Towards Active Support Systems for Architectural Designing

RABEE M. REFFAT AND JOHN S. GERO
Key Centre of Design Computing and Cognition
The University of Sydney
Australia
Email {rabee, john@arch.usyd.edu.au}

Abstract:

This paper proposes the application of a situated learning approach in designing integrated with a conventional CAD system. The approach is implemented in SLiDe (Situated Learning in Designing) and integrated as SLiDe-CAAD, to provide interactive support in designing exemplified within the composition of architectural shapes. SLiDe-CAAD is proposed to assist in exploring the design space for various alternatives of design compositions; recognising shape semantics from a design composition; and in maintaining the integrity of shape semantics or desired design concepts of interest in the design composition. SLiDe-CAAD is introduced to provide a collaboration between the designer and the computer during the process of designing.

Key Words:

CAAD systems, active designing support, situatedness, learning, artificial intelligence.

1. Introduction

The use of computers as designing support tools has implications for the way designing is carried out. During the last three decades of the twentieth century CAD systems provided support for calculation, documentation, animation and modelling of designs. On the other hand, Artificial Intelligence (AI) has been utilised to build design systems in two distinct ways; either to replicate the behaviour of designers while designing or to provide design support to designers. Much of CAD and AI-based design systems are passive systems in the sense that they do not modify their behaviours in response to the changes in the design environment. In architectural designing most of the current CAD systems can only be used at the very late stages of the design process after most of the major design decisions have been made and very few can be used during the conceptual stages of designing.

During the process of designing solutions are fluid and emergent entities, generated by dynamic and situated designing activities. Furthermore, the result of designing is not based on actions independent of what is being designed or independent of when, where and how it has been designed. In designing, neither the goal state nor the solution space is completely predetermined. The ability to provide useful designing support at the conceptual stages of designing to accommodate the situated and fluid nature of early schematic designing is important in designing. Such support creates the opportunity for conventional CAD systems to assist designers in exploring the solution space. It provides designers with useful and applicable design knowledge during the generation of design concepts and maintains the integrity of these concepts during different stages of designing. The usefulness of such knowledge is based on its applicability to a situation rather than determined a priori.

2. Situated Learning of Architectural Shape Semantics

In architectural designing, as in many other design disciplines, shape composition is an important design activity. Through shapes designers express ideas and represent elements of design, abstract concepts and construct situations. Hence, their role in designing is significant. The formation and discovery of relationships among parts of a design composition are fundamental tasks in designing (Mitchell and McCullough, 1995; Kolarevic, 1997). One of the analyses of architectural shapes in a drawing is the result of certain relationships among it parts which characterise each and every design. The abstraction and explicitness of these relationships in a recognised drawing can therefore lead to a closer and better understanding of shape semantics (Koutamanis and Mitossi, 1993). Shape semantics encapsulate design knowledge that appears in design artefacts. Shape semantics are the interpretation of visual patterns or visual forms of groups of shapes in the drawing (Jun, 1997).

Designers interpret and perceive their designs differently and discover various shape semantics related to their interest from their design compositions. Multiple representations provide the opportunity for a wide range of interpretations where each interpretation reveals certain shape semantics. Hence, multiple representations may allow implicit shape semantics in one representation to become explicit in another representation. There is a vast range of possible architectural shape semantics, which could be emerged. Shape semantics are recognised in terms of similarity of spatial relationships as well as physical properties. A group of shape semantics recognised in each representation forms an observation. A set of observations can be constructed from a set of multiple representations.

Situated learning of architectural shape semantics is mainly concerned with locating shape semantics in relation to their situations within which they were recognised in the design environment. This situatedness of any of the shape semantics is not determined a priori but constructed based upon what is there in the design environment. What make one situation different from or similar to others are the relationships they express in which relevant distinctions could be made among situations within the design environment. The constructed set of observations can be viewed as an internal design environment a learning system might construct to learn the situatedness of knowledge within that environment. The regularities of relationships among shape semantics across the observations are the triggers to learn the situatedness of these semantics. The importance of these regularities lies in the development of coherent distinctions among situations

3. SLiDe: A Computational Situated Learning System in Designing

A computational system for situated learning in designing (SLiDe) is implemented and exemplified within the domain of architectural shape semantics (Reffat and Gero, 2000). Its underlying concepts could be used in other domains. SLiDe consists of three primary modules: Generator, Recogniser and Incremental Situator. The Generator is used by the designer to develop a set of multiple representations of a design composition. This set of representations forms the initial design environment of SLiDe. The Recogniser detects the design environment and produces a set of observations, each of which consists of a group of shape semantics recognised in each representation. The Incremental Situator consists of two sub-modules: Situator and Restructuring Situator. The Situator module locates the recognised shape semantics in relation to their situations by constructing the regularities of relationships among shape semantics across the observations and clustering them in situational categories organised in a hierarchal structure. Such relationships change over time due the nature and fluidity of designing in which changes in the design environment. The Restructuring Situator updates previously learned situational categories and restructures the hierarchy accordingly.

4. Active Support Systems for Architectural Designing

4.1 Exploring various alternatives in the design space

The use of multiple representations (provided by the Generator module in SLiDe), can be useful for designers to conceptualising, exploring and perceiving their designs differently. This helps in exploring the shapes in a design composition and allows designers to have a variety of representations of what has been designed, which may lead them to different discoveries to those they may otherwise have pursued. It also helps to focus a designer’s attention to potentially hidden visual features in such designs. This is achieved through the use of the Generator module in SLiDe as shown in Part I in Figure 1. The Generator constructs the design space by developing an infinite maximal lines representation of the initial design composition. The designer selects a shape of interest from among the intersections of these infinite maximal lines. The Generator searches the design space for shapes congruent with the selected shape. The congruent shapes are highlighted as shaded areas and the generator module develops a representation of these. Designers may continue exploring the design space for other alternative shape compositions by selecting other shapes from among the intersections of the infinite maximal lines representation. Consequently, the Generator module develops a corresponding representation. For instance, an initial representation of the design composition shown in Figure 2(a) is drawn by the designer (user) as shown in Figure 2(b). The infinite maximal lines of the initial representation is generated using the Generator module in SLiDe as shown in Figure 2(c). Examples of some of the representations developed using the Generator module are shown in Figures 3(a) to (f) showing the representations N_1 to N_6 from the initial representation of the design composition shown in Figure 2(b).

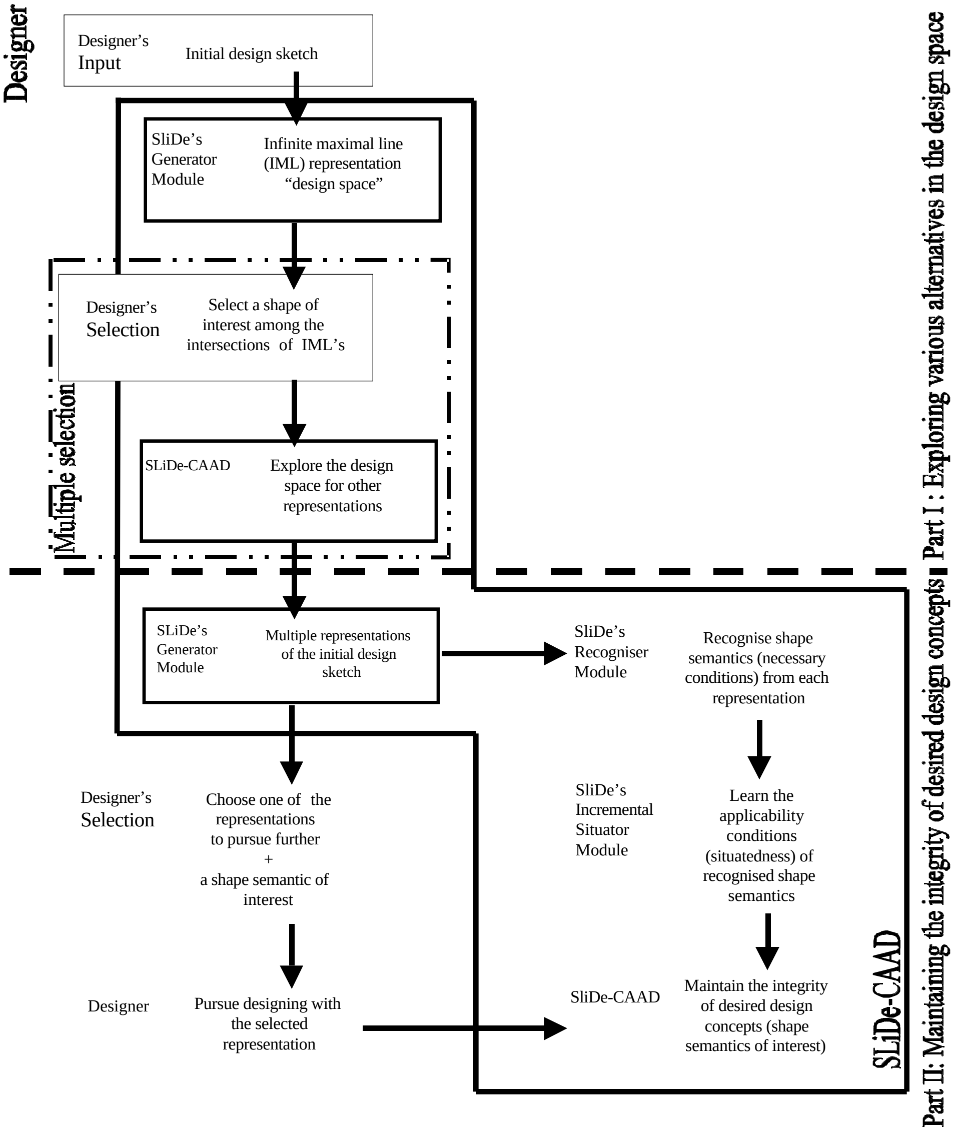

Designer
Designer's Input
Initial design sketch
SliDe's Generator Module
Infinite maximal line (IML) representation "design space"
Multiple selection
Designer's Selection
Select a shape of interest among the intersections of IML's
SLiDe-CAAD
Explore the design space for other representations
Part I : Exploring various alternatives in the design space
SLiDe's Generator Module
Multiple representations of the initial design sketch
SliDe's Recogniser Module
Recognise shape semantics (necessary conditions) from each representation
Designer's Selection
Choose one of the representations to pursue further + a shape semantic of interest
SliDe's Incremental Situator Module
Learn the applicability conditions (situatedness) of recognised shape semantics
Designer
Pursue designing with the selected representation
SliDe-CAAD
Maintain the integrity of desired design concepts (shape semantics of interest)
SLiDe-CAAD
Part II: Maintaining the integrity of desired design concepts

Figure 1*. Part I: A framework of exploring various alternative of shape compositions and Part II: A framework of maintaining the integrity of a shape semantic of interest.*

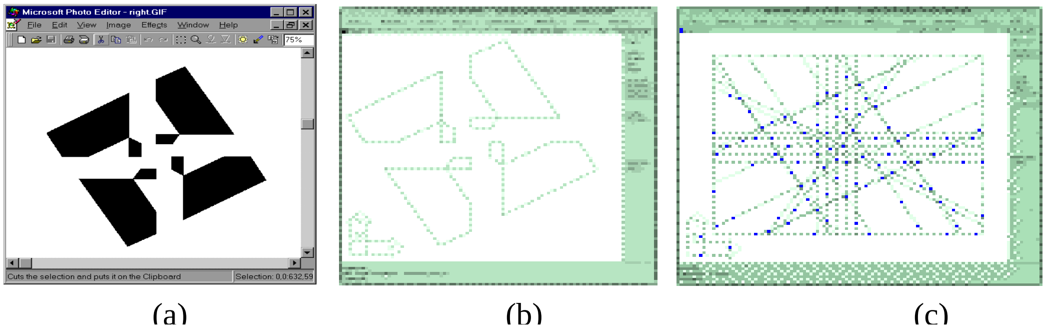

Microsoft Photo Editor - right.GIF
File Edit View Image Effects Window Help
75%
Cuts the selection and puts it on the Clipboard
Selection: 0,0:632,59
(a)
(b)
(c)

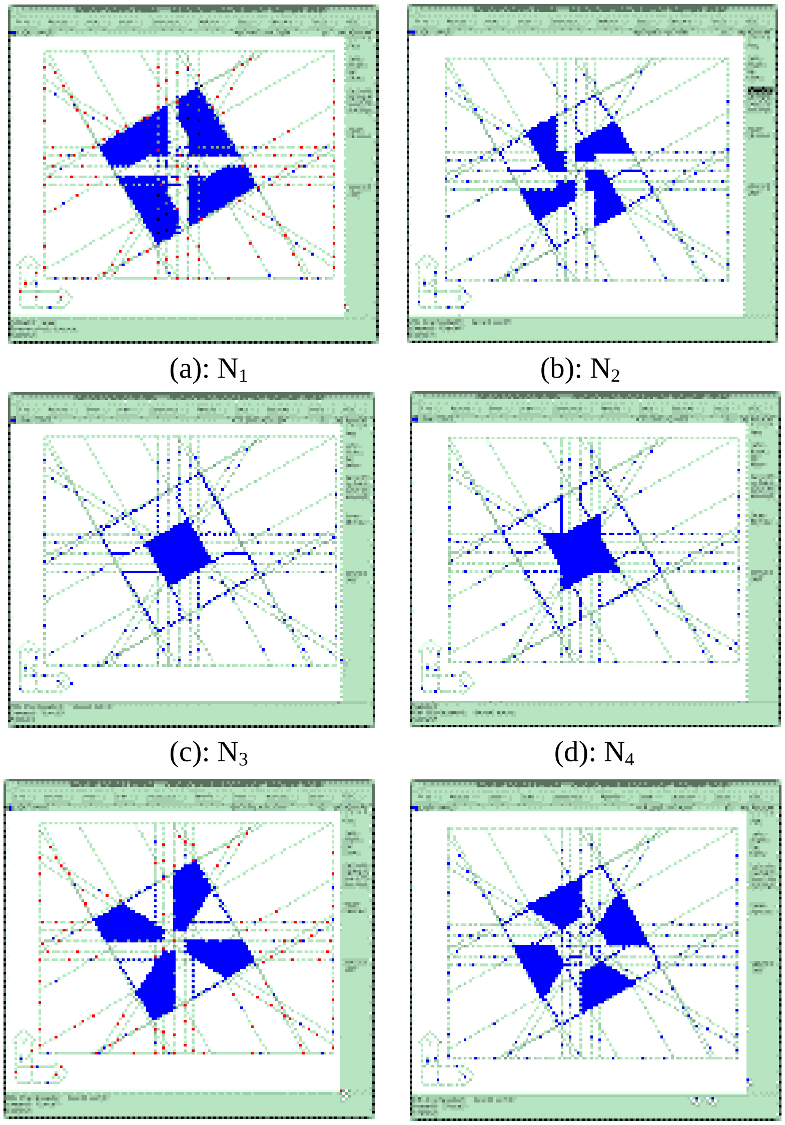
Figure 2*. (a) An example of a design composition; (b) an initial representation of the design composition; and (c) an infinite maximal lines of the initial representation produced using the Generator module.*

4.2 Maintaining the Integrity of Desired Design Concepts while Designing

The designer may select one of the developed representations to pursue further. The designer might be attracted to one or more of the recognised shape semantics in this representation. During conceptual designing, designers usually revise the selected alternative so changes take place in the design composition. It would be useful to provide designers with a tool that has the capacity to maintain particular concepts that have attracted them in their designs at earlier stages when other changes take place. Maintaining the integrity of desired and developed concepts while designing is another way in which SLiDe-CAAD could provide support in architectural designing. A simplified approach to maintaining the shape semantic of interest in the design composition has been proposed by Gero and Jun (1995). The shape semantic that is to be kept is constrained to exist independently of other operations based on the sufficient and necessary conditions of that shape semantic, ie situation independent. It is proposed here to maintain the desired shape semantic through preserving both its necessary (predetermined) conditions and applicability (constructed) conditions within which it was recognised across the constructed set of observations in SLiDe. This would help not only to maintain the shape semantic of interest but also its situatedness. Maintaining the situation within which the shape semantic of interest was recognised helps with maintaining the integrity among shapes in the design composition as a whole. This means that the desired shape semantic is constrained by both necessary and applicability conditions. This is achieved through maintaining the other shape semantics within which the shape semantic of interest is situated as outlined in Part II in Figure 1. The Recogniser module in SLiDe can be used to detect shape semantics in each of the developed representations shown in Figures 3(a) to (f). The result of using the Recogniser module is a set of observations constructed as shown in Table 1 from the developed representations. The Incremental Situator module in SLiDe can be used to learn the applicability conditions (situatedness), of recognised shape semantics across the observations constructed using the Recogniser module. The result of using the Incremental Situator module is a set of situational categories as shown in Figure 4, each of which includes the regularities of relationships among a group of shape semantics within which they were recognised. These situational categories present the applicability conditions of the recognised shape semantics.

Table 1*. A set of observations produced using the Recogniser module to detect shape semantics in the developed representations shown in Figures 3(a) to (f). $\boldsymbol{A_d}$ represents the semantic adjacency, $\boldsymbol{C_e}$ centrality and $\boldsymbol{R_n}$ cyclic rotation.*

Representation No.	Corresponding Observation (O_n)	
N_1	O_1	$\boldsymbol{R_n}$ **,** $\boldsymbol{A_d}$
N_2	O_2	$\boldsymbol{R_n}$ **,** $\boldsymbol{A_d}$
N_3	O_3	$\boldsymbol{R_n}$ **,** $\boldsymbol{C_e}$ **,** $\boldsymbol{A_d}$
N_4	O_4	$\boldsymbol{R_n}$ **,** $\boldsymbol{C_e}$ **,** $\boldsymbol{A_d}$
N_5	O_5	$\boldsymbol{R_n}$ **,** $\boldsymbol{A_d}$
N_6	O_6	$\boldsymbol{R_n}$ **,** $\boldsymbol{A_d}$



(a): N_1 (b): N_2

(c): N_3 (d): N_4

(e): N_5 (f): N_6

Figure 3*. Various alternatives of shape compositions produced while exploring the design space of the initial design composition using the Generator module, from (a) to (f) show the developed representations N_1 to N_6 from the original shape in Figure 2(a).*

Within each situational category, if a certain shape semantic is selected to be applied the remaining shape semantics within this category form the applicability conditions, ie situatedness, of that shape semantic. In Figure 4, there are two learned situational categories: $\boldsymbol{C_{s1}}$ and $\boldsymbol{C_{s2}}$. In $\boldsymbol{C_{s1}}$, there is a regularity among the shape semantics $\boldsymbol{R_n}$, $\boldsymbol{C_e}$ and $\boldsymbol{A_d}$ that refer to cyclic rotation, centrality and adjacency respectively. Within $\boldsymbol{C_{s1}}$, if cyclic rotation is selected to be applied, then both centrality and adjacency form the applicability conditions of cyclic rotation. In other words, cyclic rotation is situated within centrality and adjacency. In $\boldsymbol{C_{s2}}$, there is another situation where cyclic rotation is recognised in conjunction with adjacency.

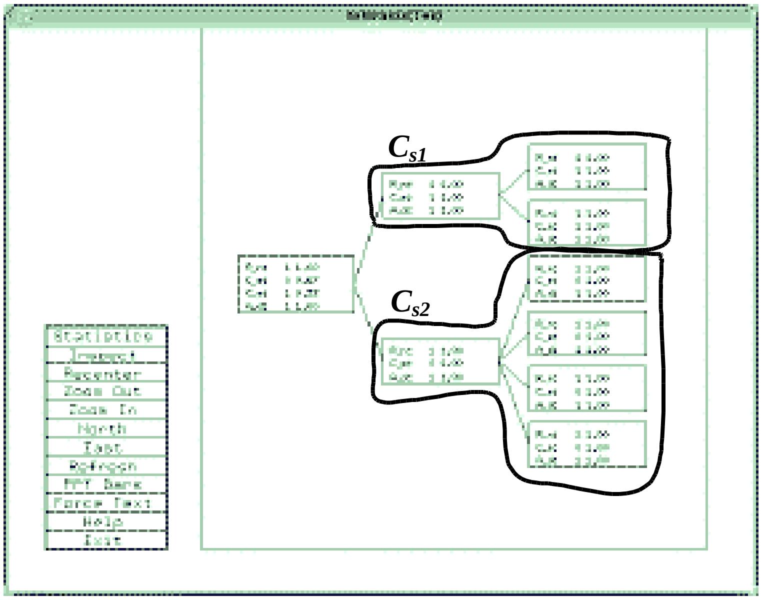
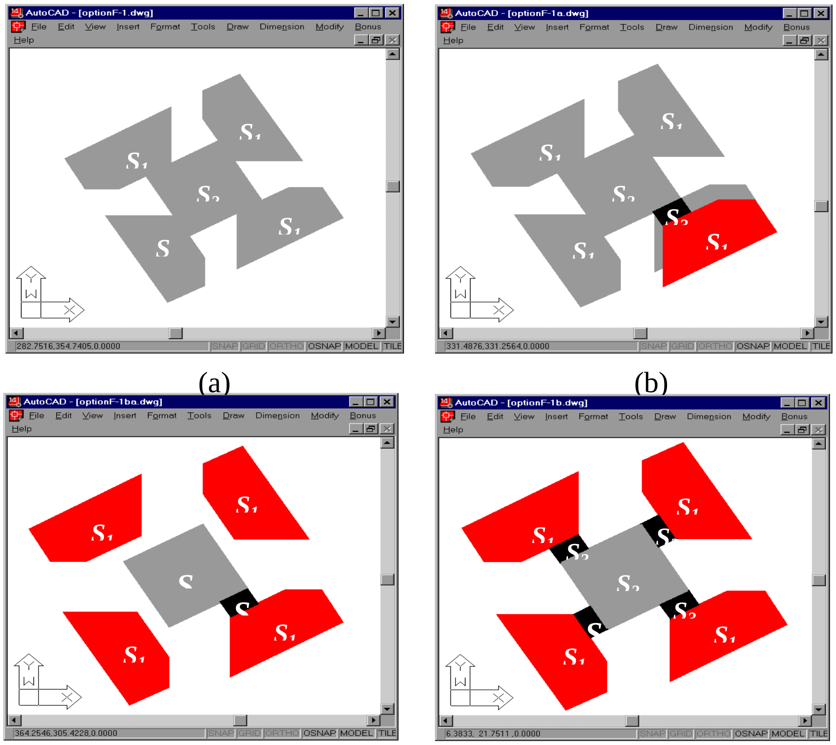


***Figure 4**. The result of using the Incremental Situator module in SLiDe to learn the applicability conditions of recognised shape semantics across the observations wherein two situational categories* $\mathbf{C_{s1}}$ *and* $\mathbf{C_{s2}}$ *are learned.*

These applicability conditions for each recognised shape semantic can be used to maintain its situatedness in the design composition as well as the integrity of the design composition while revising the design. SLiDe-CAAD can help to dynamically change the association between the parts in a design composition based upon the shape semantic of interest indicated by the designer by maintaining its applicability conditions. The selected shape semantic is considered by SLiDe-CAAD as the knowledge in focus or a desired design concept. So, whenever designers revise their designs, SLiDe-CAAD would automatically maintain the integrity of their desired design concepts through maintaining the related shape semantics that define the situation.

For example, assuming that the designer selected one of the new developed representations, say representation N_3 in Figure 3(c), to pursue. This selected representation becomes the current design composition that the designer acts on as shown in Figure 5(a). The designer indicated cyclic rotation ($\boldsymbol{R_n}$) among the group of four shapes $\boldsymbol{S_1}$ in the design composition as the shape semantic of interest. Some time later, the designer decided to add or insert a space, in the form of a new shape $\boldsymbol{S_3}$, between the two shapes $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$ in the design composition as shown in Figure 5(b). Such addition required moving the shape $\boldsymbol{S_1}$ from its previous location. Hence, the cyclic rotation ($\boldsymbol{R_n}$) among the group of shapes $\boldsymbol{S_1}$ is disturbed by moving one of the shapes $\boldsymbol{S_1}$ and changing its distance from the rotation centre of its group. Yet, there is a possibility to maintain the cyclic rotation by moving the other shapes $\boldsymbol{S_1}$ with the new distance from the rotation centre as shown in Figure 5(c). Maintaining the distance between each of the congruent shapes $\boldsymbol{S_1}$ and the rotation centre is one of the necessary and sufficient conditions of cyclic rotation. In spite of maintaining the cyclic rotation in the design composition, the adjacency ($\boldsymbol{A_d}$) between each of the shapes $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$ is disturbed. From the learned situational category ($\boldsymbol{C_{s1}}$) in SLiDe, adjacency ($\boldsymbol{A_d}$) is one of the applicability conditions of cyclic rotation. Since, cyclic rotation is the knowledge in focus, SLiDe-CAAD helps in maintaining the situatedness of cyclic rotation by preserving its applicability conditions. As a result, the shape $\boldsymbol{S_3}$ is inserted between each of the shapes $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$ to maintain the adjacency among them as shown in Figure 5(d). As can be seen from this example maintaining both the necessary and applicability conditions provides a rich support to maintain the integrity in a design composition as a whole. The necessary and sufficient conditions help in maintaining the shape semantics and the applicability conditions help in maintaining its situatedness.


AutoCAD - [optionF-1.dwg]
File Edit View Insert Format Tools Draw Dimension Modify Bonus Help
S₁
S₁
S₂
S₁
S
282.7516,354.7405,0.0000
SNAP GRID ORTHO OSNAP MODEL TILE
AutoCAD - [optionF-1a.dwg]
S₁
S₁
S₂
S₂
S₁
S₁
331.4876,331.2564,0.0000
AutoCAD - [optionF-1ba.dwg]
S₁
S₁
S
S
S₁
S₁
364.2546,305.4228,0.0000
AutoCAD - [optionF-1b.dwg]
S₁
S₂
S
S₁
S₂
S₂
S₂
S₁
S₁
6.3833, 21.7511 ,0.0000

(a) (b)

(c) (d)

Figure 5 *(a) One of the developed representations is selected by the designer to further pursue in designing; (b) a new space added by the designer at a later stage; (c) SLiDe-CAAD could help in maintaining the of cyclic rotation; and (d) SLiDe-CAAD could help in maintaining the situatedness of cyclic rotation.*

5. Conclusion

This paper outlined some features that SLiDe-CAAD can provide to conventional CAD systems that help both interactivity and designing support in the preliminary stages of designing. These features help in exploring the design space for various shape compositions and maintain the integrity of shape semantics of interest in the design composition and its situatedness. The purpose of SLiDe-CAAD is not to replace the designer, but to assist through a form of collaboration with the designer in designing and producing a solution. Considering the use of both necessary and applicability conditions to maintain the design integrity adds a dimension to parametric design; that is the situatedness of design knowledge. The main concern in parametric design is the imposition of constrained relationships, semantics, on the shape of objects which enables shape manipulation by adjusting several geometrical attributes in some fixed relation to each other, or in relation to explicit changes applied to other shapes, or the location of other objects (Kalay, 1989;

Rossignac, Borrel and Nackman, 1989). Such constraints are predetermined and the concern is to maintain certain semantic in the shape regardless of its interdependency with other semantics. Such interdependency can not be predefined but rather learned based on the situation within which that shape semantic is recognised. Furthermore, these interdependencies and relationships among shape semantics change based on the observations constructed from the various representations developed while designing. The Incremental Situator module in SLiDe can be used to capture the effect of such change in the situatedness of recognised shape semantics. These features provide the potential to change the nature of currently passive conventional CAD systems to be more active and responsive CAAD support system at the very early stages of designing.

References

[1] Gero, J. S. and Jun, H. **Visual semantic emergence to support creative design: A computational view**. In J. S. Gero, M. L. Maher and F. Sudweeks (eds), *Preprints Computational Models of Creative Design*, Key Centre of Design Computing, University of Sydney, Sydney, Australia, pp. 87-117, 1995.

[2] Jun, H. J. **Emergence of shape semantics of architectural drawings in CAAD systems**. Ph.D. Thesis, Department of Architectural and Design Science, University of Sydney, Sydney, 1997.

[3] Kalay, Y. E. **Modeling Objects and Environments**. Wiley, New York, 1989.

[4] Kolarevic, B. **Regulating lines and geometric relations as a framework for exploring shape, dimension and geometric organization in design**. In R. Junge (ed.), CAAD Futures 1997, Kluwer, Dordrecht, pp. 163-170, 1997.

[5] Koutamanis, A. and Mitossi, V. **Computer vision in architectural design**. Design Studies, 14(1): 40-57, 1993.

[6] Mitchell, W. J. and McCullough, M. **Digital Design Media**. Van Nostrand Reinhold, New York, 1995.

[7] Reffat, Rabee and Gero, John. **Computational situated learning in designing: Application to architectural shape semantics.** In J. S. Gero (ed.), *Artificial Intelligence in Design'00*, Kluwer, Dordrecht (to appear), 2000.

[8] Rossignac, J. R., Borrel, P. and Nackman, L. **Interactive design with sequences of parameterised transformations**. In V. Akman et al (eds), Intelligent CAD Systems II: Implementational Issues, Springer-Verlag, Berlin, pp. 92-125, 1989.

This is a copy of the paper: Reffat, R. and Gero, J. S. (2000) Towards active support systems for architectural designing, *in* D. Donath (ed.), *eCAADe18*, Bauhaus-Universitat, Weimar, pp. 143-147.