



Article

Automatic Multi-Camera Extrinsic Parameter Calibration Based on Pedestrian Torsors [†]

Anh Minh Truong ^{1,*}, Wilfried Philips ¹, Nikos Deligiannis ², Lusine Abrahamyan ² and Junzhi Guan ^{3,‡}

¹ TELIN-IPI, Ghent University—imec, St-Pietersnieuwstraat 41, B-9000 Gent, Belgium; Wilfried.Philips@UGent.be

² ETRO Department, Vrije Universiteit Brussel—imec, Pleinlaan 2, B-1050 Brussels, Belgium; ndeligia@etrovub.be (N.D.); alusine@etrovub.be (L.A.)

³ CETC Key Laboratory of Aerospace Information Applications, Shijiazhuang 050000, China; guanjunzhi@hotmail.com

* Correspondence: anhminh.truong@UGent.be; Tel.: +32-484-62-95-32

† This paper is an extended version of our paper published in Anh Minh Truong, Wilfried Philips, Junzhi Guan, Nikos Deligiannis, and Lusine Abrahamyan. Automatic Extrinsic Calibration of Camera Networks Based on Pedestrians. In Proceedings of International Conference on Distributed Smart Cameras (ICDSC 2019), Trento, Italy, 9–11 September 2019.

‡ These authors contributed equally to this work.

Received: 25 September 2019; Accepted: 12 November 2019; Published: 15 November 2019



Abstract: Extrinsic camera calibration is essential for any computer vision task in a camera network. Typically, researchers place a calibration object in the scene to calibrate all the cameras in a camera network. However, when installing cameras in the field, this approach can be costly and impractical, especially when recalibration is needed. This paper proposes a novel, accurate and fully automatic extrinsic calibration framework for camera networks with partially overlapping views. The proposed method considers the pedestrians in the observed scene as the calibration objects and analyzes the pedestrian tracks to obtain extrinsic parameters. Compared to the state of the art, the new method is fully automatic and robust in various environments. Our method detect human poses in the camera images and then models walking persons as vertical sticks. We apply a brute-force method to determines the correspondence between persons in multiple camera images. This information along with 3D estimated locations of the top and the bottom of the pedestrians are then used to compute the extrinsic calibration matrices. We also propose a novel method to calibrate the camera network by only using the top and centerline of the person when the bottom of the person is not available in heavily occluded scenes. We verified the robustness of the method in different camera setups and for both single and multiple walking people. The results show that the triangulation error of a few centimeters can be obtained. Typically, it requires less than one minute of observing the walking people to reach this accuracy in controlled environments. It also just takes a few minutes to collect enough data for the calibration in uncontrolled environments. Our proposed method can perform well in various situations such as multi-person, occlusions, or even at real intersections on the street.

Keywords: extrinsic calibration; camera network; pedestrians;

1. Introduction

Extrinsic camera calibration provides the coordinate system transformations from 3D world coordinates to 3D camera coordinates for all the cameras in the network. This information is essential for many machine vision applications such as tracking, augmented reality, free view image synthesis, 3D reconstruction [1–3], or transferring the well-trained recognition models to different camera setups [4].

The classical methods [5–7] require a sufficient number of point correspondences of calibration objects to estimate the extrinsic parameters accurately. In addition, the calibration objects also have to be well-observed among all cameras. Moreover, calibrating cameras without any mistakes by using classical methods requires a certain level of skill while sending skilled technicians onsite to recalibrate cameras is costly and time-consuming. It also would be even worse because the cameras also need to be recalibrated after the cameras are adjusted or moved. Additionally, this traditional approach does not work for historic multi-camera video sequences in which no calibration objects were recorded.

Hartley et al. [8] proposed an auto-calibration method based on scene reconstruction from arbitrary features. Due to the interactive fashion as well as a large number of parameters has to be estimated, this method is slow and not always be able to achieve reliable results. Analyzing human data in images and video is the main concern of many machine vision applications. Thus, to leverage the information which also is extracted to serve other high-level tasks, many autocalibration methods [9–14] based on pedestrians are proposed. However, they are sensitive to noise as well as impractical for several situations in practice.

The proposed method relies on finding humans in images and estimating their centerline. For this purpose, we use OpenPose [15,16]. Human pose estimation is also an important task for various machine vision applications, such as action recognition, motion capture, sports, etc. Many real-time human estimation methods [15–21] have been proposed in recent years. Therefore, human pose estimation is fast enough for use in extrinsic calibration. Moreover, pose estimation is useful in itself for video analytics. Therefore, if an application already includes pose estimation, we can reuse this “for free” in calibration. Note that, in this paper, we assume the intrinsic camera parameters have been estimated before the extrinsic camera calibration.

The first contribution of our paper is that we replace the ellipse based detection of the top and the bottom of pedestrians by an approach based on modern human pose estimators. Thus, it provides more robust detection of people and a more accurate estimation of their centerlines. Then, we analyze the amount of video of walking people it needs to reach the desired accuracy. We also combine the proposed method calibration method with a random sampling strategy to deal with the noise and outliers in the estimated human pose data. It shortens the required time of data collection, and also makes our method robust to outliers and noise.

The second contribution is that we propose an automatic method to also handle the case of multiple pedestrians simultaneously in the scene. In [14], this case was handled by manual annotation. In the paper, we propose a brute-force, but still fast, method to effectively find the correspondences and also eliminate correspondences that have poor estimated human pose. We show that it produces accurate results and more complicated methods are not needed.

The third contribution, we proposed a novel extrinsic calibration method based on just the information of the top and the centerlines of the pedestrian. Thus, it helps the proposed calibration system can work well even without the information of the bottom of the pedestrians (which usually happens in heavily occluded scenes). Therefore, the proposed method can be applied to a wide range of usage scenarios, including indoor and outdoor scenes. The experimental results show that the proposed method can achieve very precise accuracy in many challenging scenarios including real intersection, heavily occluded, and multiple people scenes.

The rest of the paper is organized as follows. We discuss the related work in Section 1. In Section 2, we describe the architecture of our calibration method for a pair of cameras in detail. In Section 3, we explain the way to extend the proposed method for a camera network as well as the novel extrinsic calibration based on just the top and centerline of the walking people. We present the obtained results and the detailed analysis of our experiments in Section 4. Finally, we discuss the conclusion and future work in Section 5.

2. Related Work

Lv et al. [12,22] detect and select the walking human from video sequences by the transition of foreground object shapes. They represent the pedestrians as vertical “walking sticks” of the same height in the 3D environment. Then, they compute the vertical vanishing point and the horizon line based on the vertical “walking sticks”. Li et al. [11] proposed a single view camera calibration method that directly estimates the focal length, the tilting angle, and the camera height by using a nonlinear regression model from the observed head and feet points of a walking human.

In [9], Liu et al. proposed a fully automatic calibration method for monocular stationary cameras. They leverage relative 3D pedestrian height distribution to eliminate false pedestrian detections in moderately crowded scenes. In [10], Liu et al. extended their earlier work to camera network calibration. Iteratively, they incorporate robust matching with a partial direct linear transform. Due to the reliance on vanishing point (intersection of near-parallel lines) estimation, a small error of head (or feet) detection could lead to a big error of extrinsic parameters. On the other hand, our method estimates the extrinsic parameters based on estimated 3D positions of the head and feet which is much less sensitive to noise. Moreover, their methods cannot work if there is no feet information in the heavily occluded scene while the proposed method still performs well in that situation. In [23], Lucas et al. proposed a method for urban areas by combining the information from the pedestrians and structures that have parallel and orthogonal lines, such as buildings and road lines. Method [23] addresses the information of pedestrians in a similar manner as [10], and it is only applicable to single view applications.

Most methods [9,10,12] assume that moving pedestrians walk on a planar, horizontal surface. Possegger et al. [13] proposed an unsupervised extrinsic self-calibration method for a network of static cameras and pan-tilt-zoom cameras solely based on correspondences between tracks of a walking human. Then, they eliminate the outliers of feet and head detection by estimating pairwise homographies between the camera views based on the detected locations of feet and head. Finally, they compute the extrinsic parameters of the cameras by solving a non-linear optimization problem to minimize the reprojection error. Therefore, it tends to get stuck in local optima without a good initialization which was not presented in their work. In contrast, our method can have a precise estimated 3D position of the head and feet based on a robust human pose detector for the extrinsic calibration. Our method also does not require the person to walk on a plane surface.

Hödlmoser et al. [24] proposed a novel method to estimate the essential matrix based on the locations of the feet and head of the single pedestrian from the video sequences. After that, the extrinsic parameters were extracted by essential matrix decomposition. To find out the unique and proper solution for the rotation parameters and translation parameters, they needed to apply the chirality check [25,26]. Therefore, their method is quite sensitive to erroneous of the estimated locations between head (or foot) correspondences in different camera views. If the pedestrian walks along a straight line which occurs quite often in practice, all head and foot positions lie in the same plane. This leads to the degenerate case for the essential matrix estimation [27]. Therefore, they can only obtain the homography matrix between two views. However, in this case, their method cannot obtain the unique extrinsic parameters by decomposing the homography matrix [28,29]. Thus, it cannot find the extrinsic parameters with reasonable accuracy in this case.

Our paper is based on the work of Guan et al. [14]; our method does not require that the pedestrian walks on a plane surface (e.g., walking on steps and stairs), as long as the posture of the pedestrian remains the same while walking. The method obtains the extrinsic parameters by computing the 3D rigid body transformation that optimally aligns two sets of points corresponding to top and bottom of the pedestrian. The correspondence of these points between camera views is assumed to be known. In practice, this method, therefore, requires manual annotations to differentiate between multiple people and is not fully automatic. In [14], the top and bottom detection was implemented based on change detection and is not very robust w.r.t. noise and occlusion. In contrast, our propose a method is fully automated and uses a more robust human pose detector. In [30], Lettry et al. proposed a method

to solve correspondences for camera calibration based on multiple pedestrians. However, their method produces incorrect correspondences which degrade the accuracy of the calibration.

3. Proposed Method

Figure 1 shows the block diagram of the proposed framework of multi-camera calibration based on walking pedestrians. First, the proposed method calibrates all the cameras in the network in a pairwise fashion. Then, if the ground truth measurements are available (at least 3 points), we can apply the refinement and alignment to further optimize the extrinsic parameters. If the bottom of the pedestrians cannot be observed, we apply the novel method to estimate the extrinsic parameters based on just the top and the centerline of the walking person (Section 3.5). However, if the bottom of the pedestrians can be observed properly, we apply the extrinsic calibration method based on the top and the bottom that extracted from OpenPose [16] to have more accurate extrinsic parameters.

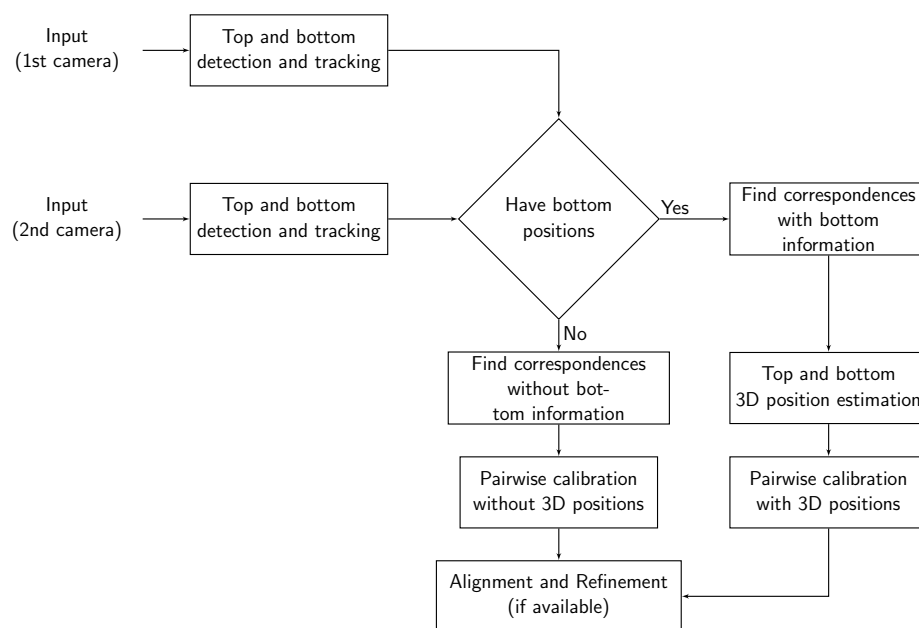


Figure 1. Block scheme of the proposed calibration method.

3.1. Extract the Positions of the Top and the Bottom of the the Observed Pedestrian in Image Coordinates

First, we assume that the frame synchronization, as well as the intrinsic calibration for all cameras in the network, have been done before the extrinsic calibration. To obtain the position of a walking person in the image, we apply the human pose estimation method in [16]. This produces a skeleton model of all major body joints. Because the locations of the head joints are not stable enough in these skeletons, we use the neck joint locations instead as the top of the observed pedestrian.

In this work, we compute the bottom of the observed pedestrian in two different ways. The first one is the midpoint of the left ankle joint and the right ankle joint as the bottom position of a walking person. However, the ankle of the walking person does not always appear in the mildly occluded scene. However, the hip's joints of the walking person could be well observed in the mildly occluded scene. Thus, we also investigate to extract the bottom part of the body by the middle point of the left hip joint and the right hip joint.

In the case that the bottom of the pedestrians cannot be observed, to determine the image positions of the top and the centerline of a walking person, we propose to detect the bounding boxes of the walking person (are extracted by YOLO [31]) in the first step. We estimate the centerline of a pedestrian by the line from the center of the top edge to the center bottom edge of the bounding box.

3.2. Extrinsic Camera Calibration Based on a Pedestrian

Consider a camera network with K cameras C_1, C_2, \dots, C_K . Let $\mathbf{r}^{(w)} = (X_w, Y_w, Z_w)^T$ be a point in a 3D world coordinate system that is visible to the camera, where the superscript T denotes a matrix transposition. In our work, each camera in the camera network has its own distinct camera coordinate system. We denote a 3D point in the coordinate system of camera k as $\mathbf{r}^{(k)} = (X^{(k)}, Y^{(k)}, Z^{(k)})^T$. Without loss of generality, we choose the coordinate system of the camera C_1 as the world coordinate system as follows:

$$\mathbf{r}^{(w)} = \mathbf{r}^{(1)}. \quad (1)$$

Thus, we can present the transformation between the camera coordinates $\mathbf{r}^{(k)}$ and the world coordinates $\mathbf{r}^{(w)}$ as follows:

$$\mathbf{r}^{(k)} = R^{(k)}\mathbf{r}^{(w)} + c^{(k)}, \quad (2)$$

where $c^{(k)}$ are the coordinates of the origin of the global world coordinate system with regards to the local coordinate system of camera k . $R^{(k)}$ is the rotation matrix which is a 3×3 matrix.

To obtain the extrinsic parameters for all cameras in the camera network, we first calibrate the camera network in a pairwise fashion. Thus, let us consider a camera system (which is composed of two cameras) where a person moving between N different locations while keeping a fixed posture (the bottom and the top of the person can be observed from both cameras). Let $\tilde{\mathbf{u}}_{\text{bottom}}^{(k)}(t)$ and $\tilde{\mathbf{u}}_{\text{top}}^{(k)}(t)$ be the image positions of the bottom (feet or hip) and top (neck) at the t -th locations in camera k (where $k \in \{1, 2\}$). Let $\tilde{\mathbf{x}}_{\text{bottom}}^{(k)}(t)$, and $\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t)$ be the normalized image coordinates $(x, y, 1)$ of the bottom and the top, respectively. We obtain the unknown Z coordinates of the bottom $Z_{\text{bottom}}^{(k)}(t)$ and Z coordinates of the top $Z_{\text{top}}^{(k)}(t)$ for camera k by applying the proposed method in [14]. Suppose that person walks upright and has height h where h is measured from the top of the pedestrian to the bottom of the pedestrian. Let $\mathbf{r}_{\text{top}}^{(k)}(t) = Z_{\text{top}}^{(k)}(t)\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t)$ and $\mathbf{r}_{\text{bottom}}^{(k)}(t) = Z_{\text{bottom}}^{(k)}(t)\tilde{\mathbf{x}}_{\text{bottom}}^{(k)}(t)$ be the 3D camera coordinates of the top and bottom. Thus, we have:

$$\mathbf{r}_{\text{top}}^{(k)}(t) - \mathbf{r}_{\text{bottom}}^{(k)}(t) = Z_{\text{top}}^{(k)}(t)\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t) - Z_{\text{bottom}}^{(k)}(t)\tilde{\mathbf{x}}_{\text{bottom}}^{(k)}(t) = h\mathbf{e}_z^{(k)}, \quad (3)$$

where $\mathbf{e}_z^{(k)}$ is unit vector of the person within camera k .

From $\tilde{\mathbf{x}}_{\text{bottom}}^{(k)}(t)$ and $\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t)$, it is possible to compute a 3D vector $\mathbf{m}^{(k)}(t) = \tilde{\mathbf{x}}_{\text{bottom}}^{(k)}(t) \times \tilde{\mathbf{x}}_{\text{top}}^{(k)}(t)$ which is perpendicular to the unique vertical plane containing the origin of camera k , $\tilde{\mathbf{x}}_{\text{bottom}}^{(k)}(t)$, and $\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t)$. At a given time instant, the intersection of all of those planes is a line along the vertical direction. We could also cancel h in $(\mathbf{m}^{(k)}(t))^T h\mathbf{e}_z^{(k)} = 0$ which leads to Equation (4) because $h\mathbf{e}_z^{(k)}$ is on the aforementioned plane.

$$(\mathbf{m}^{(k)}(t))^T \mathbf{e}_z^{(k)} = 0. \quad (4)$$

Therefore, $\mathbf{e}_z^{(k)}$ is determined by SVD of matrix $M^{(k)} = (\mathbf{m}^{(k)}(t))^T$. As explained in [14], once $\mathbf{e}_z^{(k)}$ is determined, we can compute the 3D locations of the of the bottom and top w.r.t. by the least-squares solutions up to a constant factor h . Finally, we apply the orthogonal Procrustes analysis [32] to estimate the rigid body transformation (relative camera pose in 3D space) between two sets of 3D points (top and bottom).

3.3. Robust Extrinsic Calibration

The proposed calibration method can obtain the appropriate extrinsic parameters when the number of inliers is large enough to compensate for the bad effect of the outliers. However, in practice,

we do not always have enough samples to have a precise calibration, e.g., calibrate camera network from videos that were recorded a long time ago. The accuracy of the calibration does not only depend on the number of the samples but also the distribution of locations. For example, using more random locations from different spots of the room helps the proposed method improve accuracy (Figure 2). However, the result of calibration from 20 s (approximately 300 locations) in Table 1 is worse than the result of the calibration from 20 random locations of the room (which is selected randomly from different spots of the scene) in Table 2. It is easy to understand because the person just slowly walks in the room, most of the locations collected in 20 s are so close to each other, and different cameras may see different points of the top or the bottom of the pedestrians. This means many of those locations provide similar information which is not so useful to improve the result of the calibration. Moreover, the outliers of the detection make the information around some spots of the room become inconsistency.

Table 1. The calibration results on Camera Network 1 (CN1) within 1000 experiments of the proposed method. $\delta\mathbf{r}^{(w)}$, $\delta\mathbf{u}^{(p)}$, $\delta\mathbf{u}^{(r)}$, and $\delta\mathbf{u}^{(rr)}$ denotes the triangulation error, projection error, reprojection error, and relative reprojection error, respectively.

Collecting Data Time (s)	10	20	30	40	50
Using all Locations					
$\delta\mathbf{r}^{(w)}$ (cm)	5.203	2.443	1.655	1.540	1.533
$\delta\mathbf{u}^{(p)}$ (pixel)	15.005	9.006	4.863	4.648	4.627
$\delta\mathbf{u}^{(r)}$ (pixel)	52.804	7.544	4.581	4.341	4.316
Random Samples					
$\delta\mathbf{r}^{(w)}$ (cm)	2.465	1.742	1.582	1.585	1.544
$\delta\mathbf{u}^{(p)}$ (pixel)	5.952	4.839	4.670	4.613	4.595
$\delta\mathbf{u}^{(r)}$ (pixel)	5.435	4.496	4.355	4.300	4.280

Table 2. Comparison between our method, the method of Guan et al. [14], and the method of Hödlmoser et al. [24] on Camera Network 1 (CN1), Camera Network 2 (CN2). We randomly select 20 locations of the pedestrians in the scene to calibrate the CN1, and CN2. $\delta\mathbf{r}^{(w)}$, $\delta\mathbf{u}^{(p)}$, $\delta\mathbf{u}^{(r)}$, and $\delta\mathbf{u}^{(rr)}$ denotes the triangulation error, projection error, reprojection error, and relative reprojection error, respectively. The bold numbers are the best results among the mentioned methods.

	Proposed Method (Feet)		Proposed Method (Hip)		Guan et al. [14]		Hödlmoser et al. [24]	
	CN1	CN2	CN1	CN2	CN1	CN2	CN1	CN2
$\delta\mathbf{r}^{(w)}$ (cm)	1.33	2.2	1.45	2.2	1.30	3.16	1.30	63.7
$\delta\mathbf{u}^{(p)}$ (pixel)	3.98	5.8	4.60	5.8	4.14	6.72	4.15	106.4
$\delta\mathbf{u}^{(r)}$ (pixel)	3.76	5.0	4.33	5.0	4.09	6.20	4.09	104.3
$\delta\mathbf{u}^{(rr)}$ —top (%)	1.8	1.7	1.8	1.7	1.9	7.0	1.9	43
$\delta\mathbf{u}^{(rr)}$ —bottom (%)	2.8	2.0	2.8	2.0	3.0	4.1	3.0	39

In [14], Guan et al. proposed a method based on RANSAC [33] to obtain a subset of top (or bottom) locations which are likely to be observed similarly in different views. However, the estimated 3D points which satisfy this criterion, do not always agree with the same unit vector $\mathbf{e}_z^{(k)}$. Furthermore, the pose of the walking people can slightly change during the video sequences which also makes the unit vector of the centerlines slightly changed. So, including all locations that are likely to be observed similarly cannot effectively improve the calibration results. It can also produce bad $\mathbf{e}_z^{(k)}$ estimation which leads to a bad extrinsic calibration. Instead, we need to find an optimal sparse subset of collected locations without the outliers to improve the performance of the proposed method on short video sequences (short amount of time to collect data). Therefore, we propose a random sample scheme as Algorithm 1 to obtain the sparse subset of collected locations which produces the most stable extrinsic calibration (low reprojection error).

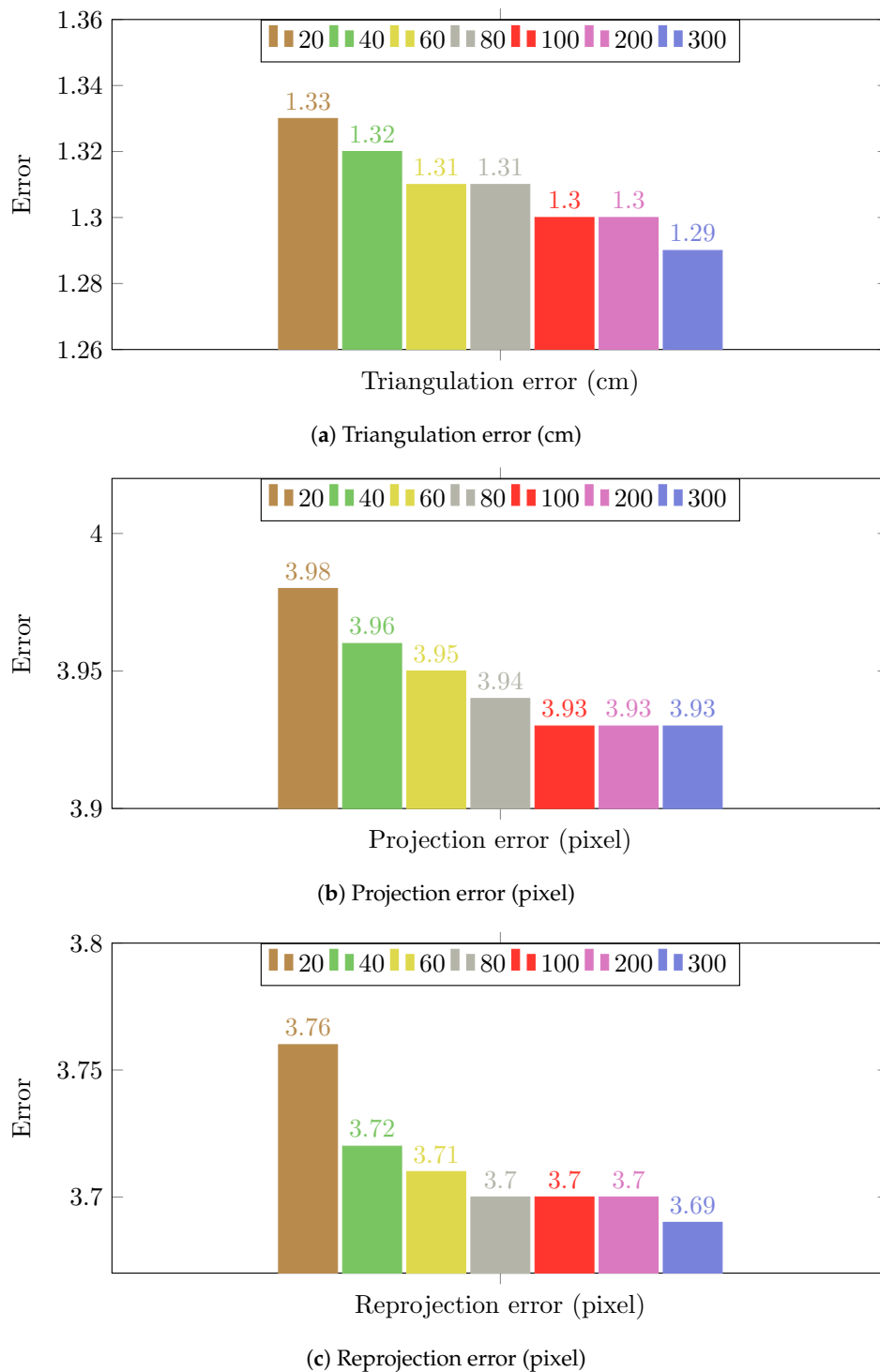


Figure 2. The calibration results of the proposed method on Camera Network 1 (multiple people walking in an empty room) by using different number of random locations. Each location yields two calibration samples (top and bottom).

Algorithm 1: The steps of random sample scheme to calibrate the extrinsic parameter of camera a and camera b

Input: $H^{(a)}$ and $H^{(b)}$ —lists of pair locations of top and bottom of the pedestrian in the video sequence, $L^{(ab)}$ —the list of frame indices of key locations (as in Algorithm 2), and the number of repetition M .

Output: the extrinsic parameters.

Step 1. For each frame index i in $L^{(ab)}$, we randomly select a location at frame t that is neighbor of it ($t - i < \epsilon_{time}$). In our experiment, we chose the $\epsilon_{time} = 10$. Then, we compute extrinsic parameters using the method that we presented in Section 3.2.

Step 2. Count the number of pairs agreeing with the extrinsic parameters (inliers). A pair is considered to agree with the extrinsic parameters if the reprojected error of that pair is smaller than the threshold ϵ_{error} :

Step 3. Repeat Steps 1 and 2 until the number of inliers reaches a certain threshold or the number of repetitions is greater than M .

Step 4. Choose the extrinsic parameters that has the highest number of the inliers (the most stable) based on the method that we presented in Section 3.2.

Algorithm 2: Determine the list of key pair locations

Input: $H^{(a)}$ and $H^{(b)}$ —lists of pair locations of top and bottom of a pedestrian i in the video sequence of camera a and camera b , respectively.

Output: $L^{(ab)}$ —frame indices of the key locations in the video sequence of camera a and camera b .

add 0 to $L^{(ab)}$.

assign the frame index of the last key location $k = 0$.

forall time steps t **do**

if $\|H^{(a)}(t) - H^{(a)}(k)\|_2 > \epsilon_{distance}$ **and** $\|H^{(b)}(t) - H^{(b)}(k)\|_2 > \epsilon_{distance}$ **then**
add t to $L^{(ab)}$.
assign the frame index of the last key location $k = t$

return frame indices of the key locations $L^{(ab)}$.

3.4. Automatically Estimates Extrinsic Parameters Based on Multiple Pedestrians

In [14], Guan et al. had to manually annotate the correspondences in the scene to calibrate the camera network. However, it is not really convenient for the customers to annotate the data, especially, if the number of cameras in the network is huge. Thus, we propose a simple and fast method to find the correspondences between different cameras. We use Openpose [16] to estimate 2D skeleton models of humans in the images. In practice, the estimated locations of the necks and feet (or hip) joints thus obtained are sometimes inconsistent between views (e.g., a different physical point is indicated the feet joints or hip joints in two views). The proposed calibration method is insensitive to this problem, as long as the number of observed skeletons is large enough. Otherwise, the results will be poor if people are observed in an insufficient number of locations (e.g., the method will fail if only a single person, always in the same position, is observed).

In controlled environments, these conditions can be easily enforced by providing instructions to the walking people. However, in uncontrolled environments with multiple pedestrians, we cannot order the people to walk by our instruction. In addition, people tend to pass the scene in a short amount of time such as walking along a straight line (insufficient number of locations). Hence, it is difficult to gather data in different locations of the scene for precise calibration.

To handle this problem, we propose an easy and robust brute-force method to solve the association problem (which pedestrian in one camera corresponds to an observation in another camera). First, we

apply a simple object matching algorithm based on feature matching to track the pedestrians for each camera. Let $H_i^{(k)} = \{(\tilde{\mathbf{u}}_{\text{bottom}}^{(k)}(m), \tilde{\mathbf{u}}_{\text{top}}^{(k)}(m)) \dots, (\tilde{\mathbf{u}}_{\text{bottom}}^{(k)}(n), \tilde{\mathbf{u}}_{\text{top}}^{(k)}(n))\}$ be the set of all locations of the top and bottom (feet or hip) of person i from frame m to frame n in camera k with $k \in \{a, b\}$.

Furthermore, let $H^{(k)} = \{H_0^{(k)}, H_1^{(k)}, \dots, H_q^{(k)}\}$ be a set of locations of the top and bottom of all pedestrians in the scene of camera k with q is the number of pedestrians in this scene. We compute all possible correspondences $C^{(ab)}$ between camera a and camera b by generating all pairs of elements from $H^{(a)}$ and $H^{(b)}$. Then, we calibrate the pair of cameras with each generated correspondences to estimate the matching rate as presented in the Algorithm 3. The matching rate represents the percentage of good reprojected results of the top and bottom positions of pedestrians where a good reprojected result means having the relative reprojection error less than *threshold* (equals 0.05 in our experiments).

Algorithm 3: Compute matching rate of the extrinsic parameters between camera a and camera b

```

total number of pairs  $n_{pairs} \leftarrow 0$ ;
total number of matched pairs  $n_{matched} \leftarrow 0$ ;
forall time steps  $t$  do
     $H^{(a)}(t) \leftarrow$  pairs of top and bottom locations of camera  $a$  at time step  $t$ ;
     $H^{(b)}(t) \leftarrow$  pairs of top and bottom locations of camera  $b$  at time step  $t$ ;
     $C^{(ab)}(t) \leftarrow$  combinations of  $H^{(a)}(t)$  and  $H^{(b)}(t)$ ;
    forall combination  $c$  in  $C^{(ab)}(t)$  do
         $matched(c) \leftarrow 0$ ;
        forall pair  $p$  of top and bottom locations in  $c$  do
            if reprojection error of  $p < threshold$  then
                 $matched(c)$  increased by 1;
         $length_a \leftarrow$  number of pairs in  $H^{(a)}(t)$ ;
         $length_b \leftarrow$  number of pairs in  $H^{(b)}(t)$ ;
         $n_{matched}$  increased by  $max_c(matched(c))$ ;
         $n_{pairs}$  increased by  $min(length_a, length_b)$ ;
return matching rate  $\leftarrow n_{matched} / n_{pairs}$ ;

```

Finally, the top highest matching rate correspondences are selected to calibrate the pair of cameras. Thus, the difficulty of gathering data in uncontrolled environments also be solved by combining the sample locations of the highest matching rate correspondences from different spots of the scene. To avoid a combinatorial increase in the number of computations, the frames with too many pedestrians are removed. Moreover, the frames with too many pedestrians also have much worse pose estimation results as well as poor human tracking results. This makes it is not useful already to use these frames. Also, using the frames that have a very high number of people does not improve the accuracy of the calibration. Therefore, in practice, we can calibrate the camera network from the parts of video sequences which has a low enough number of pedestrians. In our experiments, we choose 5 pedestrians as the threshold, therefore the number of combination of correspondences for each frame is always less than 5! which does not take too much time to verify all possibilities. The brute-force approach can definitely be replaced by a random sampling scheme to make it runs faster. However, it may skip many appropriate correspondences for the calibration. Therefore, we decide to keep the brute-force approach to maintain the accuracy of the proposed method which is much more important for many applications in practice.

3.5. Extrinsic Calibration When Feet and Hip Joints Are Not Available

In Section 3.2 we proposed to estimate the normal vector using 2D image positions of the top and bottom pedestrians. As the feet and hip positions are not known, we propose to use line positions to do the estimation instead. Suppose a person moves to N different positions while keeping a fixed posture. Suppose that all cameras see the top of the person. At each time t , we first calculate $\tilde{\mathbf{u}}_{\text{top}}^{(k)}(t)$ and the centerline using the technique described above for camera k . We denote the projection of the centerline in the image as

$$y = a^k(t)x + b^k(t). \quad (5)$$

Select two different points on that line and denote the normalized homogeneous image coordinates as $(x_1^{(k)}, ax_1^{(k)}, 1)^T$ and $(x_2^{(k)}, ax_2^{(k)}, 1)^T$. The cross product of these two vectors is $(a^k(t)(x_1^{(k)} - x_2^{(k)}), x_2^{(k)} - x_1^{(k)}, b^k(t)(x_1^{(k)} - x_2^{(k)}))^T$. By canceling out $(x_1^{(k)} - x_2^{(k)})$, we define $\mathbf{m}^{(k)}(t) = (-a^{(k)}(t), 1, -b^{(k)}(t))^T$, which is the normal of the plane spanned by the camera center and the center line of the pedestrian. It is obvious that $\mathbf{e}_z^{(k)}$ is on that plane, so that $\mathbf{m}^{(k)}(t)$ and $\mathbf{e}_z^{(k)}$ are orthogonal as Equation (4).

Therefore, $\mathbf{e}_z^{(k)}$ can be determined in the same fashion as presented in Section 3.2 by using SVD decomposition of $(\mathbf{m}^{(k)}(t))^T$: $\mathbf{e}_z^{(k)}$ is the singular vector which is corresponding to the lowest singular value. $\mathbf{e}_z^{(k)}$ is also the normal of the plane composed by all the top positions when the person walks upright on a flat horizontal surface. Since all top positions of a single pedestrian lie on a plane when the person walks on a horizontal surface, there is the homography between any two views of the plane. Various methods [28,29,34] have been proposed to estimate the homography between two views given at least 4 non-collinear corresponding points and solve the structures from motion problem by decomposing the homography matrix. Weng et al. [28] produces two solutions of the extrinsic parameters by decomposing the homography matrix. However, the unique and proper solution cannot be obtained without additional knowledge of the scene. Here we propose a method to obtain the unique extrinsic parameters by decomposing the homography matrix with the estimated $\mathbf{e}_z^{(k)}$.

Pairwise Planar Homography: as we select the coordinate frame of camera 1 as the world coordinate frame, we have

$$\mathbf{r}_{\text{top}}^{(k)}(t) = R^{(k)}\mathbf{r}_{\text{top}}^{(1)}(t) + \mathbf{c}^{(k)} \quad (6)$$

In the coordinate system of camera 1, the top of the pedestrian lies on the plane

$$(\mathbf{n}^{(1)})^T \mathbf{r}_{\text{top}}^{(1)}(t) = d^{(1)} \Leftrightarrow \frac{1}{d^{(1)}} (\mathbf{n}^{(1)})^T \mathbf{r}_{\text{top}}^{(1)}(t) = 1, \quad (7)$$

with $d^{(1)}$ is the Euclidean distance between the center of camera 1 and the plane, and $\mathbf{n}^{(1)}$ is the normal vector of the plane w.r.t the camera coordinate system of camera C_1 . Substituting Equation (7) in Equation (6) gives

$$\mathbf{r}_{\text{top}}^{(k)}(t) = R^{(k)}\mathbf{r}_{\text{top}}^{(1)}(t) + \mathbf{c}^{(k)} \frac{1}{d^{(1)}} (\mathbf{n}^{(1)})^T \mathbf{r}_{\text{top}}^{(1)}(t) \quad (8)$$

$$= \left(R^{(k)} + \mathbf{c}^{(k)} \frac{1}{d^{(1)}} (\mathbf{n}^{(1)})^T \right) \mathbf{r}_{\text{top}}^{(1)}(t). \quad (9)$$

Since $\mathbf{r}_{\text{top}}^{(k)}(t) = Z_{\text{top}}^{(k)}(t)\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t)$, and $\mathbf{r}_{\text{top}}^{(1)}(t) = Z_{\text{top}}^{(1)}(t)\tilde{\mathbf{x}}_{\text{top}}^{(1)}(t)$, Equation (9) leads to

$$\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t) \sim H^{(k)}\tilde{\mathbf{x}}_{\text{top}}^{(1)}(t), \quad (10)$$

where

$$H^{(k)} = \left(R^{(k)} + \mathbf{c}^{(k)} \frac{1}{d^{(1)}} \left(\mathbf{n}^{(1)} \right)^T \right), \quad (11)$$

which is the homography matrix. Since the vectors from both sides of Equation (11) are parallel, their cross product is zero, which leads to

$$\tilde{\mathbf{x}}_{\text{top}}^{(k)}(t) \times \left(H^{(k)} \tilde{\mathbf{x}}_{\text{top}}^{(1)}(t) \right) = 0, \quad (12)$$

with \times representing the cross product.

Notice that $H^{(k)}$ depends only upon eight independent coefficients, i.e., the three rotation coefficients, the three coordinates of the translation, and the two parameters representing the orientation of the plane. Each point imposes two independent constraints on $H^{(k)}$, so we need at least four corresponding points (top positions) to solve uniquely for $H^{(k)}$. The four points should be in a general configuration in the plane (no three of them are collinear). Notice that $H^{(k)}$ can be recovered up to a scale factor, so we get the homography matrix in the form of

$$H_e^{(k)} = \lambda \left(R^{(k)} + \mathbf{c}^{(k)} \frac{1}{d^{(1)}} \left(\mathbf{n}^{(1)} \right)^T \right). \quad (13)$$

Decomposition of the Homography Matrix: Once $H_e^{(k)}$ is obtained, we now discuss how to get extrinsic parameters by decomposing the homography matrix. It has been proven in the literature [28,29] that decomposing the homography matrix will give two candidate solutions for the extrinsic parameters. We will give detail about how to get the unique extrinsic parameters by using additional information.

Ma et al. [35] gave four solutions of the extrinsic parameters by SVD of the homography matrix, only two of which satisfy the positive depth constraint (the Z coordinate of the normal vector of the plane need to be positive, since the camera can see only points that are in front of it). Decomposition of the homography matrix proceeds as follows:

1. Normalization of the homography matrix. The normalized homography matrix is computed as

$$H^{(k)} = H_e^{(k)} / \sigma_2 \left(H_e^{(k)} \right), \quad (14)$$

where $\sigma_2 \left(H_e^{(k)} \right)$ is the second largest singular value of $H_e^{(k)}$.

2. Compute the SVD of $\left(H^{(k)} \right)^T H^{(k)}$ as

$$\left(H^{(k)} \right)^T H^{(k)} = \left(V^{(k)} \right)^T S^{(k)} V^{(k)}. \quad (15)$$

Define $\mathbf{v}_1^{(k)}, \mathbf{v}_2^{(k)}, \mathbf{v}_3^{(k)}$ as the three column vectors of $V^{(k)}$, and $\sigma_1^{(k)}, \sigma_2^{(k)}, \sigma_3^{(k)}$ as the eigenvalues of $H^{(k)}$. Compute two more unit vectors by

$$\mathbf{u}_1^{(k)} = \frac{\sqrt{1 - \left(\sigma_3^{(k)} \right)^2} \mathbf{v}_1^{(k)} + \sqrt{\left(\sigma_1^{(k)} \right)^2 - 1} \mathbf{v}_3^{(k)}}{\sqrt{\left(\sigma_1^{(k)} \right)^2 - \left(\sigma_3^{(k)} \right)^2}} \quad (16)$$

and

$$\mathbf{u}_2^{(k)} = \frac{\sqrt{1 - \left(\sigma_3^{(k)} \right)^2} \mathbf{v}_1^{(k)} - \sqrt{\left(\sigma_1^{(k)} \right)^2 - 1} \mathbf{v}_3^{(k)}}{\sqrt{\left(\sigma_1^{(k)} \right)^2 - \left(\sigma_3^{(k)} \right)^2}}. \quad (17)$$

3. Define matrices

$$\begin{aligned}
U_1^{(k)} &= [\mathbf{v}_2^{(k)}, \mathbf{u}_1^{(k)}, \mathbf{v}_2^{(k)} \times \mathbf{u}_1^{(k)}], \\
U_2^{(k)} &= [\mathbf{v}_2^{(k)}, \mathbf{u}_2^{(k)}, \mathbf{v}_2^{(k)} \times \mathbf{u}_2^{(k)}], \\
W_1^{(k)} &= [H\mathbf{v}_2^{(k)}, H\mathbf{u}_1^{(k)}, (H\mathbf{v}_2^{(k)}) \times (H\mathbf{u}_1^{(k)})], \\
W_2^{(k)} &= [H\mathbf{v}_2^{(k)}, H\mathbf{u}_2^{(k)}, (H\mathbf{v}_2^{(k)}) \times (H\mathbf{u}_2^{(k)})].
\end{aligned} \tag{18}$$

Finally the four possible solutions of extrinsic parameters and normal vector of the plane are calculated as

$$\begin{aligned}
R_1^{(k)} &= W_1^{(k)} (U_1^{(k)})^T, \\
\mathbf{n}_1^{(1)} &= \mathbf{v}_2^{(k)} \times \mathbf{u}_1^{(k)}, \\
\mathbf{c}_1^{(k)} \frac{1}{d^{(1)}} &= (H - R_1^{(k)}) \mathbf{n}_1^{(1)},
\end{aligned} \tag{19}$$

$$\begin{aligned}
R_2^{(k)} &= W_2^{(k)} (U_2^{(k)})^T, \\
\mathbf{n}_2^{(1)} &= \mathbf{v}_2^{(k)} \times \mathbf{u}_2^{(k)}, \\
\mathbf{c}_2^{(k)} \frac{1}{d^{(1)}} &= (H - R_2^{(k)}) \mathbf{n}_2^{(1)},
\end{aligned} \tag{20}$$

$$\begin{aligned}
R_3^{(k)} &= R_1^{(k)}, \\
\mathbf{n}_3^{(1)} &= -\mathbf{n}_1^{(1)}, \\
\mathbf{c}_3^{(k)} \frac{1}{d^{(1)}} &= -\mathbf{c}_1^{(k)} \frac{1}{d^{(1)}},
\end{aligned} \tag{21}$$

$$\begin{aligned}
R_4^{(k)} &= R_2^{(k)}, \\
\mathbf{n}_4^{(1)} &= -\mathbf{n}_2^{(1)}, \\
\mathbf{c}_4^{(k)} \frac{1}{d^{(1)}} &= -\mathbf{c}_2^{(k)} \frac{1}{d^{(1)}}.
\end{aligned} \tag{22}$$

Notice that only two solutions of the above four solutions satisfy the positive depth constraint (i.e., the third coordinate of the normal vector for the plane should be positive since the camera can only see points that are in front of it). So either Equation (19) or Equation (21) is the possible solution. Similarly, either Equation (20) or Equation (22) is the possible solution. Once two possible solutions are obtained, we obtain the correct one as follows. From Equation (4), we estimate $\mathbf{e}_z^{(1)}$, which is the normal of the plane composed by all the top positions of the pedestrian. $\mathbf{e}_z^{(1)}$ should be equal to the estimated $\mathbf{n}^{(1)}$ if there is no noise in the data. In practice, it rarely happens due to all kinds of noise. Thus we propose to use the angle between $\mathbf{e}_z^{(1)}$ and $\mathbf{n}^{(1)}$ as a criterion to get the correct $\mathbf{n}^{(1)}$ and the corresponding rotation and translation parameters. The correct solution should have a smaller angle to $\mathbf{e}_z^{(1)}$. Finally, to calibrate the camera network with the torsors of multiple pedestrians, we choose the highest matching rate correspondence for the calibration. In this case, we use the detected bounding boxes of the pedestrians which are extracted by YOLO [31] to determine the top and bottom of the pedestrians. We choose the center of the top edge and the center bottom edge of the bounding boxes as top and bottom of the pedestrians.

3.6. Joint Extrinsic Refinement for All Cameras in the Network

In Sections 3.2–3.4, we present the extrinsic calibration for the camera network in pairwise fashion. However, the extrinsic parameters of the cameras are obtained based on algebraic distance minimization without taking the property of the projective geometry of the cameras into account. Thus, it increases the triangulation error and projection error when we combine the information from all available cameras. To solve this problem, we jointly refine the extrinsic parameters of all cameras by minimizing the total reprojection error based on the method proposed in [14,36]). The objective function defines by the mean-squared discrepancy between the observed image positions of bottom and top of the pedestrian, and their reprojections. Finally, we optimize the extrinsic parameters by an iterative gradient descent procedure.

4. Experimental Results

4.1. Performance Measures

In order to evaluate the performance of our method with ground truth points, we compute the triangulation error ($\delta \mathbf{r}^{(w)}$), projection error ($\delta \mathbf{u}^{(p)}$), and reprojection error ($\delta \mathbf{u}^{(r)}$) [14]. In practice, the ground truth points are not always available to measure the performance of the calibration. Thus, we can only measure the calibration by computing reprojection error based on the top (or the bottom) positions of detected pedestrians. However, different cameras have different resolutions. Moreover, the height of the pedestrians at different locations in an image is different. Let N be the number of ground truth 3D sample points and K is the number of cameras in the network. We define the relative reprojection error as follows:

$$\delta \mathbf{u}^{(rr)} = \frac{1}{MK} \sum_{m=1}^M \sum_{k=1}^K \frac{\|\mathbf{u}_{mk} - \hat{\mathbf{u}}_{mk}^{(rr)}\|_2}{h_{mk}}, \quad (23)$$

where N be the number of ground truth 3D sample points, K is the number of cameras in the network, M is the number of pedestrians, h_m is the height in the image of person m -th in the camera k -th. \mathbf{u}_{mk} is the observed pixel coordinates of the top (or the bottom) of person m -th in the camera k -th. $\hat{\mathbf{u}}_{mk}^{(rr)}$ is the estimated location of the top (or the bottom), which are obtained through reprojection.

4.2. Calibration with Controlled Environment

Calibration with single person. We evaluate our method with a multi-camera tracking system composed of four side-view cameras. For simplicity, we call it Camera Network 1. The cameras were mounted at a height of about 3 m at each corner of a room (8.6 m by 4.8 m). The resolutions of the all videos are 780 by 580 pixel (Figure 3). We obtain the intrinsic parameters by [7]. We compare our method to the calibration method of Hödlmoser et al. [24] and Guan et al. [14].

Figure 3 shows an example of the detected bottom and top positions of the pedestrians of the person in a scene. For single person case, we apply the refinement method which proposed in [14] to obtain the final extrinsic parameters (Table 2). Table 2 shows that our method has slightly more accurate results than state-of-the-art methods. The differences in accuracy among different methods are small because this case is the simplest situation. Therefore, most of the existing methods can achieve very high accuracy on these video sequences.

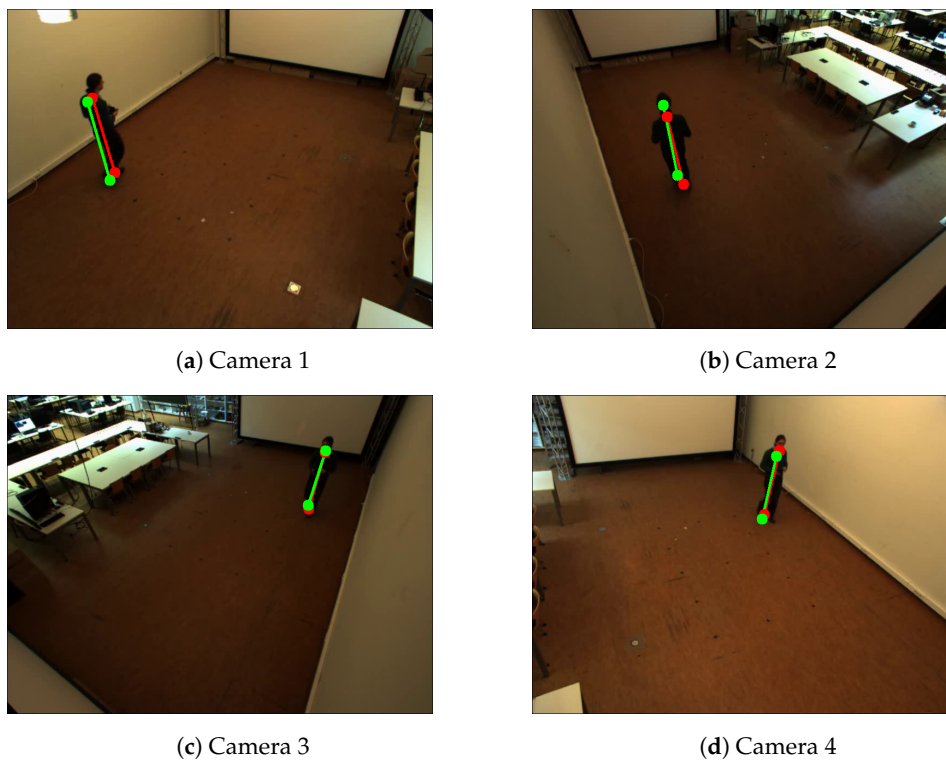


Figure 3. Example of detected the bottom and top positions of the pedestrians of Camera Network 1—single person walking in an empty room. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

In addition, the proposed method also requires a very short amount of time to collect the sample data for the calibration. As shown in Table 3, to collect enough data for the calibration, the person only needs to walk around the room with the total accumulated moving distance is around 20 m (by normal walking speed). This is the distance of walking around the room 2–3 times. Thus, the proposed method is fast and convenient for users to calibrate camera networks.

Table 3. Success percentages (the triangulation error is below 15 cm) within 1000 experiments of the proposed method for the Camera Network 1.

Accumulated Moving Distance (cm)	650	1000	2000	2500	4000
Successful Percentage	0.35	0.83	0.97	0.99	1.0

Calibration based on a single person in a mildly occluded scene. In order to show that our method works in a complex real-life environment room setup, we also evaluate our method on a setup with three cameras in a kitchen (Figure 4). The cameras were mounted at a height of about 2 m at different corners of a room. The resolutions of the videos are 640 by 480 pixels. We call it Camera Network 2 for simplicity. Table 2 shows that our method outperforms the method proposed by Guan et al. [14]. Note that we also recorded a scene where the person was cleaning the kitchen floor. Despite the movement and occlusions while cleaning the floor, the proposed method still produces a reasonable accuracy (the triangulation error is less than 10 cm) as shown in Tables 4 and 5. It shows the stability and robustness of our method to the occlusion.



Figure 4. Example of detected the bottom and top positions of the pedestrians of Camera Network 2—single person doing household work in an kitchen room. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

Table 4. The calibration results on Camera Network 2 (CN2) within 1000 experiments of the proposed method. The table shows the results of our method on video sequence where the subject was cleaning the room. $\delta \mathbf{r}^{(w)}$, $\delta \mathbf{u}^{(p)}$, $\delta \mathbf{u}^{(r)}$, and $\delta \mathbf{u}^{(rr)}$ denotes the triangulation error, projection error, reprojection error, and relative reprojection error, respectively.

Collecting Data Time (s)	10	20	30	40	50
Using all Locations					
$\delta \mathbf{r}^{(w)}$ (cm)	17.979	14.245	11.940	10.448	11.198
$\delta \mathbf{u}^{(p)}$ (pixel)	214.329	182.073	364.136	115.872	165.385
$\delta \mathbf{u}^{(r)}$ (pixel)	196.218	129.085	289.050	104.330	90.784
Random Samples					
$\delta \mathbf{r}^{(w)}$ (cm)	5.320	4.511	3.041	3.243	2.912
$\delta \mathbf{u}^{(p)}$ (pixel)	35.694	27.482	13.196	13.649	12.623
$\delta \mathbf{u}^{(r)}$ (pixel)	37.039	36.265	12.655	12.592	12.099

Table 5. The calibration results on Camera Network 2 (CN2) within 1000 experiments of the proposed method. The table shows the results of our method on video sequence where the subject was walking in the room. $\delta\mathbf{r}^{(w)}$, $\delta\mathbf{u}^{(p)}$, $\delta\mathbf{u}^{(r)}$, and $\delta\mathbf{u}^{(rr)}$ denotes the triangulation error, projection error, reprojection error, and relative reprojection error, respectively.

Collecting Data Time (s)	10	20	30	40	50
Using All Locations					
$\delta\mathbf{r}^{(w)}$ (cm)	4.499	3.602	2.689	2.504	2.456
$\delta\mathbf{u}^{(p)}$ (pixel)	37.253	22.162	15.705	10.922	9.818
$\delta\mathbf{u}^{(r)}$ (pixel)	39.925	21.332	13.780	10.096	8.927
Random Samples					
$\delta\mathbf{r}^{(w)}$ (cm)	2.618	2.224	2.144	2.169	2.161
$\delta\mathbf{u}^{(p)}$ (pixel)	14.634	11.331	9.198	7.933	7.156
$\delta\mathbf{u}^{(r)}$ (pixel)	15.643	11.242	8.783	7.412	6.567

Robust extrinsic calibration. In Table 2, we present the calibration results based on different locations in the scene to evaluate the performance of the proposed method in a general sense. However, in practice, we have to collect the data consecutively rather than select arbitrary locations in space.

Thus, we also conduct another experiment of camera calibration based on a segment with regards to the amount of time that we use to collect data. Table 6 shows the successful rate to have accurate extrinsic parameters with the proposed method with all samples and a random-sample scheme. We apply the refinement method which proposed in [14] to obtain the final extrinsic parameters in Table 6. While cleaning the floor, we cannot keep one posture from the beginning until the end. Typically, we have to lean forward or to kneel for cleaning different places in the room. Thus, the result of our method by using all available locations on CN2—cleaning the floor sequences (Table 4) are degraded dramatically. By applying the random sample scheme, the proposed method can eliminate the locations produced from improper postures. Therefore, this improves both the successful percentage and the accuracy of the calibration. Tables 1, 5 and 6 also show that the random sample scheme improves the successful percentage of the calibration and the accuracy of the calibration in the case that the pedestrian was walking with the same posture.

Table 6. Success percentages (the triangulation error is below 15 cm) within 1000 experiments of the proposed method which is conducted with the Camera Network 1 (CN1), and Camera Network 2 (CN2).

Collecting Data Time (s)	10	20	30	40	50
All locations					
CN1—Successful Percentage	0.896	0.972	0.995	1.0	1.0
CN2 (cleaning the Floor)—Successful Percentage	0.339	0.489	0.539	0.62	0.607
CN2 (walking)—Successful Percentage	0.836	0.929	0.980	1.0	1.0
Random Samples					
CN1—Successful Percentage	0.975	0.998	1.0	1.0	1.0
CN2 (Cleaning the Floor)—Successful Percentage	0.590	0.736	0.786	0.807	0.868
CN2 (Walking)—Successful Percentage	0.910	0.978	0.999	1.0	1.0

Calibration with multiple walking people. We evaluate our calibration method on the EPFL-Terrace sequences [37], which is a public multi-camera pedestrian video dataset (Figure 5). This dataset includes two sequences and 7 subjects, which were shot outside our building on a terrace with four DV cameras. In this paper, we call it Camera Network 3 for simplicity.

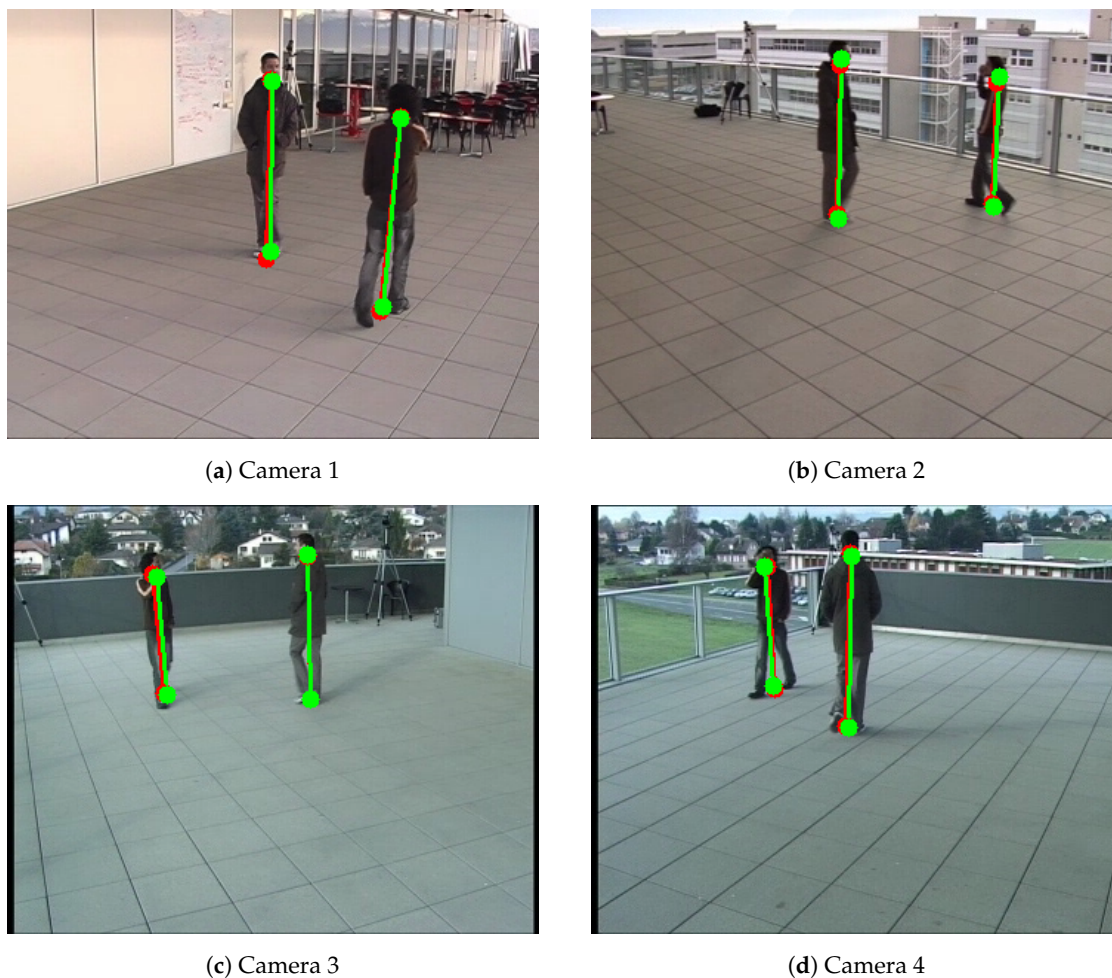


Figure 5. Example of detected the bottom and top positions of the pedestrians of Camera Network 3—EPFL-Terrace dataset [37]. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

It only takes approximately 270 s and 210 s on EPFL-Terrace dataset and CN3 to solve the correspondences and obtain the extrinsic parameters, respectively (we implemented the code to run on Intel(R) Core(TM) i7-8086K CPU @ 4.00GHz with Python 3). Table 7 also shows very high accuracy results for very challenging multiple people sequences. It proves the proposed scheme to correspondences in the video sequences can deal with outliers and failed tracking results to provide very appropriate pairs of correspondences.

Table 7. The calibration results on EPFL-Terrace (CN3) and Camera Network 5 (CN5). For EPFL-Terrace (CN3) and Camera Network 5 (CN5), we apply the proposed method on the first half of the video to obtain the extrinsic parameters. $\delta \mathbf{r}^{(w)}$, $\delta \mathbf{u}^{(p)}$, $\delta \mathbf{u}^{(r)}$ are not available for these sequences. Thus, the $\delta \mathbf{u}^{(rr)}$ denotes relative reprojection error is the only one available for these sequences.

	Proposed Method (Feet)		Proposed Method (Hip)	
	CN3	CN5	CN3	CN5
$\delta \mathbf{u}^{(rr)}$ —top (%)	2.1	12.6	2.1	12.6
$\delta \mathbf{u}^{(rr)}$ —bottom (%)	2.3	17.2	2.1	12.6

Comparison between calibration based on single walking person and calibration based on multiple walking. We evaluate the proposed camera calibration based on multiple pedestrians by calibrating the Camera Network 1 with an extra sequence to compare with the result of the calibration based on a single person. This sequence was shot with 3 subjects walking at the same time in the empty room (Figure 6). We calibrated Camera Network 4—CN4 (Figures 7 and 8) to compare the performance of the calibration in both single person and multiple people situations. This network has five cameras, three of which are mounted at a height of about 3 m, and two cameras were mounted at a height of around 2 m. The resolution of all cameras is 780 by 580 pixels. In the experiments, we also change the orientation and locations of the cameras in the Camera Network 4 to verify the performance of the proposed method from different viewpoints.

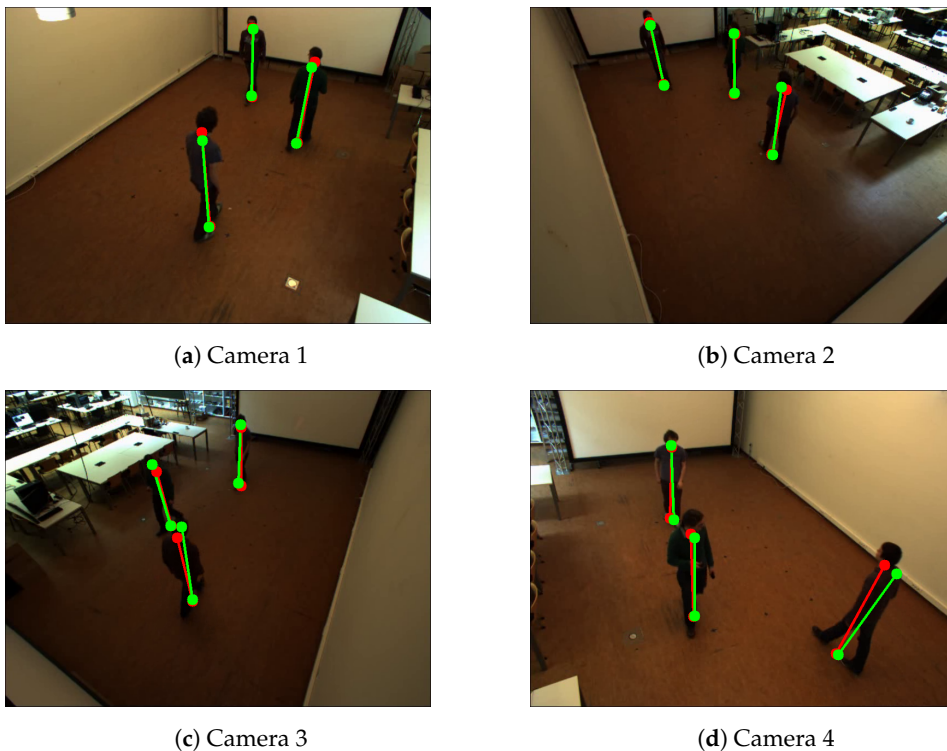


Figure 6. Example of detected the bottom and top positions of the pedestrians of Camera Network 1—multiple people walking in an empty room. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

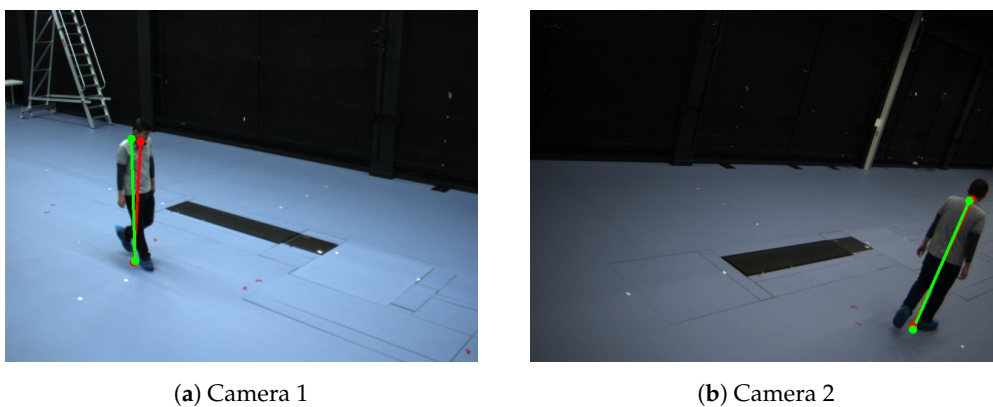
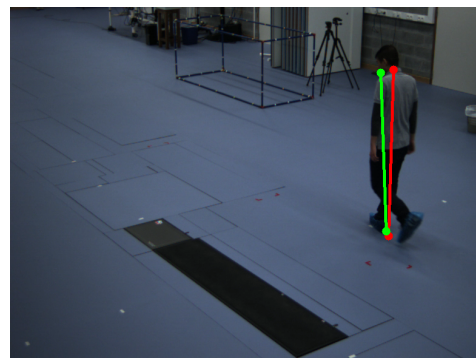


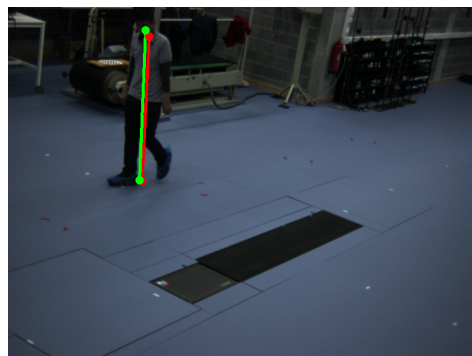
Figure 7. *Cont.*



(c) Camera 3

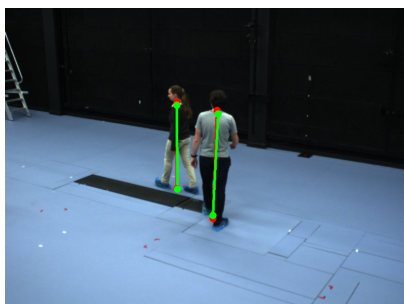


(d) Camera 4

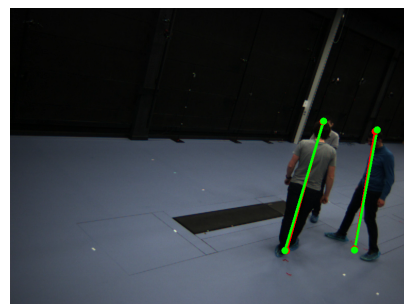


(e) Camera 5

Figure 7. Example of detected the bottom and top positions of the pedestrians of Camera Network 4—single person in an empty room. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.



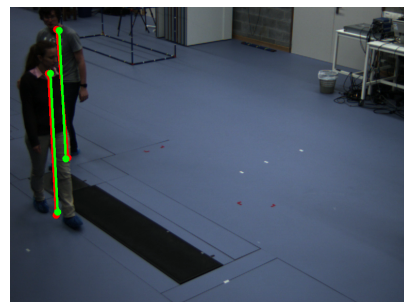
(a) Camera 1



(b) Camera 2

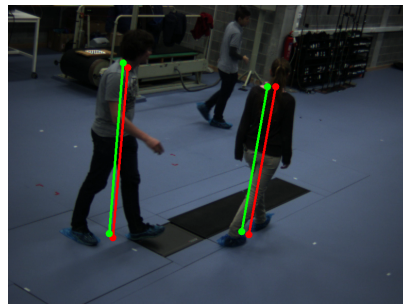


(c) Camera 3



(d) Camera 4

Figure 8. *Cont.*



(e) Camera 5

Figure 8. Example of detected the bottom and top positions of the pedestrians of Camera Network 4—single person in an empty room. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

Table 2 and Table 8 show the results of the proposed method in the case that there is only one subject in the scene of Camera Network 1 and Camera Network 4, respectively. Table 9 shows the results of the proposed method in multiple walking people case which is reasonably close to the calibration results in the single person case. It shows that the proposed method works well in both cases.

Table 8. Calibration results of camera calibration based on single person of the Camera Network 4. $\delta\mathbf{r}^{(w)}$, $\delta\mathbf{u}^{(p)}$, $\delta\mathbf{u}^{(r)}$, and $\delta\mathbf{u}^{(rr)}$ denotes the triangulation error, projection error, reprojection error, and relative reprojection error, respectively.

	Proposed Method (Feet)					Proposed Method (Hip)				
	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5
$\delta\mathbf{r}^{(w)}$ (cm)	5.14	3.88	5.22	4.92	8.52	5.60	6.71	6.99	7.66	8.09
$\delta\mathbf{u}^{(p)}$ (pixel)	6.04	5.52	6.31	6.21	10.67	6.35	8.39	8.92	9.84	10.86
$\delta\mathbf{u}^{(r)}$ (pixel)	2.21	3.40	2.09	2.27	4.74	2.10	3.71	3.96	3.53	5.55
$\delta\mathbf{u}^{(rr)}$ —top (%)	2.3	2.2	2.4	2.7	3.1	2.8	2.2	2.6	2.7	2.7
$\delta\mathbf{u}^{(rr)}$ —bottom (%)	1.7	2.8	1.9	2.3	2.5	2.2	2.5	2.1	3.1	3.3

Table 9. Calibration results of camera calibration based on multiple walking people of the Camera Network 1 and the Camera Network 4. Note that, for Camera Network 1, we also apply the refinement method which proposed in [14] to obtain the final extrinsic parameters. $\delta\mathbf{r}^{(w)}$, $\delta\mathbf{u}^{(p)}$, $\delta\mathbf{u}^{(r)}$, and $\delta\mathbf{u}^{(rr)}$ denotes the triangulation error, projection error, reprojection error, and relative reprojection error, respectively.

	Proposed Method (Feet)						Proposed Method (Hip)					
	CN1	S1	S2	S3	S4	S5	CN1	S1	S2	S3	S4	S5
$\delta\mathbf{r}^{(w)}$ (cm)	4.75	7.79	6.49	5.13	5.84	10.24	1.45	5.84	7.85	8.15	9.37	7.25
$\delta\mathbf{u}^{(p)}$ (pixel)	5.23	10.52	7.53	6.22	6.55	11.58	4.61	7.09	9.20	10.03	10.66	8.71
$\delta\mathbf{u}^{(r)}$ (pixel)	3.08	5.37	1.88	2.49	1.57	2.88	4.33	3.40	3.85	3.73	3.58	3.93
$\delta\mathbf{u}^{(rr)}$ —top (%)	2.1	5.8	2.2	2.9	2.7	6.4	2.5	3.5	2.8	4.4	3.4	3.3
$\delta\mathbf{u}^{(rr)}$ —bottom (%)	2.2	5.8	1.8	2.0	2.0	5.7	2.3	2.6	1.7	2.5	2.2	2.2

In general, using the feet to calibrate the camera network produce much better results in single-person cases because the $h\mathbf{e}_z^{(k)}$ vector is much longer which makes centerline vector estimation becomes more consistent. In the single person case, we can collect the data of the walking person continuously. Even in the case of occlusion, we can easily associate the data because they belong to only

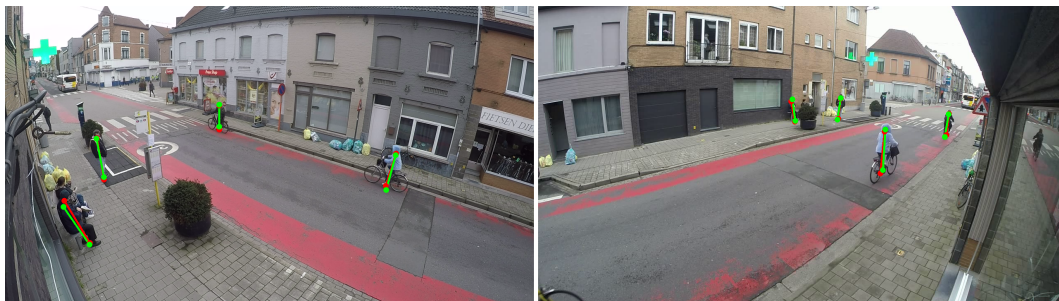
one person. However, in multiple pedestrian case, the data of one person normally are scattered and labeled with different indices due to the occlusion and tracking failures. Thus, the number of useful samples that we could extract in the same amount of time can be reduced. However, the proposed extrinsic calibration method based on multiple walking people in Section 3.4 still obtains the accuracy which is close to the single person case (Tables 8 and 9). Finally, using hip joints also improves the calibration results where the bottom parts of the pedestrians are hard to observe (e.g., S1 and S5 in Table 9).

4.3. Calibration with Uncontrolled Environment

Calibration at intersections. To show that our method can be applied to a real-life situation, we also recorded several video sequences at an intersection in Ghent to evaluate the proposed method (Figure 9). The pedestrians in this scene are quite small (about 60 pixels height). We call it Camera Network 5 for simplicity.



(a) Camera 0 and Camera 1, respectively



(b) Camera 2 and Camera 3, respectively



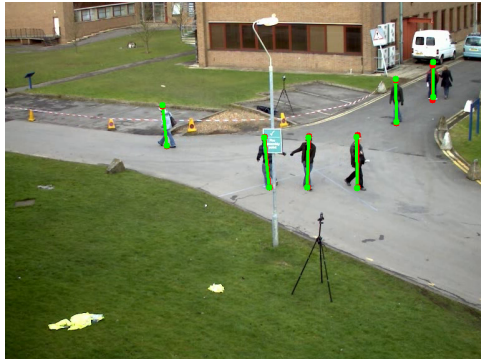
(c) Camera 4 and Camera 5, respectively

Figure 9. Example of detected the bottom and top positions of the pedestrians of Camera network 5—multiple pedestrians at an intersection. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

Table 2 shows our method has reasonably low error among different circumstances. However, in the intersection case, the pedestrians appear in some regions that are too small to detect by the human pose estimation. In addition, when the trajectories of the pedestrians are too short, the estimated extrinsic parameters normally have high relative reprojection error. Thus, the matching rates of them are too small to be selected by the proposed method in Section 3.4. Hence, the proposed method also could not find the samples at some regions of the scene, which lead which leads to a higher relative reprojection error for those regions. However, the errors in these regions are still small enough for applications like multi-camera tracking for the intersection scenes.

4.4. Calibration with Crowded Scene

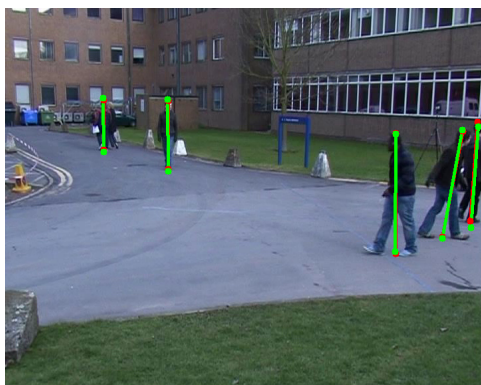
In this work, we also apply the proposed method on PETS2009-S2L1 dataset (CN6), which is a crowded scene with multiple pedestrians [38]. Due to the small resolution of the pedestrians in the scenes, the results of OpenPose [15,16] are not stable enough to calibrate the camera network. Thus, we only use the detected bounding boxes of the pedestrians which are extracted by YOLO [31]. We choose the center of the top edge and the center bottom edge of the bounding boxes as top and bottom of the pedestrians. We select the coordinate system of the camera 001 as the world coordinate system. To make the method work on this dataset, we also remove the limitation of the number of pedestrians that appear in the scene. Because of the orientation of the camera view 004, the tracking algorithm has a poor performance on this scene. Note that, the camera view 002 of the dataset is not available. Therefore, the result of the proposed method is not available for this scene. Even though the proposed method only works best for the scene that does not have so many pedestrians, the relative reprojection errors showed in Table 10 are still reasonably low. Figure 10 shows the example of the detected top and bottom of PETS2009-S2L1 dataset.



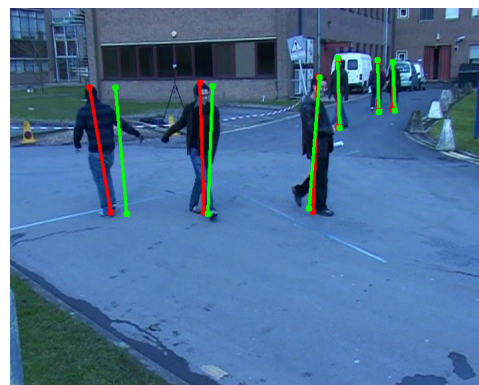
(a) Camera 001



(b) Camera 003

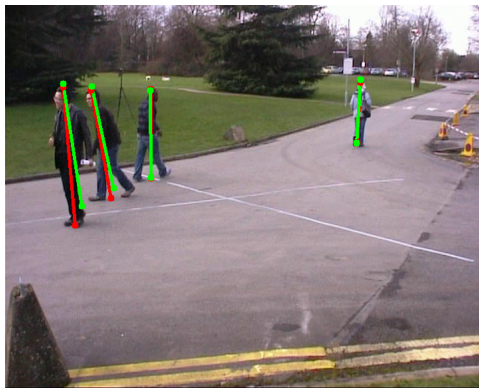


(c) Camera 005

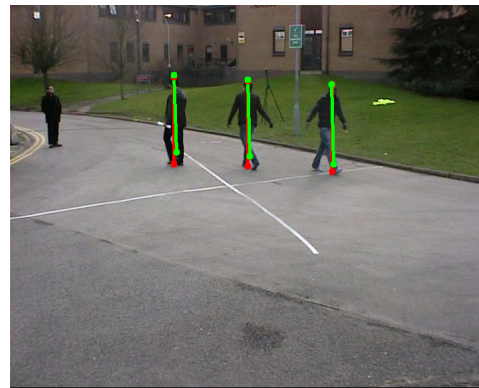


(d) Camera 006

Figure 10. Cont.



(e) Camera 007



(f) Camera 008

Figure 10. Example of detected the bottom and top positions of the pedestrians of Camera Network 6—PETS2009-S2L1 dataset. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

Table 10. Calibration results of camera calibration based on multiple walking people of the Camera Network 6. $\delta \mathbf{u}^{(rr)}$ denotes the relative reprojection error.

	001-003	001-005	001-006	001-007	001-008
$\delta \mathbf{u}^{(rr)}$ —Top (%)	2.5	5.1	8.6	7.4	6.0
$\delta \mathbf{u}^{(rr)}$ —Bottom (%)	1.9	5.9	7.3	12.5	10.0

4.5. Calibration with without Feet and Hip Joints

We evaluated the proposed method in Section 3.5 with Camera Network 7—a heavily occluded scene in an office (Figure 11). Since the scene is heavily occluded, the feet information is not enough for the extrinsic calibration based on feet joints. The extrinsic calibration based on hip joints is also not always available for all camera views. Table 11 shows that the proposed extrinsic calibration based on just the position of the top and the centerline can still obtain the accuracy that is very close to the extrinsic calibration based on hip joints. In the case of camera pair 003-004, the extrinsic calibration based on hip joints cannot collect enough data to estimate extrinsic parameters on single person sequences. On the other hand, the proposed method in Section 3.5 is still able to produce the extrinsic parameters with low relative reprojection errors. Figure 11 shows the example of the detected top and bottom of Camera Network 7.



(a) Camera 001



(b) Camera 002

Figure 11. Cont.

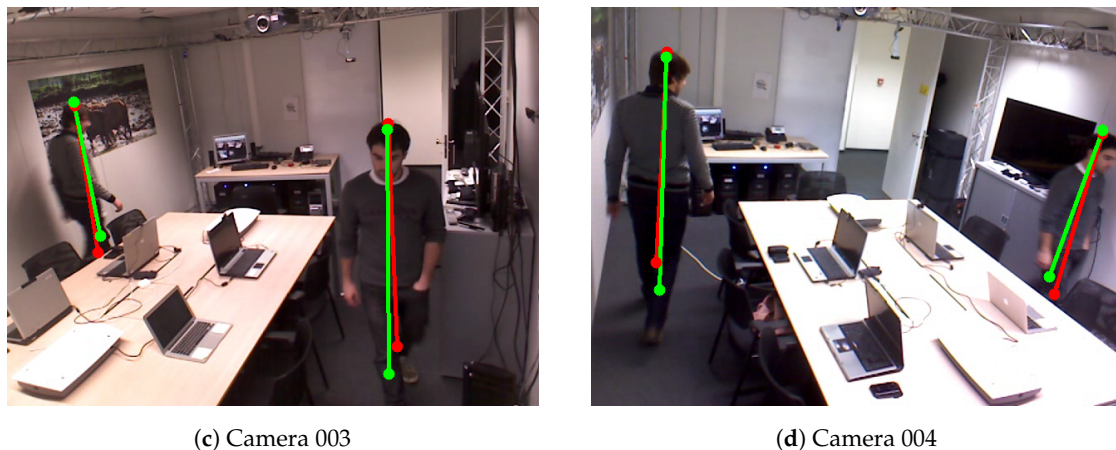


Figure 11. Example of detected the bottom and top positions of the pedestrians of Camera Network 7. Red color represents detected positions by the human pose estimation method. Green color represents the detected reprojected positions.

Table 11. Calibration results of camera calibration based on multiple walking people of the Camera Network 7. The extrinsic parameters was estimated based on the proposed method in Section 3.2 hip joints and the proposed method in Section 3.5 without both hip and feet information. $\delta \mathbf{u}^{(rr)}$ denotes the relative reprojected error.

	Single Person				Multiple People			
	001-002		003-004		001-002		003-004	
	Without Hip	Hip	Without Hip	Hip	Without Hip	Hip	Without Hip	Hip
$\delta \mathbf{u}^{(rr)}$ —top (%)	2.8	1.2	7.7	N/A	2.7	1.4	1.3	1.2
$\delta \mathbf{u}^{(rr)}$ —bottom (%)	6.7	6.4	8.4	N/A	7.5	8.5	7.6	10.6

5. Conclusions

In this paper, we present a simple and robust method to leverage the human pose estimation for the computation of 3D positions of the top and bottom of the pedestrians. To handle the case where multiple pedestrians are in the scene, we also developed a brute-force method to select appropriate top and bottom locations for the extrinsic camera calibration. For indoor camera networks which are intended for people surveillance, the feet of pedestrians are usually occluded by the furniture. This is the degenerate for most of the current state-of-the-art calibration methods due to the coplanarity of all the positions of the top of a single pedestrian. To the best of our knowledge, no work exists to deal with camera network calibration for this specific scenario. We proposed the extrinsic calibration by using a walking human as the calibration object, assuming only the top of the pedestrian and centerline of the person are visible. Thus, the proposed method can be very useful for many of the existing indoor multi-camera visual surveillance systems. The proposed method can work well in various environments as well as robust against occlusion compared to state-of-the-art methods. More importantly, the proposed method can work completely automatically without manually selecting proper input data for the calibration method. In the future, we will investigate a regional selection method to handle the case where the walking trajectory is too short.

Author Contributions: Conceptualization, A.M.T., W.P. and J.G.; Formal analysis, A.M.T. and J.G.; Investigation, J.G.; Methodology, A.M.T. and J.G.; Project administration, W.P.; Software, A.M.T. and J.G.; Supervision, W.P. and N.D.; Visualization, A.M.T.; Writing—original draft, A.M.T.; Writing—review & editing, W.P. and L.A.

Funding: This research was funded by the Flemish Fund for Scientific Research FWO-Flanders through the grant 3G014718.

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

Abbreviations

The following abbreviations are used in this manuscript:

RANSAC	Random sample consensus
2D	Two-dimensional
3D	Three-dimensional
SVD	Singular Value Decomposition

References

1. Dimitrievski, M.; Veelaert, P.; Philips, W. Behavioral pedestrian tracking using a camera and lidar sensors on a moving vehicle. *Sensors* **2019**, *19*, 391. [[CrossRef](#)] [[PubMed](#)]
2. Chaurasia, G.; Duchene, S.; Sorkine-Hornung, O.; Drettakis, G. Depth Synthesis and Local Warps for Plausible Image-based Navigation. *ACM Trans. Graph.* **2013**, *32*, 30. [[CrossRef](#)]
3. Sebe, I.O.; Hu, J.; You, S.; Neumann, U. 3D Video Surveillance with Augmented Virtual Environments. In Proceedings of the First ACM SIGMM International Workshop on Video Surveillance (IWVS '03), Berkeley, CA, USA, 2–8 November 2003; ACM: New York, NY, USA, 2003; pp. 107–112.
4. Zhang, Z.; Zhao, Y.; Wang, Y.; Liu, J.; Yao, Z.; Tang, J. Transferring Training Instances for Convenient Cross-View Object Classification in Surveillance. *Trans. Inf. For. Sec.* **2013**, *8*, 1632–1641. [[CrossRef](#)]
5. Hall, E.L.; Tio, J.B.; McPherson, C.A.; Sadjadi, F.A. Measuring Curved Surfaces for Robot Vision. *Computer* **1982**, *15*, 42–54. [[CrossRef](#)]
6. Tsai, R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE J. Robot. Autom.* **1987**, *3*, 323–344. [[CrossRef](#)]
7. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [[CrossRef](#)]
8. Hartley, R.I. An algorithm for self calibration from several views. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 21–23 June 1994; pp. 908–912.
9. Liu, J.; Collins, R.; Liu, Y. Surveillance camera autocalibration based on pedestrian height distributions. In Proceedings of the British Machine Vision Conference, Dundee, UK, 28 August–2 September 2011.
10. Liu, J.; Collins, R.; Liu, Y. Robust autocalibration for a surveillance camera network. In Proceedings of the IEEE Workshop on Applications of Computer Vision, Clearwater Beach, FL, USA, 15–17 January 2013; pp. 433–440.
11. Li, S.; Nguyen, V.H.; Ma, M.; Jin, C.B.; Do, T.D.; Kim, H. A simplified nonlinear regression method for human height estimation in video surveillance. *EURASIP J. Image Video Process.* **2015**, *2015*, 32. [[CrossRef](#)]
12. Lv, F.; Zhao, T.; Nevatia, R. Camera calibration from video of a walking human. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 1513–1518. [[PubMed](#)]
13. Possegger, H.; Rüther, M.; Sternig, S.; Mauthner, T.; Klopschitz, M.; Roth, P.M.; Bischof, H. Unsupervised Calibration of Camera Networks and Virtual PTZ Cameras. In Proceedings of the Computer Vision Winter Workshop, Breckenridge, CO, USA, 9–11 January 2012.
14. Guan, J.; Deboeverie, F.; Slembrouck, M.; Van Haerenborgh, D.; Van Cauwelaert, D.; Veelaert, P.; Philips, W. Extrinsic calibration of camera networks based on pedestrians. *Sensors* **2016**, *16*, 654. [[CrossRef](#)] [[PubMed](#)]
15. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In Proceedings of the Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 June 2017.
16. Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S.E.; Sheikh, Y. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. *arXiv* **2018**, arXiv:1812.08008.
17. Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S.E.; Sheikh, Y. Convolutional pose machines. In Proceedings of the Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016.
18. Fang, H.S.; Xie, S.; Tai, Y.W.; Lu, C. RMPE: Regional Multi-person Pose Estimation. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
19. Simon, T.; Joo, H.; Matthews, I.; Sheikh, Y. Hand Keypoint Detection in Single Images using Multiview Bootstrapping. In Proceedings of the Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 June 2017.

20. Xiu, Y.; Li, J.; Wang, H.; Fang, Y.; Lu, C. Pose Flow: Efficient Online Pose Tracking. In Proceedings of the British Machine Vision Conference (BMVC), Newcastle, UK, 3–6 September 2018.
21. Xiao, B.; Wu, H.; Wei, Y. Simple Baselines for Human Pose Estimation and Tracking. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
22. Lv, F.; Zhao, T.; Nevatia, R. Self-calibration of a camera from video of a walking human. In Proceedings of the Object Recognition Supported by User Interaction for Service Robots; Quebec, QC, Canada, 11–15 August 2002; Volume 1, pp. 562–567.
23. Teixeira, L.; Maffra, F.; Badii, A. Scene Understanding for Auto-Calibration of Surveillance Cameras. In *Advances in Visual Computing*; Bebis, G., Boyle, R., Parvin, B., Koracin, D., McMahan, R., Jerald, J., Zhang, H., Drucker, S.M., Kambhamettu, C., El Choubassi, M., et al., Eds.; Springer International Publishing: Cham, Switzerland, 2014; pp. 671–682.
24. Hödlmoser, M.; Kampel, M. Multiple Camera Self-calibration and 3D Reconstruction Using Pedestrians. In *Advances in Visual Computing*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 1–10.
25. Hartley, R.I. Chirality. *Int. J. Comput. Vis.* **1998**, *26*, 41–61. [[CrossRef](#)]
26. Werner, T.; Pajdla, T. Cheirality in epipolar geometry. In Proceedings of the Eighth IEEE International Conference on Computer Vision (ICCV 2001), Vancouver, BC, Canada, 7–14 July 2001; Volume 1, pp. 548–553.
27. Pollefeys, M.; Koch, R.; Gool, L.V. Self-Calibration and Metric Reconstruction In spite of Varying and Unknown Intrinsic Camera Parameters. *Int. J. Comput. Vis.* **1999**, *32*, 7–25. [[CrossRef](#)]
28. Weng, J.; Ahuja, N.; Huang, T.S. Motion and structure from point correspondences with error estimation: planar surfaces. *IEEE Trans. Signal Process* **1991**, *39*, 2691–2717. [[CrossRef](#)]
29. Faugeras, O.; Lustman, F. Motion and Structure from Motion in a Piecewise Planar Environment. *Int. J. Pattern Recognit. Artif. Intell. IJPRAI* **1988**, *2*, 485–508. [[CrossRef](#)]
30. Lettry, L.; Dragon, R.; Van Gool, L. Markov Chain Monte Carlo Cascade for Camera Network Calibration Based on Unconstrained Pedestrian Tracklets. In *Computer Vision – ACCV 2016*; Lai, S.H., Lepetit, V., Nishino, K., Sato, Y., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2017; Volume 10112, pp. 400–415.
31. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
32. Schönemann, P. A generalized solution of the orthogonal procrustes problem. *Psychometrika* **1966**, *31*, 1–10. [[CrossRef](#)]
33. Fischler, M.A.; Bolles, R.C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Commun. ACM* **1981**, *24*, 133–135. [[CrossRef](#)]
34. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*, 2nd ed.; Cambridge University Press: New York, NY, USA, 2003.
35. Ma, Y.; Soatto, S.; Kosecka, J.; Sastry, S.S. *An Invitation to 3-D Vision: From Images to Geometric Models*; Springer: New York, NY, USA, 2003.
36. Bouguet, J.Y. Camera Calibration Toolbox for Matlab. 2015. Available online: http://www.vision.caltech.edu/bouguetj/calib_doc/ (accessed on 14 April 2019).
37. Berclaz, J.; Fleuret, F.; Turetken, E.; Fua, P. Multiple Object Tracking Using K-Shortest Paths Optimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1806–1819. [[CrossRef](#)] [[PubMed](#)]
38. Ferryman, J.; Shahrokni, A. PETS2009: Dataset and challenge. In Proceedings of the 2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, Snowbird, UT, USA, 7–9 December 2009; pp. 1–6.

