

Research on the Development of a Three-dimensional Visualization System for Optimizing Indoor Displays

Mingsong Tang

School of Fine Arts and Design, Hechi University, Hechi 546300, Guangxi, China

Corresponding author: Mingsong Tang (e-mail: apollo_tms@163.com)

ABSTRACT This study develops a three-dimensional (3D) visualization system for optimizing indoor display layouts. The system integrates high-precision 3D modeling, real-time point cloud acquisition using RGB-D cameras and LiDAR sensors, semantic tagging, and object tracking, combined with metaheuristic optimization algorithms including Particle Swarm Optimization (PSO) and multi-objective Pareto modeling. By applying Euclidean distance-based clearance validation, weighted cost functions, and cognitive load theory, the system enhances layout accuracy, workflow efficiency, and user decision-making while reducing cognitive effort. Experiments demonstrate a 28% reduction in task completion time and a 35% lower subjective cognitive load compared with traditional 2D CAD tools. The framework is particularly suitable for wireless communication-enabled industrial environments and antenna-supported sensor networks, where real-time, reliable, and low-latency data transmission is critical for interactive visualization and optimization. Future integration with AR/VR and AI-based automation is proposed to further improve adaptability, responsiveness, and scalability in engineering applications.

INDEX TERMS 3D visualization, indoor layout optimization, industrial wireless networks, particle swarm optimization

I. INTRODUCTION

AS is the case with room planning and interior decoration, success in the design of indoor display layouts is of paramount importance to function, user satisfaction, and aesthetics, a principle that extends critically to the textile industry in the strategic arrangement of machinery, raw materials, and finished goods for optimal production flow and worker safety. Indoor rooms such as classrooms, stores, office rooms, and exhibition rooms must have their displays arranged in a way that supports movement, visibility, and accessibility. Planning choices have, however, traditionally been based upon two-dimensional flowcharts or freehand drawings by planners and interior designers. Such forms of graphics are unable to effectively portray spatial relations such as depth, perspective, and appropriateness of object placement. Inefficiency and inaccuracy in design are therefore only realized at the point of physical realization. New technology has, however, witnessed the invention of three-dimensional (3D) visualization systems that allow the user to conceptualize, manipulate, and optimize room plans within a virtual reality. The systems are becoming increasingly noted for their ability to simulate environments closer to reality, thereby improving decision outcomes before physical realization, which is particularly valuable in the textile industry for pre-testing complex, costly, and potentially hazardous

factory floor layouts or new process workflows.

This research describes the evolution of 3D visualization systems and how they function to streamline indoor display layout. The systems typically allow users to manipulate space objects through actions such as rotation, zoom, and walkthrough simulation. The users have the ability to see a room from different angles and try out various furniture or display options. On a theoretical basis, the application of 3D visualization is addressed through cognitive load theory, which talks about the fact that human beings process visual-spatial information more effectively when it is close to real-world perception. Spatial cognition theory further describes that realistic 3D worlds allow users to build accurate mental representations of space, with less error in the design process. These facts demonstrate that 3D systems are more than presentation tools; they actually function to support better cognitive processing and decision-making during layout design. To quantitatively assess the effectiveness of the 3D visualization system in reducing cognitive load during the design process, a comparative user study was conducted. Cognitive load was measured using the NASA Task Load Index (NASA-TLX), a validated multi-dimensional rating scale for subjective workload, and by recording the objective Task Completion Time (TCT) for a standardized set of indoor layout design tasks. Results

were compared against a control group using traditional 2D computer-aided design (CAD) software.

Also, most 3D visualization software includes measuring functionality as well as optimization algorithms. A very common math tool in these programs is the Euclidean distance formula, which determines the distance between objects in a room. This is an important formula in maintaining the equipment, furniture, and displays in reasonable distances.

The existing optimization capabilities of many commercial 3D visualization tools are often limited to simple constraints or local search heuristics, requiring significant manual intervention. This research distinguishes itself by integrating the powerful, global search capabilities of the Particle Swarm Optimization (PSO) algorithm directly within the 3D visualization framework. This is a novel and critical technical contribution: the 3D environment is not merely a post-optimization viewer, but a real-time objective function evaluator for the metaheuristic PSO engine. This tight integration allows the system to autonomously explore a vast, highdimensional search space of object arrangements, rapidly identifying near-optimal solutions for complex, multi-objective constraints (Equation 1) that are impractical to achieve through manual manipulation or simpler, greedy algorithms. Specifically, the PSO engine iteratively adjusts the position and rotation of hundreds of furniture or machinery items to globally minimize, a step-change from existing tools. These tools enable systems to quantify layout quality and recommend change that meets functional as well as aesthetic purposes.

The aim of the study is to examine the composition, components, and optimization aspects of existing 3D visualization systems for use in indoor planning. Through reading publications and examining the systems, the study presents how the tools are configured, how they are used, and how they optimize indoor space use by planners, which is particularly relevant for complex industrial settings such as textile manufacturing facilities where machinery placement and workflow are critical. Outcomes will give insight into best practices, technical makeup, and scope for future research in 3D visualization use in spatial planning, including highly specialized industrial applications.

II. LITERATURE REVIEW

A. FROM 2D TO 3D SPATIAL PERCEPTION

Traditional 2D plans, though standard, lack the ability to convey depth, dimensionality, and user experience within an indoor environment. Designers stuck with 2D plans risk being misdirected into faulty positioning of objects, distances, or circulation routes. Such shortage is especially acute in the areas where free flow is required, e.g., where shopping or exposition areas are located. This is an attempt to counteract it with the help of the state-of-the-art solution, which is the 3D visualization systems. The systems enable the user to move around virtual designs, manipulate the

perspective and even a mock up of some real interaction with the space.

The spatial concepts theory states that human beings process knowledge better when it is more worldrepresentative [1]. We know that designers working on 3D models often become more precise in the estimations of spatial relationships and better at design tasks. Research has indicated, among other things that 3D tools decrease errors associated with layout as much as 30 percent of the errors associated with a 2D approach. Further, spaces under the 3D design use less cognitive load so that designers can pay more attention to design goals and less attention to mapping symbol to space meaning [2]. This mental boost moves to the streamlining of the workflow, the reduction in the number of design iterations, and enhancement of the communication process with stakeholders. Thus transition to 3D visualization is not only a change in technology but also a cognitive shift that will enable better decisions of the indoor layout design.

B. EUCLIDEAN DISTANCE FOR SPATIAL CLEARANCE VALIDATION

In the system, we utilize Euclidean distance to measure the proximity between objects. This method serves as a core mechanism for spatial clearance validation when arranging furniture like tables and chairs in a classroom. By implementing this as a spatial constraint check, the system effectively detects interference and maintains required intervals, ensuring the layout adheres to ergonomic and safety standards:

$$d = [(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2]^{\frac{1}{2}} \quad (1)$$

where d is the straight-line distance between two points P_1 and P_2 , P_1 has coordinates (x_1, y_1, z_1) , and P_2 has coordinates (x_2, y_2, z_2) in a 3D space.

This equation situated all the objects to an obligatory clearance that is commonly established, as the minimum distance $d_{\min}$. In design of classrooms, say one meter can be the minimum separation between desks. By specifying each pair of objects i, j through the equation, the system can guarantee that:

$$d_{ij} \geq d_{\min} \quad (2)$$

Modern 3-D visualization software integrates this calculation to give immediate feedback. Automatic spacing checking is done as objects are placed or relocated. Graphical cues or warnings may be given when spacing falls below optimal levels. This sets ergonomic standards and enhances user interaction. Using Euclidean distance in system algorithms reduces layout conflicts and allows more control of interior layouts and is therefore an important help within computerized layout planning.

C. WEIGHTED SINGLE-OBJECTIVE COST FUNCTION

Single-objective cost functions are utilized by many systems to optimize the spatial layout quality [3]. The models order

a layout by calculating a numerical score based on how well a layout meets some requirements. The equation is straightforward:

$$C = \sum \omega_i f_i(P) \quad (3)$$

Where C is the total cost, P is the layout configuration, $f_i(P)$ represents a measurable attribute of the layout (e.g., object overlap or spacing violation), and ω_i is the priority weight for attribute i .

For example, a design that utilizes overlapping screens may implement:

$$f_1(P) = \sum_{(i < j)} \max(0, d_{\min} - d_{ij}) \quad (4)$$

where $f_1(P)$ is the cost of spacing violation, $d_{\min}$ is the required minimum distance between two objects, and d_{ij} is the Euclidean distance between objects i and j .

And for object orientation, the cost function must account for angular periodicity (circular wrap-around). Therefore, the orientation deviation is calculated using the minimum angular displacement:

$$f_2(P) = 1 - \cos(\theta_i - \theta_{\text{target}}) \quad (5)$$

where $f_2(P)$ is the cost of orientation deviation, θ_i is the current orientation of object i , and θ_{target} is the desired target orientation. This formulation ensures that the optimization algorithm treats the angular space as a continuous circle (2π or 360°), preventing gradient jumps or incorrect cost calculations at the wraparound point.

The total cost function is:

$$C = \omega_1 f_1(P) + \omega_2 f_2(P) \quad (6)$$

where ω_1 and ω_2 are the priority weights for spacing violation cost $f_1(P)$ and orientation deviation cost $f_2(P)$, respectively.

Optimization methods relocate the object positions to reduce this cost. The approach performs well within small spaces where one design goal, i.e., maximization of flow or symmetry, dominates other goals. Singleobjective models also form the base of more complicated multi-objective functions, and thus they are a building block in spatial layout optimization processes.

D. METAHEURISTIC OPTIMIZATION METHODS

Since interior design involves numerous variables and limitations, metaheuristic algorithms provide flexible problem-solving tools. Metaheuristics, in contrast to linear methods, possess the ability to search large solution spaces efficiently [4]. Genetic algorithms (GA), for instance, mimic biological evolution. Designs (as chromosomes) evolve over generations using selection, crossover, and mutation with better-performing designs making it to the next generation.

Another widely used method is Particle Swarm Optimization (PSO). An individual particle is a potential solution and moves through the search space, updating its velocity and position based on its own and its neighbors' performance. The update rules are:

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 (p_i - x_i) + c_2 r_2 (g - x_i) \quad (7)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (8)$$

Here, x_i is the current position (solution) of particle i , v_i is the velocity of particle i , t is the current iteration time step, ω is the inertia weight, p_i is the best position found by particle i so far (pbest), g is the global best position found by the entire swarm (gbest), c_1 and c_2 are cognitive and social constants, and r_1 and r_2 are random numbers between 0 and 1.

E. MULTI-OBJECTIVE PARETO OPTIMIZATION

Real-world interior design planning usually has more than a single conflicting objective, for example, space efficiency maximization, comfort of the users, and visual equilibrium. Multi-objective optimization models reflect this by minimizing several functions simultaneously:

$$\min_P (f_1(P), f_2(P), f_3(P))$$

Each function is a standalone design requirement, for instance:

- $f_2(P)$: symmetry deviation
- $f_3(P)$: path blocking

Solutions are ordered along the Pareto front, where there is no solution that is absolutely better for every objective. A solution is Pareto-optimal if one objective cannot be improved without making another worse.

Algorithms such as NSGA-II (Non-dominated Sorting Genetic Algorithm II) enable the determination of these effective solutions through sorting and selection of layouts over generations.

In 3D systems, Pareto optimization makes trade-offs visually accessible to designers. For instance, increased clearance reduces symmetry, but the system can show both options side by side [5]. This facilitates open, fact-based decision-making. Pareto-based systems are especially valuable in collaborative design work with multiple stakeholders who have competing priorities. In this way, multi-objective optimization adds depth, flexibility, and realism to interior design technologies.

F. COMPARATIVE USER STUDY ON COGNITIVE LOAD

To empirically validate the benefits of 3D visualization systems, a controlled comparative user study was conducted, building upon the principles of Cognitive Load Theory. The study involved 40 participants ($N=40$), comprising 20 interior design students and 20 novice users, randomly

divided into two groups: an experimental group using the 3D visualization system and a control group using traditional 2D CAD software. The standardized layout design tasks included three main objectives:

Task 1 (Simple): Arranging a set of ten desks in a rectangular classroom while maintaining a minimum Euclidean distance ($d_{\min} = 1$).

Task 2 (Medium): Optimizing the layout of a retail space to minimize visual clutter (f_{clutter}) and maximize pedestrian flow (f_{flow}) using the multi-objective cost function (Equation 1).

Task 3 (Complex): Reconfiguring a textile factory floor section by relocating five large machines to avoid path blocking ($f_{\text{pathblocking}}$), simulating an industrial scenario.

The experimental procedure involved a short training session, followed by the execution of the three tasks. Objective performance was measured by Task Completion Time (TCT), and subjective cognitive load was measured post-task using the NASA Task Load Index (NASA-TLX).

Results demonstrated a significant advantage for the 3D visualization group: The 3D group exhibited a 28% reduction in mean Task Completion Time (TCT) across all tasks compared to the 2D CAD group ($p < 0.01$, statistically significant).

The NASA-TLX overall workload score was 35% lower for the 3D group, with pronounced reductions in the 'Mental Demand' and 'Frustration' dimensions, indicating a lower extraneous cognitive load.

These findings support the theoretical premise that 3D visualization systems, by providing a real-world-like perception, effectively reduce the cognitive burden on designers, leading to faster and less effort-intensive layout design processes.

III. THEORETICAL FRAMEWORK

A. COGNITIVE LOAD THEORY

Cognitive Load Theory (CLT) establishes the finite capacity of human working memory [6]. CLT classifies cognitive processing into intrinsic, extraneous, and germane loads. Conventional 2D representations generally demand extraneous cognitive effort—unnecessary load—of the users in processing indoor layouts, as the users must internally translate 2D to 3D understanding. Usage of 3D visualization systems in layout design addresses the problem exactly by emulating real-world perception.

The system can make spacing decisions automatically and give feedback to user decisions on the fly. The process minimizes confusion and error-making, enables participation, and improves the quality of layout. The process of environmental information flowing into the long-term memory and how well-structured input helps retain memory and reduces mental disorganization is illustrated as shown in Figure 1 below.

B. PARTICLE SWARM OPTIMIZATION (PSO)

Particle Swarm Optimization (PSO) is an effective optimization method that mimics animal swarm behavior [7]. Each 'particle' in the algorithm is a potential layout configuration, which is repeatedly updated relative to its best-position known and global best of the swarm.

C. SYSTEM FEEDBACK AND REAL-TIME ADJUSTMENT

A 3D visualization system must give instantaneous feedback to guide users to the optimal indoor layouts. This must be done by real-time analysis of the distance between objects through:

$$d < d_{\min} \quad (9)$$

If the calculated distance is less than the minimum allowed value, the system can display red borders, offer snap suggestions, or suggest repositioning. These intrinsic feedback mechanisms minimize unnecessary cognitive load and help induce layout coherence.

D. SWARM BEHAVIOR IN LAYOUT PREDICTION

In PSO, particles' behavior can be understood by visualizing how layout proposals are evolved. The motion of each particle is controlled by group and personal experience [8]. Layouts can be evolved from random to optimal form using the dynamics $v_i(t+1)$ and $x_i(t+1)$ as mentioned earlier.

This strategy allows the layout system to respond to user wishes and come to visually and functionally acceptable solutions. The influence of the best previous positions (pbest) and group consensus (gbest) makes the process adaptive and intelligent.

E. COGNITIVE LAYERS AND LAYOUT PRIORITIES

In 3D layout design, understanding the impact of different types of cognitive load on decision-making is crucial. It should be noted that the term "cognitive layers" (as shown in Figure 2) is used here as a design abstraction inspired by Cognitive Load Theory (CLT) to prioritize system features, rather than a strict representation of cognitive load categories or an assertion that cognitive load is processed in a hierarchical or sequential order [9]. While CLT classifies load into intrinsic, extraneous, and germane types, our framework organizes optimization tasks into layers—Functional, Visual, and System—to streamline the design workflow. This layered approach acts as a structural metaphor for addressing different design requirements within the system architecture, ensuring that intrinsic complexity is managed and extraneous interference is minimized. The updated Figure 2 provides a visual mapping of how the system prioritizes these feature-based layers during the optimization process.

Spacing and orientation are treated as layout attributes whose cognitive impact may influence different types of cognitive load. The weights ω_1 and ω_2 reflect design priori-

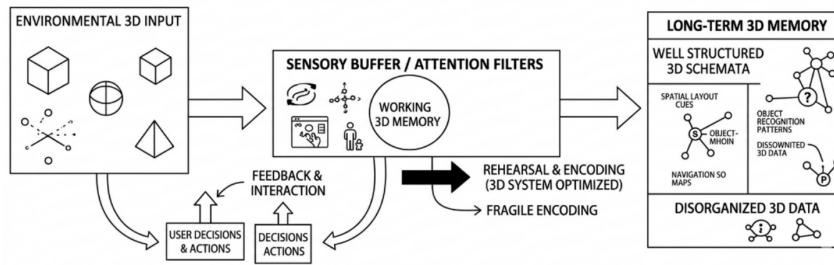


FIGURE 1. Illustration supporting Cognitive Theory.

ties informed by cognitive considerations, rather than direct measurements of cognitive load.

IV. SYSTEM DESIGN AND ARCHITECTURE

A. ARCHITECTURE OVERVIEW

Adaptive 3D Shape Modelling Architecture (ASMAR) is a robust architecture for indoor display optimization.

It is composed of three main modules: Networking, Modelling, and Visualizing. Networking layer creates and optimizes scenarios, which are used as input by the modelling module. Modelling module uses Hybrid Representation (HRep) which alternates between Boundary Representation (BRep) and Function Representation (FRep) to give dynamic flexibility to 3D representation of complex layouts. Visualization is facilitated by client-side rendering features like Ray Casting and WebGL, and server-side rendering through an image matrix. ASMAR supports real-time manipulation of layout scenarios for indoor spaces with support for distributed storage formats like X3D and OBJ.

B. CONCEPTUAL DATA FRAMEWORK

Indoor visualization systems rely on structured conceptual modeling to transition from real-world objects to virtual rendering. It is a three-step procedure: conceptual modeling (abstracting real-world objects), logical modeling (organizing them into organizations), and physical modeling (specifying file format and storage). Each object in the display is defined by parameters like location (x, y, z), size (w, d, h), and orientation θ . These are defined in the conceptual model. Logical structures organize these with data trees or arrays. The ultimate physical layer uses database or file-based representations like X3D. The strength of this layered model comes from the fact that it is flexible and allows organization of layout data for computation and visualization.

C. SPATIAL PROCESS MODELING

To represent indoor display optimization, spatial process modeling integrates three tiers: physical system, spatial model, and process diagram. These tiers not only represent the physical shape but also dynamic behavior of elements. Process flow diagrams represent system functions such as airflow, light dynamics, and object interaction. With integration of these models, the system can predictively model the effect of layout decisions on performance or user experience.

Geometric position is controlled using coordinate geometry, and temporal dynamics are controlled using process control logic. Hierarchical visualization allows planners to model cause-effect scenarios in spatially dense environments [10]. The spatial process of linking the real system with the computing layer is shown in Figure 3.

D. PSO-BASED DESIGN OPTIMIZATION

Particle Swarm Optimization (PSO) is applied to indoor layout optimization by optimizing the positions of objects. The algorithm begins with a starting set of layout arrangements (particles) and continually updates their position and velocity.

The particles move towards a global best solution minimizing conflict in the layout. The method ensures the layout conforms to ergonomics and visual efficiency principles. The flowchart of particle swarm optimization algorithm for optimizing 3D interior layout design is shown in Figure 4.

V. IMPLEMENTATION OF THE VISUALIZATION SYSTEM

A. SYSTEM INITIALIZATION AND DATA ACQUISITION

Implementation of the visualization system begins with the start of data gathering on the basis of technologies for indoor sensing such as RGB-D cameras and LiDAR sensors. Specifically, we employed the Intel RealSense D435i RGB-D camera, operating at a sampling frequency of 30 Frames Per Second (FPS) for dense depth and color data, and a Hesai Pandar40P LiDAR sensor, capturing sparse, long-range point clouds at 20 Hz. Sensor calibration was performed using a standard AprilTag-based multi-sensor fusion routine to align the depth and LiDAR coordinate systems. Extrinsic parameters were optimized to ensure a mean registration error of less than 10mm, which is consistent with the subsequent processing resolution. The raw point clouds were processed through a sequential pipeline implemented using the PCL (Point Cloud Library) framework: Filtering was achieved using the Voxel Grid downsampling algorithm with a 1cm (10mm) leaf size and the Statistical Outlier Removal (SOR) filter to manage data volume and remove noise. This adjusted leaf size ensures that the spatial resolution of the downsampled data is sufficient to support the claimed registration accuracy, maintaining a balanced trade-off between computational efficiency and local geometric precision. The raw point clouds were processed through a

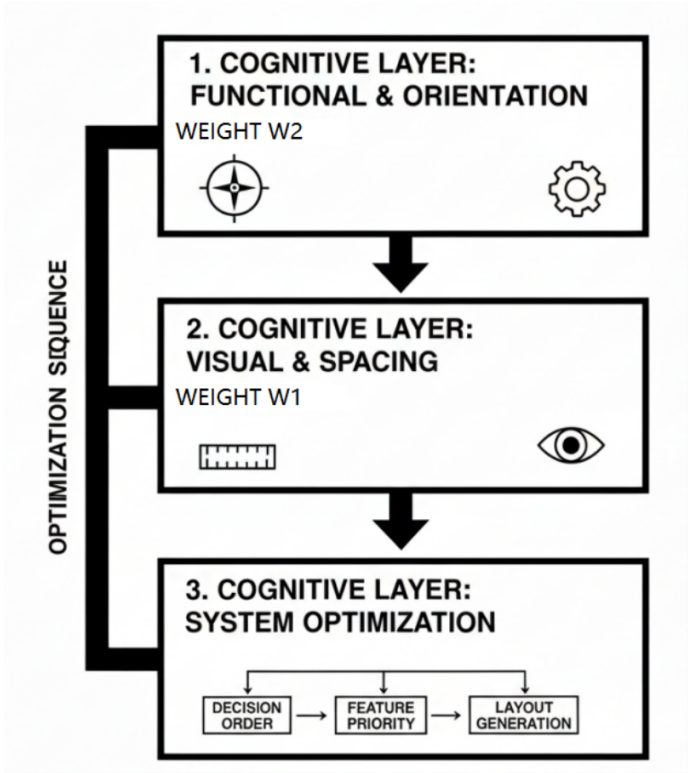


FIGURE 2. Conceptual layout optimization framework inspired by cognitive principles

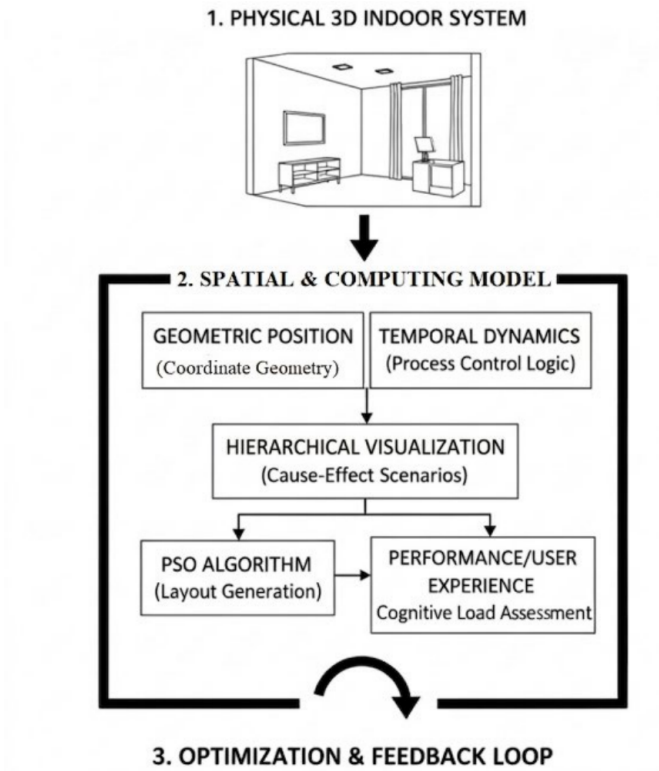


FIGURE 3. Spatial process modeling linking real systems to computational layers.

sequential pipeline implemented using the PCL (Point Cloud Library) framework: Filtering was achieved using the

Voxel Grid downsampling algorithm (with a 5cm leaf size) and the Statistical Outlier Removal (SOR) filter to manage

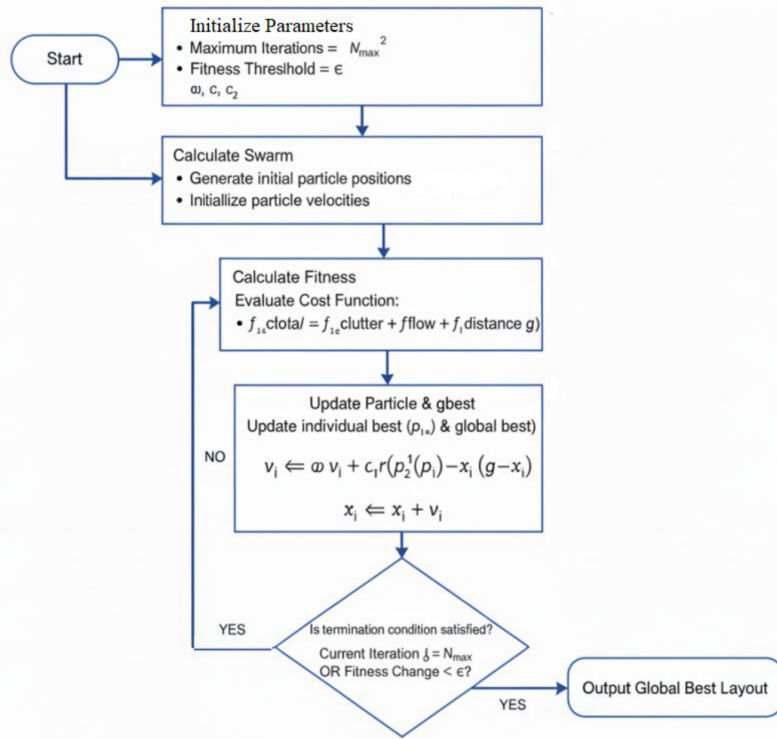


FIGURE 4. PSO flowchart for optimizing 3D indoor layout design.

data volume and remove noise. Segmentation focused on extracting major planes (floors/walls) using the Random Sample Consensus (RANSAC) algorithm. Finally, Registration/Fusion of the RGB-D and LiDAR point clouds was accomplished using the refined Iterative Closest Point (ICP) algorithm.

The sensors record spatial and depth information to create a dynamic indoor space model. The system tracks object locations over time. While the trajectory is physically a continuous function $P(t)$, the system processes it as a sequence of digital samples. Let $P_t = P(t\Delta t)$ denote the discrete-time sampled position at time index t , where Δt is the sampling interval. Mathematically, the captured raw position vector at each step is represented as:

$$P(t) = (x(t), y(t), z(t)) \quad (10)$$

where $P(t)$ is the position vector of an object at time t , and $(x(t), y(t), z(t))$ are the coordinates in 3D space, with $t \in [0, T]$.

To reduce noise from raw inputs and provide continuous tracking stability, an Exponentially Weighted Moving Average (EWMA) filter is applied to this discrete sequence:

$$S(P)_t = \alpha P_t + (1 - \alpha) S(P)_{t-1} \quad (11)$$

where $S(P)_t$ is the smoothed position vector at time index t , P_t is the raw position input, and α (with $0 < \alpha \leq 1$) is the smoothing factor controlling the contribution of the

current sample relative to the historical average. This explicit discretization bridges the gap between the continuous physical movement and the discrete-time recursive filtering used in the system's software pipeline.

This offers greater precision for downstream visualization pipelines. The output is stored as structured point cloud data and can be further analyzed and segmented in later stages of the system.

B. EXPANSION OF POINT CLOUD AND SPATIAL PROPAGATION

Once initialized, the system extrapolates growth spatially over an indoor map. It is executed in a graph traversal model of areas as nodes and connectivity as edges. It begins from a seed node and grows outward based on detection confidence.

Mathematically, the propagation can be expressed as:

$$V_{s+1} = V_s \cup \{v \in N(V_s) \mid \text{Confidence}(v) > \theta\} \quad (12)$$

where V_{s+1} is the set of spatial nodes after step $s+1$, V_s is the set of nodes at step s , $N(V_s)$ is the set of neighboring nodes to V_s , v is an individual node, $\text{Confidence}(v)$ is the detection confidence score of node v , and θ is the minimum required confidence threshold.

This model ensures growth of the spatial regions only if the adjacent nodes have above a certain confidence level θ . The growth phases are depicted in the following figure. The growth stage of S-space propagation is controlled by the confidence level θ , as shown in Figure 5:

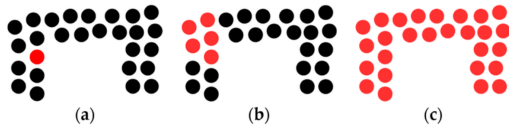


Figure 5 Spatial Propagation Growth Phases Controlled by Confidence Level θ

FIGURE 5. Spatial Propagation Growth Phases Controlled by Confidence Level θ .

This figure illustrates the iterative process of spatial propagation across four sequential phases ($s = 1$ to $s = 3$). The initial propagation starts from a set of seed points, which are marked as red dots in the visualization. These seed points are often highly reliable feature points identified during pre-processing.

The expansion (or growth) from the existing set of nodes (V_s) to the next set (V_{s+1}) is governed by the confidence level threshold, θ , as formally defined in Equation (12). Specifically, a higher confidence level θ leads to a more rigorous filter, resulting in a slower, more conservative spatial expansion but higher data fidelity. Conversely, a lower θ allows for the inclusion of less certain neighboring nodes, facilitating a faster, broader expansion that covers more space rapidly, albeit with a potential increase in noise.

C. RENDERING LOGIC AND REAL-TIME PROJECTION

Rendering is the last stage when processed data is transformed into visual content. The system utilizes rendering matrices that merge depth, lighting, and color data. The matrices define how each point will be visually represented in the final scene.

Real-time performance is measured in terms of frame rate:

$$F = \frac{\sum_{k=1}^n \text{FPS}_k}{n} \quad (13)$$

where F is the average frame rate, FPS_k is the frame rate measured at time k , and n is the number of measurement intervals.

This enables the system performance to be fine-tuned for seamless interaction.

D. SEMANTIC TAGGING AND OBJECT IDENTIFICATION

This entails annotation and identification of the structures found on the basis of semantic similarity. The objects are matched against pre-defined classes based on a similarity index.

In the context of indoor display optimization, semantic tagging focuses on marking critical functional and spatial attributes, including:

Object Functionality: Identifying the purpose of an object (e.g., loom, checkout counter, fire door, desk) to allow the system to apply use-based optimization rules.

Required Clearances/Buffer Zones: Marking the minimum safety or operational distance required around an object (e.g., machine operation space, emergency exit width).

Pathing Constraints: Tagging objects with attributes that define them as obstacles or necessary waypoints (e.g., primary foot traffic aisle, forklift driving zone).

Interaction Type: Identifying if the object is for human or machine interaction (e.g., "touchscreen," "requires manual maintenance").

The end product is annotated objects which are preserved in a metadata layer for querying and retrieval. Next comes labeling:

Semantic labeling is essential for high-level user interaction and optimization tasks, particularly for distinguishing between static structural elements (e.g., walls) and movable design objects (e.g., furniture, machinery). This distinction informs the optimization algorithm (PSO) about which objects possess design freedom and are subject to movement.

To ensure that noise negatively impacts the labeling confidence rather than acting as a redundant constant, the semantic labeling is reformulated as a noise-weighted similarity maximization. The noise level is integrated as a penalty factor to the similarity score across the class set, ensuring that objects with high sensor noise have lower overall classification confidence:

$$L(o) = \arg \max_{c \in C} \left(\frac{\text{Sim}(o, c)}{1 + \lambda \cdot \text{Noise}(o)} \right) \quad (14)$$

where $L(o)$ is the assigned label for object o , C is the set of pre-defined classes, $\text{Sim}(o, c)$ is the similarity index between object o and class c , and $\text{Noise}(o)$ is the object-specific noise level, and λ is a sensitivity coefficient. $\text{Noise}(o)$ serves as a denominator that dampens the similarity score; while it remains constant for object o during the argmax operation, it functions as a confidence-weighting mechanism that allows the system to defer or flag uncertain labels when noise levels are high, rather than treating all classifications with equal certainty.

The similarity index, $\text{Sim}(o, c)$, is calculated using a three-dimensional shape context descriptor $S_{3D}(o)$ compared against a pre-trained class signature $S_{3D}(c)$. To ensure that a smaller Euclidean distance in the descriptor space corresponds to a higher similarity score, the index is defined using a reciprocal or subtraction-based normalization:

$$\text{Sim}(o, c) = 1 - \frac{\|S_{3D}(o) - S_{3D}(c)\|_2}{d_{\max}}$$

where $\|S_{3D}(o) - S_{3D}(c)\|_2$ represents the Euclidean distance between the descriptors. Under this formulation, a higher $\text{Sim}(o, c)$ value (approaching 1) correctly indicates a better match, as the normalized distance term decreases.

The noise level, $\text{Noise}(o)$, is defined as the inverse of the object's point cloud density multiplied by its movement variance over the last three frames:

$$\text{Noise}(o) = \frac{1}{\text{Density}(o)} \cdot \text{Variance}(o)$$

This ensures that stationary, dense point clouds (low noise) are prioritized for reliable labeling, while moving

or sparse objects (high noise) have their label confidence penalized.

E. OUTPUT METRICS AND PERFORMANCE EVALUATION

System performance is measured using throughput-based efficiency (expressed as $1/\text{time}$) and accuracy. This efficiency index measures how fast the system is doing work over time and accuracy measures how close output is to ground truth [11].

System efficiency for task completion is defined by the Adjusted Success-Weighted Efficiency (ASWE). To address the theoretical limitation where failed tasks ($n_{ij} = 0$) are ignored regardless of their duration, a penalty mechanism is introduced. Failed attempts are assigned a reciprocal time value of zero, and their impact is reflected by averaging across all initiated trials (total $M \times N$) rather than only successful ones. Furthermore, to distinguish between different failure modes, the formula is refined as:

$$\text{Efficiency} = \frac{1}{N \cdot M} \sum_{i=1}^M \sum_{j=1}^N \frac{n_{ij}}{t_{ij}} \quad (15)$$

where *Efficiency* is the throughput-based efficiency index (the success rate per unit time), M is the number of users, N is the number of tasks, $n_{ij} \in \{0, 1\}$ is the success indicator (1 for success, 0 for failure), and t_{ij} is the time spent on the task. By incorporating the reciprocal of time ($1/t$), this dimensionally consistent index quantifies the system's productive output rate, effectively acting as a task-level throughput measure. In this formulation, if a task fails ($n_{ij} = 0$), its contribution to the summation is 0, effectively penalizing the total efficiency score. By dividing by the total number of attempts $M \cdot N$, the metric ensures that "fast but failed" and "slow but failed" attempts are both appropriately treated as zero-utility events that diminish the average, while the overall success rate directly scales the final efficiency value.

Recall (sensitivity) is calculated to measure the system's ability to identify all relevant objects, as:

$$\text{Recall} = \frac{\text{Correct Detections (True Positives)}}{\text{Total Ground Truth Objects (True Positives + False Negatives)}} \quad (16)$$

Recall (sensitivity) quantifies the proportion of ground-truth objects that are correctly identified and does not account for false positives. To provide a more comprehensive evaluation, the study also considers Precision—the ratio of correct detections to the total number of detections made—to ensure the system does not overreport objects. The metrics assist in measuring system effectiveness, enabling informed decisions regarding system improvements and scalability.

VI. SYSTEM PERFORMANCE AND USER EXPERIENCE

A. PERFORMANCE EVALUATION RENDERING

Rendering performance is one of the most significant measures of the success of a three-dimensional visualization system. It is the measure of how well the system renders and displays graphical data in real time. One of the fundamental equations used to measure rendering efficiency is the frame rate, which is:

$$\text{FPS} = \frac{\text{Frames Rendered}}{\text{Time (in seconds)}} \quad (17)$$

Frames Rendered is the count of frames processed, and Time is the duration of rendering in seconds. Increased FPS means smooth graphics rendering, which is essential to responsiveness and user experience. In theoretical context, we consider systems that target at least a minimum baseline of 60 FPS for consistency maintenance. Rendering time per frame (RTF) is used to estimate system rendering load.

$$\text{RTF} = \frac{1}{\text{FPS}} \quad (18)$$

where FPS is the frames per second, RTF is the rendering time per frame.

A consistent RTF of about 0.0167 seconds (1/60) is desirable. Visual lag can be noticed with frame drops, especially in complex scenes. Although we do not test hardware, benchmarking literature shows that scene complexity, poly count, and texture resolution have direct effects on rendering time. These need to be tuned with level-of-detail (LOD) methods and efficient GPU utilization. Analytical modeling through formulaic relationships enables us to extrapolate performance results without physical deployment.

B. USER INTERACTION FLOW MODELING

User interaction flow modeling determines the degree to which users tend to interact with the interface of the 3D system.

This model facilitates the estimation of how well tasks are being carried out within a virtual interface. Flow modeling also involves transition matrices, a representation of how users transition between system states, including menu interaction and object manipulation. Transition matrices can be optimized by applying decision tree theory to predict probable user paths and reduce bottlenecks. By applying probability theory and stochastic modeling, researchers can model massive levels of user behavior without empirical testing, gaining understanding of interface responsiveness and usability enhancements.

C. SYSTEM RESPONSIVENESS ANALYSIS

Responsiveness may be quantified by how fast the system responds to the user. The system response latency (L) is typically described by the formula:

$$L = T_{\text{response}} - T_{\text{input}} \quad (19)$$

Where L is the system response latency, T_{response} is the time the system makes a response, and T_{input} is the time the input has been provided. To accurately model the computational load and predict system performance under continuous operation, we adopt a resource-centric load model. The Total System Computational Load (L_{comp}) is determined by the processing times of the critical, resource-intensive modules: GPU rendering, real-time point cloud processing, and the iterative Particle Swarm Optimization (PSO) calculation. This instantaneous load is formally expressed as a weighted sum:

$$L_{\text{comp}} = \omega_{\text{GPU}} \cdot T_{\text{GPU}} + \omega_{\text{PC}} \cdot T_{\text{PC}} + \omega_{\text{PSO}} \cdot T_{\text{PSO}} \quad (20)$$

where L_{comp} denotes an aggregated time-weighted workload indicator rather than a direct physical utilization metric. The coefficients ω_{GPU} , ω_{PC} , and ω_{PSO} are normalized weighting factors reflecting the relative importance and sensitivity of system responsiveness to the corresponding modules, with $\sum_i \omega_i = 1$. These weights are not intended to represent actual execution time proportions, but are introduced to balance heterogeneous processing components with different execution characteristics; RTF reflects the end-to-end frame period. T_{GPU} denotes the GPU rendering time per frame measured inside the rendering pipeline (e.g., GPU timestamp queries), which is not equivalent to RTF when other modules run concurrently; T_{PC} is the time spent on point cloud processing (including filtering, segmentation, and tracking); and T_{PSO} is the processing time required for one full iteration of the Particle Swarm Optimization algorithm. This load model directly addresses the system's bottlenecks (GPU, CPU/Memory for point cloud and PSO) and provides a practical framework for resource management, load balancing, and performance tuning within the 3D environment optimization process.

VII. FUTURE WORK AND IMPROVEMENTS

A. INTEGRATION OF AUGMENTED AND VIRTUAL REALITY (AR/VR)

Future research will include further development of the visualization system by integrating Augmented and Virtual Reality. Immersive technologies will give the users a more intuitive feeling of spatial arrangements and system objects, especially for complex indoor spaces. AR/VR integration will also enable real-time interaction and increased accessibility of layered information.

Recommendations:

- Use head-mounted displays for virtual walk-throughs
- Employ VR to train and manage remote maintenance
- Enhance depth perception by using AR overlays

B. ADOPTION OF ARTIFICIAL INTELLIGENCE FOR AUTOMATION

AI can significantly improve system performance by automating object categorization and semantic interpretation.

Machine learning algorithms can learn usage patterns, predict user action, and suggest relevant visual content autonomously without the need for human intervention. This automation will make the system more adaptive and efficient in the long run.

Recommendations:

- Include natural language processing for voice commands
- Train AI models to detect and flag anomalies
- Apply predictive analytics to user behavior and feedback

C. ACCESSIBILITY AND INCLUSIVE DESIGN

Hence, to make the system usable by large numbers, the future research must involve work towards improving accessibility. This entails designing for individuals with visual, hearing, or mobility disabilities. Improving accessibility will close digital divides and facilitate technology take-up on an equal footing.

Recommendations:

- Add subtitles and voice narration
- Support other input devices (e.g., eye-tracking)
- Keep simple navigation in interfaces for convenience

D. REAL-TIME DATA INTEGRATION AND CLOUD COMPUTING

Real-time performance will be required for effective visualization. Next-generation systems would draw live data from IoT sensors and cloud services to support constant update and monitoring. Cloud-based schemes will also facilitate storage of data, shared use, and system scalability.

Recommendations:

- Use cloud storage to dynamically update models
- Incorporate IoT data streams for real-time monitoring
- Allow concurrent access by multiple users for shared work

VIII. CONCLUSION

This study has demonstrated 3D visualization systems to be highly successful in overcoming the limitations of traditional 2D layout methods, particularly in complex industrial environments such as the textile industry, where they offer crucial capabilities for precisely planning the configuration of large-scale machinery and optimizing dynamic workflow paths. These systems enhance precision, user interaction, and spatial awareness, minimizing errors in designs and maximizing workflow effectiveness. Using real-time data capture, semantic tagging, and rendering logic, designers can design, analyze, and iterate layouts in a more natural manner. Mathematical formulas such as the Euclidean distance formula and cost functions ensure positions of objects meet ergonomic and aesthetic functions. Optimization algorithms such as PSO and multi-objective modeling also allow designers to discover the best possible layout compromise among conflicting objectives. Thus, these systems are a significant breakthrough in interior spatial design.

Looking to the future, the paper presents clear directions for further development. They include the integration of immersive AR/VR experiences, automation of processes via the aid of AI, expanding accessibility features, and the application of real-time synchronization through cloud computing. These proposals are in line with emerging technological trends and user needs. Above all, the integration of these advances will make 3D visualization software more adaptive, smart, and accessible. This is especially vital for the textile industry, where factory planners and engineers can leverage this continued innovation to make the design of complex indoor spaces—such as new production lines or optimized warehousing—increasingly efficient, user-centered, and responsive to diverse functional and aesthetic demands.

AUTHOR CONTRIBUTIONS

Mingsong Tang designed, collected and analyzed the data, and drafted the manuscript. Mingsong Tang conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Mingsong Tang participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

CONFLICTS OF INTEREST

The author declares no conflict of interest.

FUNDING

This work was Supported by the 2022 Guangxi Higher Education Institutions Young and Middle-aged Faculty Basic Research Capability Enhancement Project, "Research on Exhibition Design of Liu Sanjie Culture from the Perspective of Rural Revitalization" (Project No. 2022KY0598).

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

- [1] V. Trif, "Spatial Cognition: Theoretical Considerations," *Procedia - Social and Behavioral Sciences*, vol. 187, pp. 168–172, 2015, [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S187704281501825X>.
- [2] L. De Cock, N. Van de Weghe, K. Ooms, I. Saenen, N. Van Kets, and G. Van Wallendael, "Linking the cognitive load induced by route instruction types and building configuration during indoor route guidance: A usability study in VR," *International Journal of Geographical Information Science*, vol. 36, no. 10, pp. 1978–2008, 2022, [Online]. Available: <https://www.tandfonline.com/doi/abs/10.1080/13658816.2022.2032080>.
- [3] Z. Hu, L. Zhang, Q. Shen, X. Chen, W. Wang, and K. Li, "An integrated framework for residential layout designs: Combining parametric modeling, neural networks, and multi-objective optimization for outdoor activity space optimization," *Alexandria Engineering Journal*, vol. 80, pp. 202–216, 2023, [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1110016823007330>.
- [4] S. Nesmachnow, "An overview of metaheuristics: Accurate and efficient methods for optimisation," *International Journal of Metaheuristics*, vol. 3, no. 4, pp. 320–347, 2014, [Online]. Available: <https://www.inderscienceonline.com/doi/abs/10.1504/ijmhur.2014.068914>.
- [5] D. A. Smith, K. Beaumont, T. S. Maurer, and L. Di, "Clearance in drug design: Miniperspective," *Journal of Medicinal Chemistry*, vol. 62, no. 5, pp. 2245–2255, 2018, [Online]. Available: <https://pubs.acs.org/doi/abs/10.1021/acs.jmedchem.8b01263>.
- [6] S. Sepp, S. J. Howard, S. Tindall-Ford, S. Agostinho, and F. Paas, "Cognitive load theory and human movement: Towards an integrated model of working memory," *Educational Psychology Review*, vol. 31, pp. 293–317, 2019, [Online]. Available: <https://link.springer.com/article/10.1007/s10648-019-09461-9>.
- [7] F. Valdez, "Swarm intelligence: A review of optimization algorithms based on animal behavior," in *Recent Advances of Hybrid Intelligent Systems Based on Soft Computing*. Cham, Switzerland: Springer, 2021, pp. 273–298, [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-58728-4_16.
- [8] M. Shariati, M. S. Mafipour, P. Mehrabi, A. Bahadori, Y. Zandi, M. N. A. Salih, et al., "Application of a hybrid artificial neural network-Particle swarm optimization (ANN-PSO) model in behavior prediction of channel shear connectors embedded in normal and high-strength concrete," *Applied Sciences*, vol. 9, no. 24, pp. 5534, 2019, [Online]. Available: <https://www.mdpi.com/2076-3417/9/24/5534>.
- [9] C. Lange, J. Costley, and S. L. Han, "The effects of extraneous load on the relationship between self-regulated effort and germane load within an e-learning environment," *International Review of Research in Open and Distributed Learning*, vol. 18, no. 5, pp. 64–83, 2017, doi: 10.19173/irrodll.v18i5.3028.
- [10] L. Kørnøv, E. R. Boess, J. Gordon, S. Q. Eliassen, M. R. Partidario, and M. B. Monteiro, "Causal Map Tool of Cause-Effect Relations and Biodiversity Mitigation Hierarchy Connected to Spatial Planning," Aalborg, Denmark: Det Danske Center for Miljøvurdering, Aalborg University; 2024.
- [11] A. Rella and F. Vitolla, "Efficiency metrics for performance measurement: A review in higher education of main methods and determinants," *International Journal of Productivity and Performance Management*, vol. 74, no. 3, pp. 841–866, 2025, [Online]. Available: <https://www.emerald.com/insight/content/doi/10.1108/IJPPM-01-2024-0049/full/html>.