Approximated Environment Features With Application to Trajectory Annotation

Sebastian Feld
Mobile and Distributed Systems Group
LMU Munich
Munich, Germany
Email: sebastian.feld@ifi.lmu.de

Martin Werner
Mobile and Distributed Systems Group
LMU Munich
Munich, Germany
Email: martin.werner@ifi.lmu.de

Claudia Linnhoff-Popien
Mobile and Distributed Systems Group
LMU Munich
Munich, Germany
Email: linnhoff@ifi.lmu.de

***Abstract*—Indoor location-based services is an ever emerging field of research and application. In particular indoor navigation, i.e. positioning and route planning in freely walkable space, gets much attention. The philosophy of this paper is to incorporate isovists as a numerical representation of the local environment to enable new types of indoor LBS. This paper has got two main contributions. First, we define discrete isovists as an approximation of exact isovists using a simple ray casting approach. Second, we define, create, and use isovist feature cubes for semantic evaluations of floor plans as well as trajectory annotation.**

I. Introduction

The installation of cost-efficient GPS receiver into modern smartphones has made location-based services (LBS) in outdoor scenarios available for millions of people [1]. Use cases like finding points of interest (POI) in the vicinity, geocaching or navigation systems are amongst the most popular tasks when using a smartphone. In particular the transformation of road networks into graphs using vertices and edges and the subsequent efficient solving of the shortest path problem has got received vast attention in the last years [2].

Calculating and processing routes get more complicated when leaving the quite restricted road network and considering constrained free space. Examples of scenarios having a high degree of freedom of movement are public parks or complex buildings like hospitals or airports. When driving a car on a highway there are few possibilities to act besides driving ahead or changing the highway at an interchange. In contrast, a pedestrian inside a building can potentially choose to walk in every direction while avoiding obstacles.

Geographic information systems (GIS) are essentially one of the foundations for creating location-based services [3]. GIS deal with the acquisition, processing, organization, analysis, and presentation of spatial data. The tasks just stated can get highly complex based on the underlying use case. There is the problem of defining the appropriate number and types of representations necessary for the given use case, and the not to be underestimated modelling effort to capture the spatial data. Given the example of a common office, it is not directly obvious which level of detail is suitable to include into the system (just walls, static furniture, mobile items), which resolution is needed (decimeter, centimeter, millimeter), and how to deal with curved surfaces (preferably exact curve representation or piecewise linear approximation).

One possibility to represent spatial data such as floor plans are occupancy grids [4]. An occupancy grid is a 2D binary matrix with 1's representing cells that contain obstacles and 0's representing free and accessible space. There is a whole range of applications where occupancy grids are used for perception and navigation, mostly with mobile robots in the plane [5] or underwater [6]. The main drawback is the fact that occupancy grids are highly erroneous, at least caused by the mapping algorithms and the sensors prone to noise [7]. Nevertheless, there are huge benefits regarding the performance of certain operations and the simplicity of their generation. In time of need a simple photograph of an analog floor plan may be sufficient.

The philosophy of this paper is to incorporate the concept of isovists for creating new types of indoor LBS [8]. An isovist is basically the volume of space that can be seen from a given point in space [9]. This concept is widely used for decades to study the human perception of indoor space. As we focus on 2D maps, our understanding of an isovist is the area of a floor plan that is visible from a given point in the plane.

This paper has got two main contributions. First, we propose an approximation of isovists using a simple ray casting approach, thus defining discrete isovists. Second, we define and create isovist feature stacks that will be used for semantic evaluations of floor plans as well as trajectory annotation. Basically, we want to identify significant changes of the local environment using the isovist feature stacks just mentioned.

There are several applications conceivable such as the annotation of routes for creating textual representations usable in audio guidance, the enhancement of map quality by identifying doors or other structures that are previously not contained in the map, or the scheduling and planning of mobile robots, for instance, that perform tasks just at certain spots like large rooms or doors.

The remainder of this paper is structured as follows: In Section II we review related work in the field of isovist analysis and the relation between occupancy grids and indoor location-based services. Section III introduces our concept of discrete isovists, isovist feature stack, feature maps, and trajectory annotation. After evaluating our approach in Section IV we conclude our paper and address envisioned future work in Section V.

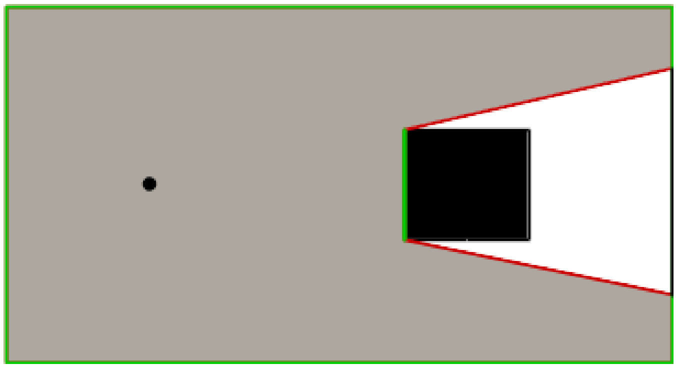


Fig. 1. An example of an isovist in an environment with one obstacle (black rectangle). The black circle indicates the point of view, the gray surface is the area A_x of the isovist itself, the green lines indicate the real-surface perimeter P_x, and the red lines correspond to the occlusivity Q_x.

II. Related Work

This section reviews related work regarding isovist analysis and the relation between occupancy grids and indoor LBS.

A. Isovist Analysis

A key research question of behavioral researchers and environmental psychologists is the connection between visual properties of an environment and human's subjective reaction to it. Probably the first mention of the term isovist has been done by Tandy in [10], where an isovist got introduced as all the points in space that are visible from a given point of view. Thus, an isovist can be regarded as a unique fingerprint of a specific spatial configuration of a certain point. It is not unique in the environment, but identical isovists have identical features. Based on Tandy's work, Benedikt has developed a formal definition of isovists and introduced a set of measurements that enable the numerical analysis of a given spatial environment [9]. Based on Benedikt's definition there are six isovist features for a 2D environment:

1) A_x: the *area* of the isovist. The higher the value the more space is visible from the viewpoint.
2) P_x: the *real-surface perimeter* of the isovist. It indicates the amount of visible obstacle surface like walls, for example.
3) Q_x: the *occlusivity* of the isovist. It indicates the length of the occluding radial boundary. It can be imagined as a virtual ray passing an obstacle and going through the room.
4) $M_{2,x}$: the *variance* of the isovist's radius.
5) $M_{3,x}$: the *skewness* of the isovist's radius.
6) N_x: the *circularity* of the isovist. This isoperimetric quotient is calculated using $N_x = |\partial V_x|^2/4\pi A_x$, whereas $|\partial V_x|$ indicates the isovist's perimeter.

Figure 1 illustrates an isovist with its area (gray surface), real-surface perimeter (green lines), and occlusivity (red lines).

The calculation of exact isovists in a GIS data model is complicated. An isovist is a polygon having two types of edges. One type is given by edges lying on top of building geometry (e.g. walls) and the other type is given by edges crossing free space. For simplicity, we will discuss the case of a GIS model consisting of line segments only. Note that all other entities in two-dimensional geometry can be sufficiently represented as line segments. An easy way to obtain an isovist polygon is given by the following procedure creating a list of points: First, a candidate line segment of the building geometry is to be retrieved. This can be done, for example, by intersecting a random line through the viewpoint of the isovist with building geometry. The nearest segment intersection between this ray and the GIS model will be a bootstrap segment. Using this segment, we can follow the right hand rule by choosing the adjacent line segment which forms the smallest angle with the current segment. We do so as long as there is such a segment and the triangle formed by the line segment and the basepoint is empty. In case there is no adjacent geometry to follow, we intersect a ray from the base point through the endpoint of the current segment and create a free space segment for our polygon. We insert the intersection point to the list of points and continue using the right hand rule as above. In case the triangle is nonempty, we have to find the first line segment inside it such that the triangle defined by intersecting a ray through this point with the current segment, the segment start and the base point is empty. We then use this line to create a free space segment from the intersection point to the first point of the previously identified line segment and continue with a right hand rule polygon traversal. Finally, this defines a polygon in which every line segment of the polygon is either lying fully in free space touching building geometry in the end points or lies entirely on top of building geometry. This polygon is the isovist.

As evident from the discussion above, there are complicated case distinctions and non-trivial GIS queries such as finding the nearest intersecting line segment, checking whether a triangle is empty or not, enumerating all segments that fall inside a triangle, and similar. While these can be sped up by careful spatial indexing such as R* trees [11], the overall complexity is high. Furthermore, long line segments can deteriorate the indexing performance and we might be compelled to subdividing all geometry to line segments that are short enough for efficient querying. But then, the number of line segments might become high and the overall complexity of the algorithm increases.

With our idea of discrete isvosists, we gain a high amount of performance for accepting a bounded amount of error due to discretization. While the computation of the exact isovist is linear in the number of line segments of the isovist and, thus, depends mainly on the map, it can be fastened by spatial indexing. However, spatial indexing works better for small line segments increasing the complexity of the computation. Additionally, all further computations using the isovist will be parametrized by the number of edges of the isovist, which also increases when subdividing line segments. For the discrete isovist, we first subdivide to a constant length given by the rasterization. But additionally, we simplify geometry directly to pixels such that line intersection can be done using line painting algorithms such as Bresenham algorithm [12]. This leads to a fast approximation of the exact isovist. We follow the same idea of calculating the isovist by turning a scanline around the clock from the base point. But this time, we just intersect this line with the building geometry abstracted in the raster map using a variant of line painting. This can be done quickly and with limited error during computation. However, the number of rays to be cast into the environment needs to be chosen carefully. If the number too high, the computational cost exceeds the calculation of an exact isovist; if it is chosen too small, chances are that it will miss important geometry. The same is true for choosing the discretization parameter (e.g., the pixel size). All measures derived from discrete isovists will be limited to the accuracy of representation as pixels.

B. Occupancy Grids and Indoor LBS

As already stated in the introduction, the scientific community of robotics and automation widely adopts occupancy grids for the perception of space and the navigation therein. Besides that, the calculation of alternative routes inside buildings is still an emerging field of research. The authors in [13] define alternative routes using the topological concept of homotopy [14], [15]. The rather complex test on homotopy has been simplified using occupancy grids as the underlying map representation. Two routes having the same start and end point are called homotopic and thus equivalent, if the polygon generated by the concatenation of the routes has got no obstacles (i.e., black pixel) inside. Furthermore, the authors proposed algorithms to generate said alternative routes.

Based on the definition and algorithms proposed in [13] there were continuing works based on alternative routes calculated using occupancy grids. The author of [16] created a heatmap showing the frequency of a pixel's traversal and based on that defined a congestion probability. The authors of [17] defined archetypal routes, i.e. preferably diverse routes that got selected by a fuzzy clustering algorithm based on a set of simple to compute features. As mentioned, both works base on occupancy grids.

III. Concept

Given an occupancy grid in the form of a monochrome bitmap, whereas white pixels indicate free and walkable space and black pixels indicate obstacles like walls or furniture. Based on this floor plan, we generate a navigation graph that will be used in the evaluation, whereas each white pixel is a vertex and each neighboring white pixels create an edge. Horizontal and vertical neighboring pixels create an edge weight of 1, diagonal neighbors get an edge weight of $\sqrt{2}$.

A. Discrete Isovists

Our first contribution is the approximation of an isovist, i.e. the definition of discrete isovists. This paper's main focus is on simplicity, for which reason we propose a simple ray casting approach. We define a parameter α and accordingly "shoot" $360/\alpha$ rays radially from a given vantage point. The expansion of a ray proceeds until it hits an obstacle or the map's boundary. For each of the point's rays we store the parameter α, the coordinates of the starting and end point as well as the length.

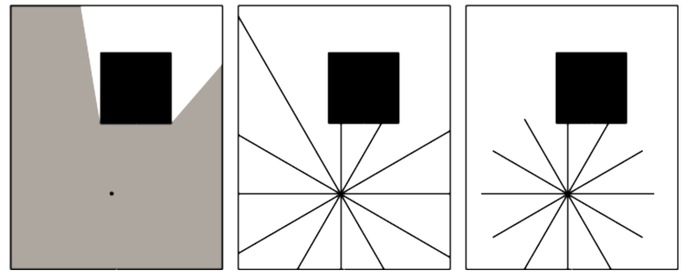


Fig. 2. An exemplary exact isovist (left-hand side), a discrete isovist with $\alpha = 30$ and $\gamma = \infty$ (middle), and a discrete isovist with $\alpha = 30$ and $\gamma = 30$ (right-hand side).

Besides the number of rays there is another parameter γ, i.e. the maximal length of the rays serving as a length threshold. If γ is set to infinite, then the ray expansion just stops when hitting an obstacle or the map's boundary. If γ is set to a natural number, however, the expansion stops not later than the threshold. The configuration of this parameter has got effects on both, the algorithm's runtime and the results. Figure 2 shows an exemplary exact isovist (left-hand side), a discrete isovist with $\alpha = 30$ and $\gamma = \infty$ (middle), and a discrete isovist with $\alpha = 30$ and $\gamma = 30$ (right-hand side).

The actual goal of the approximation is the determination of the six isovist features rather than the isovist itself. We define the *area* of the discrete isovist (A_x) to be the area of the polygon that emerges when connecting the end points of the rays. The *variance* ($M_{2,x}$) and *skewness* ($M_{3,x}$) of the isovist is defined as the variance and skewness of the length of the rays based on the Euclidean norm. The feature *circularity* is defined as the squared perimeter of the isovist divided by $4\pi A_x$. The area A_x is already described, we define the perimeter of the isovist as the perimeter of the polygon that emerges when connecting the end points of the rays.

The choice of simplicity and the resulting ray casting approach lead to the fact that the two features *real-surface perimeter* (P_x) and occlusivity (Q_x) can not properly be approximated. Thus, we remain with few simple operations to approximate four of the six isovist features.

One should keep in mind that the discretization using raster maps introduces systematic errors. For example, every pixel that contains a marginal amount of building geometry will get occupied, i.e. black. When measuring area or distance by counting, this should be considered: All measures of free space will be smaller than they truly are, as black pixels might not be fully occupied by geometry.

B. Isovist Feature Stack

After defining discrete isovists and the corresponding features, we now need to prepare the concept to make it suitable for floor plan and trajectory analysis. We take a floor plan given as a bitmap, thus a 2D map, calculate the discrete isovist

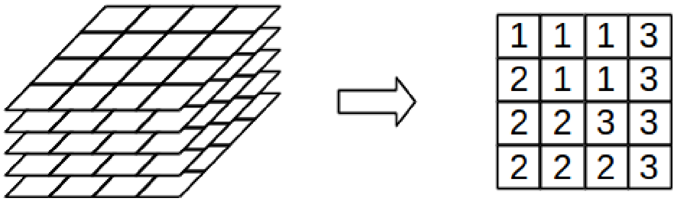


Fig. 3. Transformation of an isovist feature stack to a feature map.

features for each walkable pixel, and stack the four measures over the 2D space in order to get a isovist feature stack. The stack consisting of the occupancy grid and four isovist features for every x and y cell is visualized in Figure 3, left-hand side.

C. Feature Maps

The first application of the isovist feature stack is the analysis of the underlying floor plan. For this, we create a feature map by clustering the four features using a clustering algorithm like k-means [18], for example. Using this approach, we compress the feature stack into a feature map storing the cluster IDs (see right-hand side of Figure 3). Additionally, this representation can be considered as a dasymetric mapping [19], since it represents data as stepped statistical surfaces. It is not a choropleth map because the shape of the data polygons are based on the collected and represented data and not on predefined areal units.

D. Trajectory Annotation

The second application of the isovist feature stack is the analysis of routes inside the underlying floor plan. Given a set of routes, each as a vector of x and y coordinates, we select the appropriate features from the feature stack. The approach is now similar to the one before, but instead of clustering we investigate the local minima and maxima of the resulting four time series. For calculating the extrema we need a parameter μ, the size of the sliding window. The implementation of our algorithm automatically highlights the resulting extrema in both, the feature space (time series) and the observation space (floor plan).

IV. Evaluation

We base our evaluation on a floor plan of the main building of the Technische Universität München (TUM). This real world scenario is appropriate because of its complexity due to numerous entries, rooms, hallways, and a patio. We have performed several analysis, below there are exemplary and representative results in form of a route that traverses the building in a large circle (see for example Figure 5b).

A. Error Distribution

We want to examine the difference between the discrete and exact isovist features, i.e. the error of the discretization. Since there are different scales we have to use the normalized root-mean-square error (NRMSE). We have calculated the six exact isovist features using a shape file of the floor plan and performed a grid tesselation such that one pixel corresponds to $0.5m$ in the real environment. Furthermore, we have calculated the four discrete isovist features using several parameter configurations, each indicated in the respective paragraph.

TABLE I
NRMSE of Discrete Isovist Features

α	γ	A_x	$M_{2,x}$	$M_{3,x}$	N_x
1	∞	0.34	0.29	0.05	0.04
5	∞	0.32	0.29	0.08	0.08
20	∞	0.25	0.32	0.27	0.23
60	∞	0.15	0.30	0.37	0.28
1	300	0.34	0.33	0.23	0.13
5	300	0.25	0.29	0.28	0.24
20	300	0.25	0.29	0.28	0.24
60	300	0.16	0.32	0.37	0.28

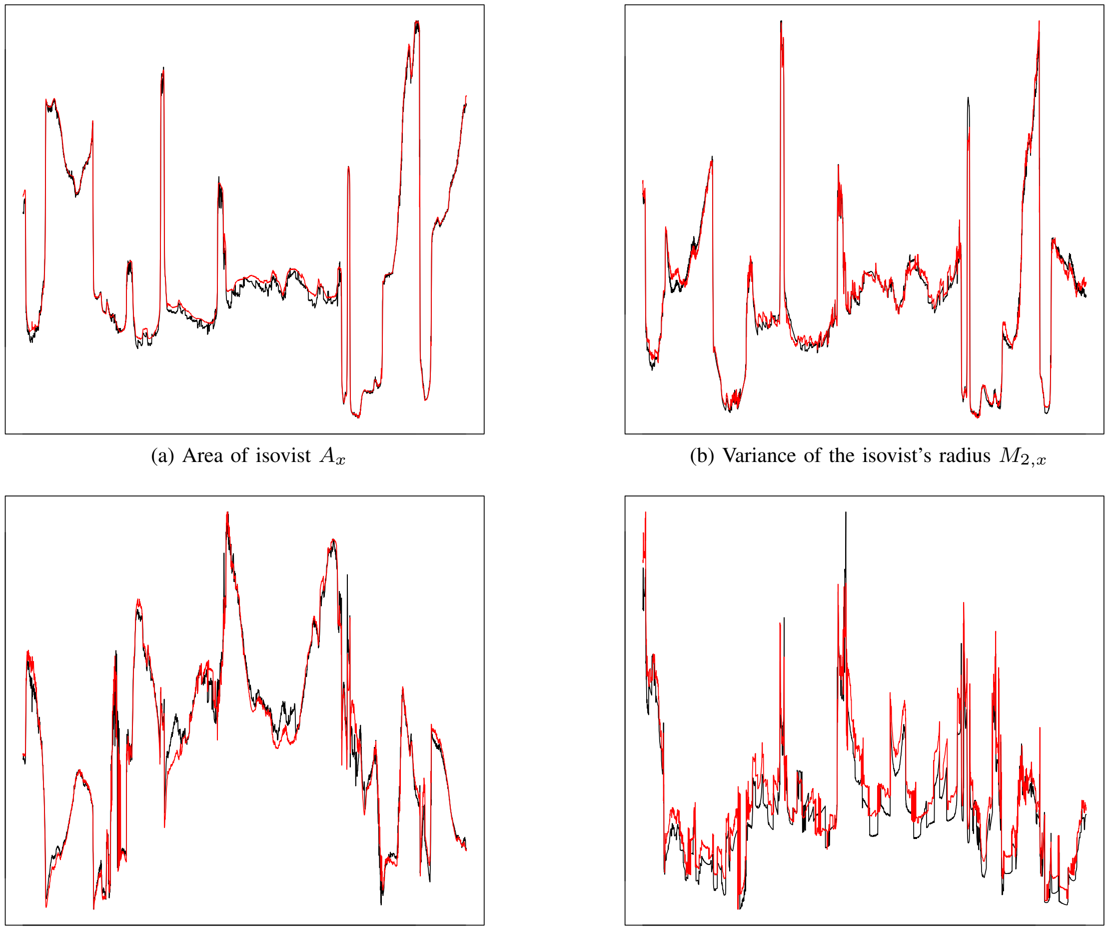
Table I shows the NRMSE of the discretization using four different values for the ray angle α and two different values for the length threshold γ. To get a better impression, Figure 4 shows the resulting features for configuration $\alpha = 1$ and $\gamma = \infty$. The x-axis represents the time, i.e. the progress of the mentioned route traversing the building, and the y-axis indicates the corresponding value of the discrete isovist feature at that particular point.

B. Trajectory Annotation

We now perform an analysis of the measures from a time series perspective. The setup is still based on the route traversing the TUM building in a circle, see for example 5b. For each point of the route we get the corresponding exact and discrete isovist features and highlight the local minima and maxima in both, time series representation and the floorplan. As already stated, there is the parameter μ representing the size of the sliding window for extrema calculation. For a better presentation we fix $\mu = 300$. At first we want to highlight and discuss the results for area and variance of exact isovist.

The values regarding the area (A_x) of the exact isovists and the visualization of the minima and maxima can be seen in Figures 5a and 5b, respectively. The greatest maximum (last one) is located at the spot where the route traverses the patio having a far view, right before it enters the building again. A quite interesting scope is the very stable part between the 3rd and 4th maximum, see Figure 5a around the middle and in Figure 5b the long floor going upright. The hallway is very monotone, that is why we experience constancy in both, feature space and observation space. The two lowest minima (the two last ones) are the spots of the route where a pedestrian would be cramped.

The values regarding the variance ($M_{2,x}$) of the exact isovists and again the visualization of the minima and maxima can be seen in Figures 5c and 5d, respectively. A kind observation is the fact that the maxima correspond exactly to passed or traversed doors. The highest minimum (4th) can be regarded as a "diagonal" door and gets highlighted as well (see Figure 5d, around the middle of the left-handed upright hallway), although not as a maximum.



(a) Area of isovist A_x

(b) Variance of the isovist's radius $M_{2,x}$

(c) Skewness of the isovist's radius $M_{3,x}$

(d) Circularity of the isovist N_x

Fig. 4. Comparison of the exact isovist features (black line) and the discrete isovist features (red line) using parameter configuration $\alpha = 1$ and $\gamma = \infty$. The x-axis represents the time, i.e. the progress of the route traversing the building, and the y-axis indicates the corresponding value of the discrete isovist feature at that particular point.

After discussing the values of the exact isovist, we now turn to the discretization. For this, we set $\gamma = \infty$ (unrestricted rays), $\alpha = 1$ $(360/1 = 360$ rays) and again $\mu = 300$. See Figure 6 showing the area (A_x) and variance $(M_{2,x})$ of the discrete isovists in both, the feature space and observation space. It is easy to see that the time series of the exact and discrete isovists have got a very similar process. Accordingly, the positions of the extrema relate strongly to the ones of the exact isovists. Thus, the discussion of the exact values holds here as well.

V. Conclusion

In this paper we proposed to take simple geospatial trajectories and floor plans to annotate them by using features that represent the local environment. Isovists are suitable means for this, as they are used as a numerical representation of spatial impressions of humans since decades.

We have introduced discrete isovists, i.e. approximated exact isovists using a simple ray casting approach. Furthermore, the resulting discrete isovist features have been used to create an isovist feature stack which, in turn, has been utilized to perform a semantic evaluation of routes in an indoor navigation scenario. It is apparent that isovists can be used to annotate routes.

As future work we envision to perform more sophisticated information retrieval on the isovist feature stack, the simple implementation of the remaining two isovist features, and the additional use of circular statistics [20].

Acknowledgment

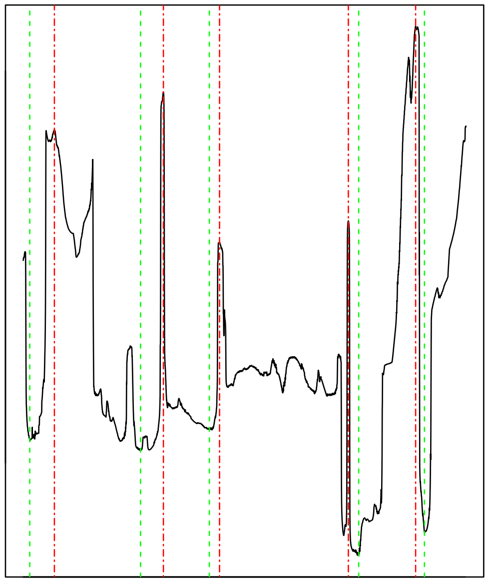
The authors would like to thank Hao Lyu, who provided the GIS-based exact isovist implementation and related datasets as well as for his helpful comments on this study.

References

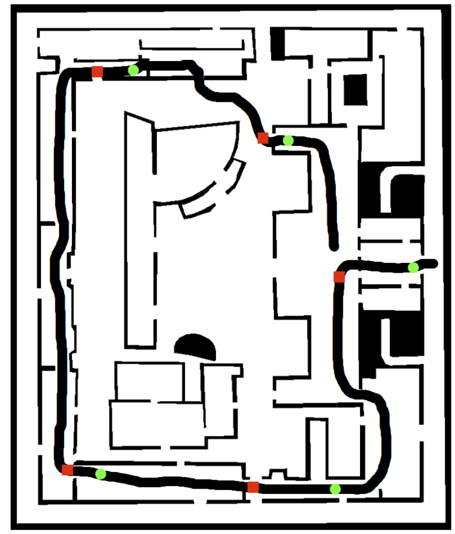
[1] A. Küpper, *Location-based Services: Fundamentals and Operation*. John Wiley & Sons, Ltd, 2005.

[2] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck, "Route planning in transportation networks," *arXiv preprint arXiv:1504.05140*, 2015.

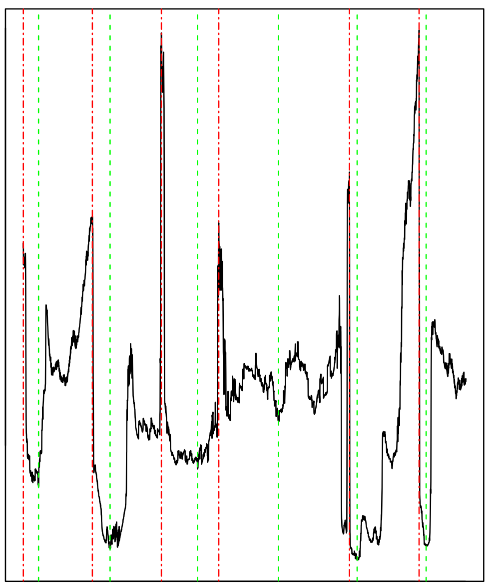
[3] A. Brimicombe, "Gis: Where are the frontiers now?" in *Proceedings of GIS-2001, International Conference and Exhibition*, 2002, pp. 33–45.



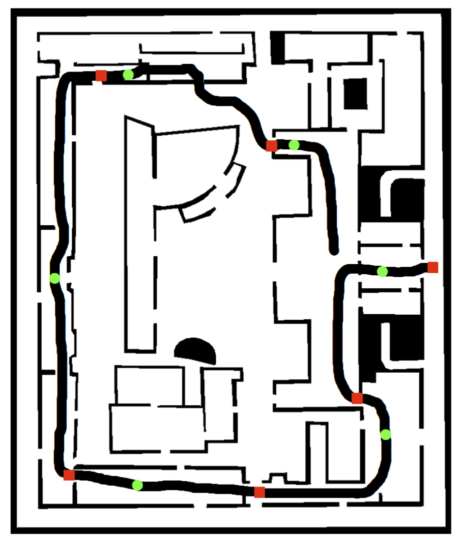
(a) Area of isovist (A_x) in feature space



(b) Area of isovist (A_x) in observation space



(c) Variance of radius ($M_{2,x}$) in feature space



(d) Variance of radius ($M_{2,x}$) in observation space

Fig. 5. Local extrema of exact area and variance of isovist. Green lines/circles indicate local minima and red lines/circles indicate local maxima.

[4] H. Moravec and A. Elfes, "High resolution maps from wide angle sonar," in *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 1985, pp. 116–121.

[5] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[6] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual navigation for mobile robots: A survey," *Journal of Intelligent and Robotic Systems*, vol. 53, no. 3, pp. 263–296, 2008.

[7] S. Thrun, "Learning occupancy grid maps with forward sensor models," *Autonomous Robots*, vol. 15, no. 2, pp. 111–127, 2003.

[8] M. Werner, "Indoor location-based services: Prerequisites and foundations," 2014.

[9] M. L. Benedikt, "To take hold of space: Isovists and isovist fields," *Environment and Planning B: Planning and Design*, vol. 6, no. 1, pp. 47–65, 1979.

[10] C. R. Tandy, "The isovist method of landscape survey," in *Methods of Landscape Analysis*, C. Murray, Ed. Landscape Research Group, London, 1967, pp. 9–10.

[11] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger, "The r*-tree: An efficient and robust access method for points and rectangles," in *ACM SIGMOD Record*, vol. 19, no. 2. ACM, 1990, pp. 322–331.

[12] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems journal*, vol. 4, no. 1, pp. 25–30, 1965.

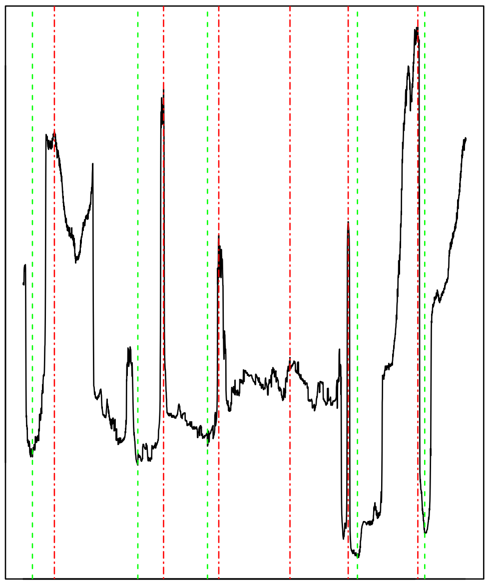
[13] M. Werner and S. Feld, "Homotopy and alternative routes in indoor navigation scenarios," in *Proceedings of the 5th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2014)*. IEEE, 2014, pp. 230–238.

[14] M. A. Armstrong, *Basic Topology*. Springer, 1983.

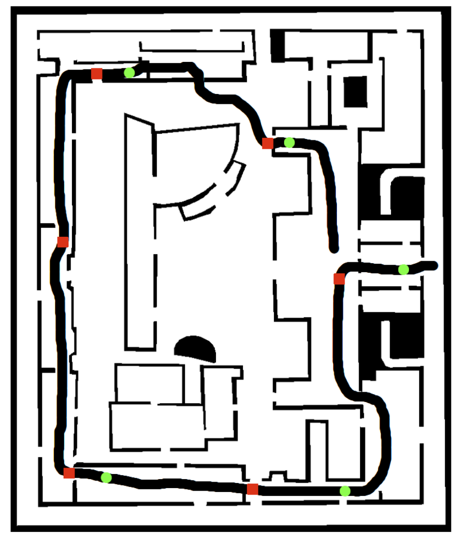
[15] G. E. Bredon, *Topology and Geometry*. Springer, 1993.

[16] S. Feld, "Scoring of alternative routes using implicit building topologies," in *Proceedings of the Science and Information Conference (SAI 2015)*. IEEE, 2015, pp. 329–336.

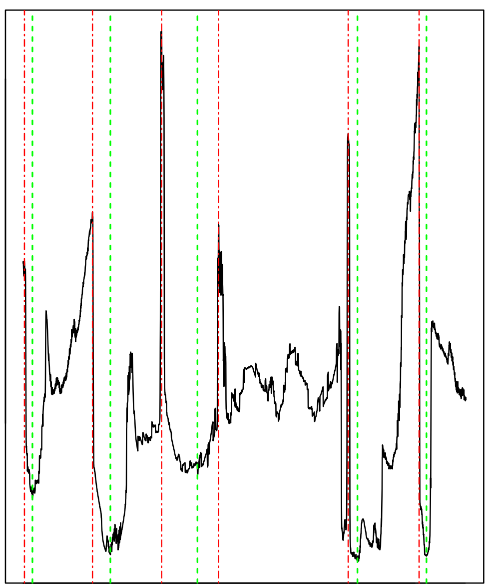
[17] S. Feld, M. Werner, M. Schönfeld, and S. Hasler, "Archetypes of alternative routes in buildings," in *Proceedings of the 6th International*



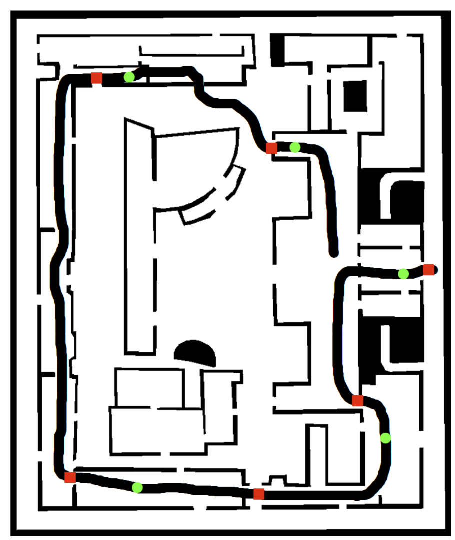
(a) Area of isovist (A_x) in feature space



(b) Area of isovist (A_x) in observation space



(c) Variance of radius ($M_{2,x}$) in feature space



(d) Variance of radius ($M_{2,x}$) in observation space

Fig. 6. Local extrema of discrete area and variance of isovist using $\alpha = 1$ and $\gamma = \infty$. Green lines/circles indicate local minima and red lines/circles indicate local maxima.

Conference on Indoor Positioning and Indoor Navigation (IPIN 2015). IEEE, 2015, pp. 1–10.

[18] J. A. Hartigan, *Clustering algorithms*. John Wiley & Sons, Inc. New York, NY, USA, 1975.

[19] A. Petrov, “One hundred years of dasymetric mapping: Back to the origin,” *The Cartographic Journal*, vol. 49, no. 3, pp. 256–264, 2012.

[20] S. R. Jammalamadaka and A. Sengupta, *Topics in Circular Statistics*. World Scientific, 2001, vol. 5.