JS Gero, B Tversky and T Knight (eds), *Visual and Spatial Reasoning in Design III*, 2004 Key Centre of Design Computing and Cognition, University of Sydney, pp 139-162

QUALITATIVE REPRESENTATION AND REASONING IN DESIGN: A HIERARCHY OF SHAPE AND SPATIAL LANGUAGES

JULIE R JUPP AND JOHN S GERO
University of Sydney, Australia

Abstract. A novel approach to shape and spatial representation and reasoning is proposed in this paper. We develop (i) a schema for qualitatively encoding two-dimensional design diagrams; and (ii) methods of recognition and classification for design reasoning. The schema is founded on boundary- and graph-based landmarks and utilizes qualitative feature based (QFB) representation techniques. The encoder invokes a qualitative schema describing morphological, topological and mereological features. The generic encoding format deals with recognition of invariant patterns and QFB representation facilitates robust measures of similarity using clustering algorithms. The proposed encoder-analyser (E-A) is demonstrated in experimental results. The preliminary study compares and classifies a variety of building plans. This paper concludes by illustrating how the E-A can be used as a basis for design reasoning.

1. Introduction

In all visual design domains, designers are influenced by their understanding of the external visual world. During the design process designers often use diagrams to facilitate problem solving which involves reasoning with and without ideas concerned with shape and form. Visual and spatial reasoning are fundamental in the solution process since recognising and manipulating shape is essential to external representations of design ideas.

To facilitate reasoning and design analysis a need exists to construct a representation scheme that automates recognition for two-dimensional (2D) design diagrams. Many solutions to the problem of recognition in 2D images have been proposed using a variety of data structures. The choice of data structure and applications to represent 2D images is crucial to the type of analysis tasks required. Generally, approaches to representation can be divided into quantitative and qualitative methods. Representational specifications for 2D images can be further divided into: grammar-like and non-grammar-like formalisms. We focus on qualitative feature-based (QFB) specification to explore a multiple level representation of 2D information. Within this framework we may then obtain a measure of design similarities.

The approach is structured as follows. We have chosen to deal with the problem of shape and spatial recognition in design drawings. Shape and spatial recognition is only one aspect of image processing but is important in reasoning and design analysis tasks. In particular we look at architectural design analysis and the 2D building plan. This problem can be decomposed into two stages: encoding for representing building plan elements and relations in terms of morphology, topology and mereology; and matching the generated code with known feature patterns for a particular class of design. The encoding schema corresponds to invariant coding procedures. The procedure employs qualitative descriptions of boundary- and graph-based landmarks invariant to scaling, rotation and shift in patterns. The output of the encoder acts as the input to a pattern recognition scheme, which consists of clustering algorithms. The complete framework of the Encoder-Analyser (E-A) is shown in Figure 1.

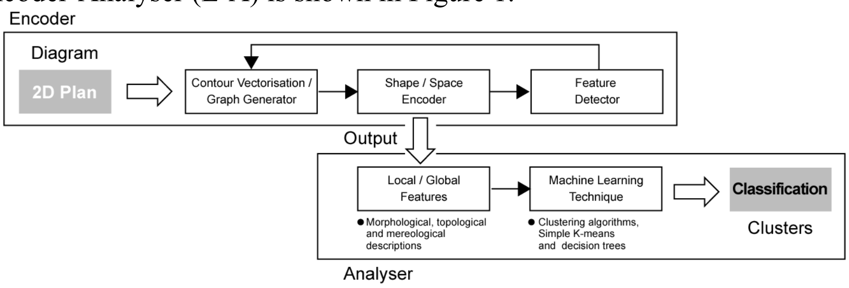


Figure 1. Encoder-analyser (E-A) framework

We propose a qualitative encoding schema based on the description of rectilinear shapes and spaces using symbolic values. A generic, string format is used and the encoding procedure converts the drawing into a one-dimensional (1D) representation and a set of graphs. The core idea is that design drawings can be uniquely characterised by the representation of embedded shape and spatial features. Each embedded shape and spatial feature is described by qualitative values and stored as a series of symbols in a 1D string and graphs. Features are identified based on semantics and string pattern matching techniques. The similarity between two design diagrams is then computed using clustering algorithms.

The remainder of this paper is divided into six sections. A survey of previous work is carried out in Section 2. The qualitative representation of 2D diagrams as 1D discrete strings and graph diagrams is described in Section 3. This schema builds on previous work in shape representation (Park and Gero 1997; Gero and Jupp 2003) by adding spatial descriptors. These concepts are demonstrated in a preliminary study on residential building plans and the results are presented in Section 4. The study compares the design drawings of Frank Lloyd Wright, Louis Kahn and Mario Botta; illustrating that similarity based classifications of design are reliant on a combination of shape and spatial features. Section 5 discusses various issues of this approach and Section 6 concludes the paper.

2. Shape and Space

2.1. VISUAL PERCEPTION AND COGNITION

Visual perception and cognitive research converge on the study of object description (Wertheimer 1923). Object's such as 2D diagrams carry with them a great deal of information. From a viewer's perspective, a diagram or image is immediately understandable or not, based on the ease with which it can be processed. This depends on both elements and relationships (Klinger and Salingaros 2000). Elements and structures are called orderly when an observer can perceive their overall arrangement as a consequence of individual elements or relationships (Arnheim 1969). Our ability to perceive allows us to order through identifying, differentiating and associating features such as: the number of repeated occurrences, degrees within shape elements, and coherent units and structures. Ordering makes it possible to focus on what belongs together and what is segregated.

However, although we can perceive an organised structure in a diagram, arrangements may limit what is directly apparent in the perception of individual elements. For example, experiments in perception show that the mind organizes visual patterns spontaneously in such a way that the simplest available structure results (Zipf 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 2(e).

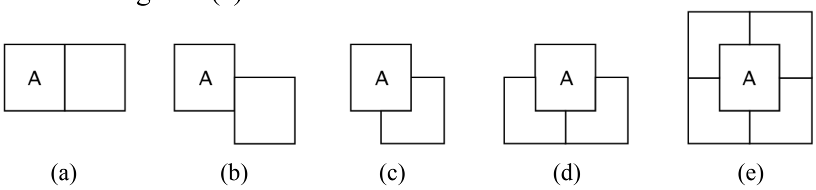


Figure 2. (a) Meet/ met-by, (b) offset, (c), (d) and (e) contains/ contained-by

The arrangements outside a shape do not always reflect its inner structure. Geometrical elements used to describe the shape transform due to the different relationships created by its connections with other shapes. For example, in Figure 2 the geometry of Shape A remains constant and is made up of four right angles, yet corresponding elements that previously defined the object are transformed by the addition of another object. Figure 2 illustrates some possible combinations for different types of connectivity for Shape A. Shape A maintains the same morphological description of four adjacent right-angles. However when shape A is combined with one, two, or more other shapes (in a finite number of ways) it produces: (a) a new description at its intersections and/or (b) additional intersections.

Thus, we assume that when perceiving a diagram, individuals perceive an interrelation between the whole and its parts, as well as a hierarchical scale of importance by which some elements or relations are more dominant than others. It is therefore, not adequate to represent shapes only in terms of its internal or isolated structures, they need to be represented in relation to the organisation which they are part of. The shape's physicality may be explicit and yet misleading, because a description of it may not correspond to the arrangements embedded in its contours.

Thus, different levels of processing are required in the recognition of diagrams, creating the need to take sensory data as input and produce higher-level information. Marr's (1982), computational theory of vision proposes that recognition and association are ultimately achieved using abstract features of relations that represent meaningful properties of the external world. Although Marr's theory is based on computational geometry algorithms its approach is related to the problem we are addressing. Descriptions of 2D diagrams should therefore be constructed at a variety of levels, abstracting away from the original representation and characterising, rather than exactly replicating perceived shapes and spatial relations. This raises the question: how can we represent shape and space computationally so that the description contains the knowledge required to recognise higher-level information for design analysis and reasoning tasks? We investigate one approach based on qualitative representation and reasoning.

2.2. QUALITATIVE REPRESENTATION

There is a large body of literature on qualitative representation. Schemes are commonly described as being either region-based or boundary-based. Region-based schemes found descriptions on shape interior (Brady and Asada 1984; Randell 1992; Cohn 1995; 1997). In contrast, boundary-based schemes typically describe types of localised features round the bounding edge of a region (Leyton 1988; 2001; Cinque and Lombardi 1995). Schemas of the boundary-based approach use descriptors which are ultimately analysable in terms of qualitative variation.

In design reasoning, qualitative representations of shape and space have not been as extensively studied. Gero and Park (1997) developed a schema founded on Freeman's chain coding scheme (1961) using landmark-based qualitative codes. Until recently, their approach was restricted to representing the outline or silhouette of shapes in isolation. A schema that extended landmark descriptions to include topological information about groups of shapes was developed by Gero and Jupp (2003). They proposed descriptions based on intersection-type applied to boundary landmarks. We extend this approach in a schema for shape and spatial representation that defines a hierarchy for a three-class qualitative language.

2.3. REASONING AND SIMILARITY-BASED ANALYSIS

Similarity is an important concept in design analysis and reasoning tasks. Once we are able to represent diagrams canonically, patterns and features identified can be compared to obtain a measure of their “likeness”. This can then be exploited for various tasks such as automated classification and information retrieval. The notion of “likeness” can be highly subjective, since it depends on the criteria chosen and therefore contextual knowledge is required.

Despite this requirement, there have been various solutions proposed for automated image processing and comparative analysis. Attneave (1966), in his theory on visual perception proposed that significant points (such as corners) contain the high information content necessary for successful shape recognition. Methods of polygon approximation (Pavlidis 1977) and formal grammars have also be used to define patterns, where parsing is used for matching. Similar approaches include feature extraction where recognition is achieved using either statistical pattern recognition (Gero and Kazakov 2001) or machine learning techniques (Colagrossi et al 2003). Colagrossi et al (2003) classify a sample of paintings by Mondrian by ascribing lines and areas a value called an order and use a neural network based classifier. Watanabe et al (1995) proposed a technique for recognition of table-form documents using graphs. A variety of form recognition studies base their systems on summarising line intersection information in a 1D string (Ting et al 1995; Lin et al 1996). The use of strings is based on structural and syntactic pattern recognition methods where a set of high level symbols is used to represent a pattern.

We investigate a similar approach in describing salient pictorial features as symbols and patterns of symbols. We implement clustering algorithms as the principal means for matching and computing a measure of similarity. The following section describes a computational means of representing shapes and spatial relations symbolically.

3. Qualitative Representation Schema

The description of shapes and space in any symbolic scheme may be treated as the problem of describing distinctive characteristics at the categorical level. Since shape and spatial characteristics can be treated as features, the representation of sketches and drawings involves recognizing, capturing and representing these features qualitatively as discrete symbols. The aim of any QFB representation is to produce a canonical representation analogous to a natural language that captures information relating to the qualitative character of the diagram.

The schema focuses on representing rectilinear shapes and their spatial relations. We establish shape and spatial features as classes derivable from the intersection of contours under the following conditions:

i) *bounded rectilinear polyline shape* – a shape 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 a vertex without covering any vertex more than once. Shapes are closed, without holes and oriented vertically and horizontally;
ii) *shape aggregation* – a shape that satisfies the conditions in (i) and exists as an aggregation of two or more other shapes.

The first principle of the approach is the encoding of vertices where qualitative changes occur. The system looks at vertices of shape contours and graph edges and captures distinctive physical characteristics. On each singular contour vertex or graph edge, a landmark value for a particular design quality (shape attribute or spatial relation) is abstracted into a single symbol.

We use standard first order logic and set membership notation with the following symbols: constants; connectives: $\wedge$ (and), $\vee$ (or), $\Rightarrow$ (if… then); quantifiers:, and sets: $\subseteq$ (is a subset of), $\cap$ (the intersection of). This specification method provides descriptions represented in terms of position, length, relation and area.

3.1. QFB SHAPE REPRESENTATION

We take the representation of shape contours and add intersection semantics to the vertices. Encoding follows where vertices are scanned and labelled in a counter-clockwise direction. As a result the symbol strings that represent the outlines of shapes are cyclic. The following three discrete stages describe the first class of qualitative representation in the schema hierarchy.

physicality ⟶ *symbol*

This specification method provides a description for shape attributes represented in terms of intersection type for contours: their relative position and length. Intersection attributes are encoded into qualitative value signs at the vertex as a landmark point. Landmarks are set when a new contour is compared to the previous contour. The schema can be defined by the following in relation to a 2D diagram:

Definition 1: Let v_c be a vertex, where c is the list of contours that intersect at v and q the qualitative symbol value that describes its intersection type.

A vertex must carry a minimum of two contours and includes both external (boundary) and internal contours.

$$v_c = q \tag{1}$$

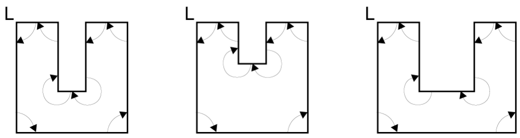
Definition 2: (convex) Let **L** be the symbol value produced by two contours intersecting at a vertex when viewed from (inside) the acute angle $\angle$.

$$\angle\{v_{(c \cap c+1)}\} \Rightarrow q = \mathbf{L} \tag{2}$$

Definition 3: (concave) Let $\urcorner$ be the symbol value produced by two contours intersecting at a vertex when viewed from the complementary angle $\angle$.

$$comp\angle\{v_{(c \cap c+1)}\} \Rightarrow q = \urcorner \tag{3}$$

As a consequence of the nature of the intersection types two shapes that look geometrically different may nonetheless have the same qualitative description. An example is shown in Figure 3, where a sample of geometrically different shapes are described by the sequence: L,L,L,L,L,$\urcorner$,$\urcorner$,L (commencing at landmark L for all three shapes).



(a) (b) (c)

Figure 3. U-Shape examples L, L, L, L, L, $\urcorner$, $\urcorner$, L

Geometric differences are included by adding three auxiliary attributes for relative lengths of segments (Gero and Park 1997). Definitions 1 and 2 are annotated with a symbol value indicating relative length. The landmark provides a ratio to distinguish the relative difference under the labels of *equal to*, *greater than* or *less than*. These auxiliary codes describe lengths between the previous contour and the current contour. We define equal to: $q_=$; greater than: $q_>$; and less than: $q_<$; where q is the qualitative symbol value **L** or $\urcorner$. Thus in Figure 3, shape (a) is described by the sequence: $L_>, L_=, L_=, L_<, L_>, \urcorner_<, \urcorner_>, L_<$; shape (b) is described by: $L_>, L_=, L_=, L_<, L_=, \urcorner_<, \urcorner_>, L_=$; and (c) is described by: $L_>, L_>, L_<, L_<, L_>, \urcorner_<, \urcorner_>, L_<$.

Where there is contact of more than two contours at a single vertex, the representation of shape attributes is transformed (Gero and Jupp 2003). Vertices of this type can be described by one of the following three qualitative symbol values describing intersection type.

Definition 4: (straight + two right angles) Let **T** be the symbol value produced by three contours intersecting at a vertex when viewed from (inside) either of the two acute angles $\angle$.

$$\angle^j, \angle^k\{v_{(c \cap c+1 \cap c+2)}\} \Rightarrow q = \mathbf{T} \tag{4}$$

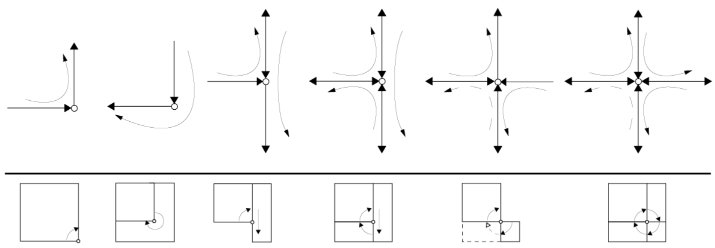
Definition 5: (complement of straight + two right angles) Let ⊥ be the symbol value produced by three contours intersecting at a vertex when viewed from the complementary of the two acute angles ∠.

$$comp\angle^{j,k}\left\{v_{(c\,\cap\, c+1\,\cap\, c+2)}\right\} \Rightarrow q = \perp \qquad (5)$$

Definition 6: (four right-angles and its own complement) Let **+** be the symbol value produced by four contours intersecting at a vertex when viewed from the inside any of its acute angles ∠.

$$\angle^{j}, \angle^{k}, \angle^{l}, \angle^{m}\left\{v_{(c\,\cap\, c+1\,\cap\, c+2\,\cap\, c+3)}\right\} \Rightarrow q = + \qquad (6)$$

A distinction is made between morphological descriptions (L, ˥,) and topological descriptions (T, ⊥, and +) where the latter focuses on concepts of connectedness that emerge from descriptions of shape aggregation. Thus, a critical difference exists in the scanning and labelling of vertices for aggregated shapes. Isolated shapes (and embedded shapes) contain vertices with only two contours each (see examples in Figures 4 (a) and (b)), and therefore have one scanning direction. Aggregated shapes can contain vertices with three or four contours i.e. multi-region vertices, and therefore have more than one scanning direction, Figure 4.



(a) (b) (c) (d) (e) (f)

Figure 4. Scanning directions

Figure 4 illustrates vertices involved in one (a), two (b), (c) and (e), three (d), (e) and (f) or four (f) regions.

symbol ⟶ regularity

The physicality and connectivity of a shape is described as a sequence of symbols which is assumed to denote design characteristics of a building plan. Some of these characteristics are easy to identify from structural regularities in symbol strings, while others are more difficult because they appear in more complex patterns. Transformation from sequences of symbols (unstructured) to regularities (structured) brings interpretation possibilities.

Patterns that reflect basic repetitions and convexity are: indentation, protrusion, iteration, alternation and symmetry. Iteration refers to a repetition of patterns with no interval; alternation refers to a repetition of patterns with irregular intervals; and symmetry refers to a reflective arrangement of patterns (not necessarily expressed as visual symmetry). The five syntactic regularities and their definitions are listed below.

Definition 7: (indentation) Let $\mathcal{I}$ be the symbol for indentation where n is an integer:

$$\mathcal{I} = \mathsf{L}\; n\,(\rceil)\; \mathsf{L} \tag{7}$$

Definition 8: (protrusion) Let $\mathcal{P}$ be the symbol for protrusion where n is an integer:

$$\mathcal{P} = \rceil\; n\,(\mathsf{L})\; \rceil \tag{8}$$

Definition 9: (iteration) Let $\mathcal{E}$ be the symbol for iteration where n is an integer:

$$\mathcal{E} = n\,(\mathsf{L}) \wedge n\,(\rceil) \wedge n\,(\mathsf{T}) \wedge n\,(\perp) \wedge n\,(+) \tag{9}$$

Definition 10: (alteration) Let $\mathcal{A}$ be the symbol for alternation where n is an integer:

$$\mathcal{A} = n\,(\mathsf{L}) \wedge\vee\; n\,(\rceil) \wedge\vee\; n\,(\mathsf{T}) \wedge\vee\; n\,(\perp) \wedge\vee\; n\,(+) \tag{10}$$

Definition 11: (symmetry) Let $\mathcal{S}$ be the symbol for symmetry where n is an integer, d is the class descriptor and $compd$ is the complement of d:

$$\mathcal{S} = \{\, n\,(d) \wedge compd \,\} \tag{11}$$

A pattern of symbol sequences can denote specific categories of shape classes that are well known or familiar in contour.

regularity $\longrightarrow$ *feature*

Syntactic regularities identified from the symbol sequence become shape features. Discovering visual patterns plays an important role in organising and providing order and is known also as shape semantics. Shape features are recognised by matching symbols with an existing feature knowledge base. Since shape features are derived from basic neighbouring shape elements we describe them as local. The five syntactic regularities listed above define five atomic local shape features, i.e., indentation, protrusion, iteration, alternation and symmetry.

Conceptual units are also defined for local shape features, which correspond to how they can be chunked. These units define four discrete levels (Gero and Park 1997). The terminology used for these conceptual units correspond to terms used in natural language. Conceptual units and their definitions are provided in Table 1.

Local shape features are used as the basis for reasoning about design diagrams. For example, it is possible to determine categorical information about shapes, since by identifying syntactic regularities patterns can be compared. In the following section we extend this schema to include spatial relations by abstracting two additional levels of information. Each level is in keeping with the same three discrete stages presented in this section.

TABLE 1. Conceptual unit definitions (after Gero and Park 1997)

Unit	Definition
Word	Sequence referring to a shape pattern with a particular design meaning
Phrase	Sequence in which one or more words show a distinctive pattern of structural arrangement
Sentence	An aggregation of words and phrases so that it refers to a closed and complete shape contour
Paragraph	A group of sentences where an aggregation of shapes are described without any spatial relationships

3.3. QFB SPATIAL REPRESENTATION

The formal treatment of visual languages is often based on graph representations. In the following we utilise graphs in order to represent spatial information. We maintain our analogy with language since by generalising descriptions of both adjacency and area in to a QFB language we are essentially moving from symbols related by one relationship (linear ordering given by sequencing) to multiple relationships which can be represented by graphs. Further, in assuming that diagrams can be represented by graphs, a spatial language is a set of such graphs abstracted from the original contour representation. The aim of constructing spatial descriptors as a second hierarchy of a qualitative codes is to produce spatial (global) features.

3.3.1. Graph diagrams designed for representing topology

The QFB approach to spatial descriptors is based on graph diagrams derived from the original contour representation. Graphs abstracted from contours are able to represent spatial topologies, which denote adjacency (Mantyla 1988). Graphs, as duals of the spatial layout, are constructed by locating new vertices in the centres of all bounded rectilinear polyline shapes, as well as one other vertex within the external region or background of the diagram.

Using this approach we examine two types of spatial relations. First, symbol values derived to represent properties of adjacency. Second, symbols are derived for area descriptions of regions. Figure 5 shows the original contours in (a) the location of vertices in (b) and graph diagram (c).

We consider how to define syntax and semantics from these graphs. In particular we ignore all structural constraints and simply regard the QFB language as a set of graph diagrams. We utilise symbol values produced in the previous level as our principal building blocks. In keeping with the previous three discrete stages described in Section 3.1 we present the second level of representation in the same format.

physicality ⟶ (dyad) symbol
Let us define an abstract syntax for QFB spatial descriptions of graphs.

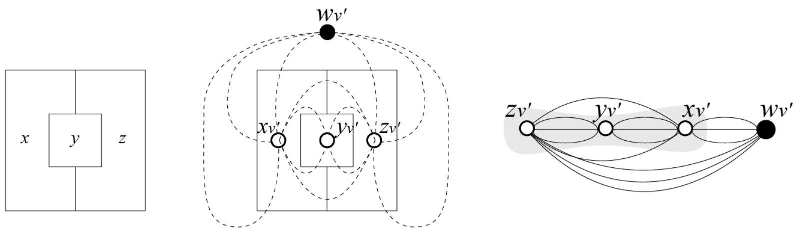


Figure 5. (a) Shapes *x*, *y* and *z*, (b) location of vertices and edges, *wv'*, *xv'*, *yv'*, *zv'*; and (c) sequenced graph diagram

Definition 12: Let *G* be an undirected graph with vertices $\boldsymbol{v'}$ located at centre of regions *r* and where edges $\boldsymbol{e}$ have a mapping defining for each edge the vertices it connects.

$$e \subseteq v', v' + 1 \tag{12}$$

After a graph is constructed it must then be sequenced (Kaufmann 1984) and labelled. The term topology network is used to label such a labelled graph. We label each edge with a pair of symbols; derived from the values of the previous level (for intersection), i.e., L, ⏋, T, ⊥, +. Therefore, labels assigned to edges correspond to the labels of the two vertices belonging to a shape contour. Edge labels are defined by the following:

Definition 13: Let $\boldsymbol{ds}$ be the set of dyad symbols for vertices $\boldsymbol{v_c}, \boldsymbol{v_{c+1}}$

$$ds \subseteq \{ (\text{L}, ⏋, \top, \perp, +) \wedge\vee (\text{L}, ⏋, \top, \perp, +) \} \tag{13}$$

Edges can be labelled therefore with one of 15 dyad symbol values to produce an adjacency description. The 15 dyad symbols have auxiliary symbol values indicating the relative area of regions.

Definition 14: Let a regular polygon be a region $\boldsymbol{r}$ and have an area $\boldsymbol{a}$ that is represented at the vertex $\boldsymbol{v'}$. The area of a regular polygon with *n* sides and side length *s* is given by:

$$a_{n\text{-}gon} = \tfrac{1}{4} ns^2 \cot\left(\pi/n\right) \tag{14}$$

A landmark is set to the numeric point of the magnitude of adjacent region areas providing a ratio to distinguish the relative area under the labels of *equal to*, *greater than* or *less than*, or *infinite* for all external vertices. We define equal to: $ds_{=}$; greater than: $ds_{>}$; less than: $ds_{<}$; and infinite: ds_{∞}; where ds is the qualitative dyad symbol value. If vertex v' is external define $a = \infty$.

Continuing the example given in Figure 5 we illustrate these mappings in Figure 6. Figure 6(b) shows four vertices: wv', xv', yv', and zv', (wv', is an external vertex), eight edges and six new (abstract) regions.

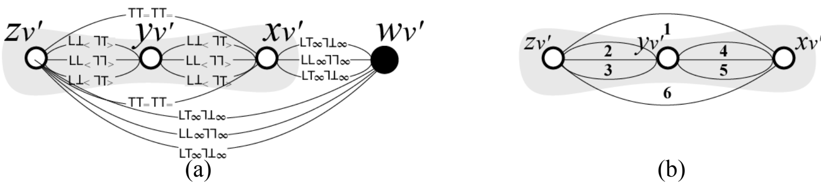


Figure 6. (a) Network: sequenced and labelled graph and (b) six new regions

In Figure 6 (a), edges are labelled according to the intersection type of the two vertices belonging to the contour it crosses (a dyad symbol) as well as the values describing the relative area of regions. Graph vertex labels are not required and thus abstract syntax is produced only for edges by the set ds and $\{= ; < ; > ; \infty\}$. This specification method provides a description for spatial attributes in terms of adjacency and area descriptors. In order to analyse the topology network semantics are defined.

symbol ⟶ regularity
The representation of dyad symbols reveals distinctive topological characteristics that can be recognised from syntactic regularities. Some of these characteristics are easy to identify, while others are more difficult. Unlike the morphological characteristics, topological characteristics contain variations depending on the viewpoint (orientation) of $\top$ and/or $\perp$ intersections. Depending ons their orientation, these dyad symbols can define two types of adjacency.

Topological features recognised in syntactic regularities of dyad symbols include: *complete adjacency*, *partial adjacency* and *offset*. Complete adjacency refers to a region having total adjacency along a boundary with another region; partial adjacency refers to a region having only incomplete adjacency along a boundary with another region; and offset refers to a region having adjacency shared along more than one boundary with another region. Definitions for adjacency regularities are provided:

Definition 15: C is a set of the ds: $\{\llcorner; \top; +\} \wedge\vee \{\llcorner; \top; +\}$; where C is a semantic symbol value denoting complete adjacency, and the set $\{k\}$ is labelled according to intersection type:

$$C \subseteq \left\{ (\llcorner \wedge \llcorner); (\top \wedge \top)^{*}; (+ \wedge +); (\llcorner \wedge \top)^{*}; (\top \wedge +)^{*}; (\llcorner \wedge +) \right\} \quad (15)$$

Definition 16: $\mathcal{R}$ is a set of the ds: $\{\llcorner; \urcorner; \top; \bot; +\} \wedge\vee \{\llcorner; \urcorner; \top; \bot; +\}$; where $\mathcal{R}$ is a semantic symbol value denoting partial adjacency, and the set $\{k\}$ is labelled according to intersection type:

$$\mathcal{R} \subseteq \{(\urcorner\wedge\urcorner)^*;(\top\wedge\top)^*;(\llcorner\wedge\top)^*;(\bot\wedge\bot);(\top\wedge\bot)^*;(\top\wedge+)^*;(\bot\wedge+);(\llcorner\wedge\bot)\} \quad (16)$$

Definition 17: $\mathcal{O}$ is a set of the ds: $\{\llcorner; \urcorner; \top; \bot\} \wedge\vee \{\llcorner; \urcorner; \top; \bot\}$; where $\mathcal{O}$ is a dyad symbol denoting offset, and the set $\{k\}$ is labelled according to intersection type:

$$\mathcal{O} \subseteq \{(\urcorner\wedge\urcorner)^*; (\llcorner\wedge\urcorner); (\urcorner\wedge\top); (\urcorner\wedge\bot); (\urcorner\wedge+); (\top\wedge\bot)^*\} \quad (17)$$

Note * denotes an exception, defined by the orientation of the intersection type relative to the adjacent region. Adjacency and area descriptions form semantic strings which are not oriented. All regions have four or more adjacency symbol values.

Topological features identified for the example from Figure 5 are illustrated in Figure 7. Figure 7(a) shows the six abstract regions and Figure 7(b) features identified from their dyad symbol values.

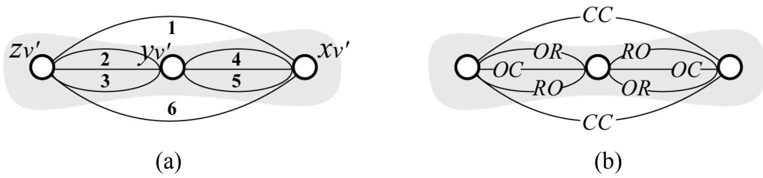


Figure 7. (a) Network: sequenced and labelled graph and (b) six new regions

In Figure 7(b), edges are labelled according to their feature set. The relations defined above can now be described symbolically, such that the spatial relationships can now be described semantically. Shape x to Shape y is offset and represented by $\mathcal{O}$; Shape x to Shape z has complete adjacency and represented by $\mathcal{C}$; Shape y to Shape x and z has complete and partial adjacency and represented by $\mathcal{C}$ and $\mathcal{R}$; Shape z to Shape x has complete adjacency and represented by $\mathcal{C}$.

regularity ⟶ feature

From the representation of dyad symbols we are able to add a level to the way in which we may reason about the diagram. Semantic regularities identified in dyad symbols produce spatial features termed global since neighbourhoods include multiple regions. Like local shape features, global spatial features are labelled by matching an existing feature knowledge base. The topological features identified at this second level are the first of two kinds of global spatial features and provide a basis for reasoning about spatial relations.

It becomes possible to determine categorical information about shape aggregations in spatial terms. The three syntactic regularities defined above can be seen as three spatial feature categories. Commonalities between these topological characteristics can be determined by comparing matchings and mismatchings. Comparison can be made either by comparing topological feature categories or by comparing single topological features.

3.3.2. Dual networks designed for representing mereology

Graphs are useful in organising two-dimensional drawings because different types and levels of features can be abstracted. In the previous section information about topological relations was abstracted from graph diagrams to produce topology networks, where labels are drawn from a finite alphabet. In this section we use the dual of the topology network to derive composite symbol values describing relations of contact and organisation. A topology network's dual is constructed by locating new vertices in the centres of all abstract regions whose edge does not connect with the external vertex. Using this approach we examine additional descriptions of spatial relations. The network in Figure 7(a) may be re-represented by abstracting its dual. Figure 8(a) shows the topology network and Figure 8(b) shows the dual topology network consisting of six new vertices (*f*-*k*), and six edges.

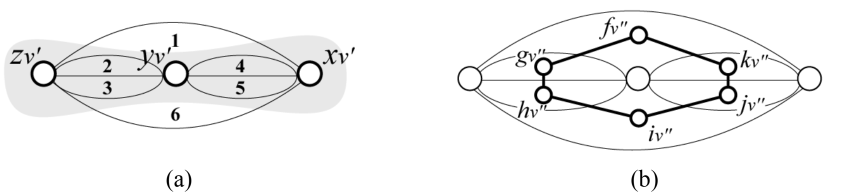


Figure 8. (a) Topology network and (b) dual topology network

We consider how to define syntax for the dual since topological features identified at the previous level can be translated further into meaningful spatial semantics. In order to do this we utilise the concept of mereology.

Mereology is an attempt to lay down the general principles underlying the relationships between a whole and its parts. The relations have been formally defined by thirteen interval relations (Allen 1984) for the temporal domain. This has allowed the formulation of ontological laws pertaining to the boundaries and interiors of wholes as well as to relations of contact and organisation (Aurnague and Vieu 1993). Since we are only interested in those instances where identities are in contact, the notions of "before/ after" do not apply here. Further, because of constraint conditions the relations: "starts/ started-by"; "finishes/ finished-by"; "during" and "equals" are also not applicable. As in the previous two levels we use the same three discrete stages to present the final level of representation.

physicality ⟶ *composite symbol*

Let us define an abstract syntax for QFB spatial descriptions of dual networks.

Definition 18: Let *DN* be a dual network with vertices v'' located at centre of abstract regions r' and where new edges e' have a mapping defining for each edge the vertices it connects.

$$e' \subseteq v'' , v'' + 1 \tag{18}$$

By constructing the dual of a topology network it is possible to abstract additional information. The dual carries with it a description of higher-level mereological relations. For each new edge e' labels are derived from the features identified at the previous level for topology, i.e., $\mathcal{C}$, $\mathcal{R}$ and $\mathcal{O}$, and correspond to graph edges $G(e)$. By taking the dual, composite symbol values are produced. Composite symbols are specified for organisation identities, Definitions of the three semantic regularities are provided below:

Definition 19: cs is a subset of topology feature types: $\{\mathcal{C}\wedge\vee\mathcal{R}\wedge\vee\mathcal{O}\}$; where cs is a composite symbol value, and is labelled according to feature symbols:

$$cs \subseteq \{(\mathcal{C}\,\mathcal{R}); (\mathcal{C}\mathcal{O}); (\mathcal{R}\mathcal{R})\}; \{(\mathcal{C}\mathcal{C})\}; \{(\mathcal{R}\mathcal{O}); (\mathcal{O}\mathcal{O})\} \tag{19}$$

This specification method provides a description of a 2D diagram relating to mereology.

composite symbol $\longrightarrow$ *regularity*

Definitions for basic semantic interpretations have been developed in order to reason about rectilinear spatial properties. Composite symbols allow semantic regularities to be identified. Dual networks are undirected and as a consequently regularities in composite symbols identify three pattern types: "overlaps/ overlapped-by", "meets/ met-by", and contains/ contained-by". Definitions for contact-organisation identities are given below.

Definition 20: (*Overlaps/ Overlapped-by*) Let $\mathcal{V}$ be the symbol for overlaps/ overlapped-by with *n* an integer.

$$\mathcal{V} \subseteq \{(\mathcal{C}\,\mathcal{R})\vee(\mathcal{C}\mathcal{O})\vee(\mathcal{R}\mathcal{R})\vee(\mathcal{R}\mathcal{O})\vee(\mathcal{O}\mathcal{O});\ n \leq 2\ [(\mathcal{R}\mathcal{O})\vee(\mathcal{O}\mathcal{O})]\} \tag{20}$$

Definition 21: (*Meets/ Met-by*) Let $\mathcal{M}$ be the symbol for meets/ met-by with *n* an integer.

$$\mathcal{M} \subseteq \{n(\mathcal{C}\mathcal{C}); n(\mathcal{C}\mathcal{C})\wedge(\mathcal{R}\mathcal{O})\vee(\mathcal{O}\mathcal{O}); n(\mathcal{C}\mathcal{C})\wedge n(\mathcal{C}\mathcal{R}) \vee (\mathcal{C}\mathcal{O}) \vee (\mathcal{R}\mathcal{R}); \\ n(\mathcal{C}\mathcal{C})\wedge n(\mathcal{C}\mathcal{R})\vee (\mathcal{C}\mathcal{O}) \vee (\mathcal{R}\mathcal{R})\wedge (\mathcal{R}\mathcal{O}) \vee (\mathcal{O}\mathcal{O})\} \tag{21}$$

Definition 22: (*Contains/ Contained-by*) Let $\mathcal{U}$ be the symbol for contains/ contained-by with *n* an integer.

$$\mathcal{U} \subseteq \{n(\mathcal{R}\mathcal{O})\vee(\mathcal{O}\mathcal{O})\wedge\vee(\mathcal{C}\mathcal{O}); 2(\mathcal{R}\mathcal{O})\vee(\mathcal{O}\mathcal{O})\wedge(\mathcal{C}\mathcal{R})\vee(\mathcal{C}\mathcal{O})\vee(\mathcal{R}\mathcal{R}); \\ n(\mathcal{C}\mathcal{C})\wedge n(\mathcal{R}\mathcal{O})\vee(\mathcal{O}\mathcal{O}); n{>}2(\mathcal{C}\mathcal{R})\vee(\mathcal{C}\mathcal{O})\vee(\mathcal{R}\mathcal{R});$$

$$n(CC) \wedge n(C\mathcal{R}) \vee (CO) \vee (\mathcal{RR}) \wedge n(\mathcal{R}O) \vee (OO); \} \quad (22)$$

Referring to the example, the relations defined above can now be described symbolically. Figure 9(a) shows the topology network and Figure 9(b) shows the dual topology network consisting of six new vertices (*f-k*), and six edges.

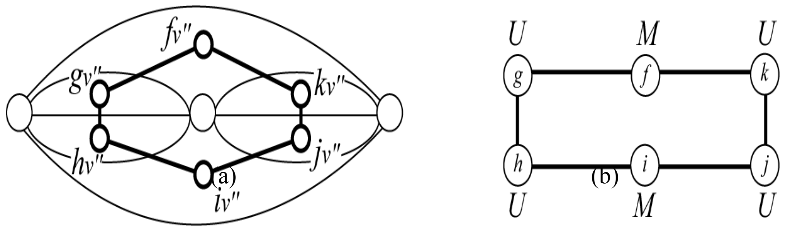


Figure 9. (a) Topology network, and (b) dual topology network

Spatial relationships between abstract regions *f, g, h, i, j, and k* (from shapes *x*, *y* and *z*) can now be described semantically as meets/ met-by and represented by $\mathcal{M}$, and contains/ contained-by and represented by $\mathcal{U}$.

regularity ⟶ feature

Once regularities of syntax patterns have been identified, each pattern is categorized. The three syntactic regularities defined above can now be seen as three spatial feature categories, i.e., overlaps/ overlapped-by, meets/ met-by, and contains/ contained-by. In addition to identifying mereological relations, it is possible to use these as features for the purposes of reasoning about the 2D plan as a whole.

This three class schema forms a hierarchical qualitative language for 2D architectural plan drawings that describes information about both shape and spatial relations in terms of shape structure, arrangement, area and organisation.

Knowledge about the spatial relationships of shapes plays an important role in early stages of design. As is often the case in architectural planning design, the organisations of shape are as significant as the shapes themselves. We have been able to extract qualitative representations for basic shape and spatial features from boundary- and graph-based landmarks and show how semantic information is carried from one representation to another. The schema extends current qualitative shape representation methods and specialises them for spatial analysis tasks, which have previously been difficult. Given the three levels of descriptive languages that describe 2D diagrams, we can represent plan drawings canonically in order to make comparisons based on their similarity.

4. Drawing Differentiation and Classification

The ability to differentiate and judge similarities between architectural plan drawings has motivated our approach to a hierarchy of QFB representation languages. The assumption we make is that designs are communicated in different ways and by sampling a corpus of 2D plans it is possible to identify patterns that distinguishes designs and their development over time. The type, frequency and sequence of features may be seen as the basis of the differentiation of a design and its classification. Shape and spatial features are the particular dimensions by which we measure plan drawing similarities. Feature values are used to perform clustering in order to recognize design attributes exemplar to different architects. This is similar to applications of author-recognition to written text. The idea applied here to architectural plan drawings investigates the possibility of distinguishing between architects or identifying an architect's different stylistic periods.

4.1. DRAWING ANALYSIS PROCEDURE

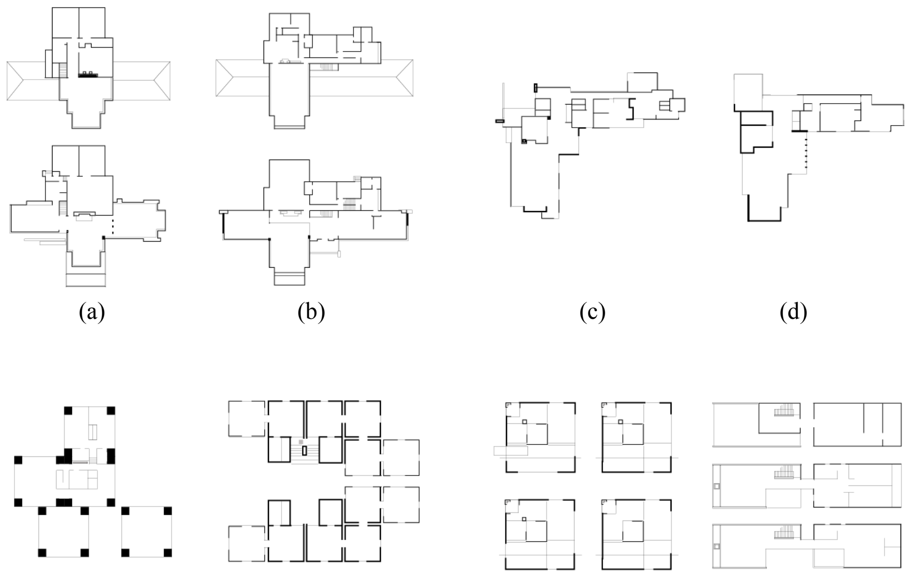
We have automated the encoding procedure and combined this with a machine learning method in order to measure design similarities. The E-A consists of four discrete sequential processes. These processes include: contour vectorisation and graph generator, shape/space encoder, feature detector, feature classifier, and continues in three cycles until a plan drawing, its graph diagram and dual have been encoded. The method of feature identification and re-representation is organised cyclically when more abstract features are identified on the basis of current available features, a new representation on the basis of these new features is produced. The resulting stings and symbol values are canonical representations of the original design drawing.

The drawing analysis technique implemented uses a similarity measure based on a clustering algorithm applied to the dataset. Output can then be analysed to extract design categories. The clustering technique is integrated by incorporating Weka 3.2 (Witten and Frank 2000) classes in to the E-A. Using this approach, any rectilinear architectural plan drawing can be handled.

We present a preliminary study to test the descriptive strength of shape and spatial features and evaluate the schema's ability to classify plan drawings. In this study we analyse the similarities of building plans by creating clusters that partition their features into similar groups, where features close to one another are assumed to be similar.

4.2. EXPERIMENT: RESIDENTIAL BUILDING PLAN ANALYSIS

Samples of work from three prominent architects are analysed in this study using the simple k-means clustering algorithm. Figure 10 illustrates a sample of plan drawings by Frank Lloyd Wright, Louis Kahn and Mario Botta, arranged in chronological order of production.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 10. Architectural plan drawings: (a), (b), (c) and (d) Frank Lloyd Wright; (e) and (f) Mario Botta; (g) and (h) Louis Kahn

Plan drawings in Figures 10(a) and (b) are the Roberts House (1908) and the Baker House (1909), regarded by historians and critics as belonging to Wright's "Prairie" style. Plan drawings in Figures 10(c) and (d) are the Garrison House (1940) and the Pope House (1940) and belong to Wright's "Usonian" style. Plan drawings in Figures 10(e) and (f) are the Alder Residence (1954) and Fleisher Residence (1959) and are two of Kahn's earlier designs. Plan drawings in Figures 10(g) and (h) are the Riva San Vitale House (1971) and the Ligornetto House (1976) and are also two of Botta's earliest designs.

Plans were encoded and features identified at each hierarchy of the schema. Figure 11 illustrates the presence and proportion of the local and global features categories extracted.

From Figure 11 we can see plans defined as a set of data items and characterized by their features at both local and global levels.

4.3.1. Clustering QFB categories

Using principal component analysis, two local feature categories: iteration, protrusion and four global feature categories: partial adjacency, offset, overlap and contains, were evaluated as the best set of attributes for clustering. This is significant since clustering relies on a combination of feature categories where the ratio of local to global features is 1: 2.

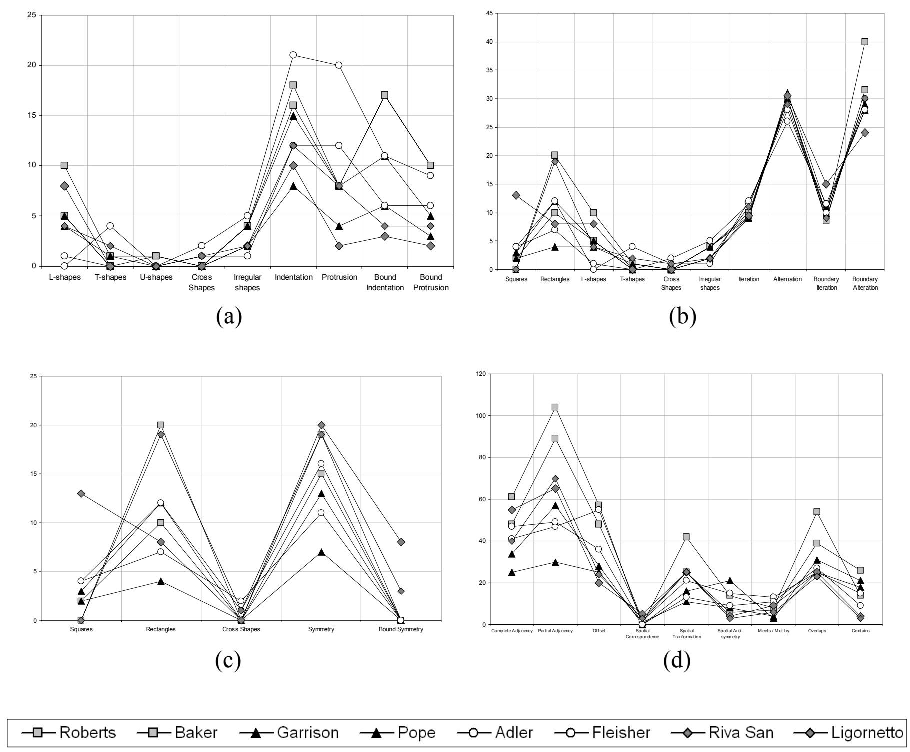


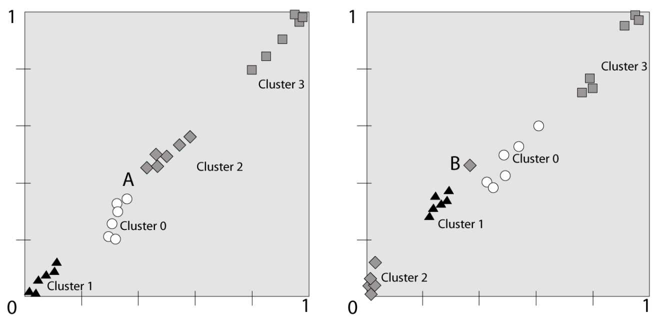
Figure 11. The occurrence of features on local/ global feature category for: plan drawings (a), (b) local and (c), (d) global feature categories

The k-means algorithm grouped plans creating one set of clusters that partition the data into similar groups. Samples close to one another are assumed to be similar. Classification found the correct number of clusters, meaning that the k-means did not lose any, which is possible. The cluster visualisation in Figure 12 shows the four clusters produced.

To visualise clusters we divided feature categories into two groups. Figure 12(a) shows clusters of features: indentation, partial adjacency and Figure 12(b) features: protrusion, offset, overlaps, contains. Variables were normalised in order to compare each feature category.

The simple k-means algorithm clustered the plans correctly: *cluster 0*: Kahn, *cluster 1*: Wright “Usonian”, *cluster 2*: Wright “Prairie”, and *cluster 3*: Botta. This is significant since clustering was able to differentiate between all three architects as well as differentiating between two stylistic periods of a single architect, i.e., Wright's Prairie and Usonian designs.

Further insights can be interpreted from the clustering. In Figure 11(a), at point **A** the k-means algorithm clusters 0 and 2 together by the smallest overall distance. Since the distance between clusters is an indication of their similarity we can infer that clusters 0 and 2 share more similarities. This could be interpreted that Kahn and Botta's two samples of plan drawings share more commonalities with one another for occurrences of indentation, and partial adjacency. Other insights include similarities between both Kahn and Botta's residential designs and Wright's Prairies houses. The k-means algorithm identified greater similarity between Wright's Prairie and Kahn and Botta's residential designs than for Wright's Usonian houses.



(a) (b)

Figure 12. Clustering Result for k-means: (a) indentation, partial adjacency, and (b) protrusion, offset, overlapped, contains

In addition, the distance from a feature to a cluster reflects its degree of membership. In Figure 11(a), at point **B** the k-means algorithm incorrectly clusters an attribute, a protrusion feature category, from cluster 2 with cluster 0. This indicates that Botta's Ligornetto residence shares more protrusion similarities with Kahn's Fleisher residence.

We can verify the visualisation representing the clustering results by comparing metrics against insights derived from the visualisation. From Table 2, features whose cluster is 0, 1 or 3 have degrees of membership that are on average very high with low standard deviations. These features belong very strongly to their cluster. The textual description of what differentiates clusters 0 and 2 illustrate that they are more similar.

Based on these metrics, simple k-means clustering effectively compares plan drawings for one or more features and analyses the extent of feature similarities as well as identifying what differentiates them. The results, although not for a large enough data set to analyse further statistically, are promising given the distinct partitioning within the results and the visual similarities that we may intuitively see from the three architect's plan drawings. The overall performance of the encoding schema demonstrates in this preliminary study the similarities and differences in all eight plans.

TABLE 2. Similarity of feature categories

Cluster ID	%	Size	Architect	Feature Differentiator	Mean	Stdev.
0	25	2	Kahn	Morphology: # protrusion	11	3.2
				Topology: # partial adjacency	48	1.4
				Mereology:	17	5.6
1	25	2	Wright: Usonian	Morphology: # indentation	4.5	0.7
				Topology: # offset	26.0	2.1
				Mereology: contains	13.5, 19.5	3.5, 2.1
2	25	2	Botta	Morphology: # protrusion	5.0	4.2
				Topology: # partial adjacency	67.5	3.5
				Mereology: overlaps	23, 24	1.4
3	25	2	Wright: Prairie	Morphology: # indentation	7.5	3.5
				Topology: # partial adjacency, offset	96.5, 52.5	10.6, 6.3
				Mereology: overlaps, contains	33.5, 46, 26	12,10.6, 8.4

5. Discussion

There have been few reports about hierarchical QFB recognition and automatic design knowledge acquisition. Although Gero and Kazakov (2001) attached qualitative representation and reasoning to architectural plans, as an important subject for making categorisation available, the approach did not utilise spatial knowledge and handled only the outlines of plan drawings. For more meaningful design analysis the representation and reasoning problem can be solved with additional information describing not only plan morphology, but also characteristics of plan topology and mereology.

Our preliminary study demonstrates analysis and reasoning of existing (feature) knowledge in 2D architectural design. The results indicate that qualitative representation of shape and spatial relations can be used to identify similarities of feature categories based on the clustering of data sets. Clusters of the plans sampled show the ability to learn the appropriate range for matching. Although the number of building plans analysed in the study is small and clustered into discrete data sets, the sample does demonstrate the basic dynamics of forming clusters based on similarities between feature categories. For more complex data types such as large commercial plan drawings and greater sizes of data sets, we need only to test more complex clustering algorithms and matching functions, such as decision trees (Jain et al 1999). How well this works in practice must be determined in future research. Further, our assumption that architectural plan drawings are composed of closed rectilinear polyline shapes is not always applicable to design drawings in which many designs are composed of angles and curves.

However it is possible to relax encoding constraints in order to handle non-rectilinear shapes and incorporate a wider range of designs.

The E-A model presented here provides the basis for new kinds of design tools. The applications of our technique are wide ranging and include design diagram identification, indexing, retrieval, and robust description for 2D diagrams in computational design reasoning. Current CAD systems are unable to aid the designer in the perception of figures and gestalts and in the recognition and categorisation of shape and spatial characteristics. Categorisation of design features is important and influential during designing since it enables the designer to extend design knowledge by grouping or classifying according to some distinguishable properties. The approach presented in the E-A model can potentially assist designers in useful ways by "amplifying the mind's eye" (Fish and Scrivener 1999). Automatic identification of visual similarities makes past designs relevant to present ones and consequently information about a design can be categorized and re-categorised. A fully automated approach to classification of a variety of shape and spatial features like that presented here is required if the advantages of computer-aided design and planning is to be exploited in support systems.

6. Conclusion

This paper addresses qualitative representation concepts for shape and spatial relations of rectilinear 2D diagrams. In particular, the paper proposes an E-A model for local and global feature categories to acquire morphological, topological and mereological knowledge. By constructing a hierarchy of qualitative languages for shape and space we have automated the recognition, capture and re-representation of 2D design features. Our model has demonstrated that similarity exists not only between shape features, but also between relationships of spatial features, providing an additional level of reasoning about architectural plans. Together, shape and spatial feature categories present a novel approach to reason about 2D design.

Acknowledgements

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

References:

Allen, JF: 1984, Towards a general theory of action and time, *Artificial Intelligence* **23**: 123-154.

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

Aurnague, M and Vieu, L: 1993, A three-level approach to the semantics of space, *in* C Zelinsky-Wibbelt (ed), *The Semantics of Prepositions. From Mental Processing to Natural Language Processing*, Mouton/de Gruyter, Berlin, pp. 393-439.

Brady, M and Asada, H: 1984, Smoothed local symmetries and their implementation, *International Journal of Robotics Research* **3**(3): 36-61.

Cinque, L and Lombardi, L: 1995, Shape description and recognition by a multi-resolution approach, *Image and Vision Computing* **13**: 599-607.

Cohn, AG:1997, Qualitative spatial representation and reasoning techniques, *in* G Brewka, C Habel and B Nebel (eds), *Proceedings of KI-97*, LNAI, Springer-Verlag, **1303**, pp. 1-30.

Cohn, AG: 1995, A hierarchical representation of qualitative shape based on connection and convexity, *in* AU Frank and W Kuhn (eds), *Spatial Information Theory: A Theoretical Basis for GIS*, Lecture Notes in Computer Science, Springer-Verlag, **988**, pp. 311-326.

Colagrossi, A, Sciarrone, F and Seccaroni, C: 2003, A methodology for automating the classification of works of art using neural networks, *Leonardo* **36** (1): 69-69.

Fish, J and Scrivener, S: 1999, Amplifying the mind's eye: Sketching and visual cognition, *Leonardo* **23**(1): 117-126.

Freeman, H: 1961, On the encoding of arbitrary geometric configurations, *IRE Trans. On Electronic Computers* **EC-10**: 260-268.

Gero, JS and Jupp, J: 2003, Feature-based qualitative representations of plans, *in* A Choutgrajank, E Charoenslip, K Keatruangkamala and W Nakapan (eds), *CAADRIA03*, Rangsit University, Bangkok, pp. 117-128.

Gero, JS and Kazakov, V: 2001, Entropic similarity and complexity measures for architectural drawings, *in* JS Gero B Tversky and T Purcell (eds), *Visual and Spatial Reasoning in Design II*, Key Centre of Design Computing and Cognition, University of Sydney, Sydney, pp. 147-161.

Gero, JS 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.

Jain, A, Murty, M, Flynn, P: 1999, Algorithms for clustering data: A review, *ACM Computing Surveys* **32**(3): 264-322.

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

Klinger, A and Salingaros, NA: 2000, Complexity and visual images, *Environment and Planning B: Planning and Design* **27**: 537-547.

Leyton, M: 1988, A process-grammar for shape, *Artificial Intelligence* **34**: 213-247.

Leyton, M: 2001, *A Generative Theory of Shape,* Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg New York.

Lin, J-Y; Lee, C-W and Chen, Z: 1996, Identification of business forms using relationships between adjacent frames, *Machine Vision Applications* **9**(2): 56-64.

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

Marr, D: 1982, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, WH Freeman, New York.

Pavlidis, T: 1977, *Structural Pattern Recognition*, Springer-Verlag, Berlin.

Randell, DA, Cui, Z and Cohn, AG: 1992, A spatial logic based on regions and connection, *Proc. 3rd Int. Conf. on Knowledge Representation and Reasoning*, Cambridge MA, pp. 165-176.

Ting, A, Leung, M-K, Hui, S-C and Chan, K-Y: 1995 A syntactic business form classifier, *Proceedings 3rd ICDAR*, pp. 301-304.

Watanabe, T, Luo, Q and Sugie, N: 1995, Layout recognition of multi-kinds of table-form documents, *IEEE Trans. Pattern Anal. Mach. Intell*. **17**(4): 432-445.

Wertheimer, M: 1923, Untersuchungen zur Lehre von der Gestalt, *Psychologische Forschung* **4**: 301-350.

Witten, IH and Frank, E: 2000, *Data Mining: Practical Machine Learning Tools with Java Implementations*, Morgan Kaufmann, San Francisco.

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