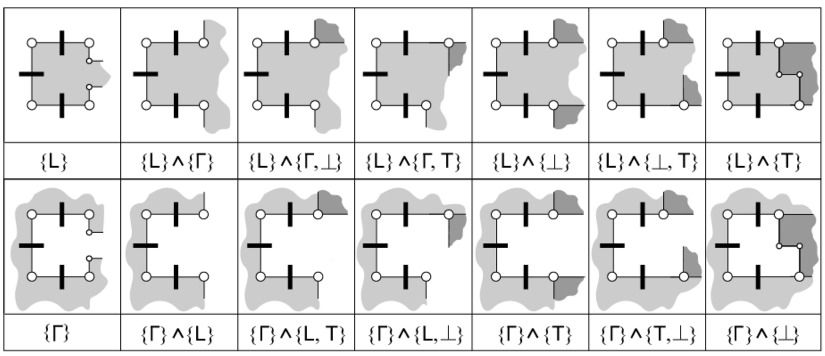


Figure 6. Intersection variants for “L”, “G’, “T”and “⊥”.

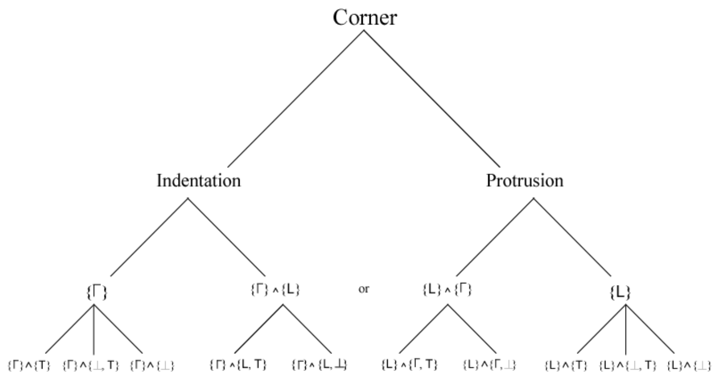


Figure 7. Hierarchy of corner intersection.

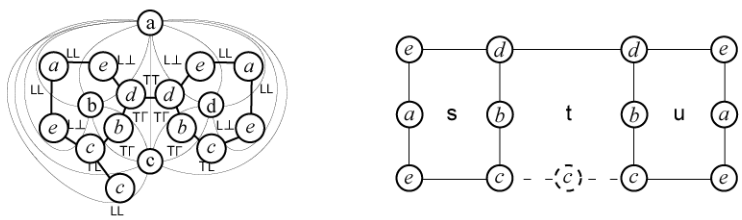
Continuing our example in Figure 5, Table 1 shows the five possible types of intersection for its arcs. They can be named: a exterior corner, b interior corner, c outward corner, d inward corner, and e recess corner.

Table 1. Semantic Labels for Vertex Graphs

a	b	c	d	e
(L,L) ∧ (L,L)	(T,T) ∧ (L, ⊥)	(L,T) ∧ (L, ⊥)∧ (T, G)	(L,L) ∧ (L,L)	(T, G) ∧ (L,T) ∧ (T, G)

3.3 VERTEX GRAPHS

The semantic net graph is sequenced (Kaufman, 1984) and labelled. Given the semantics of vertices outlined in the Table 1 it is now possible to re-represent the semantic net graph in Figure 5(b) as a vertex graph where vertices are the type of vertex in the semantic net graph and the arcs are connections between them. This process is repeated for the vertex graph's dual. In our example, there are five different types of vertex as shown in the graph in Figure 8.



(a) (b)

Figure 8. Re-representing the semantic net graph as a vertex graph, where s, t and u are the labels of the shapes in Figure 5(a).

The graph shown in dark lines in Figure 8(a) can be redrawn as shown in Figure 8(b). Information about the topology can now be obtained. In order to analyze the vertex graph model there are some semantic attributes necessary to interpret it. Table 2 shows some of these attributes:

Table 2. Some semantic attributions to vertices with three arcs intersecting

Feature	Chord	Additional constraints
Protrusion:		
4 arc	e/a	Every *a* is connected to *e*.
5 arc	ec/c/ce	*ec* is connected to *c* & the other *ec* is connected to *c*.
Indentation:		
2 arc	c/b/d	All *b* is connected to *c*; or All *b* is connected to *d*.
4 arc	dd /ee	All *d* are connected to *e*; or All *e* are connected to *d*.
Symmetry:		
Radial		All features are symmetrical rotated about a central axis.
Mirror		All features are symmetrical reflected about a central axis.
Step	e/c	All c are connected to e.
T-shape	b/cd	All *b* are adjacent to *cd*.

Using the interpretation shown in Table 2 and the graph in Figure 8(b) it is possible to detect the following features from our simple example:

- There is mirror symmetry in the all shape features and their spatial relations;
- There is an protrusion formed between shape contours s and u;
- Shape contours s and u are protrusions;
- There is an indentation formed in shape contour t.

Discussion

In this paper we show how intersection can form the basis of a qualitative representation for 2D architectural plan drawings that carries information about both shape and spatial relations in terms of structure, connectivity and arrangement. Knowledge about the arrangement of shapes plays an important role in early stages of design. As is the case in architectural planning design, it is often features in the connections that are as significant as the shapes themselves.

We have been able to show some basic shape features that can be obtained from graph theoretic modelling and how semantic information is carried from one representation to another. This syntactic representation scheme for qualitative characteristics of drawingsextends current methods and specialises them for design analysis tasks, tasks which previously were difficult.

Gero and Damski (1999) have demonstrated that it is also possible to derive complex features by concatenating simple features. Such complex features can then be connected to domain knowledge so that other forms of analysis can then be carried out. An obvious next question is whether it is possible to describe shape features that reflect spatial semantics. For example if shape or spatial features can be derived for a space to be described as open, warm or safe. This however is beyond the scope of this paper.

Acknowledgements

This research is supported by a Australian Postgraduate Research Award. Computing support is provided by the Key Centre of Design Computing and Cognition.

References:

Arnheim, R.: 1969, *Art and Visual Perception*, Berkeley and Los Angeles, University of California Press.

Egenhofer, M..J. and Shariff, A..R.: 1998, Metric details for natural-language spatial relations, *ACM Transactions on Information Systems* **16**(4), 295-321.

Gero, J.S. and Damski, J.: 1999, Feature-based qualitative modelling of objects, *in* G. Augenbroe and C. Eastman (eds), *Computers in Building,* Kluwer, Boston, pp. 309-320.

Gero J.S. and Jupp J.: 2002, Measuring the information content of architectural plans, **,** *in* P.L. Hippolyte and E. Miralles (eds), *SIGraDi Caracas 2002,* Ediciones Universidad Central de Venezuela, Caracas, pp. 155-158.

Gero, J.S. and Park, S-H.: 1997, Computable feature-based qualitative modelling of shape and space, *in* R Junge (ed.), *CAADFutures 1997,* Kluwer, Dordrecht, pp. 821-830.

Kaufman, A.: 1967, *Graphs, Dynamic Programming, and Finite Games*, Academic Press, New York.

Mantyla, M.: 1988, *An Introduction to Solid Modelling,* Computer Science Press, Rockville.

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

Treisman A.M. and Gelade, G.: 1980, A feature-integration theory of attention, *Cognitive Psychology* **14**, 97-136.

Zipf, G.K.:1949, *Human Behavior and the Principle of Least Effort*, Cambridge, Mass, Addison-Wesley,