IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 7, 2018, accepted December 22, 2018, date of publication January 8, 2019, date of current version January 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2891224

ESTHER: Joint Camera Self-Calibration and Automatic Radial Distortion Correction From Tracking of Walking Humans

ZHENG TANG[1], (Student Member, IEEE), YEN-SHUO LIN[2], KUAN-HUI LEE[3], JENQ-NENG HWANG[1], (Fellow, IEEE), AND JEN-HUI CHUANG[4], (Senior Member, IEEE)

[1]Department of Electrical and Computer Engineering, University of Washington, Seattle, WA 98195, USA
[2]Applied Materials, Santa Clara, CA 95054, USA
[3]Toyota Research Institute, Los Altos, CA 94022, USA
[4]Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan

Corresponding author: Zheng Tang (zhtang@uw.edu)

This work was supported in part by Prism Skylabs, Inc., in part by Yunshiwanwei, Inc., and in part by the Ministry of Science and Technology of Taiwan under Grant 104-2917-I-009-022.

ABSTRACT Camera calibration and radial distortion correction are the crucial prerequisites for many applications in image big data and computer vision. Many existing works rely on the Manhattan world assumption to estimate the camera parameters automatically; however, they may perform poorly when there was lack of man-made structure in the scene. As walking humans are the common objects in video surveillance, they have also been used for camera self-calibration, but the main challenges include the noise reduction for the estimation of vanishing points, the relaxation of assumptions on unknown camera parameters, and the radial distortion correction. In this paper, we present a novel framework for camera self-calibration and automatic radial distortion correction. Our approach starts with the reliable human body segmentation that is facilitated by robust object tracking. Mean shift clustering and Laplace linear regression are, respectively, introduced in the estimation of the vertical vanishing point and the horizon line. The estimation of distribution algorithm, an evolutionary optimization scheme, is then utilized to optimize the camera parameters and the distortion coefficients, in which all the unknowns in camera projection can be fine-tuned simultaneously. Experiments on the three public benchmarks and our own captured dataset demonstrate the robustness of the proposed method. The superiority of this algorithm is also verified by the capability of reliably converting 2D object tracking into 3D space.

INDEX TERMS Camera calibration, estimation of distribution algorithm, multiple object tracking, radial distortion correction, self-calibration, video surveillance.

I. INTRODUCTION

We have witnessed in recent years an unprecedented explosion in the availability of and access to image big data, which contribute to the rapid development of computer vision algorithms. In many applications, such as 3D object tracking [1], [2], people localization [3] and 3D scene reconstruction [4], we need to establish the correspondence between the 2D image plane and the 3D space in real world. Most existing works adopt the pinhole camera model to compute the 3D-to-2D projection relationship, i.e., camera calibration. The camera parameters for projection consist of intrinsic parameters, which encode the *camera coordinate system* (CCS), and extrinsic parameters, which describe the transformation to the *world coordinate system* (WCS). Sometimes, the camera may also suffer from radial distortion, manifested in form of the ''fish-eye'' effect. The computation of camera parameters and distortion coefficients can be formulated as a Perspective-n-Point problem when sufficient measurements of 3D points are available, which may be derived from some calibration templates. However, these manual solutions require time-consuming annotation and interaction at the scene, which make them infeasible for a large-scale camera network. Moreover, for the widely installed *pan–tilt–zoom* (PTZ) cameras, the camera parameters may change occasionally that makes the previous measurements invalid. Therefore, many approaches have been proposed to

automatically calibrate the cameras based on assumptions on the camera scenes. This category of methods is termed as *camera self-calibration*.

Most methods in camera self-calibration try to find the vanishing points of parallel lines in the 3D real world. Caprile and Torre [5] first propose to recover both intrinsic and extrinsic parameters from given vanishing points. Later, many works [6], [7], based on the Manhattan world assumption, utilize vanishing points from regular architectural structures in the scene for camera calibration. However, the Manhattan world assumption is invalid for many scenarios, where the observation of common video objects, e.g., pedestrians and vehicles, can thus be utilized for camera self-calibration.

Lv *et al.* [8] propose a method for camera self-calibration from observation of a human walking on a planar surface. Each human instance can be modeled as a vertical pole with constant height that is perpendicular to the ground plane, from which they calculate the vertical vanishing point, V_∞, and the horizon line, L_∞. Then the camera parameters can be computed based on some assumptions on the intrinsic camera parameters. Though many other algorithms [9]–[18] have been developed to improve their performance, this task is still facing a few challenges. First, Mohedano and Garcia [19] analyze the limitation of single-camera-based self-calibration from human tracking, from which they conclude that this formulation is not applicable for a camera with unknown aspect ratio of focal lengths, principal point coordinates and skew. In other words, to apply this method, we need to assume that the focal length is the only unknown intrinsic camera parameter to be estimated. The ambiguity caused by such assumption leads to the increase of reprojection error. The second challenge lies in noise reduction for the estimation of V_∞ and L_∞. The noise and outliers are mainly caused by the uncertainty in head/foot localization. Among the previous works, RANSAC has been the most popular approach adopted [9], [12], [14], [17]. Unfortunately, in most scenarios where the number of outliers overwhelms inliers, the performance of RANSAC degrades. Additionally, the threshold to indicate inliers in RANSAC needs to be fine-tuned for different scenarios. Last but not least, all the previous methods cannot be applied to a severely distorted camera, such as a wide-angle or fish-eye camera, which requires additional estimation of distortion coefficients.

In this paper, we propose a novel framework for joint camera self-calibration and automatic radial distortion correction from the tracking of walking humans. Our work has been partially described in [18]; here, we further introduce radial distortion correction by evolutionary optimization based on the minimization of human height variance. To the best of our knowledge, this is the first work on video-object-based automatic recovery from radial distortion. In addition, we give more detailed explanation on each algorithmic component and conduct new experiments on public benchmarks with in-depth analysis of the comparison results. We also illustrate how the proposed framework can benefit *multiple object tracking* (MOT). In brief, we first collect head/foot points of walking humans based on adaptive segmentation and tracking. Mean shift clustering and Laplace linear regression are respectively employed in the estimation of V_∞ and L_∞ to overcome the deficiencies of RANSAC. To relax the assumptions on unknown intrinsic camera parameters, we take advantage of the evolutionary algorithm to optimize camera parameters. The final step is to correct radial distortion, which also exploits evolutionary optimization to search for the optimal distortion coefficients.

The rest of this paper is organized as follows. In Section II, we give a brief review of related works. The methodology of our proposed framework is covered in Section III. Section IV presents the experimental results and detailed analyses. Finally, we draw the conclusion in Section V.

II. RELATED WORK

A. SELF-CALIBRATION FROM HUMAN TRACKING

Many related algorithms have been developed based on the method proposed in [8]. More specifically, Lv *et al.* [9] improve their own work by applying RANSAC in vanishing points estimation. They also optimize camera parameters based on the Levenberg-Marquardt (LM) algorithm. Krahnstoever and Mendonca [10] exploit Bayesian estimation for noise reduction. Junejo and Foroosh [11] adopt a different formulation based on two decomposed foot-to-head harmonic homologies, in which outliers are removed using the truncated quadratic function. Wu *et al.* [12] also apply RANSAC to the estimation of vanishing points from input head and foot locations. Kusakunniran *et al.* [13] introduce direct computation of projection matrix without decomposition into physical parameters. Liu *et al.* [14] present a new framework for optimizing camera parameters, such that the predicted relative human height distribution matches with the prior knowledge. Recently, Huang *et al.* [15] develop a novel scheme that detects the image points of toes on the ground plane, which can directly infer the two vanishing points on L_∞. The work [16] proposes pre- and post-processing stages to improve the estimation of V_∞ and L_∞. Führ and Jung [17] adopt a nonlinear cost function aiming to mostly align the orientation of the reprojected poles. In our previous work [18], the cost function to be minimized is designed as the reprojection error on the ground plane and we utilize evolutionary optimization to simultaneously optimize all the camera parameters.

Despite the improvement of these methods in noise reduction, there are still many difficulties to be addressed. First, as concluded in [19], the estimation of V_∞ and L_∞ depends on the unrealistic assumptions of fixed aspect ratio and principal point. In [9] and [17], there have been attempts to relax these assumptions, but their formulations can only simultaneously optimize three out of twelve variables in the projection matrix. Additionally, the method in [9] requires the prior knowledge of the height of each human. Other limitations of the mentioned works also prohibit their applications in real world. More specifically, in [8], [9], and [15], they assume

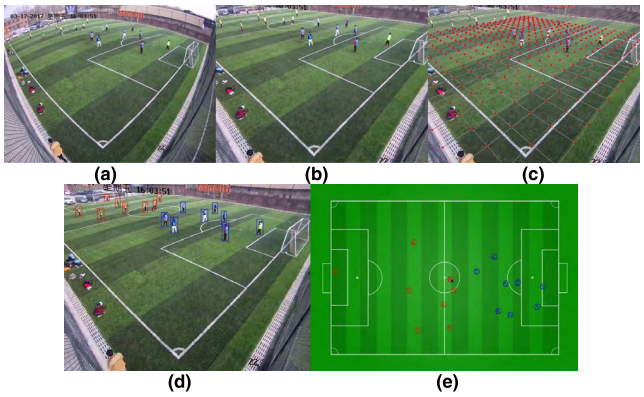


FIGURE 1. Demonstration of the proposed method. (a) Original frame image. (b) Frame image after automatic radial distortion correction. (c) 3D grid on the ground plane generated from camera self-calibration. (d) 2D object tracking. (e) Back projected 3D tracking.

that the leg-crossing period can be accurately detected. The method in [10] assumes that all objects are moving at constant velocity and the noise model of measurements is known. The work [14] assumes that the variation of relative human heights is sufficiently small. Finally, all the previous methods [9]–[18] assume that the camera is not distorted, and their only goal is to estimate the camera projection matrix.

B. AUTOMATIC RADIAL DISTORTION CORRECTION

Most existing approaches for automatic radial distortion correction exploit the Manhattan world assumption. Devernay and Faugeras [20] extract edges in a video sequence and optimize the distortion model such that it can best transform curved edges into straight line segments. The works [21], [22] also attempt to recover straight lines observed in the scene for distortion correction of multiple cameras. As far as we know, the proposed method is the first work that addresses radial distortion correction based on video objects.

C. ESTIMATION OF DISTRIBUTION ALGORITHM

The *estimation of distribution algorithm* (EDA), also known as the *probabilistic model-building genetic algorithm* (PMBGA), is a category that belongs to the class of *evolutionary algorithms* (EAs). It is inspired from the metaphor of biological evolution. The main difference between EDA and other EAs is that the probability model guiding the search for the optimal solution is explicit instead of implicit. EDA has been applied to some research in image processing, such as fitness evaluation in 3D vehicle modeling [23], but never in the field of camera calibration. In this paper, the *estimation of multivariate normal algorithm – global* ($\text{EMNA}_{\text{global}}$) [24], a type of multivariate EDA, is adopted for the optimization of camera parameters and distortion coefficients. The advantages of EDA over most of other metaheuristics have been reviewed in detail in [25], including its capability to adapt the operators to the problem structure, availability of roadmap in problem solution, prior knowledge exploitation and reduced memory storage. Furthermore, since the sampling of population at each generation can be built into parallel processing, the computation can be highly boosted when GPUs are available.

III. METHODOLOGY

The proposed framework mainly depends on the evolutionary algorithm to search for the optimal camera parameters and distortion coefficients, and it is entitled ESTHER, short for "Evolutionary Self-calibration from Tracking of Humans for Enhancing Robustness." The overview of our architecture is shown in Fig. 2.

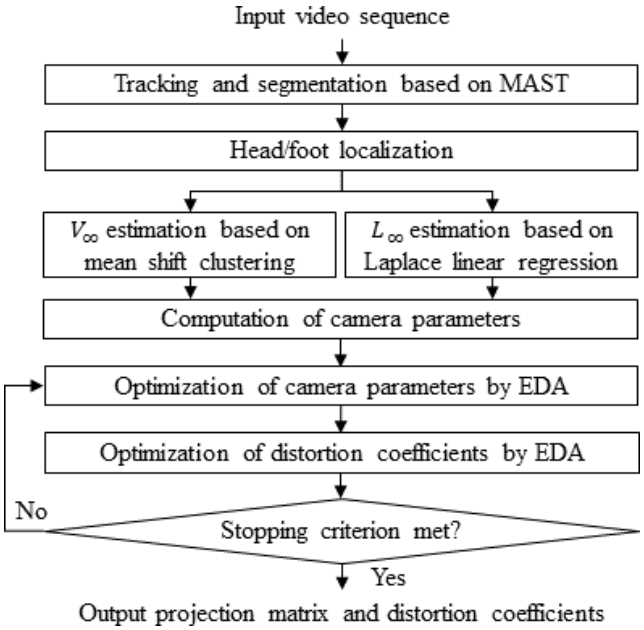


FIGURE 2. Flow diagram of ESTHER. The role of each algorithmic component is detailed in Section III.

The proposed method assumes that there is a major planar surface that people can walk on, i.e., the ground plane, in the *field of view* (FOV) of a single static camera. We also require at least one walking human with three different positions, which are not on the same straight line, observable in the scene. An approximate range of the camera height above the ground plane is assumed known. Compared with the assumptions made in other works discussed in Section II-A, our scenario is more realistic.

As shown in Fig. 3, the camera geometry, i.e., WCS, used in this paper is a Cartesian coordinate system in 3D space. The ground plane coincides with the plane defined by the X- and Y-axes. The Z-axis is pointing upward with respect to the ground plane and it passes through the camera position. The camera height is denoted as t_Z.

A. TRACKING AND HEAD/FOOT LOCALIZATION

To estimate V_∞ and L_∞ for camera calibration, the first step is to model each human instance as a pole perpendicular to the ground plane, which is equivalent to the 3D localization of head and foot points of the segmented human body.

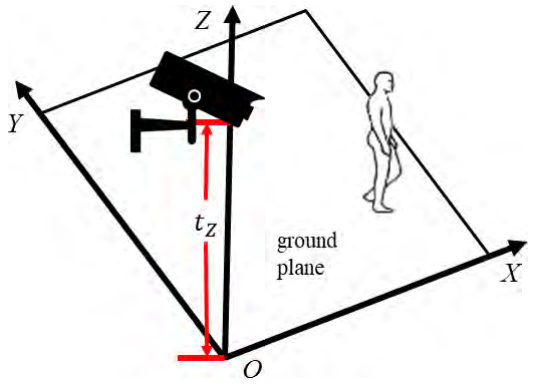


FIGURE 3. The camera geometry is a Cartesian coordinate system in 3D space.

Most previous methods [8]–[16] assume they have accurate human tracking and segmentation data as input. For more practical usage, instead, we combine the state-of-the-art multi-target tracking and segmentation to support head/foot localization.

In [26] and [27], we presented the *multi-kernel adaptive segmentation and tracking* (MAST) system for robust tracking and segmentation. The main goal of MAST is to address the problem of *object merging* during tracking by segmentation, i.e., failure in segmentation when some parts of the object(s) share similar color with the background.

To further improve the segmentation accuracy, we change the original segmentation module in MAST into one of the state-of-the-art change detection approaches, i.e., SuBSENSE [28], which relies on feedback from pixel-level background dynamics to adaptively control the local sensitivity and update rate. In the upgraded MAST system, the feedback penalty weight computed based on color similarity between the current frame and background is applied to the distance thresholds in SuBSENSE. Moreover, we add a shadow detection module based on YCbCr color space into SuBSENSE, which is triggered when a pixel is classified as foreground. Similarly, the feedback penalty weight computed from chromaticity similarity is used to define the thresholds in shadow detection. Thus, we enforce less shadow to be detected around the object region, so that more foreground can be preserved to support robust tracking by segmentation. Note that the original background model in MAST is built in single channel, however, here the mean of all background samples is used for background modeling.

The procedure of head/foot localization is demonstrated in Fig. 4. From the output of MAST, the bounding box and segmented foreground blob for each object instance can be derived. We compute the first moment of each foreground blob to determine its major axis. Each foreground blob can therefore be approximated as a pole representing its orientation. The two intersecting points between the major axis and the bounding box are chosen as the head and foot locations. This scheme has also been effectively adopted in several other works [14], [17], [29].

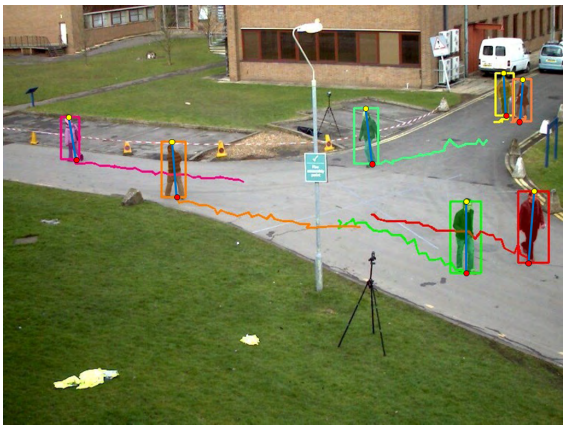


FIGURE 4. Head/foot localization from tracking and segmentation. The colored rectangles are bounding boxes from 2D tracking. The segmented foreground masks are also overlaid in color. The blue line segments denote the major axes of the foreground blobs. The yellow and red dots respectively denote the located head and foot points.

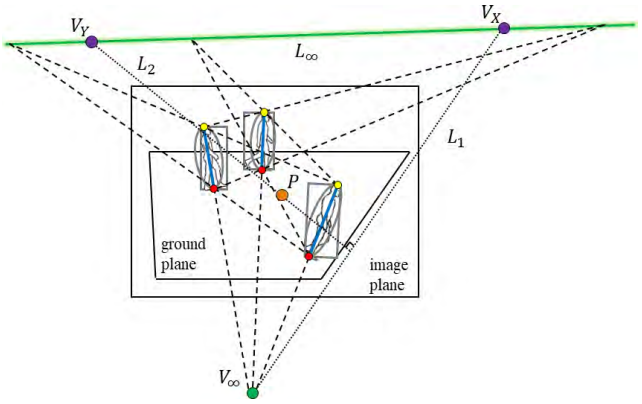


FIGURE 5. Geometry of vanishing points estimation in self-calibration (ideal scenario). The three blue poles represent three instances of the same person appearing at different frames, with yellow and red dots indicating their head and foot points, respectively. Point V_∞ in green marks the vertical vanishing point. The green line L_∞ denotes the horizon line. The dashed lines are auxiliary lines for the computation of V_∞ and L_∞. Points V_X and V_Y in purple are the other two vanishing points located on L_∞. Point P in orange gives the principal point of the camera. The dotted lines L_1 and L_2 are auxiliary lines to locate V_Y.

B. ESTIMATION OF VANISHING POINTS

All the instances of humans can be modeled as poles perpendicular to the ground plane. Ideally, if all the head and foot points are located correctly, i.e., there is neither noise nor outlier, and there is no radial distortion, V_∞ and L_∞ can be easily determined as illustrated in Fig. 5. The straight lines passing through the head and foot points at all object instances should converge at one point, i.e., the vertical vanishing point, V_∞. Similarly, if we draw a straight line to connect the head points of the same object at two different instances and another straight line connecting their foot points, the intersection of the two lines should lie on the horizon line, L_∞, which is defined as the extension of the ground plane at infinity. However, due to the existence of noise and outliers, this scenario is unrealistic in real world. There are always many candidate

points of V_∞, each generated by a pair of object instances. Similarly, the candidate points of L_∞ may not lie on the same straight line.

To estimate the location of V_∞, we propose a method based on mean shift clustering. The sensitivity to noise in head/foot localization is usually high for V_∞ estimation, because each object instance is associated with all the others in the point set of V_∞ candidates. Since the number of outliers can easily overwhelm inliers in most cases, the performance of RANSAC is not sufficiently robust. The problem can be better solved by applying mean shift clustering, because when spatially close clusters are merged together, the shape of the final cluster of inliers is not constrained. On the contrary, the cluster of inliers in RANSAC must form a circle. More specifically, the estimation of V_∞ is defined as

$$\begin{aligned} V_\infty &= \text{mean}\left(C^*\right), \\ \text{s.t. } C^* &= \arg\max_{C\in\{C\}} \#\left(C\right), \end{aligned} \tag{1}$$

where C denotes each cluster in mean shift clustering. The functions mean $(\cdot)$ and $\#(\cdot)$ respectively represent the computations of mean point and the number of candidate points. The mean shift window bandwidth is empirically set as $BW = 1 \times 10^3$ pixels in our experiments. In every iteration, an unvisited V_∞ candidate is randomly selected as the initial mean point. Then we conduct mean shift based on the window bandwidth BW until the moving distance of each mean point is smaller than a threshold $\tau_{BW} = BW \times e^{-3}$. The iteration is repeated until there is no more unvisited point left. Then, all the clusters whose mean points are within $BW/2$ are merged together. The estimated V_∞ is chosen as the mean point of the cluster with the most inliers, whereas other clusters are treated as outliers.

Laplace linear regression is proposed for the estimation of L_∞. As discussed in Section II, a drawback of the traditional method based on RANSAC is that the threshold parameter needs to be fine-tuned depending on different camera views. Hence, we leverage robust linear regression based on probabilistic modeling to avoid this configuration. The noise modeling by linear regression using Gaussian distribution can perform poorly when there are outliers in the data. As deviations are penalized quadratically by squared error, outliers will have greater influence on the line fitting than inliers. On the other hand, if we use Laplace distribution, its heavy tails can enforce higher likelihood to be assigned to points far away without the need to perturb the line [30]. Therefore, the result will be more robust. The likelihood model of Laplace linear regression is given as

$$p\left(\mathbf{v}|\mathbf{u},\mathbf{w}\right) = \text{Laplace}\left(\mathbf{v}|\mathbf{w}^T\mathbf{u}\right) \propto \exp\left(-\left|\mathbf{v}-\mathbf{w}^T\mathbf{u}\right|\right), \tag{2}$$

where $\mathbf{u}$ and $\mathbf{v}$ are the vectors containing the 2D coordinates of the candidate points for L_∞, and $\mathbf{w}$ represents the parameters of L_∞ that we aim to estimate. This problem can be formulated as constrained optimization,

$$\begin{aligned} &\min_{\mathbf{w},\mathbf{r}} \sum\nolimits_l r_l = \min_{\mathbf{w},\mathbf{r}^+,\mathbf{r}^-} \sum\nolimits_l \left(r_l^+ + r_l^-\right), \\ &\text{s.t. } r_l^+ \geq 0, \quad r_l^- \geq 0, \ \mathbf{w}^T u_l + r_l^+ - r_l^- = v_l, \end{aligned} \tag{3}$$

in which $r_l \triangleq r_l^+ - r_l^-$ is the l'th residual that can be split into positive and negative residuals, so that the objective function becomes a linear objective. This problem can be solved by linear programming solvers such as CVX [31]. The standard formulation is as follows,

$$\min_{\boldsymbol{\theta}} \mathbf{f}^T\boldsymbol{\theta} \quad \text{s.t. } \mathbf{A}\boldsymbol{\theta} \leq \mathbf{b}, \quad \mathbf{A}_{\text{eq}}\boldsymbol{\theta} = \mathbf{b}_{\text{eq}}, \ \mathbf{LB} \leq \boldsymbol{\theta} \leq \mathbf{UB}, \tag{4}$$

where $\boldsymbol{\theta} = \left(\mathbf{w},\mathbf{r}^+,\mathbf{r}^-\right)$, $\mathbf{f} = [\mathbf{0},\mathbf{1},\mathbf{1}]$, $\mathbf{A}_{\text{eq}} = [\mathbf{u},\mathbf{I},-\mathbf{I}]$, $\mathbf{b}_{\text{eq}} = \mathbf{v}$, $\mathbf{LB} = [-\infty\mathbf{1},\mathbf{0},\mathbf{0}]$ and the rest are empty sets.

Finally, based on the estimated V_∞ and L_∞, the other two vanishing points that lie on L_∞, namely V_X and V_Y, can be computed. As demonstrated in Fig. 5, first we initialize the location of the principal point P at the center of the image. The optimization for a more accurate location of P will be addressed in Section III-D. The next step is to randomly locate a V_X on L_∞. Then we draw an auxiliary line L_1 that connects V_X and V_∞, and another line L_2 that is perpendicular to L_1 and passes through P. Since the principal point of a camera should be the orthocenter of the triangle formed by three vanishing points [5], V_Y can be located at the intersection between L_∞ and L_2.

C. COMPUTATION OF CAMERA PARAMETERS

In a general pinhole camera model, the goal of camera calibration is to find a 3×4 projection matrix $\mathbf{P}$ that can project every 3D point (X, Y, Z) to its corresponding 2D pixel location (u, v) by

$$[u, v, 1]^T \sim \mathbf{P} \cdot [X, Y, Z, 1]^T . \tag{5}$$

This projection matrix can be decomposed into three matrices, including the intrinsic parameter matrix $\mathbf{K}$ that contains five intrinsic parameters (focal length in x-direction f_u, focal length in y-direction f_v, coordinates of principal point c_u and c_v, and skew s), the rotation matrix $\mathbf{R}$ defined by three extrinsic parameters (roll angle around Z-axis γ, pan angle around Y-axis α, and tilt angle around X-axis β), as well as the translation matrix $\boldsymbol{T}$ with the other three extrinsic parameters (translation along X-axis t_X, translation along Y-axis t_Y, and translation along Z-axis t_Z). Their relations are provided below:

$$\begin{aligned} &\mathbf{P} = \mathbf{K} \cdot [\mathbf{R}|\mathbf{t}], \\ &\text{s.t., } \mathbf{K} = \begin{bmatrix} f_u & s & c_u \\ 0 & f_v & c_v \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} t_X \\ t_Y \\ t_Z \end{bmatrix}, \quad \text{and} \\ &\mathbf{R} = \mathbf{R}_Z \cdot \mathbf{R}_Y \cdot \mathbf{R}_X, \\ &\mathbf{R}_Z = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}, \end{aligned}$$

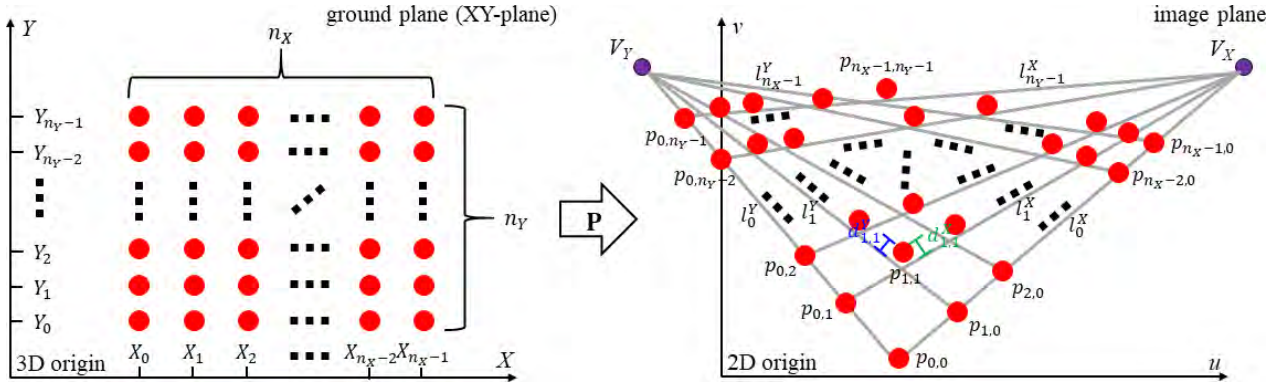


FIGURE 6. Optimization of camera parameters based on EDA. A set of $n_X \times n_Y$ grid points on the 3D ground plane are projected to 2D. Because of reprojection error, a projected 2D point $p_{i,j}$ ($i > 0, j > 0$) may not locate at the intersection between l_j^X, connecting V_X with $p_{0,j}$, and l_i^Y, connecting V_Y with $p_{i,0}$. Hence, we can use the distances from $p_{i,j}$ to these two lines, $d_{i,j}^X$ and $d_{i,j}^Y$, to indicate the reprojection error that we aim to minimize.

$$\mathbf{R}_Y = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix},$$
$$\mathbf{R}_X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\beta & -\sin\beta \\ 0 & \sin\beta & \cos\beta \end{bmatrix}. \tag{6}$$

Based on the assumptions on fixed intrinsic camera parameters [8]–[17], i.e., $f_u = f_v$, (c_u, c_v) located at the image center and $s = 0$, the camera parameters can be computed from given locations of P, V_X and V_Y as follows.

$$\gamma = \tan^{-1}\left(\frac{v_{V_Y} - v_{V_X}}{u_{V_X} - u_{V_Y}}\right),$$
$$f_u = f_v = \sqrt{-\left(v_{V_X}^{\text{rot}} \cdot v_{V_Y}^{\text{rot}} + u_{V_X}^{\text{rot}} \cdot u_{V_Y}^{\text{rot}}\right)}, \tag{7}$$
$$\text{s.t. } v_{V_X}^{\text{rot}} = \cos\gamma\left(v_P - v_{V_X}\right) - \sin\gamma\left(u_{V_X} - u_P\right),$$
$$v_{V_Y}^{\text{rot}} = \cos\gamma\left(v_P - v_{V_Y}\right) - \sin\gamma\left(u_{V_Y} - u_P\right),$$
$$u_{V_X}^{\text{rot}} = \cos\gamma\left(u_{V_X} - u_P\right) + \sin\gamma\left(v_P - v_{V_X}\right),$$
$$u_{V_Y}^{\text{rot}} = \cos\gamma\left(u_{V_Y} - u_P\right) + \sin\gamma\left(v_P - v_{V_Y}\right), \tag{8}$$
$$\beta = -\tan^{-1}\left(\frac{v_{V_X}^{\text{rot}}}{f_u}\right), \tag{9}$$
$$\alpha = -\tan^{-1}\left(\frac{\cos\beta \cdot u_{V_X}^{\text{rot}}}{f_u}\right). \tag{10}$$

According to the camera geometry in Fig. 3, the translation parameters t_X and t_Y are zero. And t_Z is equal to the camera height, whose approximate range is assumed known. The camera parameters will be further optimized by EDA, which will be addressed in Section III-D.

D. OPTIMIZATION OF CAMERA PARAMETERS BY EDA

As discussed in the review paper [19], the major limitation of all self-calibration methods based on the estimation of V_∞ and L_∞ is their unrealistic assumptions on unknown intrinsic camera parameters, which give rise to increasing reprojection error. To relax these assumptions, we formulate the optimization of camera parameters based on the minimization of reprojection error on the ground plane.

To start with, a set of $n_X \times n_Y$ grid points are generated on the ground plane in 3D space, i.e., the XY-plane. Using the initial camera parameters computed by estimated V_X and V_Y, the grid points can be projected to 2D (see Fig. 6). The horizontal and vertical gridlines are parallel with X- and Y-axes respectively. Within this grid, the 3D coordinates of the intersecting points are denoted as $\{(X_i, Y_j, 0)\}$, where $i = 0, 1, \ldots, n_X - 1$ and $j = 0, 1, \ldots, n_Y - 1$. Their corresponding projected 2D pixel locations are $\{p_{i,j}\}$. According to the definition of vanishing points, if we generate n_Y straight lines, noted l_j^X, on the image plane that each connects V_X with one of the points $p_{0,j}$ on the edge of the grid, all the other grid points $p_{i,j}$ ($i > 0, j > 0$) should fall on these lines. However, due to reprojection error, some $p_{i,j}$ may not lie on l_j^X, and the Euclidean distance between them is denoted as $d_{i,j}^X$. Similarly, the distance between $p_{i,j}$ and the corresponding straight line l_i^Y that connects V_Y and $p_{i,0}$ is denoted as $d_{i,j}^Y$. Now we can define the objective function of this optimization problem by

$$\mathbf{P}^* = \underset{\mathbf{P}\in\text{Rng}_\mathbf{P}}{\arg\min}\ \text{E}\left(d_{i,j}^X + d_{i,j}^Y\right),$$
$$\text{s.t., } d_{i,j}^X = \text{dist}\left(l_j^X, p_{i,j}\right), \quad d_{i,j}^Y = \text{dist}\left(l_i^Y, p_{i,j}\right), \tag{11}$$

where $\text{E}(\cdot)$ computes the expected value that is equivalent to the reprojection error on the ground plane. The function $\text{dist}(\cdot)$ measures Euclidean distance in pixel. In our formulation, $\mathbf{P}$ is decomposed into 11 camera parameters to be optimized. The initial range for each parameter, noted $\text{Rng}_\mathbf{P}$, is empirically set as $0.02 \times f_u$ for f_u, $0.02 \times f_v$ for f_v, 20 pixels for c_u and c_v, 20 degrees for γ, β and α, 200 mm for t_Z, and 0 for s, t_X and t_Y.

The optimization problem in (11) is formulated as multivariate EDA. The local optima of camera parameters, which

Algorithm 1 Optimization of Camera Parameters by EDA

input: initial range $\text{Rng}_{\mathbf{P}}$, sample size of initial population R, sample size of selected population $N < R$, maximum number of generations $g_{\max}$, stopping threshold of decreasing ratio τ_c

output: optima of camera parameters

1: generate initial population P $(0) \longleftarrow R$ sets of parameters sampled uniformly in the 11D space within $\text{Rng}_{\mathbf{P}}$; $g \longleftarrow 0$;
2: **while** ($g > 1$ and $\frac{c_{g-2}-c_{g-1}}{c_{g-2}} > \tau_c$) and $g < g_{\max}$ **do**
3: acquire each set of parameters from P (g);
4: project $n_X \times n_Y$ 3D grid on the ground plane to 2D;
5: measure error distance $d_{i,j}^X$ from each $p_{i,j}$ to l_j^X;
6: measure error distance $d_{i,j}^Y$ from each $p_{i,j}$ to l_i^Y;
7: select the population of promising solutions S $(g) \longleftarrow N$ individuals within P (g) that have smaller cost values $c_g = \text{E}\left(d_{i,j}^X + d_{i,j}^Y\right)$;
8: build probabilistic model M $(g) = N\left(\mu_g, \sigma_g\right) \longleftarrow$ eleven-variate normal density function from S (g);
9: P $(g+1) \longleftarrow R$ individuals sampled fromM (g);
10: $g \longleftarrow g + 1$;
11: **end while**
12: output μ_g of M (g).

induce minimal reprojection error, can be searched for in their given ranges simultaneously. Therefore, the assumptions on fixed aspect ratio and principal point can be relaxed effectively. We do not need to know the real human heights and other measurements in the scene for this formulation. Following the conventional EDA pseudocode [24], the detailed algorithmic procedure is illustrated in Algorithm 1. The sizes of initial and selected populations are empirically set as 2,000 and 20, respectively. The stopping criterion is that the decreasing ratio of the reprojection error is smaller than a threshold τ_c or the number of generations is larger than $g_{\max}$. In our experiments, we set $\tau_c = 0.1$ and $g_{\max} = 100$. As for the 3D grid on the ground plane, its size is empirically set as 10×10, where each grid point is 1 meter away from its neighbors.

E. RADIAL DISTORTION CORRECTION BY EDA

A major problem of the above procedure is that it does not work for a camera suffering from radial distortion, where L_∞ becomes a nonlinear curve and the vertical poles would not converge at V_∞. To address this issue, we propose to optimize the distortion coefficients by EDA that will enable camera self-calibration for wide-angle cameras.

For a pixel point (u, v) in a distorted frame image, the corrected pixel point (u', v') can be represented as

$$\begin{aligned} u' &= u\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right), \\ v' &= v\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right), \\ \text{s.t., } r^2 &= u^2 + v^2, \end{aligned} \tag{12}$$

where $\mathbf{k} = [k_1, k_2, k_3]^T$ is the vector of distortion coefficients to be estimated.

From (5), the projection of head and foot points to their corresponding pixel locations is given as

$$\begin{aligned} \left[\lambda u'_{\text{head}}, \lambda v'_{\text{head}}, \lambda\right]^T &= \mathbf{P} \cdot [X_{\text{foot}}, Y_{\text{foot}}, H, 1]^T, \\ \left[\lambda u'_{\text{foot}}, \lambda v'_{\text{foot}}, \lambda\right]^T &= \mathbf{P} \cdot [X_{\text{foot}}, Y_{\text{foot}}, 0, 1]^T, \end{aligned} \tag{13}$$

where λ is the scale factor and H is the human height in 3D space. The X- and Y-coordinates of head and foot points are considered the same, because we assume that the human body is always standing upright on the ground plane.

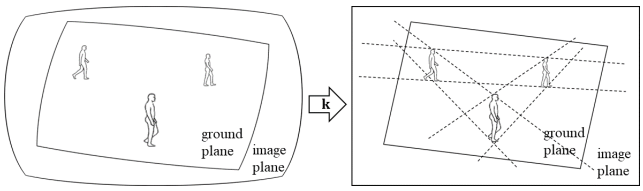


FIGURE 7. Radial distortion correction by EDA optimization. The vector of radial distortion coefficients, k, is optimized based on the minimization of human height variance.

As demonstrated in Fig. 7, it is intuitive that the measured 3D height of the same walking person can vary largely when the camera is under radial distortion. Thus, the objective function of this optimization problem is designed as,

$$\begin{aligned} \mathbf{k}^* &= \underset{\mathbf{k} \in \text{Rng}_{\mathbf{k}}}{\arg\min}\, \text{E}\left(\Delta H_{o,t}^2\right), \\ \text{s.t., } \Delta H_{o,t} &= \frac{H_{o,t} - \overline{H_o}}{\overline{H_o}}, \end{aligned} \tag{14}$$

where E $(\cdot)$ computes the expected value and $\Delta H_{o,t}$ is the relative human height offset of the o'th object at the t'th frame. The human height $H_{o,t}$ can be solved from (13). The mean of the o'th object's height is denoted as $\overline{H_o}$. $\text{Rng}_{\mathbf{k}}$ is the initial range for the optimization of $\mathbf{k}$. The normalization in the relative human height is to mitigate the influence of height offset between different people. This nonlinear optimization problem can be solved using EDA, where the probabilistic model is a three-variate normal distribution.

The detailed algorithmic procedure is described in the form of pseudocode in Algorithm 2. The distortion coefficients will gradually converge to the values that generate the lowest variance of the relative human height. All the configuration settings are the same as Algorithm 1, except the initial range $\text{Rng}_{\mathbf{k}}$, which is set as 0.5 for k_1, 5.0 for k_2 and 0 for k_3.

Algorithm 1 and Algorithm 2 can be considered as a two-stage evolutionary optimization process. They are repeated iteratively until the stopping criterion is met, i.e., the decreasing ratio of relative human height variance is smaller than a specific threshold, which is empirically set as 0.01.

IV. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed method, we conduct various experiments on video sequences from three public benchmarks and our own captured dataset. There are two

Algorithm 2 Radial Distortion Correction by EDA

input: initial range $\mathrm{Rng}_{\mathbf{k}}$, sample size of initial population R, sample size of selected population $N < R$, maximum number of generations $g_{\max}$, stopping threshold of decreasing ratio τ_c, optimized projection matrix $\mathbf{P}^*$

output: optimal distortion coefficients

1: generate initial population $\mathrm{P}(0) \longleftarrow R$ sets of distortion coefficients sampled uniformly in the 3D space within $\mathrm{Rng}_{\mathbf{k}}$; $g \longleftarrow 0$;
2: **while** ($g > 1$ and $\frac{c_{g-2}-c_{g-1}}{c_{g-2}} > \tau_c$) and $g < g_{\max}$ **do**
3: acquire each set of coefficients from $\mathrm{P}(g)$;
4: correct pixel locations by (12) and estimate the 3D human height of each instance by solving (13);
5: calculate $\Delta H_{o,t}$ in (14);
6: select the population of promising solutions $\mathrm{S}(g) \longleftarrow N$ individuals within $\mathrm{P}(g)$ that have smaller cost values $c_g = \mathrm{E}\left(\Delta H_{o,t}^2\right)$;
7: build probabilistic model $\mathrm{M}(g) = N\left(\mu_g, \sigma_g\right) \longleftarrow$ three-variate normal density function from $\mathrm{S}(g)$;
8: $\mathrm{P}(g+1) \longleftarrow R$ individuals sampled from $\mathrm{M}(g)$;
9: $g \longleftarrow g + 1$;
10: **end while**
11: output μ_g of $\mathrm{M}(g)$.

TABLE 1. Details of experimental video sequences.

Seq. #	Dataset	Res. (pix.)	Rt. (fps)	Len. (s)	No. of objects
1. Outdoor	[32]	1280 × 960	15	400	20
2. Indoor	[32]	1280 × 960	15	60	10
3. Terrace	[33]	360 × 288	25	200	8
4. PETS09-S2L1	[34]	768 × 576	7	114	19
5. AVG-TownCentre	[34]	1920 × 1080	2.5	225	226
6. Soccer-S1	Ours	2048 × 1536	25	120	16
7. Soccer-S2	Ours	2048 × 1536	25	120	16
8. Soccer-S3	Ours	1280 × 720	25	120	16
9. Soccer-S4	Ours	2048 × 1536	25	120	16

video sequences, *Outdoor*and *Indoor*, from the VPTZ benchmark [32] for virtual PTZ camera simulation and a sequence, *Terrace*, from the EPFL benchmark [33] for multi-camera pedestrian detection and tracking. These three sequences have also been adopted in the work [16] for experimental comparison. Besides, we use two video sequences from the *MOTChallenge* 3D benchmark [34], *PETS09-S2L1* and *AVG-TownCentre*. They have been used in [17] for the evaluation of self-calibration and 3D tracking. Finally, to emphasize our effectiveness in radial distortion correction, we also include four sequences that are synchronously recorded at a soccer game by fish-eye cameras. They are respectively denoted as *Soccer-S1*, *Soccer-S2*, *Soccer-S3* and *Soccer-S4*. The details of all test sequences are summarized in Table 1.

A. COMPARISON OF CAMERA SELF-CALIBRATION

The proposed method, ESTHER, is compared with several state-of-the-art approaches in camera self-calibration from human tracking listed as follows.

- Our earlier method [18], which does not include radial distortion correction based on the minimization of human height variance
- The method by Führ and Jung [17] which employs RANSAC for noise reduction and formulates a nonlinear optimization problem based on the reprojected poles of walking humans
- A recent method by Brouwers *et al.* [16], which adds a pre-processing step to filter away detection outliers and a post-processing step for tilt angle optimization based on human height distribution
- The method by Liu *et al.* [14] that utilizes predicted human height distribution to optimize the focal length
- Another method by Liu *et al.* [35] based on [14] but leverages multi-camera information
- The method by Wu *et al.* [12] that employs RANSAC for noise reduction in the estimation of V_∞ and L_∞
- The original method by Lv *et al.* [8] without any scheme of noise reduction or optimization

The works [18], [17], [16], [35] are considered the state-of-the-art in this field, as they are all published within the recent five years. The experimental results of [17], [16], [14], and [35] are derived from their published papers. As for [8], [12], and [18] and the proposed method, the head and foot points located from MAST are used as their input, where the default configuration parameters for MAST and SuBSENSE are applied. For [12], the RANSAC threshold for L_∞ estimation is fine-tuned for each video sequence, and the corresponding threshold for V_∞ estimation is set to be the same as our bandwidth in mean shift clustering, i.e., 1×10^3 pixels.

The experimental results of camera calibration on each of the nine test sequences are presented in Table 2. For evaluation, we measure the absolute differences from the ground truths for camera parameters, including f (the average of f_u and f_v), c_u, c_v, γ, β, and t_Z. Apparently, the proposed method, ESTHER, exhibits the best overall performance across all the metrics. The qualitative performance of ESTHER is displayed in Fig. 8. Especially, we demonstrate significant improvement on videos with strong radial distortion, i.e., Seq. #1, #2, #5, #6, #7, #8 and #9, which validates the effectiveness of the proposed scheme for radial distortion correction based on evolutionary optimization. In the other test sequences, i.e., Seq. #3 and #4, because the distortion effect is minor, our previous approach [18] also achieves robust performance. Thus, the advantage of relaxing assumptions on unknown intrinsic parameters is validated. All the other approaches assume that the principal point locates at the center of the frame image, but in most scenarios, there is a non-negligible distance between these two points. In our algorithm, however, the principal point coordinates can be effectively optimized through the minimization of reprojection error. Besides, we also generate better estimation of the focal length by relaxing the constraint on aspect ratio. The performance of the method [17] on Seq #4 and #5 is comparable to ours, due to the similar nonlinear

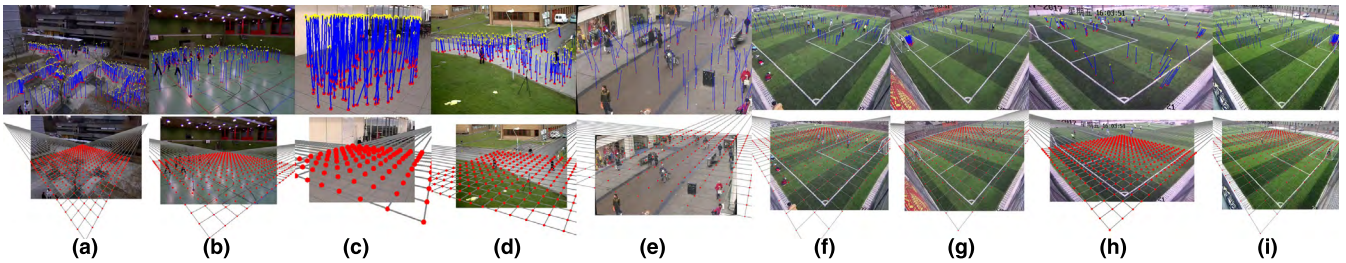


FIGURE 8. Qualitative visualization of ESTHER on test video sequences. The first row shows the located head/foot points. The second row shows the back projected 3D grid on the ground plane. (a) Seq. #1. (b) Seq. #2. (c) Seq. #3. (d) Seq. #4. (e) Seq. #5. (f) Seq. #6. (g) Seq. #7. (h) Seq. #8. (i) Seq. #9.

TABLE 2. Experimental comparison of camera calibration.

Seq. # & Method	Δf (pix.)	Δc_u (pix.)	Δc_v (pix.)	$\Delta\gamma$ (deg.)	$\Delta\beta$ (deg.)	Δt_Z (mm)
1 - ESTHER	**121.5**	*23.3*	**12.7**	*1.64*	*0.39*	**50**
1 - Tang et al. [18]	*124.6*	**19.2**	16.0	1.82	1.17	78
1 - Brouwers et al. [16]	179.0	43.9	*14.8*	**1.14**	**0.22**	*62*
1 - Liu et al. [14]	347.0	43.9	*14.8*	N/A[a]	N/A[a]	N/A[a]
1 - Liu et al. [35]	229.0	43.9	*14.8*	N/A[a]	N/A[a]	N/A[a]
1 - Wu et al. [12]	251.9	43.9	*14.8*	8.68	3.94	N/A[a]
1 - Lv et al. [8]	382.7	43.9	*14.8*	15.01	5.47	N/A[a]
2 - ESTHER	**126.5**	**15.1**	*13.7*	*2.61*	1.57	**97**
2 - Tang et al. [18]	*126.8*	*19.0*	**11.2**	2.90	*1.18*	*115*
2 - Brouwers et al. [16]	265.0	41.2	18.0	**0.27**	**0.33**	790
2 - Wu et al. [12]	362.0	41.2	18.0	6.45	2.64	N/A[a]
2 - Lv et al. [8]	520.3	41.2	18.0	8.93	3.98	N/A[a]
3 - ESTHER	**11.5**	**4.5**	*2.9*	*2.78*	2.07	*116*
3 - Tang et al. [18]	*13.1*	*5.3*	**2.8**	3.49	*1.75*	**112**
3 - Brouwers et al. [16]	43.0	11.5	9.6	**2.91**	**0.63**	520
3 - Wu et al. [12]	28.6	11.5	9.6	7.30	3.04	N/A[a]
3 - Lv et al. [8]	34.6	11.5	9.6	11.69	2.07	N/A[a]
4 - ESTHER	52.2	*13.8*	6.0	*2.46*	**1.45**	**294**
4 - Tang et al. [18]	**51.8**	**12.0**	7.9	**1.84**	*1.75*	*327*
4 - Führ et al. [17]	*52.0*	59.8	**5.4**	N/A[a]	N/A[a]	N/A[a]
4 - Wu et al. [12]	60.5	59.8	**5.4**	2.77	1.92	N/A[a]
4 - Lv et al. [8]	89.6	59.8	**5.4**	7.56	3.29	N/A[a]
5 - ESTHER	**158.5**	*24.9*	*15.9*	*3.17*	**1.89**	*176*
5 - Tang et al. [18]	200.1	25.4	16.2	**3.06**	*2.24*	**175**
5 - Führ et al. [17]	*197.1*	**0.5**	**0.5**	N/A[a]	N/A[a]	N/A[a]
5 - Wu et al. [12]	253.6	**0.5**	**0.5**	4.96	4.17	N/A[a]
5 - Lv et al. [8]	280.0	**0.5**	**0.5**	9.41	6.82	N/A[a]
6 - ESTHER	**185.2**	**14.6**	**21.1**	**3.14**	**1.26**	**86**
6 - Tang et al. [18]	*240.7*	*34.4*	*24.5*	*6.68*	*4.10*	*195*
6 - Wu et al. [12]	258.7	62.8	27.2	8.11	5.21	N/A[a]
6 - Lv et al. [8]	292.5	62.8	27.2	15.72	8.47	N/A[a]
7 - ESTHER	**191.3**	**22.1**	**17.6**	**2.01**	**1.53**	**121**
7 - Tang et al. [18]	*249.5*	*30.5*	38.8	*4.95*	*3.26*	*149*
7 - Wu et al. [12]	278.2	61.2	*24.1*	7.91	4.90	N/A[a]
7 - Lv et al. [8]	311.5	61.2	*24.1*	15.46	7.41	N/A[a]
8 - ESTHER	**123.3**	**4.9**	**9.6**	**1.80**	**0.92**	**119**
8 - Tang et al. [18]	*131.1*	*18.7*	*13.0*	*2.73*	*1.79*	*197*
8 - Wu et al. [12]	132.9	43.3	14.5	6.85	2.17	N/A[a]
8 - Lv et al. [8]	178.4	43.3	14.5	8.12	4.89	N/A[a]
9 - ESTHER	**219.5**	**16.5**	**19.3**	**2.33**	**1.49**	**162**
9 - Tang et al. [18]	260.0	*38.7*	*22.9*	*3.77*	*2.38*	*226*
9 - Wu et al. [12]	*258.7*	57.6	30.6	7.19	4.75	N/A[a]
9 - Lv et al. [8]	293.3	57.6	30.6	14.83	8.54	N/A[a]

Red-bold entries indicate the best results in the corresponding columns for each video sequence, and blue italics the second-best.

[a]Because the estimation of the corresponding camera parameters is not considered in these methods, the ground truths are applied.

optimization of camera parameters. The experimental results by Brouwers *et al.* [16] are only available on the first three video sequences. Because of their extra processing steps that fine-tune the rotation angles, they perform better in the estimation of γ and β, but the computation of the other camera parameters is less reliable. As for the method by Liu *et al.* [14], [35], they only compare their performance of focal length estimation on the *Outdoor*sequence. Though the cues from multiple cameras can be leveraged to improve estimation accuracy in [35], the final results are still far from matching our expectation. With noise removal by RANSAC, Wu *et al.* [12] enhance the reliability of the original work [8], but due to the lack of optimization process, their method fails in most cases. Finally, the poor performance of the original method [8] verifies the necessity of noise reduction and optimization schemes in camera self-calibration.

B. COMPARISON OF DISTORTION CORRECTION

To verify the effectiveness of our proposed human-tracking-based radial distortion correction, we compare with another method based on the Manhattan world assumption, similar to [20]–[22]. In the method for comparison, the strong edges are collected from the Sobel edge detector, preceded by Gaussian blur filtering (see Fig. 9). After filtering away short and weak edges, the strong edges, noted $\{l\}$, are each approximated by second-order polynomial regression. The cost function of the optimization problem is given as

$$\mathbf{k}^* = \arg\min_{\mathbf{k}\in \mathrm{Rng}_{\mathbf{k}}} \sum\nolimits_{l\in\{l\}} \mathrm{curv}\,(l), \tag{15}$$

where curv $(\cdot)$ computes the curvature of an edge segment. In (15), we search for an optimal set of distortion coefficients that can maximally recover the "curved" straight lines. To solve this problem, we can utilize EDA optimization, whose configuration is the same as Algorithm 2.

Experiments are conducted on the seven test sequences with strong radial distortion. The experimental results are summarized in Table 3, where the implementation based on the Manhattan world assumption is denoted as "ESTHER (MWA)." In nearly all the comparisons, the proposed scheme that minimizes human height variance outperforms its opponents. The qualitative performance of the two methods can be visualized in Fig. 10. ESTHER (MWA) tends to overfit the distortion parameters, which is obvious around the frame borders. It is because the detected edges in an image are usually noisy, containing some outliers that are not linear in the undistorted frame image, e.g., branches of trees, some sidelines on the sport courts, etc. They cannot be easily filtered away without prior knowledge on the scenes. However, the minimization of relative human height variance is more reliable against noise in most cases, leading to higher accuracy and more natural distortion correction.

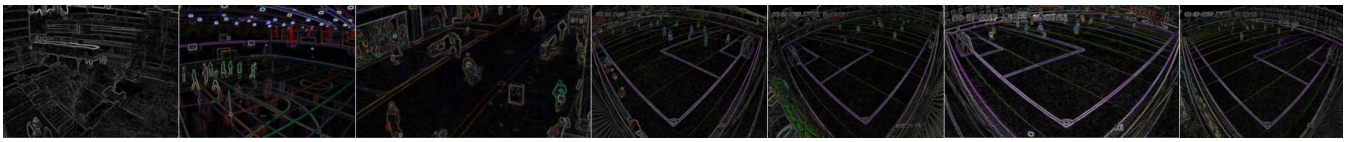


FIGURE 9. Edges detected by Sobel edge detector for the method to be compared with in radial distortion correction. (a) Seq. #1. (b) Seq. #2. (c) Seq. #5. (d) Seq. #6. (e) Seq. #7. (f) Seq. #8. (g) Seq. #9.

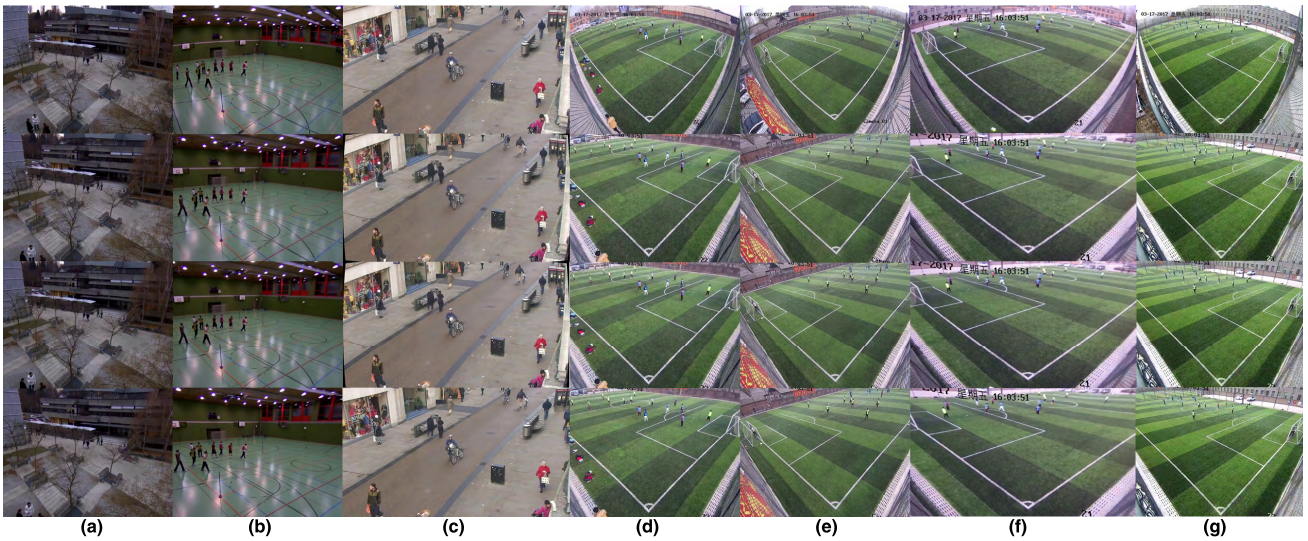


FIGURE 10. Comparison of radial distortion correction. The first row shows the original distorted frame images. The second row shows the images corrected by ground-truth distortion coefficients. The third row shows the images corrected by distortion coefficients estimated by ESTHER. The fourth row shows the images corrected by ESTHER (MWA). (a) Seq. #1. (b) Seq. #2. (c) Seq. #5. (d) Seq. #6. (e) Seq. #7. (f) Seq. #8. (g) Seq. #9.

TABLE 3. Experimental comparison of radial distortion correction.

Seq. # & Method	k_1	k_2
1 - Ground truth	-0.374	0.159
1 - ESTHER	**-0.383**	**0.176**
1 - ESTHER (MWA)	*-0.346*	*0.119*
2 - Ground truth	-0.365	0.131
2 - ESTHER	**-0.327**	**0.117**
2 - ESTHER (MWA)	*-0.479*	*0.198*
5 - Ground truth	-0.602	4.702
5 - ESTHER	**-0.595**	*4.730*
5 - ESTHER (MWA)	*-0.579*	**4.685**
6 - Ground truth	-0.312	0.098
6 - ESTHER	**-0.316**	**0.102**
6 - ESTHER (MWA)	*-0.348*	*0.124*
7 - Ground truth	-0.308	0.101
7 - ESTHER	**-0.322**	**0.107**
7 - ESTHER (MWA)	*-0.351*	*0.119*
8 - Ground truth	-0.469	0.225
8 - ESTHER	**-0.509**	**0.241**
8 - ESTHER (MWA)	*-0.593*	*0.278*
9 - Ground truth	-0.304	0.097
9 - ESTHER	**-0.319**	**0.103**
9 - ESTHER (MWA)	*-0.346*	*0.119*

Red-bold entries indicate the best results in the corresponding columns for each video sequence, and blue italics the second-best.

C. ABLATION STUDY

We further study the effect of each individual algorithmic component. For ablation study, we adopt the *Outdoor* sequence, i.e., Seq. #1, in our experiments. The experimental results are presented in Table 4, where "LLR" and "MSC" respectively stand for Laplace linear regression and mean shift clustering. We not only compare with the scenarios where some of the modules are missing, but also the cases when EDA is substituted by the LM algorithm for optimization and/or RANSAC is adopted for the estimation of vanishing points. All the experiments are conducted under the same implementations and configuration parameters.

In Table 4, we can observe that all the methods with either EDA or LM optimization show significant improvement in estimation accuracy, as the extra information from the scene and video objects is exploited in the minimization of cost functions. Moreover, the constraints on unknown intrinsic camera parameters are relaxed. Both EDA and LM algorithm can successfully minimize the cost values, but the effectiveness of evolutionary optimization is superior. It is because LM optimization is based on stochastic gradient descent, which starts searching from local region. But evolutionary algorithm directly works on the global solution domain, and thus can better avoid local optima.

The noise reduction in L_∞ and V_∞ estimation is key to the optimization of camera parameters, as accurate vanishing points usually generate good initial values for optimization. In Table 4, we can also learn the enhanced robustness by Laplace linear regression and mean shift clustering. In the estimation of L_∞, the line fitting based on probabilistic modeling is more reliable than RANSAC, as the entire set of data points is exploited. As for V_∞ estimation, since the shape of the cluster for inliers can be tightly defined in mean shift clustering, our proposed scheme also outperforms RANSAC.

TABLE 4. Ablation study of the proposed framework.

Method				Δf (pix.)	Δc_u (pix.)	Δc_v (pix.)	$\Delta\gamma$ (deg.)	$\Delta\beta$ (deg.)	Δt_Z (mm)	Δk_1	Δk_2	$\mathrm{E}(d_{i,j}^X + d_{i,j}^Y)$ (pix.)	$\mathrm{E}(\Delta H_{o,t})$ (%)
dist. coeff. opt.	cam. param. opt.	L_∞ est.	V_∞ est.										
EDA	EDA	LLR	MSC	**121.5**	*23.3*	**12.7**	*1.64*	**0.39**	*50*	**0.009**	**0.017**	**1.06e-3**	*1.47*
LM	EDA	LLR	MSC	128.4	26.8	*13.0*	**1.27**	0.94	**47**	0.058	0.032	*2.06e-1*	1.92
EDA	LM	LLR	MSC	129.7	31.2	17.5	1.85	1.11	58	*0.030*	*0.021*	2.95e-3	**1.43**
LM	LM	LLR	MSC	132.9	33.4	16.3	2.03	*0.72*	72	0.049	0.029	7.17e-1	2.05
N/A	EDA	LLR	MSC	*124.6*	**19.2**	16.0	1.82	1.17	78	N/A[a]	N/A[a]	5.69e-3	6.09
EDA	N/A	LLR	MSC	167.3	34.1	16.2	3.94	2.94	N/A[a]	0.051	0.042	3.69	1.90
N/A	N/A	LLR	MSC	185.1	43.9	14.8	5.06	3.26	N/A[a]	N/A[a]	N/A[a]	4.65	5.81
N/A	N/A	RANSAC	MSC	207.5	43.9	14.8	6.22	3.63	N/A[a]	N/A[a]	N/A[a]	3.12	4.97
N/A	N/A	LLR	RANSAC	261.6	43.9	14.8	8.99	3.46	N/A[a]	N/A[a]	N/A[a]	5.00	8.02
N/A	N/A	RANSAC	RANSAC	351.9	43.9	14.8	8.68	3.94	N/A[a]	N/A[a]	N/A[a]	4.21	5.91
N/A	N/A	N/A	MSC	409.3	43.9	14.8	9.69	4.32	N/A[a]	N/A[a]	N/A[a]	8.10	6.40
N/A	N/A	LLR	N/A	430.0	43.9	14.8	9.95	4.28	N/A[a]	N/A[a]	N/A[a]	5.83	5.79
N/A	N/A	N/A	N/A	382.7	43.9	14.8	15.01	5.47	N/A[a]	N/A[a]	N/A[a]	6.11	7.13

Red-bold entries indicate the best results in the corresponding columns, and blue italics the second-best.
[a]Because the estimation of the corresponding camera parameters is not considered in these methods, the ground truths are applied.

D. COMPUTATIONAL COMPLEXITY ANALYSIS

The computational complexity analysis of our proposed algorithm is provided as follows. Assume that the number of collected object instances is N. In radial distortion correction, the relative human height offset is computed at every EDA generation, so the computation time is $\mathrm{O}(N)$. As for the implementation based on the Manhattan world assumption, the computation time is $\mathrm{O}(\#(l))$. Though the number of edge segments in a frame image is usually smaller than the number of human instances in a video sequence, the optimization performance can be highly sensitive to the quality of the selected frame image. As for the optimization of camera parameters, compared to other approaches using nonlinear optimization based on human height distribution [9], [17], whose computation time is $\mathrm{O}(N)$, our formulation based on reprojection error on the ground plane only depends on the size of the pre-defined 3D grid points. Thus, the computation time is only $\mathrm{O}(1)$. In mean shift clustering, we visit every candidate point once, each generated by a pair of human instances, so that the computation time is $\mathrm{O}(N^2)$, which is the *best conceivable runtime* (BCR). This may be slower than RANSAC based on random sampling, but we exploit all the useful information. Finally, in Laplace linear regression, the runtime is also $\mathrm{O}(N^2)$.

The proposed framework has been implemented in C++ with the support of the OpenCV 3 library. It is run on an Intel Core i7-7700HQ PC with 4 cores, 2.80 GHz processor and 16 GB RAM in the Windows 10 environment. To ensure fast computation, we start self-calibration and radial distortion correction once the numbers of candidate points for L_∞ and V_∞ estimation both exceed 1,000. After head/foot localization, the algorithmic process takes 48.7 seconds to complete.

E. APPLICATION TO 3D OBJECT TRACKING

An intuitive application of camera self-calibration is to back project the 2D tracking [36]–[38] into 3D space. In this section, we demonstrate the utilization of ESTHER in MOT.

In single-camera object tracking, both test sequences, *PETS09-S2L1* and *AVG-TownCentre*, are included in the *MOTChallenge* 3D benchmark [34]. They have also been adopted for experiments in the work by Führ and Jung [17], who test the self-calibration strategy with their own 3D tracking algorithm [39]. In our work, the camera projection matrix estimated by ESTHER is applied to the state-of-the-art tracking-by-detection method on the benchmark [34], i.e., MOANA [40]. In Table 5, we present the experimental results of object tracking, where the metric we use is the *multiple object tracking accuracy* (MOTA) [41]. The accurate back projection of object instances to 3D space by ESTHER largely improves the tracking accuracy of MOANA. Though the nonlinear optimization by Führ and Jung [17] can significantly improve their initial estimation, their tracking accuracy is still inferior to the proposed method. Finally, we also compare with another state-of-the-art, DP + NMS [42], and the baseline by Leal-Taixé *et al.* [43]. Their tracking predictions are also less reliable than ours. A qualitative demonstration of 2D-to-3D back projection by ESTHER is shown in Fig. 11. The demo video sequences are available on our website.[1]

TABLE 5. Experimental comparison of single-camera MOT in MOTA (%).

Method	*PETS09-S2L1*	*AVG-TownCentre*
3D MOANA [40] + ESTHER[a]	**81.5**	**46.1**
2D MOANA [40]	75.2	41.8
Führ et al. [17] opt.	55.3	*44.9*
Führ et al. [17] init.	51.4	19.9
DP+NMS [42]	58.1	38.0
Baseline [43]	*77.5*	35.9

Red-bold entries indicate the best results in the corresponding columns, and blue italics the second-best.
[a]Our performance can be found on the website of the *MOTChallenge* 3D benchmark [34]: https://motchallenge.net/tracker/MOANA.

V. CONCLUSION

In this paper, we present a novel framework for camera self-calibration and radial distortion correction from tracking of walking humans. There are three critical challenges to be overcome, i.e., the relaxation of assumptions on unknown intrinsic camera parameters, the estimation of vanishing

[1]Available at http://allison.ee.washington.edu/thomas/esther/

FIGURE 11. Back projection of MOT into 3D space based on MOANA. (a) Video frame with 2D trajectories. (b) Back projected trajectories in 3D.

points against noise, and the automatic computation of distortion coefficients. To address these problems, we propose several innovative schemes in the process of estimation and optimization. The main contributions of this paper in terms of novelty include: 1) evolutionary optimization of distortion coefficients based on human height variance minimization; 2) camera parameters optimization using EDA that aims to minimize the reprojection error on the ground plane; 3) mean shift clustering for the removal of outliers in the estimation of the vertical vanishing point; 4) the estimation of horizon line based on Laplace linear regression that avoids additional fine-tuning; 5) a robust segmentation and tracking system that can adaptively refine the foreground masks to support optimal head/foot localization; 6) state-of-the-art performance demonstrated on several public benchmarks and our own dataset, enabling applications in 3D object tracking.

REFERENCES

[1] K.-H. Lee, J.-N. Hwang, and S.-I. Chen, "Model-based vehicle localization based on 3-D constrained multiple-kernel tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 1, pp. 38–50, Jan. 2014.

[2] K.-H. Lee, J.-N. Hwang, J.-Y. Yu, and K.-Z. Lee, "Vehicle tracking iterative by Kalman-based constrained multiple-kernel and 3-D model-based localization," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2013, pp. 2396–2399.

[3] Y.-S. Lin, K.-H. Lo, H.-T. Chen, and J.-H. Chuang, "Vanishing point-based image transforms for enhancement of probabilistic occupancy map-based people localization," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5586–5598, Dec. 2014.

[4] Z. Tang, R. Gu, and J.-N. Hwang, "Joint multi-view people tracking and pose estimation for 3D scene reconstruction," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2018, pp. 1–6.

[5] B. Caprile and V. Torre, "Using vanishing points for camera calibration," *Int. J. Comput. Vis.*, vol. 4, no. 2, pp. 127–139, 1990.

[6] D. Liebowitz, A. Criminisi, and A. Zisserman, "Creating architectural models from images," *EuroGraphics*, vol. 18, no. 3, pp. 39–50, 1999.

[7] J. Deutscher, M. Isard, and J. MacCormick, "Automatic camera calibration from a single manhattan image," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2002, pp. 175–188.

[8] F. Lv, T. Zhao, and R. Nevatia, "Self-calibration of a camera from video of a walking human," in *Proc. IEEE Int. Conf. Pattern Recognit. (ICPR)*, vol. 1, Aug. 2002, pp. 562–567.

[9] F.-J. Lv, T. Zhao, and R. Nevatia, "Camera calibration from video of a walking human," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 9, pp. 1513–1518, Sep. 2006.

[10] N. Krahnstoever and P. R. S. Mendonça, "Autocalibration from tracks of walking people," in *Proc. Conf. Brit. Mach. Vis. (BMVC)*, 2006.

[11] I. Junejo and H. Foroosh, "Robust auto-calibration from pedestrians," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2006, pp. 92–97.

[12] Q. Wu, T.-C. Shao, and T. Chen, "Robust self-calibration from single image using RANSAC," in *Advances in Visual Computing*. Berlin, Germany: Springer, 2007, pp. 230–237.

[13] W. Kusakunniran, H. Li, and J. Zhang, "A direct method to self-calibrate a surveillance camera by observing a walking pedestrian," in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, 2009, pp. 250–255.

[14] J. Liu, R. T. Collins, and Y. Liu, "Surveillance camera autocalibration based on pedestrian height distributions," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2011.

[15] S. Huang, X. Ying, J. Rong, Z. Shang, and H. Zha, "Camera calibration from periodic motion of a pedestrian," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3025–3033.

[16] G. M. Brouwers, M. H. Zwemer, and R. G. Wijnhoven, "Automatic calibration of stationary surveillance cameras in the wild," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 743–759.

[17] G. Führ and C. R. Jung, "Camera self-calibration based on nonlinear optimization and applications in surveillance systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 5, pp. 1132–1142, May 2014.

[18] Z. Tang, Y.-S. Lin, K.-H. Lee, J.-N. Hwang, J.-H. Chuang, and Z. Fang, "Camera self-calibration from tracking of moving persons," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 260–265.

[19] R. Mohedano and N. Garcia, "Capabilities and limitations of mono-camera pedestrian-based autocalibration," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2010, pp. 4705–4708.

[20] F. Devernay and O. D. Faugeras, "Automatic calibration and removal of distortion from scenes of structured environments," *Proc. SPIE*, vol. 2567, pp. 62–73, Sep. 1995.

[21] J. H. Brito, R. Angst, K. Köser, and M. Pollefeys, "Radial distortion self-calibration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2013, pp. 1368–1375.

[22] C. Wu, "Critical configurations for radial distortion self-calibration," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 25–32.

[23] K.-H. Lee, J.-N. Hwang, and S.-I. Chen, "Model-based vehicle localization based on three-dimensional constrained multiple-kernel tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 1, pp. 38–50, Jun. 2014.

[24] P. Larranaga and J. A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, 2nd ed. Norwell, MA, USA: Kluwer, 2002.

[25] M. Hauschild and M. Pelikan, "An introduction and survey of estimation of distribution algorithms," *Swarm Evol. Comput.*, vol. 1, no. 3, pp. 111–128, 2011.

[26] Z. Tang, J.-N. Hwang, Y.-S. Lin, and J.-H. Chuang, "Multiple-kernel adaptive segmentation and tracking (MAST) for robust object tracking," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 1115–1119.

[27] Y.-G. Lee, Z. Tang, and J.-N. Hwang, "Online-learning-based human tracking across non-overlapping cameras," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 2870–2883, Oct. 2017.

[28] P. L. St-Charles, G. A. Bilodeau, and R. Bergevin, "SuBSENSE: A universal change detection method with local adaptive sensitivity," *IEEE Trans. Image Process.*, vol. 24, no. 1, pp. 359–373, Jan. 2015.

[29] T. Chen, A. D. Bimbo, F. Pernici, and G. Serra, "Accurate self-calibration of two cameras by observations of a moving person on a ground plane," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Sep. 2007, pp. 129–134.

[30] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. Cambridge, MA, USA: MIT Press, 2012.

[31] M. Grant and S. Boyd. *CVX: MATLAB Software for Disciplined Convex Programming*. Accessed: Jan. 4, 2016. [Online]. Available: http://stanford.edu/_boyd/cvx

[32] H. Possegger *et al.*, "Unsupervised calibration of camera networks and virtual PTZ cameras," in *Proc. Workshop Comput. Vis. Winter (CVWW)*, vol. 13, 2012.

[33] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua, "Multi-camera people tracking with a probabilistic occupancy map," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 267–282, Feb. 2008.

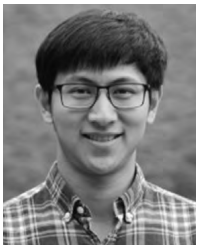
[34] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. (Apr. 2015). "MOTChallenge 2015: Towards a benchmark for multi-target tracking." [Online]. Available: https://arxiv.org/abs/1504.01942

[35] J. Liu, R. T. Collins, and Y. Liu, "Robust autocalibration for a surveillance camera network," in *Proc. IEEE Workshop Appl. Comput. Vis. (WACV)*, Jan. 2013, pp. 433–440.

[36] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[37] S. Zhang, W. Lu, W. Xing, and L. Zhang, "Learning scale-adaptive tight correlation filter for object tracking," *IEEE Trans. Cybern.*, to be published.

[38] S. Zhang, W. Lu, W. Xing, and L. Zhang, "Using fuzzy least squares support vector machine with metric learning for object tracking," *Pattern Recognit.*, vol. 84, pp. 112–125, Dec. 2018.
[39] Z. Tang, G. Wang, H. Xiao, A. Zheng, and J.-N. Hwang, "Single-camera and inter-camera vehicle tracking and 3D speed estimation based on fusion of visual and semantic features," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recogn. Workshops (CVPRW)*, Jun. 2018, pp. 108–115.
[40] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *Image Video Process.*, vol. 2008, no. 1, pp. 1–10, 2008.
[41] G. Führ and C. R. Jung, "Combining patch matching and detection for robust pedestrian tracking in monocular calibrated cameras," *Pattern Recognit. Lett.*, vol. 39, pp. 11–20, Apr. 2014.
[42] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jan. 2011, pp. 1201–1208.
[43] L. Leal-Taixé, G. Pons-Moll, and B. Rosenhahn, "Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Nov. 2011, pp. 120–127.



ZHENG TANG (S'16) received the B.Sc. (Eng.) degree (Hons.) in telecommunications engineering with management from a joint program between the Beijing University of Posts and Telecommunications, Beijing, China, and the Queen Mary University of London, London, U.K., in 2014, and the M.S. degree in electrical engineering from the University of Washington, Seattle, WA, USA, in 2016, where he is currently pursuing the Ph.D. degree in electrical and computer engineering.

From 2014 to 2018, he was a Research Assistant with the Information Processing Lab, University of Washington. Since 2018, he has been an Intelligent Video Analytics Intern with NVIDIA.

Mr. Tang received the Finalist IBM Best Student Paper Award and the Finalist Intel Best Student Paper Award at the 2016 International Conference on Pattern Recognition. He led the team from the University of Washington to win the Track 2 (AI City Applications) of the 2017 IEEE Smart World NVIDIA AI City Challenge. His team was the winner of the Track 1 (Traffic Flow Analysis) and the Track 3 (Multi-camera Vehicle Detection and Reidentification) in the AI City Challenge Workshop at the 2018 IEEE Conference on Computer Vision and Pattern Recognition.

YEN-SHUO LIN received the B.S. degree in electronic engineering from Chang Gung University, Taoyuan, Taiwan, in 2009, the M.S. degree in space science from National Central University, Taoyuan, in 2011, and the Ph.D. degree in computer science from National Chiao Tung University, Hsinchu, Taiwan, in 2016.

He has been a Software Engineer with Applied Materials, Inc., since 2017. His research interests include computer vision, machine learning, deep learning, and pattern recognition.

KUAN-HUI LEE received the B.S. degree from National Taiwan Ocean University, Keelung, Taiwan, in 2003, the M.S. degree from the National Cheng Kung University, Tainan, Taiwan, in 2005, and the Ph.D. degree from the University of Washington, Seattle, WA, USA, in 2015, all in electrical engineering.

From 2007 to 2009, he was with HTC Corporation, where he was involved in developing multimedia applications on smart phones. He has been a Research Scientist with the Toyota Research Institute, since 2016. His current research interests include computer vision and machine learning for autonomous driving.

JENQ-NENG HWANG (F'01) received the B.S. and M.S. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1981 and 1983, respectively, and the Ph.D. degree from the University of Southern California. In 1989, he joined the Department of Electrical and Computer Engineering (ECE), University of Washington, Seattle, WA, USA, where he has been promoted to Full Professor, in 1999. He was the Associate Chair for Research, from 2003 to 2005 and from 2011 to 2015. He is currently the Associate Chair for Global Affairs and International Development with the ECE Department. He is the Founder and the Co-Director of the Information Processing Lab., which received several AI City Challenges Awards. He has written more than 330 journals, conference papers, and book chapters in the areas of machine learning, multimedia signal processing, and multimedia system integration and networking. He has authored a textbook *Multimedia Networking: From Theory to Practice* (Cambridge University Press). He has close working relationship with the industry on multimedia signal processing and multimedia networking.

Dr. Hwang is currently a Founding Member of the Multimedia Signal Processing Technical Committee of the IEEE Signal Processing Society. He is also a member of the Multimedia Technical Committee of the IEEE Communication Society and the Multimedia Signal Processing Technical Committee of the IEEE Signal Processing Society. He received the 1995 IEEE Signal Processing Society's Best Journal Paper Award. He was the Program Co-Chair of the ICASSP 1998 and the ISCAS 2009. He has served as the Program Co-Chair of the IEEE ICME 2016. He is currently on the Editorial Board of the *ZTE Communications*, ETRI, IJDMB, and JSPS journals. He was the Society's Representative of the IEEE Neural Network Council, from 1996 to 2000. He has served as an Associate Editor for the IEEE T-SP, T-NN, T-CSVT, T-IP, and the *IEEE Signal Processing Magazine*.

JEN-HUI CHUANG (SM'06) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1980, the M.S. degree in electrical and computer engineering from the University of California at Santa Barbara, Santa Barbara, CA, USA, in 1983, and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, Urbana, IL, USA, in 1991.

Since then, he has been on the faculty of the Department of Computer and Information Science, National Chiao Tung University (NCTU). From 2004 to 2005, he was the Chairman of the Department of Computer and Information Science, NCTU. From 2006 to 2007, he was the Vice Dean of the College of Computer Science. He is currently serving as the Dean of the College of Computer Science, and also the Director of the Computer Vision Research Center, NCTU. His research interests include signal and image processing, computer vision and pattern recognition, robotics, and potential-based 3-D modeling. He was the President of the Chinese Image Processing and Pattern Recognition Society, from 2015 to 2016, and has served as the Governing Board Member of the International Association of Pattern Recognition, from 2013 to 2016. He is currently serving as a Governing Board Member of the IEEE Taipei Chapter. He has served as an Associate Editor of *Signal Processing*, from 2005 to 2010, and has been an Associate Editor of the *Journal of Information and Science and Engineering*, since 2009.