Automatic Calibration of Stationary Surveillance Cameras in the Wild

Guido M.Y.E. Brouwers[1](✉), Matthijs H. Zwemer[1,2], Rob G.J. Wijnhoven[1], and Peter H.N. de With[2]

[1] ViNotion B.V., Eindhoven, The Netherlands
{guido.brouwers,rob.wijnhoven}@vinotion.nl
[2] Eindhoven University of Technology, Eindhoven, The Netherlands
{m.zwemer,p.h.n.de.With}@tue.nl

Abstract. We present a fully automatic camera calibration algorithm for monocular stationary surveillance cameras. We exploit only information from pedestrians tracks and generate a full camera calibration matrix based on vanishing-point geometry. This paper presents the first combination of several existing components of calibration systems from literature. The algorithm introduces novel pre- and post-processing stages that improve estimation of the horizon line and the vertical vanishing point. The scale factor is determined using an average body height, enabling extraction of metric information without manual measurement in the scene. Instead of evaluating performance on a limited number of camera configurations (video seq.) as in literature, we have performed extensive simulations of the calibration algorithm for a large range of camera configurations. Simulations reveal that metric information can be extracted with an average error of 1.95 % and the derived focal length is more accurate than the reported systems in literature. Calibration experiments with real-world surveillance datasets in which no restrictions are made on pedestrian movement and position, show that the performance is comparable (max. error 3.7 %) to the simulations, thereby confirming feasibility of the system.

Keywords: Automatic camera calibration · Vanishing points

1 Introduction

The growth of video cameras for surveillance and security implies more automatic analysis using object detection and tracking of moving objects in the scene. To obtain a global understanding of the environment, individual detection results from multiple cameras can be combined. For more accurate global understanding, it is required to convert the pixel-based position information of detected objects in the individual cameras, to a global coordinate system (GPS). To this end, each individual camera needs to be calibrated as a first and crucial step.

The most common model to relate pixel positions to real-world coordinates is the pinhole camera model [5]. In this model, the camera is assumed to make

G. Hua and H. Jégou (Eds.): ECCV 2016 Workshops, Part II, LNCS 9914, pp. 743–759, 2016.
DOI: 10.1007/978-3-319-48881-3_52

a perfect perspective transformation (a matrix), which is described by intrinsic and extrinsic parameters of the camera. The intrinsic parameters are: pixel skew, principal point location, focal length and aspect ratio of the pixels. The extrinsic parameters describe the orientation and position of the camera with respect to a world coordinate system by a rotation and a translation. The process of finding the model parameters that best describe the mapping of scene onto the image plane of the camera is called camera calibration.

The golden standard for camera calibration [5] uses a pre-defined calibration object [19] that is physically placed in the scene. Camera calibration involves finding the corresponding key points of the object in the image plane and describing the mapping of world coordinates to image coordinates. However, in surveillance scenes the camera is typically positioned high above the ground plane, leading to impractically large calibration objects covering a large part of the scene. Other calibration techniques exploit camera motion (Maybank *et al.* [14], Hartley [6]) or stereo cameras (Faugeras and Toscani *et al.* [4]), to extract multiple views from the scene. However, because most surveillance cameras are static cameras these techniques cannot be used. Stereo cameras explicitly create multiple views, but require two physical cameras that are typically not available.

A different calibration method uses vanishing points. A vanishing point is a point where parallel lines from the 3D world intersect in the image. These lines can be generated from static objects in the scenes (such as buildings, roads or light poles), or by linking moving objects at different positions over time (such as pedestrians). Static scenes do not always contain structures with parallel lines. In contrast, there are always moving objects in surveillance scenes, which makes this approach attractive. In literature, different proposals use the concept of vanishing points. However, these approaches either require very constrained object motion [7,8,15], require additional manual annotation of orthogonal directions in the scene [12,13], or only calibrate the camera up to a scale factor [8–10,12,13,15]. To our knowledge, there exists no solution that results in an accurate calibration for a large range of camera configurations in uncontrolled surveillance scenes; automatic camera calibration does not work in unconstrained cases.

This paper proposes a fully automatic calibration method for monocular stationary cameras in surveillance scenes based on the concept of vanishing points. These points are extracted from pedestrian tracks, where no constraints are imposed on the movement of pedestrians. We define the camera calibration as a process, which is based as the extraction of the vanishing points with the following determination of the camera parameters. The main contributions to this process are (1) a pre-processing step that improves estimation of the vertical vanishing point, (2) a post-processing step that exploits the height distribution of pedestrians to improve horizon line estimation, (3) determination of the camera height (scale factor) using an average body height and (4) an extensive simulation of the total process, showing that the algorithm obtains an accurate calibration for a large range of camera configurations as used in real-world scenes.

1.1 Related Work

Vanishing points are also extracted from object motion in the scene by Lv *et al.* [12,13] and Kusakunniran *et al.* [9]. Although they proved that moving pedestrians can be used as a calibration object, the accuracy of the algorithms is not sufficient for practical applications. Krahnstoever *et al.* [8] and Micusik *et al.* [15] use the homography between the head and foot plane to estimate the vanishing point and horizon line. Although providing the calibration upto a scale factor, they require a constrained pedestrian movement and location. Liu *et al.* [10] propose to use the predicted relative human height distribution to optimize the camera parameters. Although providing a fully automated calibration method, they exploit only a single vanishing point which is not robust for a large range of camera orientations. Recent work from Huang *et al.* [7] proposes to extract vanishing points from detected locations from pedestrian feet and only calculate the intrinsic camera parameters.

All previously mentioned methods use pixel-based foreground detection to estimate head and feet locations of pedestrians, which makes them impractical for crowded scenes with occlusions. Additionally, these methods require at least one known distance in the scene to be able to translate pixels to real distances. Although in controlled scenes the camera calibration from moving pedestrians is possible, many irregularities occur which complicate the accurate detection of vanishing points. Different types of pedestrian appearances, postures and gait patterns result in noisy point data containing many outliers. To solve the previous issues, we have concentrated particularly on the work of Kusakunniran [9] and Liu [10]. Our strategy is to extend this work such that we can extract camera parameters in uncontrolled surveillance scenes with pedestrians, while omitting background subtraction and avoiding scene irregularity issues.

2 Approach

To calibrate the camera, we propose to use vertical and parallel lines in the scene to detect the vertical vanishing point and the horizon line. These lines are extracted from head and feet positions of tracked pedestrians. Then, a general technique is used to extract camera parameters from the obtained vanishing points. It should be noted that the approach is not limited to tracking pedestrians, but applies to any object class for which two orthogonal vanishing points can be extracted. The overview of the system is shown in Fig. 1. First, we compute vertical lines by connecting head and feet positions of pedestrians. The intersecting point of these lines is the location of the vertical vanishing point. Second, points on the horizon line are extracted by computing parallel lines between head and feet positions at different points in time. The horizon line is then robustly fitted by a line fitting algorithm. Afterwards, the locations of the vertical vanishing point and horizon line are used to compute a full camera calibration. In the post-processing step, the pedestrian height distribution is used to refine the camera parameters. Finally and for distance calibration involving translation of pixel positions to metric locations, a scale factor is computed by

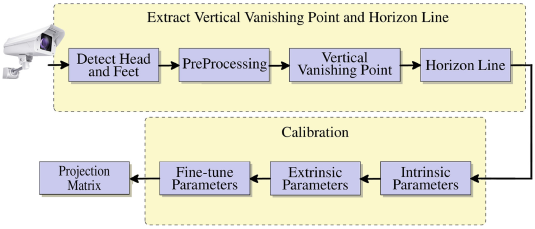


Fig. 1. Block diagram of the proposed calibration system.

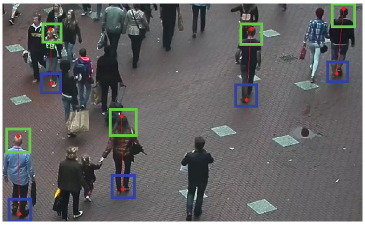


Fig. 2. Example head and feet detections.

using the average body height of pedestrians. To retrieve more accurate camera parameters and establish a stable average setting, the complete algorithm is executed multiple times on subsets of the detected pedestrian positions which addresses the noise in the parameters of single cycles.

2.1 Head and Feet Detection

The head and feet positions are detected using two object detectors, which are trained offline using a large training set and are fixed for the experiments. We apply the Histogram of Oriented Gradients (HOG) detector [2] to individually detect head and feet (Fig. 2). The detector for feet is trained with images in which the person feet are visibly positioned as a pair. A vertical line can be found from a pedestrian during walking, at the moment (cross-legged phase) in which the line between head and feet best represents a vertical pole. Head and feet detections are matched by vertically shifting the found head detection of each person downwards and then measuring the overlap with the possible feet position. When the overlap is sufficiently large, the head and feet are matched and used in the calibration algorithm. Due to small localization errors in both head and feet positions and the fact that pedestrians are not in a perfectly upright position, the set of matched detections contains noisy data. This will be filtered in the next step.

2.2 Pre-processing

The matched detections are filtered as a pre-processing step, such that only the best matched detections are used to compute the vanishing points and outliers are omitted. To this end, the matched detections are sorted by the horizontal positions of the feet locations. For each detection, the vertical derivative of the line between head and feet is computed. Because the width of the image is substantially smaller than the distance to the vertical vanishing point, we can linearly approximate the tangential line related to the vertical derivative by a first-order line. After extreme outliers are removed, this line is fitted through the derivatives using a least-squares method. Derivatives that have a distance larger than an empirical threshold are removed from the dataset. Finally, the remaining inliers are used to compute the vertical vanishing point and the horizon line.

2.3 Vertical Vanishing Point and Horizon Line

The vertical vanishing point location is computed using the method from Kusakunniran *et al.* [9]. Pedestrians are considered vertical poles in the scene and the collinearity of the head, feet and vertical vanishing point of each pedestrian are used to calculate the exact location of the vertical vanishing point.

Parallel lines constructed from key points of pedestrians that are also parallel to the ground plane intersect at a point on the horizon line. We can combine each pair of head and feet detections of the same pedestrian to define such parallel lines and compute the intersection points. Multiple intersection points lead then to the definition of the horizon line. The iterative line-finding algorithm which is used to extract the horizon line is described below.

The horizon line is estimated by a least-squares algorithm. Next, inliers are selected based on a their distance to the found horizon line, which should be smaller than a pre-determined threshold T. These inliers are used to fit a new line by the same least-squares approach so that the process becomes iterative. The iterative process stops when the support of the line in terms of inliers does not further improve. This approach always leads to a satisfactory solution in our experiments. The support of a line has been experimentally defined by a weighted sum W of contributions of the individual points i having an L2-distance D_i to the current estimate of the horizon line. Each contribution is scaled with the threshold to a fraction D_i/T and exponentially weighted. This leads to the least-squares weight W specified by

$$W = \sum_{i=1}^{M} \exp \frac{-D_i^2}{T^2}. \tag{1}$$

The pre-defined threshold T depends on the accuracy of the detections and on the orientation of the camera. If the orientation of the camera is facing down at a certain angle, the intersection points are more sensitive to noise, i.e. the spread of the intersection points will be larger. A normal distribution is fitted on the intersection points. The threshold T is then determined as the standard deviation of that normal distribution.

2.4 Calibration Algorithm

The derived horizon line and the vertical vanishing point are now used to directly determine the camera parameters. As we assume zero skew, square pixels and the principal point being at the center of the image, the focal length is the only intrinsic camera parameter left to be determined.

The focal length represents the distance from the camera center to the principal point. Using the geometric properties described by Orghidan *et al.* [17], the distance can be computed using only the distance from the vertical vanishing point to the principal point and the distance from the horizon line to the principal point. The orientation of the camera is described by a rotation matrix, which is composed of a rotation around the tilt and the roll angle, thus around the x-axis and z-axis, respectively. The tilt angle θ_t is defined as the angle between the focal line and the z-axis of the world coordinate system. Because any line between the camera center $\mathbf{O_c}$ and the horizon line is horizontal, the tilt angle is computed by

$$\theta_t = 90^\circ + \arctan\left(\frac{||\mathbf{O_i V_i}||}{f}\right), \tag{2}$$

where $\mathbf{V_i}$ is the point on the horizon line which is closest to the principal point, $\mathbf{f}$ is the focal length and $\mathbf{O_i}$ is the center of the image. The roll angle is equal to the angle between the line from the vertical vanishing point to the principal point and the vertical line through the principal point. The translation defined in the extrinsic parameters is a vector $\mathbf{t}$ pointing from the camera origin to the world origin. We choose the point on the ground plane that is directly beneath the camera center as the world origin, so that the position of the camera center in world coordinates is described by $P_{cam}(x, y, z) = (0, 0, s)$. This introduces the well-known scale factor s being equal to the camera height. The translation vector $\mathbf{t}$ is computed by

$$\mathbf{t} = -R \cdot P_{cam}, \tag{3}$$

where R is the rotation matrix. If metric information is required, the scale factor s must be determined to relate distances in our world coordinate system to metric distances. The scale factor can be computed if at least one metric distance in the scene is known. Inspired by [3], the average body height of the detected pedestrians is used, as this information is readily available. Because the positions of the feet are on the ground plane, these locations can be determined by

$$s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = [P_1, P_2, P_4]^{-1} \begin{bmatrix} u_f \\ v_f \\ 1 \end{bmatrix}, \tag{4}$$

where (u_f, v_f) are the image coordinates of the feet and P_i denotes the i^{th} column of the projection matrix $P = K[R|t]$, where K is the calibration matrix. The world coordinates of the head are situated on the line from the camera center through the image plane at the pixel location of the head, towards infinity. The point on this line that is closest to the vertical line passing through the position of the feet, is defined as the position of the head. The L2-distance between the two points is equal to the body height of the pedestrian. The scale factor is chosen such that the measured average body height of the pedestrians is equal to the *a-priori* known country-wide average [16]. Note that the standard deviation of the country-wide height distribution has no influence if sufficient samples are available. The worst-case deviation on average height globally is 12 % (1.58–1.80 m), but this never occurs because outliers (airports, children) are averaged.

2.5 Post-processing

Noise present in the head and feet detections of pedestrians affects the locations of the intersection points that determine the location of the vertical vanishing point and the horizon line. When the tilt of the camera is close to horizontal, vertical lines in the scene are almost parallel in the image plane, which makes the intersection points sensitive to noise. As a result, the average position of the intersection points is shifted upwards, which decreases exponentially when the tilt of the camera increases (facing more downwards), see Fig. 4c. The intersection points that determine the horizon line undergo a similar effect when the camera is facing down, see Fig. 4b. As a consequence of these shifting effects, the resulting focal length and camera tilt are estimated at a lower value than they should be. Summarizing, the post-processing compensates the above shifting effect for cameras facing downwards.

This compensation is performed by evaluating the pedestrian height distribution, as motivated by [10]. The pedestrian height is assumed to be a normal distribution, as shown by Millar [16]. The distribution has the smallest variance when the tilt is estimated correctly. The tilt is computed as follows. The pedestrian height distribution is calculated for a range of tilt angles (from -5 to $+15°$ in steps of $0.5°$, relative to the initially estimated tilt angle). We select the angle with the smallest variance as the best angle for the tilt. Figure 7 shows one-over standard deviation of the body height distribution for the range of evaluated tilt angles (for three different cameras with three different true tilt angles). As can be seen in the figure, this optimum value is slightly too high. Therefore, the selected angle is averaged with the initial estimate when starting the pre-processing, leading to the final estimated tilt.

In retrospect, our introduced pre-processing has also solved a special case that introduces a shift in the vertical vanishing point. This occurs when the camera is horizontally looking forward, so that the tilt angle is $90°$. Our pre-processing has corrected this effect.

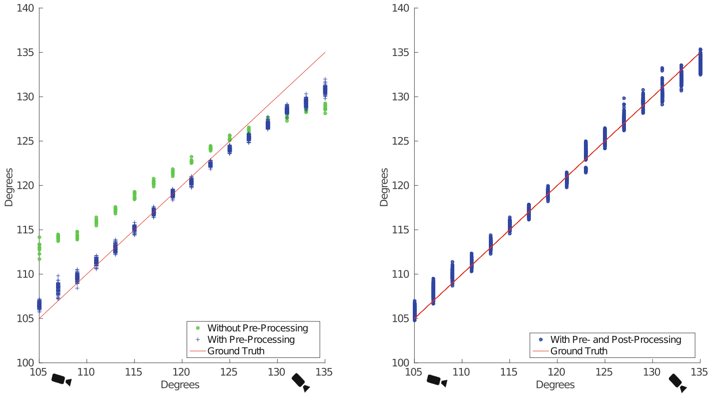
3 Model Simulation and Its Tolerances

The purpose of model simulation is to evaluate the robustness against model errors which should be controlled for certain camera orientations. Secondly, the accuracy of the detected camera parameters should be evaluated, and the influence of error propagation in the calculation when input parameters are noisy.

The advantage of using simulated data is that the ground truth of every step in the algorithm is known. The first step of creating a simulation is defining a scene and calculating the projection matrix for this scene. Next, the input data for our algorithm is computed by creating a random set of head and feet positions in the 3D-world, where the height of the pedestrian is defined by a normal distribution $\mathcal{N}(\mu, \sigma)$ with $\mu = 1.74\,\mathrm{m}$ and $\sigma = 0.065\,\mathrm{m}$, as in [16]. The pixel positions of the head and feet positions in the image plane are computed using the projection matrix. These image coordinates are used as input for our calibration algorithm. For our experiments, we model the errors as noise sources, so that the robustness of the system can be evaluated with respect to (1) the localization errors, (2) our assumptions about pedestrians walking perfectly vertical and (3) the assumptions on intrinsic parameters of the camera. We first simulate the system to evaluate the camera parameters since this is the main goal. However, we extend our simulation to evaluate also the measured distances in the scene, which are representative for the accuracy of mapping towards a global coordinate system.

3.1 Error Sources of the Model

We define two noise sources in our simulation system. The first source models the localization error of the detector for the head and feet positions in the image plane. The second source is a horizontal offset of the head position in the 3D-world, which originates from the ideal assumption on pedestrians walking perfectly vertical through the scene. This is not true in practice, leading to an error in the horizontal offset as well. All noise sources are assumed to have normal distributions. The localization error of the detector is determined by manually annotating head and feet positions of a real-life data sequence, which serves as ground truth. The head and feet detectors are applied to this dataset and the localization error is determined. We have found that the localization errors can be described by a standard deviation of $\sigma_x = 7\,\%$ [width of head] and $\sigma_y = 11\,\%$ in the x- and y-direction for the head detector and a standard deviation of $\sigma_x = 10\,\%$ and $\sigma_y = 16\,\%$ for the feet detector.



(a) With and without pre-processing

(b) Post-processing

Fig. 3. Accuracy of the camera tilt estimation, the red line shows perfect estimation. (Color figure online)

Estimating the noise level of head positions in the 3D world is not trivial, since ground-truth data is not available. In order to create a coarse estimate of the noise level, manual pedestrian annotations of a calibrated video sequence are used to plot the distributions of the intersection points of the vertical vanishing point and the horizon line. The input noise of the model simulation is adjusted such that the simulated distributions match with the annotated distributions. Input noise originates from two noise sources: errors perpendicular to the walking direction and errors in the walking direction. We have empirically estimated the noise perpendicular to the walking direction to have $\sigma = 0.08\,\mathrm{m}$ and the noise parallel to the walking direction to have $\sigma = 0.10\,\mathrm{m}$. These values are employed in all further simulation experiments.

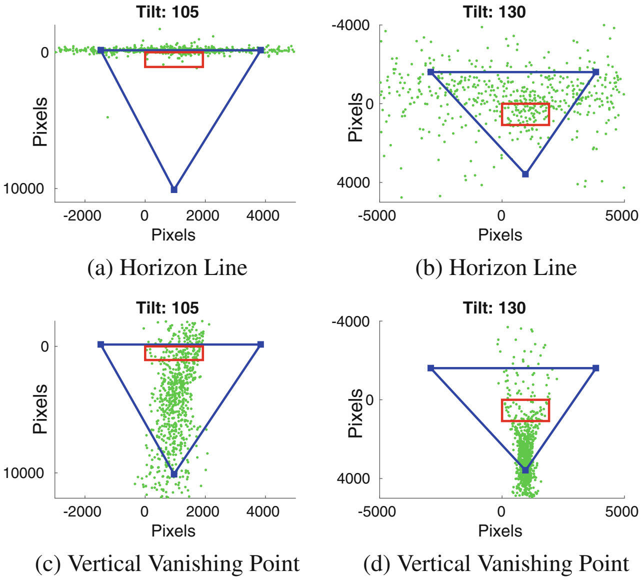
3.2 Experiments on Error Propagation in the Model

In our experiments, we aim to optimize the tilt estimation, because the tilt angle has the largest influence on the camera calibration. In order to optimize the estimation of the camera tilt, we need to optimize the location of the vertical vanishing point and horizon line. Below, the dependencies of the tilt on various input parameters are evaluated: pre-processing, the camera orientation and post-processing.

Pre-Processing: To evaluate the effect of the pre-processing on the locations of the intersection points, which determine the vertical vanishing point and the horizon line, the tilt is computed for various camera orientations and 150 times for each orientation. Each time the simulation creates new detection sets using the noise sources. Results of the detected tilt with and without pre-processing are shown in Fig. 3a. The red line depicts the ground-truth value for the camera tilt. The blue points are the estimated camera tilts for the various simulations, with an average error of 3.7°. It can be observed that the pre-processing stage clearly removes outliers such that the estimation of the position of the vertical vanishing point is improved, especially for cameras with a tilt lower than 120°. The average error decreases to 1.3°, giving a 65 % improvement.

Camera Orientation: The position of the vanishing points depends on the detection error in the image plane and on the camera orientation. The influence of the orientation on the error of the vanishing-point locations is evaluated. The simulation environment is used to create datasets for camera tilts ranging from 105 to 135°. For each orientation, 100 datasets are produced and individual vertical vanishing points and horizon lines are computed.

Figure 4 shows the intersection points of the horizon line and vertical vanishing point for a camera tilt of 105° and 130°. The red rectangle represents the image plane, the intersection points are depicted in green and the blue triangular corners represent a set of three orthogonal vanishing points, where the bottom corner is the vertical vanishing point and the two top corners are two vanishing points on the horizon line in orthogonal directions. For a camera tilt of 105°, which is close to horizontal, the intersection points of the horizon line lie close to the ground truth, while intersection points of the vertical vanishing point are widely spread and shifted upwards. For a camera tilt of 130°, the opposite effect



(a) Horizon Line (b) Horizon Line

(c) Vertical Vanishing Point (d) Vertical Vanishing Point

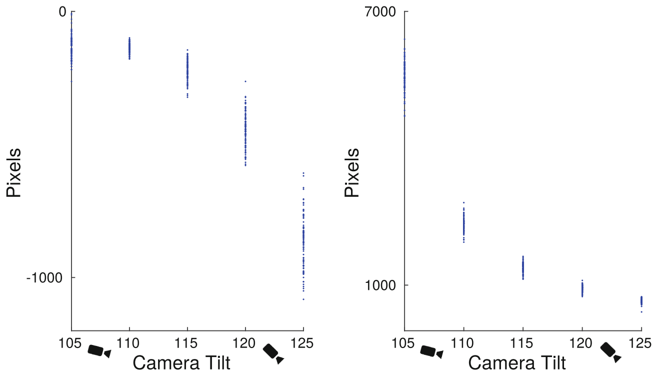
Fig. 4. Shift of the intersection-point blobs for (a, c) 105 and (b, d) 130° tilt. (Color figure online)

is visible. Figure 5 shows the vertical location error of the computed vertical vanishing point and horizon line. It can be seen that when the camera tilt increases, the downwards shift of the horizon line increases exponentially. When the camera tilt decreases, the upwards shift of the vertical vanishing point increases exponentially. The variance of the vertical location error of the horizon line grows when the tilt increases, which means that the calibration algorithm is more sensitive to noise in the head and feet positions. A similar effect is visible in the location error of the vertical vanishing point for decreasing camera tilt.

Post-Processing: Results from the previous experiments show that the detected vanishing point and horizon line have a vertical localization error. The post-processing stage aims at improving the tilt estimation of the calibration algorithm. The effect of the post-processing is evaluated by computing the tilt for simulated scenes with various camera orientations. The results of the calibration algorithm with post-processing are shown in Fig. 3b. It can be observed that the post-processing improves the tilt estimation for camera tilts higher than 120°. The average tilt error is reduced from 1.3° when using only pre-processing, to 0.7° with full processing (an improvement of 46 %).

3.3 Monte-Carlo Simulation

The model simulation can be used to derive the error distributions of the camera parameters by performing Monte-Carlo simulation. These error distributions are



(a) Horizon Line (b) Vertical Vanishing Point

Fig. 5. Vertical shift of the predicted vanishing points with respect to the ground truth.

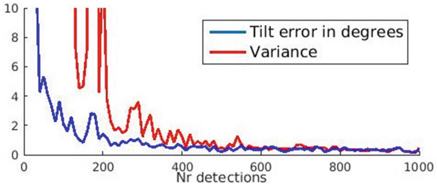


Fig. 6. Accuracy of tilt estimation for increasing number of detections (40 simulation iterations per detection).

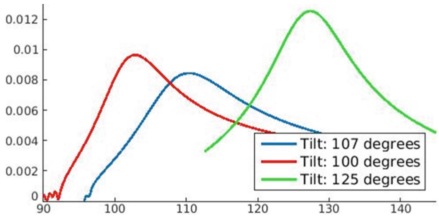
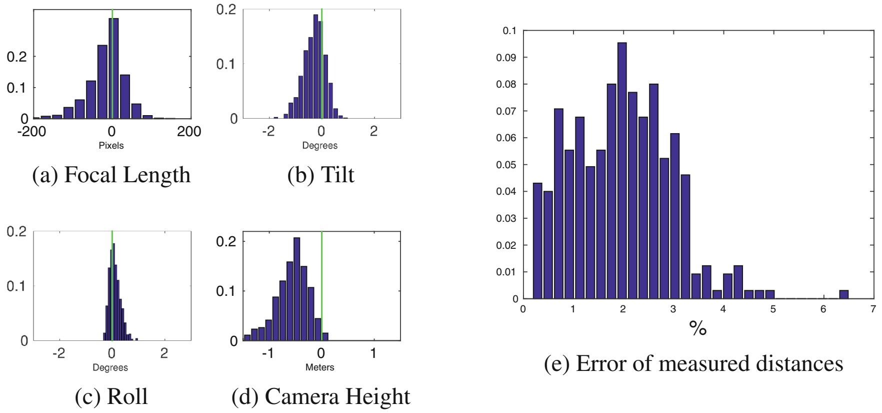


Fig. 7. Postprocessing, 1/standard deviation of the body height distribution for several camera tilts.

computed by calibrating 1,000 simulated scenes using typical surveillance camera configurations. For the simulations, the focal length is varied within 1,000–4,000 pixels, the tilt in 110–130°, the roll from -5 to $+5°$ and the camera height within 4–10 m. The error distributions are computed for the focal length, tilt, roll and camera height. Figure 6 shows how the tilt error decreases with respect to the number of input pedestrian locations (from the object detector). Both the absolute error and its variance decrease quickly and converge after a few hundred locations.

Figure 8 shows the error distributions of the previous parameters via Monte-Carlo simulations. The standard deviations of the focal length error, camera tilt and camera roll are 47.1 pixels, 0.41° and 0.21°, respectively. The mean of the camera height distribution is lower than the ground truth. This is due to the fact that when the estimated tilt and focal length are smaller than the actual values, the detected body height of the pedestrian is larger and the scale factor will be smaller giving a lower camera height. The average error of measured distances in



(a) Focal Length (b) Tilt

(c) Roll (d) Camera Height

(e) Error of measured distances

Fig. 8. Distribution of the estimated camera parameters from Monte-Carlo simulations.

the scene is 1.95 %, while almost all errors are below 3.2 %. These results show that accurate detection of camera parameters is obtained for various camera orientations.

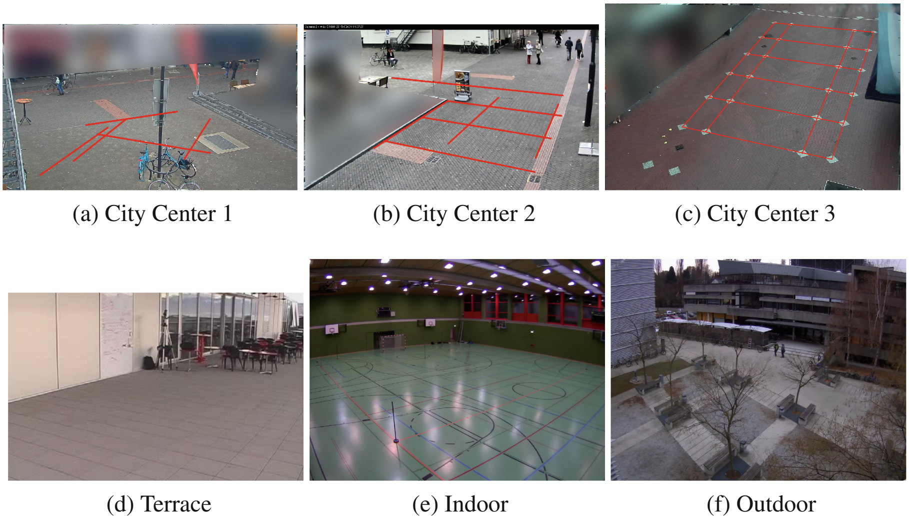
4 Experimental Results of Complete System

Three public datasets are used to compare the proposed calibration algorithm with the provided ground-truth parameters (tilt, roll, focal length and camera height) of these sequences: Terrace[1] and Indoor/Outdoor[2]. In addition, three novel datasets are created, two of which are fully calibrated (City Center 1 and 2), so that they provide ground-truth information on both intrinsic and extrinsic camera parameters. The intrinsic parameters are calibrated using a checker board and the MATLAB camera calibration toolbox. The true extrinsic parameters are computed by manually extracting parallel lines from the ground plane and computing the vanishing points. In City Center 3, only measured distances are available which will be used for evaluation. Examples of the datasets are shown in Fig. 9, where the red lines represent the manually measured distances.

4.1 Experiment 1: Fully Calibrated Cameras

The first experiment comprises a field test with the two fully calibrated cameras from the City Center 1 and 2 datasets. The head and feet detector is applied to 30 min of video with approximately 2,500 and 200 pedestrians, resulting in 7,120 and 2,569 detections in the first and second dataset, respectively. After

[1] Terrace: CVLab EPFL database of the University of Lausanne [1].
[2] Indoor/Outdoor: ICG lab of the Graz University of Technology [18].



(a) City Center 1

(b) City Center 2

(c) City Center 3

(d) Terrace

(e) Indoor

(f) Outdoor

Fig. 9. Examples of the datasets used in the experiments, red lines are manually measured distances in the scene. (Color figure online)

pre-processing, approximately half of the detections remain and are used for the calibration algorithm. Resulting camera parameters are compared with the ground-truth values. The results are shown in Table 1. The tilt estimation is accurate up to 0.43° for both datasets. The estimated roll is less accurate, caused by a horizontal displacement of the principal point from the image center. The focal length and camera height are estimated accurately.

Next, several manually measured distances in the City Center 1–3 datasets are compared with estimated values from our algorithm. Pixel positions of the measured points on the ground plane are manually annotated. These pixel positions are converted back to world coordinates using the predicted projection matrix. For each combination of two points we compare the estimated distance with our manual measurement. The average error over all distances is shown in Table 3. The first two datasets have an error of approximately 2.5 %, while the third dataset has a larger error, which is due to a curved ground plane.

4.2 Experiment 2: Public Datasets

The proposed system is compared to two auto-calibration methods described by Liu *et al.* [10] (2011) and Liu *et al.* [11] (2013). The method described in [10] uses foreground blobs and estimates the vertical vanishing point and maximum-likelihood focal length. Publication [11] concentrates on calibration of multi-camera networks. This method uses the result of [10] to make a coarse estimation of the camera parameters, of which only the focal length can be used for comparison. The resulting estimated focal lengths and the ground truth are presented in Table 4. The proposed algorithm has the highest accuracy. Note that

Table 1. Ground truth and estimated values of camera parameters of the City Center datasets

Sequence		Tilt (deg)	Roll (deg)	f (pixels)	Height (m)
City C 1 1920 × 1080	GT	107.66	−0.92	1693	6.02
	Est	107.88	−1.38	1631	6.16
City C 2 1920 × 1080	GT	107.56	−0.71	1368	4.60
	Est	107.99	2.95	1397	4.37

Table 2. Ground truth and estimated values of camera parameters of the public datasets (No. of detections: 905, 1,958 and 1,665.)

Sequence		Tilt (deg)	Roll (deg)	f (pixels)	Height (m)
Indoor 1280 × 960	GT	104.31	0.07	1048	4.57
	Est	103.98	−0.20	787	3.78
Outdoor 1280 × 960	GT	109.11	0.96	1198	8.78
	Est	108.89	−0.18	1019	8.83
Terrace 360 × 288	GT	108.41	1.19	807	2.45
	Est	105.50	1.82	850	1.93

Table 3. Estimated distances in City Center datasets

Sequence	City Center 1	City C 2	City C 3
Error (%)	2.48	2.62	3.7

Table 4. Estimated focal lengths for the Outdoor dataset

Algorithm	GT	Liu [10]	Liu [11]	Prop. alg
f (pixels)	1,198	1,545	1,427	1019
Error (%)	-	29	19	15

the detected focal length is smaller than the actual value. This is due to the fact that the horizon line is detected slightly below the actual value and the vertical vanishing point is detected above the actual value (discussed later). However, this does not affect the detected camera orientation. Finally, the three public datasets are calibrated using the proposed algorithm, of which the results are presented in Table 2. For all sequences, the parameters are estimated accurately. The roll is detected up to 1.14° accuracy, the focal length up to 179 pixels and the camera height up to 0.53 m. The tilt estimation of the Terrace sequence is less accurate. Because of the low camera height (2.45 m), the detector was not working optimally, resulting in noisy head and feet detections. Moreover, the small focal length combined with the small camera tilt makes the calculation of the tilt sensitive to noise. When the detected focal length is smaller than the actual value, the detected camera height will also be lower than the actual value. However, if other camera parameters are correct, the accuracy of detected distances in the scene will not be influenced. We have found empirically that the errors compensate exactly (zero error) but this is difficult to prove analytically.

Note that the performance of our calibration cannot be fully benchmarked, since insufficient data is available from the algorithms from literature. Moreover, implementations are not available so that a full simulation can also not be performed. Most of the methods that use pedestrians to derive a vanishing point use controlled scenes with restrictions (e.g. numbers, movement etc.), so that the method do often not apply to unconstrained datasets. As indicated above, in some cases the focal length can be used. All possible objective comparisons have been presented.

Comparing our algorithm with [11] when ignoring our pre- and post-processing stages will lead to a similar performance, because both algorithms use the idea of vanishing point and horizon line estimation from pedestrian locations. In our extensive simulation experiments we have shown that our novel pre- and post-processing improve performance. Specifically, pre-processing improves camera tilt estimation from 3.7 to 1.3° error and post-processing further reduces the error to 0.7°. This strongly suggests that our algorithm outperforms [11].

5 Discussion

The proposed camera model is based on assumptions that do not always hold, e.g. zero skew, square pixels and a flat ground plane. This inevitably results in errors in detected camera parameters and measured distances in the scene. Despite these imperfections, the algorithm is capable of detecting distances in the scene with a limited error of only 3.7 %. The camera roll is affected by a horizontal offset of the principal point, whereas the effect on the other camera parameters of the model imperfections is negligible.

In some of the experiments, we observe that the estimated focal length has a significant error (Indoor sequence in Table 2). The estimated focal length and camera height are related and a smaller estimated focal length results in a smaller estimated camera height. Errors in the derived focal length are thus compensated by the detected camera height and have no influence on detected distances in the scene. Consequently, errors in the focal length do not hamper highly accurate determination of position information of moving objects in the scene.

6 Conclusion

We have presented a novel fully automatic camera calibration algorithm for monocular stationary cameras. We focus on surveillance scenes where typically pedestrians move through the camera view, and use only this information as input for the calibration algorithm. The system can also be used for scenes with other moving objects. After collecting location information from several pedestrians, a full camera calibration matrix is generated based on vanishing-point geometry. This matrix can be used to calculate the real-world position of any moving object in the scene.

First, we propose a pre-processing step which improves estimation of the vertical vanishing point, reducing the error in camera tilt estimation from 3.7 to 1.3°. Second, a novel post-processing stage exploits the height distribution of pedestrians to improve horizon line estimation, further reducing the tilt error to 0.7°. As a third contribution, the scale factor is determined using an average body height, enabling extraction of metric information without manual measurement in the scene. Next, we have performed extensive simulations of the total calibration algorithm for a large range of camera configurations. Monte-Carlo simulations have been used to accurately model the error sources and have shown that derived camera parameters are accurate. Even metric information can be extracted with a low average and maximum error of 1.95 % and 3.2 %, respectively. Benchmarking of the algorithm has shown that the estimated focal length of our system is more accurate than the reported systems in literature. Finally, the algorithm is evaluated using several real-world surveillance datasets in which no restrictions are made on pedestrian movement and position. In real datasets, the error figures are largely the same (metric errors of max. 3.7 %) as in the simulations which confirms the feasibility of the solution.

References

1. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple object tracking using K-shortest paths pptimization. IEEE Trans. Pattern Anal. Mach. Intell. (2011)
2. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893, June 2005
3. Dubska, M., Herout, A., Sochor, J.: Automatic camera calibration for traffic understanding. In: Proceedings of the British Machine Vision Conference. BMVA Press (2014)
4. Faugeras, O.D., Toscani, G.: The calibration problem for stereo. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 1986, Miami Beach, FL, 22–26 June 1986, pp. 15–20. IEEE (1986). IEEE Publ. 86CH2290-5
5. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2004). ISBN 0521540518
6. Hartley, R.I.: Self-calibration from multiple views with a rotating camera. In: Eklundh, J.-O. (ed.) ECCV 1994. LNCS, vol. 800, pp. 471–478. Springer, Heidelberg (1994). doi:10.1007/3-540-57956-7_52
7. Huang, S., Ying, X., Rong, J., Shang, Z., Zha, H.: Camera calibration from periodic motion of a pedestrian. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016
8. Krahnstoever, N., Mendonca, P.: Bayesian autocalibration for surveillance. In: Tenth IEEE International Conference on Computer Vision, ICCV 2005, vol. 2, pp. 1858–1865, October 2005
9. Kusakunniran, W., Li, H., Zhang, J.: A direct method to self-calibrate a surveillance camera by observing a walking pedestrian. In: Digital Image Computing: Techniques and Applications, DICTA 2009, pp. 250–255, December 2009
10. Liu, J., Collins, R.T., Liu, Y.: Surveillance camera autocalibration based on pedestrian height distributions. In: British Machine Vision Conference (BMVC) (2011)
11. Liu, J., Collins, R., Liu, Y.: Robust autocalibration for a surveillance camera network. In: 2013 IEEE Workshop on Applications of Computer Vision (WACV), pp. 433–440, January 2013
12. Lv, F., Zhao, T., Nevatia, R.: Self-calibration of a camera from video of a walking human. In: Proceedings of 16th International Conference on Pattern Recognition, 2002, vol. 1, pp. 562–567 (2002)
13. Lv, F., Zhao, T., Nevatia, R.: Camera calibration from video of a walking human. IEEE Trans. Pattern Anal. Mach. Intell. **28**(9), 1513–1518 (2006)

14. Maybank, S.J., Faugeras, O.D.: A theory of self-calibration of a moving camera. Int. J. Comput. Vision **8**(2), 123–151 (1992). http://dx.doi.org/10.1007/BF00127171
15. Micusik, B., Pajdla, T.: Simultaneous surveillance camera calibration and foot-head homology estimation from human detections. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1562–1569, June 2010
16. Millar, W.: Distribution of body weight and height: comparison of estimates based on self-reported and observed measures. J. Epidemiol. Community Health **40**(4), 319–323 (1986)
17. Orghidan, R., Salvi, J., Gordan, M., Orza, B.: Camera calibration using two or three vanishing points. In: 2012 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 123–130, September 2012
18. Possegger, H., Rther, M., Sternig, S., Mauthner, T., Klopschitz, M., Roth, P.M., Bischof, H.: Unsupervised calibration of camera networks and virtual PTZ cameras. In: Proceedings of Computer Vision Winter Workshop (CVWW) (2012). Supplemental Video, Dataset, Code
19. Zhang, Z.: A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. **22**(11), 1330–1334 (2000)