buildings

MDPI

Article

A Cost-Effective System for Indoor Three-Dimensional Occupant Positioning and Trajectory Reconstruction

Xiaomei Zhao *, Shuo Li, Zhan Zhao and Honggang Li

Shandong Key Laboratory of Intelligent Buildings Technology, School of Information and Electrical Engineering, Shandong Jianzhu University, Jinan 250101, China
* Correspondence: zhaoxiaomei20@sdjzu.edu.cn

Abstract: Accurate indoor occupancy information extraction plays a crucial role in building energy conservation. Vision-based methods are popularly used for occupancy information extraction because of their high accuracy. However, previous vision-based methods either only provide 2D occupancy information or require expensive equipment. In this paper, we propose a cost-effective indoor occupancy information extraction system that estimates occupant positions and trajectories in 3D using a single RGB camera. The proposed system provides an inverse proportional model to estimate the distance between a human head and the camera according to pixel-heights of human heads, eliminating the dependence on expensive depth sensors. The 3D position coordinates of human heads are calculated based on the above model. The proposed system also associates the 3D position coordinates of human heads with human tracking results by assigning the 3D coordinates of human heads to the corresponding human IDs from a tracking module, obtaining the 3D trajectory of each person. Experimental results demonstrate that the proposed system successfully calculates accurate 3D positions and trajectories of indoor occupants with only one surveillance camera. In conclusion, the proposed system is a low-cost and high-accuracy indoor occupancy information extraction system that has high potential in reducing building energy consumption.

Keywords: 3D occupant positioning; 3D trajectory reconstruction; surveillance camera; building energy saving

check for updates

Citation: Zhao, X.; Li, S.; Zhao, Z.; Li, H. A Cost-Effective System for Indoor Three-Dimensional Occupant Positioning and Trajectory Reconstruction. *Buildings* **2023**, *13*, 2832. https://doi.org/10.3390/buildings13112832

Academic Editor: Adrian Pitts

Received: 7 October 2023
Revised: 3 November 2023
Accepted: 9 November 2023
Published: 11 November 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

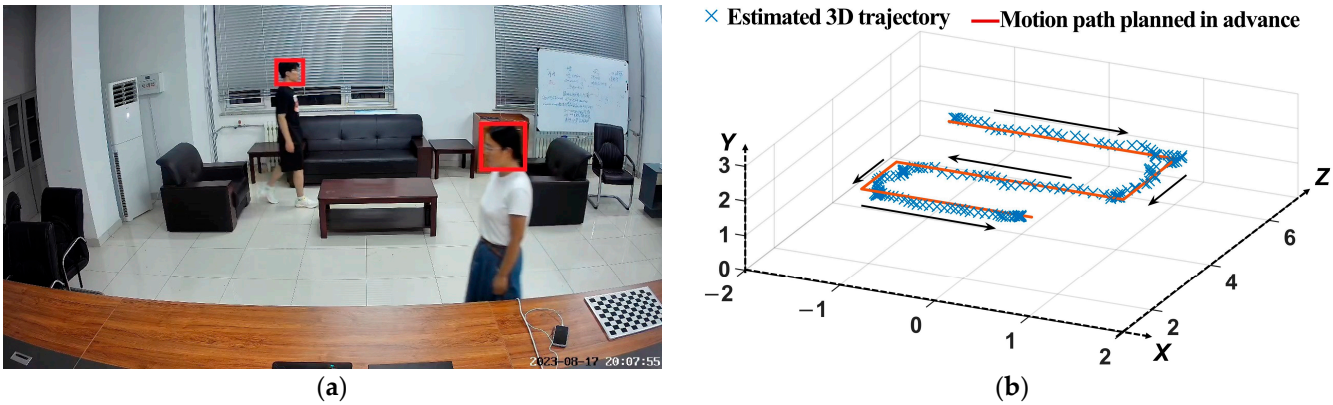
1. Introduction

Research has shown that individuals spend 85–90% of their time indoors [1]. To provide healthy and comfortable indoor environments for occupants, buildings consume about 40% of the worldwide energy [2]. However, the heating, ventilation, air-conditioning (HVAC), lighting, and plug load systems generally run fully on, regardless of whether the building occupancy rate reaches 100% [3], wasting large amounts of energy. Occupancy-based control systems have great potential to improve building energy efficiency [4,5]. For example, Pang et al. [6] showed that occupancy-driven HVAC operation could reduce building energy consumption by 20–45%; Zou et al. [7] showed that their occupancy-driven lighting system could reduce energy consumption by 93% compared to the static lighting control scheme; Tekler et al. [8] showed that their occupancy-driven plug load management system could reduce building energy consumption by 7.5%; and Yang et al. [9] showed that their occupant-centric stratum ventilation system could reduce energy consumption by 2.3–8.1%. Therefore, obtaining accurate occupancy information has clear benefits in reducing building energy use.

Occupancy information can be obtained using various kinds of sensors, such as passive infrared (PIR) [10], CO_2 [11,12], sound [13,14], radar [15], Wi-Fi [16–18], Bluetooth [19,20], ultra-wideband (UWB) [21,22], vision [23], and so on. While some methods use single sensor types, other researchers have studied combinations of multiple kinds of sensors. Tekler et al. [24] combined CO_2, Wi-Fi connected devices, and many other sensor data

types to predict occupancy. Tan et al. [25] combined temperature, humidity, image, and many other sensor data types to detect occupancy. Qaisar et al. [26] combined temperature, humidity, HVAC operations, and many other sensor data types to predict occupancy. However, sensors have varying capabilities and limitations. PIR sensors are insensitive to stationary occupants [27]. CO_2 sensors respond slowly and are easily influenced by many factors, such as unpredictable opening of windows [28]. Sound sensors can be triggered by sound from non-human sources [29]. Radar sensors can be obscured by large body movements [15]. Wi-Fi, Bluetooth, and UWB sensors require occupants' involvement, as with holding smartphones or UWB tags [30]. If occupants do not hold these devices, they cannot be detected. Compared to other sensors, vision sensors can capture richer data with greater accuracy [23]. In addition, because surveillance cameras have been widely installed in public places such as offices, schools, shopping malls, and so on, vision-based occupancy information extraction technologies can be easily integrated into existing vision surveillance systems at a very low cost. Therefore, vision-based occupancy information extraction has attracted the attention of many researchers.

Vision-based occupancy information extraction methods fall into two categories: 2D-based [23,31–33] and 3D-based methods [27,30,34,35]. The 2D-based methods generally use object detection networks such as YOLOX [36] or Faster RCNN [37] to detect human bodies or heads. An example of 2D-based human head detection results is shown in Figure 1a, showing the presence, number, and 2D positions of human heads in images. However, the 2D positions in images cannot provide the exact locations of humans in the 3D world and then cannot provide accurate occupant location information for intelligent building energy saving systems.



(**a**) (**b**)

Figure 1. Comparison of different types of vision-based occupancy information extraction methods: (**a**) a 2D-based method, with red boxes indicating the detected human heads; and (**b**) a 3D-based method, with the point set indicating the 3D trajectory of an occupant.

In recent years, many 3D-based methods have been proposed to address the above limitations. Wang et al. [34] fused the skeleton key points which are extracted from images captured by multiple cameras to reconstruct the 3D positions of humans. Wang et al. [27] utilized multiple RGB-D cameras to capture RGB images and depth images from different views, and used these captured RGB-D images to estimate human poses. The 3D positions of humans are reconstructed by fusing these estimated human poses. Zhou et al. [35] estimated the 3D positions of objects (including humans) by combining images captured by a surveillance camera and a building information model (BIM). However, these preceding 3D methods only estimated the 3D positions of occupants, not their 3D trajectories. Dai et al. [30] proposed an indoor 3D human trajectory reconstruction method combining the video captured by a monocular surveillance camera and a static point cloud map. The static point cloud map is built in advance by a LiDAR-based simultaneous localization and mapping (SLAM) algorithm. However, this method's dependence on a point cloud map increases its implementation cost.

In this paper, we propose a cost-effective indoor 3D occupant positioning and trajectory reconstruction system. This system does not need to build any large models, such as BIM and point cloud maps, in advance and requires no special cameras such as RGB-D cameras. It only uses a monocular customer surveillance camera to calculate 3D positions and reconstruct 3D trajectories of occupants. Therefore, the proposed system can be easily integrated into existing vision surveillance systems at a very low cost to provide accurate occupancy information for intelligent building energy management systems.

The main contributions of this paper are as follows:

(1) We propose an inverse proportional model to estimate the distance between human heads and the camera in the direction of the camera optical axis according to pixel-heights of human heads. With the help of this model, the 3D position coordinates of human heads can be calculated based on a single RGB camera. Compared with previous 3D positioning methods, our proposed method is significantly more cost-effective.
(2) We propose a 3D occupant trajectory reconstruction method that associates the 3D position coordinates of human heads with human tracking results according to the degree of overlap between binary masks of human heads and human bodies. This proposed method takes advantage of both the low cost of our 3D positioning method and the stability of human body tracking.
(3) We perform experiments on both 3D occupant positioning and 3D occupant trajectory reconstruction datasets. Experimental results show that our proposed system successfully calculates accurate 3D position coordinates and 3D trajectories of indoor occupants with only one RGB camera, demonstrating the effectiveness of the proposed system.

2. Methods

2.1. Overview

The flowchart of our proposed cost-effective system for indoor 3D occupant positioning and trajectory reconstruction is shown in Figure 2. As shown in this figure, our system contains seven modules: video capture module, camera calibration module, distortion correction module, instance segmentation module, 3D coordinates calculation module, tracking module, and 3D trajectory generation module.

The video capture module receives input video from a single surveillance camera. The camera calibration module calibrates the surveillance camera by the camera calibration function of OpenCV and a calibration board. The distortion correction module uses the distortion parameters calculated by the camera calibration module to remove both radial and tangential distortions. The instance segmentation module segments each instance of human body and human head. The 3D coordinates calculation module calculates the 3D coordinates of human heads. The tracking module receives the human body bounding boxes from the instance segmentation module and tracks them. The 3D trajectory generation module associates the 3D coordinates of human heads provided by the 3D coordinates calculation module and the IDs provided by the tracking module to generate the 3D trajectory of each person.

As shown in Figure 2, our 3D occupant positioning method contains the distortion correction module, the instance segmentation module, and the 3D coordinates calculation module; our 3D occupant trajectory reconstruction method contains the above three modules as well as the tracking module and the 3D trajectory generation module.

In the following, we will introduce our 3D occupant positioning method and our 3D occupant trajectory reconstruction method in turn.

2.2. Three-Dimensional Occupant Positioning

Our 3D occupant positioning method contains the distortion correction module, the instance segmentation module, and the 3D coordinates calculation module.

The distortion correction module uses the distortion parameters calculated by the camera calibration module to remove radial and tangential distortions in the captured videos and images. This module makes use of OpenCV's distortion correction functions.

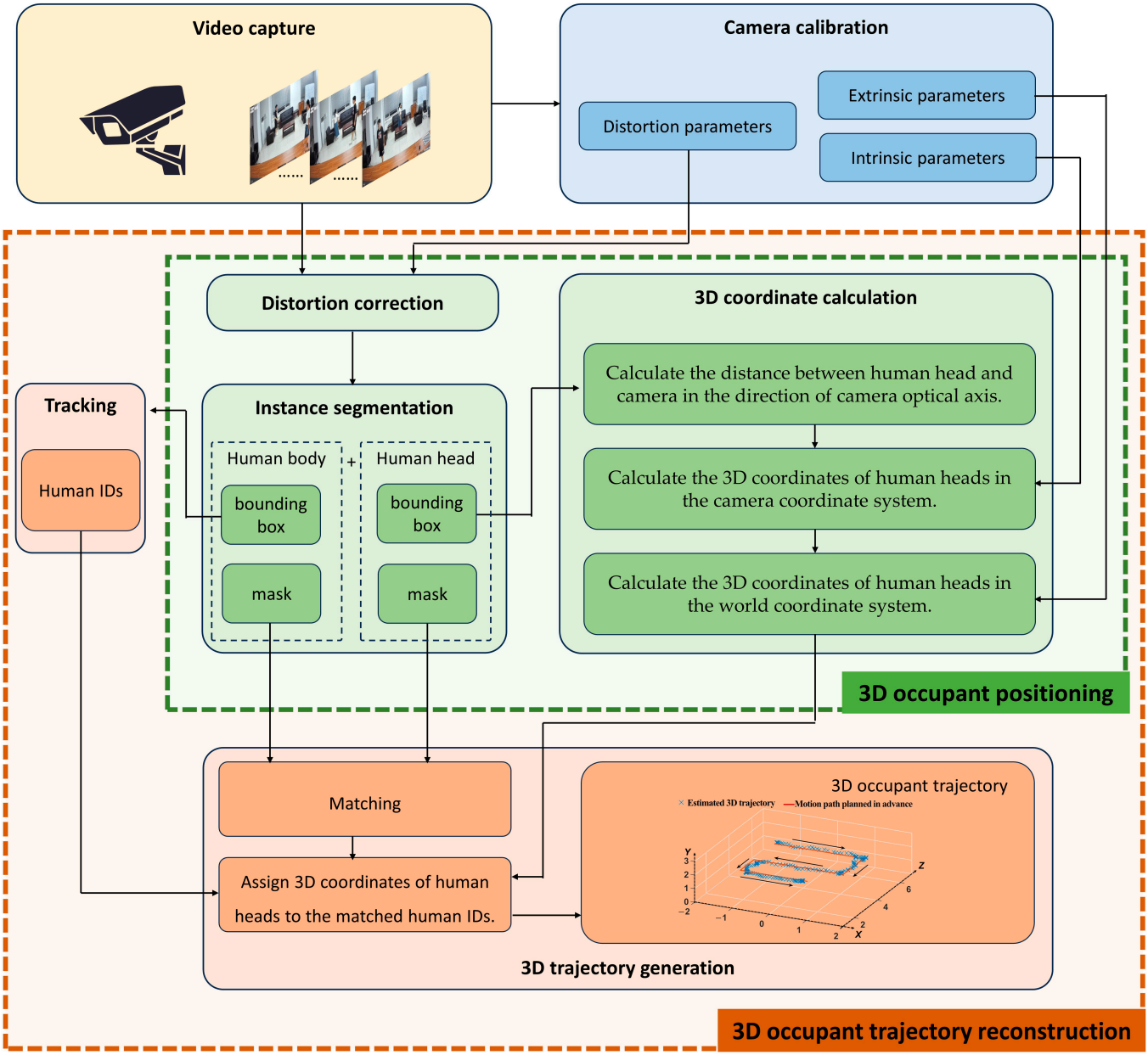


Figure 2. The flowchart of our proposed cost-effective system for indoor 3D occupant positioning and trajectory reconstruction.

The instance segmentation module segments each instance of human body and human head. Each instance segmentation result contains a classification label, bounding box coordinates, and a segmentation mask. The classification label indicates whether the segmented instance is a human body or a human head. The bounding box coordinates indicate the bounding box of the segmented instance. The segmentation mask is a binary mask delineating the segmented instance. For convenience, we reorganize these outputs into two groups of segmentation results: human bodies and human heads. Both types of instance segmentation results contain the corresponding bounding boxes and binary masks. The instance segmentation module uses the state-of-the-art real-time YOLOV8 method [38]. In our 3D occupant positioning method, only the bounding boxes of human heads are further used for 3D coordinates calculation.

The 3D coordinates calculation module uses the bounding boxes of human heads provided by the instance segmentation module to calculate the 3D coordinates of human heads. As shown in Figure 2, this module has three steps. First, it calculates the distance

between the head and the camera in the direction of camera optical axis. Second, it calculates the 3D coordinates of each head in the camera coordinate system. Third, it calculates the 3D coordinates of each head in the world coordinate system.

Since the distortion correction module and the instance segmentation module use previous mature algorithms, a detailed introduction to these two modules will not be provided in the following. Next, we will focus on introducing the three steps of the 3D coordinates calculation module.

(1) Calculate the distance between the head and the camera in the direction of camera optical axis

In the following, d denotes the distance between a specific head and the camera in the direction of camera optical axis. In this step, we build an inverse proportional model to calculate d according to the pixel-measured human head height provided by the human head bounding box.

Figure 3 shows the measurements needed to perform human head imaging. This figure is helpful in understanding the derivation of our inverse proportional model. In Figure 3, a, b, and c are three points on the imaging plane; A, B, and C are three points on the plane of the head, which is perpendicular to the camera's optical axis; O is the camera optical center; a, O, and A are collinear; b, O, and B are collinear; c, O, and C are collinear; $\overline{h}$ is the physical height of the head on the imaging plane, which equals the distance between a and b; H is the actual physical height of the head in the 3D world, which equals the distance between A and B; f is the focal length of camera, which equals the distance between c and O; and d is the distance between the head and the camera in the direction of camera optical axis, which equals the distance between C and O.

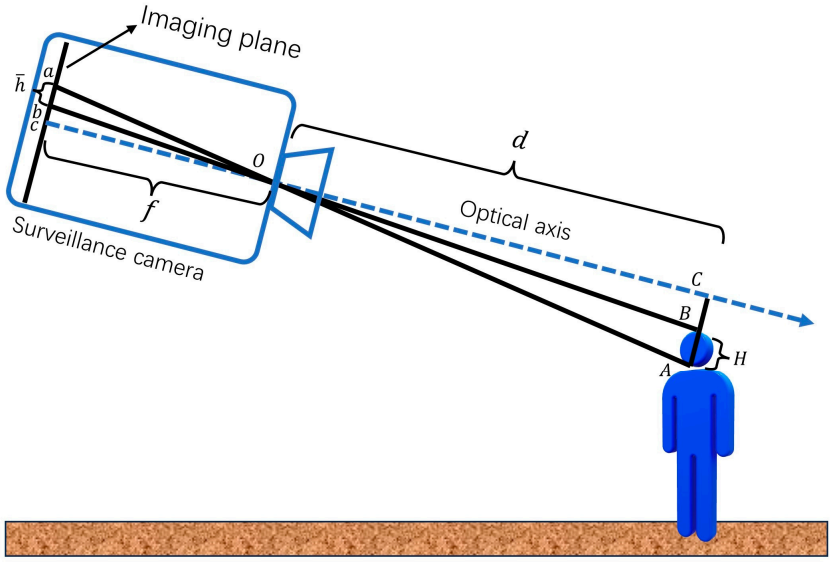


Figure 3. The schematic diagram of human head imaging. In this figure, a, b, and c are three points on the imaging plane; A, B, and C are three points on the plane of the head, which is perpendicular to the camera's optical axis; O is the camera optical center; $\overline{h}$ is the physical height of the head on the imaging plane; H is the actual physical height of the head in the 3D world; f is the focal length of the camera; d is the distance between the head and the camera in the direction of the camera optical axis.

Since triangle ΔacO is similar to triangle ΔACO, it follows that

$$\frac{|ac|}{|AC|} = \frac{f}{d} \tag{1}$$

where $|ac|$ denotes the distance between a and c, and $|AC|$ denotes the distance between A and C. Equation (1) can be rewritten as

$$|ac| = \frac{f}{d}|AC| \tag{2}$$

Since triangle ΔbcO is similar to triangle ΔBCO, it follows that

$$\frac{|bc|}{|BC|} = \frac{f}{d} \tag{3}$$

where $|bc|$ denotes the distance between b and c, and $|BC|$ denotes the distance between B and C. Equation (3) can be rewritten as

$$|bc| = \frac{f}{d}|BC| \tag{4}$$

We then calculate the difference between Equations (2) and (4):

$$|ac| - |bc| = \frac{f}{d}(|AC| - |BC|) \tag{5}$$

Because $|ac| - |bc| = |ab| = \overline{h}$, and $|AC| - |BC| = |AB| = H$, Equation (5) can be rewritten as

$$\overline{h} = \frac{f}{d}H \tag{6}$$

There exists a scale factor α between the pixels in the image and the actual physical size. The pixel-measured head height h conforms to the following equation:

$$h = \alpha \cdot \overline{h} = \frac{\alpha \cdot f}{d}H \tag{7}$$

Equation (7) can be further rewritten as

$$d = \frac{\alpha \cdot f}{h}H \tag{8}$$

Using f_α to denote $\alpha \cdot f$, Equation (8) can be rewritten as

$$d = \frac{f_\alpha}{h}H \tag{9}$$

Equation (9) is the inverse proportional model between h and d. In our system, f_α is provided by the camera calibration module as an intrinsic parameter, and h is provided by the head bounding box. For adults, the body to head ratio is approximately equal to 7. In our system, we first use H = 0.25 m to calculate the primary 3D coordinates of heads in the 3D world using the 3D coordinates calculation module and then use the Y-coordinate of the head in the 3D world as the body height. Finally, we use the human body height divided by 7 as the new H and use this new H to recalculate the 3D coordinates of human heads in the 3D world.

We use box_n^k to denote the bounding box of the n^{th} head in the k^{th} frame. box_n^k is provided by the instance segmentation module. $box_n^k = \left\{u_{n,ul}^k, v_{n,ul}^k, w_n^k, h_n^k\right\}$, where $u_{n,ul}^k$ and $v_{n,ul}^k$ are the horizontal and vertical coordinates, respectively, of the upper-left corner of the bounding box; w_n^k and h_n^k are the width and height, respectively, of the bounding box. We use h_n^k as the pixel-height of the head. The distance d_n^k between the n^{th} head in the k^{th} frame and the camera in the direction of the camera optical axis can be calculated as

$$d_n^k = \frac{f_\alpha}{h_n^k} \cdot H \tag{10}$$

(2) Calculate the 3D coordinates of human heads in the camera coordinate system

The camera coordinate system uses the typical X-, Y-, and Z-axes. We use $PC_n^k = \begin{bmatrix} XC_n^k & YC_n^k & ZC_n^k \end{bmatrix}^{-1}$ to denote the 3D coordinates of the nth head in the kth frame in the camera coordinate system with the optical axis of the camera serving as the Z-axis. Therefore, ZC_n^k equals to d_n^k:

$$ZC_n^k = d_n^k = \frac{f_\alpha}{h_n^k} \cdot H \tag{11}$$

According to the camera imaging principle [39], the relationship between the 2D coordinates in images $\begin{bmatrix} u & v \end{bmatrix}^{-1}$ and the 3D coordinates in the camera coordinate system $\begin{bmatrix} XC & YC & ZC \end{bmatrix}^{-1}$ is

$$ZC \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & 0 & \Delta u \\ 0 & \alpha & \Delta v \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} XC \\ YC \\ ZC \end{bmatrix} \tag{12}$$

where α denotes the scale factor between the pixels in the image and the actual physical size; Δu is the horizontal offset; Δv is the vertical offset; and f is the focal length of the camera. We can rewrite Equation (11) as

$$\begin{cases} u = \frac{\alpha \cdot f}{ZC} \cdot XC + \Delta u \\ v = \frac{\alpha \cdot f}{ZC} \cdot YC + \Delta v \end{cases} \tag{13}$$

Similarly to Equation (9), we use f_α to denote $\alpha \cdot f$ and obtain

$$\begin{cases} u = \frac{f_\alpha}{ZC} \cdot XC + \Delta u \\ v = \frac{f_\alpha}{ZC} \cdot YC + \Delta v \end{cases} \tag{14}$$

Equation (12) can be further rewritten as

$$\begin{cases} XC = \frac{ZC}{f_\alpha}(u - \Delta u) \\ YC = \frac{ZC}{f_\alpha}(v - \Delta v) \end{cases} \tag{15}$$

In our system, ZC is calculated using Equation (11); f_α, Δu, and Δv are provided by the camera calibration module as intrinsic parameters; u and v are calculated from the head bounding box coordinates. Using the nth head in the kth frame as an example, the horizontal and vertical coordinates of this head are denoted as u_n^k and v_n^k, respectively. u_n^k and v_n^k are then calculated as

$$\begin{cases} u_n^k = u_{n,ul}^k + \frac{w_n^k}{2} \\ v_n^k = v_{n,ul}^k + \frac{h_n^k}{2} \end{cases} \tag{16}$$

where $u_{n,ul}^k$ and $v_{n,ul}^k$ are the horizontal and vertical coordinates of the upper-left corner of box_n^k; h_n^k and w_n^k are the height and width of box_n^k; and $box_n^k = \left\{ u_{n,ul}^k, v_{n,ul}^k, w_n^k, h_n^k \right\}$ is the bounding box of the nth head in the kth frame provided by the instance segmentation module. We then use ZC_n^k, u_n^k, and v_n^k to replace ZC, u, and v in Equation (15) to obtain

$$\begin{cases} XC_n^k = \frac{ZC_n^k}{f_\alpha}\left(u_{n,ul}^k + \frac{w_n^k}{2} - \Delta u \right) \\ YC_n^k = \frac{ZC_n^k}{f_\alpha}\left(v_{n,ul}^k + \frac{h_n^k}{2} - \Delta v \right) \end{cases} \tag{17}$$

In conclusion, ZC_n^k is calculated using Equation (11), and XC_n^k and YC_n^k are calculated using Equation (17). We finally obtain the 3D coordinates of the nth head in the kth frame in the camera coordinate system: $PC_n^k = \begin{bmatrix} XC_n^k & YC_n^k & ZC_n^k \end{bmatrix}^{-1}$.

(3) Calculate the 3D coordinates of human heads in the world coordinate system

The 3D coordinates of heads in the world coordinate system are calculated based the 3D coordinates in the camera coordinate system and the extrinsic parameters provided by the camera calibration module. For the nth head in the kth frame, the 3D coordinates in the world coordinate system $PW_n^k = \begin{bmatrix} XW_n^k & YW_n^k & ZW_n^k \end{bmatrix}^{-1}$ are calculated as

$$\begin{bmatrix} XW_n^k \\ YW_n^k \\ ZW_n^k \end{bmatrix} = R^{-1} \left(\begin{bmatrix} XC_n^k \\ YC_n^k \\ ZC_n^k \end{bmatrix} - t \right) \tag{18}$$

where $\begin{bmatrix} XC_n^k & YC_n^k & ZC_n^k \end{bmatrix}^{-1}$ are the coordinates of the head in the camera coordinate system calculated by Equations (11) and (17); R is the rotation matrix between the world coordinate system and the camera coordinate system with a size of 3×3; and t is the translation vector between the world coordinate system and the camera coordinate system with a size of 3×1. R and t are extrinsic parameters provided by the camera calibration module. $PW_n^k = \begin{bmatrix} XW_n^k & YW_n^k & ZW_n^k \end{bmatrix}^{-1}$ is the 3D position coordinate that we aim to calculate in our 3D occupant positioning method.

2.3. Three-Dimensional Occupant Trajectory Generation

Our 3D occupant trajectory generation method includes the three modules contained in the 3D occupant positioning method, as well as the tracking module and the 3D trajectory generation module.

As described in the last subsection, the instance segmentation results contain the bounding boxes and binary masks of human bodies, as well as the bounding boxes and binary masks of human heads. In our 3D occupant trajectory generation method, the bounding boxes of human bodies are further used for tracking. The binary masks of human bodies and human heads are further used for matching.

The tracking module receives the human body bounding boxes outputted by the instance segmentation module and tracks them. This module assigns a consistent ID to the same human appearing in different frames. This module is built using BoT-SORT [40].

The 3D trajectory generation module generates the 3D trajectory of each occupant by combining the information about heads and bodies. $Head^k = \left\{ head_n^k \middle| n = 1, 2, \ldots\ldots, N^k \right\}$ denotes the information about all of the heads in the kth frame used in the 3D trajectory generation module. $head_n^k$ denotes the information for the nth head in the kth frame, with $head_n^k = \left(Hmask_m^k,\ PW_n^k \right)$. $Hmask_m^k$ denotes the binary mask of the nth head in the kth frame and is provided by the instance segmentation module. PW_n^k denotes the 3D position coordinates of the nth head in the kth frame and is provided by the 3D coordinates calculation module. N^k denotes the total number of heads detected by the instance segmentation module in the kth frame. $Body^k = \left\{ body_m^k \middle| m = 1, 2, \ldots\ldots, M^k \right\}$ denotes all of the human body information in the kth frame used in the 3D trajectory generation module. $body_m^k$ denotes the information of the nth body in the kth frame, with $body_m^k = \left(Bmask_m^k,\ ID_m^k \right)$. $Bmask_m^k$ denotes the binary mask of the mth body in the kth frame and is provided by the instance segmentation module. ID_m^k denotes the ID of the mth body in the kth frame and is provided by the tracking module. The tracking module assigns a consistent ID to identical humans appearing in different frames. M^k denotes the total number of bodies detected by the instance segmentation module in the kth frame.

In the 3D trajectory generation module, we first match $head_n^k$ and $body_m^k$ based on the degree of overlap between their binary masks using the test

$$\begin{cases} \frac{\left|Hmask_m^k \cap Bmask_m^k\right|}{\left|Hmask_m^k\right|} > T, \ head_n^k \text{ matches } body_m^k \\ \frac{\left|Hmask_m^k \cap Bmask_m^k\right|}{\left|Hmask_m^k\right|} < T, \ head_n^k \text{ mismatches } body_m^k \end{cases} \tag{19}$$

where $\left|Hmask_m^k \cap Bmask_m^k\right|$ denotes the overlapping area between the head and body masks $Hmask_m^k$ and $Bmask_m^k$, respectively.; $\left|Hmask_m^k\right|$ denotes the area of $Hmask_m^k$, T is the threshold used to determine whether $head_n^k$ matches $body_m^k$. If $head_n^k$ successfully matches $body_m^k$, then we associate ID_m^k of $body_m^k$ and PW_n^k of $head_n^k$. If $head_n^k$ mismatches $body_m^k$, no operation is performed.

For each human body, we try to find its corresponding human head and assign the 3D coordinates of this human head to the ID of this human body. Then, we will obtain a sequence of 3D coordinates for each individual ID, which is the 3D trajectory of the corresponding occupant. Through the above method, we obtain the 3D trajectory of each human appearing in the video.

3. Experiments

This paper proposes a 3D occupant positioning and trajectory reconstruction system. In the following, we first introduce the experimental datasets, then present and analyze the 3D positioning results and the 3D trajectory reconstruction results, and finally present and analyze the running time of the proposed system.

3.1. Datasets

We built two datasets to evaluate the performance of the proposed cost-effective 3D occupant positioning and trajectory reconstructing system. One dataset consists of images used to evaluate the accuracy of 3D occupant positioning. The other dataset consists of videos used to evaluate the performance of 3D occupant trajectory reconstruction.

3.1.1. Dataset for 3D Occupant Positioning

Figure 4 shows how the 3D positioning dataset was created. In this figure, the 3D coordinate system marked in red is the world coordinate system. As shown in this figure, the origin of the world coordinate system is located on the ground directly below the surveillance camera, with the directions of the X-, Y-, and Z-axes indicated. When we captured the images that used to evaluate the accuracy of 3D positioning, we instructed occupants to stand at the cross junctions of the floor tiles. Because the size of the floor tiles is fixed, we obtained the ground-truth of the X- and Z-coordinates of each head in the world coordinate system by counting the number of floor tiles. The height of a human measured by a ruler was used as the ground-truth of the Y-coordinate of each head in the world coordinate system. Using this method, we collected images from different scenes and annotated the ground-truth of the 3D coordinates of the heads in the world coordinate system.

Details of our 3D positioning dataset are shown in Table 1. We acquired our dataset in five scenes. In each scene, we acquired three image sets. The number of images contained in each set and the size of the images are shown in Table 1.

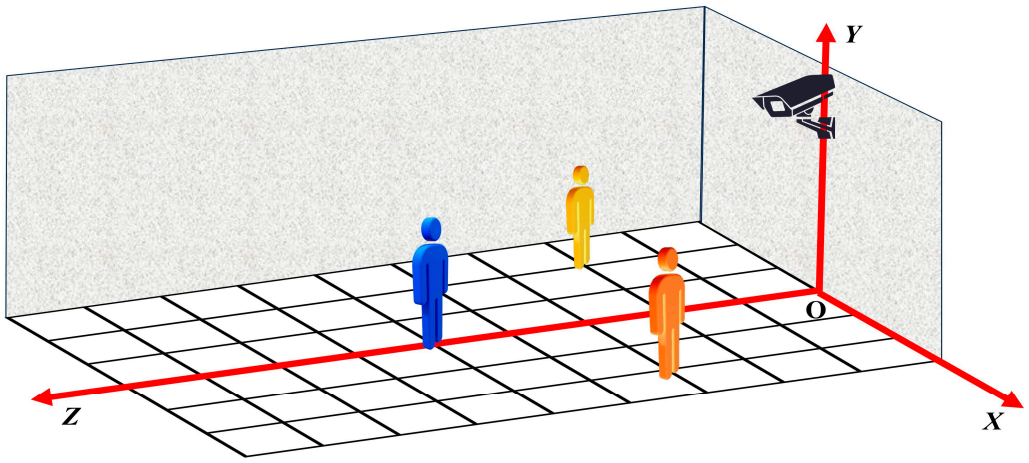


Figure 4. The diagram of creating the 3D occupant positioning dataset. The *X*-, *Y*-, and *Z*- axes marked in red denote the 3D world coordinate system; *O* denotes the origin of the world coordinate system.

Table 1. Details of the 3D occupant positioning dataset.

Image Sets	Number of Images	Image Size (Pixels)	Scene
Image Set A-1	21	3840 × 2160	Meeting room
Image Set A-2	21	3840 × 2160	
Image Set A-3	11	3840 × 2160	
Image Set B-1	9	3840 × 2160	Reception room
Image Set B-2	18	3840 × 2160	
Image Set B-3	10	3840 × 2160	
Image Set C-1	20	3840 × 2160	Elevator hall
Image Set C-2	45	3840 × 2160	
Image Set C-3	23	3840 × 2160	
Image Set D-1	9	3840 × 2160	Classroom
Image Set D-2	13	3840 × 2160	
Image Set D-3	10	3840 × 2160	
Image Set E-1	26	3840 × 2160	Hallway
Image Set E-2	26	3840 × 2160	
Image Set E-3	23	3840 × 2160	

3.1.2. Dataset for 3D Occupant Trajectory Reconstruction

Figure 5 shows how we created the 3D occupant trajectory reconstruction dataset. The world coordinate system used in this dataset is same as the world coordinate system used in the 3D occupant positioning dataset. When we recorded the videos used to evaluate the performance of 3D trajectory reconstruction, we planned the motion paths for occupants in advance. In Figure 5, the blue dotted lines are the planned walking paths and the blue solid lines are the planned motion paths of the human head.

Details of our 3D occupant trajectory reconstruction dataset are shown in Table 2. We acquired five videos from each of five scenes, with the size and number of frames as shown in Table 2.

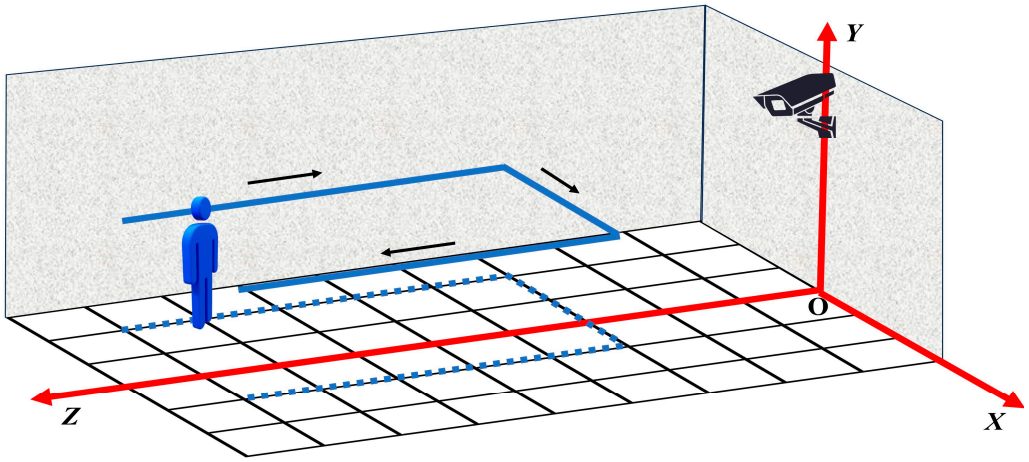


Figure 5. The diagram of creating the 3D occupant trajectory reconstruction dataset. The *X*-, *Y*-, and *Z*-axes marked in red denote the 3D world coordinate system; *O* denotes the origin of the world coordinate system. The blue dotted lines are the walking paths planned in advance. The blue solid lines are the motion paths of the head planned in advance.

Table 2. Details of the 3D occupant trajectory reconstruction dataset.

Videos	Number of Frames	Image Size (Pixels)	Scene
Video A	243	1280 × 720	Meeting room
Video B	253	3840 × 2160	Reception room
Video C	432	3840 × 2160	Elevator hall
Video D	528	3840 × 2160	Classroom
Video E	487	3840 × 2160	Hallway

3.2. Three-Dimensional Occupant Positioning

The 3D occupant positioning dataset described in Section 3.1.1 was used in our 3D positioning experiments. In these experiments, all the images were resized to 1280 × 720. We annotated the ground-truth 3D coordinates of heads in the world coordinate system by the method described in Section 3.1.1. To evaluate the accuracy of our 3D positioning method, we calculated the mean absolute error in the *X*-, *Y*-, and *Z*-axes and the spatial error in the 3D world using Equations (20)–(23).

$$X\ Error = \frac{1}{N}\sum_{n=1}^{N}\left|XW_n^p - XW_n^{gt}\right| \tag{20}$$

$$Y\ Error = \frac{1}{N}\sum_{n=1}^{N}\left|YW_n^p - YW_n^{gt}\right| \tag{21}$$

$$Z\ Error = \frac{1}{N}\sum_{n=1}^{N}\left|ZW_n^p - ZW_n^{gt}\right| \tag{22}$$

$$Spatial\ Error = \frac{1}{N}\sum_{n=1}^{N}\sqrt[2]{\left(XW_n^p - XW_n^{gt}\right)^2 + \left(YW_n^p - YW_n^{gt}\right)^2 + \left(ZW_n^p - ZW_n^{gt}\right)^2} \tag{23}$$

In the preceding equations, XW_n^p, YW_n^p, and ZW_n^p are the predicted *X*-, *Y*-, and *Z*-coordinates of the *n*th head in the world coordinate system; XW_n^{gt}, YW_n^{gt}, and ZW_n^{gt} are the ground-truth values of the *X*-, *Y*-, and *Z*-coordinates of the *n*th head in the world coordinate system, and *N* denotes the total number of heads. The 3D positioning evaluation results are shown in Table 3.

Table 3. Three-dimensional occupant positioning evaluation results.

Image Sets	*X* Error (cm)	*Y* Error (cm)	*Z* Error (cm)	Spatial Error (cm)
Image Set A-1	3.28	1.89	11.54	12.75
Image Set A-2	3.30	2.87	14.00	15.38
Image Set A-3	4.40	2.55	15.37	16.91
Image Set B-1	3.16	5.59	6.73	10.23
Image Set B-2	3.99	4.66	14.08	16.04
Image Set B-3	3.82	5.61	19.77	22.31
Image Set C-1	6.94	2.72	16.77	18.65
Image Set C-2	6.53	1.76	11.46	14.51
Image Set C-3	4.85	2.52	15.61	17.43
Image Set D-1	5.87	0.95	16.10	17.60
Image Set D-2	5.88	1.18	16.13	18.13
Image Set D-3	5.10	0.82	18.50	19.49
Image Set E-1	4.47	3.57	11.46	13.98
Image Set E-2	5.99	2.97	15.82	17.85
Image Set E-3	6.00	3.30	11.58	14.71
Mean	4.91	2.86	14.33	16.40

As shown in Table 3, in all of our image sets, the errors in the *X*- and *Y*-axes are small, at 4.91 cm and 2.86 cm, respectively. In contrast, the *Z*-axis error is large at 14.33 cm. The *Z*-axis error is the primary source of the final spatial error.

We also compared the mean spatial error of our method with those of other 3D positioning algorithms: Dai [30], EPI + TOF [27], and Zhou [35]. The mean spatial errors of these methods were taken from their articles. If multiple groups of data were used in their articles, we calculated their average values. Our 3D occupant positioning method obtained the depth information based on our proposed inverse proportional model of the head height in the image and the distance between the head and camera in the direction of the camera optical axis. This depth information could also be obtained by depth estimation methods. Therefore, we also built a 3D positioning module by combining depth estimation [41] and 3D reconstruction, whose mean spatial error is shown in the first line of Table 4. In addition to the mean spatial errors, Table 4 also shows the dependent devices or models of different algorithms. For example, Dai's Baseline [30] uses one surveillance camera and Dai's Baseline + BKF + GC [30] uses one surveillance camera as well as a 3D point cloud map of the environment constructed in advance.

Table 4. Comparisons of different algorithms based on the year of publication, the dependent devices or models, and mean spatial error.

Method	Year	Dependent Devices or Models	Mean Spatial Error (cm)
Depth Estimation [41] + 3D Reconstruction	2022	1 surveillance camera	103.00
Dai's Baseline [30]	2022	1 surveillance camera	54.00
Dai's Baseline + BKF [30]	2022	1 surveillance camera	42.00
Dai's Baseline + BKF + GC [30]	2022	1 surveillance camera + 3D point cloud map	13.67
EPI+TOF [27]	2023	4 RGBD cameras	10.05
Zhou et al. [35]	2023	1 surveillance camera + BIM	16.60
Our 3D Occupant Positioning Method	2023	1 surveillance camera	16.40

Our 3D occupant positioning method calculates the 3D coordinates of heads based on a single surveillance camera. Table 4 shows that, compared to other methods that also use

only one surveillance camera, our 3D occupant positioning method obtained the smallest mean spatial error, achieving the best performance. In addition, the mean spatial error of our algorithm was slightly lower than that of Zhou's algorithm, and slightly higher than those of Dai's Baseline + BKF + GC and EPI + TOF. It should be noted that, although Dai's Baseline + BKF + GC and EPI + TOF have lower mean spatial errors than our algorithm, it comes at a much higher implementation cost. For example, Dai's Baseline + BKF + GC not only requires a surveillance camera but also requires a 3D point cloud map of the environment that is built in advance using an expensive device. EPI + TOF requires four RGBD cameras, which are much more expensive than surveillance cameras. Our 3D occupant positioning method effectively balances the 3D positioning accuracy and the cost. It can be integrated into existing vision surveillance systems without upgrading equipment.

3.3. Three-Dimensional Occupant Trajectory Reconstruction

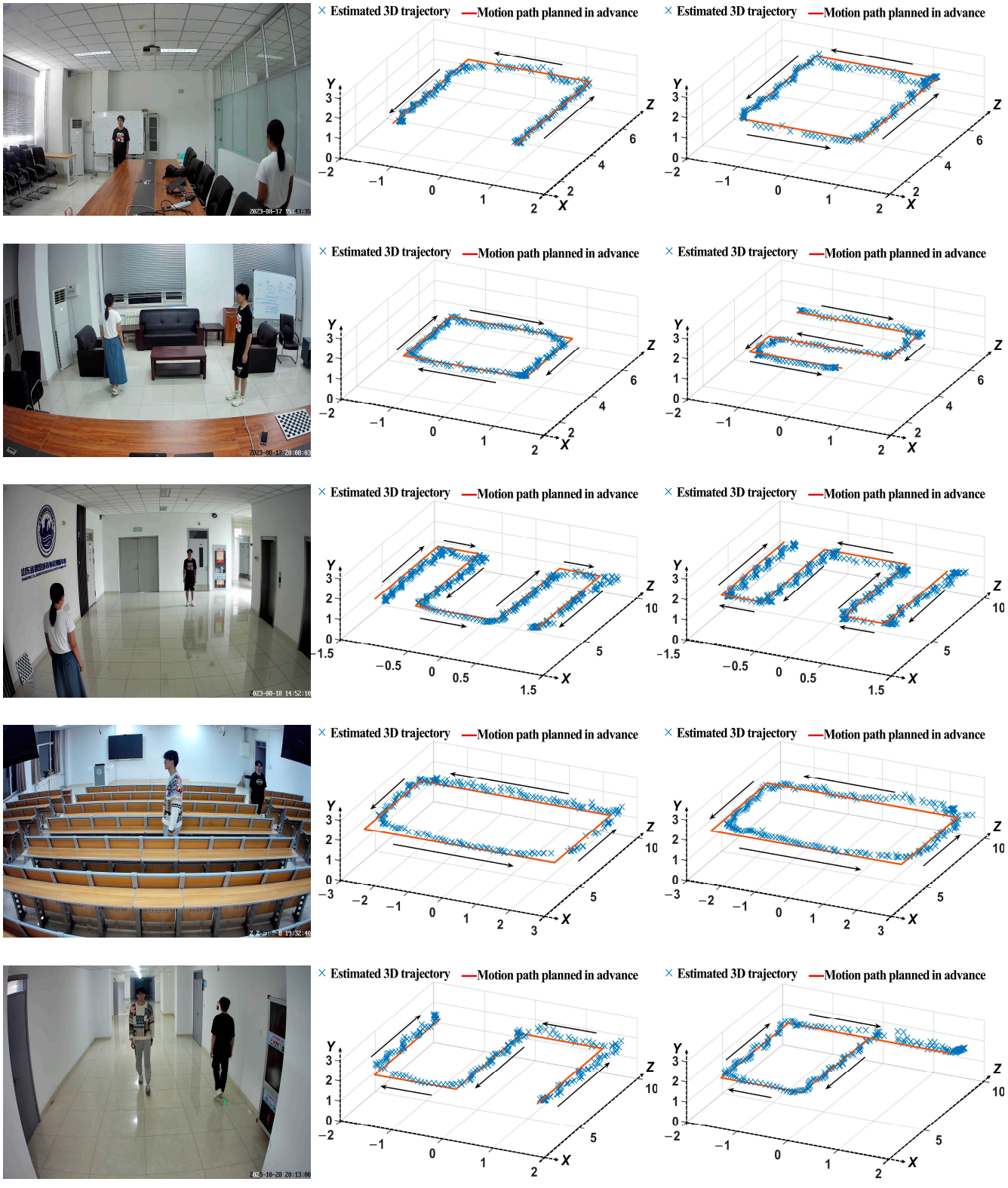
Accurately annotating the 3D trajectories of occupants is very difficult. As mentioned in Section 3.1.2, we used planned motion paths when recording videos of occupants. However, it is difficult to calculate the 3D trajectory reconstruction accuracy based on these motion paths planned in advance for two reasons. First, although we planned the motion paths in advance, we could not ensure that the occupants walked exactly along these paths, especially around bends. Second, the motion paths were composed of lines, while the 3D trajectories calculated based on the videos were point sequences. There is not a one-to-one correspondence between the planned paths and the 3D trajectories calculated from the videos. For these above reasons, we cannot quantitatively calculate the accuracy of the 3D trajectory reconstruction. Instead, we qualitatively show the performance of our 3D occupant trajectory reconstruction method. The 3D occupant trajectory reconstruction dataset described in Section 3.1.2 was used in our trajectory reconstruction experiments. In these experiments, all the frames were resized to 1280×720.

Figure 6 shows the 3D occupant trajectory reconstruction results of our five videos. As shown in these figures, these five videos were captured in different scenes. Their 3D trajectory reconstruction results were consistent with the planned motion paths, demonstrating that our method successfully reconstructed the 3D trajectories of occupants.

From our literature review, most vision-based occupancy information extraction methods either only extract 2D occupancy information or only provide 3D position information. Dai et al. [30] proposed an indoor 3D human trajectory reconstruction method using surveillance cameras and a 3D point cloud map. Comparisons between our method and Dai's [30] are shown in Table 5. Dai obtained the ground-truth of 3D trajectories using two Velodyne LiDAR sensors, which were expensive. Therefore, they could evaluate their trajectories by calculating trajectory errors. Unfortunately, we lacked similar equipment to capture our ground-truth trajectories. Therefore, we compared our method with Dai's qualitatively as shown in Table 5. Table 5 shows that our method is faster and lower cost than the method proposed by Dai et al. [30].

Table 5. Comparisons with other vision-based 3D human trajectory reconstruction methods based on the year of publication, the dependent device or model, and running time.

Methods	Year	Dependent Device or Model	Running TIME (ms/frame)
Indoor 3D human trajectory reconstruction [30]	2022	Surveillance cameras +3D point cloud map	140
Our 3D occupant trajectory reconstruction method	2023	1 surveillance camera	68



(**a**) (**b**) (**c**)

Figure 6. The 3D trajectory reconstruction results of the videos captured in five different scenes. From top to bottom: meeting room, reception room, elevator hall, classroom, and hallway. In addition, (**a**) displays one frame of each video, (**b**) displays the 3D trajectory reconstruction result of one occupant in each video, and (**c**) displays the 3D trajectory reconstruction result of the other occupant in each video. In (**b**,**c**), the red lines are the planned motion paths of heads; the blue points are the predicted 3D trajectories of the heads; the black arrows indicate the direction of movement.

3.4. Running Time Analysis

We measured the duration of our 3D occupant positioning and trajectory reconstruction system, with results given in Table 6. All of our experiments were implemented using PyTorch on a system with one NVidia RTX 3090 GPU and one Intel Xeon Platinum 8358P CPU. As shown in Table 6, the running time of our 3D occupant positioning method was 33 ms/frame and the running time of our 3D occupant trajectory reconstruction method was 70 ms/frame. Our 3D occupant positioning method can run in real time. Its speed is faster than that of our 3D occupant trajectory reconstruction method. This is because, compared to our trajectory reconstruction method, our positioning method does not include the tracking module and the 3D trajectory generation module.

Table 6. Running time of our 3D occupant positioning and 3D occupant trajectory reconstruction methods.

	Methods	3D Occupant Positioning	3D Occupant Trajectory Reconstruction
Modules	Distortion correction	✓	✓
	Instance segmentation	✓	✓
	3D coordinates calculation	✓	✓
	Tracking	✗	✓
	3D trajectory generation	✗	✓
Running Time (ms/frame)		33	70

To further analyze the time cost of each step of the proposed system, we measured the duration of each individual module of our system, with results given in Table 7. As shown in Table 7, the time was mainly spent on the distortion correction, instance segmentation, and tracking steps, requiring 15 ms/frame, 17 ms/frame, and 33 ms/frame, respectively. The average running times of the 3D coordinates calculation and 3D trajectory generation steps were only 1 ms/frame and 4 ms/frame, respectively. This is because the distortion correction, instance segmentation, and tracking modules needed complex pixel-level operations, while the 3D coordinates calculation and 3D trajectory generation modules only needed to process the bounding box coordinates and segmentation masks.

Table 7. Running time of each module of the proposed cost-effective system for indoor 3D occupant positioning and trajectory reconstruction.

Modules	Running Time (ms/Frame)
Distortion correction	15
Instance segmentation	17
3D coordinates calculation	1
Tracking	33
3D trajectory generation	4
Total	70

4. Discussion

Buildings are closely related to people's lives. To construct safe, comfortable, and energy-efficient buildings, numerous researchers have conducted extensive studies [42–45]. In this paper, we focus on the study of how to extract 3D occupant information at a low cost and propose a cost-effective system for indoor 3D occupant positioning and trajectory reconstruction. This system performs 3D positioning and trajectory reconstruction using a single surveillance camera. The comparisons between our 3D positioning method and previous methods have been presented in Table 4. Table 4 not only compares the mean spatial error of different methods, but also compares the publication years and dependent devices or models of different methods. On one hand, this table demonstrates that our 3D

positioning method has the smallest mean spatial error compared to other methods that only use one surveillance camera. On the other hand, it shows that our method has comparable positioning accuracy to other state-of-the-art high-precision positioning methods, while having a lower cost. The comparisons between our 3D trajectory reconstruction method and the previous method are shown in Table 5. Due to the limited number of current vision-based 3D occupant trajectory reconstruction algorithms, Table 5 only compares the indoor 3D human trajectory reconstruction method proposed by Dai et al. [30] and our method. In addition, as we do not have expensive high-precision equipment, such as LiDAR as used in [30], to obtain the ground-truth of 3D trajectory, Table 5 only compares the publication year, the dependent device or model, and running time of different methods. This table demonstrates that our 3D occupant trajectory reconstruction method has a lower cost and a faster speed than Dai's method.

Our algorithm provides accurate 3D occupant positioning information for intelligent building energy management systems. Previously, most occupancy information extraction methods only detected human bodies or heads in 2D images. These methods only provide information about the presence and number of occupants, but nothing about their locations in 3D, thus limiting their application with HVAC, lighting, and many other control systems. For example, in a large meeting room, there are many electric lights throughout the room. When we detect the appearance of occupants in the meeting room, we do not need to turn on all of the lights. Only the lights above the occupants need to be turned on but that requires the occupants' 3D location. Our algorithm also provides accurate 3D occupant trajectories, enabling smarter energy management systems. For example, when many people in the meeting room are moving toward the door, we can predict that the meeting is over and people will leave the meeting room one after another. This enables the system to turn off the air conditioner in advance and direct the elevator to be ready in advance, improving comfort and convenience while saving energy.

Because it uses regular surveillance cameras, our method provides a cost-effective solution that can be easily integrated into existing vision surveillance systems. However, vision-based approaches may raise some privacy concerns. We plan to address these concerns by: (1) employing face blurring methods to conceal faces; (2) implementing 3D positioning and trajectory reconstruction locally, uploading only the 3D trajectory information to the energy management system; and (3) employing network security protection technologies like firewalls to prevent illegal access to vision data. We also plan to conduct a deeper study on protecting occupants' privacy in vision-based surveillance systems in the future, continually improving our approach to address evolving privacy challenges.

5. Conclusions

In this paper, we propose a low-cost system for 3D occupant positioning and trajectory reconstruction. Compared with previous algorithms, the proposed system does not require the construction of large-scale models in advance, as with 3D point cloud maps or BIM, and it does not need special cameras. This proposed system could be easily and directly integrated into existing vision surveillance systems at a very low cost, further promoting the development of intelligent building energy management technology.

There are some limitations in our proposed system. Our system is based on surveillance cameras, which may raise privacy concerns. Our system requires a cumbersome camera calibration process to obtain the intrinsic and extrinsic camera parameters. The field of view of a single surveillance camera is limited, which limits our 3D positioning and trajectory reconstruction area. In the future, we plan to address these limitations by employing technologies to protect privacy, introducing a more convenient calibration algorithm, and combining surveillance cameras in different rooms to calculate the 3D positions and trajectories of occupants in the entire building.

Author Contributions: Conceptualization, X.Z., S.L., Z.Z. and H.L.; methodology, X.Z. and S.L.; software, X.Z. and S.L.; validation, S.L., Z.Z. and H.L.; writing—original draft preparation, X.Z. and S.L.; writing—review and editing, X.Z., S.L., Z.Z. and H.L.; visualization, X.Z. and S.L.; project administration, X.Z.; funding acquisition, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of Shandong Province, grant number ZR2021QF094 and the Youth Innovation Team Technology Project of Higher School in Shandong Province, grant number 2022KJ204.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy concerns.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Long, C.M.; Suh, H.H.; Catalano, P.J.; Koutrakis, P. Using time-and size-resolved particulate data to quantify indoor penetration and deposition behavior. *Environ. Sci. Technol.* **2001**, *35*, 2089–2099. [CrossRef] [PubMed]
2. D'Oca, S.; Hong, T.; Langevin, J. The human dimensions of energy use in buildings: A review. *Renew. Sust. Energy Rev.* **2018**, *81*, 731–742. [CrossRef]
3. Kang, X.; Yan, D.; An, J.; Jin, Y.; Sun, H. Typical weekly occupancy profiles in non-residential buildings based on mobile positioning data. *Energy Build.* **2021**, *250*, 111264. [CrossRef]
4. Zhang, R.; Kong, M.; Dong, B.; O'Neill, Z.; Cheng, H.; Hu, F.; Zhang, J. Development of a testing and evaluation protocol for occupancy sensing technologies in building HVAC controls: A case study of representative people counting sensors. *Build. Environ.* **2022**, *208*, 108610. [CrossRef]
5. Sayed, A.N.; Himeur, Y.; Bensaali, F. Deep and transfer learning for building occupancy detection: A review and comparative analysis. *Eng. Appl. Artif. Intel.* **2022**, *115*, 105254. [CrossRef]
6. Pang, Z.; Chen, Y.; Zhang, J.; O'Neill, Z.; Cheng, H.; Dong, B. Nationwide HVAC energy-saving potential quantification for office buildings with occupant-centric controls in various climates. *Appl. Energy* **2020**, *279*, 115727. [CrossRef]
7. Zou, H.; Zhou, Y.; Jiang, H.; Chien, S.-C.; Xie, L.; Spanos, C.J. WinLight: A WiFi-based occupancy-driven lighting control system for smart building. *Energy Build.* **2018**, *158*, 924–938. [CrossRef]
8. Tekler, Z.D.; Low, R.; Yuen, C.; Blessing, L. Plug-Mate: An IoT-based occupancy-driven plug load management system in smart buildings. *Build. Environ.* **2022**, *223*, 109472. [CrossRef]
9. Yang, B.; Liu, Y.; Liu, P.; Wang, F.; Cheng, X.; Lv, Z. A novel occupant-centric stratum ventilation system using computer vision: Occupant detection, thermal comfort, air quality, and energy savings. *Build. Environ.* **2023**, *237*, 110332. [CrossRef]
10. Zhang, J.; Zhao, T.; Zhou, X.; Wang, J.; Zhang, X.; Qin, C.; Luo, M. Room zonal location and activity intensity recognition model for residential occupant using passive-infrared sensors and machine learning. *Build. Simul.* **2022**, *15*, 1133–1144. [CrossRef]
11. Franco, A.; Leccese, F. Measurement of CO_2 concentration for occupancy estimation in educational buildings with energy efficiency purposes. *J. Build. Eng.* **2020**, *32*, 101714. [CrossRef]
12. Kampezidou, S.I.; Ray, A.T.; Duncan, S.; Balchanos, M.G.; Mavris, D.N. Real-time occupancy detection with physics-informed pattern-recognition machines based on limited CO_2 and temperature sensors. *Energy Build.* **2021**, *242*, 110863. [CrossRef]
13. Khan, A.; Nicholson, J.; Mellor, S.; Jackson, D.; Ladha, K.; Ladha, C.; Hand, J.; Clarke, J.; Olivier, P.; Plötz, T. Occupancy monitoring using environmental & context sensors and a hierarchical analysis framework. In Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings, Memphis, TN, USA, 3–6 November 2014. [CrossRef]
14. Uziel, S.; Elste, T.; Kattanek, W.; Hollosi, D.; Gerlach, S.; Goetze, S. Networked embedded acoustic processing system for smart building applications. In Proceedings of the 2013 Conference on Design and Architectures for Signal and Image Processing, Cagliari, Italy, 8–10 October 2013.
15. Islam, S.M.M.; Droitcour, A.; Yavari, E.; Lubecke, V.M.; Boric-Lubecke, O. Building occupancy estimation using microwave Doppler radar and wavelet transform. *Build. Environ.* **2023**, *236*, 110233. [CrossRef]
16. Tang, C.; Li, W.; Vishwakarma, S.; Chetty, K.; Julier, S.; Woodbridge, K. Occupancy detection and people counting using wifi passive radar. In Proceedings of the 2020 IEEE Radar Conference (RadarConf20), Florence, Italy, 21–25 September 2020. [CrossRef]
17. Zaidi, A.; Ahuja, R.; Shahabi, C. Differentially Private Occupancy Monitoring from WiFi Access Points. In Proceedings of the 2022 23rd IEEE International Conference on Mobile Data Management, Paphos, Cyprus, 6–9 June 2022. [CrossRef]
18. Abolhassani, S.S.; Zandifar, A.; Ghourchian, N.; Amayri, M.; Bouguila, N.; Eicker, U. Improving residential building energy simulations through occupancy data derived from commercial off-the-shelf Wi-Fi sensing technology. *Energy Build.* **2022**, *272*, 112354. [CrossRef]
19. Tekler, Z.D.; Low, R.; Gunay, B.; Andersen, R.K.; Blessing, L. A scalable Bluetooth Low Energy approach to identify occupancy patterns and profiles in office spaces. *Build. Environ.* **2020**, *171*, 106681. [CrossRef]

20. Apolónia, F.; Ferreira, P.M.; Cecílio, J. Buildings Occupancy Estimation: Preliminary Results Using Bluetooth Signals and Artificial Neural Networks. In *Communications in Computer and Information Science*; Springer: Cham, Switzerland, 2022; Volume 1525, pp. 567–579. [CrossRef]
21. Brown, R.; Ghavami, N.; Adjrad, M.; Ghavami, M.; Dudley, S. Occupancy based household energy disaggregation using ultra wideband radar and electrical signature profiles. *Energy Build.* **2017**, *141*, 134–141. [CrossRef]
22. Xu, Y.; Shmaliy, Y.S.; Li, Y.; Chen, X. UWB-based indoor human localization with time-delayed data using EFIR filtering. *IEEE Access* **2017**, *5*, 16676–16683. [CrossRef]
23. Sun, K.; Liu, P.; Xing, T.; Zhao, Q.; Wang, X. A fusion framework for vision-based indoor occupancy estimation. *Build. Environ.* **2022**, *225*, 109631. [CrossRef]
24. Tekler, Z.D.; Chong, A. Occupancy prediction using deep learning approaches across multiple space types: A minimum sensing strategy. *Build. Environ.* **2022**, *226*, 109689. [CrossRef]
25. Tan, S.Y.; Jacoby, M.; Saha, H.; Florita, A.; Henze, G.; Sarkar, S. Multimodal sensor fusion framework for residential building occupancy detection. *Energy Build.* **2022**, *258*, 111828. [CrossRef]
26. Qaisar, I.; Sun, K.; Zhao, Q.; Xing, T.; Yan, H. Multi-sensor Based Occupancy Prediction in a Multi-zone Office Building with Transformer. *Buildings* **2023**, *13*, 2002. [CrossRef]
27. Wang, H.; Wang, G.; Li, X. An RGB-D camera-based indoor occupancy positioning system for complex and densely populated scenarios. *Indoor Built Environ.* **2023**, *32*, 1198–1212. [CrossRef]
28. Sun, K.; Zhao, Q.; Zou, J. A review of building occupancy measurement systems. *Energy Build.* **2020**, *216*, 109965. [CrossRef]
29. Labeodan, T.; Zeiler, W.; Boxem, G.; Zhao, Y. Occupancy measurement in commercial office buildings for demand-driven control applications–A survey and detection system evaluation. *Energy Build.* **2015**, *93*, 303–314. [CrossRef]
30. Dai, Y.; Wen, C.; Wu, H.; Guo, Y.; Chen, L.; Wang, C. Indoor 3D human trajectory reconstruction using surveillance camera videos and point clouds. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 2482–2495. [CrossRef]
31. Sun, K.; Ma, X.; Liu, P.; Zhao, Q. MPSN: Motion-aware Pseudo-Siamese Network for indoor video head detection in buildings. *Build. Environ.* **2022**, *222*, 109354. [CrossRef]
32. Hu, S.; Wang, P.; Hoare, C.; O'Donnell, J. Building Occupancy Detection and Localization Using CCTV Camera and Deep Learning. *IEEE Internet Things* **2023**, *10*, 597–608. [CrossRef]
33. Wang, C.; Zhang, Y.; Zhou, Y.; Sun, S.; Zhang, H.; Wang, Y. Automatic detection of indoor occupancy based on improved YOLOv5 model. *Neural Comput. Appl.* **2023**, *35*, 2575–2599. [CrossRef]
34. Wang, H.; Wang, G.; Li, X. Image-based occupancy positioning system using pose-estimation model for demand-oriented ventilation. *J. Build. Eng.* **2021**, *39*, 102220. [CrossRef]
35. Zhou, X.; Sun, K.; Wang, J.; Zhao, J.; Feng, C.; Yang, Y.; Zhou, W. Computer vision enabled building digital twin using building information model. *IEEE Trans. Ind. Inform.* **2023**, *19*, 2684–2692. [CrossRef]
36. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. Yolox: Exceeding yolo series in 2021. *arXiv* **2021**, arXiv:2107.08430. [CrossRef]
37. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015. [CrossRef]
38. Glenn, J. Yolo by Ultralytics (Version 8.0.0). Available online: https://github.com/ultralytics/ultralytics (accessed on 8 November 2023).
39. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2003.
40. Aharon, N.; Orfaig, R.; Bobrovsky, B.-Z. BoT-SORT: Robust associations multi-pedestrian tracking. *arXiv* **2022**, arXiv:2206.14651. [CrossRef]
41. Lian, D.; Chen, X.; Li, J.; Luo, W.; Gao, S. Locating and counting heads in crowds with a depth prior. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 9056–9072. [CrossRef] [PubMed]
42. Li, T.; Zhao, W.; Liu, R.; Han, J.-Y.; Jia, P.; Cheng, C. Visualized direct shear test of the interface between gravelly sand and concrete pipe. *Can. Geotech. J.* **2023**. [CrossRef]
43. Tawil, H.; Tan, C.G.; Sulong, N.H.R.; Nazri, F.M.; Sherif, M.M.; El-Shafie, A. Mechanical and thermal properties of composite precast concrete sandwich panels: A Review. *Buildings* **2022**, *12*, 1429. [CrossRef]
44. Han, J.; Wang, J.; Jia, D.; Yan, F.; Zhao, Y.; Bai, X.; Yan, N.; Yang, G.; Liu, D. Construction technologies and mechanical effects of the pipe-jacking crossing anchor-cable group in soft stratum. *Fron. Earth Sci.* **2023**, *10*, 1019801. [CrossRef]
45. Sun, H.; Liu, Y.; Guo, X.; Zeng, K.; Mondal, A.K.; Li, J.; Yao, Y.; Chen, L. Strong, robust cellulose composite film for efficient light management in energy efficient building. *Chem. Eng. J.* **2021**, *425*, 131469. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.