Clustering of architectural floor plans: a comparison of shape representations

Eugénio Rodrigues[a,*], David Sousa-Rodrigues[b], Mafalda Teixeira de Sampayo[c], Adélio Rodrigues Gaspar[d], Álvaro Gomes[e], Carlos Henggeler Antunes[e]

[a]*ADAI, LAETA, University of Coimbra*
Rua Luís Reis Santos, Pólo II, 3030-788 Coimbra, Portugal
[b]*Centre of Complexity and Design, Faculty of Mathematics, Computing and Technology*
The Open University, Milton Keynes, MK7 6AA, United Kingdom
[c]*CIES, Department of Architecture, Lisbon University Institute*
Av. Forças Armadas, 1649-026 Lisboa, Portugal
[d]*ADAI, LAETA, Department of Mechanical Engineering, University of Coimbra*
Rua Luís Reis Santos, Pólo II, 3030-788 Coimbra, Portugal
[e]*INESC Coimbra, Department of Electrical and Computer Engineering, University of Coimbra*
Rua Luís Reis Santos, Pólo II, 3030-290 Coimbra, Portugal

Abstract

Generative design methods are able to produce a large number of potential solutions of architectural floor plans, which may be overwhelming for the decision-maker to cope with. Therefore, it is important to develop tools which organise the generated data in a meaningful manner. In this study, a comparative analysis of four architectural shape representations for the task of unsupervised clustering is presented. Three of the four shape representations are the Point Distance, the Turning Function, and the Grid-Based model approaches, which are based on known descriptors. The fourth proposed representation, Tangent Distance, calculates the distances of the contour's tangents to the shape's geometric centre. A hierarchical agglomerative clustering algorithm is used to cluster a synthetic dataset of 72 floor plans. When compared to a reference clustering, despite good perceptual results with the use of the Point Distance and Turning Function representations, the Tangent Distance descriptor (Rand index of 0.873) provides the best results. The Grid-Based descriptor presents the worst results.

Keywords: unsupervised clustering, floor plan designs, hierarchical clustering, shape representation, descriptors

1. Introduction

Generative design methods are commonly used in architectural design. These methods have several applications in the design of structural elements, facade layout, space planning, optimisa-

[*] Corresponding author.
Email address: eugenio.rodrigues@gmail.com (Eugénio Rodrigues)

tion of building form, replication of architectural styles, and urban design. The main goal is to assist building design practitioners in exploring a larger set of solutions, which a traditional trial-and-error process could never achieve. However, one of the drawbacks is that they may produce an excessive number of solutions for a human to cope with; moreover, it is just not feasible to rate solutions according to a performance criterion and then select the top-ranked ones, especially for unclear and subjective problems. An alternative approach is to organise the generated data into groups determined by common features. This allows the decision-maker to compare group types before analysing specific solutions. Therefore, to facilitate the decision-maker's task of comparison and selection, this paper presents an unsupervised clustering technique using four different shape representations. The method and the performance of these shape descriptors is analysed in a computer generated architectural floor plan showcase.

This is a typical task for machine learning techniques. In the field of machine learning there are two main subfields dealing with organisation of data: classification and clustering. While the former is used to label data according to pre-defined classes, the latter deals with unlabelled data and the task is usually to create partitions in the data while making coherent groups according to some defined metric. This is a process of identifying structures in unlabelled datasets regardless of the data type. Han and Kamber [1] classified clustering techniques into five categories: partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods.

Clustering techniques have been applied in diverse areas. Some of the most relevant applications include the classification of textual documents [2], document navigation for search engine optimisation [3–5], resource project scheduling [6], point cloud simplification [7, 8], time series analysis and clustering [9], image clustering [10], face expression [11], database retrieval of mechanical objects [12, 13], and sketch recognition [14].

The clustering of objects, according to their shape, has also been previously applied in diverse fields. The correct representation of the shape has a significant impact on the matching correctness of the algorithms [15]. For instance, Chang et al. [16] proposed a shape recognition scheme where the representation corresponds to the distance of feature points in the shape's boundary to the centroid. This shape representation presents the property of being invariant to translation as the boundary is fixed in relation to the centroid independently of its global position. As the distances of the feature points are ordered and divided by a minimum distance, this also results in invariance to scaling, rotation, and reflection. Instead of only considering the shape feature points, Yankov and Keogh [17] used the entire contour for the shape representation and a nonlinear reduction technique to cluster pathological cells.

Arkin et al. [18] represented a polygonal shape by its turning function. The shape descriptor consists in measuring the angle of the counter-clockwise tangent to the $x-$axis in each of the feature points in the polygon. Therefore, the values vary between $-\pi$ and π. As the polygon is scaled to have a length of 1, in addition to being translation invariant, the representation is also invariant to scaling. However, results depend on the starting point and the polygon's rotation and reflection.

Sajjanhar and Lu [19] suggested a grid-based representation where a shape is placed, rotated, and scaled to fit a square grid. For each cell in the grid a binary value is determined: 0 for empty and 1 for filled. Although this representation guarantees translation and scale invariance, if the grid is adaptive, the scaling is only invariant to one of the axes—the rotation invariance is dependent on the rotation of the grid to match the same shape orientation. Also, as may be expected, the results vary according to the grid size, as this changes the capability to capture the shape's details.

Siddiqi et al. [20] used a shock graph to capture the effects on the bounding contours of the singularities in the shape structure. The graph is determined according to a set of rules in a shock graph grammar which reduces it to a rooted shock tree. A recursive algorithm is then used to match two shock trees, starting from the root and proceeding through the subtrees in a depth-first approach.

Belongie et al. [21] presented an approach to measure similarity of shapes by considering the distribution of the remaining points in each reference point. As corresponding points in two similar figures have similar contexts, a transformation is used to align two shapes. The dissimilarity between them is calculated by summation over the errors between the corresponding points in the transformation.

Aiming to retrieve shapes from a database, which are similar to a query shape, Tan et al. [22] proposed a new representation based on a centroid-radii approach. According to the authors, this approach allows the modelling of convex, concave, and hollow shapes. The representation consists of a set of vectors, each one measured at regular intervals from the centroid of a concentric ring.

In Klassen et al. [23], the shapes are considered to be planar closed curves represented either as direction functions or as curvature functions. In this manner, shapes may be modelled as stretchable, compressible, and bendable strings along their extensions that are constructed from spaces of parametric curves [24, 25]. Geodesics are used to determine the dissimilitude between shapes.

Ling and Jacobs [26] classified shapes by using an inner-distance to build the shape representation of the structure or articulation parts. The inner-distance is the length of the shortest path between two reference points on the shape boundary and allows the creation of articulation invariant representations.

Shen et al. [27] proposed a method to group planar figures by their skeleton graph. The clustering is carried out by determining the common internal shape structure that belongs to the same cluster. The data is grouped by using an agglomerative clustering algorithm.

In architecture, Cha and Gero [28] investigated shape patterns to determine if any similarities, relationships, and physical properties could be recognised. de las Heras et al. [29] used run length histograms as a perceptual representation of floor plans made by architects. This approach allows the retrieval of designs with similar properties from a database. Dutta et al. [30] used a graph-based method to identify symbols in floor plans such as furniture and openings.

However, despite all of the mentioned approaches/methods, the use of clustering techniques has yet to be used to group designs in the case of automatic generation of floor plans. In a previous study, Sousa-Rodrigues et al. [31, 32] conducted an online survey directed at design and construction experts—mostly architects, engineers and architecture undergraduates—in which the majority of respondents considered the overall shape of floor plans as the most important similitude feature. This highlights the importance of having perceptually accurate algorithms for the automation of this task.

In this paper, four shape representations are studied as floor plan design descriptors under the same settings. All descriptors are vectors of similar length, and all are used to partition the same dataset with the same clustering algorithm. Three of the four shape representations are known descriptors: these are the distance to centroid [16], the turning function [18], and the grid-based model [19]. The fourth and last shape descriptor is a novel representation specifically created to capture orthogonal floor plan shapes. It consists in calculating the distance of the tangent lines to the geometric centre of the shape. The clustering procedure is an agglomerative hierarchical algorithm with Ward linkage [33] and Euclidean distance as a dissimilarity measure. The advantages and disadvantages of each shape representation are analysed in a showcase with 72 floor plan designs. These designs were generated using a specific algorithm, named Evolutionary Program for the Space Allocation Problem (EPSAP) [34–36]. The EPSAP algorithm generates alternative floor plans according to the user's specifications.

After this introductory section, section 2 describes the methods applied to the clustering of the floor plans designs. In section 3 the results for a showcase of a single-family house are presented and compared to a reference clustering partition. The discussion of the relevant results follows in section 4, as well as the analysis of the applicability of the descriptors. Finally, conclusions are drawn and future work is outlined in section 5.

2. Methodology

To determine the most suitable shape representation to be used in the cluster of orthogonal floor plans, three shape descriptors inspired by previous works and one new descriptor were implemented. These descriptors have the same vector length and shape matching algorithm using the Euclidean distance to calculate the dissimilitude between the shapes. Therefore, the computational burden is equal for the four approaches. A specific algorithm generated a dataset of floor plan designs. This synthetic dataset does not require a pre-processing mechanism for denoising the shapes, nor the application of a dimensionality reduction technique. Therefore, the focus is on the perceptual quality of the results of each shape descriptor.

2.1. Shape representation

The representation of continuous features plays an important role in machine learning techniques, either because the machine learning technique itself requires a nominal feature space—nominal features describe qualitative aspects that do not share a natural ordering relationship—or because discretisation allows for better results in the machine learning technique. The research on dataset discretisation for machine learning is vast and beyond the scope of this paper, but it is important to mention that such algorithms usually aim to maximise the interdependency between discrete attribute values and class labels, as this minimises the information loss due to the discretisation process. The process has to balance the trade-offs between these two goals and many studies have shown that several machine-learning techniques benefit from it [37–40].

In this study, the four descriptors are designed to have similar features. These are invariant to translation and scaling but sensitive to rotation and reflection. A descriptor variant that considers independent scaling of x-and y-coordinates was also analysed. The reason for these features is that, despite floor plans being generated on a blank canvas, human experts continue to have a notion of north-south and east-west framework, thus a rotated or a reflected floor plan is considered as an alternative design. Buildings have a strong relation with their environment and their form depends on the surrounding buildings, landscape, solar orientation, and so on. However, because there are no visual references around each floor plan, translation does not affect the human perception of that shape. As a result, rotation and reflection were considered as features that influence the clustering result. Nevertheless, invariance to rotation and reflection could be easily achieved by ordering the descriptor vector or considering the distribution of these values.

2.1.1. Point Distance (PD) descriptor

Based on Chang et al.'s [16] shape representation, the Point Distance (PD) descriptor has points marked on the shape silhouette at equal segment lengths. The starting point is the nearest shape perimeter point in relation to the top-left corner of the shape bounding box and the points are distributed in a counter-clockwise direction. Our implementation differs from Chang et al.'s representation as the reference point is not the shape's centroid, which is defined as the average of the $x-$ and $y-$coordinates of all perimeter points, but instead considers the geometric centre of the bounding box as the reference point. The shape descriptor is then a vector of normalised values—corresponding to the distance from the reference point to the ordered perimeter points divided by the longest point distance.

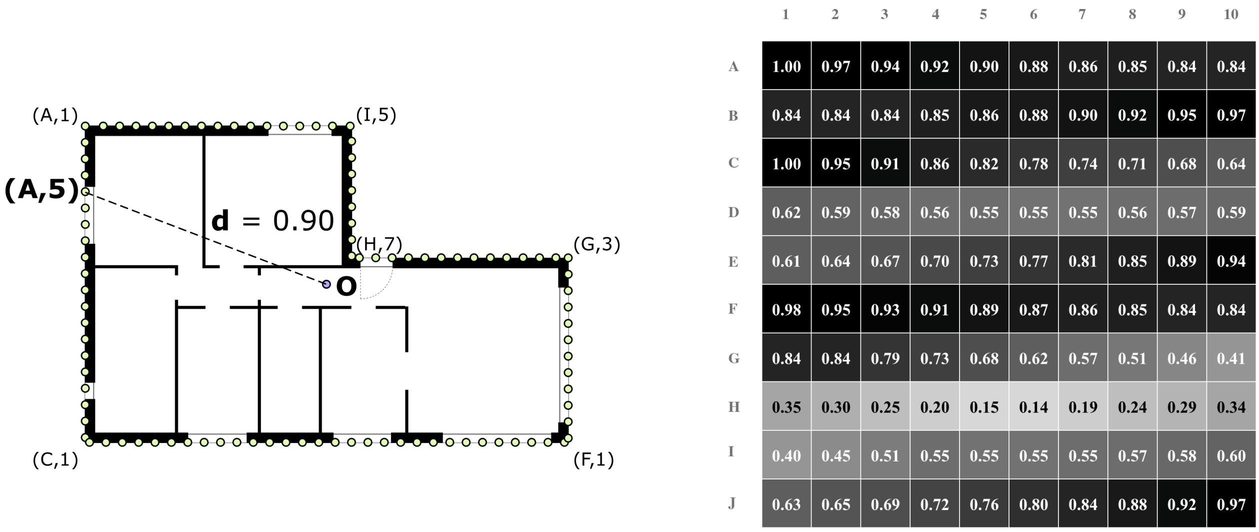
Figure 1a illustrates an example of the marked perimeter point (A,5) and its normalised distance to the centre (0.90). The example represents the descriptor variant where the $x-$coordinate and $y-$coordinate scales are preserved. Figure 1b depicts the representation vector of normalised values ranging from 0 (white) to 1 (black) in a gradient matrix form[1], where the first vector point is (A,1) and concludes in point (J,10). In the floor plan image, the wall corners are marked with the corresponding matrix point to depict the counter-clockwise order of the marked points.

2.1.2. Turning Function (TF) descriptor

The second shape descriptor is based on Arkin et al.'s [18] turning function. This consists in determining the counter-clockwise angle to the $x-$axis of a tangent in each feature point along the shape contour. The feature points are marked at equal distances.

Figure 2a depicts an example where the turning function angle is measured at point (B,3), with the value of $3\pi/2$, in the descriptor variant of preserved aspect ratio. The feature points start with the initial point (A,1), which is the nearest perimeter point to the top-left corner

[1] The gradient matrix of the four representations is used only for visual comparison of different floor plans. The agglomerative hierarchical algorithm uses each data point as a $1-$dimensional vector.



	1	2	3	4	5	6	7	8	9	10
A	1.00	0.97	0.94	0.92	0.90	0.88	0.86	0.85	0.84	0.84
B	0.84	0.84	0.84	0.85	0.86	0.88	0.90	0.92	0.95	0.97
C	1.00	0.95	0.91	0.86	0.82	0.78	0.74	0.71	0.68	0.64
D	0.62	0.59	0.58	0.56	0.55	0.55	0.55	0.56	0.57	0.59
E	0.61	0.64	0.67	0.70	0.73	0.77	0.81	0.85	0.89	0.94
F	0.98	0.95	0.93	0.91	0.89	0.87	0.86	0.85	0.84	0.84
G	0.84	0.84	0.79	0.73	0.68	0.62	0.57	0.51	0.46	0.41
H	0.35	0.30	0.25	0.20	0.15	0.14	0.19	0.24	0.29	0.34
I	0.40	0.45	0.51	0.55	0.55	0.55	0.55	0.57	0.58	0.60
J	0.63	0.65	0.69	0.72	0.76	0.80	0.84	0.88	0.92	0.97

(a) (b)

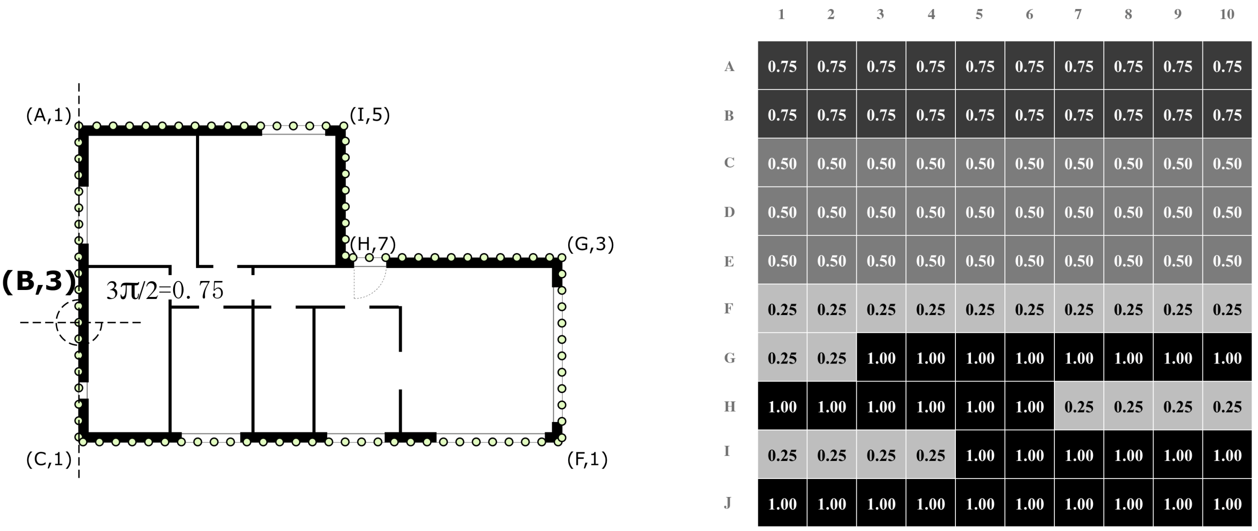
Figure 1: Point Distance (PD) descriptor. (a) Example of the normalised distance for the point (A,5) with value of 0.90, which corresponds to its real distance divided by the longest distance of all silhouette points. The wall corners are marked with the matrix index to depict the counter-clockwise order of the feature points. (b) Vector in the form of a gradient matrix (white is 0 and black is 1) of the normalised distances.

of the shape bounding box, and contours the shape silhouette in a counter-clockwise manner. Therefore, the values vary between 0 and 2π that are then normalised to have values ranging from 0 to 1. Figure 2b illustrates the vector of the Turning Function (TF) descriptor as a gradient matrix. As the floor plans are orthogonal, the shape edges only take on four possible values $\{\pi/2, \pi, 3\pi/2, 2\pi\} = \{0.25, 0.50, 0.75, 1.00\}$.

2.1.3. Grid-Based (GB) descriptor

The Grid-Based descriptor is inspired on Sajjanhar and Lu's [19] work and consists in placing the shape under a square grid parallel to the exterior walls of the floor plans. For each cell in the grid, the centre may (1) or may not (0) be occupied by the shape area. The representation is a vector of binary values with the length equal to the number of cells. The values correspond to reading the grid from left-to-right and top-to-bottom.

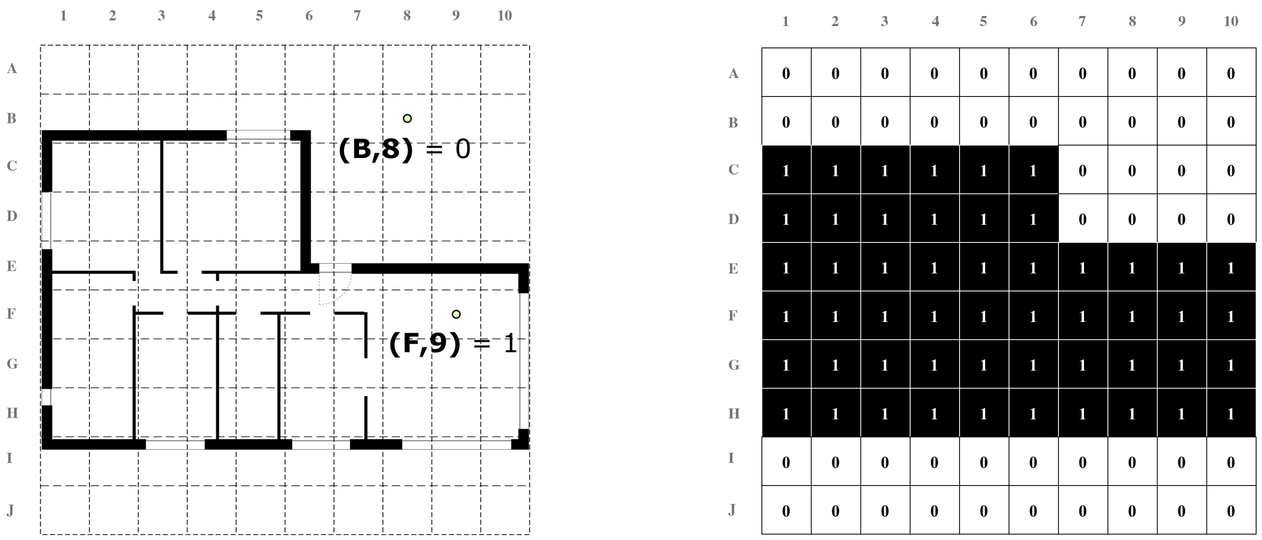
Figure 3a illustrates an example of a floor plan overlaid by a grid. In the example, point (B,8) has a value of 0 while (F,9) has a value of 1 depending on whether the floor plan area is under that cell centre or not. Figure 3b represents the corresponding binary vector as a matrix. Each matrix entry has the corresponding value in the overlaid grid in the floor plan.



	1	2	3	4	5	6	7	8	9	10
A	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75
B	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75
C	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
D	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
E	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50
F	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25
G	0.25	0.25	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
H	1.00	1.00	1.00	1.00	1.00	1.00	0.25	0.25	0.25	0.25
I	0.25	0.25	0.25	0.25	1.00	1.00	1.00	1.00	1.00	1.00
J	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

(a) (b)

Figure 2: Turning Function (TF) descriptor. (a) Example of the measuring angle in point (B,3) that has the value of 0.75, which corresponds to $3\pi/2$. The wall corners are marked with the matrix index to depict the counter-clockwise order of the feature points. (b) Vector in the form of a gradient matrix, where 0 is white and 1 is black, for angles ranging from 0 to 2π.



	1	2	3	4	5	6	7	8	9	10
A	0	0	0	0	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0	0	0
C	1	1	1	1	1	1	0	0	0	0
D	1	1	1	1	1	1	0	0	0	0
E	1	1	1	1	1	1	1	1	1	1
F	1	1	1	1	1	1	1	1	1	1
G	1	1	1	1	1	1	1	1	1	1
H	1	1	1	1	1	1	1	1	1	1
I	0	0	0	0	0	0	0	0	0	0
J	0	0	0	0	0	0	0	0	0	0

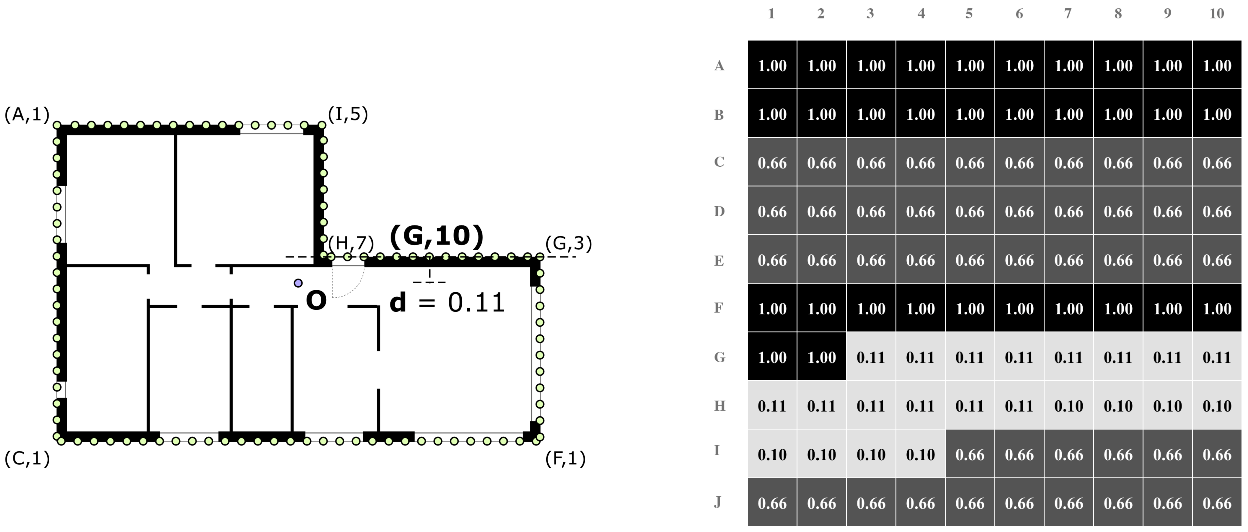
(a) (b)

Figure 3: Grid-Based (GB) descriptor. (a) Example of two point measurements. Point (B,8) is outside the floor plan area thus having the value of 0. Meanwhile, point (F,9) falls within the floor plan area and has a value of 1. (b) Vector in the form of a matrix (white is 0 and black is 1) depicting the corresponding cell value in the overlaid grid in the floor plan. Only the cell centre is used to measure the presence of the floor plan.

2.1.4. Tangent Distance (TD) descriptor

The Tangent Distance (TD) descriptor consists in determining the distance of a straight-line tangent to the shape contour to the bounding box centre. As floor plans are orthogonal shapes, ultimately the tangent line coincides with the exterior wall. The shape has its perimeter marked with points at regular length intervals starting on the nearest point on the shape perimeter to the top-left bounding rectangle. In every point, a straight line is drawn tangent to the shape and the distance is measured to the centre point. The vector has its values normalised—measured distance divided by the longest distance.

Figure 4a depicts an example of the descriptor variant for preserved aspect ratio. The feature point (G,10) has a normalised distance value of 0.11 of its tangent to the centre. Figure 4b illustrates the resulting vector in the form of a gradient matrix.



	1	2	3	4	5	6	7	8	9	10
A	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
B	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
C	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66
D	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66
E	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66
F	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
G	1.00	1.00	0.11	0.11	0.11	0.11	0.11	0.11	0.11	0.11
H	0.11	0.11	0.11	0.11	0.11	0.11	0.10	0.10	0.10	0.10
I	0.10	0.10	0.10	0.10	0.66	0.66	0.66	0.66	0.66	0.66
J	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66	0.66

(a)

(b)

Figure 4: Tangent Distance (TD) descriptor. (a) Example of the tangent distance for the feature point (G,10), which has the normalised distance value of 0.11. The horizontal walls corners are marked following the counter-clockwise order. (b) Vector in the form of a gradient matrix, where 0 is white and 1 is black.

2.2. Clustering algorithm

The dataset was clustered using an agglomerative hierarchical algorithm with Ward linkage [33] and the Euclidean distance as the dissimilarity measure between different floor plan designs (feature vectors). Hierarchical clustering is based on the assumption that there is maximal quantifiable information when a set of elements is ungrouped, and that this information is captured by an objective function. In the case of agglomerative hierarchical clustering, the algorithm starts by considering as many clusters as the available data points and placing each data point in a cluster. It proceeds by merging two existing clusters that optimise an objective function. In this case the function is a variance criterion minimising the total within-cluster variance. At each step of the agglomerative process, the two clusters to be merged are dependent on the least increase in the total within-cluster variance. The process then proceeds iteratively until all clusters are grouped into a single global cluster.

Although the linkage criterion used in hierarchical clustering can be of different types, Ward's complete linkage aims to find compact clusters and was therefore preferred in this work. A similar linkage is the complete linkage clustering [41], where the distance between two different clusters is calculated by considering all pair-wise interactions between the elements in the two clusters. It then uses the distance of the pair of points that is farthest away from each other as the distance between the two clusters. It also aims to create compact clusters and to compute faster. For large populations it is an alternative to the Ward's criterion as it is faster. In this work, all results employed the Ward's criterion.

There are several measures available to determine the dissimilitude of two descriptor vectors [42]. In this work the dissimilitude between two feature vectors was calculated by the Euclidean distance for $N-$dimensions, with N being the length of the feature vector describing the floor plan design.

2.3. Synthetic dataset

The dataset of floor plan designs was created using a generative design algorithm, named the Evolutionary Program for the Space Allocation Program (EPSAP) [34–36]. This algorithm combines an Evolution Strategy (ES) technique and a Stochastic Hill Climbing (SHC) method in a two-stage approach. The EPSAP is capable of generating multi-story floor plans where parametric, non-rigid, and non-fixed vertical circulation elements evolve during the search process in interaction with the remaining spaces.

From a set of requirements defined by the user and given as input (see subsection 3.1 for an example of the required input information), the generative design process initialises by creating, at the first ES generation, randomly distributed and dimensioned rectangles (each corresponding to a room) in the 2-dimensional plan—each storey has its own 2-dimensional plan. Each design solution is evaluated with a weighted sum of several objectives. These objectives are connectivity (interior doors), adjacency (proximity between rooms), room dimensions and area (according to minimum size of the smallest rectangle side and minimum floor area, respectively), compactness of the floor plan, room overflow in relation to a building boundary (when specified by the user), opening dimensions (to satisfy minimum width and window-to-floor ratio), and opening orientation (when specified by the user).

At every ES generation, the SHC method is called to randomly transform the different architectural elements in the floor plan (rooms, stairs, elevators, cluster of spaces, openings, walls, and the floor plans as a whole). The SHC method applies geometric actions such as translation, reflection, rotation, stretching, alignment of elements, permutation of element type, and changes to the element's orientation. The transformation action randomly selects the element, direction, and magnitude of change from the admissible geometric values. Then, the candidate solutions are evaluated. If the action produces an equal or better solution, the change is preserved, otherwise it is discarded. The SHC stage continues iteratively until reaching the SHC termination criterion—the difference between the moving average and the last iteration of the best individuals' average performance is greater than a defined threshold. Then, solutions having better performance than the average of the population are preserved for the next ES generation, while the remaining ones are discarded and substituted with new randomly generated ones, thus initiating a new ES cycle. When the ES termination criterion is reached, the algorithm stops and displays the results to the user.

As the EPSAP produces a large number of alternative floor plans, some kind of aggregation mechanism is required to help users compare and analyse the generated solutions. This is the motivation for the development of this study as described in subsection 2.1.

3. Results

3.1. Showcase specifications

A single-family three-bedroom house was used as an illustrative example. In addition to the three bedrooms (R_{6-8}), a hall (R_1), a kitchen (R_2), a living room (R_3), a corridor (R_5), and two bathrooms (R_4 and R_9) were specified. Topologically, all spaces have connection to the hall or the corridor. The kitchen also has an interior door connecting to the living room. One of the bathrooms serves the public area of the house and the other is connected to the corridor of the private part of the house, which is connected to all bedrooms. The interior connectivity (M_{con}) is defined in Matrix (1), where 1 represents an interior door connecting two rooms and 0 indicates the absence of doors connecting them.

$$M_{con} = \begin{array}{c} \\ R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \\ R_7 \\ R_8 \\ R_9 \end{array} \begin{array}{c} \begin{array}{ccccccccc} R_1 & R_2 & R_3 & R_4 & R_5 & R_6 & R_7 & R_8 & R_9 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \tag{1}$$

All interior doors must have 0.90m wide except the living room doors, which are 1.40m. With the exception of the horizontal circulation spaces and one of the bathrooms, all remaining spaces have at least one window (the living room has two). The hall has one exterior door facing north (orientation up). No other topological requirement was added, such as opening orientation or space location on the floor plan.

The detailed showcase requirements are presented in Table 1, where the information relating to each room is listed. These include space name (M_{sn}), space function type (M_{st}, where 0 represents circulation spaces, 1 rooms, and 2 kitchens and bathrooms), minimum floor side dimension (M_{fd}), minimum floor area (M_{fa}), exterior opening width (M_{eow}) and height (M_{eoh}), space window-to-floor ratio (M_{wfr}), clear area in the outside of opening (M_{eoa}), exterior opening orientation (M_{eoo}), and interior doors minimum width (M_{idw}). The thicknesses of walls are 0.32m for the exterior wall (t_{ew}) and 0.11m for the interior wall (t_{iw}). The floor plan design (FPD) must have a construction area inferior to 200m^2 (a_c).

Using these requirements as input, the EPSAP algorithm ran a single time to generate 72 alternative floor plans from a population of 576 individuals (each individual is a candidate solution). The generative design process took 136s in a 2.8GHz Quad-core computer with 8GB of RAM. Multi-threading was used. The floor plans improved over a total of 1790 iterations by minimising penalties for not satisfying the user specifications. The best individual had a fitness of 98265.1 in the first iteration and 2.2 in the last iteration, which resulted from not attaining the aimed floor plan area.

3.2. Clustering results

As the purpose of this work was to provide the EPSAP algorithm with clustering capabilities to help the user deal with a large number of generated solutions, and because the type of shapes

Table 1: Case study specifications for spaces and openings.

Storey	Space					Ext. opening					Int. door
		M_{sn}	M_{st}	M_{fd}	M_{fa}	M_{eow}	M_{eoh}	M_{wfr}	M_{eoa}	M_{eoo}	M_{idw}
L_1	R_1	Hall	0	1.40m	5.0m^2	1.20m	2.00m		{1.80m, 3.00m}	North	0.90m
	R_2	Kitchen	2	2.60m	15.0m^2		1.00m	0.1	{3.00m, 3.00m}		0.90m
	R_3	Living room	1	4.00m	20.0m^2	{5.00m, 4.00m}	{2.40m, 2.40m}		{3.00m, 3.00m}		1.40m
	R_4	Bathroom	2	1.80m	3.0m^2						0.90m
	R_5	Corridor	0	1.40m	3.0m^2						0.90m
	R_6	Bedroom	1	3.50m	18.0m^2		1.00m	0.1	{3.00m, 3.00m}		0.90m
	R_7	Bedroom	1	3.00m	15.0m^2		1.00m	0.1	{3.00m, 3.00m}		0.90m
	R_8	Bedroom	1	2.70m	12.0m^2		1.00m	0.1	{3.00m, 3.00m}		0.90m
	R_9	Priv. Bathroom	2	1.80m	3.0m^2	0.60m	0.60m		{3.00m, 3.00m}		0.90m
$t_{ew} = 0.32m, t_{iw} = 0.11m,$ and $a_c <= 200m^2$.											

and resulting numbers are not known *a priori*, an unsupervised clustering approach was used. That is, the number of clusters does not depend on the real number of different shapes in the generated set but on the number of alternative solutions that the user wants or might analyse. As the complexity of the floor plans increases, the number of alternative shapes also grows, easily reaching numbers that become intractable for the decision-maker. The clustering mechanism is independent from the number of clusters and the number of floor plan designs, thus may be scaled up or down only affecting computation time. As the vector in every clustering process had the same length (100 values), the type of shape representation did not affect the performance of the algorithm. However, the results had significant differences depending on the shape descriptor.

During the preparatory work, a survey was conducted to determine which clustering features human experts use to group floor plans [31, 32]. The survey analysis determined the main features, such as shape and indoor room arrangement. However, human experts are generally inconsistent during the clustering process—for instance, the same individual may sometimes gather floor plans by shape and in other times by indoor space arrangement. This resulted in having groups where a floor plan A has similar shape as a floor plan B and the latter has the same internal arrangement as a floor plan C. However, C has no similarity whatsoever with A, despite the three being in the same cluster. Therefore, the results of the survey were not used as a ground truth due to this changing behaviour. As an alternative, a reference clustering was determined by typifying shapes from designs found in the dataset. Figure 5 depicts such partition (labelled from **A'** to **I'**) with the typified shape on the left of each group letter. There is the O-shape, four rotated L-shapes,

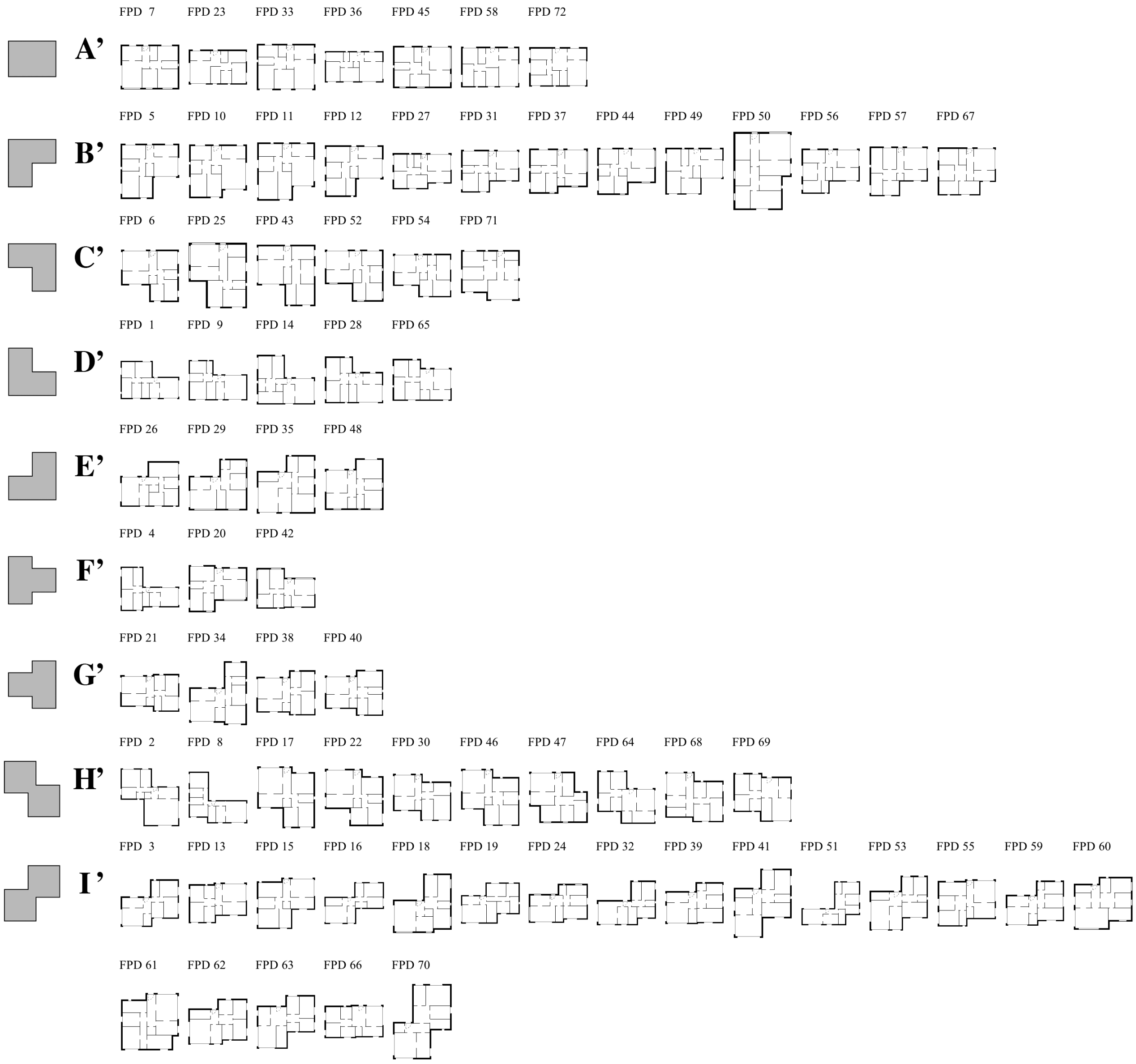

A'
FPD 7 FPD 23 FPD 33 FPD 36 FPD 45 FPD 58 FPD 72
B'
FPD 5 FPD 10 FPD 11 FPD 12 FPD 27 FPD 31 FPD 37 FPD 44 FPD 49 FPD 50 FPD 56 FPD 57 FPD 67
C'
FPD 6 FPD 25 FPD 43 FPD 52 FPD 54 FPD 71
D'
FPD 1 FPD 9 FPD 14 FPD 28 FPD 65
E'
FPD 26 FPD 29 FPD 35 FPD 48
F'
FPD 4 FPD 20 FPD 42
G'
FPD 21 FPD 34 FPD 38 FPD 40
H'
FPD 2 FPD 8 FPD 17 FPD 22 FPD 30 FPD 46 FPD 47 FPD 64 FPD 68 FPD 69
I'
FPD 3 FPD 13 FPD 15 FPD 16 FPD 18 FPD 19 FPD 24 FPD 32 FPD 39 FPD 41 FPD 51 FPD 53 FPD 55 FPD 59 FPD 60
FPD 61 FPD 62 FPD 63 FPD 66 FPD 70

Figure 5: Reference clustering and shape type by group.

two rotated T-shapes, and two reflected Z-shapes. Group **A'** (O-shape) has 7 designs; **B'** (top-left L-shape) has 13; **C'** (top-right L-shape) has 6; **D'** (L-shape) has 5; **E'** (reflected L-shape) has 4; **F'** (rotated left T-shape) has 3; **G'** (rotated right T-shape) has 4; **H'** (Z-shape) has 10; and, finally, **I'** (reflected Z-shape) has 20 designs.

Several measures have been proposed to determine the quality of the resulting groups and comparing those clusters with a reference group of the data. The measures of comparison have to be able to handle minor data perturbations as well as missing data, but remain sensitive enough when two clustering methods produce different results from the same data [43]. In Rand [43] an index is proposed that is based on a measure of similarity between two different clusterings of the same dataset and considers how each pair of data points is assigned in each clustering. If the pair of points i, j is placed together—assigned to the same cluster—in both clusterings, or if they are placed in different clusters in both clusterings, this is considered a similarity trait between the two clusterings. The dissimilarity is observed when the pair of points is placed together in one clustering and separated in the other [43]. Therefore, for any two clusterings Y, Y' of N points $X_1, X_2, \ldots, X_N$, the similarity between them is calculated by Eq. (2), where $\gamma_{ij} = 1$ if the pair of points i, j appears in both clusterings in the same relation and $\gamma_{ij} = 0$ if the pair of points does not have the same kind of relations in both clusterings.

$$c(Y, Y') = \sum_{i<j}^{N} \frac{\gamma_{ij}}{\binom{N}{2}} \tag{2}$$

Additionally, each descriptor (and its alternative variant of non-fixed aspect ratio) was evaluated according to the perceptual coherence of each group and between groups. A group is considered coherent if it presents a dominant shape (the shape that appears the highest number of times in a group) with a lower number of outlier designs. Confusion matrices are used to compare descriptor variants. These are presented in a table format where two clusterings from the same dataset can be compared by showing the number of elements that belong to the clusters of both clusterings, in each table entry. These are usually used to compare a clustering predicted by a machine learning algorithm and a clustering that is a reference clustering. The columns and rows represent each group for the two descriptors.

3.2.1. Point Distance (PD) descriptor results

For PD descriptor, Figure 6 depicts the clustering results (for fixed aspect ratio) and the group's dominant shape at left of the group letter. The group outliers were placed at the end of each group row for readability. This descriptor presents six unique dominant shapes from a total of nine possible ones, none of the groups was free from outliers, clustering accuracy of 70.83%, and Rand index of 0.861. The number of designs per group varies between 4 and 14. The group with the highest number of dominant shape designs (N_d) was group **D** with 9 and the groups with the lowest number of outliers were **D**, **G**, **H**, and **I** with one. Outliers exist in all groups.

From a perceptual analysis, when compared to the reference clustering partition, the PD descriptor is unable to have a fully coherent group. For instance, group **A** has the L-shape as the dominant shape the type and FPD 4, 8, 42, and 64 as outliers. Group **B** follows the Z-shape type and has as outliers FPD 6, 25, 43, 52, 54, and 71, which would fit better in the top-right L-shape (dominant shape absent from this partition). Group **C** only has 2 outliers (FPD 26 and 38) and has a reflected Z-shape. The top-left L-shape group **D** has only 1 outlier (FPD 20). Group **E**

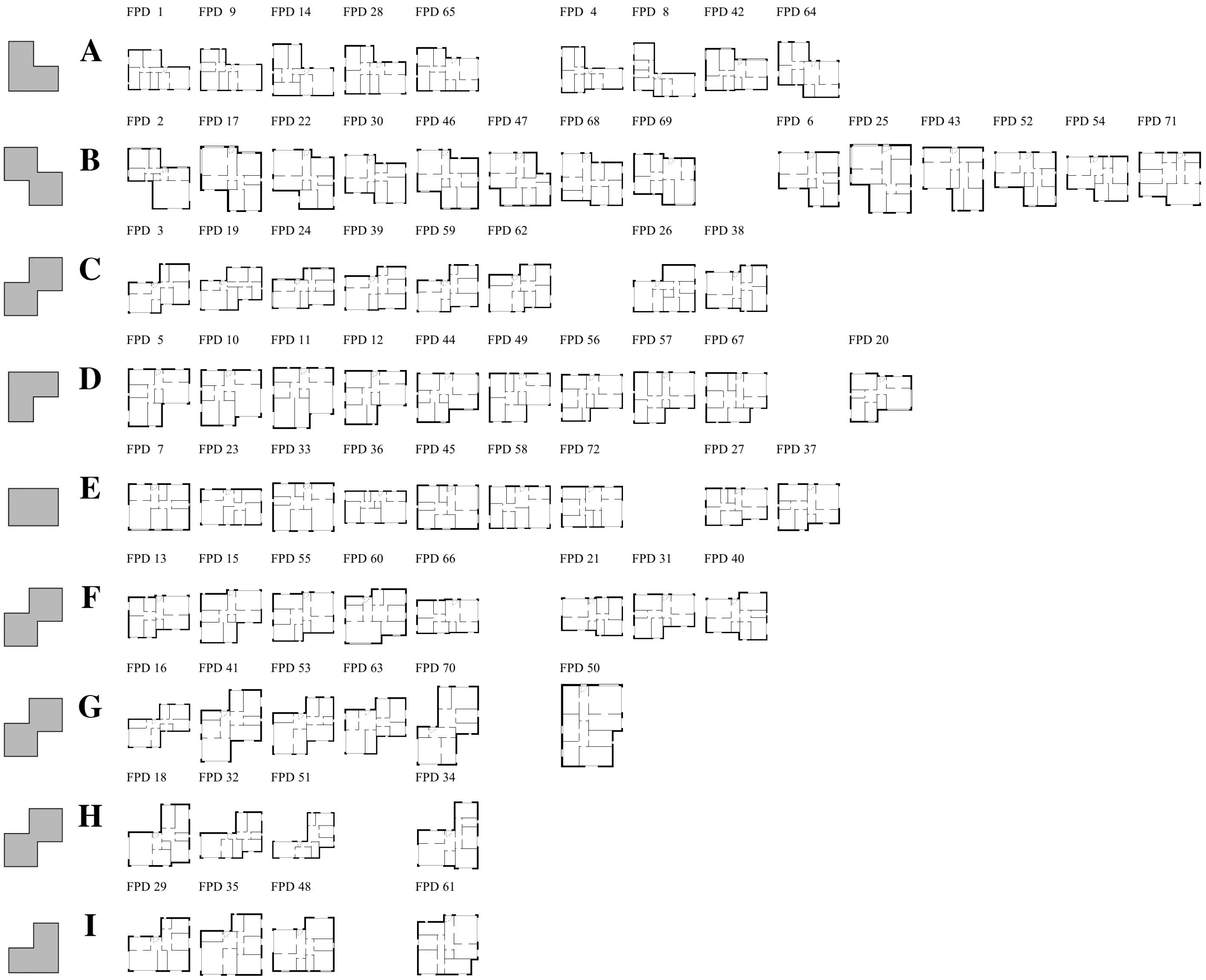


Figure 6: Clustering results using Point Distance (PD) descriptor.

aggregates the O-shape type and have 2 outliers (FPD 27 and 37) that would fit in group **D**. Group **F** and **H** have the same reflected Z-shape type as group **C** and only have one incorrectly assigned design (FPD 50 and 34, respectively). Finally, the last group **I** has a reflected L-shape with one outlier (FPD 61).

Table 2a presents the confusion matrix of this fixed aspect ratio descriptor variant against the reference clustering partition. It is noticeable that designs in partitions **B'** and **I'** are dispersed over four or more groups of the descriptor results, thus showing the difficulty of the PD descriptor in correctly determining the top-left L-shape and the reflected Z-shape types. It is also observable that the top-right L-shape (partition **C'**), rotated left T-shape (**F'**), and rotated right T-shape (**G'**) are outliers in several descriptor groups (**B**; **A** and **D**; and **C**, **F**, and **H**, respectively).

Comparing the fixed aspect ratio variant of this descriptor with the non-fixed one (see Figure A.10 in Appendix A), the performance decreases with an clustering accuracy (Ac) to 66.67%

and Rand index (R_i) to 0.852. Despite having one group with no outlier (group **C**) and finding the same number of unique shape groups (see Table 2b), the descriptor with this feature loses accuracy in groups **B**, **E**, **G**, **H**, and **I**; however, it improves in groups **C** and **D** (see Table 2c).

3.2.2. Turning Function (TF) descriptor results

Figure 7 presents the results for the TF descriptor and the dominant shape in each group. The TF descriptor has 6 unique shape groups (N_u), 2 groups without any outlier (N_o), clustering accuracy of 66.67%, and Rand index of 0.842 (R_i). The number of designs per group varies between 4 and 15. The groups with the highest number of dominant shape designs (N_d) were **C** and **D** with 8. The groups with no outliers were **D** and **H** (N_e).

The perceptual analysis of the group coherence shows that group **A** has two outliers (FPD 4 and 8) and the dominant shape type is the L-shape. Group **B** follows the Z-shape and has FPD 28, 42, and 65 incorrectly assigned. **C** has a reflected Z-shape type and the largest number of outliers (FPD 21, 26, 29, 35, 38, 40, and 48) that mix reflected L-shape and rotated right T-shape types. Group **D** has no outliers and its shape type is the top-left L-shape. Group **E** dominant shape is the top-right L-shape with 4 outliers (FPD 17, 22, 46, and 69) whose shape fits in group **B** with Z-shape type. The O-shape group is **F** and has 6 outliers (FPD 20, 27, 37, 47, 50, and 56). Groups **G**, **H**, and **I** have the same dominant shape as **C** (reflected Z-shape). **G** only has 1 outlier (FPD 31, a top-left L-shape) and **I** has 2 outliers (FPD 51 and 34).

Table 3a compares the fixed aspect ratio descriptor variant with the reference clustering partition. The designs in partitions **B'**, **F'**, **H'**, and **I'** are spread over three or more groups, thus indicating the TF descriptor's difficulty in correctly capturing the shape top-left L-shape, rotated

Table 2: Point Distance (PD) confusion matrices.

		Fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A'					7				
	B'				9	2	1	1		
	C'		6							
	D'	5								
	E'			1						3
	F'	2			1					
	G'			1			2		1	
	H'	2	8							
	I'			6			5	5	3	1

(a)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A'					7				
	B'				10	1	1			1
	C'		4			2				
	D'	5								
	E'							1	3	
	F'	2			1					
	G'						1	2	1	
	H'	2	5			3				
	I'			7			4	5	2	2

(b)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Fixed aspect ratio	A	9								
	B		9			5				
	C			2			2	4		
	D				10					
	E				1	8				
	F						4	4		
	G			3						3
	H			2					2	
	I								4	

(c)

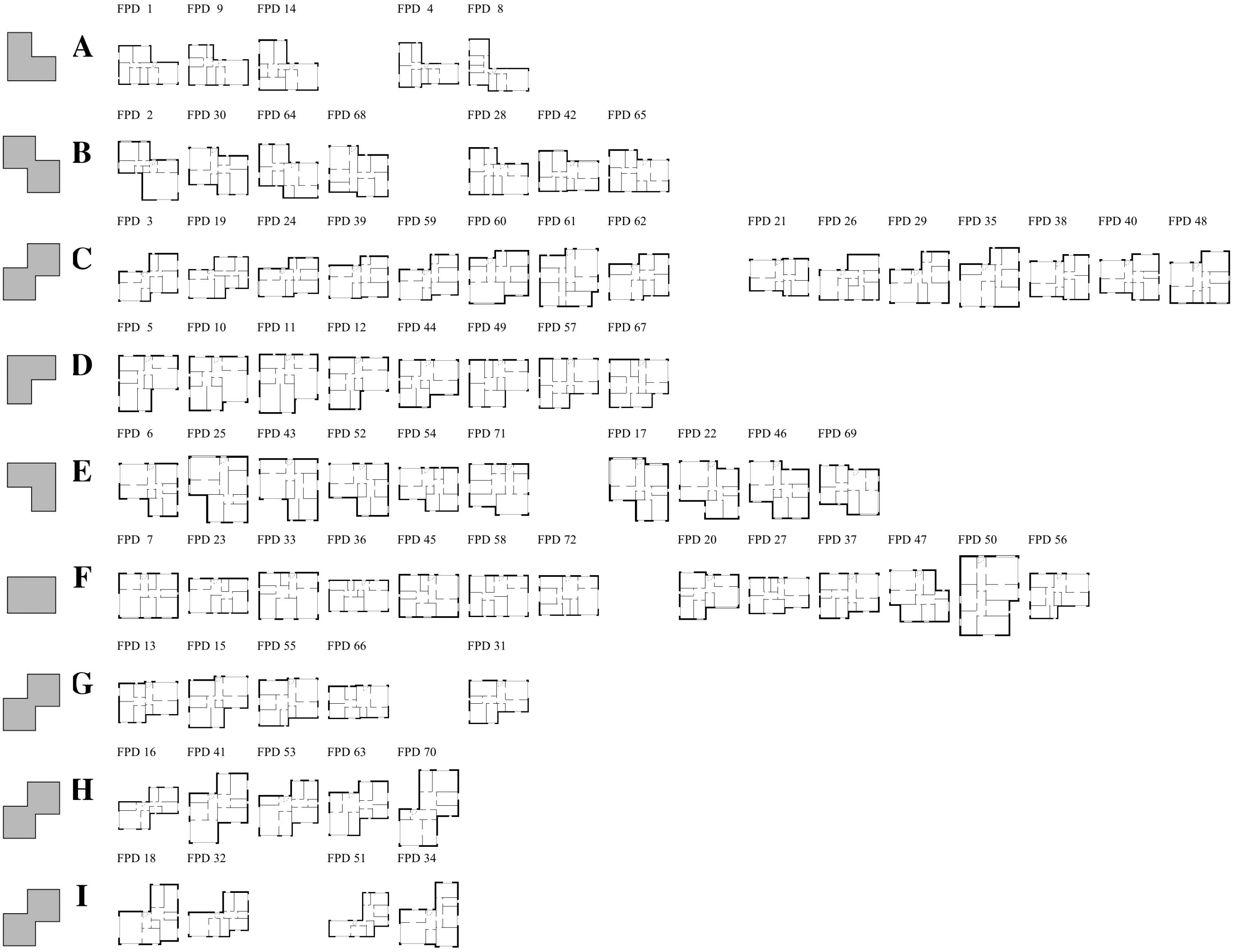


Figure 7: Clustering results using Turning Function (TF) descriptor.

left T-shape, Z-shape, and reflect Z-shape types, respectively. One may also note that shapes from partitions **E'**, **F'**, and **G'** were unable to dominate any group.

When considering the non-fixed aspect ratio descriptor variant (results are depicted in Figure A.11 in Appendix A), the performance of Ac increases to 69.44% and the R_i to 0.858. One of the two groups that had no outliers is also lost. Table 3b shows the increase of clustering accuracy for shapes in partitions **B'**, **D'**, and **F'** and decreases in **C'** and **E'**. When comparing both descriptor variants in Table 3c, group **I** has the largest shift of designs, capturing 8 that were previously in group **C**. The groups that acquire designs from other groups are **A**, **C**, **D**, **F**, and **H**.

3.2.3. *Grid-Based (GB) descriptor results*

Figure 8 illustrates the GB descriptor clustering. GB only identifies 5 unique shape groups (N_u) and one group was free from outliers (N_o). The clustering accuracy and Rand index were the lowest of all descriptors with only 55.56% (Ac) and 0.824 (R_i), respectively. The number of designs per group varies between 4 and 12. The groups with the highest number of dominant shape designs (N_d) were **C** and **G** with 8. Group **F** had no outliers (N_e). Group **I** has two dominant shapes.

GB descriptor has the lowest group coherence of all the descriptors' results. For example, group **A** and **I** have more outliers than dominant shapes—**A** (O-shape type) has FPD 1, 9, 21, 24, 27, 42, and 66 as outliers, and **B** has FPD 38, 40, and 48, and one of the two sets FPD 52, 54, and 71 (top-right L-shape) or FPD 30, 47, 69 (Z-shape). The Z-shape groups **B** and **E** have 4 (FPD 4, 14, 28, and 65) and 2 outliers (FPD 6 and 43). Groups **C**, **D**, and **H** have as dominant shape the reflected Z-shape type and has dissimilar designs FPD 26 and 29, FPD 5, 10, 11, 34, and 35, and FPD 25 and 50, respectively. Group **G**, with top-left L-shape type, has FPD 13, 15, 20, and 55 presents differing designs.

The confusion matrix, depicted in Table 4a for fixed aspect ratio, shows designs dispersed over all groups, forming heterogeneous partitions. For instance, reference clustering partitions **B'** and **I'** have designs distributed over four or more descriptor groups—**A**, **D**, **G**, and **H**, and **A**, **C**, **D**, **G**, and **H**, respectively. Therefore, the fixed aspect ratio variant of this descriptor cannot accurately capture the differences between all shapes.

However, if allowed to change the design aspect ratio, the GB descriptor significantly improves

Table 3: Turning Function (TF) confusion matrices.

		Fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A'						7			
	B'				8		4	1		
	C'					6				
	D'	3	2							
	E'			4						
	F'	1	1				1			
	G'			3						1
	H'	1	4			4	1			
	I'			8				4	5	3

(a)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A'						7			
	B'				9		2	1	1	
	C'					5	1			
	D'	5								
	E'			3						1
	F'	2			1					
	G'			1						3
	H'	1	4			4	1			
	I'			5				3	8	4

(b)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Fixed aspect ratio	A	5								
	B	3	4							
	C			5					2	8
	D				8					
	E					9	1			
	F				2		10		1	
	G							4	1	
	H								5	
	I			4						

(c)

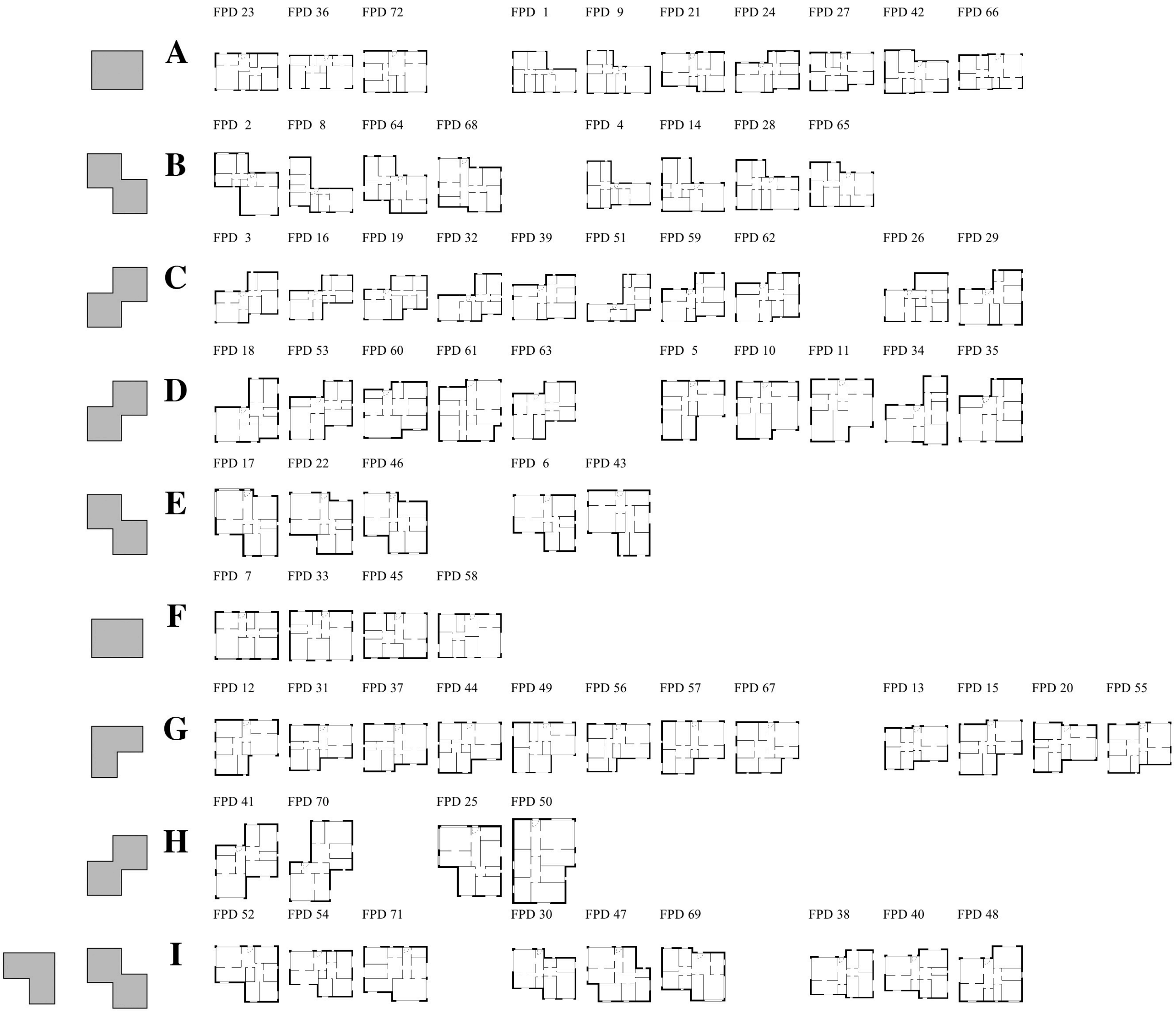


Figure 8: Clustering results using Grid-Based (GB) descriptor.

Table 4: Grid-Based (GB) confusion matrices.

		Fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A’	3					4			
	B’	1			3			8	1	
	C’					2			1	3
	D’	2	3							
	E’			2	1					1
	F’	1	1					1		
	G’	1			1					2
	H’		4			3				3
	I’	2		8	5			3	2	

(a)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A’						7			
	B’				8			5		
	C’					6				
	D’	4	1							
	E’									4
	F’	1	1		1					
	G’			1		3				
	H’	1	5			1			3	
	I’			12	6		1			1

(b)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Fixed aspect ratio	A	2	1	1	1	1	4			
	B	4	4							
	C			6	1					3
	D			4	5					1
	E					2			3	
	F						4			
	G				7			5		
	H			2	1	1				
	I		2			6				1

(c)

its accuracy, reaching 75.00% for Ac (the highest of all descriptors) and 0.874 for R_i. The group designs are depicted in Figure A.12 in Appendix A. It also achieves 7 unique shape groups (N_u) and two groups without any outlier (N_o). Table 4b shows the performance improvement in all groups as dominant shape designs increase in all partitions. The comparison of the two descriptor variants in Table 4c illustrates how designs that initially were in group **A** are now assigned to groups **A** to **F**. Other examples are the new groups **B**, **C**, **D**, and **E**, which capture designs that were assigned to several groups.

3.2.4. Tangent Distance (TD) descriptor results

The results from the TD descriptor are displayed in Figure 9. Out of all the descriptors and variants in this study, the TD descriptor presents the best results. It was able to determine 6 unique shape groups (N_u; similar to PD and TF descriptors) and only 1 group had no outliers. The clustering accuracy and Rand index were the highest of the fixed aspect ratios descriptors variant with 73.61% and 0.873 (R_i), respectively. The number of designs per cluster varies between 5 and 14. The group with the highest number of dominant shapes was **D** with 10 and the lowest number of outliers was group **C** with none.

This descriptor has the highest group coherence of all. However, there are still outliers. For instance, group **A** has the L-shape as the dominant shape type but also captures 4 outliers (FPD 4, 8, 42, and 64), three of those due to small recesses in the bottom wall. It is observable that FPD 64 clearly belongs to the Z-shape type group. Group **B** has 6 outliers (FPD 6, 25, 43, 52, 54, and 71)—all fitting the top-right L-shape instead of the dominant Z-shape type. Top-left L-shape in group **D** has a single outlier (FPD 20), which fits the rotated left T-shape due to a small recess in the top wall. For similar reasons, group **E** with O-shape type has FPD 27 (top-left L-shape) as an outlier. Groups **F** and **G** have the same reflected Z-shape type. The outliers of these groups are FPD 21 and 31 and outlier FPD 50, respectively. Despite having the same shape type, TD descriptor partitioned designs into two groups because the concave turns in the walls have different size segments. Group **H** has 2 outliers (FPD 18 and 34) in the dominant shape type reflected L-shape. Once again, the descriptor did not consider these designs with a different shape despite the small recess in the bottom wall. Finally, the last group **I**, with reflected Z-shape, has 2 outliers (FPD 38 and 40 with rotated right T-shape type).

Table 5a presents the confusion matrix for this descriptor against the reference clustering. Partition **A'** designs are fully included in group **E**. However, partition **B'** has three of its designs spread over three groups **E** to **G**, but the remaining 10 designs are assigned to group **D**. Partitions

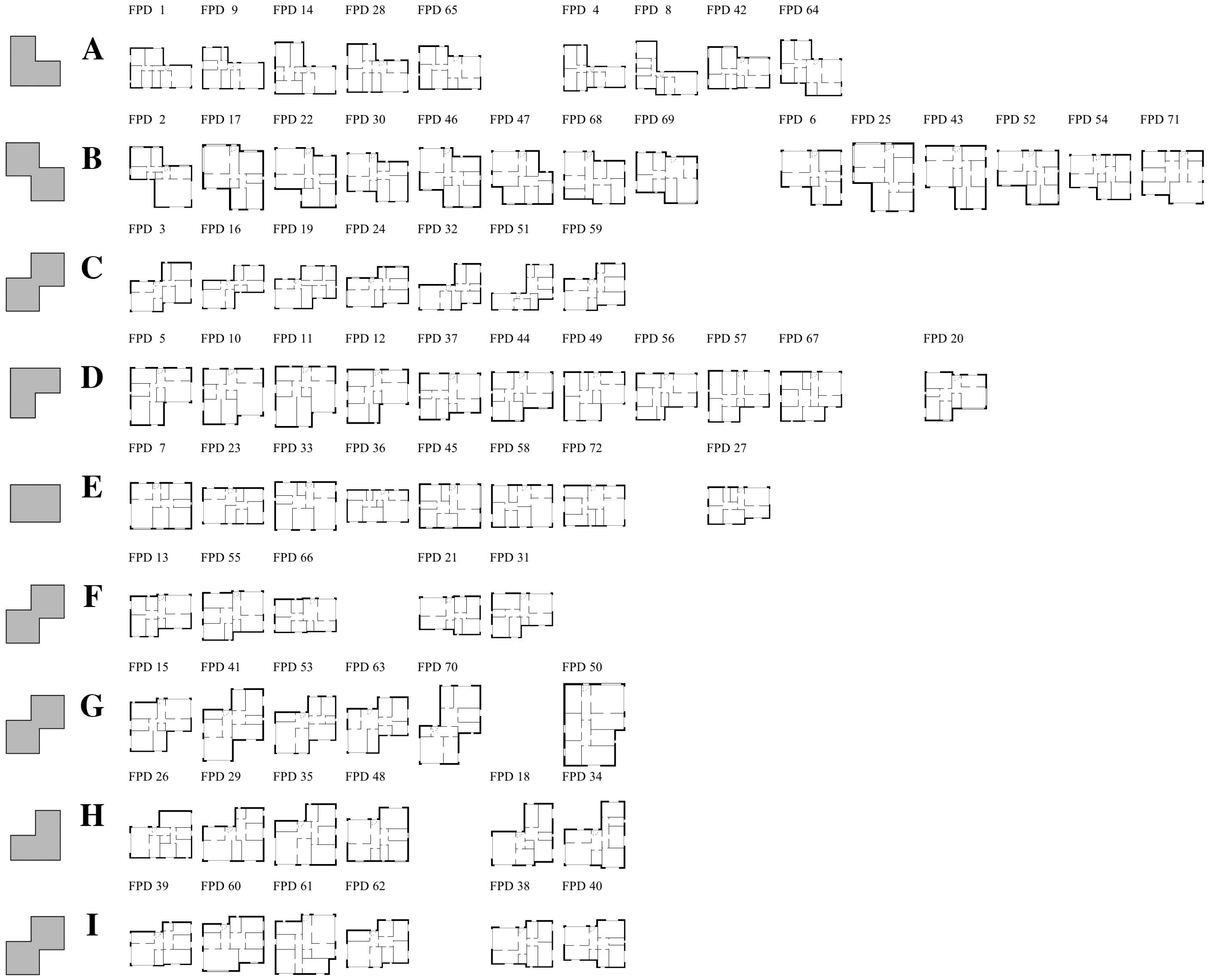


Figure 9: Clustering results using Tangent Distance (TD) descriptor.

C', **D'**, and **E'** are also assigned to a corresponding group—**B**, **A**, and **H**, respectively. Designs in partitions **G'** and **H'** are distributed over three (**F**, **H**, and **I**) and two groups (**A** and **B**). Finally, the largest reference clustering partition **I'** had its designs assigned to five groups (**C**, and **F** to **I**).

When considering the non-fixed aspect ratio descriptor variant, the descriptor underperforms slightly in the clustering accuracy, which decreases to 72.22%, but improves in the Rand index to 0.876. Reference clustering partitions **B'** and **G'** are better partitioned in this descriptor variant, but accuracy is lost for partitions **C'**, **E'**, **G'**, **H'**, and **I'** (Table 5b). Comparing both descriptor variants (Table 5c) groups **B**, **C**, and **E** to **I** have a few designs that have been shifted to other groups.

Table 5: Tangent Distance (TD) confusion matrices.

		Fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A’					7				
	B’				10	1	1	1		
	C’		6							
	D’	5								
	E’								4	
	F’	2			1					
	G’						1		1	2
	H’	2	8							
	I’			7			3	5	1	4

(a)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Reference clustering	A’					7				
	B’				12		1			
	C’		5						1	
	D’	5								
	E’							2		2
	F’	2			1					
	G’							1	3	
	H’	3	7							
	I’			7		1	4	3	1	4

(b)

		Non-fixed aspect ratio								
		A	B	C	D	E	F	G	H	I
Fixed aspect ratio	A	9								
	B	1	12						1	
	C			3				2		2
	D				11					
	E				1	7				
	F					1	3		1	
	G			4	1		1			
	H							4		2
	I						1		3	2

(c)

4. Discussion

Table 6 summarises per descriptor the number of unique shapes (N_u; number of groups with unique shape type), number of groups without outliers (N_o), the percentage of clustering accuracy (Ac; number of dominant shape designs per total of floor plan designs), and Rand index (R_i). It also lists the number of dominant shapes (N_d) and the number of outliers (N_e) per group. The descriptor with better R_i is Tangent Distance (TD) with 0.873 and 0.876 for fixed and non-fixed aspect ratio variants, respectively. However, Grid-Based (GB) presents the highest number of unique shape groups (N_u) and the highest Ac of 75% for the non-fixed aspect ratio descriptor variant.

The presence of outliers (N_e) in the Point Distance (PD) descriptor may indicate why some groups have designs dispersed by other clusters. This can result from the fact that, when there is a slight discontinuity of the exterior wall, the measured distance from the points in the perimeter dilutes such difference. This is a benefit in shapes requiring denoising; however, in datasets with no noise the results are not so good.

In the case of the Turning Function (TF) descriptor, other problem occurs. Namely, due to the absence of information in the descriptor vector resulting from wall recesses smaller than the distance between feature points—when the wall turns a small distance and turns back to the same direction. In this situation, and because this descriptor only captures the angle of the wall, the information before and after the wall change is the same. The only way to include that information is to have a feature point of the shape silhouette in it. Additionally, even if the wall recess is somehow captured, it only represents a few values in the vector as the main parts of the

Table 6: Descriptors performance.

			Group																			
			A		B		C		D		E		F		G		H		I			
Descriptor	N_u	N_o	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	*Ac*	R_i
Point Distance (PD)	**6**	0	5	4	8	6	6	2	9	1	7	2	5	3	5	1	3	1	3	1	70.83%	0.861
Turning Function (TF)	**6**	**2**	3	2	4	3	8	7	8	0	6	4	7	6	4	1	5	0	3	1	66.67%	0.842
Grid-Based (GB)	5	1	3	7	4	4	8	2	5	5	3	2	4	0	8	4	2	2	3	6	55.56%	0.824
Tangent Distance (TD)	**6**	1	5	4	8	6	7	0	10	1	7	1	3	2	5	1	4	2	4	2	**73.61%**	**0.873**

(a) Fixed aspect ratio

			Group																			
			A		B		C		D		E		F		G		H		I			
Descriptor	N_u	N_o	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	N_d	N_e	*Ac*	R_i
Point Distance (PD)	6	1	5	4	5	4	7	0	10	1	7	6	4	2	5	3	3	3	2	1	66.67%	0.852
Turning Function (TF)	6	1	5	3	4	0	5	4	9	1	5	4	7	4	3	1	8	1	4	4	69.44%	0.858
Grid-Based (GB)	**7**	**2**	4	2	5	2	12	1	8	7	6	4	7	1	5	0	3	0	4	1	**75.00%**	0.874
Tangent Distance (TD)	6	1	5	5	7	5	7	0	12	1	7	1	4	1	3	3	3	2	4	2	72.22%	**0.876**

N_u - Number of groups with unique shape; N_o - Number of groups without outliers;

N_d - Number of dominant shape designs; N_e - Number of outliers; *Ac* - Accuracy; R_i - Rand index

(b) Non-fixed aspect ratio

wall continue to have the same angle. This would be avoided only if the descriptor also measured the wall distance to a reference point.

In the results for the GB descriptor the problem is different. In this case, the descriptor vector is very sensitive to the measuring points in the grid. Therefore, if there are small variants in the shape proportions then a row of points can turn from 0 to 1 and vice-versa. For instance, a wider rectangle, when scaled to fit the measuring grid, will result in a smaller height thus having less area filled in the grid. Despite the shape being basically the same, this will result in different vectors (compare the FPD 23 in group **A** and group **F** in Figure 8 as an example of this issue). However, when dealing with adjusted aspect ratio, the performance improves for the GB descriptor.

The TD descriptor presents the best results for both variants of the aspect ratio. This is due to the fact that it incorporates the advantages of the PD and the TF descriptors, namely the ability to capture the distance of the segment and the angle change of the walls, respectively. However, when extending the use to shapes such as the equilateral triangle, square, pentagon, or other regular polygons (even a circumference), the TD descriptor will classify all of them in the same group, as the polygon tangents all have the same distance to the centre. Another issue was found with this descriptor. In some cases, when designs have the same shape type, it may consider distinct due to the sensitivity over the size of the segments in every turn of the exterior wall (see groups **F** and **G** in Figure 9 as an example).

In the case of the distance-based descriptors (PD and TD), it is possible to control their sensitivity to wall recesses in the shape perimeter by exponentiating the normalised distances. If the exponent is lower than 1, the representation reduces the sensitivity to small variations; otherwise, when greater than 1, this is increased.

It is interesting to observe that the descriptors that have the best results are all perimeter-based representations. Area-based representations, such as the GB descriptor, are too sensitive to small changes in the proportions of the shape. This approach may have better results in shapes that require denoising. However, in synthetic datasets such as the one illustrated in the showcase, area-based representation is a less reliable approach. Limitations of these descriptors may be summarised as follows:

- PD, TF, and GB descriptors are insensitive to small recesses in the perimeter;
- TF descriptor may not capture perimeter turns if the shape's silhouette step is bigger that the turn segment dimension;
- GB descriptor greatly depends on the grid resolution thus making it very sensitive to small variations in the shape proportions;
- TD descriptor may suffer from excessive sensitivity to the segments size in wall turns, thus leading to cluster designs in different groups despite having the same shape type;
- TD descriptor clusters regular polygons (triangle, square, circle, etc.) as the same shape; and,
- TD descriptor is very sensitive to shapes with noise in the perimeter.

The matching and clustering of floor plan designs has some possible applications. One of those is to use it as a clustering mechanism for results obtained from generative design methods—for example, the EPSAP algorithm already includes these mechanisms to organise data to be presented to the decision-maker. Another example is to use it within the evolving process of population-based methods. This may have two purposes. First, to select the best individuals of each group to be kept in the next generation, thus preserving the population diversity and avoiding the dominance of one shape type. Secondly, to conduct the generative process on solutions that are of interest to the user according to their defined shape type criterion. Nowadays floor plan generative methods deal with building boundaries as defined polygons. However, if the user is able to choose the aimed shape or shapes, the method may focus only on that range of candidate designs, thus reducing the computation burden by avoiding the production and evaluation of irrelevant solutions. Finally, a possible application is to use it as a retrieval process of designs in architectural design databases.

5. Conclusion

Four shape descriptors were used to capture the form of a synthetic dataset of floor plan designs and a comparison of their performance was carried out. Every descriptor had the same vector length and the same clustering algorithm was used to aggregate the floor plans.

The perceptual analysis carried out on the four descriptors shows that Tangent Distance (TD) captures better floor plan shapes and presents fewer outliers. This was due to the fact that this descriptor not only measures the distance to the geometric centre but also captures the discontinuities in the walls. The outliers resulted from excessive sensitivity to small wall recesses in the perimeter thus shifting the design to other group with a similar overall configuration.

In the case of the other descriptors, the opposite happens. The Grid-Based (GB) descriptor presents the least reliable approach and is very sensitive to different proportions in the same shape thus designs are distributed over several groups with different dominant shapes.

For the fixed aspect ratio variant, the performance of the two best descriptors was a Rand index of 0.861 and 0.873 for the Point Distance (PD) and TD, respectively. In the non-fixed aspect ratio descriptor variant, the descriptors with the best performance were the GB and TD, with a Rand index of 0.874 and 0.876, respectively.

Despite these good results, some issues still need to be tackled. Future work includes extending these approaches to non-orthogonal and multi-storey designs, to study other descriptors that capture the inner space relations in the floor plan, and to test the performance of descriptors in other types of clustering algorithms.

Acknowledgements

This work has been developed under the *Energy for Sustainability Initiative* of the University of Coimbra (UC). It has been partially supported by the Portuguese Foundation for Science and Technology (FCT), under the projects PEst INESCC UID/MULTI/00308/2013, Suscity MITP-TB/CS/0026/2013, and by FCT and European Regional Development Fund (FEDER) through COMPETE – Programa Operacional Competitividade e Internacionalização (POCI), under the project Ren4EEnIEQ (PTDC/SEM-ENE/3238/2014 and POCI-01-0145-FEDER-016760 respectively). Eugénio Rodrigues acknowledges the support of the FCT under PostDoc grant SFRH/BPD/99668/2014.

References

[1] J. Han, M. Kamber, Data mining: concepts and techniques, Morgan Kaufmann Publishers Inc., San Francisco, CA, 2nd edn., ISBN 978-1-55860-901-3, 2001.

[2] M. Al Qady, A. Kandil, Automatic clustering of construction project documents based on textual similarity, Automation in Construction 42 (2014) 36–49, ISSN 09265805, doi:10.1016/j.autcon.2014.02.006.

[3] S. Dumais, H. Chen, Hierarchical classification of Web content, in: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '00, ACM, New York, NY, USA, ISBN 1-58113-226-3, 256–263, doi:10.1145/345508.345593, 2000.

[4] F. Beil, M. Ester, X. Xu, Frequent term-based text clustering, in: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '02, ACM, New York, NY, USA, ISBN 1-58113-567-X, 436–442, doi:10.1145/775047.775110, 2002.

[5] D. Sousa-Rodrigues, Q-analysis Based Clustering of Online News, Discontinuity, Nonlinearity, and Complexity 3 (3) (2014) 227–236, ISSN 21646414, doi:10.5890/DNC.2014.09.002.

[6] M.-Y. Cheng, D.-H. Tran, Y.-W. Wu, Using a fuzzy clustering chaotic-based differential evolution with serial method to solve resource-constrained project scheduling problems, Automation in Construction 37 (2014) 88–97, ISSN 09265805, doi:10.1016/j.autcon.2013.10.002.

[7] H. Song, H.-Y. Feng, A global clustering approach to point cloud simplification with a specified data reduction ratio, Computer-Aided Design 40 (3) (2008) 281–292, ISSN 00104485, doi:10.1016/j.cad.2007.10.013.

[8] B.-Q. Shi, J. Liang, Q. Liu, Adaptive simplification of point cloud using k-means clustering, Computer-Aided Design 43 (8) (2011) 910–922, ISSN 00104485, doi:10.1016/j.cad.2011.04.001.

[9] T. W. Liao, Clustering of time series data—a survey, Pattern Recognition 38 (11) (2005) 1857 – 1874, ISSN 0031-3203, doi:10.1016/j.patcog.2005.01.025.

[10] V. V. Vikjord, R. Jenssen, Information theoretic clustering using a k-nearest neighbors approach, Pattern Recognition 47 (9) (2014) 3070–3081, ISSN 00313203, doi:10.1016/j.patcog.2014.03.018.

[11] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326, ISSN 00368075, doi:10.1126/science.290.5500.2323.

[12] J. Pu, K. Ramani, On visual similarity based 2D drawing retrieval, Computer-Aided Design 38 (3) (2006) 249–259, ISSN 00104485, doi:10.1016/j.cad.2005.10.009.

[13] S. Jayanti, Y. Kalyanaraman, K. Ramani, Shape-based clustering for 3D CAD objects: A comparative study of effectiveness, Computer-Aided Design 41 (12) (2009) 999–1007, ISSN 00104485, doi:10.1016/j.cad.2009.07.003.

[14] V. Deufemia, M. Risi, G. Tortora, Sketched symbol recognition using Latent-Dynamic Conditional Random Fields and distance-based clustering, Pattern Recognition 47 (3) (2014) 1159–1171, ISSN 00313203, doi: 10.1016/j.patcog.2013.09.016.

[15] D. Zhang, G. Lu, Review of shape representation and description techniques, Pattern Recognition 37 (1) (2004) 1–19, ISSN 00313203, doi:10.1016/j.patcog.2003.07.008.

[16] C. Chang, S. Hwang, D. Buehrer, A shape recognition scheme based on relative distances of feature points from the centroid, Pattern Recognition 24 (11) (1991) 1053–1063, ISSN 00313203, doi:10.1016/0031-3203(91) 90121-K.

[17] D. Yankov, E. Keogh, Manifold clustering of shapes, Proceedings - IEEE International Conference on Data Mining, ICDM (2006) 1167–1171ISSN 15504786, doi:10.1109/ICDM.2006.101.

[18] E. Arkin, L. Chew, D. Huttenlocher, K. Kedem, J. Mitchell, An efficiently computable metric for comparing polygonal shapes, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (3) (1991) 209–216, ISSN 01628828, doi:10.1109/34.75509.

[19] A. Sajjanhar, G. Lu, A grid based shape indexing and retrieval method, Computer Journal on Multimedia Storage and Archiving Systems 29 (1997) 131–140.

[20] K. Siddiqi, A. Shokoufandeh, S. J. Dickinson, S. W. Zucker, Shock graphs and shape matching, International Journal of Computer Vision 35 (1) (1999) 13–32, ISSN 09205691, doi:10.1023/A:1008102926703.

[21] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (24) (2002) 509–522, ISSN 01628828, doi: 10.1109/34.993558.

[22] K. L. Tan, B. C. Ooi, L. F. Thiang, Retrieving similar shapes effectively and efficiently, Multimedia Tools and Applications 19 (2003) 111–134, ISSN 13807501, doi:10.1023/A:1022142527536.

[23] E. Klassen, A. Srivastava, W. Mio, S. H. Joshi, Analysis of planar shapes using geodesic paths on shape spaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (3) (2004) 372–383, ISSN 01628828, doi: 10.1109/TPAMI.2004.1262333.

[24] A. Srivastava, S. H. Joshi, W. Mio, X. Liu, Statistical shape analysis: clustering, learning, and testing, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (4) (2005) 590–602, ISSN 01628828, doi:10.1109/TPAMI.2005.86.

[25] W. Mio, A. Srivastava, S. Joshi, On shape of plane elastic curves, International Journal of Computer Vision 73 (3) (2007) 307–324, ISSN 0920-5691, doi:10.1007/s11263-006-9968-0.

[26] H. L. H. Ling, D. Jacobs, Shape classification using the inner-distance, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (2) (2007) 1–35, ISSN 0162-8828, doi:10.1109/TPAMI.2007.41.

[27] W. Shen, Y. Wang, X. Bai, H. Wang, L. Jan Latecki, Shape clustering: Common structure discovery, Pattern Recognition 46 (2) (2013) 539–550, ISSN 00313203, doi:10.1016/j.patcog.2012.07.023.

[28] M. Y. Cha, J. S. Gero, Shape Pattern Recognition Using a Computable Pattern Representation, in: Artificial Intelligence in Design '98, Springer Netherlands, Dordrecht, ISBN 978-94-011-5121-4, 169–187, doi:10.1007/978-94-011-5121-4_9, 1998.

[29] L.-P. de las Heras, D. Fernández, A. Fornés, E. Valveny, G. Sánchez, J. Lladós, Runlength Histogram Image Signature for Perceptual Retrieval of Architectural Floor Plans, in: B. Lamiroy, J.-M. Ogier (Eds.), Graphics Recognition. Current Trends and Challenges: 10th International Workshop, GREC 2013, Bethlehem, PA, USA, August 20-21, 2013, Revised Selected Papers, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-662-44854-0, 135–146, doi:10.1007/978-3-662-44854-0_11, 2014.

[30] A. Dutta, J. Lladós, H. Bunke, U. Pal, A Product Graph Based Method for Dual Subgraph Matching Applied to Symbol Spotting, in: B. Lamiroy, J.-M. Ogier (Eds.), Graphics Recognition: Current Trends and Challenges, vol. 8746 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-662-44853-3, 11–24, doi:10.1007/978-3-662-44854-0, 2014.

[31] D. Sousa-Rodrigues, M. T. de Sampayo, E. Rodrigues, A. R. Gaspar, Á. Gomes, C. H. Antunes, Online survey for collective clustering of computer generated architectural floor plans, in: 15th International Conference on Technology Policy and Innovation, 17-19 June, Milton Keynes, UK, 2015.

[32] D. Sousa-Rodrigues, M. Teixeira de Sampayo, E. Rodrigues, A. R. Gaspar, Á. Gomes, Crowdsourced Clustering of Computer Generated Floor Plans, in: Yuhua Luo (Ed.), The 12th International Conference on Cooperative Design, Visualization & Engineering, Sept 20-23, Springer, Mallorca, Spain, ISBN 978-3-319-24132-6, 142–151, doi:10.1007/978-3-319-24132-6_17, 2015.

[33] J. H. Ward Jr, Hierarchical grouping to optimize an objective function, Journal of the American Statistical Association 58 (301) (1963) 236–244, doi:10.1080/01621459.1963.10500845.

[34] E. Rodrigues, A. Gaspar, Á. Gomes, An evolutionary strategy enhanced with a local search technique for the space allocation problem in architecture, Part 1: Methodology, Computer Aided-Design 45 (5) (2013) 887–897, ISSN 00104485, doi:10.1016/j.cad.2013.01.001.

[35] E. Rodrigues, A. Gaspar, Á. Gomes, An evolutionary strategy enhanced with a local search technique for the space allocation problem in architecture, Part 2: Validation and Performance Tests, Computer Aided-Design 45 (5) (2013) 898–910, ISSN 00104485, doi:10.1016/j.cad.2013.01.003.

[36] E. Rodrigues, A. Gaspar, Á. Gomes, An approach to the multi-level space allocation problem in architecture using a hybrid evolutionary technique, Automation in Construction 35 (2013) 482–498, ISSN 09265805, doi:10.1016/j.autcon.2013.06.005.

[37] J. Dougherty, R. Kohavi, M. Sahami, et al., Supervised and unsupervised discretization of continuous features, in: Proceedings of the 12th International Conference on Machine Learning, July 9-12, Tahoe City, California, USA, ISBN 978-1-55860-377-6, 194–202, 1995.

[38] S. Kotsiantis, D. Kanellopoulos, Discretization techniques: A recent survey, GESTS International Transactions on Computer Science and Engineering 32 (1) (2006) 47–58.

[39] J. L. Lustgarten, V. Gopalakrishnan, H. Grover, S. Visweswaran, Improving classification performance with discretization on biomedical datasets, in: AMIA Annual Symposium Proceedings, vol. 2008, American Medical Informatics Association, ISSN 1942597X, 445—449, 2008.

[40] M. Rucco, D. Sousa-Rodrigues, E. Merelli, J. Johnson, L. Falsetti, C. Nitti, A. Salvi, Neural hypernetwork approach for pulmonary embolism diagnosis, BMC Research Notes 8 (1) (2015) 617, ISSN 1756-0500, doi:10.1186/s13104-015-1554-5.

[41] D. Defays, An efficient algorithm for a complete link method, The Computer Journal 20 (4) (1977) 364–366, ISSN 14602067, doi:10.1093/comjnl/20.4.364.

[42] E. Deza, M. M. Deza, Encyclopedia of Distances, Springer Berlin Heidelberg, Berlin, Heidelberg, ISBN 978-3-642-00233-5, doi:10.1007/978-3-642-00234-2, 2009.

[43] W. M. Rand, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical association 66 (336) (1971) 846–850, doi:10.1080/01621459.1971.10482356.

Appendix A. Descriptors' results for non-fixed aspect ratio

Figures A.10, A.11, A.12, and A.13 display the resulting clustering of each of the four shape representations with non-fixed aspect ratio.

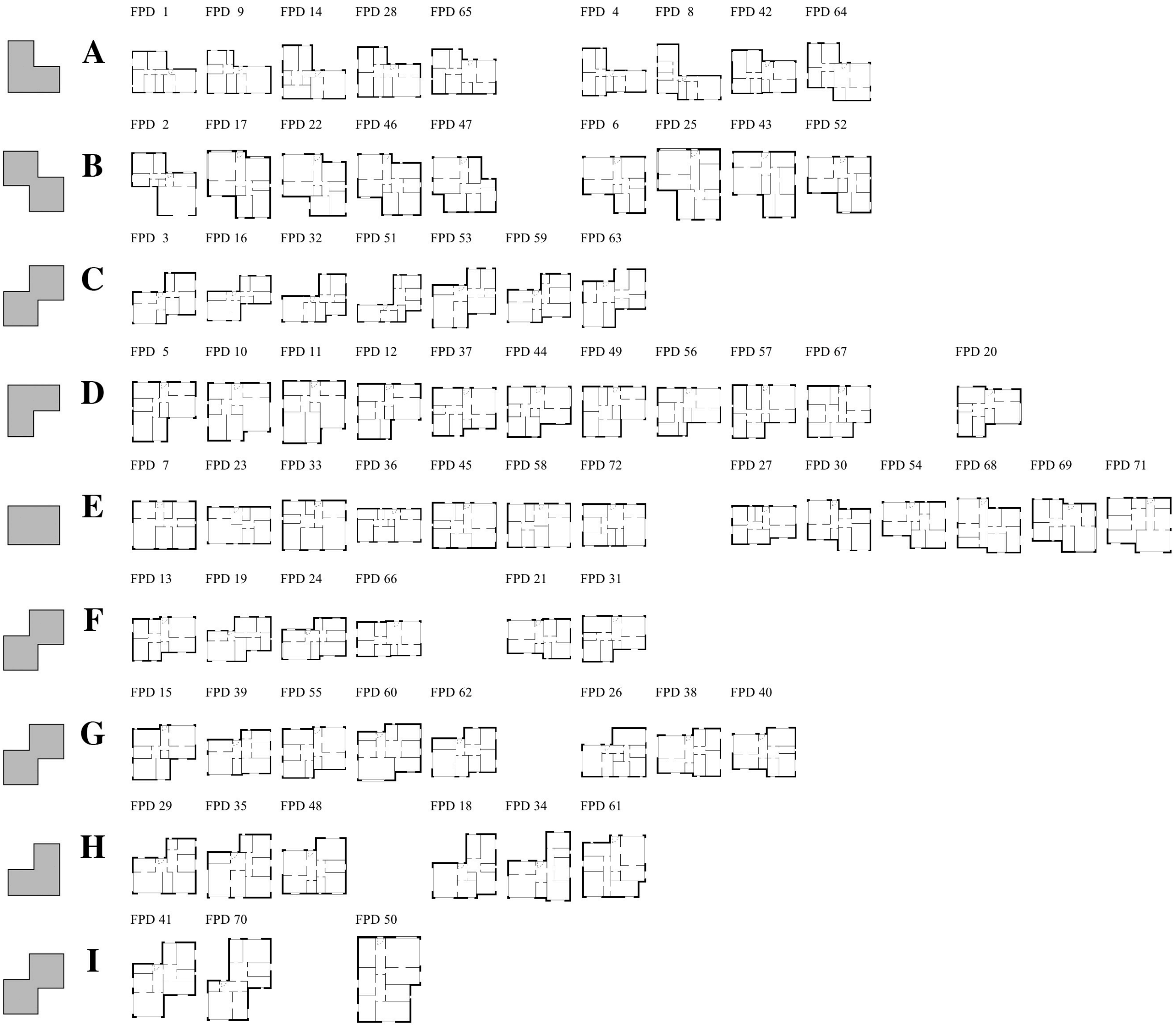


Figure A.10: Clustering results using Point Distance (PD) descriptor with non-fixed aspect ratio.

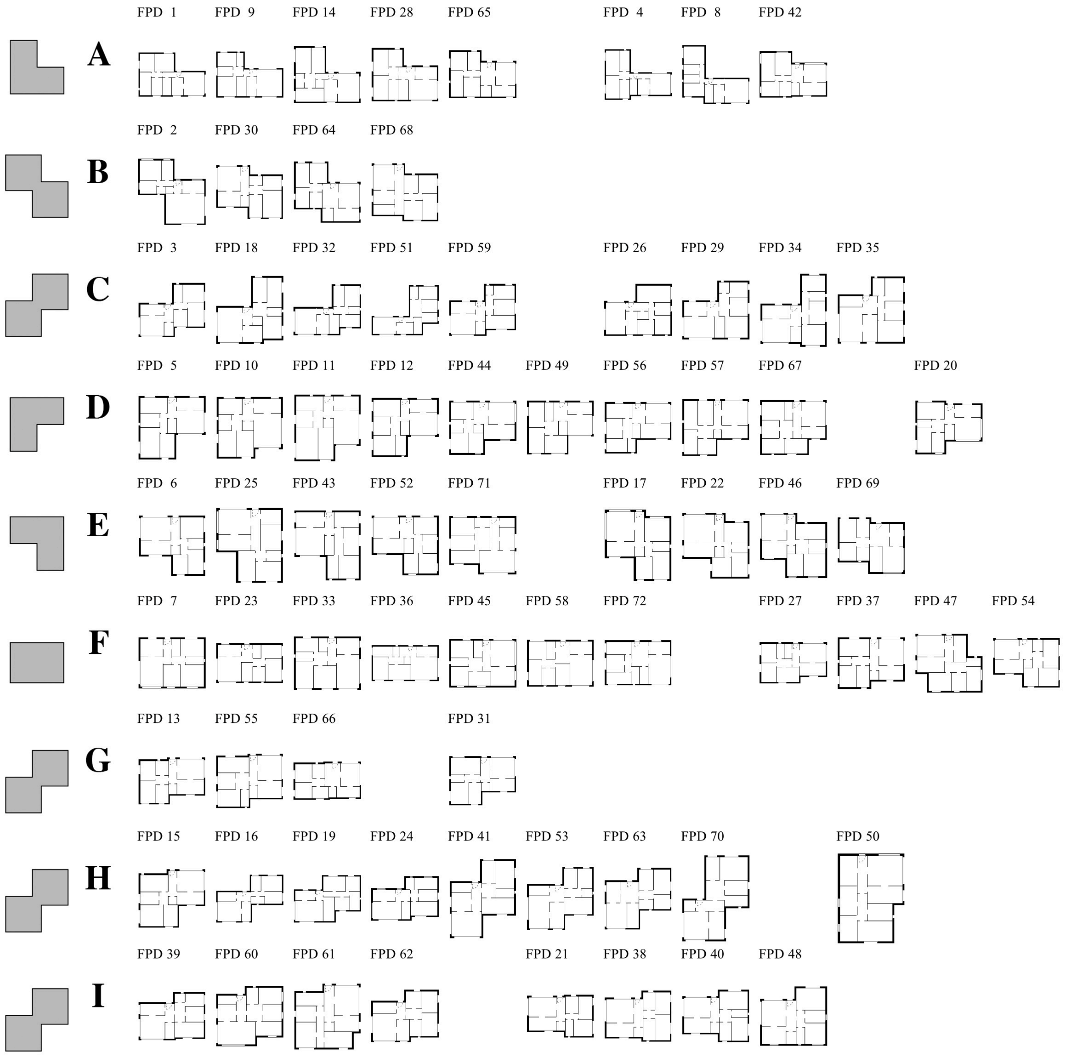


Figure A.11: Clustering results using Turning Function (TF) descriptor with non-fixed aspect ratio.

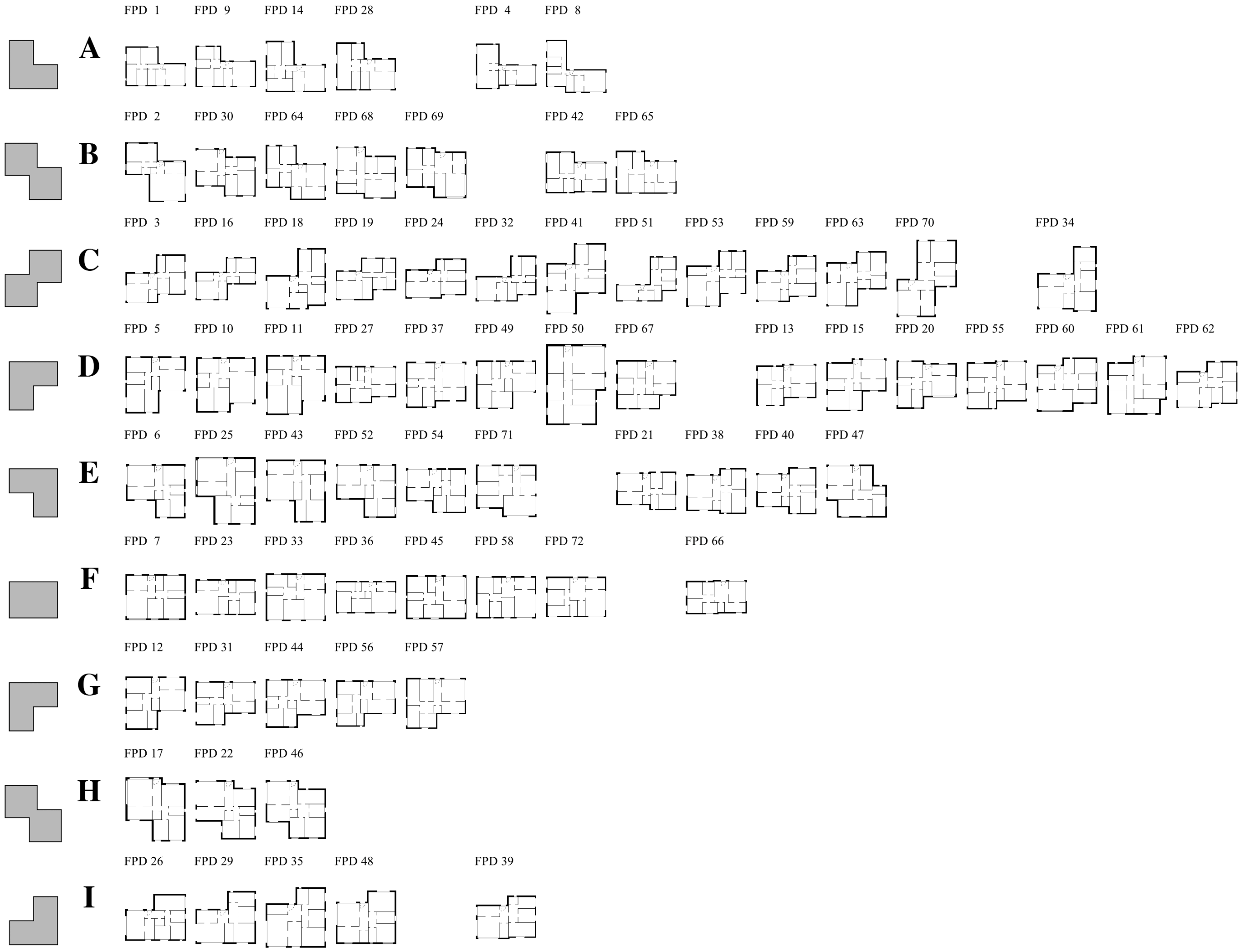

A
FPD 1 FPD 9 FPD 14 FPD 28 FPD 4 FPD 8
B
FPD 2 FPD 30 FPD 64 FPD 68 FPD 69 FPD 42 FPD 65
C
FPD 3 FPD 16 FPD 18 FPD 19 FPD 24 FPD 32 FPD 41 FPD 51 FPD 53 FPD 59 FPD 63 FPD 70 FPD 34
D
FPD 5 FPD 10 FPD 11 FPD 27 FPD 37 FPD 49 FPD 50 FPD 67 FPD 13 FPD 15 FPD 20 FPD 55 FPD 60 FPD 61 FPD 62
E
FPD 6 FPD 25 FPD 43 FPD 52 FPD 54 FPD 71 FPD 21 FPD 38 FPD 40 FPD 47
F
FPD 7 FPD 23 FPD 33 FPD 36 FPD 45 FPD 58 FPD 72 FPD 66
G
FPD 12 FPD 31 FPD 44 FPD 56 FPD 57
H
FPD 17 FPD 22 FPD 46
I
FPD 26 FPD 29 FPD 35 FPD 48 FPD 39

Figure A.12: Clustering results using Grid-Based (GB) descriptor with non-fixed aspect ratio.

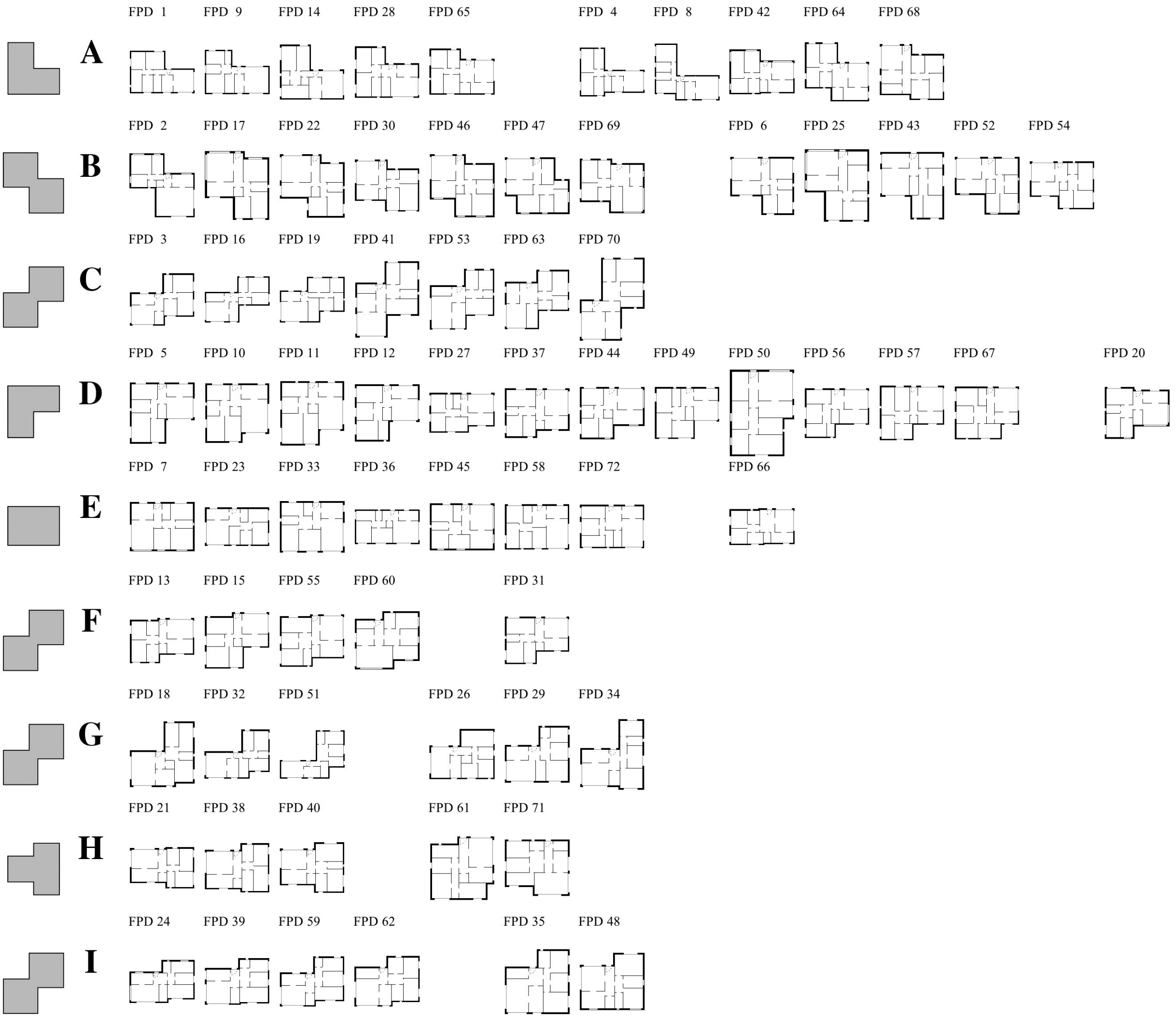


Figure A.13: Clustering results using Tangent Distance (TD) descriptor with non-fixed aspect ratio.