Web3d Learning Platform of Furniture Layout Based on Case-Based Reasoning and Distance Field

Peihua Song[1], Youyi Zheng[2], and Jinyuan Jia[1](✉)

[1] School of Software Engineering, Tongji University, Shanghai 201804, China
sph2000@126.com, jyjia@tongji.edu.cn

[2] School of Information Science and Technology, ShanghaiTech University,
Shanghai 200120, China

Abstract. For non-professional designers, it is a difficult task to arrange furniture to the right position without the interior designer's guidance. We have developed a web3d learning platform of furniture layout, which can automatically generate a reasonable layout to help people learn knowledge of interior design and improve efficiency. First the non-rectangular room shape normalization algorithms are proposed. Next, four layout modes, namely, coupled mode, enclosed mode, matrix mode and circular mode are introduced; the coupled mode and enclosed mode are solved using the case-based reasoning and distance field, respectively. Finally the furniture layout algorithm of non-rectangular bedroom and that of non-rectangular living room are given. The experimental results show that layout results of the algorithms are similar to the real designs of non-rectangular room, and the running time of the algorithms can meet the needs of the online furniture layout.

Keywords: Furniture layout · Case-based reasoning · Distance field · Learning platform

1 Introduction

The scene layout problem can be widely found in many domains such as city planning, architectural design, interior design, and three-dimensional scene modeling. It is the research focuses in computer graphics, ergonomics and optimization design. Furniture layout is one of the most important research subjects in the scene layout, which are traditionally performed manually. With the development of the internet technology and web 3D technology, there are many online three-dimensional interior design platforms, such as Home-style online platform developed by Autodesk Company. An ordinary user who has no professional interior designer knowledge can now arrange furniture using these online platforms semi-automatically, and don't need to install the off-line interior design software. However, since the furniture layout contains a lot of ergonomic knowledge which requires professional skills, it is not an easy task for ordinary user to arrange furniture in appropriate location without the guide of the designer teachers.

F. Tian et al. (Eds.): Edutainment 2017, LNCS 10345, pp. 235–250, 2017.
https://doi.org/10.1007/978-3-319-65849-0_26

To solve the problem, many scholars have proposed automatic furniture layout algorithms to guide users to arrange furniture, which also can help ordinary users to learn knowledge of interior design while reducing workload and improving efficiency for interior designer. In particular, at the international SIGGRAPH conference in 2011, two scholars have discussed this issue further [1, 2]. However, those algorithms mainly use a single case study technology or intelligent algorithm, and some of those algorithms are of long running time, and off-line. Few studies have focused on online automatic furniture layout. The requirements of online furniture automatic layout algorithms are the effectiveness, robustness, and real-time response. There remains a need for furniture layout algorithms.

In this paper, we first present the room shape standardization algorithm to extract the rectangular layout region from the non-rectangular rooms. Next we introduce four layout modes: coupled mode, enclosed mode, matrix mode and circular mode, and propose corresponding algorithms to the individual modes based reasoning and the distance field. Finally the bedroom layout algorithms and living room layout algorithms are given. Experimental computations suggest that the algorithms are extremely efficient.

2 Related Work

There exist many algorithms to the furniture automatic layout problem. Those algorithms can be divided into two categories: the algorithms based on optimization model and the algorithms based on case study.

2.1 Algorithms Based on Optimization Model

Those algorithms first give an evaluation function according to furniture layout rules defining the distance between the furniture and the angle and so on. Then, the layout result was calculated by using the evaluation function and intelligent algorithms [1–5]. Yu et al. [1] presented an algorithms based on the simulated annealing algorithm. The evaluation function of the algorithm mainly considers accessibility, visibility, pathway constraints and so on. Merrell et al. [2] presented an interactive furniture layout system based on interior design guidelines. The algorithm incorporates the layout guidelines as terms in a density function, and then generates layout suggestions. Chen et al. [3] presented a hierarchical optimization strategy and used particle swarm optimization to solve the furniture layout problem. Akase et al. [4] introduced a way of construction of gene expression and evolution process for interactive evolutionary computation. Liu et al. [5] proposed the composite model for home furnishing generation based on furniture configuration process.

The result of those algorithms depends on the rule values setting at the beginning of the layout. If the information of the furniture and that of room are changed, the rule values need to be reset and the parameters of the intelligent algorithm need to be artificially adopted.

2.2 Algorithms Based on Based on Case Study

Akazawa et al. [6] used the distances between furniture to denote the constraint relationships stored in the database. The query furniture is arranged by retrieving the database. Germer et al. [7] also presented an algorithm based on case study. But the two algorithms do not consider the relationships between the furniture layout and position of door and window. Kjølaas [8] used a set of default templates provided for most common room and furniture types to solve the furniture layout problem. Fisher et al. [9] presented a method for synthesizing 3D object arrangements from examples. The method synthesizes a diverse set of plausible new scenes by learning from a database. In the authors' previous work [10], an algorithm for rectangular room has been introduced, but the algorithm cannot solve the problem of non-rectangular room furniture layout. In this paper, we adopt the room shape standardization algorithms, improved case - based reasoning algorithms and distance field to deal with the problem of non-rectangular room furniture layout.

3 Learning Platform Overview

Figure 1 gives an overview of interaction with our platform. First the users create the room and select the furniture in a library using the online three-dimensional platform. Next, a rectangular layout region is extracted by room shape standardization algorithms. Then individual algorithms for the coupled mode and enclosed mode are introduced based on case-based reasoning and distance field. We omit the algorithms for the matrix and circular mode as they are simple and can be similarly solved. Finally the platform generates layout result to the users. The layout result and the layout case library can assist the users to learn arrange furniture knowledge, to reduce the labor intensity and to improve the labor efficiency. The uses can obtain the desired layout form the platform.

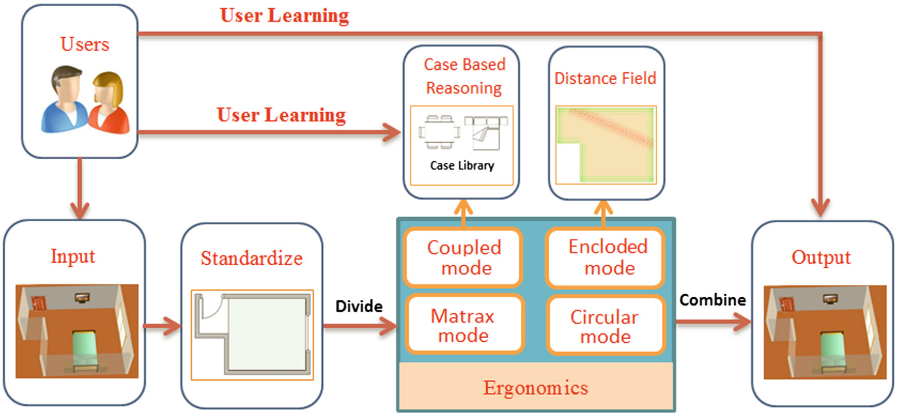


Fig. 1. Platform overview.

4 Room Shape Standardization Algorithm

The aim of room shape standardization algorithm is to extract the rectangular layout region from the non- rectangular room. Next we propose the bedroom shape standardization algorithm and the living room shape standardization algorithm.

4.1 Bedroom Shape Standardization Algorithm

A room shape is represented as a polygon $P = p_1p_2 \cdots p_n$ (see Fig. 2(a)). The polygon P contains two sets: a set of vertex $P = \{p_1, p_2 \cdots\ p_n\}$ and a set of line segments $L = \{p_1p_2, p_2p_3 \cdots\ p_{n-1}p_n, p_np_1\}$. A line segment p_ip_{i+1} represent a wall in the room with $1 \leq i \leq n$. The point p_{door} and point p_{window} represent the door position and window position, respectively. Our aim is to extract a rectangular layout region form the polygon P. The bedroom shape standardization algorithm works as follows. First, we find the bounding box of the polygon P, namely rectangle $Q = q_1q_2q_3q_4$, and its center point O (see Fig. 2(b)). Next, we obtain four midpoints of edges of layout rectangular region, namely point r_1, point r_2, point r_3 and point r_4, by those walls (see Fig. 2(c)). Finally, as is shown in Fig. 2(d), we obtain the rectangular layout region $S = s_1s_2s_3s_4$ according those midpoints. The algorithm is given below.

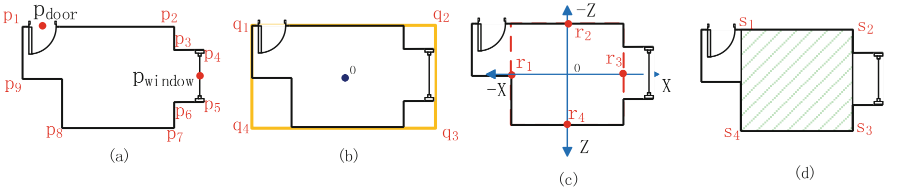


Fig. 2. An example of bedroom shape standardization algorithm. (a) The shape of a bedroom. (b) The bounding box of the room. (c) The midpoints of edges of the rectangular layout region. (d) The rectangular layout region.

Algorithm 1. Bedroom shape standardization

Input: a room shape represented as a polygon $P = p_1p_2 \cdots p_n$.

Output: the rectangular layout region $S = s_1s_2s_3s_4$.

Step 1: Find the polygon's bounding box of $Q = q_1q_2q_3q_4$ and its center point O.

Step 2: Establish a two-dimensional coordinates whose origin is the point O. Let the positive direction of the coordinate axis x to be form the point O to the point p_{window}.

Step 3: Along all line segments. Those line segments and the negative X axis, the negative Z axis, positive X axis, positive Z axis are intersected at the point set R_1, point set R_2, point set R_3 and point set R_4, respectively.

Step 4: Find the point r_1 whose x coordinate is the largest in the point set R_1. Find the point r_2 whose y coordinate is the smallest in the point set R_2. Find the point r_3 whose x coordinate is the smallest in the point set R_3. Find the point r_4 whose y coordinate is the largest in the point set R_4.

Step 5: Construct the rectangular layout region $s_1s_2s_3s_4$ by the point r_1, point r_2, point r_3, point r_4 and point O, and then return the region $s_1s_2s_3s_4$.

4.2 Living Room Shape Standardization Algorithm

As is shown in Fig. 3(a), the polygon $P = p_1p_2 \cdots p_n$ represent a living room shape. The rectangle $s_1s_2s_3s_4$ is the sofa and TV layout region which is extracted using living room shape standardization algorithm. The algorithm is given below.

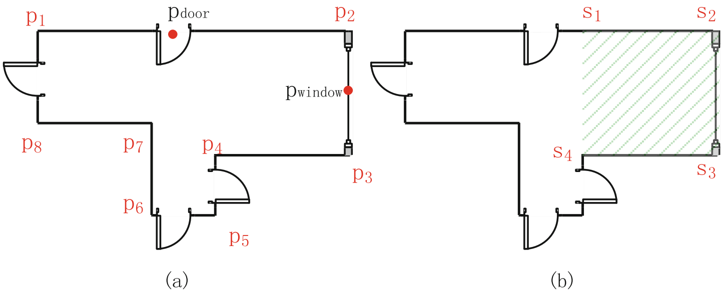


Fig. 3. An example of living room shape standardization algorithm. (a) The shape of a living room. (b) The rectangular layout region.

Algorithm 2. Living room shape standardization

Input: a living room shape represented as a polygon $P = p_1p_2 \cdots p_n$.

Output: the sofa and TV layout rectangular region $S = s_1s_2s_3s_4$.

Step 1: Find the three walls p_1p_2, p_2p_3, p_3p_4 which are adjacent to the window.

Step 2: Find all the door position set D on the walls p_1p_2, p_3p_4, and then calculate the distance between the point p_{window} with the point p_1, point p_4 and the points in set D. Find the point p_{min} to be of minimum distance. As is shown in Fig. 3(b), the point p_{min} is p_4.

Step 3: Construct the rectangular layout region $s_1s_2s_3s_4$ using the point p_2, point p_3 and point p_{min}, and then return the region $s_1s_2s_3s_4$.

5 Furniture Layout Mode

A furniture layout mode refers to the layout relationship between the furniture and room. We define four layout modes. (1) Coupled mode is to require the furniture and room to maintain a certain distance and angle to each anther, such as the bed and bedside cabinet (see Fig. 4(a)). It can be divided into two categories: furniture and furniture coupled mode, furniture and room coupled mode. (2) Enclosed mode is to require the furniture first to be placed at the corner, and then to be placed along the wall, and finally to be placed in the middle of the room, such as the desk in the study (see Fig. 4(b)). (3) Matrix mode is to require the furniture to be placed in a rectangular region, such as the desks in the classroom (see Fig. 4(c)). (4) Circular mode is a layout mode to require the furniture to be placed around the circular furniture, such as the circular dining table and chairs (see Fig. 3(d)). The combination of the four modes is called combination mode, such as the bedroom layout. The distance and angle between furniture in those layout models need to meet the ergonomic. The matrix mode and circular mode solution to be relatively easy, the distance and angle between the furniture are obtained according simply calculating. In the following section, we give the algorithms of coupled mode and enclosed mode.

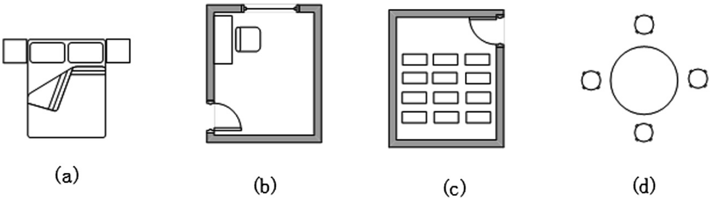


Fig. 4. Furniture layout mode. (a) Coupled mode. (b) Enclosed mode. (c) Matrix mode. (d) Circular mode.

6 Coupled Mode Algorithm

6.1 Mathematical Model

The mathematical model represents the layout relationship between the furniture in the coupled mode. Coupled mode consists of one chief furniture and n pieces of attached furniture. We use k to denote the number of the furniture, k = 1, 2… n. All the attached furniture is the same. The chief furniture contains n layout points, and each pieces of attached furniture has one layout point. As shown in Fig. 5, the chief furniture is the desk, and the attached furniture is the stool. We use the axis-aligned bounding box to represent a piece of furniture. Let *l, w* and *h* denote the length, width and height of the furniture, respectively. S = $(l/2, h/2, w/2)^{\mathrm{T}}$ is the length matrix, $\boldsymbol{S}_0$ is the chief furniture length matrix, and $\boldsymbol{S}_k$ is the attached furniture k length matrix. $\boldsymbol{P} = (x, y, z)^{\mathrm{T}}$ is the furniture position, $\boldsymbol{P}_0$ is the chief furniture position, and $\boldsymbol{P}_k$ is the attached furniture k position. a is the furniture rotation angle in the Y-axis direction, $a = \{0, 90, 180, 270\}$, a_0 is the chief furniture rotation angle, and a_k is the attached furniture k rotation angle. The main idea of coupled mode algorithm is to find the chief furniture and attached furniture layout points, and then to obtain the layout results through those layout points and matrix transformation. The position relationship between the attached furniture k and chief furniture can be expressed by the mathematical model as follows:

$$\begin{aligned} \boldsymbol{P}_k &= \boldsymbol{G}_k + \boldsymbol{T}_{\mathbf{k}} + \mathbf{D}_k, \\ \boldsymbol{G}_k &= \boldsymbol{M}_k \boldsymbol{S}_k, \\ \boldsymbol{T}_k &= \boldsymbol{P}_0 + \boldsymbol{R}_k \boldsymbol{S}_0. \end{aligned} \tag{1}$$

Where, $\boldsymbol{G} = (g_{ij})_{3\times1}$, $\boldsymbol{G}_k$ is the attached furniture k layout point; $\boldsymbol{T} = (t_{ij})_{3\times1}$; $\boldsymbol{T}_{\boldsymbol{k}}$ is the chief furniture layout point relative to the attached furniture k; $\boldsymbol{D} = (d_{ij})_{3\times1}$, $\boldsymbol{D}_k$ is

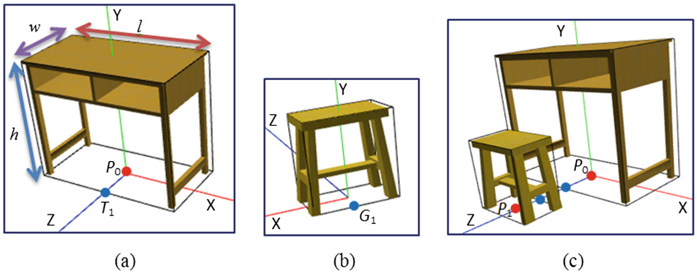


Fig. 5. An example of coupled mode. (a) The information of chief furniture. (b) The information of attached furniture. (c) The information of coupled mode.

the attached furniture k distance matrix; $\boldsymbol{M} = (m_{ij})_{3\times3}$, $\boldsymbol{M}_k$ is the attached furniture k transformation matrix; $\boldsymbol{R} = (r_{ij})_{3\times3}$, $\boldsymbol{R}_k$ is the chief furniture transformation matrix relative to the attached furniture k; $\boldsymbol{P}_0$, $\boldsymbol{S}_O$ and $\boldsymbol{S}_k$ are known. a_k, $\boldsymbol{M}_k$, $\boldsymbol{R}_k$ and $\boldsymbol{D}_k$ are set according to ergonomics and the layout [11].

As is shown in Fig. 5, the parameters of the mathematical model of the desk and stool are as follow:

$$a_0 = 0,\ a_1 = 0.$$

$$\mathbf{M}_{01} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{M}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{D}_1 = \begin{bmatrix} 0 \\ 0 \\ 20 \end{bmatrix}$$

6.2 Furniture-Furniture Coupled Mode Algorithm

Case-based reasoning has been widely used to solve various problems [12]. Now we use it to solve the coupled mode. The layout case consists of two parts: the layout problem description and the layout solution description. The layout problem is described by the vector $\mathbf{q} = (l_0, w_0, h_0, l_1, w_1, l_1)^{\mathrm{T}}$, where l_0, w_0, h_0, l_1, w_1, and l_1 denote the length, width, and height of the chief furniture, the length, width, and height of the attached furniture, respectively. The layout solution is the model parameters. The layout case name is composed of the chief furniture name, the attached furniture name and the total amount of attached furniture. Case similarity is defined as the Euclidean distance between the two layout cases. We use the nearest neighbor algorithm (K-Nearest Neighbor algorithm, K = 1) to retrieve cases.

$$\min_i \{\mathrm{Dis}(\mathrm{case}_0, \mathrm{case}_i) = (\sum_{j=1}^{6} (\mathbf{q}_{0,j} - \mathbf{q}_{i,\mathrm{j}})^2)^{1/2}\} \tag{2}$$

Where, case_0 is an await layout case, and case_i is a layout case whose name is the same as the case_0, i = {1, 2,…, m}. $\mathbf{q}_0$ is the problem description of case_0, and $\mathbf{q}_i$ is the problem description of case_i. The minimization returns the best case $\mathrm{case}_{\mathrm{best}}$ which is the most similar to case_0 in the case library. Substituting the solution of $\text{case}_{\text{best}}$ and three parameters $\boldsymbol{P}_0$, $\boldsymbol{S}_O$, $\boldsymbol{S}_k$ of the case_0 into the mathematical model and evaluate, we obtain the layout result. The algorithm is given below.

Algorithm 3. Furniture - furniture coupled mode Algorithm

Input: an await layout case_0.

Output: layout results of case_0.

Step 1: Find a set of case whose name is the same as the case_0 from the case library.

Step 2: Find the $\text{case}_{\text{best}}$ using the nearest neighbor algorithm by formula (2).

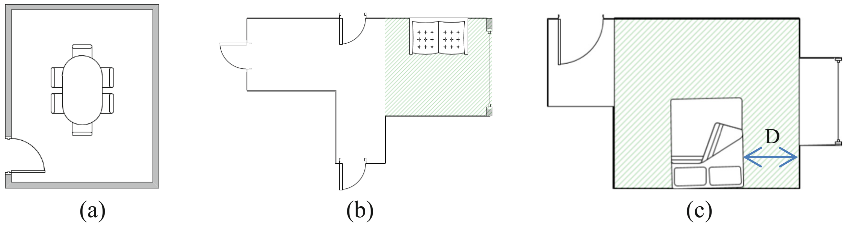
Step 3: Get the parameters form the mathematical model of $\text{case}_{\text{best}}$.

Step 4: Assign the parameters to the mathematical model of case_0 and calculate using formula (1).

Step 5: Obtain and return the layout result of case_0.

6.3 Furniture-Room Coupled Mode Algorithm

When there is a coupled layout mode between furniture and a room, we can take the room as a piece of furniture, which is converted into the furniture - furniture coupled mode. If the room shape is non-rectangular, we first need to extract its rectangular region using room shape standardization algorithm in Sect. 3, then define the coupled mode between the rectangular region and furniture. We summarize the common furniture layout rule of furniture - room coupled as follows. (1) The tables are placed in the middle of the conference room (see Fig. 6(a)). (2) The sofas should be placed along the windows of the living room (see Fig. 6(b)). (3) The bed is placed along the windows of the bedroom (see Fig. 6(c)), the distance D is decided by the room size.



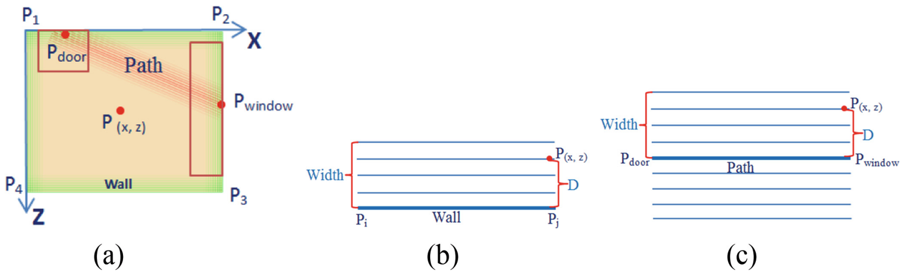
(a) (b) (c)

Fig. 6. Furniture - room coupled mode. (a) Meeting room. (b) Non-rectangular living room. (3) Non-rectangular bedroom.

7 Enclosed Mode Algorithms

7.1 Distance Field

In order to solve the enclosed mode layout problem, we use the distance field to fit the bedroom and living room furniture layout rules. Before constructing the distance field, we should make room's floor discretization. We use 1 unit to represent 1 cm in real world. For example, the length of a room is 4 m and width is 3 m, then the size of the



(a) (b) (c)

Fig. 7. (a) The distance field of a bedroom. (2) The distance field of a wall. (3) The distance field of a path.

room's floor is expressed as 400 * 300 after discrete processing. The room distance field consists of two parts: walls distance field and paths distance field. The line segment p_{window} p_{door} is called a path. For example, a bedroom distance field is shown in Fig. 7(a).

A wall distance field is an attractive energy to the furniture. We define the wall maximum energy intensity of t (t > 0). If the shortest distance from the point P (x, z) to the wall p_ip_j is D, the point P (x, y) energy value generated by the wall p_ip_j is calculated as follows.

$$V\big(P(x,z),\ Wall\big(p_ip_j\big)\big) = \begin{cases} \left(\frac{D}{Width}\right) * t, & P(x,z)\ inside\ the\ room\ and\ D < Width \\ 0, & P(x,z)\ outside\ the\ room\ or\ D\ >\ Width \end{cases} \tag{3}$$

A path distance field is a repulsive energy to the furniture. We define the path maximum energy intensity of s (s < 0). If the shortest distance from the point P (x, z) to the path $p_{door}p_{window}$ is D, the point P (x, z) energy value generated by the path $p_{door}p_{window}$ is calculated as follows.

$$V(P(x,z),\ Path(i)) = \begin{cases} \left(\frac{D}{Width}\right) * s, & P(x,z)\ inside\ the\ room\ and\ D < Width \\ 0, & P(x,z)\ outside\ the\ room\ or\ D\ >\ Width \end{cases} \tag{4}$$

When the room shape, the doors and the windows position are determined, using the formulas 3 and 4, we can calculate any point energy value in the room as follows.

$$V(P(x,z)) = \sum\nolimits_{i=1}^{n} V(P(x,z), Wall_i) + \sum\nolimits_{j=1}^{m} V\big(P(x,z), Path_j\big) \tag{5}$$

Where, n and m is the number of walls and paths, respectively.

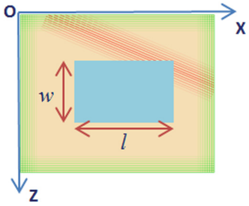


Fig. 8. Furniture energy

7.2 Furniture Energy and Furniture Energy Maximization

The furniture energy value is the sum of the energy values of the points occupied by the furniture. As is shown in Fig. 8, the rectangle is the bottom of the furniture bounding box, and the furniture energy is calculated as follows.

$$E(P(x,z),S(l,w)) = \sum\nolimits_{i=x-l/2}^{x+l/2} \sum\nolimits_{j=z-w/2}^{z+w/2} V(P(x,z)) \tag{6}$$

Where, point p(x, z) is the furniture position, l and w denote the length and width of the furniture, respectively.

In the formula (6), when the following two cases occur, the furniture energy value is zero: (1) There is a collision between the furniture with the placed furniture or the door rectangular box (see Fig. 7(a)). (2) There is a collision between the furniture with the window rectangular box and the height of the furniture is greater than the height of the window (see Fig. 7(a)).

Furniture energy maximization is to search the position in the room where the furniture energy value is the largest, which is represented as follows.

$$\max_{x,z} E(P(x,z),S(l,w)) \tag{7}$$

Where $E(P(x,z),S(l,w))$ is the energy value of the furniture in position point(x, z), and the furniture can be rotated 90°. To improve the computational efficiency, the energy value of the furniture in current position can be deduced according the energy value of the furniture in previous position [10]. Compared with the formula of distance field in [10] to be suitable for the rectangular room, this paper's formula of distance

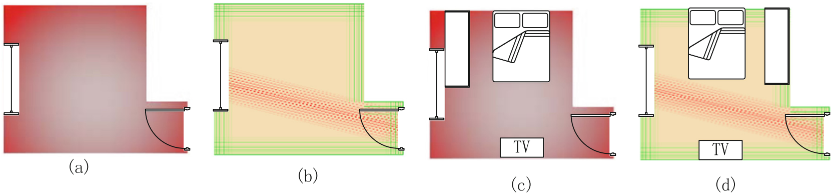


Fig. 9. (a) The energy distribution using the formula of distance field in [10]. (b) The energy distribution using the formula (5). (c) The layout result using the layout method in [10]. (d) The layout result using our layout method.

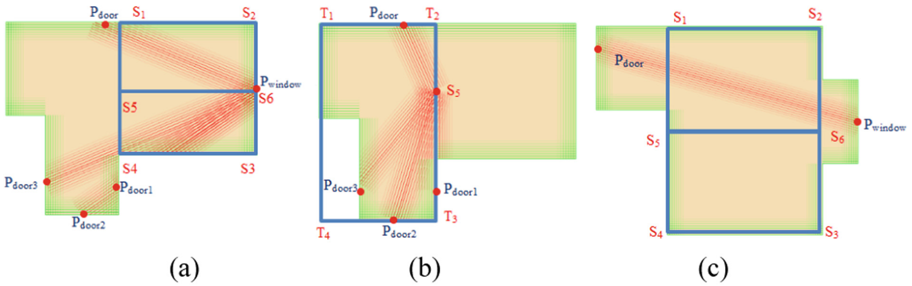


Fig. 10. (a) The energy distribution of finding the layout region of sofa and TV in a living room. (b) The energy distribution of finding the layout position of the dining-table in a living room. (c) The energy distribution of a bedroom.

field is more consistent with non-rectangular room. For example, Fig. 9(a) show the distance field of a bedroom using the formula of distance field in [10], and Fig. 9(b) show the distance field of the same bedroom using formula 5. Figure 9(c) show the layout result using the layout method in [10], and Fig. 9(d) show the layout result using our layout method. Our improved distance field places the furniture better.

8 Living Room and Bedroom Furniture Layout Algorithm

We now give detailed algorithmic descriptions to real cases using our furniture models and the corresponding reasoning methods. As the length requirement of the paper, we propose three cases, the living room layout, the bedroom layout, and the classroom.

8.1 Living Room Furniture Layout Algorithm

In a living room, we need arrange furniture include sofa, television, dining-table, and so on. The region of arranging television and sofa is called the television region and the sofa region, respectively. When the sofa and TV layout rectangular region is found using the living room shape standardization algorithm, we also need to extract the television region and sofa region from it, which is called television region and sofa region algorithm. First, we divide the rectangle $s_1s_2s_3s_4$ into two small rectangles: $s_1s_2s_5s_6$ and $s_3s_4s_5s_6$. Then, we calculate the energy values of the two small rectangles using formula 4 (see Fig. 10(a)). Finally, the small rectangle of the more energy value, that is $s_1s_2s_6s_5$, is placed the sofa, and the other small rectangle $s_3s_4s_5s_6$ is placed the television. The algorithm is summarized below.

Algorithm 4. Living room furniture layout Algorithm

Input: the living room shape and await furniture.

Output: layout results of furniture in the living room.

Step 1: Extract the sofa and TV layout rectangular region using the living room shape standardization algorithm in Sect. 3 and calculate the distance field of the living room using formula 3 and formula 4.

Step 2: Arrange the furniture which is of the coupled mode using Furniture - furniture coupled mode Algorithm in Sect. 5.

Step 3: Arrange the TV combination furniture and the sofa combination furniture using the Furniture - room coupled mode Algorithm in Sect. 5 and the television region and sofa region algorithm in Sect. 7.

Step 4: Arrange dining-table and make its energy maximization such that the furniture within the region $T_1T_2T_3T_4$ and the ling room (see Fig. 10(b)), and return the layout results.

8.2 Bedroom Furniture Layout Algorithm

In a bedroom room, we need to arrange furniture include bed, television, wardrobe, and so on. When the rectangular layout region is found using the bed room shape standardization algorithm, we also need to extract the television region and bed region. The extract method of the television region and bed region in the bedroom is similar with the extract method in living room (see Fig. 10(c)). Bed room furniture layout Algorithm is summarized below.

Algorithm 5. Bed room furniture layout Algorithm

Input: bed room shape and await furniture.

Output: layout results of furniture in the bed room.

Step 1: Extract the bed and TV layout rectangular region using the bed room shape standardization algorithm in Sect. 3 and calculate the distance field of the living room using formula 3 and formula 4.

Step 2: Arrange the furniture which is of the coupled mode using Furniture - furniture coupled mode Algorithm in Sect. 5.

Step 3: Arrange the bed combination furniture and the television combination furniture using the Furniture - room coupled mode Algorithm in Sect. 5 and the television region and bed region algorithm in Sect. 7.

Step 4: Sort the others furniture of awaiting layout by height and area.

Step 5: Arrange the others furniture one by one and make its energy maximization within the region $s_1s_2s_3s_4$ (see Fig. 10(c)), and return the layout results.

8.3 Classroom Furniture Layout Algorithm

In a classroom, we need to arrange furniture include a teacher's desk and the student's desks and stools. The classroom furniture layout algorithm is summarized below.

Algorithm 6. Classroom furniture layout Algorithm

Input: a rectangular classroom, a teacher's desk and student's desks and stools.

Output: layout results of furniture in the classroom.

Step 1: Arrange the student's desks and stools using Furniture - furniture coupled mode Algorithm in Sect. 5.

Step 2: Arrange the student's desks and stools according the matrix mode.

Step 3: Arrange the teacher's desk and the combination furniture of the student's desk and stool using the Furniture - room coupled mode Algorithm in Sect. 5.

Step 4: Return the layout results.

9 Experimental Results

We developed an online furniture layout platform using WebGL and C sharp language in Visual Studio 2015. We designed 16 common cases of furniture layout to store in database of SQL Server 2012. The computation was performed on a computer with a processor of Dual-core 2.6 GHz, main memory 4G, operation system Microsoft Windows 8 64bit and Chrome browser.

9.1 Comparisons with Prior Art

To further evaluate our method, we compare with two most recent methods, [2, 5] respectively. As there are no detailed parameters on the scene of living room and furniture sizes in [5], we manually extracted these information from their paper and used our method on their scenes and compare the running time with their method. Figure 11 and Table 1 shows the visual results and the detailed running time. In both case, our method generates comparative results to theirs while the running time is significantly faster (a factor of 30x).

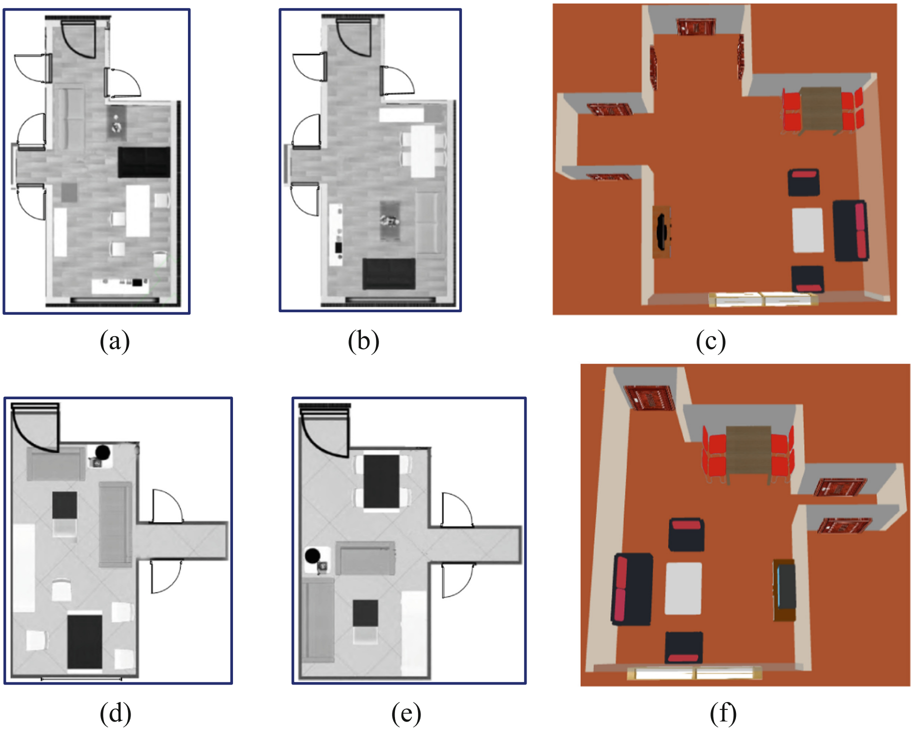


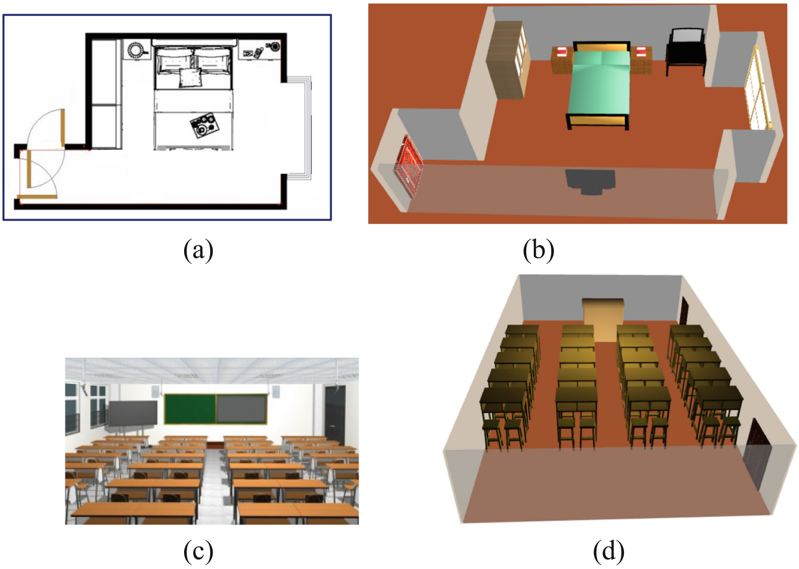
Fig. 11. (a) The layout result using the method in [2]. (b) The layout result using the method in [5]. (c) The layout result using our method. (d) The layout result using the method in [2]. (e) The layout result using the method in [5]. (f) The layout result using our method.

Table 1. Statistics of running time comparisons with [2, 5]

Method	Computer configuration	Programming tools	Average running time (s)
Method in [2]	RAM 4 GB, and 3.4 GHz CPU Intel Core i7-2600	Matlab and VS2008	217
Method in [5]	RAM 4 GB, and 3.4 GHz CPU Intel Core i7-2600	Matlab and VS2008	50.3
Our method	RAM 4 GB, and 2.6 GHz CPU Intel Core i5-3230	WebGL and VS2015	1.5

9.2 Comparison with Real Designs

We also invite an interior designer to manually design some furniture arrangements using our online system but without any algorithmic inference. The designer is allowed to manually drag an object and move it along the floor plane to place it properly given a room layout. It took almost an hour for him to finish the classroom and bedroom design whereas our system took 2.3 s. Figure 12 shows the visual results. Our method generates comparable arrangements with the designer.


(a) (b)

(c) (d)

Fig. 12. (a) The layout result of the bedroom by designer. (b) The layout result of the bedroom using our method. (c) The layout result of the classroom by designer. (d) The layout result of the classroom using our method.

10 Conclusion and Future Work

In this paper, a learning platform of furniture layout has been developed using the furniture layout algorithms based on case-based reasoning and distance field. It can automatically generate a layout to help the users to learn knowledge of furniture layout and obtain the desire layout. Compared with the traditional layout algorithms using a single method to all the different scenes, our algorithms use different layout mode to solve the different scenes layout problem. First the non-rectangular room shape normalization algorithms are presented according the room shape features. Then, four layout modes, namely coupled mode, enclosed mode, matrix mode and circular mode, are introduced by analyzing the distribution in a room. Next, we propose the algorithm of coupled mode based on the mathematical model and case-based reasoning. We also present the algorithm of enclosed mode based on the distance field. Finally we give the furniture layout algorithm of non-rectangular bedroom and that of non-rectangular living room. The experimental results show that the layout results of our method are similar to the real designs and the running time can meet real-time response requirements of online layout. In future, we plan to explore automatic furniture layout for large scale and mixed scenes. How to automatically generate the layout for mixed places such as the office and shopping malls scenes using our layout modes is worth further exploration.

Acknowledgments. The authors appreciate the comments and suggestions of all anonymous reviewers, whose comments significantly improved this paper. This work is supported by The Key Research Projects of Central University of Basic Scientific Research Funds for Cross Cooperation (201510-02), Research Fund for the Doctoral Program of Higher Education of China (No. 2013007211-0035) and Key project in scientific and technological of Jilin Province in China (No. 20140204088GX).

References

1. Yu, L.F., Yeung, S.K., Tang, C.K., et al.: Make it home: automatic optimization of furniture arrangement. ACM Trans. Graph. **30**(4) (2011). Article No. 86
2. Merrell, P., Schkufza, E., Li, Z.Y., et al.: Interactive furniture layout using interior design guidelines. ACM Trans. Graph. **30**(4) (2011). Article No. 87
3. Chen, G., Li, G., Liu, P., et al.: Hierarchical constraints with particle swarm optimization for furniture arrangement. J. Comput. Aided Des. Comput. Graph. **26**(10), 1603–1612 (2014)
4. Akase, R., Okada, Y.: Automatic 3D furniture layout based on interactive evolutionary computation. In: Proceedings of the 2013 Seventh International Conference on Complex, Intelligent, and Sofaware Intensive Systems, pp. 726–731. IEEE Computer Society, USA (2013)
5. Liu, M.-L., Jiang, H., Mao, T.-L., Wang, Z.-Q.: Composite model for home furnishing generation. Chin. J. Comput. **39**(2) (2016)
6. Akazawa, Y., Okada, Y., Niijima, K.: Automatic 3D scene generation based on contact constraints. In: Eighth International Conference on Computer Graphics and Artificial Intelligence, pp. 593–598 (2002)

7. Germer, T., Schwarz, M.: Procedural arrangement of furniture for real-time walkthroughs. Comput. Graph. Forum **28**(8), 2068–2078 (2009)
8. Kjølaas, K.A.H.: Automatic furniture population of large architectural models. Massachusetts Institute of Technology, Cambridge (2000)
9. Fisher, M., Ritchie, D., Savva, M., et al.: Example-based synthesis of 3D object arrangements. ACM Trans. Graph. **31**(6), 439–445 (2012)
10. Song, P.-H., Jia, J.-Y.: Online furniture layout method based on case-based reasoning and distance fields. J. Syst. Simul. (10), 2438–2447 (2016)
11. Zhang, Q.-M., Zheng, S.-Y.: Interior Design Data Set. China Construction Industry Press, Beijing (1991)
12. Kolodner, J.L.: An introduction to case-based reasoning. Artif. Intell. Rev. **6**(1), 3–34 (1992)