Procedural Modeling of Residential Zone subject to Urban Planning Constraints

Liying Wang, Wei Hua, and Hujun Bao

State Key Lab of CAD&CG, Zhejiang University,
310027 Hangzhou, P.R.China
{huawei,wangliying,bao}@cad.zju.edu.cn

Abstract. Besides spatial elements, their spatial configuration is an important factor affecting urban image. In practice of urban planning, good designers do thoroughly consider goals of urban design and constraints of urban planning. In this paper taking the layout problems of a residential zone as background, a framework of procedural modeling and a constrained layout optimization approach is presented to simulate the design procedure of human. The approach represents design goals as cost function subject to planning constraints. We can obtain the design plan by minimizing the objective function. During optimization, we adopt heuristic algorithm to get solutions efficiently for elements layout. Experiments show this approach is able to achieve similar design of residential zone to human work, by using less human resources.

Keywords: Procedural modeling; urban planning; constrained layout optimization

1 Introduction

1.1 Background

In recent years, many applications need highly efficient modeling for city model in both practical fields, such as digital city, national topography, urban planning and design, and virtual reality fields, such as digital entertainment, virtual learning, game and animation. So how to fast modeling city with highly realistic image becomes one promising research topic in many disciplines. New researches apply more interdisciplinary approach relating to urban planning and design, artificial intelligence, and computer graphics. Referring to the theory of urban planning and design, the basic elements forming urban image[1] consist of path, district, edge, node and landmark. Besides these spatial elements, their spatial configuration is an important factor affecting urban image through many combination ways which shows rich spatial texture, structure and human planning idea. Therefore visual design and spatial modeling for city involves these basic elements and their spatial configuration. However the spatial structure is so subtle that former researches mostly care for fast modeling elements and few combinations. Some works contribute to architecture modeling. Some deal with land use and functional zoning. Others focus on road network and building distribution. In this paper taking the layout problems of a residential zone as background, a framework of procedural modeling and a constrained layout optimization approach is presented. Layout problem is a kind of classical combinatorial optimization problem with a wide range of application. This problem means putting target objects into a bounded space or onto surface, satisfying some restrictions and reaching some optimal results. Layout problem has NP complete complexity for its geometrical and combinatorial features. Path planning is a key problem in fields of the intelligent robot and car navigator system. It also is a NPC hard problem.

1.2 Related Work

Nowadays Intelligent CAD research has been widely applied to achieve design goals semi-automatically in many fields, such as conceptual design, layout design, structural modeling for architecture and 3D scene modeling. Many modelers generate design by describing design intention as constraints and solving constraint satisfaction problem(CSP). Constraint description is classified to declarative and imperative method. From 1990's to today, there are many papers on declarative modeling. Several approaches have been attempted to solve CSP, for example mathematical programming, linear or non-linear equation solution, heuristic search of artificial intelligence. Ghassan[2] proposed hierarchical strength constraints of internal and external constraints to guarantee consistency of constraints. Coyne[3] constructed a WordsEye system to automatically convert text into representative 3D scenes. Generally CSP has many variables and iterative process need more time to get better solution, and during modeling users can intervene in an interactive and incremental way. But in general it is not suitable to generate results from a too vast search space.

Expert system tries to represent specific knowledge in abstract symbols and uses logical inference engine to get solution. Because inference process is limited to the specific problem, it rarely is adaptable to different situations. Grimsdale[4] expressed urban zoning and layout knowledge in pop-11 programming language and integrated inference engine with pop-11. It neither intervene the reference process directly nor support complex 3D models operation.

Grammar-based modeling approach has the advantage of rapidity and convenience by parameterization. Grammars such as L system, shape grammar, split grammar, are applied successfully to large-scale procedural modeling for plants, architecture and city. Parish[5] exploited extended-L system to model block partition, road network and simple building geometry. The highly abstract formalism of grammar had better include a small number of parameters, so grammar-based modeler does not show a bigger variety of urban patterns.

Object-oriented agent technique has well autonomous and extensible properties. Lechner[6] introduced agent technique to model urban zoning and road network. They avoided global rules manipulation, only processed local events and state query. Contrast to prior method, Guan[7] presented computer vision based registration method and imported augmented reality(AR) to planning residential zone. AR may reinforce people the sensibility to reality.

In the other hand, many researchers solved layout problems using AI algorithm to optimally place polygons on a bounded polygon without overlapping. Chen[8] used generic simulated annealing algorithm to search the placement order and orientation. A polygon spaces is filled with intervals data and heuristic search can exclude a lot of not good positions as earlier as possible and get an optimal layout. Particle swarm optimization is an efficient tool for non-linear and combinational and constrained optimization problems. Lei[9] et al presented a novel adaptive PSO based on multi-modified strategies to solve a layout problem of satellite cabins.

In general path planning method is heuristic search for its search space is too large and NP hard. Grid-based algorithm finds the best path by minimizing cost function. A-star algorithm[10] is a common approach to search a least cost path between a pair of grids, but it doesn't directly find a smooth path.

1.3 Contributions

Residential zone is the habitat of human being and one of important functional zones of city, its design often is a complex task of human. So a procedural modeling framework is presented to model the layout of its components following urban planning. Previous work used geometrical method for building distribution such as polygon subdivision and voronoi graph. Our former work introduced planning indices to control building density and height. Now we add spatial patterns to reinforce realistic and simulate the design procedure of human. Moreover parameterization makes it possible and controllable to fast modeling large scale scenes.

Secondly a constrained optimization approach is proposed to solve the layout problem. The approach represents design goals as cost function subject to planning constraints. The idea mainly comes from iterative adjustment and redesign during human design. Contrast to grammar-based method, the design plan is obtained by minimizing the constrained cost function. During optimization, we adopt heuristic algorithm to get solution efficiently.

Next we introduce the procedural modeling framework. Then constrained optimization approach, cost function and constraints are clarified. Solution including initial value decision and heuristic search are given in the following section. At last experiments are shown to conclude our method.

2 Procedural Modeling Framework

Fig.1 presents a hierarchical division in seven stages of city generation[11], we focus on two stages of lots and exteriors. We propose a procedural modeling framework shown in Fig.2 for residential zone with reference to a basic mode "patch-corridor-matrix" of modern landscape planning theory[12]. We specify components in a residential zone include residence, public facility, road, green land, tree, parking. Buildings (residence and public facility) within one patch develop a building cluster. Road includes two grades which are corridor connecting patches and path connecting buildings to corridors. The framework consists of interaction layer, solver layer and explanation layer.

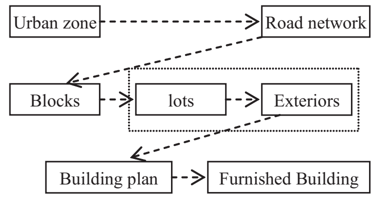


Fig. 1. Hierarchical division of city generation

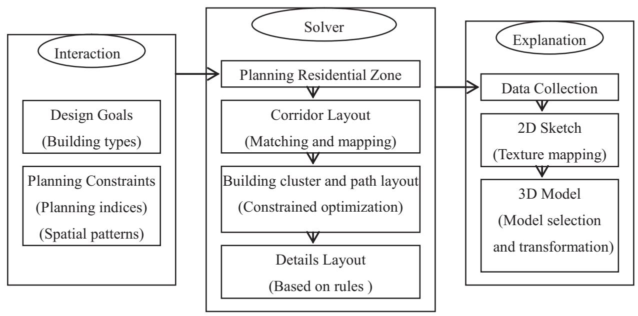


Fig. 2. Procedural modeling framework of residential zone

Interaction layer provides design goals and planning constraints. Design goals aim to put several kinds of buildings into the zone as many buildings and low costs as possible. Planning constraints include planning indices (building density, floor area ratio, the coefficient of sunshine spacing, green ratio) and spatial patterns shown in Fig.3. Spatial patterns describe the spatial structure of buildings and roads. The basic patterns of building are parallel and strew at random in different forms and the basic patterns of buildings group include determinant, centric, enclosed, symmetrical and dotted type. The basic patterns of paths have circle, direct and ending connection.

Explanation layer collects all components and outputs a 2D sketch with texture map and a 3D model whose elements are generated by transforming the selected basic models from a model library.

Solver layer solves three modeling sub-problems which are roads, building cluster and details layout. Details layout follows a certain rules, green lands are placed by searching plots in certain shapes subject to green ratio, trees are placed along road, parking facilities are beside roads.

Roads layout include corridor layout and path layout. For corridor, we reuse sample data and adapt them subject to the boundary and constraints of a planning zone. We use graph to express the boundary and corridor data. Given a planning graph(G') only with the boundary, the first step is to retrieve a match-

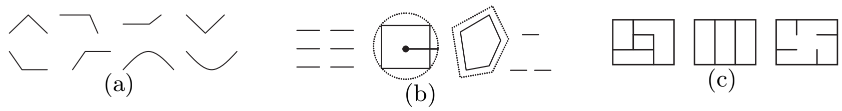


Fig. 3. Spatial basic patterns of (a) buildings (b)building groups (c)paths

ing sample graph(G). We apply a polygon shape matching algorithm[13,14] and the correlation degree of area and entry to get the most similar graph. Then G' adapts the corridor data of G. We express a graph as a topology with $\{V, E\}$, while V defines the nodes set with n members, E defines the edges set of the graph. V consists of two parts, V_p defines m nodes on the boundary, and V_g defines $v - m$ nodes inside the graph. According to mean value coordinates theory[15] a polygon's V_g can be formulated by V_p as $V_g = M_p V_p$, $M_p$is the matrix of mean value coordinates. When V_p and V_p' of G and G' are registered one by one and satisfy the alignment of feature nodes on both boundaries, then V_g' are calculated by $V_g' = M_p V_p'$. Finally the nodes and edges in G' are checked to satisfy the constraints of the planning zone. The corridors partition the planning zone into some patches and the sample planning patterns of G are reused as planning constraints. We construct a constrained optimization approach to layout buildings and paths within patches. Design goals are expressed as a cost function subject to planning constraints. By minimizing the cost function by heuristic algorithm buildings and paths are arranged efficiently.

3 Constrained Optimization Approach

Suppose we have the array of several types of buildings in a placement order, then what we optimize is to arrange the building position and orientation. The array is denoted as $\{B_i | i = 1..n\}$. A building B can be represented by a vector (A, h, v^b, p^g), where A is the building's 2D projection on x-z plane represented by a polygon, h is the height of building toward +y-axis, v^b is the orientation of the building's balcony which is toward the south namely +z-axis. p^g is the entrance to the building. The shape and the height of a building are determined by its type, but its position p and its orientation v^b is variable. Fig.4 shows a building's properties. The symbol sh represents the shadow of the building, v^g is the orientation of the entrance.

If one building is arranged a possible position, then it is constructed. What we solve for the array of buildings is a sequence of vectors $X = \{(p, v^b, s)_i | i = 1..n\}$, we use the building center to calculate p, s is a boolean variable to identify whether the building is constructed. And the orientation of the entrance v^g is from the center to the entrance. The exit path of the building is a sequence of path segments which connect the entrance to the exit p^t on the corridor.

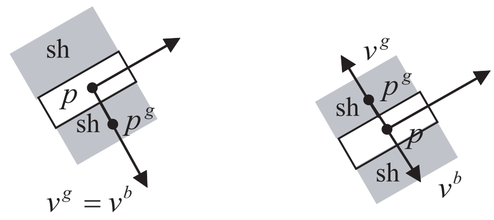


Fig. 4. The orientation relation between building's balcony and entrance,left is same, right is opposite

3.1 Cost Function of Buildings

We have mentioned that the design goals aim to put several kinds of buildings into the zone as many buildings and low costs as possible. So we propose a cost function of buildings is a weighted cost addition of constructing buildings and paths. The number of buildings is constrained by building density and floor area ratio. In order to put maximum buildings into the zone, we require the remainder building density and floor area ratio of the zone S is minimum. It is known that grouping buildings together to be border upon influences the building cost. The longer the number of buildings in a group is, the lower the cost of gable is. But the length can't be too long[12]. We express the cost of a building group C_t is a subsection linear function of the length and the total cost C of all groups is their building costs addition. And W is the cost of constructing paths. So we express the objective function is $f(X) = Min\{(w_1S + w_2C + w_3W)/(w_1 + w_2 + w_3)\}$ subject to planning constraints.

To avoid the shadow occlusion, residences are separated by a reasonable sunshine spacing. The shadow of a building is calculated approximately according to its orientation and the coefficient of sunshine spacing. The occlusion distance of a building at the southern orientation equals its height multiples the coefficient. We suppose that the orientation of shadow opposites the orientation of the building's balcony and the shadow is a rectangle region within the occlusion distance, shown in Fig.4. Any residence does not in shadows. In addition, any building has an exit path. The constraints of building density and floor area ratio, sunshine spacing and path existence are checked while positioning buildings. Moreover the building layout follows a certain patterns. We require the pattern distance $P(X)$ of buildings is minimum. In order to solve the constrained optimal problem, we use penalty function method[16] which constructs an augmented objective function to solve an unconstrained optimal problem. Thus we need minimize the augmented objective function $O(X) = f(X) + MP(X), M > 0$ to solve the building layout problem.

3.2 Pattern Distance of Buildings

We use control lines to describe basic patterns of buildings. We organize the control lines as a tree, each leaf node can be used to search the building position while others are used to develop spatial patterns for paths. Each node includes the lines information data describing the start and ending, the building orientation and path pattern. After positioning buildings, the pattern distance of buildings includes building orientation and patterns cost. Residences had better orientate to south, so we define a subsection linear function to calculate the building orientation cost. We define a linear function for pattern costs to express the spatial structure and comfort degree of a building cluster.

3.3 Cost Function of Paths

Path cost includes the segments cost, exits cost and patterns cost. The minimum cost of the path segments of a building represents shortest, smoother and uniformly roomy, that is $G = L + N + E$, while L is the length of the path segments of the building. When the directions of consecutive path segments are different, one turning occurs. So smooth degree is computed as the number of turnings N. Roomy degree E expresses the space surrounding the path segments. We compute roomy degree as the cost of the distance before a point of path meets the boundary of buildings or the patch. Its first part is computed along the directions of path segments whose value is zero when the distance is far otherwise is 1. Its second part is calculated along upward, downward, leftward, rightward directions to show whether the space of both sides is uniform, whose value is 1 when a distance of four directions is near and a difference of two directions is large between up and down or between left and right, otherwise is zero. The exits on corridors are shared by many buildings and constructed using more costs. Additionally, a linear function for pattern costs is defined to express the spatial structure and comfort degree of paths.

4 Heuristic Algorithm

4.1 Building Placement

Building placement is a standard 2D polygons layout problem subject to constraints: (1) A building locates inside the zone with setback.(2)A building doesn't intersect with others. (3)Residence isn't in shadows. (4)A building has an exit path. (5) Buildings satisfy the building density and floor area ratio. (6)Buildings try to surround control lines of pattern. We construct polygons with a flag(> 0 occupied by a residence, $= -1$ unavailable,$= -2$ shadow, < -4 occupied by a public facility).

Pairs of intersection develop internal intervals of a polygon, and we add a flag (flag$= 0$ available, > 0 occupied by a building, $= -1$ unavailable, $= -2$ shadow, $= -3$ setback space, $= -4$ shade) to manage the usage of a polygon's space.

Given an array of buildings in a placement order and patterns, we firstly solve the tree of control lines, collect background polygons filled with flags. Secondly we solve each of buildings position and orientation along control lines. When the building density and floor area ratio are not satisfied, or there isn't any space to place it, a building is not constructed and set its construction flag false. Otherwise the position on a line and the building orientation is used as the candidate value of the building. Next the building and its shadow polygons called foreground polygons are calculated and add to background after passing the placement constraints and exit path existence. Thereby the initial value of the building is decided by the candidate value, set its construction flag true. If there is any conflict, the candidate value is moved one step along its line and check again. If beyond its line, the candidate value is searched along next control line.

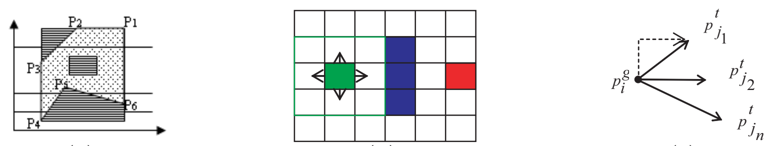
After getting initial values of all buildings, we need optimize the buildings' layout. Heuristic search may minimize the augmented objective function. We decide the descending direction through checking building groups one by one. If one building is border upon other and this group is too long then move tail part of the group to shorten it. When it is alone, move it along control line toward left or right neighbor to be border-upon other. The stages of optimal process are organized in table 1.

Table 1. the Stages of optimal process

Step1: Set building patterns and solve control lines
Step2: Initiate background polygons
Step3: Create the array of buildings and the sequence of variable vectors $X = \{(p, v, c)_i \mid i = 0..n\}$
Step4: Calculate initial value X_0,and update background polygons, compute function value $O(X)$
Step5: Set iteration generation G, $0 \le t < G$, set $t = 0$, $O(X(0)) = O(X_0)$
Step5.1: Initiate background polygons,Set $t = t + 1$ search descending direction $\triangle X(t) = (\triangle p_i, \triangle v_i, \triangle c_i)$ and max step λ, $X(t) = X_0(t) + \lambda \triangle X(t)$,$\Vert \triangle X \Vert = 1$
Step5.2: update background polygons, compute function value $O(X(t)$ as the fitness of $X(t)$
Step5.3: Choose $X = \min\{O(X(t-1)), O(X(t))\}$
Step5.4: When $t > G$ go to Step6.Else go to Step5.1
Step6: Search initial values for those buildings whose construction flags are false. If a building's construction flag is changed to be true, go to Step4.
Step7: Collect buildings' parameters, and change buildings from parallel to stew at random pattern.
Step8: Collect buildings' polygons and add to background polygons.

4.2 Path Planning

Building patterns and path patterns assist in our path planning. We plan paths in a building group to connect entrances each other and plan paths among groups to share the optimal exits. For a pair of points, path planning applies a modified A-star algorithm which can find a minimum cost path for a pair of grids. In order to obtain initial background grids, the background polygons are filled with scanning line shown in Fig.5(a). Pairs of intersection develop internal intervals of a polygon, and we add a flag (= 0 available, > 0 occupied by a building, = −1 unavailable, = $-2shadow$, = −3 setback space, < −4 occupied by a public facility) to manage the usage of a polygon's space. Then background intervals are partitioned along vertical direction. The grids on corridor are exit grids. We modify A-star algorithm three points. First is to check whether a direct path exists between a pair of grids, second is path doesn't pass diagonal grids, which means current grid is not the parent of diagonal grids, shown in Fig.5(b). Third is to impose unfavorable cost on those grids which result in turning and narrow room, which adds extra cost to the G part which has three items: path length, turning number and roomy degree. The process steps are shown in table 2.



(a) (b) (c)

Fig. 5. Path planning (a) background polygons filled with scanning line (b)Modified A-star algorithm: green grid is the entrance, blue grid is the obstacle, red grid is the target. (c) Optimal exit search algorithm for a beginning grid

Table 2. Optimal exit search algorithm for a beginning grid

Step1: Calculate the distance d_j from p_i^g to $p_j^t, j = 1..n$ then sort them to make $d_{j_1} < ... < d_{j_n}, j_1..j_n$ is the index of exits.
Shown in Fig.5(c), the worst case of the method is to search all exits
Step2: Set $d^r = \infty, m = 1$
Step3: When $m < n$, get j_m, Else stop. if path exist from p_i^g to $p_{j_m}^t$, and path length is L_{j_m} and turning number is N_{j_m}so the path cost $d_{j_m}^r = L_{j_m} + N_{j_m}$
Step4: If $d_{j_m}^r < d^r$ then set $d^r = d_{j_m}^r$otherwise set $m = m + 1$, go to Step3
Step5: If $N_{j_m} = 0$ then d^r is minimum (it is easy to prove). So stop and set $j = j_m$.Otherwise set $m = m + 1$, go to Step3

5 Experiments

We code the basic patterns of buildings and paths using numbers shown in table 3. And set the coefficient of sunshine spacing is 1 and the height of one floor is three meters, the dimension of a building is its length, height, width. We respectively use two patterns of corridors to partition the zone to plan the residential zone whose area is 108 thousand square meters. We give several types of buildings and pattern choices for each of patches, shown in table 4. We can get 2D sketch and 3D effect shown in Fig.6 in several minutes satisfying planning goals given in table 5.

Table 3. pattern definition

Building pattern(sp)	Empty	Determinant	Centric	Enclosed	Symmetry	Dotted
Value	0	1	2	3	4	5
Path pattern(pp)	Circle connection		Direct connection		Ending connection	
Value	1		2		3	
Path direction(sp)	Same	Opposite	Inside	Outside		
Value	1	0	1	0		

Table 4. Building types and pattern choices

Demo	Name	Value					
1	Residence dimension(m)	10.8,18,10.2		16,24,13	18.8,18,15.4		20,9,18
	Facility dimension(m)	30,9,30			15,6,10		
	Residence Pattern(sp/pp/pd)	1/3/1	1/2/0	0/0/0	/2/1/0	1/2/0	5/1/0
	Facility Pattern(sp/pp/pd)	1/2/0	1/2/0	1/2/0	0/0/0	1/2/0	1/2/0
2	Residence dimension(m)	16.3,60,12.3		13.9,36,13	13.8,60,14.5		
	Facility dimension(m)	15,6,10					
	Residence Pattern(sp/pp/pd)	1/3/0	1/2/0	2/2/1	2/2/1		
	Facility Pattern(sp/pp/pd)	1/2/0	1/2/0	1/2/0	1/2/0		

Table 5. Planning indices

Main Indices (plot (100%))	Planning Goals			Planning Results						
	Floor area ratio	Building density (%)	Green ratio (%)	Floor area ratio	Building density (%)	Green ratio (%)	Residence plot (%)	Facility plot (%)	Path plot (%)	Parking ratio (%)
Demo1	1.1	28	40	1.06	18.1	22	15.2	2.9	17.7	4.5
Demo1	1.8	25	40	1.6	13.4	26.4	9.5	3.9	14	2.1

6 Conclusion and Future Work

Experiments show our modeling framework is highly efficient to present spatial layout of residential zone subject to users' intention. Compared to the real image and human design shown in Fig.7, it is found that we can obtain similar planning to human work by using less human resources. Moreover our planning is useful for designers to improve incrementally and redesign more details. In future we are going to research more flexible patterns to do planning.

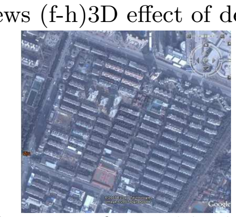
(a) demo 1: Multi-storey Residence

(b) demo2:High-rise Residence

(c) (d) (e)

(f) (g) (h)

Fig. 6. Experiments of 2D planning and 3D effect: (c-e)3D effect of demo 1 from different views (f-h)3D effect of demo 2 from different views



(a) Image of gound building

(b) Human detail building planning

Fig. 7. comparison to human planning

Acknowledgements

This research work is supported by the 973 Program of China (No. 2002CB312102) and the Cultivation Fund of the Key Scientific and Technical Innovation Project, Ministry of Education of China (No.705027) and the National High Technology Research and Development Program of China(863 program)(No.2006AA01Z338).

References

1. Wan Suzhen, et al: Urban Spatial Structure Analysis. Scientific Publishing House (2004)(in Chinese)
2. Ghassan, K., Veronique, G., Rene, C.: Controlling Object Natural Behaviors in a 3D Declarative Modeler. In proceeding of Computer Graphics International (1998)248-253
3. Coyne, B., Sproat, R.: WordsEye: An Automatic text-to-Scene Conversion System. In proceeding of ACM SIGGRAPH (2001)487-496
4. Grimsdale, R. L., Lambourn, S.M.: Generation of Virtual Reality Environments using Expert Systems. In proceedings of WSCG (1997)153-162
5. Parish YIH, Mller, P.: Procedural modeling of cities. In proceeding of ACM SIGGRAPH (2001)301-308
6. Lechner, T., Watson, B. et al: Procedural Modeling of Urban Land Use. In proceeding of ACM SIGGRAPH (2006)
7. Guan Tao, Li Lijun: The Application Research of Augmented Reality in Residential Area Planning , Journal of Engineer Graphics (2006)5:50-54(in Chinese)
8. Chen Yong, Tang Min, Tong Ruofeng et al.: Packing of Polygons Using Genetic Simulated Annealing Algorithm. Journal of Computer-Aided Design & Computer Graphics (2003)15(5):598-603(in Chinese)
9. Lei Kaiyou, Qiu yuhui: A Study of Constrained Layout Optimization Using Adaptive Particle Swarm Optimizer. Journal of Computer Research and Development(2006)43(10):1724-1731
10. Cai Zixing, Xu Guangyou, et al.: Artificial Intelligence and Application. QingHua University Publishing House (1996)(in Chinese)
11. Larive, M., Dupuy, Y., and Gaildrat, V.: Automatic Generation of Urban Zones. In proceeding of WSCG(2005)9-12
12. Li Dehua: Urban Planning Theory. Chinese Architecture Industry Publishing House (2001)(in Chinese)
13. Arkin, M., Chew, L. P. et al: An Efficiently Computable Metric for Comparing Polygonal Shapes. IEEE Trans. Pattern Analysis and Machine Intelligence(1991)13: 206-209
14. Xie Ping, Ma Xiaoyong et al.: A New Fast Algorithm for Complex Polygons Matching, Computer Engineering(2003)29(16):177-181(in Chinese)
15. Floater, M. S.: Mean value coordinates. Computer Aided Geometric Design(2003) 20(1): 19-27
16. Yuan Yaxiang, Sun Wenyu: Optimal Theory and Method. Scientific Publishing House(1997)(in Chinese)