Proximity-based grouping of buildings in urban blocks: A comparison of four algorithms

Article *in* Geocarto International · July 2015

DOI: 10.1080/10106049.2014.925002

Melih Basaraner
Yildiz Technical University

Dirk Burghardt
Technische Universität Dresden

Geocarto International, 2015
Vol. 30, No. 6, 618–632, http://dx.doi.org/10.1080/10106049.2014.925002

Proximity-based grouping of buildings in urban blocks: a comparison of four algorithms

Sinan Cetinkaya[a]*, Melih Basaraner[a] and Dirk Burghardt[b]

[a]*Division of Cartography, Faculty of Civil Engineering, Department of Geomatic Engineering, Yildiz Technical University (YTU), Istanbul, Turkey;* [b]*Institute for Cartography, Dresden University of Technology, Dresden, Germany*

(*Received 15 January 2014; final version received 13 May 2014*)

Grouping of buildings based on proximity is a pre-processing step of urban pattern (structure) recognition for contextual cartographic generalization. This paper presents a comparison of grouping algorithms for polygonal buildings in urban blocks. Four clustering algorithms, Minimum Spanning Tree (MST), Density-Based Spatial Clustering Application with Noise (DBSCAN), CHAMELEON and Adaptive Spatial Clustering based on Delaunay Triangulation (ASCDT) are reviewed and analysed to detect building groups. The success of the algorithms is evaluated based on group distribution characteristics (i.e. distribution of the buildings in groups) with two methods: S_Dbw and newly proposed Cluster Assessment Circles. A proximity matrix of the nearest distances between the building polygons, and Delaunay triangulation of building vertices are created as an input for the algorithms. A topographic data-set at 1:25,000 scale is used for the experiments. Urban block polygons are created to constrain the clustering processes from topological aspect. Findings of the experiment demonstrate that DBSCAN and ASCDT are superior to CHAMELEON and MST. Among them, MST has exhibited the worst performance for finding meaningful building groups in urban blocks.

Keywords: grouping of buildings; cluster assessment; spatial pattern; cartographic generalization

Introduction

Building groups can comprise different urban patterns based on similarity and regularity of geometric, semantic and structural characteristics of buildings and relationships between them. Maps portray these patterns with varying degrees of abstraction depending on scale and spatial context by means of generalization techniques. Hence, it is critical to detect these patterns at source scale and preserve them at target scales as much as possible so that maps can communicate geographic information without losing main messages throughout multiple scales. Moreover, generalization operators can produce better results in proximity-based smaller groups. Many measures (orientation, size, semantics, shape, etc.) should be used to detect spatial patterns (alignments, regularities, etc.) in such groups (AGENT Consortium 1999; Bobzien et al. 2008; Burghardt & Schmid 2010).

Grouping process is gradually performed in general because simultaneous use of all measures can produce too small and meaningless groups. Therefore, the first step of

*Corresponding author. Email: sicetin@yildiz.edu.tr grouping is to analyse proximity relationships because close objects can be more spatially dependent or associated to each other. This idea can be supported by both geographic and cartographic foundations. From a geographic aspect, Tobler's first law of geography denotes: 'everything is related to everything else, but near things are more related than distant things'. From a cartographic aspect, individuals tend to visually perceive close objects in graphic representations as groups according to Gestalt principles. In addition, closely neighbouring objects have greater potential of graphic conflict (i.e. the violation of minimum distance constraint) at target scales in contextual cartographic generalization owing to enlargement and they should be treated as groups to solve this conflict (Basaraner & Selcuk 2008; Yan et al. 2008).

Graph-based grouping methods constitute the most common approaches used in proximity-based grouping of buildings in urban blocks. Among them, Minimum Spanning Tree (MST) is the most widely used algorithm. However, our experiences showed that this algorithm has limited effectiveness with respect to visual perception in case of the existence of isolated buildings in an urban block. This entails the investigation of alternative algorithms that is relevant to building grouping such as hierarchical and density-based clustering algorithms, which interestingly have not been tested for polygonal buildings in the blocks. In addition, quantitative assessment of the groups has not comprehensively been addressed so far. Previous works related to building grouping for generalization usually utilize from auxiliary geometric data structures representing relationships. Regnauld (2001) generates groups of buildings through MST using centroids of buildings. In order to obtain subgroups of buildings that are subject to typification, the inconsistent edges to be eliminated are decided by proximity, homogeneity (i.e. orientation, size) and the number of buildings. Li et al. (2004) create Delaunay triangulation (DT) using the vertices of buildings. Buildings whose vertices belong to same triangle are labelled as 'neighbour'. Direct alignments between neighbouring buildings are found by means of Gestalt theory. The indirect alignments among a row/column of buildings are then built upon the direct alignment relations. So, indirect alignments can be considered as special building groups. Zhang et al. (2013) produce MST from constrained DT. A grouping process is performed by removing all of the inconsistent edges of MST based on proximity property. Pattern recognition process follows the grouping process. Anders et al. (1999) present an approach based on graph clustering techniques for automated analysis of settlement structures represented by points (i.e. centres of gravity of buildings). Anders (2003) demonstrates the success of neighbourhood graphs in finding object groups in a natural way without any parameter. Yan et al. (2008) use DT to detect topological adjacency relations of buildings and generated 2-building groups firstly, and then constructed larger, intermediate groups according to a set of rules. After aggregation and separation of intermediate groups owning common buildings, final groups were created which can be used as a basis for generalization. It should be noted that grouping and clustering terms are used interchangeably throughout the article.

This study focuses on the comparative analysis and assessment of representative grouping algorithms for buildings in urban blocks. More specifically, answers to the following research questions are sought by considering groups with different distribution characteristics (i.e. homogeneous, heterogeneous, dense and sparse groups; see Section 2.2).

(1) How effective are representative clustering algorithms in finding groups of buildings in urban blocks?

 (a) How do the algorithms respond, when distribution of buildings in the groups changes?
 (b) Howdo the algorithms respond, when blocks contain isolated buildings?
(2) How can the quality of the clustering results be measured quantitatively?

This article is organized as follows. Selection of clustering algorithms, data pre-processing, grouping algorithms and grouping assessment methods are described in Section 2. The experiments consisting of data and software, the implementation and the result sections are presented in Section 3. A discussion is made in Section 4. Finally, the conclusion is given.

Methodology

Clustering algorithms have been utilized in obtaining building groups with the block basis, and cluster validity assessment techniques have been used to compare the results of the grouping algorithms. Flow chart of the study is given in Figure 1.

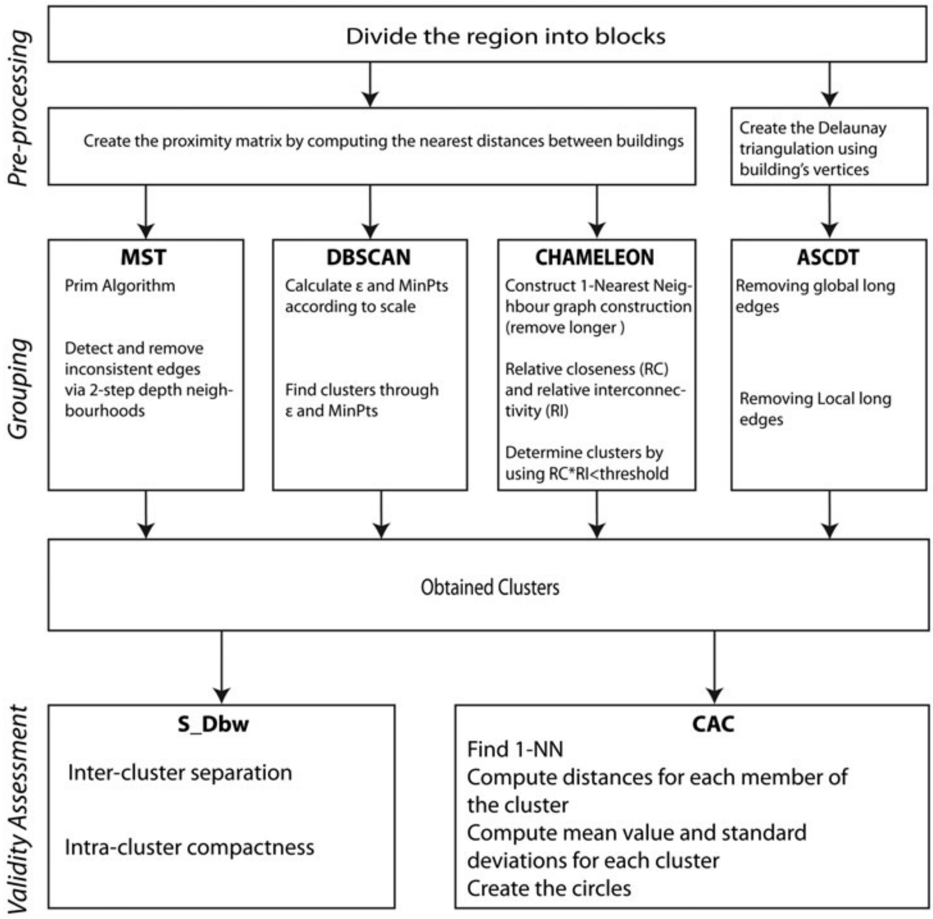

Pre-processing
Divide the region into blocks
Create the proximity matrix by computing the nearest distances between buildings
Create the Delaunay triangulation using building's vertices
Grouping
MST
Prim Algorithm
Detect and remove inconsistent edges via 2-step depth neigh-bourhoods
DBSCAN
Calculate ε and MinPts according to scale
Find clusters through ε and MinPts
CHAMELEON
Construct 1-Nearest Neig-hbour graph construction (remove longer)
Relative closeness (RC) and relative interconnec-tivity (RI)
Determine clusters by using RC*RI<threshold
ASCDT
Removing global long edges
Removing Local long edges
Obtained Clusters
Validity Assessment
S_Dbw
Inter-cluster separation
Intra-cluster compactness
CAC
Find 1-NN
Compute distances for each member of the cluster
Compute mean value and standard deviations for each cluster
Create the circles

Figure 1. Flow chart of the study.

Selection of algorithms

A large number of clustering methods are reported in the literature. It is difficult to provide a crisp categorization of clustering methods because these categories may overlap so that a method may have features from several categories (Han et al. 2009). In general, the major clustering methods can be classified into the categories shown in Table 1 (after Han et al. 2009; Rokach 2010).

Partitioning methods classifies the data into k groups, which together satisfy the following requirements: (1) each group must contain at least one object and (2) each object must belong to exactly one group. Such a partitioning method creates an initial partitioning. It then uses an iterative relocation technique that attempts to improve the partitioning by moving objects from one group to another. Such methods can find only spherical-shaped clusters and encounter difficulty in discovering clusters of arbitrary shape and cannot determine noises. *Hierarchical methods* create a hierarchical decomposition of a given set of data objects. Hierarchical methods suffer from the fact that once a step (merge or split) is done, it can never be undone. *Density-based methods* have been developed based on the notion of density. Their general idea is to continue growing a given cluster as long as the density (the number of objects or data points) in the 'neighbourhood' exceeds a threshold. Such a method is able to filter out noises (outliers) and discover clusters of arbitrary shape. *Grid-based methods* quantize the object space into a finite number of cells that form a grid structure. All of the clustering operations are performed on the grid structure (i.e. on the quantized space). The main advantage of this approach is its fast processing time, which is typically independent of the number of data objects and dependent only on the number of cells in each dimension in the quantized space (Han et al. 2009). *Graph-based methods* produce clusters via graphs. The edges of the graph connect the instances represented as nodes. Inconsistent edges are edges whose weight (in the case of clustering-length) is significantly larger than the average of nearby edge lengths (Rokach 2010).

Since it is impossible to test all of the algorithms in a single study, four representative grouping algorithms have been selected among the above categories. Main idea for selection of the algorithms is to try a representative algorithm from each category as far as possible. On the other hand, graph-based methods are more preferred for grouping buildings over other methods since they give results having the best agreement with

Table 1. Categories of grouping methods.

Categories of methods	Algorithm
Partitioning	k-means
	k-medoids
	CLARANS
Hierarchical	CURE
	BIRCH
	CHAMELEON
Density-Based	DBSCAN
	OPTICS
	DENCLUE
Graph-Based	MST
	ASCDT
Grid-Based	STING
	CLIQUE
	WaveCluster

human performance (Anders et al. 1999). Therefore, four algorithms have been selected: MST (a commonly used graph-based method), ASCDT (a recent graph-based method), Density Based Spatial Clustering Application with Nois (DBSCAN, a popular density-based methods) and CHAMELEON (a popular hierarchical method). Any algorithm from partitioning and grid-based methods has not been included because the former finds only spherical-shaped clusters, which is rather uncommon for the building groups and the latter generates the similar results with density-based methods (Han et al. 2011).

Data-specific issues

Hydrographic (e.g. river) and transportation (e.g. road) objects create logical boundaries around buildings and built-up areas. To prevent topological and logical inconsistency, buildings must not move to the other side of these objects. So, map space is partitioned into blocks using these kinds of surrounding objects by regarding their symbol sizes via buffer and overlay analysis. In other words, buildings belong to different blocks are assured not to be member of same cluster in this way (Basaraner & Selcuk 2008).

Buildings exhibit different distribution characteristics and so their groupings can vary. Group distribution characteristics (i.e. distribution of the buildings in groups) are identified by two aspects: homogeneity and density. If the nearest neighbour distances between buildings are uniform then the group is considered as ‘homogenous’, otherwise ‘heteregenous’. Group density is described with the average nearest neighbour distance between buildings and can be ‘dense’ or ‘sparse’. Figure 2 shows possible distribution characteristics that can be confronted in urban blocks.

In clustering process, nearest distances between polygons instead of distances between centres of gravities (CoGs) of buildings were used. As can be seen in Figure 3, the latter does not consider size, shape and orientation of buildings, so it can negatively affect clustering results. A proximity matrix was computed with the nearest distances between building polygons in each block but same parameter values were used for the algorithms in all of the blocks. ASCDT employs DT, while the other three methods use the proximity matrix as input.

Grouping algorithms

The clustering algorithms were specifically developed for point data. However, buildings are usually represented with polygons in spatial data-sets and spatial distances can

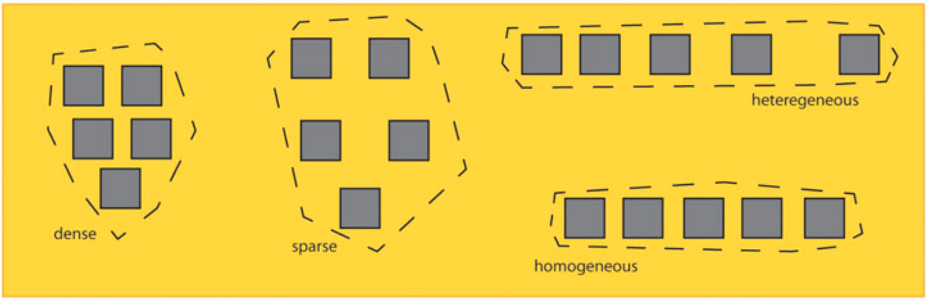


Figure 2. Distribution characteristics of buildings in the groups.

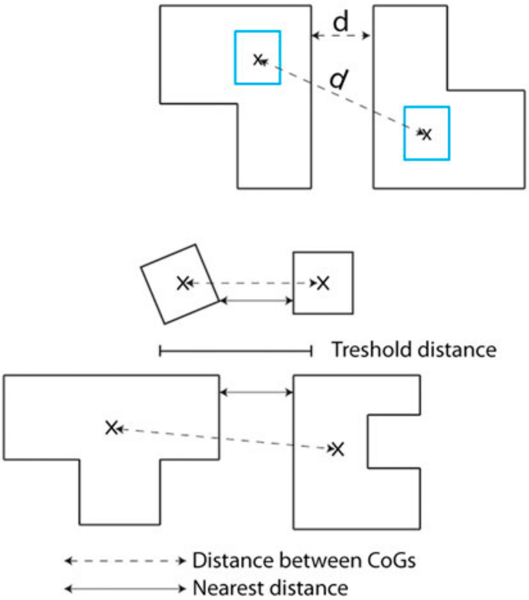


Figure 3. Disadvantage of using distances between CoGs.

be derived in different ways for polygonal buildings as mentioned in Section 2.2. Moreover, buildings take places in the blocks enclosed by linear features such as roads and grouping is performed on the block basis. Thus, the number of the buildings processed simultaneously is considerably less than the point data-sets that are used in the other clustering applications. For these reasons, grouping algorithms should be used regarding these characteristics of building groups.

MST carries out grouping process in two phases. First, MST graph is created. Then, inconsistent edges of MST are removed. Inconsistent edges are locally detected by utilizing Equation (1) (Zhang et al. 2010).

$$\text{edge}_i = \begin{cases} \text{if } w_i > I_l \cap w_i > I_r & \text{inconsistent} \\ \text{else} & \text{consistent} \end{cases} \tag{1}$$

where w_i denotes length of edge and I is a measure of significance that can be defined on both left (I_1) and right (I_r) sides of edge$_i$ and I is used as a dynamic threshold:

$$I = \max\{f \times \text{mean}, \text{mean} + n \times \sigma\} \tag{2}$$

where f and n are constants (parameters). Detailed discussion on the parameterization issue can be found in Zahn (1971).

For each edge of MST, using its neighbouring edges (with p-step depth), for its both sides, the threshold value (Equation (2)) is determined.

DBSCAN is proposed by Ester et al. (1996) for data mining purposes. It is a density-based algorithm with two global parameters, epsilon (ε) and minimum points (MinPts). An object is defined as a core object if its neighbourhood of radius ε contains at least MinPts objects (Figure 4). A core object is arbitrarily selected to begin clustering process. The objects within ε-neighbourhood of the core object and itself constitute a cluster. All members of the cluster are scanned for finding another core objects. If

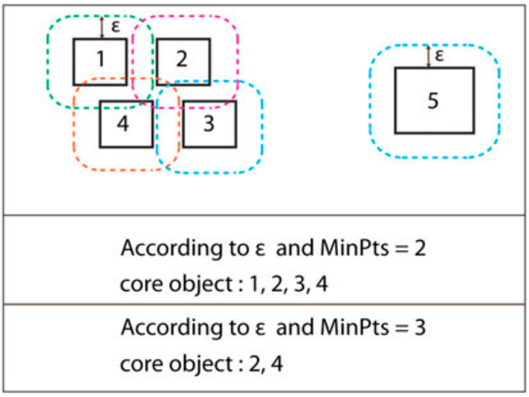


Figure 4. DBSCAN parameters.

found any, objects then within its ε-neighbourhood are added to the cluster and the scanning is resumed until all objects in the cluster are processed; otherwise a new core object that is not assigned into a cluster is selected to constitute a new cluster. This procedure continues until all core objects are assigned to a cluster. It should be noted that grouping with buffer technique used in Basaraner and Selcuk (2008) would give the same results with DBSCAN in case that MinPts equals 2 and ε equals to buffer width.

CHAMELEON is a hierarchical clustering algorithm using dynamic modelling proposed by Karypis et al. (1999). It generates clusters in two steps. In the first step, graph representation of the data-set is formed by the k-nearest neighbour graph. The resulting k-nearest neighbour graph is sparse and captures the neighbourhood of each node. CHAMELEON then applies a graph-partitioning algorithm, hMETIS (Karypis & Kumar 1999) to identify the clusters. In the second step, these clusters are further clustered using a hierarchical agglomerative clustering algorithm based on a dynamic model (relative interconnectivity and relative closeness) to determine the similarity between two clusters.

Since the number of buildings in a block is relatively small, hMETIS (Karypis & Kumar 1999) which is used to split a cluster into two sub-clusters cannot be performed. Instead of this, long 1NN edges have been removed using a statistical method to obtain initial clusters. The edges that are going to be removed are determined with Equation (3).

$$\text{if}\begin{cases} 1NN\text{ edge} > 0.5 \times \sigma + \mu & \text{remove} \\ \text{otherwise} & \text{keep} \end{cases} \tag{3}$$

where σ denotes the standard deviation and μ denotes the mean value of the 1NN edges.

Final clusters are too dependent on the pre-processing phase in CHAMELEON because, after pre-processing, only agglomeration is done but partitioning may be required for initial clusters.

ASCDT (Adaptive Spatial Clustering based on Delaunay Triangulation) is proposed by Deng et al. (2011). After obtaining a graph using DT, clustering process is performed by successively removing global long edges, local long edges and local link edges from the graph. One parameter, β, is only used for local long edges.

Actually ASCDT algorithm consists of three phases, final phase (i.e. removing of local link edges) is not necessary because necks and chain problem (Deng et al. 2011) is not expected for polygonal building data. Otherwise, (in case of performing last phase), edges that must be preserved can be removed wrongly.

Assessment methods for the grouping process

The assessment of the groupings is critical to objectively compare the results. One of the common methods used for this purpose is S_Dbw proposed by Halkidi and Vazirgiannis (2001). The S_Dbw index takes density into account to measure the inter-cluster separation. The basic idea is that for each pair of cluster centres, at least one of their densities should be larger than the density of their midpoint. The intra-cluster compactness is based on variances of cluster objects (Liu et al. 2010). S_Dbw index value is anticipated to be in the range between 0 and 1. If only one group emerges in a block, S_Dbw can just produce intra-cluster variance and it always equals to 1.

The cluster assessment circles (CAC) method is proposed in this study to be able to evaluate cluster distribution characteristics. This method does not give information about inter-clusters but intra-cluster. For each member of a cluster, first nearest neighbour distance (1-NND) is computed. For each group, mean value and standard deviation of the 1-NND are also computed. Based on the mean value and standard deviation, CAC is drawn (Figure 5). It means that the smaller the radius of the middle circle (r_{mean}), the more densely the cluster members are located in the group (i.e. dense cluster), and the closer inner and outer circles to each other (i.e. small Δr), the more homogeneous the group is.

S_Dbw index produces one index value for each block denoting the quality of grouping in the block. Although it takes into account both inter-cluster separation and intra-cluster distance, it does not give any specific information about individual cluster quality. In addition, S_Dbw cannot produce meaningful results, if the cluster shapes are concave and the data-set contains outliers (Halkidi & Vazirgiannis 2001). This deficiency is compensated with the visual interpretation. CAC index does not give specific

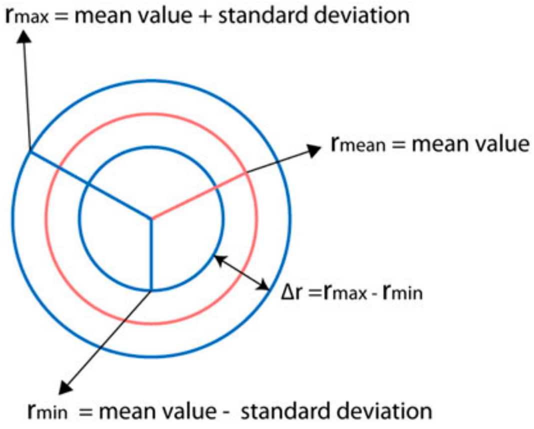

rmax = mean value + standard deviation
rmean = mean value
Δr =rmax - rmin
rmin = mean value - standard deviation

Figure 5. Elements of CAC.

information about cluster validity, but the information about the distribution characteristics of groups. So, it can contribute to the validity assessment except the existence of heterogeneous distribution in the blocks. They should be used together in the assessment of clusters. To be specific, S_Dbw and CAC provide global (i.e. block-based) and local (i.e. cluster-based) assessments, respectively.

Experiments

Data and software

A building data-set from 1: 25K topographic database was used in the experiments. This scale is base scale for medium-scale topographic data and map production in some countries. Eight urban blocks that represent typical distribution characteristics mentioned above have been selected. Blocks in densely built areas are excluded since it is impossible and unnecessary to find groups in these areas.

All the methods were coded in C++, except S_Dbw indexes which were computed in R^{TM}. Spatial data processing, analysis and visualization phases have been carried out with a GIS software.

Implementation

At the beginning, blocks were created through buffer and overlay analyses regarding road symbol sizes. Then, block-based proximity matrices were computed for MST, DBCAN and CHAMELEON algorithms, while DT was created for ASCDT algorithm. After that, parameters for clustering algorithms given in Table 2 are determined as follows:

- For MST, same parameter values proposed by previous studies such as Regnauld (2001), Zhang et al. (2010) and Zahn (1971) have been employed.
- For DBSCAN and CHAMELEON, parameter values have been determined by means of the sample blocks where different groups can be visually perceived.
- For ASCDT, parameter value has been selected as proposed by Deng et al. (2011).

Grouping process is performed with the developed code. S_Dbw index values are computed through the vertex coordinates (x, y) of building polygons and the cluster membership information of buildings. CAC values were computed through the proximity matrices and the cluster membership information of buildings.

Results

The building groups obtained from four different clustering algorithms are shown in Figure 6. The results of the assessment with S_Dbw and CAC values are given in

Table 2. Parameters of the algorithms.

Algorithm	Parameters
MST	$n = 3, f = 2, p = 2$
DBSCAN	MinPts = 2, $\varepsilon = 25$ m
CHAMELEON	RI*RC < 0.1
ASCDT	$\beta = 1$

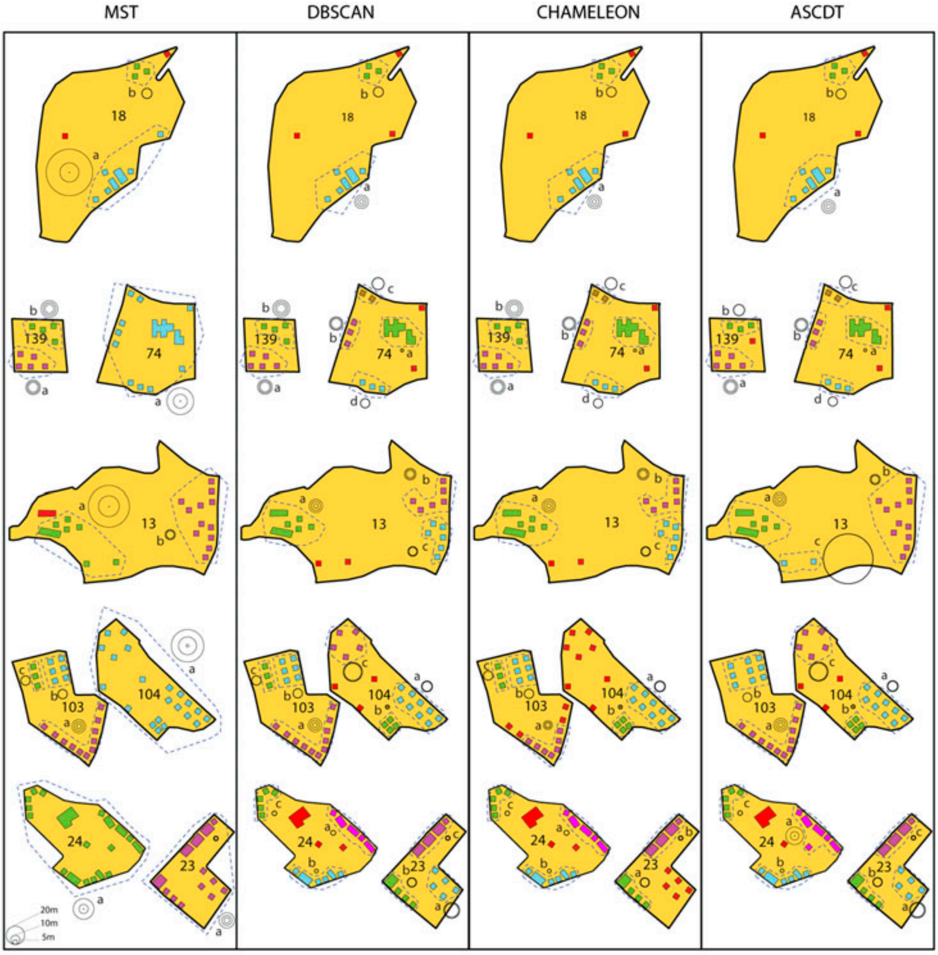


Figure 6. Building groups obtained by four clustering processes and their CAC diagrams. Buildings with same colour, delineated by dashed lines, represent a group except red buildings which are individual (isolated) and do not belong to any group. The CACs represent the mean value and the standard deviation of the 1-NN distance.

Figure 7 and Table 3, respectively. S_Dbw values converge to zero if cluster separation is distinct. Average S_Dbw index values are 0.13, 0.16, 0.22 and 0.60 for ASCDT, DBSCAN, CHAMELEON and MST, respectively. Accordingly, the clusters generated with DBSCAN, ASCDT and CHAMELEON are more distinctly separated than the ones generated with MST. r_{mean} value reflects the cluster density. If it is small, it means the cluster members are densely distributed (see CHAMELEON 104/b, in Figure 6). Otherwise, the cluster members are sparsely distributed (see DBSCAN 104/c, in Figure 6). Sparse clusters were detected only by DBSCAN and ASCDT, while dense clusters were determined by all the algorithms. However, the MST clustering was negatively affected from isolated buildings (see MST 18, 24, 74 and 104, in Figure 6). Δr value refers to homogeneity of a cluster. If it is bigger, it means distribution in a cluster is heterogeneous. Detection of the heterogeneous clusters was succeeded by ASCDT and DBSCAN (see ASCDT and DBSCAN 103/a in Figure 6).

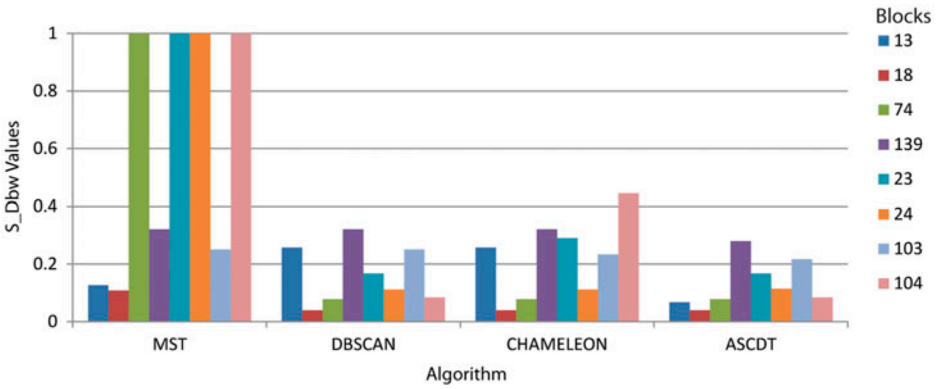


Figure 7. S_Dbw values of the algorithms corresponding to the blocks.

Table 3. CAC values of the algorithms.

		MST		DBSCAN		CHAMELEON		ASCDT	
Block	Cluster	r_{mean} (m)	Δr (m)	r_{mean} (m)	Δr (m)	r_{mean} (m)	Δr (m)	r_{mean} (m)	Δr (m)
13	a	23.06	47.33	10.07	8.91	10.07	8.91	10.07	8.91
	b	10.73	3.43	11.37	4.01	11.37	4.01	10.73	3.43
	c			10.09	2.45	10.09	2.45	57.30	0.00
18	a	22.86	71.35	10.41	10.97	10.41	10.97	10.41	10.97
	b	11.02	0.00	11.02	0.00	11.02	0.00	11.02	0.00
74	a	16.46	30.42	3.25	0.00	3.25	0.00	3.25	0.00
	b			12.77	4.27	12.77	4.27	12.77	4.27
	c			12.29	0.00	12.29	0.00	12.29	0.00
	d			10.69	0.43	10.69	0.43	10.69	0.43
139	a	14.36	5.86	14.36	5.86	14.36	5.86	14.36	5.86
	b	15.09	9.58	15.09	9.58	15.09	9.58	12.70	0.48
23	a	11.49	11.29	17.63	2.90	10.45	1.75	17.63	2.90
	b			10.45	1.75	4.85	1.91	10.45	1.75
	c			4.85	1.91			4.85	1.91
24	a	10.65	26.57	5.59	0.41	5.59	0.41	10.04	21.82
	b			4.53	0.89	4.53	0.89	4.53	0.89
	c			5.44	0.93	5.44	0.93	5.44	0.93
103	a	10.79	10.43	10.79	10.43	7.62	5.39	10.79	10.43
	b	10.34	0.72	10.34	0.72	10.34	0.72	10.34	0.70
	c	10.34	0.81	10.34	0.81	10.34	0.81		
104	a	19.04	33.11	12.19	2.01	12.19	2.01	12.19	2.01
	b			3.86	2.81	3.86	2.81	3.86	2.81
	c			21.46	2.74			21.46	2.74

Discussion

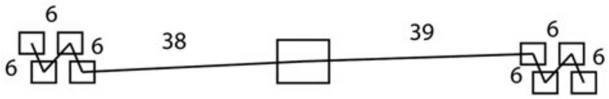
Our findings show that ASCDT and DBSCAN algorithms are the most effective, and CHAMELEON algorithm is partly effective, while the widely used MST algorithm is the least effective for detecting the building groups in the urban blocks.

The efficiency of MST was particularly low in blocks which comprise sparsely distributed clusters (see MST 23 and 104 in Figure 6). DBSCAN and ASCDT correctly found the groups with different distribution forms. However, efficiency of DBSCAN is too sensitive to the parameters. CHAMELEON detected the dense and homogeneous groups but was not able to find the sparse or heterogeneous groups.

In case of the existence of isolated buildings in a block, MST completely included such buildings into groups. This negative situation cannot be overcome with different values of parameter n (Figure 8). ASCDT tended to incorporate isolated building into a group (see ASCDT 24/a, in Figure 6) since long-building edges increase the threshold for local long edges (step 2 in ASCDT), if a building with long edges is connected to an isolated building. Because DBSCAN did not include any isolated building into any groups, it can be considered successful in this respect.

S_Dbw index indicates how distinctive the clusters in the blocks are. In this respect, according to Figure 7, groups detected by MST are the least distinctive. CAC represent both the homogeneity and density of the groups. If the radius of the middle circle (r_{mean}) is big, then the distribution in the cluster is sparse, else dense. r_{mean} values show the dense and sparse distributions with increasing order (see DBSCAN and ASCDT 104/b-a-c, in Figure 6). If difference between the radii of inner and outer circles (Δr) is big then cluster is heterogeneous else homogeneous. Δr values show the regularity of the distances among buildings (see DBSCAN 103/b, in Figure 6) and the irregularity (see DBSCAN 103/a in Figure 6).

Groups detected in this study have to be evaluated for the availability of specific patterns. In other words, these main groups based on only proximity criteria are segmented into meaningful subgroups if the geometric, semantic and/or structural characteristics and/or relationships of their objects are not same or similar. After the investigation of patterns, resulting subgroups are generalized individually but regarding relationships in main groups as far as possible. For example, the group consisting of six buildings (see ASCDT 18/a in Figure 6) contains buildings with different geometric characteristics and relationships (i.e. size, orientation) and structural characteristics and relationships (i.e. shape, alignment along the road). The group (see ASCDT 18/a in Figure 6) contains buildings with same geometric characteristics and relationships and also with same shape characteristics but different structural relationships (i.e. two distinct linear alignments). In this case, these two main groups first should be segmented and then generalized.



	edge$_i$ (38)	
	Left	Right
Edges Within 2 steps	6	39
	6	6
mean	6	22,5
stdDev	0	23,3
I for n = 3, f = 2	12	92,5
I for n = 2, f = 2	12	69,2
I for n = 1, f = 2	12	45,8

	edge$_i$ (39)	
	Left	Right
Edges Within 2 steps	38	6
	6	6
mean	22	6
stdDev	22,6	0
I for n = 3, f = 2	89,9	12
I for n = 2, f = 2	67,3	12
I for n = 1, f = 2	44,6	12

Figure 8. Disadvantage of MST grouping.

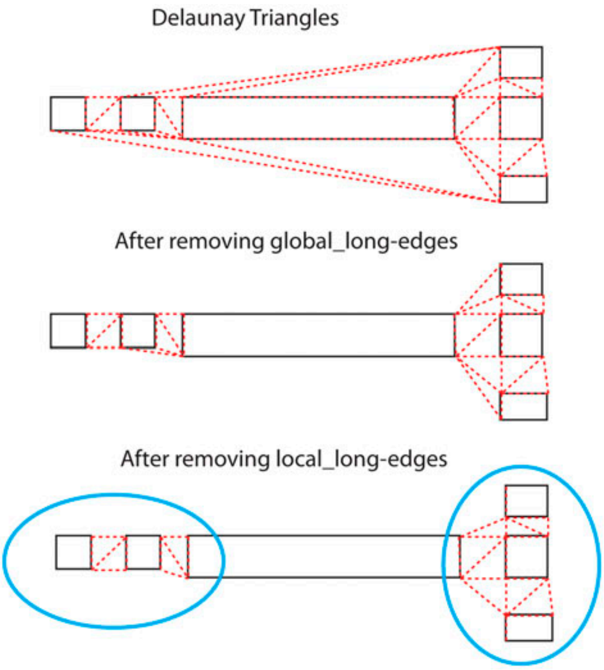

Delaunay Triangles
After removing global_long-edges
After removing local_long-edges

Figure 9. Necessity of the adaptation for ASCDT.

Parameter values of DBSCAN and CHAMELEON were determined according to 1:25K data-set, but setting their parameters for other scales should further be considered. ASCDT does usually not need any parameter adjustment and prior knowledge about data, but it requires additional adaptation to polygonal data (Figure 9). This may require interpolating extra points on the contour of the polygons. It can be considered that S_Dbw has potential usage in the assessment of the overall success of clustering algorithms, if the groups in the blocks are visually examined. However, S_Dbw does not work efficiently with concave-shaped groups (Halkidi & Vazirgiannis 2001) (see MST and ASCDT 13b and DBSCAN and CHAMELEON 13/b-c in Figure 6 and Block 13 in Figure 7). MST and ASCDT seems more successful than the others according to S_Dbw, since DBSCAN and CHAMELEON algorithms have found two concave clusters which incorrectly cause higher S_Dbw values. Besides, S_Dbw does not consider outliers (i.e. isolated buildings) in the evaluation process.

Conclusion

This article has presented a comparison of four algorithms for grouping buildings in urban blocks. Two different approaches have been used for the evaluation of the resulting groups. The idea was to investigate the responses of the clustering algorithms in case of polygonal building data and different distribution characteristics in the groups as well as demonstrate their advantages and disadvantages.

Although MST is widely used method to obtain building groups, it has produced too big groups if isolated building(s) and/or groups with different density (both sparse and dense) exist in a block. DBSCAN has produced good results independent from distribution characteristics in the groups, but epsilon (ε) parameter should be determined carefully. ASCDT works at both global and local manner, as well as does not need any threshold value. According to the S_Dbw index, it has generated better results but it has to be more adapted to polygonal data. CHAMELEON has not detected relatively sparse building groups but it has been successful in other group forms. As a result, ASCDT and DBSCAN algorithms seem more appropriate for grouping buildings in urban blocks.

S_Dbw is a commonly used method for the validity assessment of clusters; however, it does not work properly in case of non-convex groups and outliers. Another method, CAC has also been proposed to qualify distribution characteristics of the groups (i.e. intra-cluster evaluation).

Finally, as the findings of the study demonstrate that ASCDT and DBSCAN have good performance in grouping of buildings in urban blocks with different types of distributions. They can be used to group buildings in pattern recognition for generalization purposes.

Acknowledgements

We would like to thank to The Scientific and Technological Research Council of Turkey (TUBITAK) for the grant, and Mehmet Selcuk and the anonymous reviewers for valuable suggestions and comments.

References

AGENT Consortium. 1999. Selection of basic measures. Report DC1, The AGENT project; [cited 2014 Jan 8]. Available from: http://agent.ign.fr/deliverable/DC1.html.

Anders KH. 2003. A hierarchical graph-clustering approach to find groups of objects. Paper presented at: 5th ICA Workshop on Progress in Automated Map Generalization; Paris, France.

Anders KH, Sester M, Fritsch D. 1999. Analysis of settlement structures by graph-based clustering. In: Forstner W, Liedtke C-E, Buckner J, editors. SMATI 99: Semantic Modelling for the Acquisition of Topographic Information from Images and Maps; 1999 Sep 7; Munich.

Basaraner M, Selcuk M. 2008. A structure recognition technique in contextual generalisation of buildings and built-up areas. Cartogr J. 45:274–285.

Bobzien M, Burghardt D, Petzold I, Neun M, Weibel R. 2008. Multi-representation databases with explicitly modeled horizontal, vertical, and update relations. Cartogr Geogr Inf Sci. 35:3–16.

Burghardt D, Schmid S. 2010. Constraint-based evaluation of automated and manual generalised topographic maps. In: Gartner G, Ortag F, editors. Cartography in Central and Eastern Europe: Selected Papers of the 1st ICA Symposium on Cartography in Central and Eastern Europe. Lecture notes in geoinformation and cartography. Berlin: Springer; p. 147–162.

Deng M, Liu Q, Cheng T, Shi Y. 2011. An adaptive spatial clustering algorithm based on Delaunay triangulation. Comput Environ Urban Syst. 35:320–332.

Ester M, Kriegel HP, Sander J, Xu X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining; Portland, OR; p. 226–231.

Halkidi M, Vazirgiannis M. 2001. Clustering validity assessment: finding the optimal partitioning of a data set. ICDM 2001: Proceedings of the 2001 IEEE International Conference on Data Mining; 2001 Nov 29–Dec 2; San Jose, CA, USA.

Han J, Kamber M, Pei J. 2011. Data mining: concepts and techniques. 3rd ed. San Francisco (CA): Morgan Kaufmann. Chapter 10, Cluster analysis: basic concepts and methods; p. 443–496.

Han J, Lee J-G, Kamber M. 2009. An overview of clustering methods in geographic data analysis. In: Miller HJ, Han J, editors. Geographic data mining and knowledge discovery. 2nd ed. Boca Raton (FL): CRC Press; p. 149–188.

Karypis G, Han EH, Kumar V. 1999. CHAMELEON: Hierarchical clustering using dynamic modeling. Computer. 32:68–75.

Karypis G, Kumar V. 1999. Multilevel k-way hypergraph partitioning. Paper presented at: 36th Design Automation Conference; New Orleans, LA.

Li Z, Yan H, Ai T, Chen J. 2004. Automated building generalization based on urban morphology and Gestalt theory. Int J Geogr Inf Sci. 18:513–534.

Liu Y, Li Z, Xiong H, Gao X, Wu J. 2010. Understanding of internal clustering validation measures. Paper presented at: IEEE 10th International Conference on Data Mining; Sydney, Australia.

Regnauld N. 2001. Contextual building typification in automated map generalization. Algorithmica. 30:312–333.

Rokach L. 2010. A survey of clustering algorithms. In: Maimon O, Rokach L, editors. Data mining and knowledge discovery handbook. 2nd ed. New York (NY): Springer; p. 269–298.

Yan H, Weibel R, Yang B. 2008. A multi-parameter approach to automated building grouping and generalization. Geoinformatica. 12:73–89.

Zahn CT. 1971. Graph-theoretical methods for detecting and describing gestalt clusters. IEEE Trans Comput. C-20:68–86.

Zhang X, Ai T, Stoter J. 2010. Characterization and detection of building patterns in cartographic data: two algorithms. In: Yeh AGO, Shi W, Leung Y, Zhou C, editors. Advances in spatial data handling and GIS, Lecture notes in geoinformation and cartography. Berlin: Springer; p. 93–107.

Zhang X, Stoter J, Ai T, Kraak MJ, Molenaar M. 2013. Automated evaluation of building alignments in generalized maps. Int J Geogr Inf Sci. 27:1550–1571.