ARCHITECTURAL RESEARCH, Vol. 17, No. 2(June 2015). pp. 57-64 pISSN 1229-6163 eISSN 2383-5575

Exploratory Methodology for Acquiring Architectural Plans Based on Spatial Graph Similarity

Sungil Ham, Seongju Chang, Dongjun Suh and Amartuvshin Narangerel
Department of Civil & Environmental Eng, Korea Advanced Institute of Science and Technology, Seoul, Korea

http://dx.doi.org/10.5659/AIKAR.2015.17.2.57

Abstract In architectural planning, previous cases of similar spatial program provide important data for architectural design. Case-based reasoning (CBR) paradigm in the field of architectural design is closely related to the designing behavior of a planner who makes use of similar architectural designs and spatial programs in the past. In CBR, spatial graph can be constituted with most fundamental data, which can provide a method of searching spatial program by using visual graphs. This study developed a system for CBR that can analyze the similarity through graph comparison and search for buildings. This is an integrated system that is able to compare space similarity of different buildings and analyze their types, in addition to the analysis on a space within a single structure.

Keywords: Spatial Graph Similarity, Case-based Reasoning, Space Syntax

1. INTRODUCTION

In architectural planning, cases of similar spatial program play an important role as useful data in new program planning. CBR(Case-Based Reasoning) paradigm in the field of architectural design is closely related to the designing behavior of a planner who makes use of similar architectural design and spatial program in the past. Cases are particularly helpful to a planner as they can properly suggest deliberate information on similar condition in the construction planning stage. Information about the spatial graph and relation are the most fundamental data in CBR. This is because diagram that explains relation among spaces provides clues for solving a variety of programs in the process of substantiating space and shapes.

This study suggests a method for searching cases and inferring by using the similarity of spatial program, i.e., spatial graph. For the purpose of developing a CBR system based on the similarity among j-graphs which express the spatial phase relation, this study examines previous systems in the field of architectural design that used CBR as well as the cases' representation, search, and application used in these systems. Moreover, for expression and storage of spatial information, the study contemplates on the classification of the architectural spatial information under the limited condition of interior space. Finally, graph similarity theory and space syntax that derive similarity among the j-graphs were investigated. Based on these previous literatures, the paper implemented a program and drew the system results.

Corresponding Author: Sungil Ham,
Department of Civil & Environmental Eng, Korea Advanced Institute of Science and Technology, Seoul, Korea
e-mail: siham@kaist.ac.kr

This research was supported by a grant(14RERP-B090024-01) from Residential Environment Research Program funded by Ministry of Land, Infrastructure and Transport of Korean government.

2. THEORETICAL BACKGROUND

In this chapter, the theories about case-based reasoning, space syntax and graph similarity are reviewed for spatial analysis and searching.

(1) CBR(Case-Based Reasoning)

CBR refers to choosing a solution in the past that corresponds to the current demand or explaining a new situation referring to historical cases (Kolodner, 1993). In CBR process, similar cases in the past are searched first for a given new problem and they are subsequently applied and modified to the new problem in order to draw a new solution.

In CBR system, representation of design cases that are hard to structuralize is essential (Maher et al., 1995). The representation of design case is an expression of design situations such as its information and structure. Indexing and searching systems are required for finding proper design cases. These elements are the basis that constitutes the CBR system.

In the field of architectural design, diverse systems that use CBR process model have been developed so far. These systems were developed in a different manner regarding the representation of cases, searching, and application, according to researchers.

ARCHIE (Pearce et al., 1992) enables more specific and efficient search by representing cases and composing the searching system through descriptive expression of the planner about the desired case information in construction planning stage. SEED (Flemming, 1994) is an integrated software system where supports using computer are possible in the early stage of architectural design. CADRE (Bailey and Smith, 1994)is a system developed from an aspect of structure and design using geometric model. The system uses variables and conditions in the geometric model and it involves overall elements in design. Some researches (Anthony et al., 2001; Kitamura et al., 2002) are develop to share design repositories by using ontology. Other researches (Arciszewski et al., 1994; Bhatta and Goel, 1994; Sim and Duffy, 1998; 2004; Stahovich, 2000) are the use of machine learning techniques for main and sub tasks. These design case-based searching systems require analytical approach of information, representation of design, diverse search methods, and integrated design environment. The conditions for the design case-based searching system using CBR can be summarized as follows.

CBR system should be able to suggest proper cases even when the searched case does not fully satisfy the given conditions, which can be achieved by using partially satisfied condition or part of design information. Therefore, information of design cases requires appropriate representational system and indexing system (Kolodner, 1993).It should be composed such that search for diverse alternatives is possible until the retrieval of appropriate case to the given problem, through a representation of incomplete and complicated design problems in design cases based on process (Maher et al., 1995). It should be able to search appropriate cases through diverse search methods (Simon, 1996). For this, composition of accurate indexing system on diverse factors is required. It should possess integrated design environment that is used for design generation in order to apply and modify the searched cases (Fenves et al., 1995; Flemming, 1994). Accordingly, design case-based searching system should suggest retrieved design cases in a design environment that is easy to handle and familiar to the planner.

(2) Graph similarity

In addition to architectures, graphs are used in diverse fields including data modeling, Geographic Information System, chemistry, and phylogenetic tree of human evolution. In order to analyze the similarity of spatial structure between two architectural spaces, a methodology is used that can express the extent of similarity between phase graphs, which shows spatial relation, in numerical values. The analytical methods for spatial structure similarity have been developed from the graph theory to the statistical methodology. In graph theory, the most representative method of classifying graphs' similarity is isomorphism and homeomorphism methodology (H. Bunke, 1997). The methodology of maximum common sub-graph (Fernandez and Valiente, 2001) and minimum common super-graph (Horst Bunke et al., 2000) search for common joints and structures shared by the graphs. Although they can be applied to a problem of finding common genotypes, they have limitations in verifying the similarity of overall graph structure.

A methodology of statistic comparison (Stevens, 1957; Watts and Strogatz, 1998) refers to a method of statistically analyzing distance between each joint and connection relation in a network that have larger scale than the case of similarity. Representative skills include graph distance measure (H. Bunke, 1997) and iterative methodology (Jeh and Widom, 2001; Kleinberg, 1999; Melnik et al., 2002). All of these are proper for obtaining structural similarity of graphs. Iterative methodology, in particular, has a property of analyzing connection relation by giving weights to the neighboring joints, rather than simple distance between joints. Since architectural space has differing phase and characteristics according to the composition of neighboring spaces, it can be said that the iterative methodology which can incorporate such idea is more proper when analyzing similarity of architectural space. Heymans & Singh (2003) further developed the iterative methodology to invent bipartite methodology, which is an algorithm of evaluating similarity between species through similarity comparison of metabolic pathway graphs and writing the results in a new strain map. This study developed a new similarity algorithm of spatial structure based on the bipartite methodology. The method by Heymans & Singh (2003) is computed as follows. An algorithm that calculates similarity between two graphs of G1 and G2 follows four stages. First, similarity score of joints pair (a, b), where a $\in G_1$ and b$\in G_2$ is calculated through iterative process. Second, bipartite graph is composed using similarity score and optimal matching is found based on the weight of this bipartite graph. Third, similarity evaluation between every pair of the compared joints is recalculated. Finally, the similarity score of the graphs is calculated by summing up all the similarities of the compared joints and standardizing the value.

1) Computation of similarity score between two nodes

In two graphs of $G_1 = (V_1, E_1, \Lambda_1), |V_1| = n_1$ and $G_2 = (V_2, E_2, \Lambda_2), |V_2| = n_2$ G_1 and G_2 can be expressed as their neighboring matrix $A_1(n_1 \times n_1)$ and $A_2(n_2 \times n_2)$. $A_1(n_1 \times n_2)$'s similar matrix $S(a, b)$ is an expression of similarity between $a \in G_1$ and $b \in G_2$ can be obtained by infinite convergence through iterative rocedure, where the similarity between every pair of node (a, b) is imultaneously computed.

Similarity score between every pair of subjects expressed by nodes in two graphs are defined as . The similarity between every pair of node (a, b) in graph $G_1$and G_2 can be defined as a combination of similarity between the node and its neighboring node and the similarity between attributes of the node itself. $S(a, b)$, which is a similarity score between nodes, is initialized using $Sim(a, b)$ and they are simultaneously updated according to the mutual and iterative rule. If two are connected to similar nodes, then they are also similar to each other. The similarity between two nodes (a, b) can be computed by adding the similarity of connected nodes and then subtracting the difference.

$A_1 - A_4$ expresses the similarity between existence and non-existence of edge from similar nodes, while $D_1 - D_4$ expresses mismatch between edges. The term $A(a, b)$ expresses average similarity between a and b. More specific description about this term can be found in Heymans and Singh.

Similarity $S(a, b)$ is computed by fixed points. $S^0(a, b)$ is initialized into $Sim(a, b)$. $S^{(k+1)}(a, b)$ is then iteratively computed based on S^k. Here, the values are standardized after iterative computation since only the relative score is required.

Through this procedure, similarity matrix S between every pair of nodes in graph is obtained.

- Initialization

$$S^0(a,b) = Sim(a,b)$$

- Iterative stage

$$S^{(k+1)}(a,b) = \frac{A_1^k(a,b) + A_2^k(a,b) + A_3^k(a,b) + A_4^k(a,b) - D_1^k(a,b) - D_2^k(a,b) - D_3^k(a,b) - D_4^k(a,b)}{4} \times Sim(a,b)$$

$$A_1^{(k)}(a,b) = \begin{cases} \sum_{a_2 \to a, b_2 \to b} \frac{S^k(a_2,b_2)}{deg_{in}(a)deg_{in}(b)} & if\ deg_{in}(a) \neq 0\ and\ deg_{in}(b) \neq 0 \\ \sum_{a_2 \in G_1, b_2 \in G_2} \frac{S^k(a_2,b_2)}{n_1 \times n_2} & if\ deg_{in}(a) = deg_{in}(b) = 0 \\ & 0 \end{cases}$$

$$D_1^{(k)}(a,b) = \begin{cases} \sum_{a_2 \to a, b_2 \to b} \frac{S^k(a_2,b_2)}{deg_{in}(a)(n_2 - deg_{in}(b))} & if\ deg_{in}(a) \neq 0\ and\ deg_{out}(b) \neq n_2 \\ \sum_{a_2 \in G_1, b_2 \in G_2} \frac{S^k(a_2,b_2)}{n_1 \times n_2} & if\ (deg_{in}(a)) = (n_2 - deg_{in}(b)) = 0 \\ & 0 \end{cases}$$

- Standardization

$$S \leftarrow \frac{S}{\parallel S \parallel_2}$$

2) Bipartite graph matching

In second stage, the previously obtained similarity is used to find the optimal matching between graphs. After generating bipartite graph, mutual graph matching algorithm is performed. Once the G_1's set V_1 and G_2's set V_2 are obtained, bipartite graph $G = (V_1, V_2, S)$ containing similarity matrix S can be generated.

When this bipartite graph is generated, optimal graph matching can be found using Hungarian algorithm of $O((n_1+n_2)^3)$. Once the optimal matching is found, matrix $M(a,b)$ can be obtained based on it. Matrix M is a $n_1 \times n_2$ boolean matrix consisting of only 0 and 1 that takes value 1 if node a and b matches and takes value 0 otherwise.

3) Computation of similarity score between matched nodes

Similarity score can be obtained through the optimal matching that was previously derived from graph G_1 and G_2. Like the first stage, node's similarity and structural similarity should be combined to compute the similarity. Similar equations of $A_1 - A_4$ and $D_1 - D_4$ are iterated. The new sets of equation $A_1' - A_4'$ and $D_1' - D_4'$ follows the technique in previous stage, except of their usage of $M(a,b)$ instead of $Sim(a,b)$. In this stage, the value of $A_1' - A_4'$ and $D_1' - D_4'$ are standardized by squared root. This is because the maximum size of matching is smaller than the graph. Terms $A_1' - A_4'$ and $D_1' - D_4'$ sum up the similarity of optimal matching between G_1 and G_2 and its

- Iterative stage

$$S(a,b) = \frac{A_1'(a,b) + A_2'(a,b) + A_3'(a,b) + A_4'(a,b) - D_1'(a,b) + D_2'(a,b) + D_3'(a,b) + D_4'(a,b)}{4} \times Sim(a,b)$$

$$A_1'(a,b) = \begin{cases} \sum_{a_2 \to a, b_2 \to b} \frac{M(a_2,b_2)}{\sqrt{deg_{in}(a)deg_{in}(b)}} & if\ deg_{in}(a) \neq 0\ and\ deg_{in}(b) \neq 0 \\ \sum_{a_2 \in G_1, b_2 \in G_2} \frac{M(a_2,b_2)}{\sqrt{n_1 \times n_2}} & if\ deg_{in}(a) = deg_{in}(b) = 0 \\ 0 & otherwise \end{cases}$$

$$D_1'(a,b) = \begin{cases} \sum_{a_2 \to a, b_2 \to b} \frac{M^k(a_2,b_2)}{\sqrt{(deg_{in}(a)) \cdot (n_2 - deg_{in}(b))}} & if\ deg_{in}(a) \neq 0\ and\ deg_{in}(b) \neq n_2 \\ \sum_{a_2 \in G_1, b_2 \in G_2} \frac{M(a_2,b_2)}{\sqrt{n_1 \times n_2}} & if\ deg_{in}(a) = n_2 - deg_{in}(b) = 0 \\ 0 & otherwise \end{cases}$$

dissimilarity. Finally, $S(a,b)$ is calculated by summing up each term and multiply similarity value of each node.

4) Computation of similarity score

Finally, in order to obtain similarity score of the two graphs, all of the previously calculated similarity values between nodes are summed up and then standardized by dividing it by squared root of the product between the number of node in and . If this takes a value of 1, two graphs will become identical.

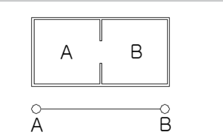
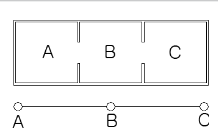
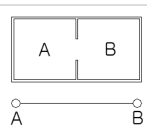
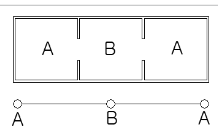
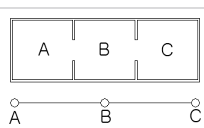
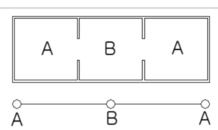
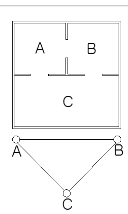
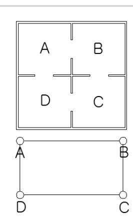
- Similarity score

$$S_{G_1,G_2}(a,b) = \frac{\sum_{a_2\in G_1, b_2\in G_2, M(a,b)=1} S(a,b)}{\sqrt{n1 \times n2}}$$

5) Analytical example

Table 1 shows an example of expressing normal spatial relationship in graph and analyzing it using bivariate algorithm.

Table 1. Similarity score of examples

Graph1	Graph2	Similarity Score
		0.427
		0.499
		0.375
		0.5843

(3) Space syntax

Space syntax is a method of analysis for spatial layouts in buildings or cities using morphology and topology. The spatial configuration of a building is defined by analyzing the connection or separation between its spaces from the perspective of passage through openings and visual perception. Bill Hillier and Julienne Hanson of University College London suggested this methodology in the early 1970s. The philosophy of the method attempts to develop a conceptual model to examine the relations between society and space from the perspective of transformation of social elements into spatial forms, and transformation of spatial elements into social forms. This shift in perspective in turn helps to develop a fundamental tool to quantitatively analyze spatial structures (Hillier, 1998). Several major concepts are discussed below.

The connectivity of a space is defined by the number of spaces that are directly accessible from a specific space. If the number is great, then a space is connected to a lot of other spaces, being the center of flow. The connectivity of a space can be calculated locally or globally in consideration of the entire spatial network. The latter value of connectivity is called the real relative asymmetry(RRA). To calculate connectivity, the mean depth (MD) for a target space needs to be calculated first. MD_k for a node(space) k is the average depth from node k to all the other nodes. The relative asymmetry(RA) value is the "mean depth(MD) expressed as a fraction of the maximum possible range of depth values for any node in a graph with the same number of nodes as the system" (Bafna, 2003). The equations for calculating MD and RA are listed below (Hillier and Hanson, 1984):

$$MD = \frac{\sum_{i=0}^{i=n} d_{i,k}}{n-1}$$

$$RA = \frac{2(MD-1)}{n-2}$$

Where RA is the relative asymmetry; MD is the mean depth of node *k; di, k* is the depth between i^{th} node and node *k* (i.e., the number of nodes, or spaces, between two nodes *i* and *k*) and *n* is the total number of nodes(spaces).

The RRA value is the value of the RA divided by the adjusted standard deviation, called the *D* value(*Dn*). The D value is the RA of a diamond-shaped (i.e., fully symmetric) network (Hillier and Hanson, 1984). Thus the RRA is a "ratio of the nodes of the given system and the RA of the central node of a diamond graph with the same number (*n*) of nodes as the system" (Bafna, 2003).

$$RRA = \frac{RA}{D_n}$$

$$RRA = \frac{6.64n \times \log(n+2) - 5.17n + 2}{(n-1)\times(n-2)}$$

Where is the D_n value of a network composed of *n* number of spaces (nodes); and *n* is the number of spaces.

The integration of a space represents the approachability of a space from other spaces, or vice versa (Hillier and Hanson, 1984). The integration value is the inverse of the RRA value. The smaller the integration value, the shallower the depth of a spatial network and the more integrated spaces are. On the other hand, the greater the value, the greater the segregation of space is:

$$\text{Integration Value} = \frac{1}{RRA}$$

A number of different forms of graphs have been developed to describe spaces and the relations between spaces depending on the analysis targets or objectives. The following section describes various types of spatial network models using an example.

3. SYSTEM IMPLEMENTATION

In previous chapter, system conditions, algorithm of the spatial analysis and search using CBR through theoretical consideration were researched. Following the analysis procedure, implementation and new algorithm of the program are examined in this chapter based on the previous algorithm and theory.

(1) Procedure for spatial analysis and search

For spatial search, buildings' attribute values should be first put into the system as fundamental data. Next, space relation is presented in a diagram using spatial graph. For obtaining similarity that provides basic data for searching buildings with similar cases, space syntax algorithm was used to compute integration. Finally, the spatial structures of the buildings are compared using each sample space's integration and graph's similarity

1) Input of building attributes

As fundamental data for searching buildings, the name of the building, charge of designing, location, zoning district, lot area, total floor area, size, structure, use, etc. are put into the system.

2) Expression of spatial composition

Examples of apartments were used for system verification in this study. Using Uniclass (Royal Inst. of British Architects, 1997) as a standard for space classification, space was classified into two categories(spaces and circulation Spaces) as is shown in Table 2. The types for space connection were divided into 3 types as is shown in Table 3. A virtual connector represents a relation that does not physically divide spaces, but does conceptually divides spaces by usage or meaning of the space. For spatial search with other use or types, additional space classification will be required.

Table 2. Classification and icons of spatial information by uniclass

classification	Content	Icon
Spaces	45 Residential spaces 45-10 Long-term residential spaces 45-10-07 Bath rooms: BA 45-10-09 Bedrooms: B 45-10-44 Kitchen/ dining rooms: KD 45-10-49 Living rooms: L 45-10-94 Verandas: V 45-10-96 Walk-in wardrobes: W	
Circulation Spaces	90 General spaces 90-10 General circulation spaces 65-10-36 Hallways: H 65-10-64 Porches: P 65-10-94 Vestibules: VE	

Table 3. Classification and icons of connection type between spaces

Connection types	Icons
Door	
Virtual Connector	
Sliding Door	

The connection between walk-in wardrobe and bedroom can be expressed in icon as follows.

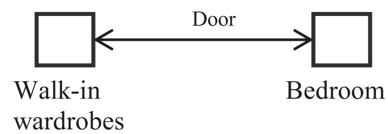


Figure 1. Example of space connection

3) Similarity analysis

For similarity analysis of architectural spaces, similarity is computed using the previously examined graph comparison similarity and space syntax's integration value.

As for the graph comparison similarity, graph's similarity is analyzed through pairwise comparison of node and edge. Here, the coincidence of nodes that indicate space has significant meaning. However, space in buildings has different characteristics depending not only on its use, but also on the relation with the whole structure. The integration of space that expresses such relation has influence on the recognition and space use among people, resulting in different perception about the space. Hence, in an analysis on the similarity of architectural space, the weight of the graph comparison similarity and the syntax integration value can be determined by users.

This paper made use of weights and assumption as follows for the case analysis.

- Assumption 1

Graph comparison similarity and space syntax's integration had identical weight on node.

- Assumption 2

The Relation of integration and graph comparison similarities

$$M_i(a,b) = 0.25/(1 + \frac{|I_a + I_b|}{2})$$

(I_a means the integration of a node)

$$A'_1(a,b) = \begin{cases} 0.5\ if\ a\ and\ b\ are\ mateched \\ 0\ otherwise \end{cases}$$

Matching values of node were assumed as in Table 4, Table 5 below.

Table 4. Matching value according to node and edge

Cases	Values
Edge and two nodes are not identical	0
Only edges are identical	0.05
Only one node is identical	0.2
One node and edge are identical	0.3
Only two nodes are identical	0.45
Two nodes and edge are identical	0.5

Table 5. Matching value without edge

Cases	Two nodes are different	Only one node is identical	Two nodes are identical
Value	0	0.25	0.5

If there is an upper group, half of its value was applied to the lower group.

Matching value considering graph comparison similarity and integration of each node.

$$M(a,b) = M_g(a,b) + M_i(a,b)$$

If all spaces are identical, $M_g(a,b)$ and $M_i(a,b)$ are 0.5. Therefore $M(a,b)$ is 1.

Such assumption on the relation between integration and similarity and relation between graph's similarity and spatial integration value requires further investigation in future studies. Specifically, more studies investigating the relation between weight and the user's case selection are required for obtaining optimized relation. This paper has significance in that it showed that similarity can be obtained by using graph comparison similarity and integration value of space syntax and that it confirmed its potential use as data for searching based on such similarities.

(2) Program implementation for searching plan

For the spatial analysis and search using the case reasoning proposed in this study, we implemented a single integrated system based on the previously examined theory.

The space syntax's algorithm was used for the spatial analysis, while graph comparison algorithm and Pwcomp(Adelman, 2003)'s program package were used for the spatial search. A package of Image J(Abràmoff et al., 2004) was used for automatic selection of space area in floor plan image and Eclipse was used as a development tool. Table 6 shows development environment

Although the analyzed data were implemented in a single application as of now, they will be stored in database and be developed as a tool for analysis and search on the Web.

Table 6. Development environment

Classification	Content
Language	Java
Environment	Java 1.8
Package	Image J, Pwcomp,
Algorithm	Space Syntax, Graph Comparison
IDE	Eclipse

The system was implemented as is shown in Figure 2. The floor plan information is brought to the task pad and icon in each room that represents space is selected to be allocated on the task pad, after which the relation is determined according to space connection types. Once the allocation and relation of the space are determined, previous cases can be searched in high-to-low order of similarity based on the spatial analysis and similarity analysis results. When the cursor clicks the icon of the space, diverse information about the space such as image and size are provided, which enhances the understanding about the case at a planning stage.

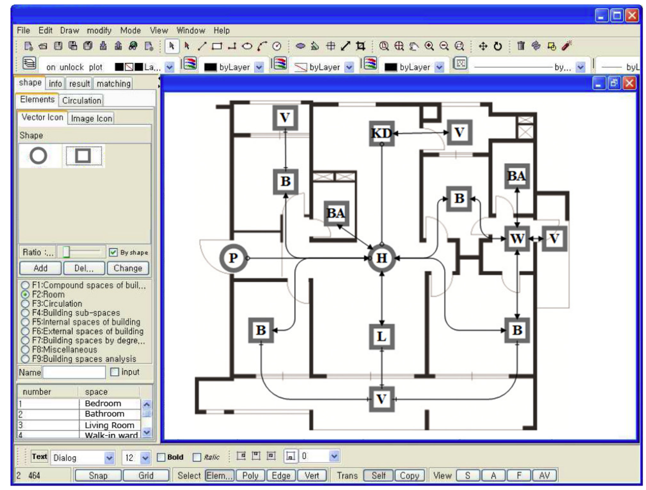


Figure 2. Architectural plan searching system

(2) Case study using searching system

Next, relative similarity was analyzed using six apartment units. In the case below, Ssangyong Sangdong of 171m^2 in Figure 3. Examples of architectural plans showed the most similar spatial composition with 2nd Daerim Seocho 132m^2 showing a similarity as high as 0.944.

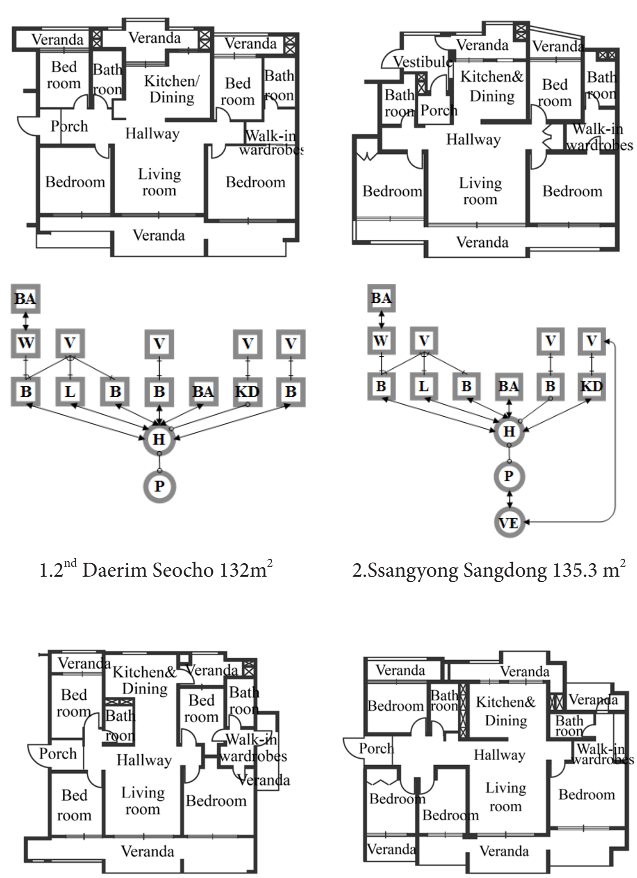


Figure 3. Examples of architectural plans

Table 7. Scores of architectural Examples

	1	2	3	4	5	6
1	1	0.837	0.875	0.819	0.656	0.928
2	0.837	1	0.760	0.765	0.726	0.838
3	0.875	0.760	1	0.879	0.601	0.800
4	0.819	0.765	0.879	1	0.529	0.731
5	0.928	0.838	0.800	0.731	1	0.667
6	0.957	0.788	0.828	0.777	0.667	1

4. CONCLUSION

This study developed a system for CBR that can search buildings through similarity of graphs that express the relation and phase of space. It is an integrated system that is able to compare similarity of space with other buildings and to analyze the types, which is an advance from the previous studies that analyzed space within a single structure.

The results of this study can be summarized as follows.

Firstly, the study investigated a methodology that analyzes the similarity of spatial program through graph comparison. This similarity has significance as fundamental search data for case reasoning.

Secondly, in comparison analysis of space within a context that includes other buildings, similarity among the buildings was analyzed different from conventional spatial analysis on an individual structure, which opens a door for methodology that can analyze and evaluate the developmental process of the space, in addition to the search of the space itself.

Lastly, in the study, it was possible to deliver spatial relation in a condensed form of visual information using spatial graph and its potential as a mediating instrument for creating realized structure in architectural design procedure was investigated. Finally, the study built a CAD-based and object-oriented program that can be used in diverse spatial analysis.

However, for obtaining the similarity for CRB search, more studies are required in the future that can examine the similarity relation with other search conditions such as the location of the space, in addition to the graph's structural similarity.

REFERENCES

Abràmoff, M. D., P. J. Magalhães, and S. J. Ram (2004), Image processing with ImageJ, Biophotonics international, 11(7), 36-43.

Adelman, J. L. (2003), PWCOMP, edited.

Anthony, L., W. C. Regli, J. E. John, and S. V. Lombeyda (2001), An approach to capturing structure, behavior, and function of artifacts in computer-aided design, Journal of Computing and Information Science in Engineering, 1(2), 186-192.

Arciszewski, T., E. Bloedorn, R. S. Michalski, M. Mustafa, and J. Wnek (1994), Machine learning of design rules: methodology and case study, Journal of Computing in Civil Engineering, 8(3), 286-308.

Bafna, S. (2003), Space Syntax, Environment and Behavior, 35(1), 17-29.

Bailey, S. F., and I. F. Smith (1994), Case-based preliminary building design, Journal of Computing in Civil Engineering, 8(4), 454-468.

Bhatta, S. R., and A. K. Goel (1994), Discovery of physical principles from design experiences, Artificial Intelligence for Engineering, Design, Analysis and Manufacturing, 8(02), 113-123.

Bunke, H. (1997), On a relation between graph edit distance and maximum common subgraph, Pattern Recogn. Lett., 18(9), 689-694.

Bunke, H., X. Jiang, and A. Kandel (2000), On the Minimum Common Supergraph of Two Graphs, Computing, 65(1), 13-25.

Fenves, S. J., J. Garrett, H. Kiliccote, K. Law, and K. Reed (1995), Computer representations of design standards and building codes: US perspective, The International Journal of Construction Information Technology, 3(1), 13-34.

Fernandez, M.-L., and G. Valiente (2001), A graph distance metric combining maximum common subgraph and minimum common supergraph, Pattern Recogn. Lett., 22(6-7), 753-758.

Flemming, U. (1994), Case-based design in the SEED system, Automation in Construction, 3(2), 123-133.

Hillier, B. (1998), Space is the machine: a configurational theory of architecture, Cambridge University Press, Cambridge.

Hillier, B., and J. Hanson (1984), The social logic of space, Cambridge University Press, Cambridge.

Jeh, G., and J. Widom (2001), SimRank: A Measure of Structural-Context Similarity, Technical Report Rep., Stanford InfoLab.

Kitamura, Y., T. Sano, K. Namba, and R. Mizoguchi (2002), A functional concept ontology and its application to automatic

identification of functional structures, Advanced Engineering Informatics, 16(2), 145-163.
Kleinberg, J. M. (1999), Authoritative sources in a hyperlinked environment, J. ACM, 46(5), 604-632.
Kolodner, J. L. (1993), Case-based reasoning, Morgan Kaufmann.
Maher, M. L., M. Balachandran, and D. M. Zhang (1995), Case-based reasoning in design, Psychology Press.
Melnik, S., H. Garcia-Molina, and E. Rahm (2002), Similarity Flooding: A Versatile Graph Matching Algorithm and its Application to Schema Matching, in 18th International Conference on Data Engineering (ICDE 2002), edited, San Jose, CA.
Pearce, M., A. K. Goel, J. L. Kolodner, C. Zimring, L. Sentosa, and R. Billington (1992), Case-based design support: A case study in architectural design, IEEE Expert, 7(5), 14-20.
Royal Inst. of British Architects, L. (1997), Uniclass Unified classification for the construction industry.
Sim, S. K., and A. H. Duffy (1998), A foundation for machine learning in design, AI EDAM, 12(02), 193-209.
Sim, S. K., and A. H. Duffy (2004), Evolving a model of learning in design, Research in Engineering Design, 15(1), 40-61.
Simon, H. A. (1996), The sciences of the artificial, MIT press.
Stahovich, T. F. (2000), LearnIT: an instance-based approach to learning and reusing design strategies, Journal of Mechanical Design, 122(3), 249-256.
Stevens, S. S. (1957), On the psychophysical law, Psychological Review, 64(3), 153-181.
Watts, D. J., and S. H. Strogatz (1998), Collective dynamics of `small-world' networks, Nature, 393(6684), 440-442.

(Received April 7, 2015/Accepted June 17, 2015)