Graph modeling of a floor plan with isovist graph

Hideyoshi Odawara and Atsushi Takizawa*

Graduate School of Human Life and Ecology, Osaka Metropolitan University, Japan
*Corresponding author: takizawa@omu.ac.jp

Abstract

Space Syntax has been the subject of considerable research regarding the geometry and topology of spaces. However, image information such as texture and color is outside the scope of Space Syntax. However, deep learning methods for images and graphs are evolving, and combining these methods is expected to improve spatial analysis. As images change depending on where they are captured, it is important to know where they are captured. Especially in spaces with few partitions or irregular systems, it can be difficult to define the center of the space. In this study, we define an isovist graph as "a spatial model that covers a closed plane with a minimum or near-minimum number of isovist while ensuring centrality and connectivity," and propose a method for rigorously obtaining such a model in a discrete problem setting. This study demonstrated that an exact solution can be obtained quickly for many of the tested spatial models in accordance with the intention.

Keywords

Spatial analysis, floor plan, isovist, cover problem, mathematical programing, lazy constraint

Introduction

Various studies have been conducted based on basic space unitization methods such as convex space, axial maps, and isovists (Benedikt, 1979) in Space Syntax (Hillier and Hanson, 1984) for spaces with concrete shapes on the scale of buildings and city blocks. Various studies have been conducted using these methods. However, image information such as texture, light, and color, which are visually more primitive, is outside the scope of space syntax. Even if the spatial composition, size, and shape are identical, the impression of a space can change significantly depending on the materials and lighting. Therefore, image-based information is essential for various types of spatial decision-making.

From the 1980s to the 2000s, when the Space Syntax method was developed and applied, the difficulty of obtaining image information as data and analyzing it seemed to be in the background. In recent years, omnidirectional cameras have become widespread, and the number of stock photos captured from various streets as street views has increased. With the recent development of deep learning technology, it has become easier to extract meaningful features from high-dimensional data in the form of images and to transform and compress images. In the field of urban analysis, the semantic segmentation of street-view images into basic spatial components, such as sky, buildings, and trees, is

widely used to analyze their areas and spatial phenomena (e.g., walkability) (Koo et al., 2022). In addition to streets, the stock of captured indoor images is also increasing, providing services such as virtual tours, and spreading to areas such as real estate (Matterport 2025; RICOH, 2025). In addition, a deep learning technique called a graph convolutional network has been developed to perform tasks such as regression and classification on the graph itself, which has multidimensional features such as nodes. Research has applied this technology to the analysis of houses (Takizawa, 2024) and streets, and the groundwork has been laid for analyzing complex graph structures in different ways.

An omnidirectional image is similar to a 3D isovist image; however, instead of depth information, it can be considered to be accompanied by general image information (Takizawa and Kinugawa, 2020). Since the information in an image changes depending on where the image is taken, it is important to know where to take the image. Usually, images are captured at the center of a space with clear partitions, such as rooms; however, in a space with few partitions or an irregular system, the boundaries of the space may become unclear, making it difficult to define the center of the space. In addition, because a virtual tour enables the continuous navigation of a space by connecting the points where images are captured, it is necessary to consider connectivity. However, when creating a 3D model using camera position estimation and photogrammetry, it is easy to specify the shooting points relatively closely within the same space. However, because the main purpose of this research is to describe the structure of a space, it is preferable to have a minimum or close-to-minimum number of shooting points at a location (center) that is as representative of the space as possible.

Based on this background and awareness of the problem, this study defines an isovist graph as "a spatial model that covers a closed plane with a minimum or near-minimum number of isovist while ensuring centrality and connectivity," and proposes a rigorous method to obtain it in a discrete problem setting. We propose a method for obtaining this rigorously in a discrete problem setting. The proposed method was applied to multiple floor plans ranging from the architectural scale to the city block scale, and the solution time and obtained graphs were verified.

In the following chapters, Section 2 describes the previous studies and the position of this research, and Section 3 presents the problem framework. Section 4 reports the concrete flow and formulation of the method. Section 5 verifies the method, Section 6 discusses the results, and Section 7 summarizes the study.

Positioning of this study in comparison with related studies

The positioning of this study was clarified in terms of the spatial model and solution method.

First, we examined the position of this study in terms of the spatial model. The concept of our isovist graph shares similarity with the function finding minimal set of isovists implemented in the Isovist_App (McElhinney, 2024). While it can be said that the method is pioneering approach, it does not consider the centrality of isovist unlike our study. Then, our isovist graph has many essential overlaps with the definition of an axial map (Hiller and Hanson 1984, P91-92). They defined an axial map as "*An axial map of the open space structure of the settlement will be the least set of such straight lines which passes through each convex space and makes all axial links...*" The idea of covering a plane with a finite number of connected unit spaces is common. However, axial maps are somewhat confusing concepts as entities. As highlighted by Batty and Rana (2004), axes are interpreted as coherent straight lines that move unimpeded in space; however, these are spatial units similar to aggregated road edges, making naive

interpretation difficult. Therefore, when analyzing an axial map as a graph, the axes given as edges must be dual-transformed into nodes. In ordinary spatial analysis, various states at arbitrary observation points are observed and weighed. However, such a correspondence cannot be obtained for nodes given by a dual transformation, and measurement at the center of the line segment or an averaging process is required. However, an isovist is a simpler spatial unit than an axial map because it covers the entire area visible from a single point in all directions. On the other hand, our study differs from the present study in that it uses isovists to generate the axial map described below and does not consider the centrality of isovists or the connection relationship between vantage points. Visibility graph analysis (VGA; Turner et al., 2001) is a well-known graph model based on the connection between an isovist's vantage points. This method generates isovists on a grid, creates a visibility graph of them, calculates various graph features such as a clustered coefficient, and visualizes the features as an isovist field on the original raster. VGA is suitable for expressing the differences in detailed features in a space because it expresses the space as a detailed field. To simplify the visibility graph using VGA, a method of thinning by maximizing the modularity of the graph (Hwang, 2013) has been proposed, but it is difficult to obtain a complete coverage of the space by the thinned representative points, and it is difficult to obtain a complete coverage of the space by the thinned representative points. The covering of the space is not guaranteed to be complete or to come to the center of the space; although different from the VGA, there are studies that generate isovist-like graphs to move as a method of indoor navigation (Hamzei et al., 2024). In these studies, the main focus was on observing walls to explore the space, and the graphs were not generated in the interest of structuring them with a small number of points that are representative of the space.

The next section describes the objective of this study in terms of the solution method. First, the function of Isovist_App uses a greedy algorithm to find the minimum isovist graph. However, the number of vantage points obtained is not guaranteed to be theoretically minimal. Next, we focus on the automatic generation of axial maps. As mentioned earlier, it was claimed that an axial map can be derived by leaving axial lines until all convex spaces are traversed and all axial lines that can be connected to other axial lines are connected without repetition. The method for deriving a linear representation of a spatial configuration relies on the prior definition of a convex partition of space. The problem of partitioning the minimum convex polygon of a polygon involves efficient algorithms for simple polygons without holes (e.g., Chazelle and Dobkin, 1985). However, it is NP-hard in the case of holes, which are common in space (Lingas ,1982). Moreover, even given the optimal convex partitions, the problem of covering these convex polygons with the minimum number of axial lines is NP-hard because it can be attributed to the set cover problem (Karp, 1972). In addition, from the viewpoint of mathematical programming, the axial map must be connected as a graph that adds cut and flow constraints, thereby making the solution more difficult. Despite these mathematical difficulties, there is a growing need for an objective method for generating axial maps, and Peponis et al. (1998), Batty and Rana (2004), and Turner (2005) studied its automatic generation. First, Peponis et al. divided space by s-partition (Peponis et al. 1997), which divides space by wall extensions instead of convex regions, and then enumerated m-lines, which represent the diagonals of other wall surfaces, and ordered m-lines, in the order in which m-lines intersect more s-partitions. Tuner et al. improved Peponis' method and proposed a heuristic called subset elimination to retain the necessary lines of sight. Batty and Rana divided the space into grids, found isovists at each point, and used some of them to cover the space. They then generate axial lines in these isovists according to criteria such as maximum length and repeat the process until there are no uncovered areas. This method gives the user the

freedom to switch between multiple criteria and change the selection of isovists and axial lines; however, it does not guarantee the connectivity of the axial lines.

All the above algorithms were heuristic and did not guarantee the minimum number of necessary axial links in their problem setting. Jung and Kim (2020) formulated Peponis' problem of finding an axial map using integer programming (without considering cyclicity) and showed that an exact solution can be found. In their formulation, they defined the constraint that axial lines are connected to each other; however, the only connection constraint is that an axial line is connected to one or more other axial lines, such that, for example, if there is a set of four axial lines $\{A, B, C, D\}$ the resulting graph is allowed to separate into multiple connected components, such as $\{A, B\}, \{C, D\}$, and so on. In general, the connectivity patterns of a graph are in the order of the number of fingers of the graph elements, making it difficult to explicitly formulate the connectivity constraints in terms of mathematical programming constraints, which require a complex formulation. One approach is to use network flows as constraints that are formulated to guarantee geographic connectivity, as in Shiode (2005). Another study by Dehbi et al. (2021) used connectivity constraints to generate indoor navigation graphs. Although these graphs are guaranteed to be single-connected graphs, the complexity of their constraints makes it difficult to apply them to large problems. In this study, we demonstrated that the lazy-constraint method can be used to solve this problem more efficiently.

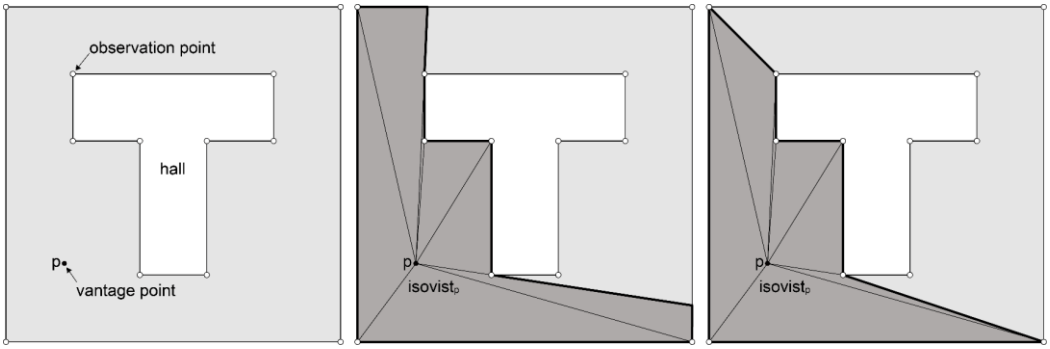
In summary, compared to previous studies, this study proposes a new model and problem setting, namely the isovist cover problem, considering centrality and connectivity. Furthermore, the solution method differs from related studies in that it devises an efficient and exact solution method that can be applied to axial map-solving, which is highly novel.

Problem framework

In this study, we assume that the class of polygons representing a connected plane is either simple or multiple connected polygons. We consider the problem of visually covering a plane using isovists generated from a finite number of vantage points. As it would be somewhat difficult to implement a geometrically rigorous calculation of the coverage, we generated observation points discretely and evaluated the coverage as an approximation by assuming that they were all visible from the isovist. In this case, in order to ensure the representation of the structure of the space, it is a constraint that the observation points are visually connected to each other. Considering a situation such as that used in filming, we aim to position the isovist's home point at what we consider the center of the space. Let P and O be sets of candidate vantage points and observation points, respectively.

Definition of isovist

Let us take as an example a plane with a T-shaped hole in a square region as shown in Figure 1(a). The polygon formed by connecting the visible observation points in sequence from the vantage point $p \in P$ is the isovist in this study. As shown in Figure 1(b), the usual isovist simply expresses the geometrical extent of the line of sight without depending on the observation points; however, the isovist in this study is a polygon composed of the periphery of the visibility graph connecting the visible observation points, and the area of the isovist is somewhat narrower (Figure 1(c)). To reduce this difference, it is necessary to narrow the interval between the generations of the observation points.

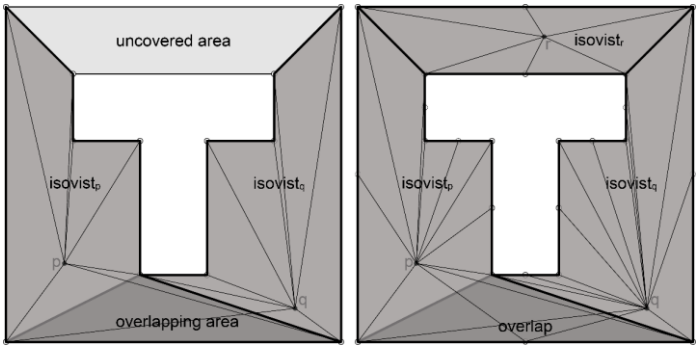


(a) Vantage and observation points (b) Normal isovist (c) Isovist

Figure 1. Target space and isovist

Covering

Next, the isovist coverage is explained. When the observation points are placed only at the endpoints of each line segment, as shown in Figure 2(a), the two monitoring points at the positions shown in the figure are sufficient to cover all the observation points. However, it can be seen that the upper part of the space is not covered by the isovist. Therefore, an observation point was added at the midpoint of each line segment, as shown in Figure 2(b). In this case, three observation points are required to cover all the observation points, and the corresponding isovist also shows that the space is completely covered. In this example, to fill the gaps in space, the observation points must be placed at the required density.



(a) Low-density observation points (b) High-density observation points

Figure 2. Explanation of covering

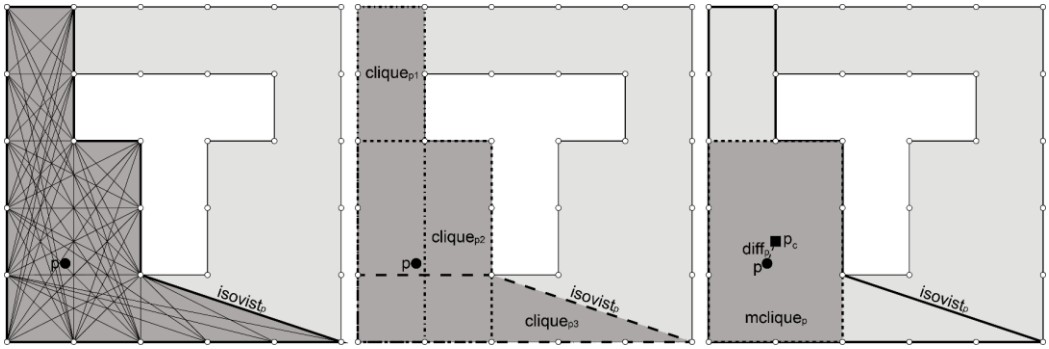
Connectivity

Connectivity has been introduced as a fundamental constraint for axial lines in the automatic generation of axial maps and is also introduced in this study because it is a prerequisite for analyzing space graphically. In this study, isovists were considered connected if their vantage points were directly visible to each other. In addition to visual connectivity, linear connectivity also exists. For example, if the connectivity of an axial map is the former, the connectivity of a segmentation map divided among road segments corresponds to the latter (van Nes and Yamu, 2021). It can be said that dynamic connectivity is a graphical representation of the topological property of space that indicates whether or not one spatial unit is continuously accessible from another, regardless of whether it is visually visible or not. If so, it is conceivable to express the isovist connectivity in terms of the isovist overlap, which is

relatively easy to formulate and solve. However, we did not adopt this method because, combined with the omnidirectional nature of isovists, it connects to distant isovists that cannot be accessed without passing through other spatial elements. The limitations of the connectivity used in this study and future directions are discussed later.

Centrality

The last key point is centrality, which is the goal of this research: to cover space with an isovist and structure the space in terms of its connectivity. To capture an omnidirectional image, it is desirable to capture the image at the center of each individual "space" as much as possible. The problem is how to define these "spaces" and their "centers." Because isovists are a class of star-shaped regions that radiate from a viewpoint, the coherence of the entire region as a space is generally weak. Conversely, convex regions are a class of regions in which any two points within them are visible and can easily be a unit of a coherent space. An isovist is a broader region class than convex regions; therefore, we evaluated spatiality in terms of convex regions contained within the isovist. For our limited isovist, the convex hull is the maximal clique in the visibility graph formed by connecting the observation points (Figure 3(a)). However, because there are generally multiple maximal cliques, it is necessary to select the one that is representative of the isovist from among their names. Figure 3(b) shows an example of the three maximal creeks. First, cliques p_1 and p_2 contain a vantage point, but not p_3, so the latter is excluded from the list of candidates. The first type of space that is easily recognized is a solid boundary (Stamps, 2005), where the region is divided by a wall. Comparing creek p_1 and creek p_2, the latter can be a more coherent unit of space because the ratio of the lengths of the walls that overlap the line segments that make up the creek is higher. To quantify this, we proposed an index that divides the number of vantage points constituting a creek by its circumference in the next section. If the space is to be divided into a small number of vantage points, the convex hull should be as large as possible. In other words, the creek area should be used as an index. Incidentally, Stamps (2005) concluded that the only two major attributes among the 25 indicators created by the Isovist are area and concavity. The creeks were ranked by the weighted sum of these two indices, and the creek with the largest sum was in the convex region of the isovist.



(a) Visibility graph. (b) Example of multiple maximum cliques. (c) Representative clique.

Figure 3. Examples of derived process maximum creeks up to representative creek

Finally, as shown in Figure 3(c), let $diff_p$ be the distance between a vantage point $p \in P$ of the isovist and the center of its representative clique. For an ideal isovist, the $diff_p$ is expected to be as close to zero as possible because the vantage point is at the center of the space.

Specific flow and formulation

Table 1 summarizes the variables used in the mathematical programming.

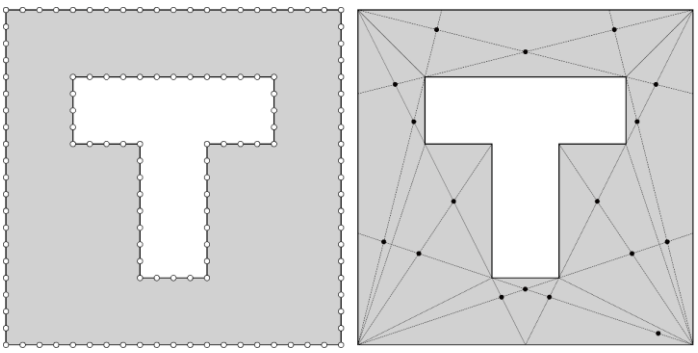
The main flow is as follows. The proposed method consists of steps 1-4. These are described in the following order.

Table 1. Symbols used in variable names

Category	Symbol	Description
Set	P	Set of candidate vantage points in the plane
	O	Set of observation
	V_p	Set of vantage points visible from $p \in P$ (including self, $V_p \subset P$)
	W_o	Set of vantage points visible from $o \in O$ ($W_o \subset P$)
	E	Set of edges formed by connecting $\forall q \in V_p \setminus \{p\}$ at $\forall p \in P$
	G	Visibility graph $G = (P, E)$
Set of lazy constraint	P_1	Valid set of vantage points ($p \in P_1 \subset P$) such that $x_p = 1$ in the solution candidates
	G_{P_1}	Induced subgraph of G by P_1
	C	A connected component of G_{P_1} with C ($C \in G_{P_1}$)
	N_C	The set of visible and invalid candidate vantage points from the vantage points contained in C
Variable	n	The number of isovist vantage points
	x_p	Whether $p \in P$ is a vantage point ($x_p \in \{0,1\}$)
Constant	$diff_p$	Euclidean distance between the vantage point $p \in P$ and the center of its representative clique

Step 1: Placement of observation points and candidate vantage points

After reading the floor plan data, the observation points were placed on the wall to evaluate the visibility of the entire space. The total number of observation points to be placed was assumed to be constant and expressed as $|O|$. First, we place observation points at the endpoints of the line segments that make up the wall surface and then place observation points on each line segment with as constant a spacing as possible until the constant is satisfied (Figure 4(a)).



(a) Observation points (b) Vantage points

Figure 4. Arrangement of observation and vantage points

Next, the isovist candidate vantage points are placed in the space. Similarly, the total number of points was assumed to be constant and denoted as $|P|$. Although isovists are often generated on a lattice, following Psarra and McElhinney (2014), we performed a preliminary experiment by uniformly arranging the candidate vantage points randomly in space, and found that vantage points close to m-lines were often selected. To improve the computational efficiency by eliminating unnecessary vantage points and because the visual connectivity of the vantage points is a constraint in this experiment, we decided to generate vantage points at the intersection of multiple m-lines (Figure 4(b)). However, in this case, points close to walls and points at nearby intersections are excluded or represented by a single point, respectively.

Step 2: Selection of isovist and maximum convex region

An isovist is generated by extracting the visible observation points for each candidate vantage point $p \in P$ and connecting them by straight lines in angular order from p. Next, we create a visibility graph by connecting these observation points, enumerate the maximal cliques of this graph, and denote the set of maximal cliques as M_p. Among the enumerated maximal cliques, the maximal clique $m \in M_p$ with the maximum value of the weighted sum of the standardized wall fraction and area, as shown in the following equation, is adopted as the representative clique.

$$\alpha \cdot \mathrm{std}\left(\frac{|O_m|}{pelimter_m}\right)_p + (1-\alpha) \cdot \mathrm{std}(area_m)_p \tag{1}$$

In the above equation, $O_m \subset O$ is the set of observation points constituting m, $pelimter_m$ and $area_m$ are the side length and area of the polygon by m, respectively, $\mathrm{std}()_p$ is the function to standardize the wall fraction and area of each maximal creek in M_p, respectively, and $\alpha \in [0.0,1.0]$ is the weight between them. Finally, let $diff_p$ be the Euclidean distance between p and the center of gravity of the convex region using its representative clique.

Step 3: Finding the minimum number of vantage points (Opt.1)

The first is the number of isovists that can cover the plane while satisfying the isovist linkage constraints. For this purpose, we define the following mixed-integer quadratic constraint programming problem.

Minimize
$$n \tag{2}$$

subject to
$$\sum_{p \in P} x_p = n \tag{3}$$

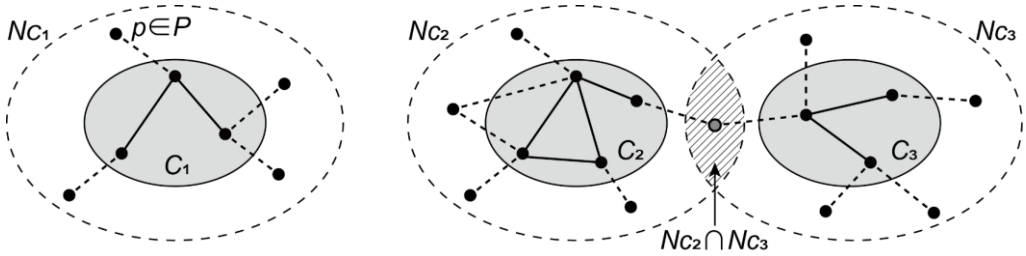
$$\sum_{p \in W_q} x_p \geq 1, \quad \forall q \in O \tag{4}$$

lazy constraint
$$\sum_{p \in C} x_p - \sum_{p \in N_C} x_p \leq |C| - t_C, \qquad \text{if } |G_{P_1}| \geq 2, \ \forall C \in G_{P_1} \tag{5}$$

$$t_C = \begin{cases} 1, & \text{if } N_C \cap N_D \neq \emptyset, \quad \exists D \in G_{P_1} \setminus C \\ 2, & \text{otherwise} \end{cases}, \quad \forall C \in G_{P_1} \tag{6}$$

Equation (4) guarantees that any observation point is visible from one of the selected vantage points. Equation (5) is a lazy constraint that guarantees that the isovist graph is connected. Lazy constraints are a bottom-up method in

which after finding a tentative solution, the tentative solution is checked to see if it is disconnected. If so, the necessary constraints are added successively to gradually strengthen the constraints. First, the set of vantage points for $x_p = 1$ is extracted from the optimization results and denoted as $P_1 \subseteq P$. Then, an induced subgraph G_{P_1} induced by P_1 is extracted from the graph $G = (P, E)$. Lazy constraint (5) is added when the number of connected components of G_{P_1} is at least two, i.e., G_{P_1} is disconnected. This constraint indicates that at least t_C vantage point changes are required in the candidate set of vantage points (neighborhoods) adjacent to the connected components and their neighbors because the node combination of each connected component does not satisfy the constraint as it is. For an isolated connected component to be connected to others, at least one of the candidate vantage points in the neighborhood must be valid. However, under the constraint that the total number of vantage points is n, it is necessary to disable the vantage points of the connected component near candidate vantage points as many times as possible. t_C is a threshold value that takes one of the values shown in Equation (6) depending on the conditions. For example, as shown in Figure 5, there is an induced subgraph consisting of three connected components $\{C_1, C_2, C_3\}$ in the space, and Let $N_{C_1}, N_{C_2}, N_{C_3}$ be their neighborhoods. In this case, $t_C = 2$ because it is necessary to validate one candidate vantage point in the neighborhood and invalidate one vantage point in the connected component. By contrast, N_{C_2} and N_{C_3} have a common set. In this case, $t_C = 1$, because the set can be connected by making one of the candidate vantage points in the common set valid and one of the vantage points of one of the connected components undirected.



(a) Isolated connected component (b) Two connected components sharing a common neighborhood

Figure 5. Difference between connected components and neighborhoods

Step 4: Select the isovist that occupies the center of the space (Opt.2)

The following is a mathematical programming problem in which the number of vantage points is set to a value greater than or equal to the minimum n obtained in Opt.1, and the isovist is chosen such that the vantage points are placed as much as possible in the center of the space while satisfying the connectivity constraints. The constraints are the same as in Opt.1. In Opt.2, n increased as necessary because there were cases where the vantage point did not reach the center of each space with the minimum n.

Minimize $$\sum_{p \in P} x_p \cdot diff_p \tag{7}$$

subject to $$\sum_{p \in P} x_p = n \tag{8}$$

$$\sum_{p \in W_q} x_p \geq 1, \quad \forall q \in O \tag{9}$$

lazy constraint

$$\sum_{p\in C} x_p - \sum_{p\in N_C} x_p \le |C| - t_C, \qquad \text{if } |G_{P_1}| \ge 2,\ \forall C \in G_{P_1} \tag{10}$$

$$t_C = \begin{cases} 1, & \text{if } N_C \cap N_D \neq \emptyset, \quad \exists D \in G_{P_1} \setminus C \\ 2, & \text{otherwise} \end{cases}, \quad \forall C \in G_{P_1} \tag{11}$$

Verification

Verification was performed using the following settings.

Floor plans and parameters

In this study, six floor plans of architectural and street scales were used, focusing on familiar spaces in this type of analysis (see Figure 6). Buildings 1,2,3 (Peponis, 1999) and the National Gallery (Turner et al., 2005) were used as well-formed spaces, and a short-term treatment facility for children with emotional disorders (Treatment Facility; Fujimoto, 2005) and Gassin (Hiller and Hanson, 1984) were used as irregular spaces.

Table 2 lists the number of vantage and observation points in each space. The number of observation points in each space was approximately 1000 and they were placed as evenly as possible on the wall. The vantage point was generated at the intersection of the m-lines and one of the vantage points within 1/3 of the minimum side length of each space model was selected.

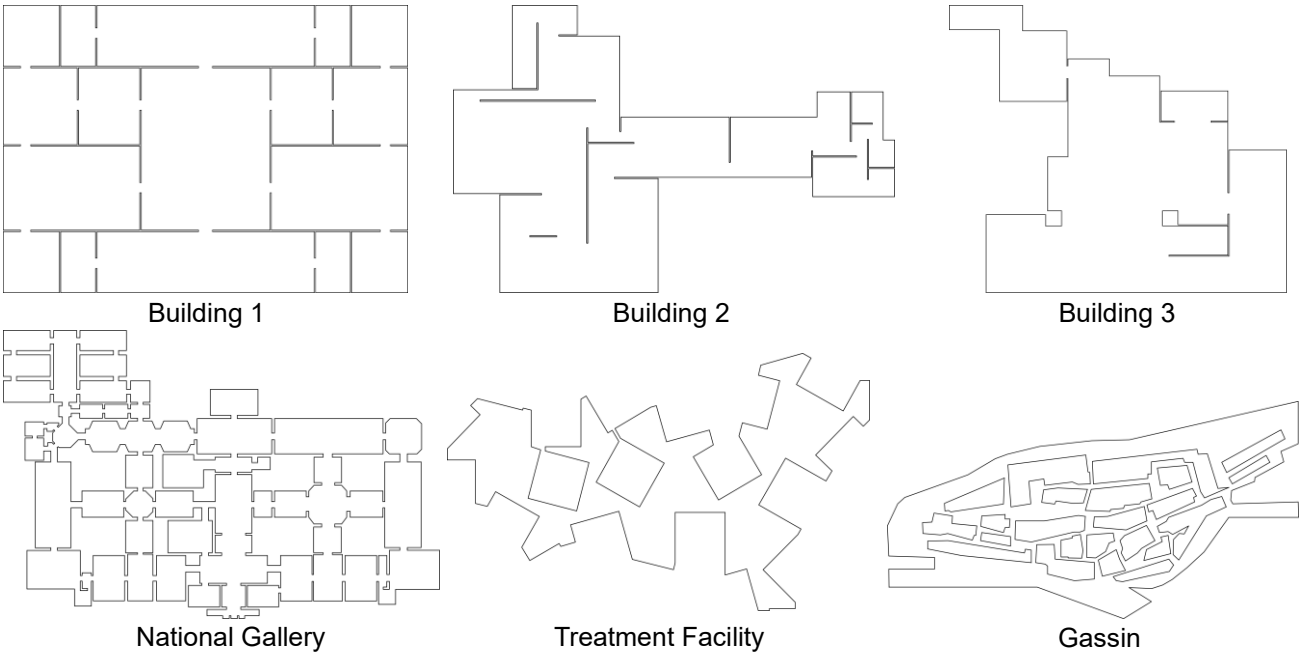


Figure 6. Floor plans used in the verification

Implementation of the Method

This system was implemented in Python 3.12, based on Grasshopper, and running on Rhinoceros 8.0. Graphs, such as clique extraction, were computed using igraph. Mathematical programming problems were implemented using the Python API of the mathematical programming solver Gurobi 12.0.1. The optimization options were Params.MIPFocus=2 (prioritizing finding the exact solution) and a 10-minute limit on the solution time. The computer used was Windows 11 Pro 24H2, Intel Core Ultra 285 K, DDR5 5200 Hz, and 192GB Memory.

Table 2. Number of vantage points and observation points placed in each floor plan

Name	$\|P\|$	$\|O\|$
Building 1	1359	1058
Building 2	488	1034
Building 3	1107	1008
National Gallery	1923	1447
Treatment Facility	1021	997
Gassin	1574	1050

Results

Opt.2 shows the results for $\alpha = 0.5$. Table 3 shows the minimum n obtained by Opt.1, the number of n for which the number of vantage points was set to be appropriate according to the author's subjective judgment in Opt.2, and the time required to solve the problem. The gap is the error between the theoretical optimal solution and the solution found; a zero gap indicates that the optimal solution has been found. The solution times varied from case to case, but Opt.1's solution times were instantaneous or at most 30 s, except for two cases in which the optimal solution was not found within the time limit. Opt.2, on the other hand, took at most 30 seconds, although this was partly due to the fact that n was increased (i.e., the concatenation constraint was more easily satisfied).

For comparison, Appendix A1 of the supplement material shows the connection constraints with network flows and their computation results, which show that they are less efficient than the lazy constraints.

Table 3. Summary of optimization results

Floor plan	n		Gap (%)		Computational time (sec)	
	Opt.1	Opt.2	Opt.1	Opt.2	Opt.1	Opt.2
Building 1	15	17	0.0	0.0	13.5	32.8
Building 2	12	←	0.0	0.0	0.1	0.5
Building 3	6	←	0.0	0.0	0.4	0.8
National Gallery	50	58	2.0	0.0	> 600	42.4
Treatment Facility	11	12	9.1	0.0	> 600	35.3
Gassin	32	←	0.0	26.0	120.4	> 600

Figures 7 and 8 show the results of the placement of each plan view using Opt.1 and Opt.2 for each plane. In addition to the vantage points and edges connecting them, the visualization of the Opt.1 results shows that the areas covered by isovists were colored to confirm that there was no coverage leakage. In the Opt.2 visualization, the main creek for each isovist is shown. The following sections describe the floor plans.

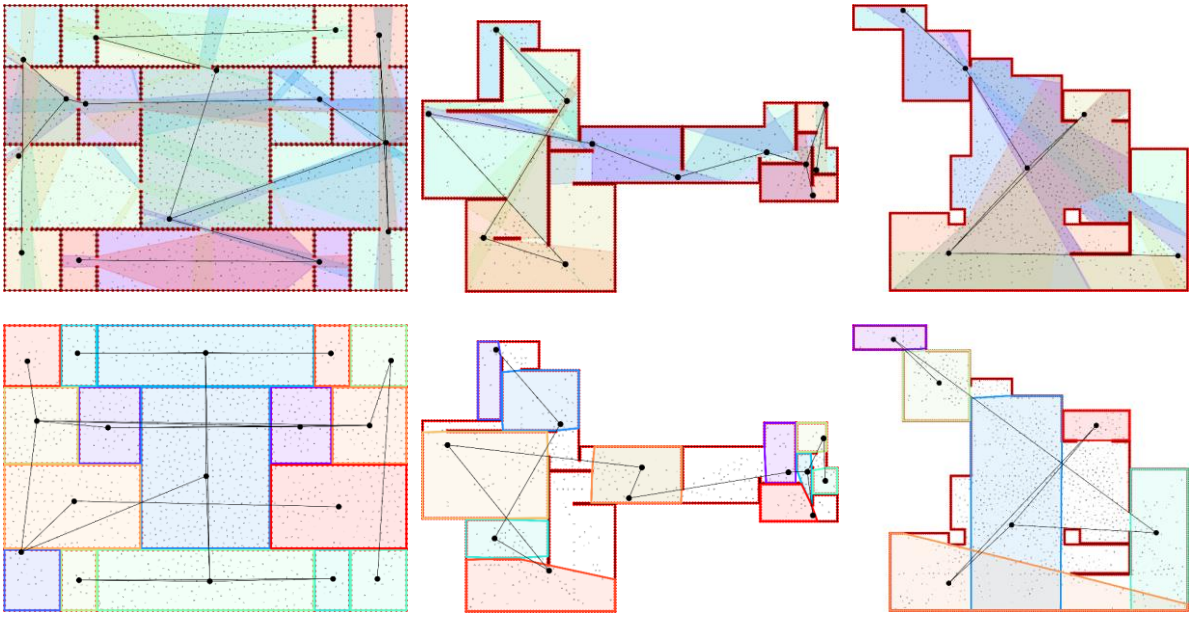


Figure 7. Optimization result 1 (Upper row: Opt.1, Lower row: Opt.2)

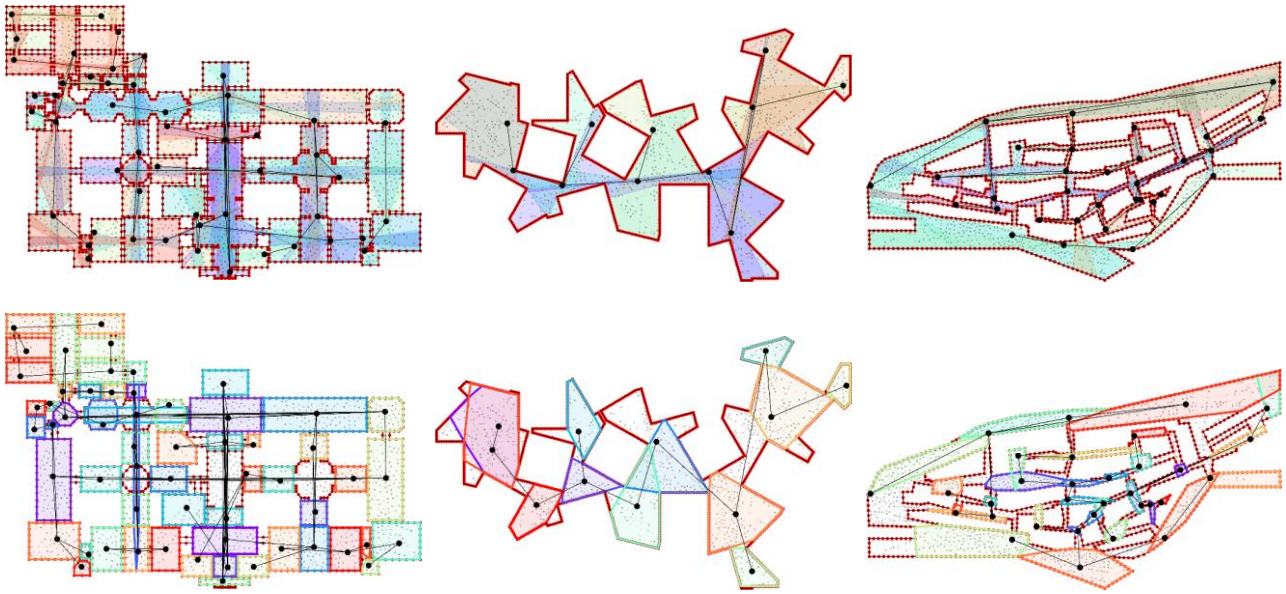


Figure 8. Optimization result 2 (Upper row: Opt.1, Lower row: Opt.2)

Building 1 consists of a series of rectangular rooms, and the center of the space and the number of points required can be easily assumed. Fifteen points were the minimum number of points required by Opt.1, but this did not place the rooms in the center of the space, but at the boundaries. Therefore, Opt.2 set the number of points to 17 to match the number of rooms. As the vantage points must be visible, the vantage points in the lower-left room are placed at the boundary. The other rooms were placed in the center of the room.

Building 2 consisted of a relatively finely segmented space with ambiguous boundaries. Both examples show the minimum number of vantages' points. Because they have the same number of vantage points, they are similar, except for the finer space on the right side.

Building 3 is loosely segmented from a wide space to a narrow space with few boundaries. Compared to Opt.1, the narrow space had a point placed in the center.

The National Gallery is a segmented space, as shown in Building 1. There was a difference between the minimum number of vantage points and the appropriate number of vantage points. However, there are still places where the vantage points are not placed, such as the circular empty space on the right-hand side.

Treatment Facility has irregular systems and unclear boundaries. Because the vantage points tended to be located near the walls with the minimum number of units, we increased the number by one and performed Opt.2. The impression is that the vantage stores are arranged in a coherent spatial unit.

Gassin ran Opt.2 with the minimum number of vantage points. The arrangement of the interior vantage points tends to be slightly different from Opt.1.

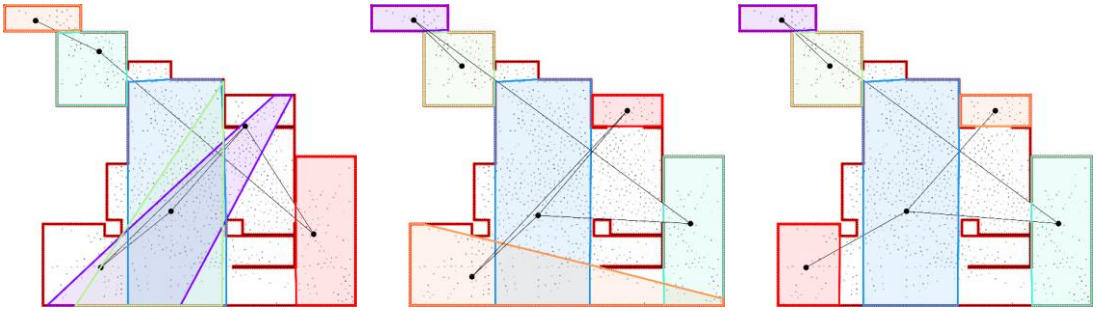
The above results show that the minimum number of vantage points by Opt.1 tended to be placed at the boundary, but by Opt.2, the number of vantage points was placed at the center of the space by increasing the number of vantage points as necessary. However, the connection relation of the vantage points is often not that of the neighbors but that of the distant vantage points, which will be discussed in the next chapter.

Discussion

The following are discussed.

Difference in α

By setting α in Opt.2, the convex hull representing the isovist can be set in consideration of the balance from a spatially closed convex hull to a convex hull with a large area. The previous results were obtained for $\alpha = 0.5$. Figure 9 shows what happens when $\alpha \in \{0.0,0.5,1.0\}$ is changed, using Building 3 as an example. For $\alpha = 0.0$, the creek with the largest area is selected, and the center of the isovist tends to shift due to the presence of multiple creeks that extend far out from small rooms. On the other hand, $\alpha = 1.0$ is the case where the creek with the largest amount of walls is selected, and the isovist's vantage point tends to be placed in the center of small spaces, including Appendix A2 in the supplemental material. $\alpha = 0.5$ is a result that balances these results overall.



(a) $\alpha = 0.0$ (b) $\alpha = 0.5$ (c) $\alpha = 1.0$

Figure 9. Differences in Opt.2 results according to α using Building 3

Connectivity

This section discusses the connectivity. The results showed a tendency to connect to distant home points. This result is somewhat unnatural from the viewpoint of spatial connectivity as perceived by humans. People may perceive spatial connectivity in a kinematic sense. In this study, the visual connectivity of the vantage points was used as a constraint. According to Turner et al (year): However, as indicated by Batty and Rana (2004), in isovist coverings,

the isovist region itself is always connected or adjacent according to the definition of the problem. This connectivity guarantees second-order visibility. As mentioned in Section 3, isovist regions themselves often intersect with distant isovist regions; therefore, expressing connectivity in this manner would result in an excessive degree of connectivity, as the isovist region itself would be connected to the distant isovist region. Alternatively, if orthodox convex partitioning is applied, for example, coverage and linear connectivity are satisfied. However, the results of Carranza and Koch (2013) show that there is a tendency for excessive partitioning, such as the 114 partitions in Gassin. Because the main objective of this study was to propose a method for solving connectivity in a rigorous and efficient manner, the issue of connectivity remains. Future work is required to improve the proposed method to consider dynamic isovist connectivity in applications such as graph analysis.

Other limitations and future issues

In this study, monitoring points were placed discretely on the wall surface to simplify the implementation of the spatial analysis. isovist is geometrically determined by the location of the vantage point; therefore, if rigor is important, it is desirable to solve the covering problem under continuous conditions. To simplify the computation time, the vantage points are placed only at the intersection points on the m-line as a rule of thumb. The effect of this limitation on the results has not been fully verified and will need to be verified in conjunction with improvements to the model. In addition, the exact solution method has some computational limitations, and the development of an approximate solution method is necessary.

Conclusion

In this study, we proposed an isovist graph as a new concept of a spatial model that covers a plane with an isovist under the condition of connectivity, mainly for architectural and street-scale spaces, and proposed a fast exact solution method for the automatic generation of axial maps by devising the connectivity constraint, which has been difficult so far. We proposed a new method to solve the problem of how to generate an axial map automatically by devising linkage constraints and verified the method using six architectural and urban area models. Several plans can be solved in a short time by sampling data, devising constraint expressions for coupling conditions, such as delay constraints, and using the latest mathematical programming solvers. Although complex plans require more computation time, and in some cases, the optimal solution was not obtained, the optimality gap was demonstrated and could be used as a guide when creating heuristics in the future. The center of space is important for this method because it determines the point at which an omnidirectional image is captured. The vantage point could be placed close to the center of the space by setting $\alpha = 0.5$. This study aimed to develop an isovist graphical model to analyze spatial configurations. However, the model cannot model local connectivity relations, such as lines of flow, because the connection relation of the vantage points is often such that distant vantage points are connected. In addition, the space was sampled discretely. Improvement in these points is necessary for practical use in the future.

Acknowledgment

We would like to thank Associate Professor Sam McElhinney at UCA Canterbury for sharing information about Isovist_App.

Funding

This research was supported by a Grant-in-Aid for Scientific Research (Grant no.23K04161).

References

Batty M and Rana S (2004) The automatic definition and generation of axial lines and axial maps. Environment and Planning B: Planning and Design 31(4): 615-640. DOI: 10.1068/b2985.

Benedikt, M. L. (1979). To take hold of space: isovists and isovist fields. Environment and Planning B: Planning and Design, 6(1): 47-65. DOI: 10.1068/b060047

Carranza PM and Koch D (2013) A computational method for generating convex maps using the medial axis transform. Proceedings of the Ninth International Space Syntax Symposium, Seoul, Korea: 064.

Chazelle B and Dobkin DP (1985) Optimal convex decompositions, Toussaint GT (Ed.), Computational Geometry. North-Holland, Amsterdam: 63-133.

Dehbi Y, Leonhardt J, Oehrlein J and Haunert JH (2021) Optimal scan planning with enforced network connectivity for the acquisition of three-dimensional indoor models. ISPRS Journal of Photogrammetry and Remote Sensing 180: 103-116. DOI: 10.1016/j.isprsjprs.2021.07.013.

Hamzei E, De Cock L, Tomko M, Van de Weghe N, and Winter S. (2024). Indoor view graph: a model to capture route and configurational information. Environment and Planning B: Urban Analytics and City Science 51(9): 2213-2231. DOI: 10.1177/23998083241241598

Hillier B and Hanson J (1984) The Social Logic of Space. Cambridge NY: Cambridge University Press, DOI: 10.1017/CBO9780511597237.

Hwang Y (2013) Network communities in the visibility graph: A new method for the discretization of space. Proceedings of the Ninth International Space Syntax Symposium, Seoul, Korea: 045.

Jung SK and Kim Y (2020) A linear programming method for finding a minimal set of axial lines representing an entire geometry of building and urban layout. Applied Sciences 10(12): 4273. DOI: 10.3390/app10124273.

Karp RM (1972) Reducibility among combinatorial problems. Complexity of Computer Computations: 85-103. DOI: 10.1007/978-1-4684-2001-2_9.

Koo BW, Guhathakurta S and Botchwey N (2022) How are neighborhood and street-level walkability factors associated with walking behaviors? A big data approach using street view images. Environment and Behavior 54(1): 211-241. DOI: 10.1177/00139165211014609

Lingas A (1982) The power of non-rectilinear holes. Proc. 9th International Colloquium of Automata languages, and Program. Lecture Notes Computer Science 140, Springer-Verlag, Berlin: 369-383.

McElhinney S (2024) The Isovist_App: a basic user guide, v1.7.

Matterport (2025) Matterport Discover. https://discover.matterport.com.

Peponis J, Wineman J, Bafna S, Rashid M and Kim SH (1998) On the generation of linear representations of spatial configuration. Environment and Planning B: Planning and Design, 25(4): 559-576. DOI: 10.1068/b250559.

Peponis J, Wineman J, Rashid M, Kim SH and Bafna S (1997) On the description of shape and spatial configuration inside buildings: convex partitions and their local properties. Environment and Planning B: Planning and Design 24: 761-781.

Psarra S and McElhinney S (2014) Just around the corner from where you are: probabilistic isovist fields, inference and embodied projection. Journal of Space Syntax 5(1): 109-132.

RICOH (2025) RICOH360 Tours. https://www.ricoh360.com/ja/tours/.

Shirabe T (2005) A model of contiguity for spatial unit allocation. Geographical Analysis 37(1): 2-16. DOI: 10.1111/J.1538-4632.2005.00605.X.

Sou Fujimoto Architects (2006) Short-term treatment facility for emotionally disturbed children, living wing, Shinkenchiku 81(10): 174-184, 234. https://ndlsearch.ndl.go.jp/books/R000000004-I8033164.

Stamps AE (2005) isovists, enclosure, and permeability theory. Environment and Planning B: Planning and Design 32(5): 735-762. DOI: 10.1068/b31138.

Takizawa A (2024) Extracting real estate values of rental apartment floor plans using graph convolutional networks. Environment and Planning B: Urban Analytics and City Science 51(6): 1195-1209. DOI: 10.1177/23998083231213894.

Takizawa A and Kinugawa H (2020) Deep learning model to reconstruct 3D cityscapes by generating depth maps from omnidirectional images and its application to visual preference prediction. Design Science 6: e28. DOI: 10.1017/dsj.2020.27.

Turner A, Doxa M, O'Sullivan D and Penn A (2001) From isovists to visibility graphs: a methodology for the analysis of architectural space. Environment and Planning B: Planning and Design 28(1): 103-121. DOI: 10.1068/b2684.

Turner A, Penn A and Hillier B (2005) An algorithmic definition of the axial map. Environment and Planning B: Planning and Design 32(3): 425-444. DOI: 10.1068/b31097.

Van Nes A and Yamu C (2021) Analysing linear spatial relationships: the measures of connectivity, integration, and choice. Introduction to Space Syntax in Urban Studies. Springer, Cham. DOI: 10.1007/978-3-030-59140-3_2.

Appendix

A1. Network flow connection constraints and solving time

Let us consider a graph in which each isovist's candidate vantage points are nodes and their connection relations are edges. The graph is directed with directed edges in opposite directions between the visible nodes. In the following equations, $w_p \in \{0,1\}$ is a binary variable indicating whether node $p \in P$ is sink or not, $E_p \subset E$ is the set of edges connected to $p \in P$ and $f_{p,q} \in \mathbb{R}_0^+$ is the flow in the direction from node p to q. Equations (A1) and (A2) set one of the active nodes to be the sink into which the flow flows. A unit flow is directed from each active node (except for the sink) to the other active node to which it is connected. If the graph is connected, the flows eventually converge on the sink, and the upper bound of the sum of the flows leaving each node is bounded by (A3). The lower bound of the difference between the sum of the flows leaving an active node and the sum of the incoming flows is bounded by equation (A4). When a node is inactive, the upper and lower bounds of the flows are zero, that is 0. However, if the node is active and not a sink, the lower limit of the difference between the output and input of the flow is 1, and the flow must be sent elsewhere. However, for an active sink, $x_p = w_p = 1$, and there is no limit to the flow into the sink. Consequently, the outgoing flows are concentrated at the sink, and the graph becomes connected.

subject to

$$\sum_{p \in P} w_p = 1 \tag{A1}$$

$$w_p \le x_p, \quad \forall p \in P \tag{A2}$$

$$\sum_{(p,q) \in E_p} f_{p,q} \le (|P| - 1) \cdot x_p, \quad \forall p \in P \tag{A3}$$

$$\sum_{(p,q) \in E_p} f_{p,q} - \sum_{(q,p) \in E_p} f_{q,p} \ge x_p - |P| \cdot w_p, \quad \forall p \in P \tag{A4}$$

Table A1 lists the network flow constraints. This was slower than the delay constraint, and Gassin was unable to find an approximate solution.

Table A1. Optimization results with network flow constraints

Floor plan	Objective		Gap (%)		Computational time (sec)	
	Opt.1	Opt.2	Opt.1	Opt.2	Opt.1	Opt.2
Building 1	15	14.2	13.3	54.3	> 600	> 600
Building 2	12	43.3	0.0	0.0	3.7	60.3
Building 3	6	15.3	0.0	0.0	13.1	13.4
National Gallery	50	42.8	6.0	21.0	> 600	> 600
Treatment Facility	11	13.3	18.2	36.4	> 600	> 600
Gassin	NA	NA	NA	NA	> 600	> 600

A2. Opt.2 results for different values of α

Table A2 lists the results of Opt.2 for different values of α. Figure A1 presents the optimization results.

Table A2. Opt.2 results for different values of α

Floor plan	Gap (%)			Objective		
	$\alpha = 0.0$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 0.0$	$\alpha = 0.5$	$\alpha = 1.0$
Building 1	0.0	0.0	0.0	44.4	14.3	14.3
Building 2	0.0	0.0	0.0	52.9	43.3	37.8
Building 3	0.0	0.0	0.0	44.0	15.3	6.7
National Gallery	0.0	0.0	0.0	75.1	42.4	31.3
Treatment Facility	0.0	0.0	0.0	61.8	12.5	8.2
Gassin	65.5	26.0	18.4	712.1	288.4	139.3

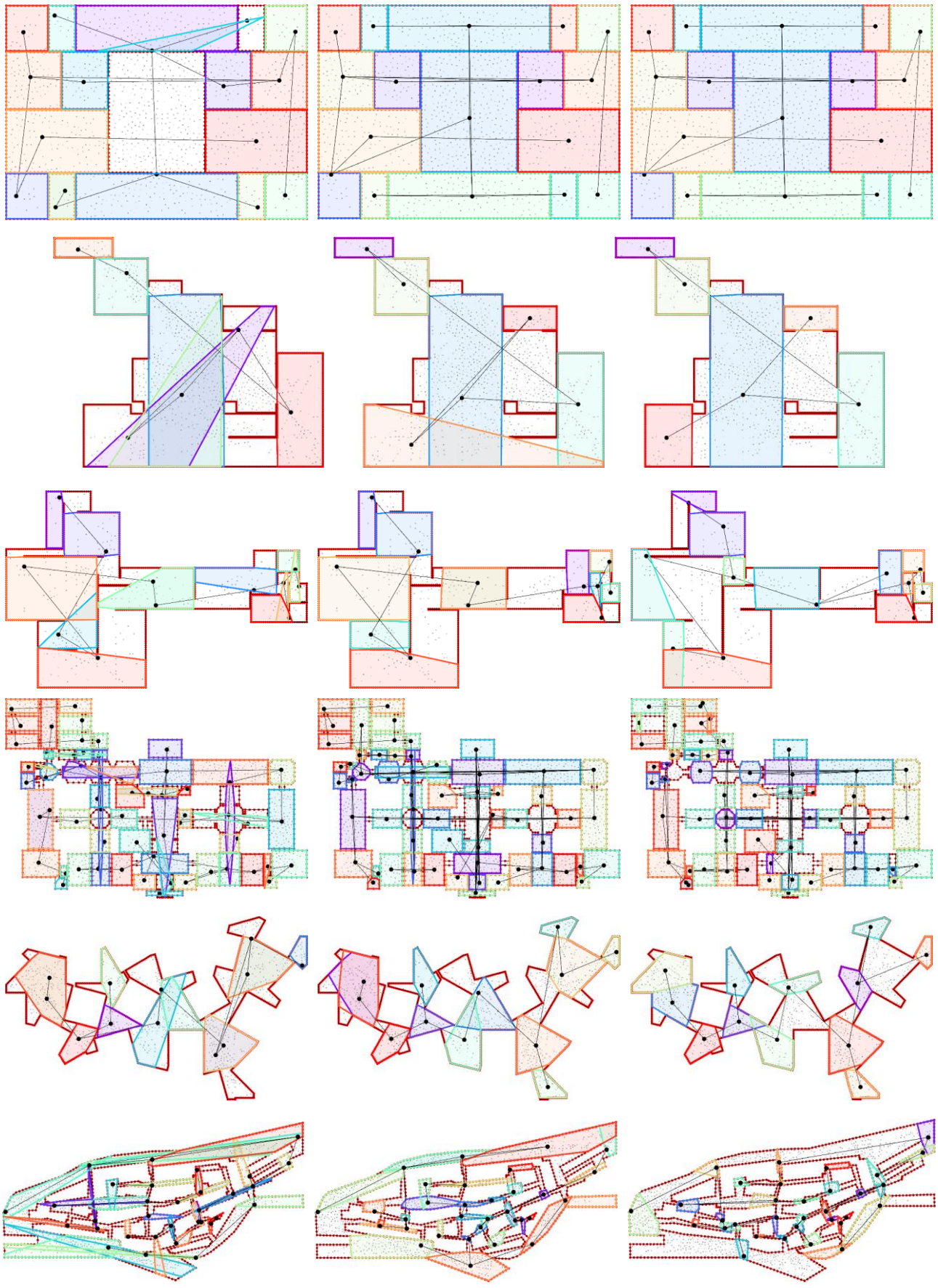


Figure A1. Optimization results for each floor plan for different α values.