Computer Animation and Virtual Worlds

WILEY

RESEARCH ARTICLE

Pedestrian Scene Coverage Control Using Perceptive Quality-Based Virtual Potential Field

Liangliang Cai[1] | Zhuocheng Liu[1] | Zhong Zhou[1,2]

[1]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China | [2]Zhongguancun Laboratory, China

Correspondence: Zhong Zhou (zz@buaa.edu.cn)

Received: 24 January 2025 | **Revised:** 21 December 2025 | **Accepted:** 7 January 2026

Keywords: coverage control | pedestrian monitoring | PTZ camera network | quality measure | virtual potential field

ABSTRACT

Comprehensive observation of target area with pedestrians through pan-tilt-zoom (PTZ) camera networks is crucial in various surveillance applications. However, the dynamic configuration of PTZ cameras increases the difficulty of coordinating multiple cameras to monitor large-scale scenes. Since coverage control in PTZ camera networks has been proven to be an NP-hard problem, many studies have adopted virtual potential field (VPF) algorithms to efficiently obtain approximate solutions. The VPF methods treat camera viewpoints as charged particles. Through repulsive forces between these particles, PTZ camera networks expand scene coverage and reduce overlap between camera fields of view (FoVs). However, VPF-based methods cannot leverage the scene layout and target priority information, failing to cover pedestrians and other critical areas. In this work, we introduce a unified perception quality measure framework that quantifies surveillance importance for scenes, cameras, and pedestrians. Building on this framework, we design a coverage control scheme using a perceptive quality-based virtual potential field. This scheme models target regions and pedestrian priorities as virtual gravitational and attractive forces. It maximizes coverage of key regions, minimizes camera overlap, and supports high-resolution monitoring and tracking of pedestrians. Extensive experiments show that our approach outperforms state-of-the-art methods, achieving superior scene and pedestrian coverage performance.

1 | Introduction

Pan-tilt-zoom (PTZ) camera networks offer wide fields of view (FoVs) and controllable orientations, which are widely deployed in large-scale environments such as shopping malls, train stations, and airports. This flexibility supports comprehensive scene monitoring and detailed observation of the target area with pedestrians. With the expanding size of the PTZ camera network and the increasing volume of collected videos, manually controlling PTZ camera networks to achieve comprehensive scene coverage is extremely difficult for human operators. Several studies [1–4] have proposed automatic control strategies, such as heuristic algorithms, for manipulating multiple PTZ cameras to maximize scene coverage. These studies are referred to as coverage control or dynamic reconfiguration of the PTZ camera network. An effective coverage control method should eliminate both camera overlaps and coverage gaps. Furthermore, it accommodates diverse operator requirements, including coverage of pedestrians and other critical areas.

Some works treated the coverage control for the PTZ camera network as the well-known art gallery problem [5], which was proven as the NP-hard problem. To mitigate the computational complexity, binary integer programming methods (BIP) [6–8], heuristic algorithms [9–13] and gradient-based algorithms [14–16] have been proposed. The BIP algorithm divides the scene into discrete spaces and achieves the optimal solution through linear programming. However, as the scale of the scene

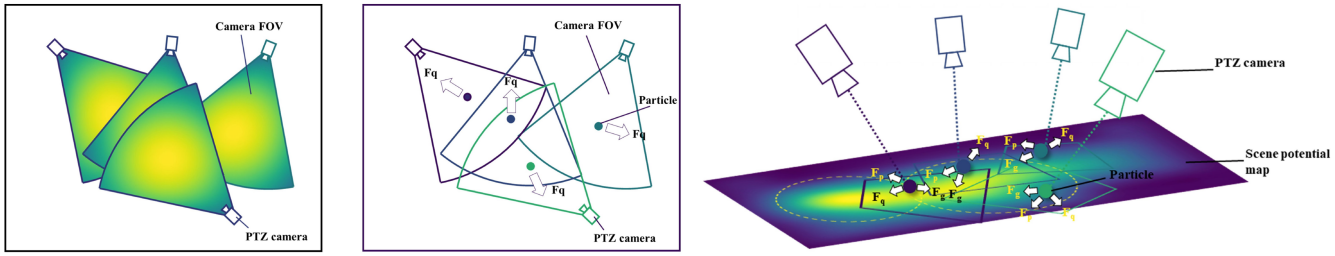

FIGURE 1 | Schematics of the gradient-based, the VPF method, and our method. (a) denotes the gradient-base method, (b) indicates the VPF method, and (c) represents our method with perception measure.

increases, the computational complexity also increases, limiting its applicability to large-scale or complex environments.

To cover the large-scale scene, the gradient-based approach quantifies the camera view through the perception metric function, and selects the gradient update direction as the optimization direction of the camera, as shown in Figure 1a. It eliminates coverage holes in the monitoring area and provides more effective coverage of specific regions based on defined objectives. However, these methods demand high computational resources, particularly in complex environments where multiple monitoring targets exist. Besides, gradient-based methods are susceptible to the initial configuration of the camera network. Poor initialization can lead to inefficient optimization or suboptimal results.

Many researchers have explored heuristic algorithms [12, 13, 17] to obtain approximate solutions in large-scale scene coverage. The virtual potential field (VPF) method is widely used among these studies. The VPF method [18] treats each camera's viewpoint as a charged particle and its field of view as a force field. By introducing repulsive forces between cameras, the VPF method achieves a uniform distribution of viewpoints across the scene, as depicted in Figure 1b. The VPF method can quickly adjust the network configuration according to the state of surrounding cameras with relatively low computational overhead. Because the method cannot leverage scene information or target priority knowledge, it fails to cover pedestrians and other critical areas. We incorporate the perception measures from gradient-based methods into the VPF framework, enhancing its ability to prioritize specific regions or objects within the scene. This integration effectively supports a dynamic and goal-oriented camera control strategy, as illustrated in Figure 1c.

As one of the key targets for scene coverage, pedestrians are detected and tracked in various vision tasks, such as human pose estimation [19], pedestrian re-identification [20], and crowd analysis [21]. Some studies [22, 23] have considered the impact of pedestrian targets on the scene coverage control. Ding et al. [24] incorporated pedestrian utility into the scene coverage control strategy. They developed a pedestrian utility model that includes the tracking utility, the view utility, and the risk of target loss. Giordano et al. [25] employed multiple utility functions to evaluate the perception quality of pedestrians, including the distance from the center, the view quality, the number of cameras per target, and the minimum parameter adjustments. These methods prompt the PTZ camera network to enhance the imaging quality of pedestrians within cameras. These functions focus only on the importance of a single type of target within individual camera views. They lack a unified evaluation framework for assessing diverse object types within the same view, such as pedestrians, regions of interest, and buildings.

In this work, we integrate a unified perception quality measure framework to standardize the evaluation of surveillance importance across diverse targets, including buildings, key observation regions, cameras and pedestrians. Because the perception quality of scenes, cameras and pedestrians relies on distinct assessment mechanisms, we design three tailored perception-quality measures within this unified framework. Building on these measures, we develop a coverage control scheme using a perceptive quality-based virtual potential field. Specifically, for each camera we construct a scene potential map from target region data (region size, building locations and heights), and model PTZ camera viewpoints as virtual charged particles. These particles experience virtual gravity derived from the scene potential map, repulsive forces from neighboring particles, and attractive forces toward pedestrians. Under the combined influence of these forces, each particle is driven toward positions that maximize scene and pedestrian perception quality. It achieves both optimal region coverage and high-resolution pedestrian monitoring, simultaneously minimizing overlap between cameras. The algorithm framework is illustrated in Figure 3.

In real-world applications, cost constraints typically limit PTZ cameras to video capture and transmission, leaving them without sufficient onboard computing power for complex image processing or analysis. Given their reliance on centralized computing nodes for all image analysis tasks, our method adopts centralized coverage control, with computation confined to the powerful sink node. Each PTZ camera only needs to report its initial position as well as orientation and receive the adjusted sensing orientation. This paper is a significant extension of our preliminary version of [26]. Compared with the work [26], the following three aspects have been added. (1) A universal perception quality measure framework is designed. The framework assesses the importance level of different scene objects, including three distinct perception quality measures for scenes, cameras, and pedestrians. (2) A coverage control scheme based on pedestrian perception quality is proposed. The target pedestrian is dynamically monitored after the complete coverage of the large scene is achieved. (3) Extensive experiments are developed and conducted, including the impact of hyperparameters in our method, the visualization

of pedestrian coverage in the simulation scene, and the comparison among runtimes of different methods. In summary, this work has three main contributions:

- A unified perception quality measure framework for different objects (such as scenes, cameras and pedestrians) is presented to evaluate the surveillance importance level from different targets.
- A coverage control scheme using a perceptive quality-based virtual potential field is developed to maximize scene coverage, minimize camera overlap, and dynamically monitor pedestrians with high resolution.
- The extensive experiments demonstrate that our coverage control scheme outperforms state-of-the-art methods, achieving superior scene and pedestrian coverage performance.

2 | Problem Formulation

To make the pedestrian scene coverage control for PTZ camera networks tractable, the reasonable assumptions are formulated.

1. The 3D scene with the PTZ camera network includes common objects such as buildings and pedestrians. Some regions in the scene become essential for various reasons, such as traffic accidents. The terrain of the scene is assumed to be flat, without significant protrusions or depressions.
2. The location of each camera is fixed. Moreover, the central server has access to the location, internal parameters, and external parameters of each PTZ camera. These parameters are essential for calculating the optimal pose of each camera relative to the center node.

We formulate the coverage control problem for a PTZ camera network for a pedestrian scene. The scene $S = \{B, T, P, C, X\}$ encompasses a sampling point set $X = \{X_1, X_2, \ldots, X_{nx}\}$, $X_i = \{x_i, y_i, 0\}$, buildings $B = \{B_1, B_2, \ldots, B_{nb}\}$, $B_i = \{x_i, y_i, 0, l_i, w_i, h_i\}$, important targets $T = \{T_1, T_2, \ldots, T_{nt}\}$, $T_i = \{x_i, y_i, 0\}$, pedestrians $P = \{P_1, P_2, \ldots, P_{np}\}$, $P_i = \{x_i, y_i, 0\} \in X$, and PTZ cameras $C = \{C_1, C_2, \ldots, C_{nc}\}$, $C_i = \{x_i, y_i, z_i\}$. The sampling point set X is obtained by sampling S discretely. Each camera has a sampling point subset $X_{ci} = \{X_1, X_2, \ldots, X_{nc(C_i)}\} \subseteq X$ of the FoV from the C_i. Each important target has a sampling point subset $X_{ti} = \{X_1, X_2, \ldots, X_{nt(T_i)}\} \subseteq X$ of the coverage range from the T_i. The position $\{x, y, z\}$ denotes the 3D sampling point in the S. The $\{l, w, h\}$ expresses the length, the width, and the height of each building. Figure 2 is an example of the S.

The first goal of the pedestrian scene coverage control is to maximize the coverage rate $\rho_{coverage}$ and to minimize the overlap rate $\rho_{overlap}$ of the surveillance scene:

$$\rho_{coverage} = \max_{\Theta} \frac{\| \bigcup_{1 \le i \le nc} X_{ci} \|}{\|X\|}, \tag{1}$$

$$\rho_{overlap} = \min_{\Theta} \frac{\| \bigcup_{1 \le i \le j \le nc} (X_{ci} \bigcap X_{cj}) \|}{\|X\|}. \tag{2}$$

Second, the coverage control for pedestrian scenes aims to maximize coverage rates of important targets:

$$\rho_t = \max_{\Theta} \frac{\| (\bigcup_{1 \le i \le nc} X_{ci}) \bigcap (\bigcup_{1 \le i \le nt} X_{ti}) \|}{\| \bigcup_{1 \le i \le nt} X_{ti} \|}. \tag{3}$$

Finally, the goal maximizes coverage rates of pedestrians:

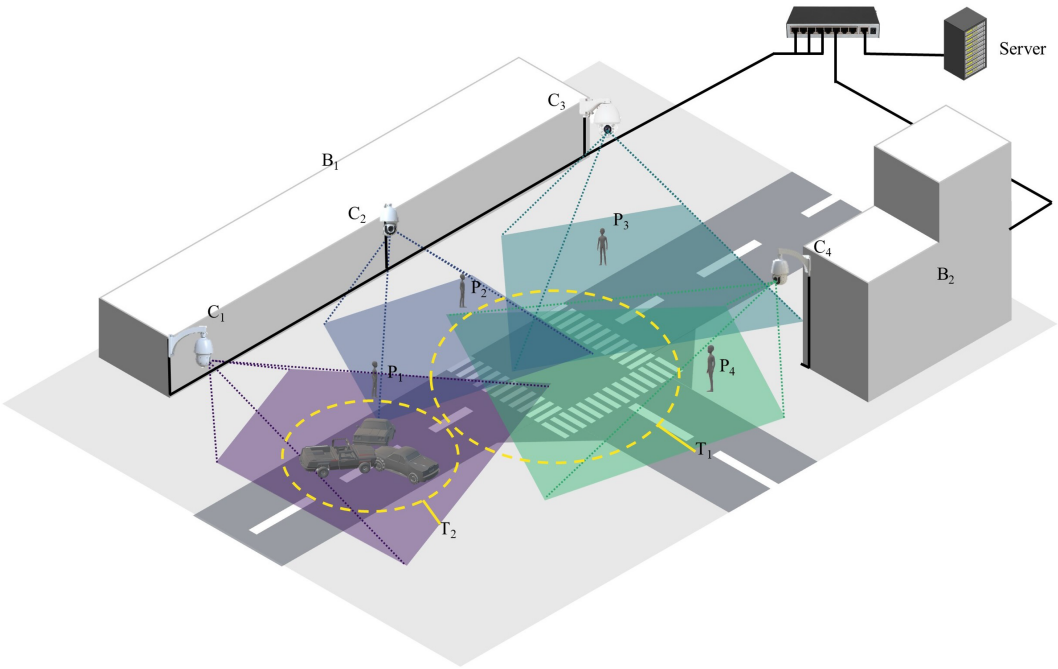

FIGURE 2 | An example of the problem formulation. Multiple PTZ cameras are deployed in a pedestrian scene. The trapezoidal regions represent the observation regions of the cameras. The cameras are connected to a central server for coordination and coverage control.

$$\rho_p = \max_{\Theta} \frac{\|(\bigcup_{1\le i\le nc} X_{ci}) \bigcap P\|}{\|P\|}. \quad (4)$$

Importantly, these three goals are in tension. Focusing on a single goal will neglect the others. The final goal should allow users to adjust the relationship among the three goals as needed:

$$\rho = \alpha_1 * (\rho_{coverage} - \rho_{overlap}) + \alpha_2 * \rho_t + \alpha_3 * \rho_p, \quad (5)$$

where α_1, α_2, and α_3 are used to adjust the importance of different goals. In this study, all three coefficients are set to 1. We try to find the optimal external parameter set of cameras $\hat{\Theta} = \{\hat{\theta}, \hat{\phi}, \hat{L}\}$ from candidates of the external parameter space Θ, which maximizes the final goal ρ of the monitored region. $\{\theta, \phi, L\}$ denotes the pan angle set, tilt angle set, and zoom set of the camera network, respectively.

3 | Coverage Control Solution for PTZ Camera Network

A pedestrian scene includes buildings, essential objects, PTZ cameras, and pedestrians. Given the final goal of problem formulation, we introduce a unified perception quality measure framework. This framework encompasses perception quality measures for buildings, key regions, cameras, and pedestrians. Considering that the VPF method does not inherently support optimization based on perceptual quality, we translate the measure scores into virtual forces. The scene perception quality is translated to the virtual gravity, the camera perception quality is translated to the virtual repulsive force, and the pedestrian perception quality is translated to the virtual attractive force. These forces are then integrated into the optimization process of the VPF method, creating a coverage control scheme based on the perceptive quality-based virtual potential field. This solution leverages the virtual forces derived from the perceptual quality assessments to guide the reconfiguration of the PTZ camera network. It ensures that the cameras focus on high-importance level regions, enhancing the overall surveillance efficiency. Figure 3 shows an algorithm framework of the proposed solution. Please see the Appendix A for the summary of the VPF method.

3.1 | Perception Quality Measure for Different Objects

Due to the inconsistency in the perception quality assessment mechanisms of scenes, cameras, and pedestrians, we develop three distinct perception quality measures for different objects. Existing studies [27] often mistakenly treat camera and scene perception quality as equivalent due to the presence of the projection of the scene in the image. However, significant differences exist between them in reality. Scene perception quality serves as a metric to evaluate the internal quality distribution within large-scale

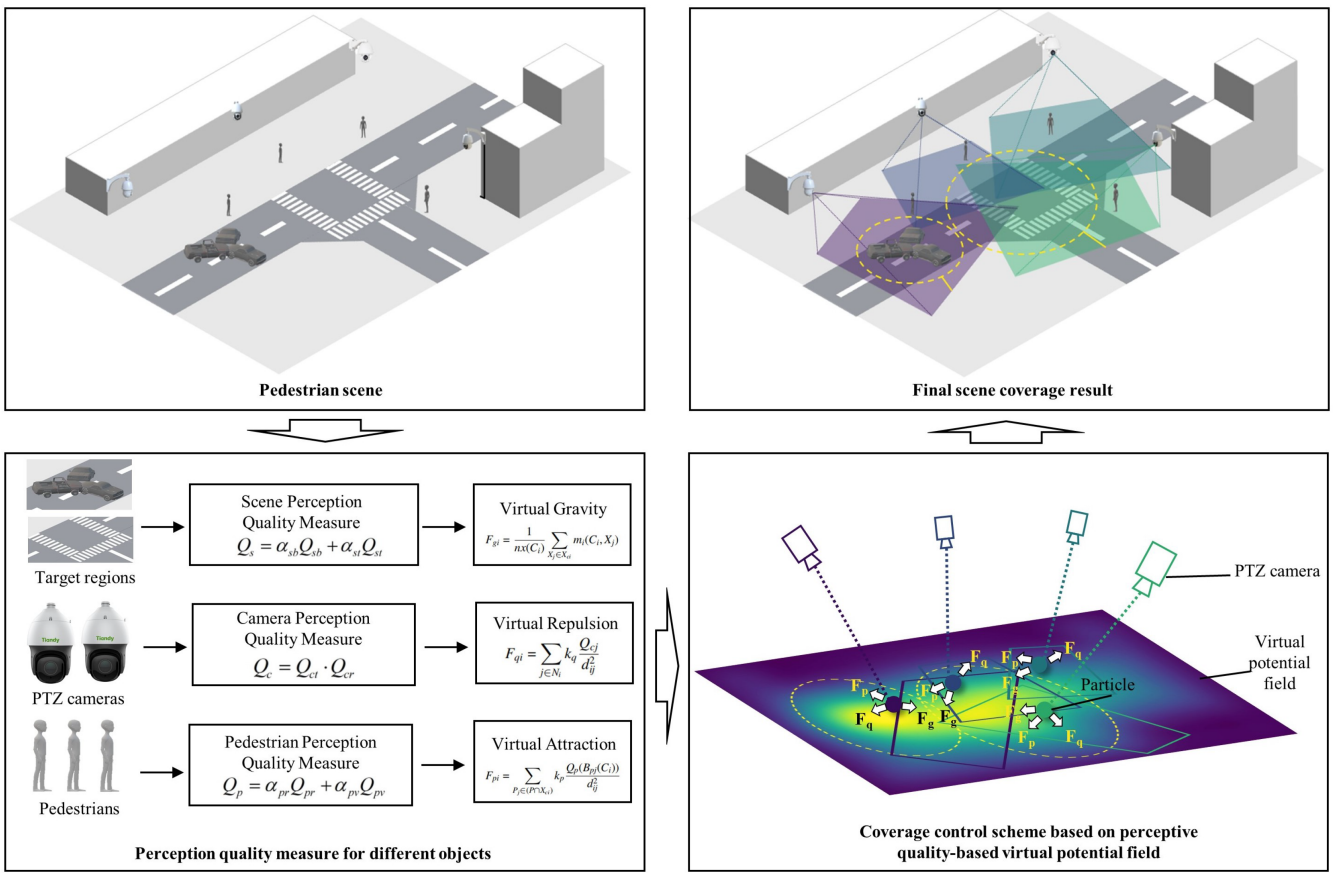

FIGURE 3 | Algorithm framework of the proposed method. The framework includes the pedestrian scene (top-left), scene perception quality measure (bottom-left), coverage control scheme based on the perceptive quality-based virtual potential field, and the final scene coverage result (top-right).

scenes, focusing on the distribution of important regions. In contrast, camera perception quality concentrates on evaluating the quality distribution in the image, independent of the scene information. This distinction is crucial for a more accurate understanding and assessment of perception quality in complex scenes.

3.1.1 | Scene Perception Quality Measure

The gradient-based method assesses the perception quality only within the FoV of the camera and assumes a uniform distribution of perception quality for the global scene [27, 28]. This assumption, however, does not align with the real-world scenario. In actual scenes, the observation region comprises high and low importance regions. Considering the diversity of objects within a scene, different metrics should be applied to assess the perception qualities of different objects. Accordingly, objects in the scene are categorized into three types: persistent, significant, and dynamic. Persistent objects are characterized by permanence and immobility (e.g., buildings). Important objects represent essential regions, such as accidents. Dynamic objects typically refer to pedestrians and vehicles in the scene.

Scene layout perception quality. For buildings in the scene, we set their perception quality $Q_{sb}(X_i) = 0$ (X_i is from the sampling point set of buildings B_i.), due to low observation importance for them. Besides, we set the edge observation quality of the scene to $Q_{sb}(X_i) = 0$ (X_i is from the edge of the scene). These assumptions can be flexibly adjusted according to requirements of operators. For other scene positions, we assume that the further the distance from the "$Q_{sb} = 0$" areas, the more observational information that location carries. Their distance between the sampling point X_i and X_j with $Q_{sb}(X_j) = 0$ are calculated as follows.

$$d(X_i) = \|X_i - X_j\|, \quad X_j \in \{X_j | Q_{sb}(X_j) = 0\}, \tag{6}$$

$$d_n(X_i) = \frac{d(X_i)}{d_{max}}, \tag{7}$$

d_{max} represents the maximum distance of $d(X_i)$ in the large-scale scene. The purpose of Equation (7) normalizes $d(X_i)$. We translate $d_n(X_i)$ to the scene layout perception quality $Q_{sb}(X_i)$ by the Gaussian blur algorithm. The scene layout perception quality $Q_{sb}(X_i)$ is

$$Q_{sb}(X_i) = \sum_{X_i' \in A(X_i)} G_n(X_i, \sigma) d_n(X_i'), \tag{8}$$

where $A(X_i)$ represents the region with center at X_i and a radius of R. $G_n(X_i, \sigma)$ denotes the normalized Gaussian kernel, which can be identified as

$$G_n(X_i, \sigma) = \frac{G(X_i, 0, \sigma)}{\sum_{X_i' \in A(X_i)} G(X_i, X_i', \sigma)}, \tag{9}$$

$$G(X_i, X_i', \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(X_i' - X_i)^2}{2\sigma^2}}. \tag{10}$$

Important target perception quality. In reality, a large scene includes some important regions such as crossroads, sites of accidents, and so on. These critical regions may be covered preferentially and called as important targets. So we formulate the perception quality of important targets by a Gaussian mixture distribution with means $\{T_i | i = 1, 2, \ldots, nt\}$ and variances $\{\sigma_i^2 | i = 1, 2, \ldots, nt\}$:

$$Q_{st}(X_i) = \frac{1}{nt} \sum_{j=1}^{nt} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{(X_i - T_j)^2}{2\sigma_j^2} \right), \tag{11}$$

where nt denotes the number of important targets, T_i denotes the location of important objects, and σ_j denotes the influence range of important object T_j. The scene perception quality measure is expressed as

$$Q_s(X_i) = \alpha_{sb} * Q_{sb}(X_i) + \alpha_{st} * Q_{st}(X_i). \tag{12}$$

The visualization results of the scene perception quality measure are presented in Figure B1 in Appendix B.

3.1.2 | Camera Perception Quality Measure

This work extends the 2D camera perception quality measure proposed by Arslan [27] to that for 3D cameras. The 3D perception quality measure combines perspective and resolution quality measures as follows.

Camera perspective quality. Visual perception quality is known to deteriorate away from the optical axis of a camera in the boundary of its field of view due to increases in lens distortion, incompleteness of visual data, and non-persistence [29]. Arslan et al. [27] only investigated the effort of the angle of view in the horizontal axis. Instead, we additionally introduce the role of the vertical axis and propose a novel perspective quality for the 3D camera model:

$$\gamma_p(C_i, X_j) = arccos \frac{(X_j - C_i)^2 (X_{cen}(C_i) - C_i)^2 - (x_j - x_{ci})^2}{2\|X_j - C_i\| \|X_{cen}(C_i) - C_i\|}, \tag{13}$$

$$\gamma_t(C_i, X_j) = arccos \frac{(X_j - C_i)^2 (X_{cen}(C_i) - C_i)^2 - (y_j - y_{ci})^2}{2\|X_j - C_i\| \|X_{cen}(C_i) - C_i\|}, \tag{14}$$

$$Q_{ct}(\gamma_p(C_i, X_j), \gamma_t(C_i, X_j)) = \frac{(\cos(\gamma_p(C_i, X_j)) - \cos(\gamma_{p0}(C_i)))}{(1 - \cos(\gamma_{p0}(C_i)))} \frac{(\cos(\gamma_t(C_i, X_j)) - \cos(\gamma_{t0}(C_i)))}{(1 - \cos(\gamma_{t0}(C_i)))}, \tag{15}$$

$\gamma_p(C_i, X_j)$ and $\gamma_t(C_i, X_j)$ are the horizontal and vertical angles between the center line of sight and the line between the camera centroid $X_{cen}(C_i)$ and the target point X_j, respectively. $\gamma_{p0}(C_i)$ and $\gamma_{t0}(C_i)$ are the maximal horizontal and vertical angles of the FoV of the camera C_i, respectively. Equation (15) limits the target point not exceeding the FoV.

Camera resolution quality. Similar to the resolution quality proposed by Arslan, our resolution quality is expressed by:

$$Q_{cr}(C_i, X_j) = \frac{e^{\left(-\frac{(\|X_j - X_{cen}(C_i)\| - D(C_i))^2}{2\sigma(C_i)^2}\right)}}{\max(Q_{cr}(C_i, X_j))}, \tag{16}$$

where $\sigma(C_i) = \frac{D(C_i)}{3}$, and $\|X_j - X_{cen}(C_i)\|$ represents the distance between the target point X_j and the camera centroid $X_{cen}(C_i)$. $D(C_i) = (L(C_i) \cdot f(C_i))/W_{ccd}(C_i)$ is the camera working distance, which denotes the distance between the object and the camera when the camera observes the target region. $f(C_i)$ is the focal length of the camera C_i, $W_{ccd}(C_i)$ is the width of the camera charge-coupled device (CCD) from the C_i, and $L(C_i)$ is the zooming level of the camera C_i. $\max(Q_{cr}(C_i, X_j))$ indicates the maximum of the $Q_{cr}(C_i, X_j)$. The camera perception quality measure is expressed as

$$Q_c(C_i, X_j, \gamma_p(C_i, X_j), \gamma_t(C_i, X_j)) = Q_{ct}(\gamma_p(C_i, X_j), \gamma_t(C_i, X_j)) \cdot Q_{cr}(C_i, X_j). \tag{17}$$

The visualization results of the camera perception quality measure are shown in Figure B2 in Appendix B.

3.1.3 | Pedestrian Perception Quality Measure

As one of the key targets for scene coverage, pedestrians are detected and tracked in various vision tasks. Hence, the pedestrian perception quality measure must fulfill the requirements of these visual tasks. Most visual analysis tasks prefer high resolution, central positioning in the image rather than at the edges, and a frontal view rather than a top-down perspective. The pedestrian re-identification technology [30] repeatedly detects pedestrian targets in images and provides the detection bounding box $B_{pi}(C_i) = \{u_{pi}(C_i), v_{pi}(C_i), w_{pi}(C_i), h_{pi}(C_i)\}$ for the target pedestrian P_i. The pedestrian perception quality comprises the resolution quality and the view quality.

Pedestrian resolution quality. The size of the detection bounding box $B_{pi}(C_i)$ corresponds to the resolution quality of the pedestrian. The larger the size of $B_{pi}(C_i)$, the more pixels the pedestrian occupies in the image I. Set $\rho_{pj}(B_{pj}(C_i))$ denotes the occupation rate of pedestrian in the image. The $\rho_{pj}(B_{pj}(C_i))$ is

$$\rho_{pj}(B_{pj}(C_i)) = max\left\{\frac{w_{pj}(C_i)}{w_I(C_i)}, \frac{h_{pj}(C_i)}{h_I(C_i)}\right\}. \tag{18}$$

The resolution quality for pedestrian is follows:

$$Q_{pr}(\rho_{pj}(B_{pj}(C_i))) = \frac{1}{1 + e^{-k_{pr}(\rho_{pj}(B_{pj}(C_i)) - \rho_0)}}. \tag{19}$$

Equation (19) represents a Logistic function that exhibits stability when $\rho_{pj}(C_i)$ is either very small (approaching 0) or very large (approaching 1). k_{pr} governs the steepness of the resolution quality curve, influencing how rapidly the quality changes. The parameter ρ_0 defines the critical point where Q_{pr} experiences a significant shift. The impact of Q_{pr} with different k_{pr} and ρ_0 were discussed in Figure B3 in Appendix B. We set k_{pr} to 10 and ρ_0 to 0.5.

Pedestrian view quality. The pedestrian in the image is expected at the center rather than the edge. Let $d_{pj}(B_{pj}(C_i))$ denote the distance between the pedestrian center and the image center. Let $d_I(C_i)$ reveals the half diagonal of the image I of the camera C_i. They are computed by:

$$d_{pj}(B_{pj}(C_i)) = \sqrt{\begin{matrix}(u_{pj}(C_i) + \frac{w_{pj}(C_i)}{2} - u_{wI}(C_i))^2 \\ + (v_{pj}(C_i) + \frac{h_{pj}(C_i)}{2} - u_{hI}(C_i))^2\end{matrix}}, \tag{20}$$

$$d_I(C_i) = \frac{\sqrt{(w_I(C_i)^2 + h_I(C_i)^2)}}{2}. \tag{21}$$

The view quality measure for pedestrians is:

$$Q_{pv}(d_{pj}(B_{pj}(C_i))) = \frac{1}{1 + e^{-k_{pv}(\frac{d_{pj}(B_{pj}(C_i))}{d_I(C_i)} - d_0)}}. \tag{22}$$

The Logistic function can sensitively capture variations in Q_{pv}. Near the center of the image, even minor pedestrian movements can cause rapid changes in Q_{pv}. Instead, at the edges of the image, significant translations of the pedestrian do not greatly impact Q_{pv}. This is advantageous for maintaining the pedestrian at the center of the image. The d_0 represents the boundary where the view quality changes rapidly. The impact of Q_{pv} with different k_{pv} and d_0 is shown in Figure B3 in Appendix B. We set k_{pv} to 10 and ρ_0 to 0.8. The pedestrian perception quality measure is expressed as

$$Q_p(B_{pj}(C_i)) = \alpha_{pr} * Q_{pr}(\rho_{pj}(B_{pj}(C_i))) + \alpha_{pv} * Q_{pv}(d_{pj}(B_{pj}(C_i))). \tag{23}$$

3.2 | Coverage Control Scheme Based on Perceptive Quality-Based Virtual Potential Field

According to potential field theory, each centroid can be treated as a virtual charged particle. There exist repulsive forces between adjacent particles. In our approach, we convert the perception qualities of different objects into different virtual forces. We introduce the concept of the scene potential map, where virtual particles are subject to the force that pushes them to positions with higher scene perception quality. The force of the scene potential map is called virtual gravity. The camera perception quality is converted into the charge carried by the particles. The greater the perception quality of the camera, the greater the repulsive force of the current particle on other particles. Pedestrians are regarded as a particular type of charged particles, and they exert virtual attraction forces on the virtual particles corresponding to the camera. Based on these virtual forces, we propose a novel virtual force analysis method to address the coverage control problem. Besides, we involve the region partition method based on perception quality measures to divide the region, which helps compute acceptable zoom values of cameras.

3.2.1 | Virtual Force Analysis Method Based on Scene Potential Map

The proposed scene potential map describes the perception quality distribution of the scene. Since the camera location differs, every camera has a unique scene potential map. We generate the scene potential map by:

$$m(C_i, X_j) = \frac{Q_s(X_j)}{\|X_j - C_i\|}, \tag{24}$$

where X_j is the ground point of the target scene, C_i indicates the location of the camera C_i, and $Q_s(X_j)$ indicates the scene perception quality formulating as Equation (12).

We treat the coverage area centroid of the PTZ camera as a virtual charged particle. Several simple force laws are defined as follows:

1. Any particle is subject to virtual gravity from the scene potential map. Under the gravity, the particle will move to the region with the best scene and camera perception quality.
2. The repulsive force exists among multiple neighboring centroid points. The repulsive force pushes the particle far away from other particles.

Considering that the centroid represents the coverage area of the PTZ camera, the gravity of every particle is the average of all scores of the scene potential map within X_{ci} of the camera C_i. The virtual gravity is defined as follows:

$$F_{gi} = \frac{1}{nx(C_i)} \sum_{X_j \in X_{ci}} m_i(C_i, X_j) \quad i \in \{1, 2, 3, \ldots, nc\}. \tag{25}$$

The direction of F_{gi} points to the point with the lowest point in the scene potential map. The repulsive force between particles is defined as:

$$Q_{ci} = \frac{1}{nx(C_i)} \sum_{X_j \in X_{ci}} Q_c(C_i, X_j, \gamma_p(C_i, X_j), \gamma_t(C_i, X_j)), \tag{26}$$

$$F_{qi} = \sum_{j \in N_i} k_q \frac{Q_{cj}}{d_{ij}^2} \qquad i \in \{1, 2, \ldots, nc\}. \tag{27}$$

d_{ij} indicates the distance between the two particles $X_{cen}(C_i)$ and $X_{cen}(C_j)$, and k_q is the experience coefficient. Q_{cj} represents the charge of particle. The greater the charge, the smaller the overlap area between the cameras. N_i is the neighboring set of the ith camera within radius r_{ci}. The direction of F_{qi} follows the parallelogram guideline. Next, we introduce the virtual force analysis method. First, the initial camera orientation is randomly set, and virtual patrials are randomly distributed in the target scene. Then we compute the total force on every particle $F_i = \alpha_g F_{gi} - \alpha_q F_{qi}$. The direction of total force F_i is decided by directions of gravity and repulsive force. Next, we update the position of every particle to a new position $(x_i + F_{ix}, y_i + F_{iy})$ in the next iteration. F_{ix} is the component of F_i on the x-axis and F_{iy} is that on the y-axis. The virtual force analysis method stops when the number of iterations exceeds the preset N_t, or the total force of each particle is smaller than the threshold ϵ.

After the final position of the centroid is obtained, we compute the pan angle θ_{ci} and tilt angle ϕ_{ci} of each camera by:

$$\theta_{ci} = \begin{cases} arctan\left(\frac{y_{ci}}{x_{ci}}\right) + \frac{\pi}{2} & x_{ci} > 0 \\ arctan\left(\frac{y_{ci}}{x_{ci}}\right) + \frac{3\pi}{2} & x_{ci} < 0 \end{cases}, \tag{28}$$

$$\phi = \frac{\pi}{2} - arcsin\left(\frac{\sqrt{x_{ci}^2 + y_{ci}^2}}{\|X_{cen} - C_i\|}\right), \tag{29}$$

where (x_{ci}, y_{ci}) denotes the viewpoint in the projection plane of the C_i, X_{cen} and C_i denote 3D positions of the centroid and the camera position. The initial orientation of the pan angle of the PTZ camera is consistent with the negative y-axis. The algorithm is shown in Algorithm 1.

3.2.2 | Region Partition Method Based on Camera Perception Quality

The region partition method aims to divided the entire scene into several regions with the highest camera perception quality. Equation (24) roughly characterizes the perception quality of the PTZ camera. The region partition method utilizes the camera perception quality measure to assign points of the target region to the finest camera. The rule is as follows.

$$V_i = \{X_k \in X | Q_c(C_i, X_k, \gamma_p(C_i, X_k), \gamma_t(C_i, X_k)) \geq Q_c(C_j, X_k, \gamma_p(C_j, X_k), \gamma_t(C_j, X_k))\}, \quad i, j \in [1, 2, \ldots, nc], \tag{30}$$

where V_i denotes the best observation region for the C_i, Q_c represents the camera perception quality, and the X denotes the sampling point set of the scene. X_k represents a sampling point in the set X.

After obtaining the region partitions, we find the best zoom value that can improve the resolution of the coverage area. We set the zoom values of all cameras to a maximum. If all target points of the partition are in the sight of the camera, the estimated zoom level will be the best zoom value. Otherwise, the zoom value decreased one. If the zoom value reaches the minimum, the iteration stops.

3.2.3 | Coverage Control Scheme Based on Pedestrian Perception Quality

In our method, pedestrians are treated as special particles that have an attraction force on the camera particle. The attraction force of the pedestrian to the camera particle satisfies the following:

$$F_{pi} = \sum_{P_j \in (P \cap X_{ci})} k_p \frac{Q_p(B_{pj}(C_i))}{d_{ij}^2} \quad i \in [1, 2, \ldots, nc], \tag{31}$$

where d_{ij} denotes the distance between 3D points of $X_{cen}(C_i)$ and P_j, and k_p indicates the empirical constant. The camera centroid will only be attracted by pedestrians when they are within the FoV range X_{ci} of the camera C_i. The direction of F_{pi} is where the camera particles to the pedestrian positions. F_p is added to the virtual force analysis of F_g and F_q.

To minimize frequent communication within the camera network, we update the parameters of cameras monitoring pedestrians using a threshold ϵ_p. When $Q_p > \epsilon_p$, the selected camera need not adjust its position. An adjustment is required only if $Q_p \leq \epsilon_p$, indicating a terrible view of the pedestrian. If the target position for the subsequent adjustment exceeds the FoV range of the PTZ camera, a new camera must be selected to dynamically tracking.

To improve the pedestrian resolution, we also adjust the zoom value of the PTZ camera as follows:

ALGORITHM 1 | Virtual force analysis method based on scene potential map.

Input: The scene $S = \{B, T, C, X\}$, the building set B, the important target set T, the camera set C, the camera parameter set Θ.
Output: the camera parameter set $\hat{\Theta}$.

```
Initialize a camera list List[nc]
for C_i in the camera set C do
  Initialize the scene potential map Map={}
  for X_j in the sampling point set X do
    compute scene perception quality Q_s(X_j) by Eq. 12
    compute scene-camera potential map m(C_i, X_j) by Eq. 24
    add {X_j : m(C_i, X_j)} into Map
  end for
  set List[i]=Map
end for
initialize the F[nc], F_g[nc], F_q[nc]
initialize t = 0
for t<N_t do
  t ← t + 1
  for i in [1 : nc] do
    compute the FoV range X_ci and centroid position (x_cen(C_i), y_cen(C_i)) of i-th camera according
      the Θ_i
    compute the F_gi by Eq. 25
    compute the direction of gravity x_gi and y_gi
    compute X-axis component F_gx(C_i) = F_gi * x_gi and Y-axis component F_gy(C_i) = F_gi * y_gi of F_gi
    for j in N_i do
      compute the j-th repulsive force F_qi(C_j) by Eq. 27
      compute the direction of repulsive force x_qi(C_j) and y_qi(C_j)
      compute X-axis component F_qx(C_j) = F_qi(C_j) * x_qi(C_j) and Y-axis component F_qy(C_i) = F_qi(C_j) * y_qi(C_j)
        of F_qi(C_j)
      compute F_qx[i] ← F_qx[i] + F_qx(C_j) and F_qy[i] ← F_qy[i] + F_qy(C_j)
    end for
    compute F_x[i] ← α_g F_gx[i] − α_q F_qx[i] and F_y[i] ← α_g F_gy[i] − α_q F_qy[i]
    if F_x(C_i)<ε then
      continue
    end if
    update the centroid position x_cen(C_i) ← x_cen(C_i) + F_x[i] and y_cen(C_i) ← y_cen(C_i) + F_y[i]
  end for
end for
for C_i in C do
  compute the optimal parameter set (θ̂(C_i), φ̂(C_i)) by Eq. 28 and 29
end for
```

$$L_{best}(C_i, B_{pj}(C_i)) = \left\lfloor min\left\{ \frac{w_I(C_i)}{w_{pj}(C_i)}, \frac{h_I(C_i)}{h_{pj}(C_i)} \right\} \cdot L_{cur} \right\rfloor, \quad (32)$$

where $min\{\frac{w_I(C_i)}{w_{pj}(C_i)}, \frac{h_I(C_i)}{h_{pj}(C_i)}\}$ denotes the inverse of occupancy between the image and the pedestrian detection box and $\lfloor\rfloor$ represents the floor function symbol. Equation (32) stands for the best zoom set for the selected camera. The algorithm is shown in Algorithm 2.

4 | Experimental Simulations

In conducting pedestrian scene coverage experiments, the deployment of the PTZ camera network in real-world scenarios lacks the repeatability afforded by simulated setups. Consequently, we adopt simulation experiments to evaluate the performance of our proposed method. In this section, we provide numerical evidence demonstrating the effectiveness of the proposed coverage control schemes. We conducted experiments on two distinct coverage goals: scene coverage and pedestrian coverage. We also conducted the experiments exploring the impact of the selection of hyperparameters α_g and α_q in Appendix C. We identified the parameter values encircled in orange in Figure C1 as the optimal settings of the α_g and α_q for different scenarios.

Baselines. In the scene coverage, we compare our method with two competing methods: distributed coverage control scheme (DCCS) [28] and a parametric voxel-based analysis (PBA) [31]. The former is a state-of-the-art gradient-based algorithm suitable for a 3D camera model. The latter approach extracts layout, buildings, and key areas from the BIM model, converts the 3D space into voxels, and applies the Non-Dominated Sorting

ALGORITHM 2 | Coverage control scheme based on pedestrian perception quality.

Input: The scene $S = \{B, T, P, C, X\}$, The buildings set B, the important target T, the pedestrain set P, the camera set C, the camera parameter set Θ
Output: the camera parameter set $\hat{\Theta}$.
run the algorithm 1 and the region partition method based on camera perception quality
initialize the $F[nc]$, $F_g[nc]$, $F_q[nc]$, $F_p[nc]$
initialize $t = 0$
for $t < N_t$ **do**
  $t \leftarrow t + 1$
  for i in $[1 : nc]$ **do**
    compute the F_{gi}, $F_{gx}(C_i)$, and $F_{gy}(C_i)$
    compute the$F_q(C_j)$, $F_{qx}(C_j)$, and $F_{qy}(C_i)$
    for $P_j \in P$ **do**
      if $P_j \in X_{ci}$ and $Q_p(B_{pj}(C_i)) \leq \epsilon_p$ **then**
        compute the F_{pi} by Eq. 31
        compute the direction of x_{pi} and y_{pi}
        compute X-axis component $F_{px}(C_i) = F_{pi} * x_{pi}$ and Y-axis component $F_{py}(C_i) = F_{pi} * y_{pi}$ of F_{pi}
        compute $F_{px}[i] \leftarrow F_{px}[i] + F_{px}(C_j)$ and $F_{py}[i] \leftarrow F_{py}[i] + F_{py}(C_j)$
      end if
    end for
    compute $F_x[i] \leftarrow \alpha_g F_{gx}[i] - \alpha_q F_{qx}[i] + \alpha_p F_{px}[i]$ and $F_y[i] \leftarrow \alpha_g F_{gy}[i] - \alpha_q F_{qy}[i] + \alpha_p F_{py}[i]$
    if $F_x(C_i) < \epsilon$ **then**
      continue
    end if
    update the centroid position $x_{cen}(C_i) \leftarrow x_{cen}(C_i) + F_x[i]$ and $y_{cen}(C_i) \leftarrow y_{cen}(C_i) + F_y[i]$
  end for
end for
for C_i in C **do**
  compute the optimal parameter set $(\hat{\theta}(C_i), \hat{\phi}(C_i))$ by Eq. 28 and 29
  compute the optimal zoom set $\hat{L}(C_i, B_{pj}(C_i))$ by Eq. 32
end for

Genetic Algorithm II (NSGA-II) to select camera parameters. For these two methods, we adopt the same parameters as suggested in their papers. In the pedestrian coverage, our method is compared with the tracking of multiple targets in the visual sensor network (VSNTMT) [25] and the collaborative sensing in a distributed PTZ camera network (CSDVSN) [24], both representative methods in the pedestrian quality measure field. The VSNTMT method defines a utility function that takes into account several factors, including the distance between pedestrian bounding boxes and the center of the image, the view quality of the camera, the number of targets covered by the camera, and the minimal steps of camera parameter adjustments. The CSDVSN method requires cameras to observe the faces of pedestrians. If the current camera cannot observe faces, it will update the orientation of all cameras and then select a new camera that can monitor the face of the pedestrian.

Metrics. Our evaluation treats the scene coverage rate and the overlap rate as critical indicators for the scene coverage. The scene coverage and camera overlap rate are computed by Equations (1) and (2), respectively. In the pedestrian coverage, the metrics are placed on comparing perceived qualities of pedestrians based by various perception quality methods.

4.1 | Experiments on the Scene Coverage

In order to evaluate the performance of our method, we constructed three types of scenes: empty scenes, scenes with buildings, and scenes with key targets. For each scene type, 20 instances were randomly generated based on the parameters listed in Table 1. These scenes were primarily used for the experimental comparisons presented in Figures 4 and 5, and Table 2. Within these scenario instances, we have assumed that the initial orientation of the camera is perpendicular to the ground, which translates (θ, ϕ) to $(0°, 90°)$. Furthermore, all cameras utilized in the experiments follow the assumption of the pinhole camera model. The hardware platform for our experiments is a computer equipped with an Intel i7 CPU and 16GB of RAM. The software platform is Matlab R2018a.

In an empty scene devoid of buildings or targets, we conducted a comparative analysis of the coverage and overlap performance of three distinct methodologies. The results presented in Figure 4 illustrated that our approach achieved a significantly higher coverage rate and a lower overlap rate than the other two methods. While PBA shares similarities with our method by considering the impact of buildings, and regions of interest on scene coverage, its voxel-based discretization of the entire 3D scene introduces many noncritical regions, such as building facades. By attending to these irrelevant areas, PBA fails to focus on the most important regions, which degrades its overall coverage performance. Furthermore, the experiment results of the DCCS can be attributed to the initial downward orientation of the cameras, resulting in gradients to be close to 0 after each iteration. Consequently, the DCCS method barely updated, yielding results strikingly similar to the initial setup, failing to meet the desired coverage results. This is a typical limitation of the DCCS method. In the 5th scene shown in Figure 4c, the cameras were concentrated in the central area of the scene. The cameras in the 10th scene were scattered around the scene. The concentrated camera placement makes it difficult for our method to achieve high-quality scene coverage

TABLE 1 | Parameter settings for simulation scenarios randomly generated.

Parameter type	Notion	Description	Scene type
Fixed parameters	Scene number	20 scenes are generated by the parameters from Table 1	All scenes
	Scene Size	The size of each scene is $100m \times 100m$	All scenes
	Sampling point set	Each scene is sampled by one unit and has a sampling point set with a size of 100×100	All scenes
	Camera number	There are 20 cameras in each virtual scene	All scenes
	Camera height	Heights of all cameras are 10m	All scenes
	Camera FoV	FoVs of all cameras are 60°	All scenes
	Building number	There are 3 buildings in each virtual scene	Scenes with buildings
	Building height	Heights of all buildings are 20m	Scenes with buildings
	Number of important target	There is an important target in each virtual scene	Scenes with targets
	Radius of important target	The radius of each important target is 3m	Scenes with targets
Random parameters	Building location	Building locations are randomly selected from the set of sampling point	Scenes with buildings
	Building area	The proportion of the building area to the total scene area is [0.05, 0.1]	Scenes with buildings
	Camera position	Camera locations are randomly selected from the set of sampling point and not fall in ranges of buildings	All scenes
	Position of important target	The position of important target is randomly selected from the set of sampling point	Scenes with targets

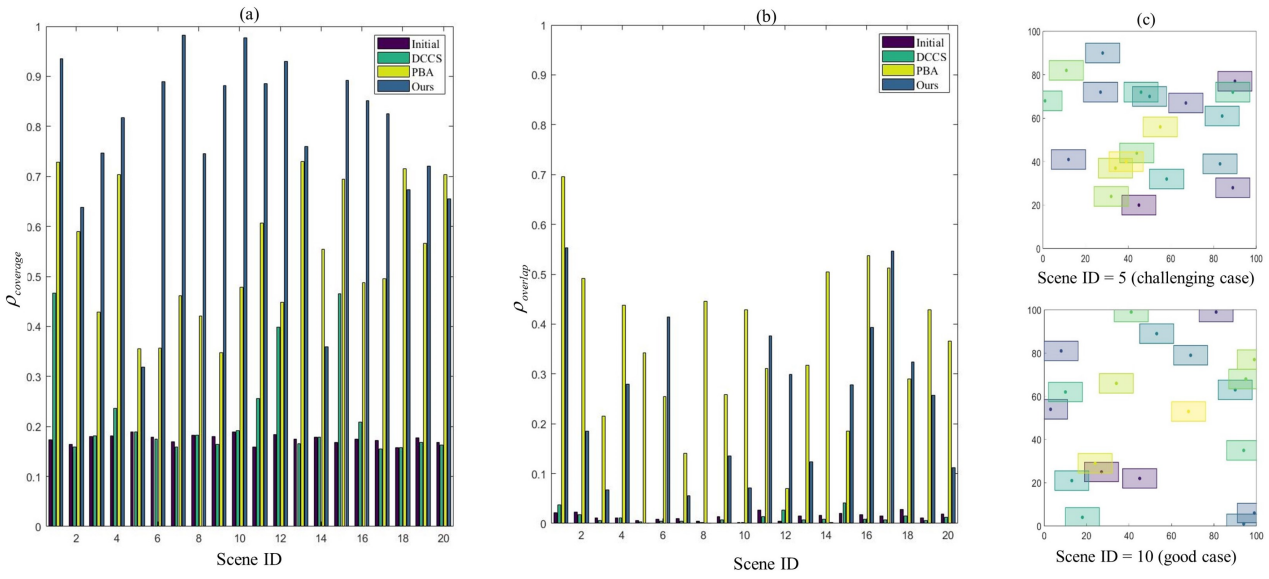

FIGURE 4 | Coverage and overlap performances with different camera positions across 20 randomly generated empty scenes. (a) Coverage ratio of different methods for each scene. (b) Overlap ratio of different methods for each scene. (c) Visualization of two representative scenes (Scene ID = 5 as a challenging case, and Scene ID = 10 as a good case). In each visualization, circles denote camera positions, and rectangles represent the corresponding camera coverage areas.

within a limited number of iterations. Nevertheless, it still produces competitive results.

We demonstrate the validity of three methods in scenes with buildings, as illustrated in Figure 5. According to Figure 5, our method also achieved the best performance in irregular-shape regions with buildings, the improvement of coverage performance, and decreases of overlap performance. However, the mean coverage ratio in Figure 5 was lower than that in Figure 4, which meant that it was challenging work to find the

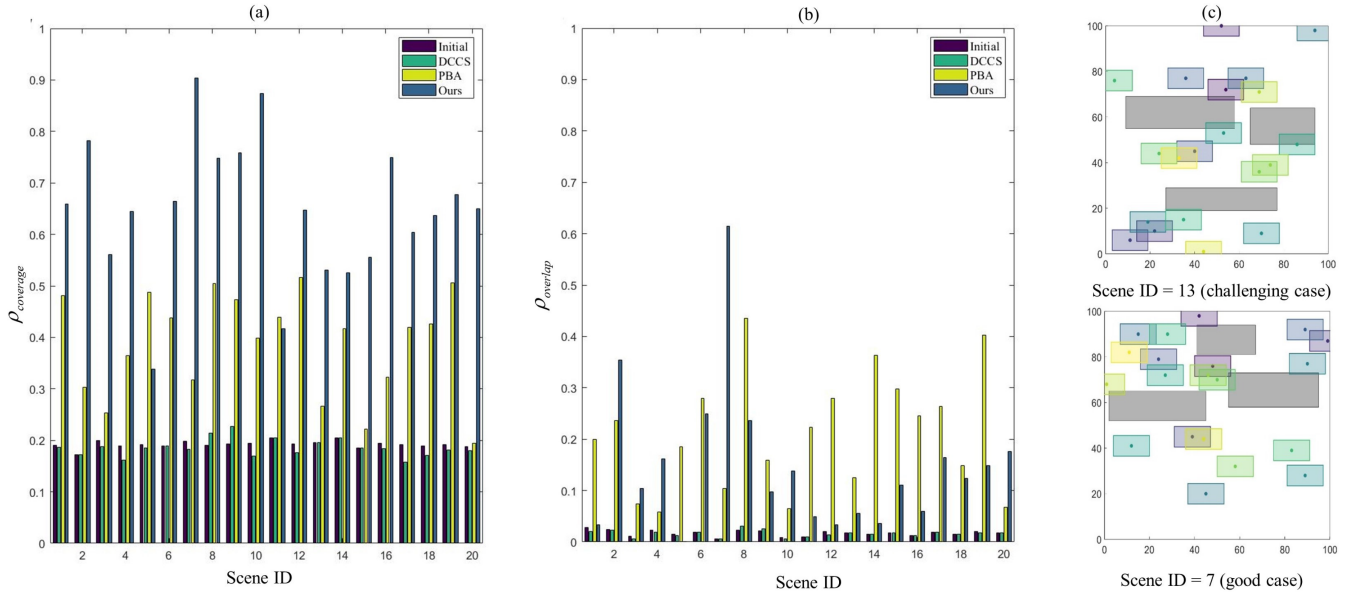

FIGURE 5 | Coverage and overlap performances with different camera positions across 20 randomly generated scenes with buildings. (a) Coverage ratio of different methods for each scene. (b) Overlap ratio of different methods for each scene. (c) Visualization of two representative scenes (Scene ID = 13 as a challenging case, and Scene ID = 7 as a good case). In each visualization, circles denote camera positions, and rectangles represent the corresponding camera coverage areas.

TABLE 2 | Results of coverage control with important objects.

Method	Coverage ratio ↑	Overlap ratio ↓	Number of successes ↑
Initial	17.63%	1.23%	1
DCCS [28]	37.44%	**8.96%**	5
PBA [31]	65.42%	40.11%	16
Ours	**84.42%**	42.16%	**18**

optimal camera network configuration in scenes with buildings. In Figure 5c, buildings in the 13th scene confined many cameras to a narrow zone, preventing effective coverage of other areas. Although, the 13th scene presents a challenging case, our method outperforms the other two approaches. Real-world camera placements are conducive to observation, so such a constrained layout would not occur in practice. We also conducted experiments in scenarios with important targets. Experimental results were shown in Table 2. "Number of Successes" in Table 2 represented the number that important targets were successfully covered in 20 scenes. The coverage ratio and the number of successful regions in our method outperformed those of other approaches. The overlap ratio achieved by our method was competitive with that of the PBA method. Therefore, our method achieved a relative balance between coverage performance and coverage for important objects. Comparing the results of our method and the PBA method, we found that our method had a higher coverage success rate of important objects, which verified the effectiveness of our method in scenes with important targets.

We conducted ablation experiments using the virtual repulsive force method (VPF-ACE) [11] and the proposed method to investigate the impact of the virtual potential field on the experimental results. To better highlight the differences in experimental results, we designed four additional simulation scenes: two with buildings and two with important targets. These scenes were not based on the parameters in Table 1, but constructed with reference to real-world scenarios. The experimental results are shown in Figure 6. The position and number of cameras were set to achieve the best visualization quality. The VFA-ACE method only considered the repulsion among cameras and lacked necessary attention to buildings and important targets. Our method pushed the viewpoints of the PTZ cameras to move away from buildings and scene edges, as depicted in Figure 6 a_3, b_3. Also, in scenes with important targets, our method updated the viewpoints of PTZ cameras to the positions close to the target centers, as presented in Figure 6 c_3, d_3. The experimental results showed that our method can realized a delicate balance between coverage performance and different coverage tasks. We also presented the experimental results of runtime experiments for scene coverage in Table D1 in Appendix D and discussed the findings.

4.2 | Experiments on the Pedestrian Coverage

In this subsection, we have constructed a simulated scene with $70m \times 70m$, which includes four buildings, eight PTZ cameras, and four pedestrians, as depicted in Figure 7a. All cameras are mounted at a height of 10 m, with a focal length of 50 mm and an image resolution of 1960×1080. The four pedestrians possess uniform physical characteristics, including a body height of 1.8 m and a shoulder width of 0.4 m. These individuals remain stationary within the scene. The initial orientation of the cameras is generated by our scene coverage control strategy. The pedestrian tracking experiment was conducted in Figure D1 in Appendix E.

Three distinct pedestrian coverage control methods of VSNTMT, CSDVSN, and our method were conducted in the first scene. Initially, we employed our scene coverage control strategy to

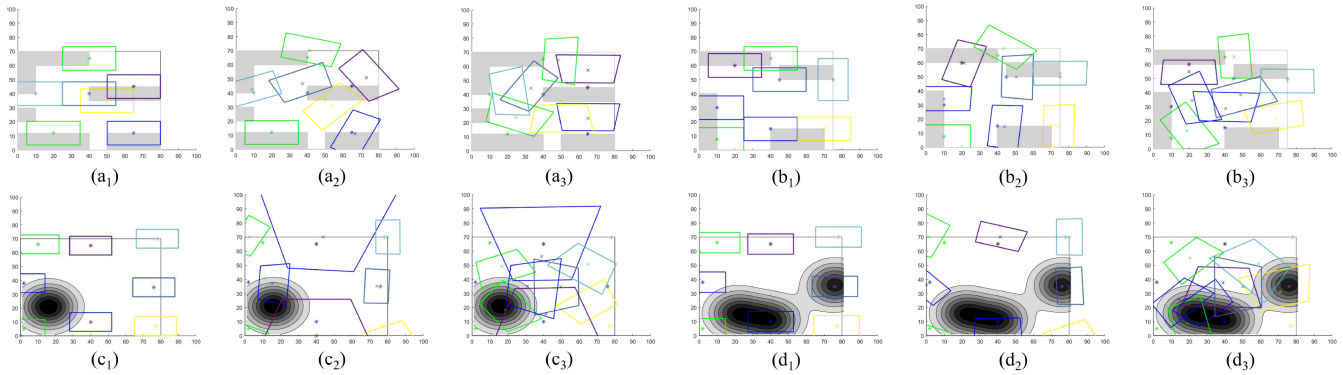

FIGURE 6 | Visual results of different methods in simulation environments. (a, b) represent scenes with buildings. (c, d) represent scenes with important objects. (1) is the initial configuration of the PTZ camera network by applying only the virtual repulsive force, and (3) is that by our method. "*" is the representative of a camera position, and "x" is the representative of a camera viewpoint. The shadow area represents the important objects, and the gray blocks represent buildings.

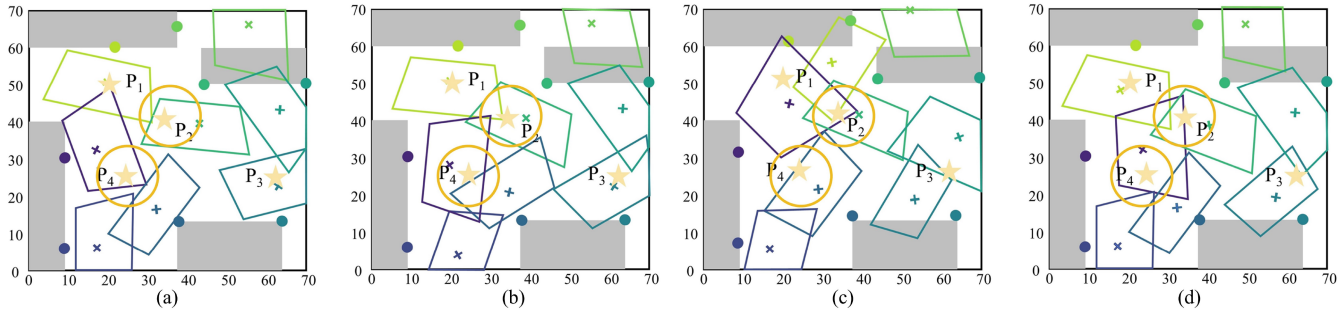

FIGURE 7 | Visual coverage results in the pedestrian coverage experiment. Gray areas represent buildings, circles indicate camera positions, "×" marks denote camera viewpoints, stars correspond to pedestrian targets, and trapezoids depict the camera field of view. (a–d) show the results of different methods: (a) our scene coverage control scheme, (b) our pedestrian coverage control scheme, (c) VSNMT, and (d) CSDVSN. The orange circles highlight significant differences across methods.

maximize the scene coverage, and the results of this experiment were depicted in Figure 7a. It was observed that pedestrians P_2 and P_4 were situated at the periphery of the image. We have also visualized these pedestrians in cameras in the first row of Figure 8. Within our proposed methodology, the perceived quality of P_2 and P_4 was inferior to that of P_1 and P_3. Figure 7b illustrated the results obtained after implementing our novel pedestrian coverage control algorithm, where the parameters α_g, α_q, and α_p were all set to 1. The second row of Figure 8 represented the images of the four individuals captured by the camera under the conditions of Figure 7b. The experimental results demonstrated that after optimization with our method, there was a general enhancement in the perceived quality of the P_2 and P_4. The size and localization of the P_2 and P_4 in the images of the second row of Figure 8 were more competitive to pedestrian analysis compared to the first row. The experimental results after optimization with the CSDVSN method were shown in Figure 7c and the third row of Figure 8. The CSDVSN method, designed to identify camera positions that capture pedestrians' facial features, often failed to maintain target visibility within a limited number of iterations. As a result, both coverage and observation quality of pedestrian targets deteriorated. Lastly, the results of optimization with the VSNTMT method were presented in Figure 7d and the fourth row of Figure 8. In this case, the camera for P_4 was updated to a better position. However, the perception quality for P_4 declined, suggesting that the VSNTMT method may not reflect appropriate perception scores. This highlighted the importance of tailoring the optimization strategies to the specific characteristics and requirements of the scene. The experimental results regarding runtime performance for pedestrian coverage were presented in Table D2 (Appendix D). The results indicated that enhancing the computational quality of pedestrian perception did not impose additional temporal overhead.

To evaluate the coverage performance of our method in crowded pedestrian environments, We constructed a simulated 50 m × 50 m scene containing either 50 or 100 pedestrians and 10 PTZ cameras, as shown in Figure 9. All cameras were randomly placed within the scene, each mounted at a height of 10 m, with a focal length of 50 mm and a resolution of 1960 × 1080. Experimental results show that after 20 iterations, cameras concentrate coverage on areas with high pedestrian density. After 50 iterations, cameras achieve complete coverage of both the scene and all pedestrians. After 100 iterations, the scene coverage is similar to that achieved after 50 iterations, indicating that our method obtains a stable scene coverage result within 50 iterations. In a 100-pedestrian scenario, our method produces stable coverage results within 11.30 s. Considering hardware wear and the

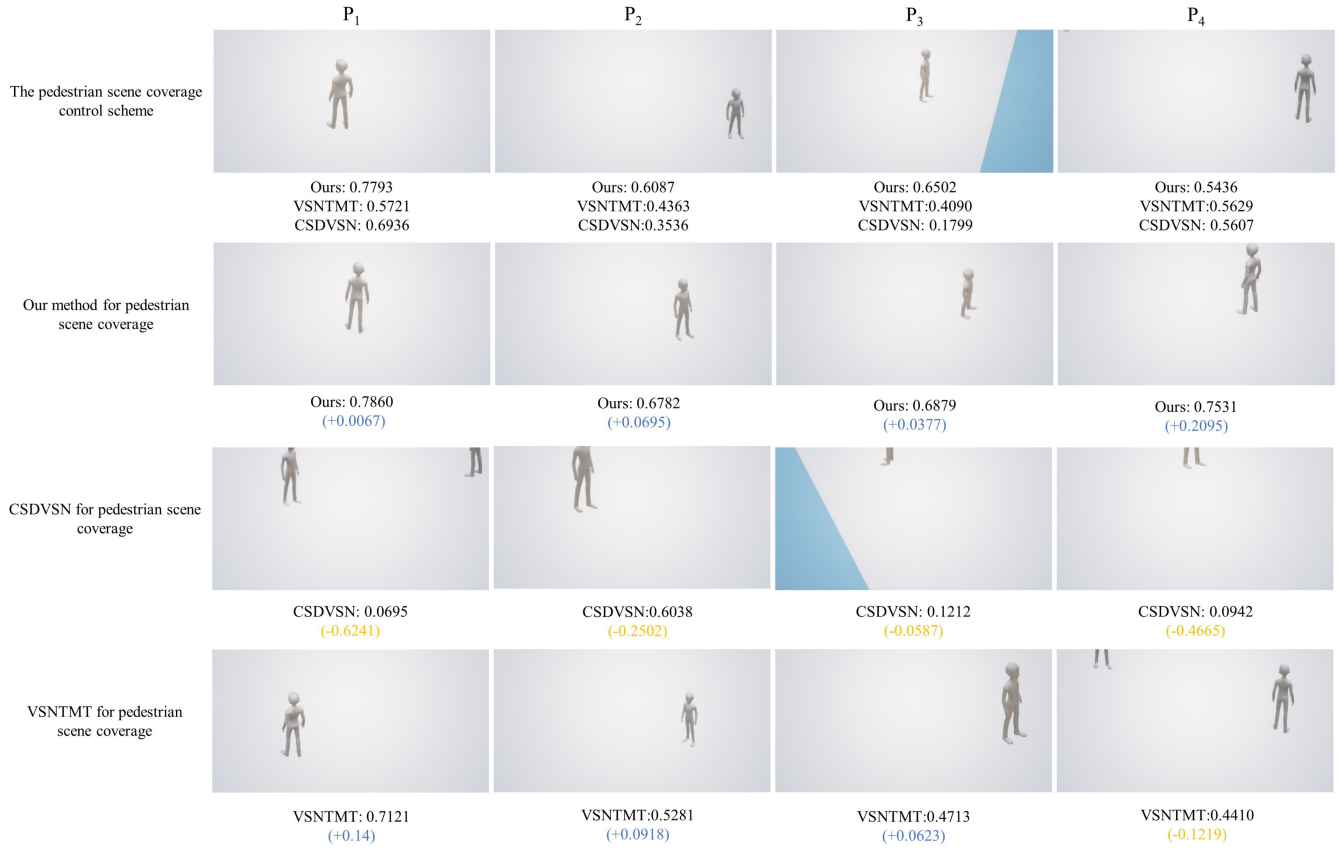

FIGURE 8 | The camera image including pedestrian from different pedestrian coverage strategy. The blue region denotes the background beyond the simulation scenario. The blue and yellow texts denote the rise and fall of perception quality.

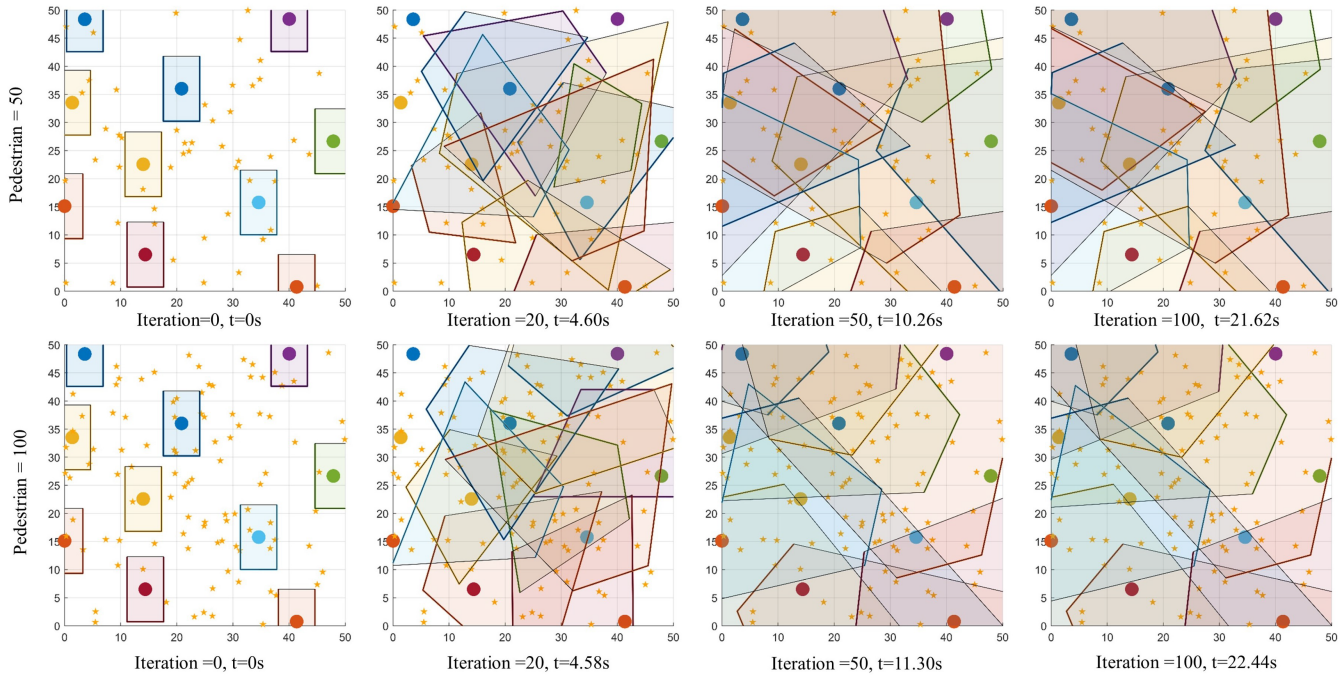

FIGURE 9 | Visualization of the coverage results under different pedestrian densities (50 and 100 pedestrians). Each row corresponds to a different pedestrian density. From left to right, the subplots show iteration = 0, 20, 50, and 100, along with the corresponding elapsed time t. Circles denote camera positions, rectangles and trapezoids represent the coverage regions of the cameras, and star markers indicate pedestrian locations.

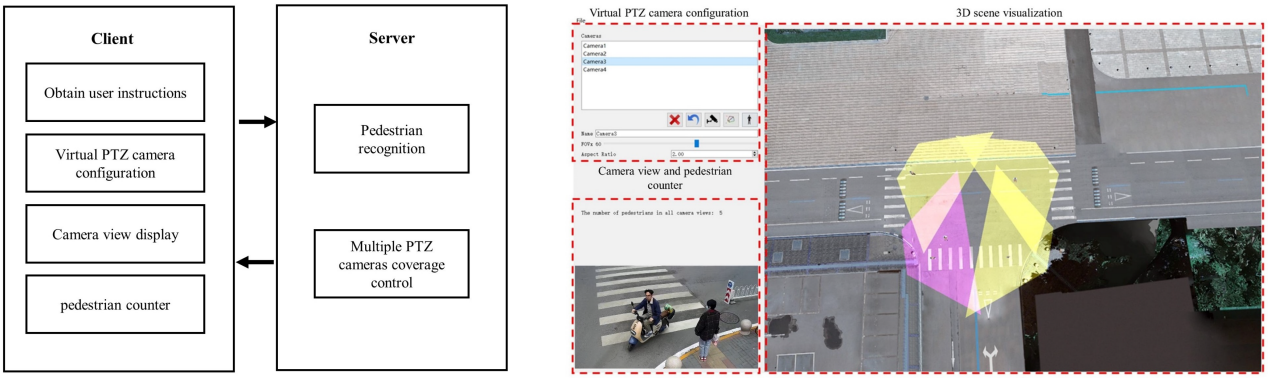

FIGURE 10 | The pedestrian scene coverage control tool. The yellow and red trapezoids represent the coverage regions of the cameras. The red trapezoid indicates the selected camera and its corresponding coverage region. (a) Client-server architecture of the system. (b) Tool interface, including virtual PTZ camera configuration, camera view, and 3D scene visualization.

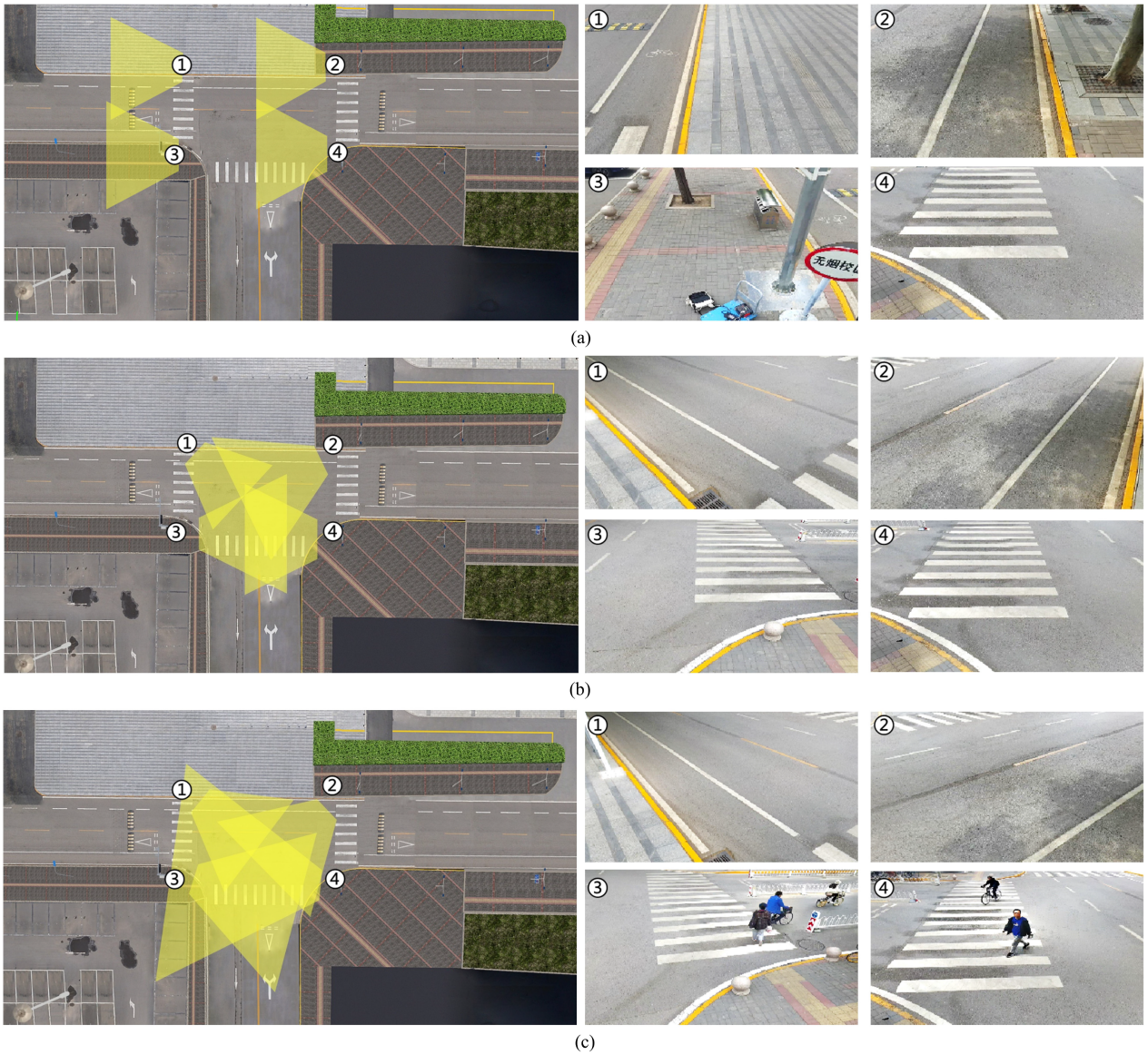

FIGURE 11 | Visual coverage results in the real scene. Numbers ①, ②, ③, and ④ indicate the camera positions. The yellow trapezoids represent the coverage regions of the cameras. (a) Initial coverage. (b) scene coverage. (c) pedestrian coverage.

need for stable monitoring, PTZ camera networks are ill-suited to frequent parameter updates. The time cost of our method fully satisfies the requirements for dynamic scene coverage.

Figure 9 demonstrates that varying the number of people has a negligible impact on our method's time cost. The difference in processing time between 50-pedestrian and 100-pedestrian scenes is only 1.04 s. This indicates that dense crowds do not introduce latency issues. These results demonstrate strong performance of our method in crowded pedestrian environments.

4.3 | Experiments on Pedestrian Scene Coverage in Real Scenes

To validate our method in real scenes, A pedestrian scene coverage control tool was designed, as demonstrated in Figure 10. The tool followed a client-server architecture. On the server side, we ran a pedestrian recognition algorithm alongside our multi-PTZ camera coverage control method. The client provided a user interface for obtaining user instructions, loading a 3D scene, configuring virtual PTZ cameras, displaying camera views and counting pedestrians. We captured PTZ camera images in real scenes and stitched them into panoramic images. We extracted the corresponding views from panoramic images based on camera parameters.

We installed four PTZ cameras at a real intersection, each mounted at a height of 3 m, as depicted in Figure 11a. In this intersection scenario, there are significant lighting variations among camera views due to differences in camera positions and orientations. Next, we built a 3D model of the same intersection and positioned virtual cameras at corresponding locations. We linked each real-world panorama to its matching virtual camera.

We conducted scene and pedestrian coverage experiments, with results shown in Figure 11b,c. For scene coverage, our approach achieves complete coverage of the intersection and a lower overlap rate compared to pedestrian coverage. Under pedestrian coverage, the coverage results highlight the crosswalk. These results demonstrate the effectiveness of our approach in real-world scenarios.

5 | Conclusion

In this work, we introduced the concept of perception quality for scenes, cameras, and pedestrians into the VPF-based method, quantifying the importance level of surveillance in different regions, cameras, and pedestrians. We proposed a universal perception quality measure framework for scenes, cameras, and pedestrians. We also developed a novel coverage control scheme based on the perceptive quality-based virtual potential field to optimize the parameters of a PTZ camera network. The scheme dynamically selected the optimal camera to monitor the target pedestrian with high resolution. Our experiments demonstrated that this coverage control scheme significantly outperformed other state-of-the-art methods, achieving remarkable improvements in the surveillance performance of scenes and pedestrians. Considering real-world surveillance applications, our focus is on the centralized control approach and fixed camera positions. These constraints make our method more suitable for scene coverage with stationary PTZ cameras. Collaborative coverage with mobile cameras, such as drone swarms, remains unexplored. In terms of system scalability and bandwidth management, the growing number of PTZ cameras brings a potential bottleneck for transmitting video streams to the central computing node. Future research will explore the coverage control for pedestrian scenes using distributed mobile PTZ cameras.

Author Contributions

Liangliang Cai: conceptualization (lead), methodology, investigation, formal analysis, writing – original draft (lead), writing – review and editing (lead), visualization. **Zhuocheng Liu:** software, validation. **Zhong Zhou:** conceptualization (supporting), resources, data curation, writing – original draft (supporting), writing – review and editing (supporting), supervision, funding acquisition.

Funding

This work was supported by the Science and Technology Project of Hainan Provincial Department of Transportation (Grant No. HNJTT-KXC-2024-3-22-02) and the National Natural Science Foundation of China (Grant No. 62272018).

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. B. Rinner and W. Wolf, "An Introduction to Distributed Smart Cameras," *Proceedings of the IEEE* 96, no. 10 (2008): 1565–1575.

2. C. Piciarelli, L. Esterle, A. Khan, B. Rinner, and G. L. Foresti, "Dynamic Reconfiguration in Camera Networks: A Short Survey," *IEEE Transactions on Circuits and Systems for Video Technology* 26, no. 5 (2015): 965–977.

3. P. R. Lewis, L. Esterle, A. Chandra, B. Rinner, J. Torresen, and X. Yao, "Static, Dynamic, and Adaptive Heterogeneity in Distributed Smart Camera Networks," *ACM Transactions on Autonomous and Adaptive Systems* 10, no. 2 (2015): 1–30.

4. D. Pianini, F. Pettinari, R. Casadei, and L. Esterle, "A Collective Adaptive Approach to Decentralised k-Coverage in Multi-Robot Systems," *ACM Transactions on Autonomous and Adaptive Systems* 17, no. 1–2 (2022): 1–39.

5. J. O'rourke and others, *Art Gallery Theorems and Algorithms*, vol. 57 (Oxford University Press, 1987).

6. U. M. Erdem and S. Sclaroff, "Automated Camera Layout to Satisfy Task-Specific and Floor Plan-Specific Coverage Requirements," *Computer Vision and Image Understanding* 103, no. 3 (2006): 156–169.

7. S. Hoseini, M. Dehghan, and H. Pedram, "Sensor Selection Using Circular Target Model in Visual Sensor Networks," in *2011 International Symposium on Computer Networks and Distributed Systems (CNDS)* (IEEE, 2011), 164–169.

8. M. Hosseini, M. Dehghan, and H. Pedram, "Sensor Selection and Configuration in Visual Sensor Networks," in *6th International Symposium on Telecommunications (IST)* (IEEE, 2012), 697–702.

9. J. Liu, S. Sridharan, C. Fookes, and T. Wark, "Optimal Camera Planning Under Versatile User Constraints in Multi-Camera Image Processing Systems," *IEEE Transactions on Image Processing* 23, no. 1 (2013): 171–184.

10. N. Conci and L. Lizzi, "Camera Placement Using Particle Swarm Optimization in Visual Surveillance Applications," in *2009 16th IEEE International Conference on Image Processing (ICIP)* (IEEE, 2009), 3485–3488.

11. H. Ma, X. Zhang, and A. Ming, "A Coverage-Enhancing Method for 3D Directional Sensor Networks," in *IEEE INFOCOM 2009* (IEEE, 2009), 2791–2795.

12. L. Yupeng, J. Peng, L. Hongchao, and J. Jingqi, "A Virtual Potential Field Based Coverage-Enhancing Algorithm for 3D Directional Sensor Networks," in *2012 6th International Conference on New Trends in Information Science, Service Science and Data Mining (ISSDM2012)* (IEEE, 2012), 225–230.

13. X. Ma and J. Kang, "A Multi-Detecting Point Based on Virtual Force-Directed Particle Swarm Optimization Algorithm for Coverage Enhancement in Directional Sensor Networks," in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)* (IEEE, 2018), 1195–1200.

14. K. Laventall and J. Cortés, "Coverage Control by Robotic Networks With Limited-Range Anisotropic Sensory," in *2008 American Control Conference* (IEEE, 2008), 2666–2671.

15. B. Hexsel, N. Chakraborty, and K. Sycara, "Coverage Control for Mobile Anisotropic Sensor Networks," in *2011 IEEE International Conference on Robotics and Automation* (IEEE, 2011), 2878–2885.

16. Y. Kantaros and M. M. Zavlanos, "Distributed Communication-Aware Coverage Control by Mobile Sensor Networks," *Automatica* 63 (2016): 209–220.

17. Y. C. Wang and D. Y. Chen, "Cooperative Monitoring Scheduling of Ptz Cameras With Splitting Vision and Its Implementation for Security Surveillance," in *Proceedings of the International Conference on Advanced Technology Innovation* (AITI, 2019), 1–12.

18. D. Tao, H. D. Ma, and L. Liu, "A Virtual Potential Field Based Coverage-Enhancing Algorithm for Directional Sensor Networks," *Journal of Software* 18, no. 5 (2007): 1152–1163.

19. W. Li, H. Liu, H. Tang, and P. Wang, "Multi-Hypothesis Representation Learning for Transformer-Based 3D Human Pose Estimation," *Pattern Recognition* 141 (2023): 109631.

20. X. Wang, Z. Sun, A. Chehri, G. Jeon, and Y. Song, "A Novel Attention-Driven Framework for Unsupervised Pedestrian Re-Identification With Clustering Optimization," *Pattern Recognition* 146 (2024): 110045.

21. S. Jiang, J. Cai, H. Zhang, Y. Liu, and Q. Liu, "Compare and Focus: Multi-Scale View Aggregation for Crowd Counting," *IEEE Transactions on Intelligent Transportation Systems* 25 (2024): 13231–13239.

22. C. Ding, J. H. Bappy, J. A. Farrell, and A. K. Roy-Chowdhury, "Opportunistic Image Acquisition of Individual and Group Activities in a Distributed Camera Network," *IEEE Transactions on Circuits and Systems for Video Technology* 27, no. 3 (2016): 664–672.

23. C. F. D. Saragih, F. M. T. R. Kinasih, C. Machbub, P. H. Rusmin, and A. S. Rohman, "Visual Servo Application Using Model Predictive Control (Mpc) Method on Pan-Tilt Camera Platform," in *2019 6th International Conference on Instrumentation, Control, and Automation (ICA)* (IEEE, 2019), 1–7.

24. C. Ding, B. Song, A. Morye, J. A. Farrell, and A. K. Roy-Chowdhury, "Collaborative Sensing in a Distributed PTZ Camera Network," *IEEE Transactions on Image Processing* 21, no. 7 (2012): 3282–3295.

25. J. Giordano, M. Lazzaretto, G. Michieletto, and A. Cenedese, "Visual Sensor Networks for Indoor Real-Time Surveillance and Tracking of Multiple Targets," *Sensors* 22, no. 7 (2022): 2661.

26. L. Cai, H. Ma, Z. Liu, Z. Li, and Z. Zhou, "Coverage Control for PTZ Camera Networks Using Scene Potential Map," in *2022 IEEE International Conference on Multimedia and Expo (ICME)* (IEEE, 2022), 1–6.

27. O. Arslan, H. Min, and D. E. Koditschek, "Voronoi-Based Coverage Control of Pan/Tilt/Zoom Camera Networks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2018), 5062–5069.

28. Q. An and Y. Shen, "Distributed Coverage Control for Mobile Camera Sensor Networks With Anisotropic Perception," *IEEE Sensors Journal* 21 (2021): 16264–16274.

29. D. Panagou, D. M. Stipanović, and P. G. Voulgaris, "Distributed Dynamic Coverage and Avoidance Control Under Anisotropic Sensing," *IEEE Transactions on Control of Network Systems* 4, no. 4 (2016): 850–862.

30. X. Zhang, Y. Ling, Y. Yang, C. Chu, and Z. Zhou, "Center-Point-Pair Detection and Context-Aware Re-Identification for End-To-End Multi-Object Tracking," *Neurocomputing* 524 (2023): 17–30.

31. S. V. T. Tran, D. Lee, H. C. Pham, L. H. Dang, C. Park, and U. K. Lee, "Leveraging BIM for Enhanced Camera Allocation Planning at Construction Job Sites: A Voxel-Based Site Coverage and Overlapping Analysis," *Buildings* 14, no. 6 (2024): 1880.

Supporting Information

Additional supporting information can be found online in the Supporting Information section. **Video S1.** Supporting Information.

Appendix A

Virtual Potential Field

The VPF method assumes a PTZ camera as a sector rotating around the center of a circle. The circle center represents the position of the camera, and the sector represents the orientation and FoV of the camera. To simplify the sector rotation model, the VPF method introduces the "centroid." The circle motion of the sector is approximated as the centroid of the sector running around the PTZ camera. Each PTZ camera has only one centroid. The VPF method transforms the PTZ camera network coverage control problem into a solvable uniform distribution problem of centroids.

The VPF method assumes that repulsive forces exist among the centroids. Under the influence of repulsion forces, adjacent centroids are driven apart, which minimizes overlap and enhances the overall coverage efficiency. This process leads to a more thorough and effective surveillance of the monitored area, thereby improving the coverage performance of the PTZ camera network. In the VPF method, each centroid is subject to repulsive forces exerted by one or more neighboring centroids. The repulsion force is computed as follows:

$$F_{qi} = \sum_{j \in N_i} \frac{k_q}{d_{ij}^2} \qquad i \in \{1, 2, \ldots, N\} \tag{A1}$$

$$F'_{qi} = F_{qi} * cos\theta \tag{A2}$$

where k_q denotes the empirical constant and d_{ij} the distance between the ith camera as well as the jth camera. F'_q denotes the component of force F_q along the tangent to the circle. After being subjected to the repulsion of F'_q, the centroid updates its orientation with a fixed angle $\triangle\theta$ each time and gradually arrives in the optimal position through multiple iterations. When the centroid is subjected to $F'_q = 0$, it arrives at the ideal orientation. Besides, when the centroid vibrates reciprocally around the PTZ camera, F'_q denotes very small. We set the force threshold to ϵ. When $F'_q < \epsilon$, the centroid reaches a steady state. The algorithm of the VPF method is shown as in Algorithm 3. In order to improve the algorithm efficiency, the maximum iteration N_t is set.

Traditional VPF algorithms primarily focus on the information from PTZ cameras, neglecting the utilization of scene and pedestrian information.

The rotation model diverges significantly from the actual camera model, leading to results still exhibiting considerable discrepancies in real-world scenarios.

ALGORITHM 3 | VPF.

Input: The scene $S = \{C, X\}$, sampling point set $X = \{X_1, X_2, \ldots, X_{nx}\}, X_i = \{x_i, y_i\}$, PTZ cameras $C = \{C_1, C_2, \ldots, C_{nc}\}$, $C_i = \{x_i, y_i\}$, the iteration time $t = 0$, the pan angle set $\theta = \{\theta_1, \theta_2, \ldots, \theta_{nc}\}$ of PTZ camera.

Output: The extra parameter set $\hat{\theta}$ of PTZ cameras

```
while each ||F'_qi|| > ε and t < N_t do
    t ← t + 1
    for i in [1 : nc] do
        F_qi ← 0
        for j in N_i do
            compute F_qij ← k_q / d_ij^2
            F_qi ← F_qi + F_qij
        end for
        compute F'_qi ← F_qi * cosθ_i
        if F'_qi > 0 then
            θ_i ← θ_i + △θ
        else if F'_qi < 0 then
            θ_i ← θ_i − △θ
        else
            θ_i ← θ_i
        end if
        recompute the centroid of i-th camera.
    end for
end while
```

Appendix B

Visualization of Perception Quality Measure for Different Objects

We provided visual examples of the scene and camera perception quality measures, respectively, as shown in Figures B1 and B2. In Figure B1, a scene with two buildings and two important target regions was presented. Two important target region included an accident scene and an intersection, respectively. We calculated Equation (8), Equation (11), and Equation (12) based on the defined scene and used the Viridis color map to visualize the results in Figure B1d–f. In Figure B1d, the scene layout perception quality Q_{sb} of the buildings and edges of the scene was close to zero (the blue region), while Q_{sb} increased gradually as it moved away from these regions. The scene layout perception quality can guide the camera network to avoid buildings and scene edge regions. In Figure B1e, the regions corresponding to the accident and intersection were marked as high importance level by the important target perceptual quality. Cameras can move toward regions with the high importance level. In Figure B1f, the scene perception quality Q_s combined importance levels of the scene layout and important targets.

We also visualized the camera perspective, resolution and perception quality by Equation (15), Equation (16), and Equation (17). The visual results of the Q_{ct} were shown in Figure B2a. The camera perspective quality emphasized the central area of the image, making it more important than the edges, which helped the target focus on the center. Visual results of the Q_{cr} was depicted in Figure B2b. The camera resolution quality maintained the target at an appropriate distance from the camera to ensure optimal resolution. A distance that was too close or too far will negatively affect the resolution quality of the target. The visual result of Q_c was illustrated in Figure B2c.

We discussed the influence of hyperparameters k_p, ρ_0, and d_0 and visualized results of the pedestrian perception quality over ρ_p and d_p. The experiment comparison was presented in Figure B3.

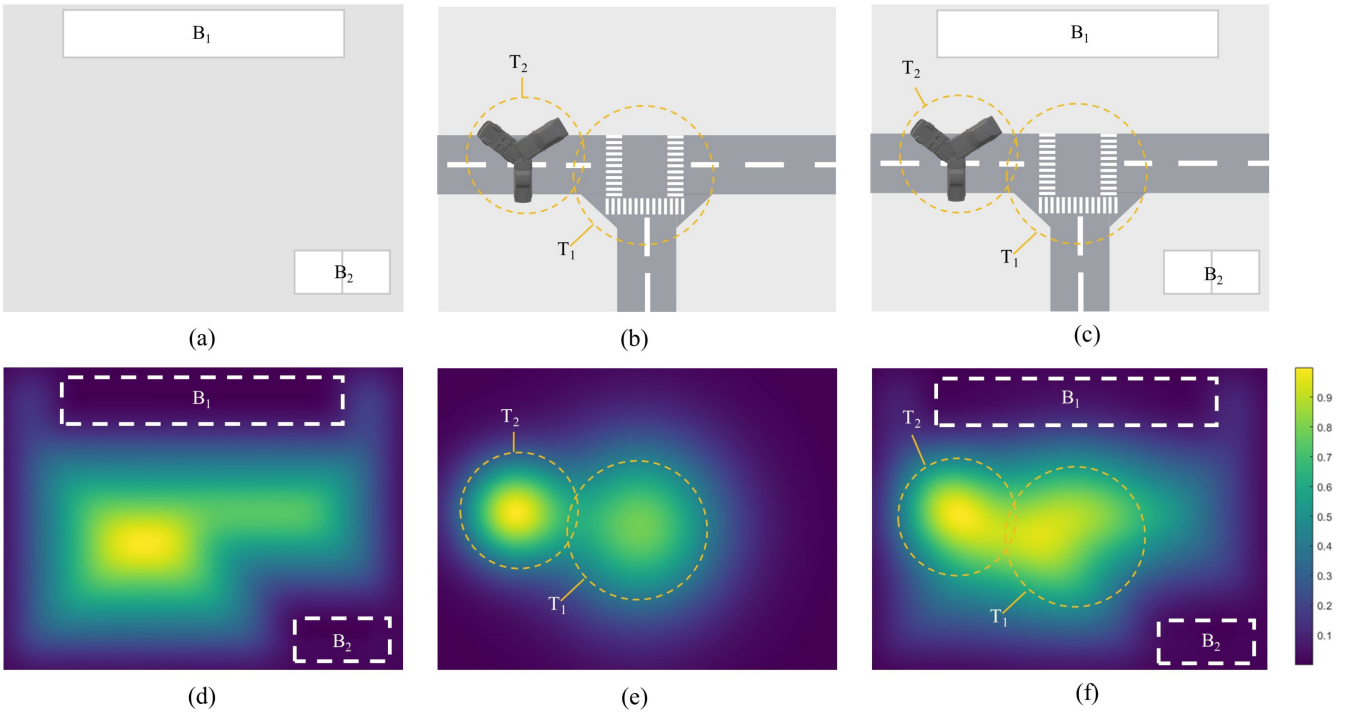

FIGURE B1 | The visualization of scene perception quality. (a) denotes the scene with buildings, (b) represents the scene with important regions, and (c) indicates the scene with both buildings and important regions. (d), (e), and (f) denote different scene visualizations of scene perception quality. The blue region represents the region with low scene perception quality, and the yellow one denotes the region with high quality.

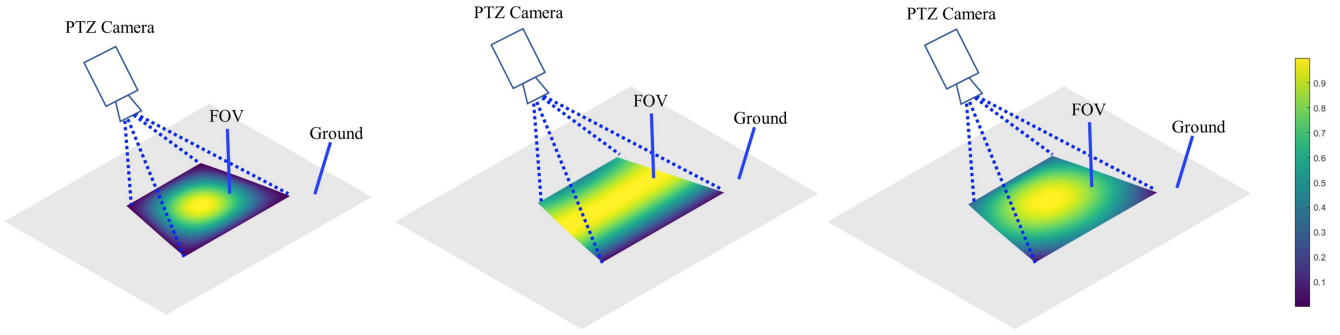

FIGURE B2 | The visualization of camera perception quality. The blue region represents the low importance level and the yellow one denotes the high level.

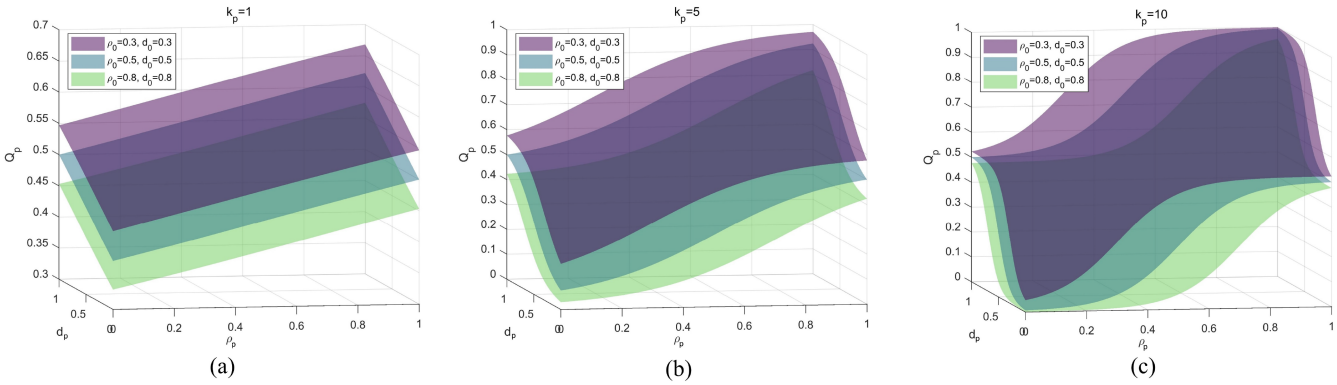

FIGURE B3 | Comparison of pedestrian perception quality with different hyperparameters.

Appendix C

Influence of Hyperparameters

We discussed the influence of hyperparameters α_g and α_q on experiment results during the optimization process. Specifically, α_g is associated with the hyperparameter of F_g, while α_q pertains to F_q. To rigorously test the effects of these hyperparameters, we constructed three distinct scenarios: an empty scene, a scene with buildings, and a scene containing significant targets. Each scenario was randomly generated 20 times, and the average coverage and overlap rates from these experiments were calculated. In Figure C1, we assigned the 0.5, 1, 2, 3, and 4 to α_g and α_q. In the empty scene and the scene with buildings, the results indicated that when the value of α_g exceeded that of α_q, or vice versa, our method outperformed the case where α_g equaled α_q in terms of both coverage and camera overlap rates. This phenomenon denoted that when the hyperparameters of F_g and F_q were close, they competed considerably during the optimization process, decreasing both scene coverage and camera overlap. Conversely, when one hyperparameter dominated, it can achieve high coverage rates but at the expense of increased camera overlap. In the scene with targets, as α_g increased, the coverage rate becomed larger and the overlap rate becomed smaller. Given our objective of maximizing scene coverage while minimizing camera overlap, we identified the parameter values encircled in orange in Figure C1 as the optimal hyperparameter settings for different scenarios.

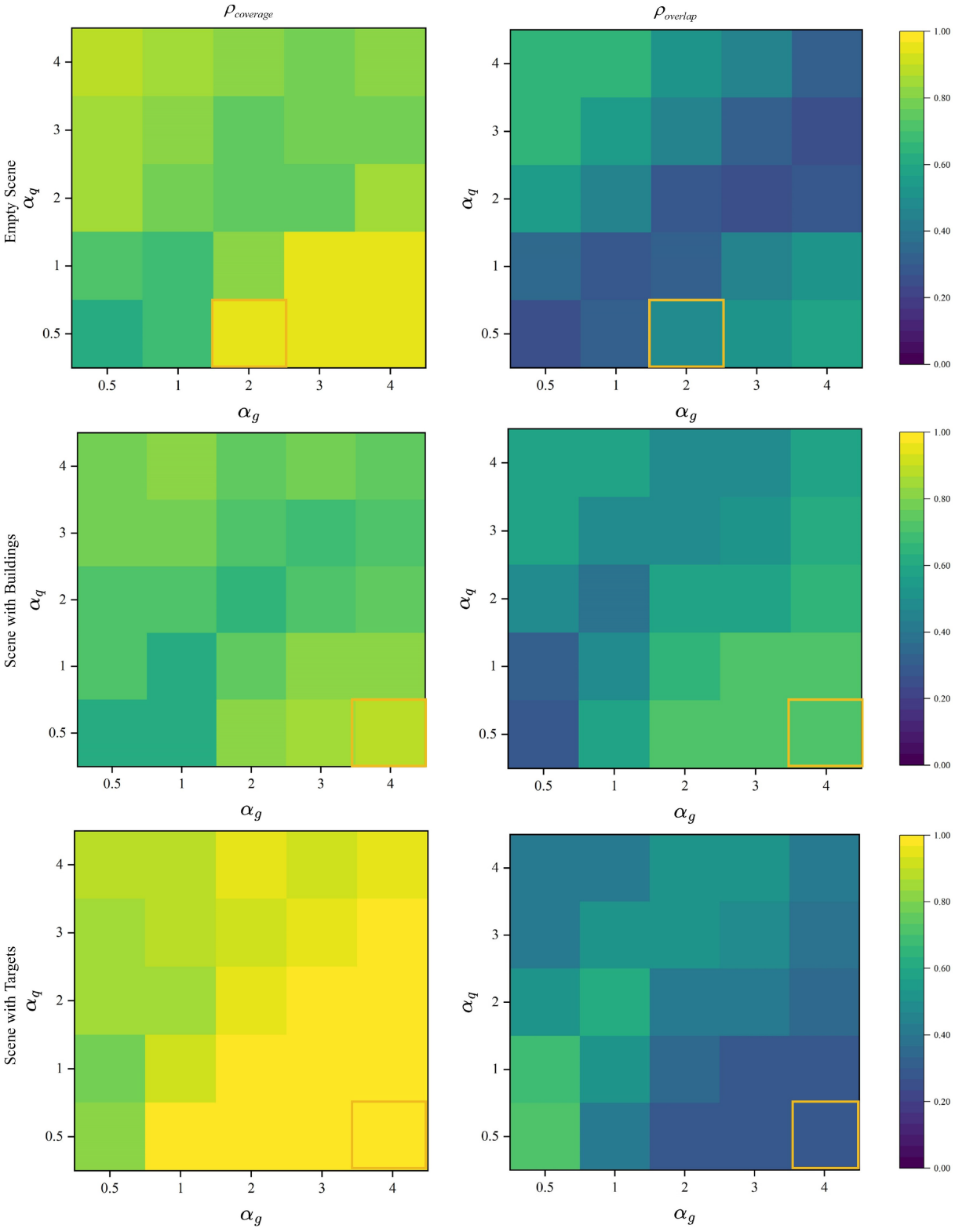

FIGURE C1 | Performance comparison of the our method with different α_g and α_q. Three distinct scenarios were designed for performance comparison. The orange box indicates the best hyperparameter setting.

Appendix D

Runtime Experiments

The runtime experiments of the scene coverage were illustrated in Table D1. For the scene coverage, we set the maximum number of iterations to 50. The sampling point intervals were chosen as 1, 2, 3, and 4 units, allowing us to explore the trade-off between the solution space and the computational efficiency. We experimented with 20 distinct scenarios, calculating the average runtime for each method. Our analysis demonstrated that the VPF-ACE method outperformed the others in terms of runtime, while the DCCS showed the slowest performance. Our proposed method demonstrated competitive performance comparable to the VPF-ACE method. It suggested that despite the additional computational overhead introduced by the perception quality calculations, our method did not suffer from a significant increase in execution time.

We also compared the temporal costs associated with three pedestrian coverage control methods. The sampling point intervals were set at 1, 2, 3, and 4. As shown in Table D2, the temporal expenditures for the three pedestrian coverage control algorithms were comparable to those of the scene coverage control method. This indicated that enhancing the computational quality of pedestrian perception did not impose additional overhead.

TABLE D1 | Runtime across different methods in scene coverage experiments.

		Time (s)			
Scene	**Method**	**Interval = 1**	**Interval = 2**	**Interval = 3**	**Interval = 4**
Empty scene	DCCS [28]	16.45	11.16	11.27	10.92
	VFA-ACE [11]	**9.73**	**8.53**	**8.90**	**7.32**
	Ours	9.56	8.87	8.91	7.54
Scene with buildings	DCCS [28]	17.75	12.56	12.17	11.62
	VFA-ACE [11]	**10.37**	**9.35**	**8.09**	**7.50**
	Ours	10.65	9.94	8.21	7.98
Scene with important target	DCCS [28]	17.15	12.65	11.71	11.13
	VFA-ACE [11]	**9.98**	**8.99**	**7.78**	**7.34**
	Ours	10.13	9.33	7.96	7.55

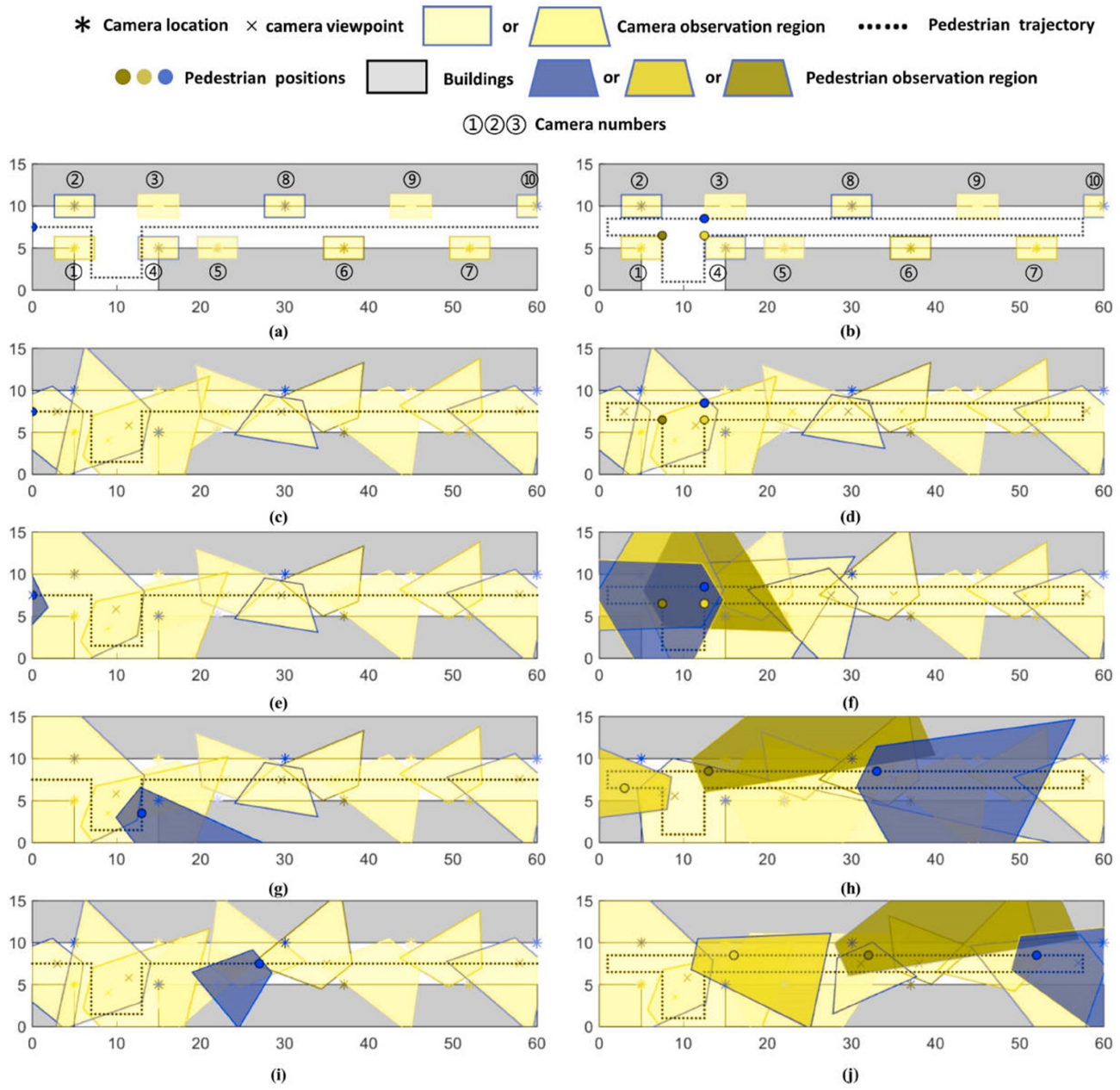

FIGURE D1 | The visualization of pedestrian tracking experiment.

TABLE D2 | The runtime for pedestrian coverage with different sampling point interval.

	Time (s)			
Method	**Interval = 1**	**Interval = 2**	**Interval = 3**	**Interval = 4**
Ours_scene	10.65	9.94	8.21	7.98
CSDVSN [24]	10.69	10.16	8.05	7.62
VSNTMT [25]	10.53	10.02	8.34	8.11
Ours_pedestrian	10.78	10.17	8.33	8.08

Appendix E

The Visualization of Pedestrian Tracking Experiments

In the pedestrian tracking experiments, we have established a simulated scenario with the size of $15m \times 60m$, as illustrated in Figure D1a. This scenario comprises three buildings, 10 PTZ cameras, and two sets of pedestrian trajectories. These PTZ cameras are arranged along both sides of the road, installed at a height of 3 m, with a focal length of 50 mm and an image resolution of 1280×720. Their initial orientation is downward, perpendicular to the ground. Two sets of pedestrian trajectories have been designed, as shown in Figure D1a,b. In Figure D1a, a single pedestrian (represented by a blue point) walks along a black dotted line in the center of the road from the left side to the right side. In Figure D1b, three pedestrians (marked with brown, yellow, and blue points) are placed at a crossroad within the simulated scenario, moving in a clockwise direction along a circular path. These four pedestrians share uniform physical characteristics: a height of 1.8 m, a shoulder width of 0.4 m, and a walking speed of 0.5 m/s.

In both single-person and three-people tracking scenarios, we captured scene coverage statuses at different moments (1, 40, and 80 s) using our pedestrian coverage control method, as shown in Figure D1e–j. The α_g, α_q, and α_p were set to 1, 1, and 5, ensuring that pedestrians will not lose track of them. As depicted in the figure, our approach achieved pedestrian tracking while maintaining coverage of the entire scene. When a pedestrian moved beyond the surveillance range of the current camera, our method seamlessly transitioned to the camera that offered the optimal observation quality for the pedestrian.