Hybrid Motion Model for Multiple Object Tracking in Mobile Devices

Yubin Wu, Hao Sheng, *Member, IEEE*, Yang Zhang, Shuai Wang, Zhang Xiong, *Member, IEEE*, and Wei Ke, *Member, IEEE*

***Abstract*—For an intelligent transportation system, multiple object tracking (MOT) is more challenging from the traditional static surveillance camera to mobile devices of the Internet of Things (IoT). To cope with this problem, previous works always rely on additional information from multivision, various sensors, or precalibration. Only based on a monocular camera, we propose a hybrid motion model to improve the tracking accuracy in mobile devices. First, the model evaluates camera motion hypotheses by measuring optical flow similarity and transition smoothness to perform robust camera trajectory estimation. Second, along the camera trajectory, smooth dynamic projection is used to map objects from image to world coordinate. Third, to deal with trajectory motion inconsistency, which is caused by occlusion and interaction of long time interval, tracklet motion is described by the multimode motion filter for adaptive modeling. Fourth, in tracklets association, we propose a spatiotemporal evaluation mechanism, which achieves higher discriminability in motion measurement. Experiments on MOT15, MOT17, and KITTI benchmarks show that our proposed method improves the trajectory accuracy, especially in mobile devices and our method achieves competitive results over other state-of-the-art methods.**

***Index Terms*—Hybrid motion model, mobile devices, multiple object tracking (MOT), tracking by tracklet.**

I. Introduction

MULTIPLE object tracking (MOT) plays an important role in many fields related to the Internet of Things (IoT) [1], such as intelligent transportation, video surveillance, etc. With the development of mobile devices, videos from automobile, UAV, robot, and mobile phone offer more data and bring greater challenges for MOT. In this article, we address the motion modeling problem of MOT in mobile devices. It is difficult to measure and predict object motion without additional sensor or precalibration.

Recently, remarkable progress has been achieved in object detection [2], [3], [4], which promotes the popular tracking-by-detection paradigm for MOT. Despite the high accuracy of detectors, false and missing detections still have impacts. To solve this problem, trackers [5], [6], [7], [8] are proposed to generate tracklets (short trajectories) with high confidence to reduce false positives (FPs) in detections. By tracking the objects as detections or tracklets, the key problem is to correctly associate objects among multiple frames. To find the optimal association, many successful algorithms are proposed, e.g., min-cost flow [9], conditional random field [10], [11], [12], multiple hypothesis tracking (MHT) [13], etc. The associations are built based on affinity measurement, which consider appearance consistency and motion prediction. In crowded scenarios, the lower similarity discrimination caused by illumination and pose changes makes appearance unreliable. Therefore, motion information is applied as another basis of association.

In the traditional surveillance system, cameras are assumed to be static, where motion information can be obtained intuitively through image coordinates. However, under mobile devices, the relative movement between the object and camera leads to great changes in image coordinates. The MOT system on the mobile device will have additional difficulty. For instance, as shown in Fig. 1, the object in the red box is almost in sync with the camera, so that its image coordinates change little. On the contrary, both the size and image coordinates of the blue one vary greatly. Our proposed model uses world coordinates, which eliminates the interference caused by mobile devices. As shown in Fig. 1(d), the movement of two objects has a similar pattern except for the opposite direction. This phenomenon makes it difficult to build a unified motion model by image coordinates. Therefore, associating with world coordinates is effective to measure object motion in mobile devices. In Fig. 1(c) and (d), the black-dashed arrow indicates the length between the beginning of two objects in the vertical direction. In the world coordinate, there is a significant distance between two objects while their image coordinates are highly similar instead. As a result, spatial constraints

Manuscript received 22 May 2022; revised 22 August 2022; accepted 26 October 2022. Date of publication 4 November 2022; date of current version 7 March 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2019YFB2102200; in part by the National Natural Science Foundation of China under Grant 61872025; in part by the Science and Technology Development Fund, Macau, under Grant 0001/2018/AFJ; and in part by the Open Fund of the State Key Laboratory of Software Development Environment under Grant SKLSDE2021ZX-03. *(Yubin Wu and Yang Zhang are co-first authors.) (Corresponding author: Hao Sheng.)*

Yubin Wu and Shuai Wang are with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and also with the School of Informatics, Beihang Hangzhou Innovation Institute Yuhang, Hangzhou 310023, China (e-mail: yubin.wu@buaa.edu.cn; shuaiwang@buaa.edu.cn).

Hao Sheng and Zhang Xiong are with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, also with the School of Informatics, Beihang Hangzhou Innovation Institute Yuhang, Hangzhou 311121, China, and also with the Faculty of Applied Sciences, Macao Polytechnic University, Macau, China (e-mail: shenghao@buaa.edu.cn; xiongz@buaa.edu.cn).

Yang Zhang is with the College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China (e-mail: yang_zh@mail.buct.edu.cn).

Wei Ke is with the Faculty of Applied Sciences, Macao Polytechnic University, Macau, China (e-mail: wke@mpu.edu.mo).

Digital Object Identifier 10.1109/JIOT.2022.3219627

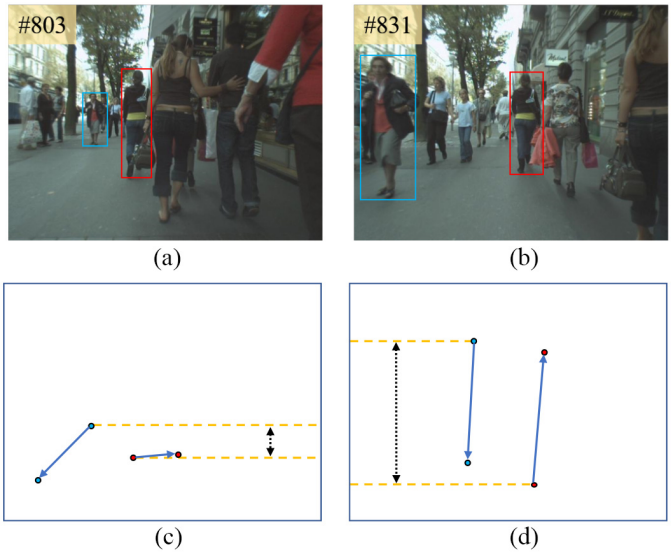

Fig. 1. (a) and (b) Detections of two pedestrians, colored in blue and red, with an interval of 2 s (frames 803–831). (c) and (d) Image coordinates and world coordinates (top view) change of two objects, respectively, where the moving direction is indicated by the arrow.

are sensitive when using the image coordinate, but the algorithm using world coordinates can detect and track objects more accurately. Consequently, it is necessary to construct a motion model with the world coordinate for MOT in mobile devices.

When tracking in mobile devices, acquiring the world coordinates of the objects always rely on multivision, depth sensors, radar, etc., which brings extra hardware and computing expenses. Considering that image coordinate take the camera as reference, camera trajectory can be used to compute the ground position of the objects indirectly. To obtain the camera trajectory, many motion-position estimation methods [14], [15], [16] are proposed. However, these approaches are not designed for MOT and unsuitable for locating the camera from the monocular system without additional information (e.g., location from GPS and depth from RGB-D sensor). In addition, a large number of moving objects in the video sequences bring difficulties for point matching between frames.

In this article, we propose a hybrid motion model to address the challenges posed by mobile devices in MOT. Due to different moving directions relative to camera devices, there are great changes in motion states of similar objects, which are amplified by the motion of the devices. Therefore, it is difficult for the MOT system to establish a unified motion model. To solve this problem, the existing MOT methods often rely on additional information, including multicamera, sensors, calibration, etc. These bring more hardware requirements and computational burdens. In a monocular uncalibrated system, our proposed method make full use of the information (background, detections, and horizon) in the video scene to estimate the camera trajectory, and realize the mapping of the objects from the image to the world coordinates along the camera trajectory. We use the geometric perspective with horizon line to simplify the calculation and reduce error in visual mapping. Furthermore, the high confidence tracklets for association is generated by the ground position and height of objects.

Multiobject tracking in the world coordinate can not only avoid the influence of motion devices but also increase the discrimination of motion measurement. The change of motion state between adjacent tracklets is approximately stationary, but is usually nonstationary between tracklets with long interval. In this case, the incompatibility motion state of tracklets leads to inconsistent measurement and prediction by a single motion model. To solve this problem, multimode motion filter (MMF) is proposed to estimate motion of adjacent and long spaced tracklets. MMF establishes prediction and error estimation for different motion modes. Meanwhile, we proposed a spatiotemporal evaluation mechanism (STEM) to evaluate the similarity of tracklets by motion metrics in MMF and appearance feature.

On MOT15, MOT17, and KITTI benchmarks, experiments demonstrate the higher tracking accuracy. As shown in the leaderboard, our method is competitive with other state-of-the-art trackers.

In summary, the main contributions of this article are summarized as follows.

1) To solve the problem of object motion modeling under mobile devices, we propose a hybrid motion model based on world coordinates using scene information.
2) To adapt to the monocular uncalibrated system and reduce the computational complexity of projection, we propose smooth dynamic projection for object coordinate mapping according to the perspective of the imaging system with horizon.
3) To solve incompatibility between adjacent and long spaced tracklets, MMF is established for the adaptability of modeling.
4) To provide accurate affinity measurement in the tracklets association, STEM is proposed with error variance estimator of motion.

The remainder of this article is organized as follows. Related work is discussed in Section II. The hybrid motion model for mobile devices is presented in Section III. MHT based on the hybrid motion model is described in Section IV. The experimental results are shown and analyzed in Section V followed by the conclusion in Section VI.

II. Related Work

In this section, we analyze the merits and weaknesses of recent tracking methods, especially in the mobile devices.

A. Tracking-by-Detection

With preprovided detection, MOT methods focus on data association algorithms, which are divided into online and batch, according to whether the information of subsequent frames is considered. Online methods [17], [18], [19], [20], [21], [22], [23] meet the needs of real-time processing without the subsequent information, but sacrifice the trajectory integrity. Most of these methods [17], [18], [19], [20], [21], [22] focus on the improvement of detector through spatiotemporal affinity, but relies on

redetection in tracking. Stadler and Beyerer [23] solved the occlusion problem through heuristic trajectory management. Zhang et al. [24] proposed a multiplex labeling graph for near-online tracking in intelligent devices. The batch methods utilize global information to achieve higher tracking accuracy at the expense of speed. Methods proposed in [25] and [26] solve MOT by lifted disjoint paths model which is conducive to global optimization. Graph network is naturally suitable for modeling MOT problems. With the development of the graph neural network (GNN), some GNN-based methods [27], [28] are proposed recently for further association.

MHT is one of the earliest successful methods proposed in [29]. The main idea of MHT is to establish a hypothesis tree for all possible association nodes, then evaluate and solve the global hypothesis. In the case of dense objects in long video, this strategy has the disadvantage of large time and space costs. To overcome this defect, the hypothesis decision is transformed into maximum-weight-independent set (MWIS) [30], and Sheng et al. [7] proposed a category transfer model for further efficiency optimization. To apply the method to mobile devices, we incorporate pruning and gating strategies and use sliding window. The hypotheses are significantly reduced to make the algorithm achieve near-online performance.

To improve the accuracy of tracking, some methods use tracklets as association units instead of detections. In [5], [6], [7], [8], and [31], the tracklets are generated from detection for association. By reducing detection errors and improving the reliability of association, tracking-by-tracklet has achieved success. Therefore, we use tracklet-level MHT (TLMHT) proposed in [7] as the baseline to implement our method.

B. Tracking in Mobile Devices

Tracking multiple objects in mobile devices is a complex problem involving many vision techniques. Solutions can be divided into three categories according to the vision sensor. First, the vision matching is established by the multiple cameras with preacquired location or calibration. In [32], [33], [34], [35], and [36], pairing of stereo camera is used to track objects. Stereo cameras provide additional information to estimate camera motion and restore the world coordinates of the objects. Second, 3-D reconstruction can be obtained to detect and track objects if the depths are provided. In [37], [38], [39], and [40], RGB-D data or laser points provide the basis for segmentation and detection of the object, and then the global motion of the object can be estimated by depth change. These methods depend on depth sensor, and need much computation in complex scenes with variety motion. Third, in the case of monocular visual, precalibration with assumptions, GPS, or odometry are used to estimate camera trajectory. Ess et al. [41] established a basic paradigm to track multiple objects in mobile platform. Wojek et al. [42] focused on 3-D scene understanding for traffic scenes. Then, Choi et al. [43] proposed a general framework by integrating the tracking method in mobile devices. However, calibration is still required and the camera speed is assumed to be constant. Some methods [44], [45] are designed with lower computation expenses for tracking on UAVs. Gao et al. [16] used odometry in mobile phone to track vehicle in GPS blocked environments. In cope with dense scenes, these methods rely on additional sensor information and lack of adaptability for complex motion interaction. On the contrary, we propose a hybrid motion model with minimum requirement and assumptions to achieve competitive results in the latest benchmarks.

C. Motion Estimation

In terms of visual odometry, many successful methods are proposed in the SLAM field. The method proposed in [15] is based on the RDB-D sensor, and the other [14] calibrates the monocular vision system precisely. Without additional information, our method is initialized with the detection information and suitable for MOT application. For motion segmentation, Liu et al. [46] and Shen et al. [47] used optical flow to acquire point trajectories for segmentation, which inspired us to make background segmentation under mobile devices.

The motion model based on the filter has achieved success by considering multiple errors. The Kalman filter is proposed early in [48] and provides a basic framework. However, a single-motion model is not adapted to deal with unpredictable motion changes. To model multiple modes, Bar-Shalom [49] used multiple Kalman Filters with different transition matrices. Genovesio et al. [50] established a probability model to measure the switching between modes. Recently, LSTM-based motion estimation [8] is proposed with neural network, while rely on specialized training step. Inspired by [50], we simplify the model representation and give the STEM for MOT problem.

III. Hybrid Motion Model for Mobile Devices

In this section, we integrate the camera motion of the mobile device, the camera-object motion projection, and the different motion modes of objects into hybrid motion model.

A. Model Overview

In order to detect and track multiple objects in mobile devices, the main idea of the hybrid motion model is to use world coordinates, which are not affected by camera motion. The world coordinates of an object can be projected from detections along with the camera trajectory. In this way, camera motion is required as an indirect quantity for coordinates mapping. As shown in Fig. 2, the model is mainly composed of three parts as follows.

Camera Motion: Different from methods in [41], [42], [43], and [51], our model does not require strict assumptions of precalibration or GPS data. Our method only needs to segment the background according to the optical flow and detections. To enhance the robustness and reduce error, the camera trajectory is obtained by evaluating and maximizing the motion probability. As iteration frame by frame, the motion state is updated to get the camera trajectory.

Camera-Object Motion: With camera trajectory, each object is mapped from image to world coordinate by smooth dynamic projection. Since both the objects and the camera are moving, the mapping method based on point matching is not feasible. To adapt to the dynamic change of object and camera, the

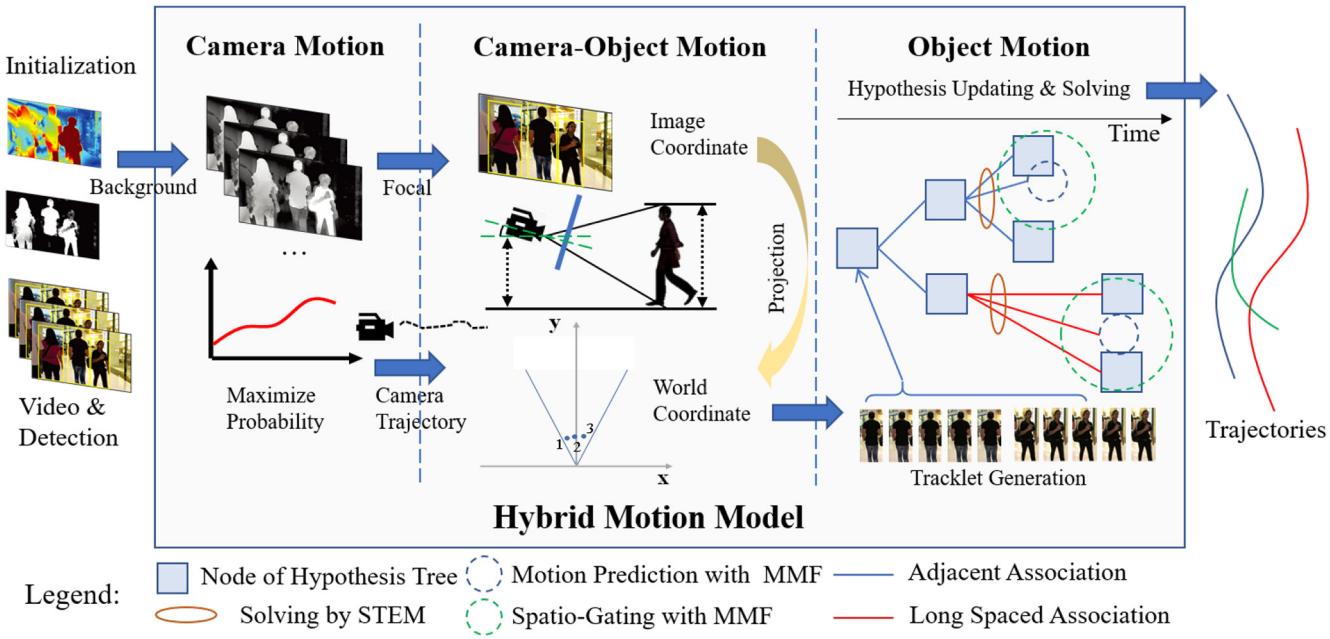

Fig. 2. Structure of Hybrid Motion Model. First, for camera motion, with segmented background, the camera trajectory is obtained by maximizing probability of motion hypotheses (Section III-B). Second, for camera-object motion, the objects are projected to the world coordinate by smooth dynamic projection in Section III-C. Third, for object motion, with tracklets generated by in Section IV-A, the motion of adjacent and long spaced tracklets are modeled by MMF (Section III-D) and candidate hypotheses are solved with STEM (Section IV-B).

TABLE I
NOTATION

Notation	Example	Meaning
Italic	a, b, A, B	Scalar
Lowercase bold	$\mathbf{a}, \mathbf{b}$	Vector
Capitalcase bold	$\mathbf{A}, \mathbf{B}$	Matrix
Blackboard	$\mathbb{A}, \mathbb{B}$	Set
Calligraphic	$\mathcal{A}, \mathcal{B}$	Function/Distribution
θ_i	θ_1, θ_2	Threshold/Weight
Number sequence	$1:n$	$1, 2, \cdots, n$

projection is established by taking the horizon as reference. In the projection, the heights of the camera and object are smoothed by information in previous frames. Besides, only the focal length is required to obtain the position of each object in the world coordinate.

Object Motion: In motion modeling, after tracklet generation, it is inconsistent to measure and predict motion state by same parameters due to missing detection and long intervals between tracklets. To solve this problem, MMF is proposed for adjacent and long spaced associations. With motion prediction and spatiogating by MMF, we propose STEM to evaluate hypotheses in the MHT framework.

To clarify the meaning of the equation in this article, styles of notation are used as summarized in Table I.

B. Camera Motion Estimation

In the first step of the hybrid motion model, camera motion estimation provides camera trajectory for world coordinate mapping. The basic geometric matching is initialized to obtain reliable feature points by finding the static background in adjacent frames.

Given set $\mathbb{P}_{t-1}$ and $\mathbb{P}_t$ be the feature points [52] between frame $t-1$ to t. Then, the initial Translation ($\mathbf{T}_t$) and Rotation ($\mathbf{R}_t$) of camera at frame t is obtained by eight-point algorithm. In order to eliminate the noise caused by mismatching, RANSAC [53] is used in iterations to minimize the cost function

$$\operatorname{argmin}\left(\frac{1}{|\mathbb{P}_t|}\sum_{i=1}^{|\mathbb{P}_t|}\mathcal{D}_{sp}\left(\mathbb{P}_{t-1}^i, \mathbb{P}_t^i; \mathbf{F}_t^*\right)\right) \tag{1}$$

where $\mathbf{F}_t^*$ is the estimated fundamental matrix from randomly selected pairs from $\mathbb{P}_t$ in each iteration and $\mathcal{D}_{sp}$ compute Sampson distance of each pair $(\mathbb{P}_{t-1}^i, \mathbb{P}_t^i)$.

The continuous pairing process often brings accumulation and estimation errors. To reduce these errors, the video sequence is divided by the sliding window, where camera motion hypotheses are proposed with different samples of frames, pairing points and parameters. To obtain the optimal trajectory, camera motion hypothesis in each video segment is measured by the probability function

$$\mathcal{P}_t^i = \underbrace{P(F_t|\mathbb{C}_t)}_{a}\underbrace{P\left(H_t^i|H_t^{i+1}\right)}_{b}\underbrace{P(\mathbb{C}_t|\mathbb{C}_{t-1})}_{c} \tag{2}$$

where (2)(a) is the flow similarity F_t at time t of camera state set $\mathbb{C}_t$. It controls the flow change caused by camera moving. Equation (2)(b) is the hypothesis probability updated from H_t^i to H_t^{i+1} at time t, modeling the association between camera motion hypothesis. Equation (2)(c) represents the transition which controls the smoothness of camera movements in state set $\mathbb{C}_t$ between time $t-1$ to t. Through the iterative process, the probability of motion hypothesis is maximized, which represent the optimal motion estimation.

1) Initialization for Estimation: Before hypothesis evaluation, it needs to initialize the background segmentation and estimate the focal length. Static area in background is overlapped in video fragments to meet the point matching requirement. In MOT applications, images are generally full of various objects with complex interaction. Considering detections describe the position of objects, a cluster model to segment background is trained by the pixel sample and optical flow.

In a pixel sample, the real-time superpixel segmentation [54] is used to form the training set $\mathbb{T}_v$ with label l

$$\mathbb{T}_v = \{(C_i, l)|i = 1{:}n\}, l = \begin{cases} 0, & \text{otherwise} \\ 1, & \text{if} \quad l \in \mathbb{D}_{\text{Hconf}} \end{cases} \tag{3}$$

where C_i represents the color feature of superpixel i from the video and $\mathbb{D}_{\text{Hconf}}$ represents the detection set of each video with confidence higher than threshold θ_1 from raw detections. Then, we train a Linear SVM with $\mathbb{T}_v$ to provide cluster score S_{clu} for all superpixels.

Second, optical flow [55] is used as local motion information. In the static background, the optical flow field is nearly a smooth surface in velocity space. However, object motion is inconsistent with the static background. To measure the background probability, the distance score S_{dis} is obtained by distance between object optical flow and background optical flow (S_{dis} is mapped from 0 to 1).

Based on the fusion of S_{clu} and S_{dis}, the measurement function $\mathcal{M}$ is formulated with weight parameter θ_2 to extract the static surface from the background in each video

$$\mathcal{M}(S_{\text{clu}}, S_{\text{dis}}, \theta_2) = \theta_2 \cdot S_{\text{clu}} + (1 - \theta_2) \cdot S_{\text{dis}}. \tag{4}$$

In camera motion estimation and objects projection, the focal length is an essential parameter. The matching $x_{t-1} \sim \mathbf{M}x_t$ between point pairs (x_{t-1}, x_t) is used to estimate the focal length. Camera position change between adjacent frames is approximated to rotating around the center of the background without translation. Also, the radial distortion is ignored in practice, therefore the mapping is expressed as

$$\mathbf{M} = \mathbf{K}\mathbf{R}_{t-1\Rightarrow t}\mathbf{K}^{-1} \tag{5}$$

where $\mathbf{K}$ is the internal parameter matrix of camera and $\mathbf{R}_{t-1\Rightarrow t}$ represents rotation matrix between point pairs (x_{t-1}, x_t). For simplicity of notation, the pixel center is defined as the image center ($c_x = c_y = 0$). Thus, (5) is expanded

$$\begin{bmatrix} m_0 & m_1 & m_2 \\ m_3 & m_4 & m_5 \\ m_6 & m_7 & m_8 \end{bmatrix} \sim \begin{bmatrix} r_{00} & r_{01} & r_{02}f \\ r_{10} & r_{11} & r_{12}f \\ r_{20}/f & r_{21}/f & r_{22} \end{bmatrix} \tag{6}$$

where $\mathbf{R}_{t-1\Rightarrow t} = [r_{ij}]$. The first two rows (or columns) of $\mathbf{R}_{t-1\Rightarrow t}$ must have the same norm and be orthogonal (even if the matrix is scaled). From this, the focal length can be obtained by solving the equation

$$f^2 = \frac{m_5^2 - m_2^2}{m_0^2 + m_1^2 - m_3^2 - m_4^2}, \quad \text{if} \quad m_0^2 + m_1^2 \neq m_3^2 + m_4^2 \tag{7}$$

or

$$f^2 = -\frac{m_5 m_2}{m_0 m_3 + m_1 m_4}, \quad \text{if} \quad m_0 m_3 \neq -m_1 m_4. \tag{8}$$

2) Motion Hypothesis Measurement and Updating: To evaluate the motion hypothesis of the camera, flow similarity is measured by camera movement from frame $t-1$ to t and is computed in terms of optical flow ϕ. The predicted optical flow $\phi_{\mathbb{C}_t}$ at frame t is obtained by the camera translation matrix $\mathbf{T}_t$ and the current camera state $\mathbb{C}_t$ (including the focal length and other parameters). The equation of flow similarity $P(F_t|\mathbb{C}_t)$ is formulated as follows:

$$P(F_t|\mathbb{C}_t) = \frac{1}{|\mathbb{P}^*|}\sum_{i=1}^{|\mathbb{P}^*|}\left(\frac{1}{\sqrt{\mathcal{D}(\phi, \phi_{\mathbb{C}_t})}} + 1\right) \tag{9}$$

where $\mathbb{P}^*$ is the selected set of pairing points and $\mathcal{D}$ calculates the difference of flow displacement.

In the probability function, transition smoothness measures the stability of camera motion from frame $t-1$ to t. Different from the assumption in [43] that the camera speed is constant in the whole video, our model allows speed changes dynamically. The speed of the camera has a limited mathematical expectation and the acceleration is relatively stable. Therefore, the transition motion of the camera is modeled as normal distribution to control smoothness

$$\begin{aligned} P(\mathbb{C}_t|\mathbb{C}_{t-1}) &= \mathcal{N}\left(\frac{(\mathbf{T}_{t-1} \times \mathbf{s}_t)(\mathbf{e}) + \Delta\mathbf{T}_t(\mathbf{e})}{\mathbf{e}}, 1\right) \\ \Delta\mathbf{T}_t &= (\mathbf{T}_{t-1} \times \mathbf{s}_t) - (\mathbf{T}_{t-2} \times \mathbf{s}_t) \end{aligned} \tag{10}$$

where $\mathbf{s}_t$ is the scale factor of the transition. The scale factor is computed by the displacement of matching points on the axis with the maximum camera movement. Here, $\mathbf{e}$ donates the axis of maximum movement and $\Delta\mathbf{T}_t$ represents the velocity of camera transition.

The parameters, pairing points, and frames in each window are selected to propose different hypotheses. In order to maximize the probability of the motion hypothesis, we implement the iteration to get the solutions. The hypothesis from H_t^i to H_t^{i+1} is accept with maximum $\mathcal{P}_t$.

In camera position update, the model tends to adopt conservative estimation which indicates that the position of the camera tends to be stable. Therefore, $P(H_t^i|H_t^{i+1})$ is assumed as a normal distribution $P(H_t^i|H_t^{i+1}) = \mathcal{N}(H_t^i, 1)$, where the hypothesis can be modeled using the current camera position H_t^i. The update strategy evaluates the probability according to (2). The motion hypothesis with higher flow similarity and translation stability is retained. With sliding window, the camera trajectory is estimated for world coordinate projection.

C. Smooth Dynamic Projection

In the second step of the hybrid motion model, the objects are mapped from image to world coordinate by smooth dynamic projection. In order to make the projection more robust and practical, we consider change of camera height h_c and rotation angle of pitch α. The motion of objects and camera is measured in the same plane (ground), basic geometric perspective with horizon line h is shown in Fig. 3.

In the initial frame, vertical projection of a camera on the ground is considered as the world coordinate original point $(0, 0, 0)$. The height of the camera off the ground plane is

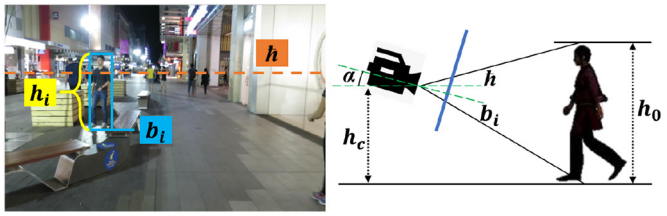

Fig. 3. Geometric perspective of camera and object in smooth dynamic projection. Notations are described in (11).

h_c. With minimal parameters and object world coordinate (X, Y, Z), smooth dynamic projection can be established as height relation (a) and coordinate mapping (b)

$$\begin{aligned} \text{(a)} \quad & h_c^t = \frac{1}{|\mathbb{D}^t|}\sum_{i=1}^{|\mathbb{D}^t|}\left(\frac{h_o}{h_i}\left(H_h^t - b_i\right)\right) \\ \text{(b)} \quad & Y = \frac{fh_c}{f\sin\alpha - (c_y - y)\cos\alpha} \\ & X = Y\frac{(x - c_x\cos\alpha)}{f}. \end{aligned} \tag{11}$$

As shown in Fig. 3, h_o represents the average height of object in real world. $\mathbb{D}^t$ is the detection set at frame t and H_h^t is the height of horizon in image at time t. For each object detected in the image, h_i and b_i are the height and the bottom of the bounding box, respectively.

Equation 11(b) represent the relationship between world (X, Y) and image coordinates (x, y) of the object. (c_x, c_y) is the center of the image plane. The camera rotation angle of pitch *alpha* is computed by

$$\text{alpha} = 2\arctan\left(\frac{c_y - H_h^t}{2f}\right). \tag{12}$$

1) Horizon Height Estimation: According to (11) and Fig. 3, in most scenarios, the horizon can be used to simplify calculations. We sample from the train set in experiments to estimate the horizon height H_{h1}^t, which follows a normal distribution with image center height $0.5 \times H_{\text{image}}$ as mean and scale factor $\mathbf{s}_t$ as variance

$$H_{h1}^t \sim \mathcal{N}\left(0.5 \cdot H_{\text{image}}, ||\mathbf{s}_t||\right). \tag{13}$$

By detecting vanishing points [56] at lines parallel to the horizon, the position of the horizon is obtained for projection. The set of vanishing points $\mathbb{V} = \{(x_v, y_v, c_v)_{1:v_n}\}$ with confidence c_v can be obtained. In order to reduce the interference caused by the lines not parallel to the ground, horizon height H_{h2}^2 is modeled by the set of vanishing points $\mathbb{V}$ as normal distribution

$$H_{h2}^t \sim \mathcal{N}\left(y_v^{\max(c_v)}, 1\right). \tag{14}$$

With (13) and (14), we combine horizon estimation and vanishing point to get the aggregated horizon height H_{h3}^t:

$$H_{h3}^t = \frac{||\mathbf{s}_t||H_{h1}^t + H_{h2}^t}{||\mathbf{s}_t|| + 1} \tag{15}$$

2) Update and Smooth: In each tracking window, the information in previous frames can be used to update and smooth the estimation in the current frame. For H_h^{t-1} of the previous frame and the current horizon height H_h^t

$$H_h^t = (1 - ||\mathbf{s}_t||)H_h^{t-1} + ||\mathbf{s}_t||H_{h3}^t. \tag{16}$$

The camera height in previous frame h_c^{t-1} is used to get smoothed camera height h_{c*}^t

$$h_{c*}^t = (1 - ||\mathbf{s}_t||)h_c^{t-1} + ||\mathbf{s}_t||h_c^t \tag{17}$$

where h_c^t is computed by (11)(a) and (16). Both the camera and object height are smoothed with the scale factor and the world coordinates of each object are obtained according to the projection (11).

D. Multimode Motion Filter

In the third step of the hybrid motion model, the object motion is modeled by MMF with different motion parameters. The motion state vector of the object is expressed as $\mathbf{x}_t = (x_t, y_t, z_t, v_{xt}, v_{yt}, v_{zt})^T$ at frame t with coordinate (x_t, y_t, z_t) and speed (v_{xt}, v_{yt}, v_{zt}) in 3-D. In practice, it is considered that all objects move in the same plane $(z = 0)$, so the following descriptions are in 2-D form to simplify the equation.

For the MOT problem, tracklets are often break with fragments due to missing detection and occlusion. When the interval between tracklets is short, the speed of the object is not change dramatically, so the prediction of the motion model can achieve high accuracy. However, after a long interval, the same motion model is often unable to predict the location of the object due to the uncertainty movement. Therefore, we divide the motion of the object into two modes: 1) continuous and 2) discontinuous. In the continuous mode, the tracklets are connected with adjacent association and the motion state is relatively stable. In the discontinuous mode, the interval between tracklets obviously affect the state of movement.

In order to model two different motion modes and adapt to the disturbance of the mobile devices. State prediction and measurement of object motion based on multiple Kalman filters are defined as follows:

$$\mathbf{x}_{t+1} = \mathbf{F}^{m_i}\mathbf{x}_t + \mathbf{w}_{t+1}^{m_i} \tag{18}$$

$$\mathbf{z}_{t+1} = \mathbf{H}^{m_i}\mathbf{x}_{t+1} + \mathbf{v}_{t+1}^{m_i} \tag{19}$$

where $m_i \in \mathbb{M}$ with $i \in \{0 : n\}$ represent successive mode with frame set $\mathbb{T} \in \{T_1 : T_n\}$ identifying the beginnings and ends of the modes. The matrix $\mathbf{H}$ is matrix of observation $\mathbf{z}_t$. Vectors $\mathbf{w}_t$ and $\mathbf{v}_t$ represent the model noise and the measurement noise, respectively. The transition matrix $\mathbf{F}$ is associated to the mode m_i, which defines the motion mode by the filter. Specifically, the transition matrices of continuous mode $(\mathbf{F}_1)$ and discontinuous mode $(\mathbf{F}_2)$ are defined as

$$\mathbf{F}_1 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \mathbf{F}_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{20}$$

where $\mathbf{F}_1$ models uniform linear motion and $\mathbf{F}_2$ remain the transmission of the speed vector $\mathbf{v}(v_{xt}, v_{yt})$ in the estimation of the state parameter. For a discontinuous motion filter, $\mathbf{v}$ is ignored if the estimated gain is low.

With (18) and (19), each possible mode sequence for $\mathbb{M}$ is considered to filter the state parameter optimally. To reduce the cost of parameterization calculation, the possible mode at frame t without approximation is described as

$$P(\mathbf{x}_{t+1}, \mathbf{z}_{1:t+1}) = P\big(\mathbf{x}_{t+1}, \mathbf{z}_{T_i:t+1}\big) \propto \sum_{m_i \in \mathbb{M}} P\big(\mathbf{x}_{t+1}|\mathbf{z}_{T_i:t+1}, m_i\big) P\big(m_i|\mathbf{z}_{T_i:t+1}\big) \tag{21}$$

where $P(\mathbf{x}_{t+1}|\mathbf{z}_{T_i:t+1}, m_i)$ is assumed to follow a normal distribution $\mathcal{N}(\hat{\mathbf{x}}_{t+1}^{m_i}, \hat{\mathbf{P}}_{t+1}^{m_i})$, where $\mathbf{x}_{t+1}^{m_i}$ is mode conditional mean and $\mathbf{P}_{t+1}^{m_i}$ is the covariance matrix. T_i represents the start time of the current motion mode. The conditional probability according to the multiple motion filter is given as follows:

$$\begin{aligned} &P\big(\mathbf{x}_{t+1}|\mathbf{z}_{T_i:t+1}, m_i\big) \\ &= \frac{P(\mathbf{z}_{t+1}|\mathbf{x}_{t+1}, m_i)}{P\big(\mathbf{z}_{t+1}|\mathbf{z}_{T_i:t}, m_i\big)} P\big(\mathbf{x}_{t+1}|\mathbf{z}_{T_i:t}, m_i\big) \\ &= \frac{P(\mathbf{z}_{t+1}|\mathbf{x}_{t+1}, m_i)}{P\big(\mathbf{z}_{t+1}|\mathbf{z}_{T_i:t}, m_i\big)} \int P(\mathbf{x}_{t+1}|\mathbf{x}_t, m_i) P\big(\mathbf{x}_t|\mathbf{z}_{T_i:t}, m_i\big) d\mathbf{x}_t. \end{aligned} \tag{22}$$

According to the consistency of observations and predictions, the uniform assumption of motion mode is given

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^{m^*} \quad \text{and} \quad \hat{\mathbf{P}}_t = \hat{\mathbf{P}}_t^{m^*} \tag{23}$$

where m^* is the motion mode with the most probability.

The parameters of different motion modes are mixed to predict and update the object motion, and the association cycle of multi Kalman filter is given as

$$\begin{aligned} \text{predict}: \bar{\mathbf{x}}_{t+1}^{m_i} &= \mathbf{F}^{m_i} \hat{\mathbf{x}}_t^{m_i} \\ \bar{\mathbf{P}}_{t+1}^{m_i} &= \mathbf{F}^{m_i} \hat{\mathbf{P}}_t^{m_i} \mathbf{F}^{m_i \top} + \mathbf{Q}_t^{m_i} \end{aligned} \tag{24}$$

where $\mathbf{Q}_t^{m_i}$ and $\mathbf{R}_t$ are the white noise covariance matrices of model noise $\mathbf{w}$ and the measurement noise $\mathbf{v}$.

$$\begin{aligned} \text{update}: \mathbf{K}_{t+1}^{m_i} &= \frac{\bar{\mathbf{P}}_{t+1}^{m_i} \mathbf{H}^{m_i \top}}{\mathbf{H}^{m_i} \bar{\mathbf{P}}_{t+1}^{m_i} \mathbf{H}^{m_i \top} + \mathbf{R}_{t+1}} \\ \hat{\mathbf{P}}_{t+1}^{m_i} &= \big(\mathbf{I} - \mathbf{K}_{t+1}^{m_i} \mathbf{H}^{m_i}\big) \bar{\mathbf{P}}_{t+1}^{m_i} \\ \hat{\mathbf{x}}_{t+1}^{m_i} &= \bar{\mathbf{x}}_{t+1}^{m_i} + \mathbf{K}_{t+1}^{m_i} \big(\mathbf{z}_{t+1} - \mathbf{H}^{m_i} \bar{\mathbf{x}}_{t+1}^{m_i}\big) \end{aligned} \tag{25}$$

where $\mathbf{K}_{t+1}^{m_i}$ is adaptive Kalman gain, and $\mathbf{I}$ represents the identity matrix measured innovation. Then, in (21), the mode probability $P(m_i|\mathbf{z}_{T_i:t+1})$ is derived as

$$P\big(m_i|\mathbf{z}_{T_i:t+1}\big) \propto P\big(\mathbf{z}_{t+1}|\mathbf{z}_{T_i:t}, m_i\big) P\big(m_i|\mathbf{z}_{T_i:t}\big) \tag{26}$$

where

$$P\big(\mathbf{z}_{t+1}|\mathbf{z}_{T_i:t}, m_i\big) \sim \mathcal{N}\Big(\mathbf{H}\bar{\mathbf{x}}_{t+1}^{m_i}, \mathbf{H}\bar{\mathbf{P}}_{t+1}^{m_i} \mathbf{H}^{\top} + \bar{\mathbf{R}}_t^{m_i}\Big). \tag{27}$$

In the process of object motion estimation, if the next measurement contradicts the uniform assumption (23), the filter mode switches. To detect mode switches in tracking, the transport between two modes is measured as

$$\text{argmax} P\big(\mathbf{z}_{t+1}|\mathbf{z}_{T_i:t}, m_i\big) \neq \text{argmax} P\big(m_i|\mathbf{z}_{T_i:t}\big). \tag{28}$$

According to (28), the model switches between the two modes is detect. Each filter is reinitialized with the previous measurement parameters. However, in the Discontinuous mode, the object motion state may experience an irregular change. Therefore, we use a smooth strategy fuses two independent filtering processes, which are forward and backward in temporal order to solve this problem. In this way, the parameter differences between the two modes are balanced.

IV. Multiple Hypothesis Tracking Based on Hybrid Motion Model

In this section, we implement a hybrid motion model in the MHT framework to solve the MOT problem. MHT establishes hypothesis trees for all possible tracklet associations, where motion of nodes is measured by the proposed model. Considering that confidence of tracklets has a great influence on tracking accuracy, world coordinates with height information is used to generate high confidence tracklets. To select and evaluate candidate tracklets for hypothesis updating and solving, STEM is proposed based on motion measurement in the model.

The tracking framework can be summarized into three steps.

1) *Tracklet Generation:* With world coordinates and height information provided by smooth dynamic projection, the false detections are filtered to generate high confidence tracklets.
2) *Hypothesis Updating:* From the initial tracklet, MHT maintain hypothesis trees that contains all possible associations. The probability of each branch in hypothesis tree is measured by STEM.
3) *Hypothesis Solving:* To avoid the exponential growth of hypotheses, the N-Scan pruning proposed in [13] is used to set a time window at size N. Then, the data association is formulated as MWIS to get objects trajectories. We solve MWIS by category transfer model proposed in [7].

A. Tracklet Generation by Smooth Dynamic Projection

The scale relationship between each object and the scene are estimated by mapping between image and world coordinates. This height information can be used to further filter error detection and generate more accurate tracklets.

Given the detection set $\mathbb{D}_t = \{d_{1:k}\}$ at frame t. FPs in $\mathbb{D}_t$ can be filtered before generating tracklets. Note that all objects are assumed to move on the ground, the bottom b_i of $d_i \in \mathbb{D}_t$ is considered below the horizon estimated in Section III-C. Therefore, the detection confidence c_{d_i} is modified by

$$c_{d_i} = \frac{h_i}{b_i + h_i - H_h^t} \times c_{d_i}, \quad \text{if } H_h^t \leq b_i \tag{29}$$

and the height h_i of object i should obey the normal distribution $h_i \sim \mathcal{N}(\hat{h}_i, 0.5)$ under perspective, where the expected mean height $\hat{h}_i$ is estimated by

$$\hat{h}_i = \frac{h_o\big(H_h^t - bi\big)}{h_c} \tag{30}$$

where detections with height probability less than threshold θ_3 are filtered.

In tracklets generation, the K-partite graph is modeled for detection set $\mathbb{D}_{k_1:k_2}$ from frame k_1 to k_2 in windows k. Different from TLMHT [7], we only use world coordinates to generate high-confidence tracklets for near-online tracking. In this way, the K-partite graph is defined as follows.

1) *Node Set* $\mathbb{N}$*:* In initialization, $\mathbb{N}$ only includes detections. In subsequent iterations, $\mathbb{N}$ contains the tracklets left by the previous matching window, which length is less than threshold θ_4.
2) *Edge Set* $\mathbb{E}$*:* Except the edge between the detection, also contain the edge between the tracklets and the detection.
3) *Weight Set* $\mathbb{W}$*:* Represent the similarity of nodes between edges, measured by appearance feature and motion state.

The Weight includes motion measurement using world coordinates (X, Y) can be expressed as $w_{\text{mot},ij} = ||(X_i, Y_i), (X_j, Y_j)||_2$. Each graph is associated with a score set and weight set. Scores serve as cost coefficients in the various discrete optimization formulations used to rank solutions. The solution of the graph can be solved by linear programming proposed in [8] to find the maximum sum of all weights. By solving the K-partite graph, tracklets are generated as the nodes of MHT. In addition, we propose an improvement to incorporate the remaining individual detection into the node to improve the recall of tracking method.

B. STEM for Hypothesis Updating and Solving

In hypothesis updating, MHT maintains possible tracklets in each window. Existing hypotheses are updated to link all candidate nodes of tracklets. The evaluation of the branch in hypothesis tree determines the effect of tracking method. Based on the hybrid motion model, the object motion change is used to calculate the weight score of edge.

The edge between adjacent tracklets is considered to represent continuous motion mode, and the edge between long spaced tracklets belongs to discontinuous motion mode. Here, MMF is used to measure the object motion and predict the mode switch. Due to the false and missing detection, if the tracklets are mismatched in spatial area, a dummy node is predicted according to the object motion model of discontinuous mode by (24).

In order to improve the recall and precision of association in mobile devices, we propose a STEM based on variance estimator. The count of prediction errors at time t is $\mathcal{C}(e)_t^{m_i}$ with current mode m_i. The parameters are estimated by

$$\begin{aligned}
\mathcal{C}(e)_{t+1}^{m_i} &= \mathcal{C}(e)_t^{m_i} + 1 \\
\bar{e}_t &= \mathbf{K}_t^{m_i}\left(\mathbf{z}_t - \mathbf{H}\bar{\mathbf{x}}_t^{m_i}\right) \\
d_{t+1}^{m_i} &= d_t^{m_i} + \frac{\bar{e}_{t+1} - d_t^{m_i}}{\mathcal{C}(e)_{t+1}^{m_i}} \\
\mathbf{M}_{t+1}^{m_i} &= \mathbf{M}_t^{m_i} + \left(\bar{e}_{t+1} - d_{t+1}^{m_i}\right)^{\top}\left(\bar{e}_{t+1} - d_{t+1}^{m_i}\right)
\end{aligned} \tag{31}$$

where $\bar{e}_t$ is average error at time t, $d_{t+1}^{m_i}$ represent mean deviation of $\bar{e}_t$ and $\mathbf{M}_{t+1}^{m_i}$ is square matrix of $d_{t+1}^{m_i}$. Thus, the covariance matrix is estimated as

$$\mathbf{C}_t = \frac{\mathbf{M}_t}{\mathcal{C}(e)_t}. \tag{32}$$

Following the iteration of (31) and (32), the basic spatial gating can be estimated by a normal distribution with current state and covariance matrix. When object motion switched to discontinuous mode, the optimal association always out of the estimated space range. To deal with this problem, the covariance matrix is reinitialized based on the mode switch. The parameters are reset to initial value $(\mathcal{C}(e)_0, d_0, \mathbf{M}_0)$ when detect that the object motion is switched to the discontinuous mode. To get a robust result in spatial gating prediction, a similar smoothing method as [57] is implemented in iteration.

In hypothesis solving, the logarithm-likelihood ratio is used to measure the motion similarity between hypotheses. The object location probability P is measured by normal distribution $\mathcal{N}(\bar{\mathbf{x}}_{t+1}^{m_i}, ||\mathbf{Q}_t^{m_i}||)$ with $\bar{\mathbf{x}}_{t+1}^{m_i}$ and $\mathbf{Q}_t^{m_i}$ predicted by (24) in Section III-D. For null hypothesis ϕ which indicates false association, the probability $P(L_t \in \phi) = 1/\hat{A}$, where $\hat{A}$ represents the estimated area of world coordinate. The motion score for hypothesis branch L at time t is defined as

$$\begin{aligned}
\mathcal{S}_{\text{mot}}^t &= \ln\left(\frac{P(L_{1:t} \in H)}{P(L_{1:t} \in \phi)}\right) \\
&= \ln\left(\frac{\prod_{1:t} P(L_t | L_{t-1} \in H)}{\prod_{1:t} P(L_t \in \phi)}\right)
\end{aligned} \tag{33}$$

where H means association nodes belongs to the same object.

Accordingly, the aggregated score $\mathcal{S}^*$ of hypothesis $H = l_{1:k}$ is defined as

$$\mathcal{S}^* = \left(\sum_{i=1}^{k} c_{d_i}; (1-\theta_5)\mathcal{S}_{\text{mot}}^k + \theta_5 \sum_{j=1}^{k-1} \mathcal{S}_{\text{app}}\left(l_j, l_{j+1}\right)\right) \tag{34}$$

where c_{d_i} is confidence of each detection in tracklet l and θ_5 is the weight for motion and appearance score computed by adaptive method proposed in [58]. The appearance features of tracklets are represented by linear mean of detections feature extracted by the real-time Re-Id network [59]. The score $\mathcal{S}_{\text{app}}$ is cosine distance of tracklet feature: $w_{\text{app},ij} = \cos(\text{app}_i, \text{app}_j)$. By measuring similarity scores, the associations are solved as baseline [7] to get trajectories for all objects.

V. Experiments

In this section, we first introduce data sets and evaluation metrics. Then, parameters are shown with visual analysis. To verify the effects of our method, we performed component analysis and compared it with various methods. Through ablation experiments, each component is evaluated in quantization. For general results, our method is compared with state-of-the-art in MOT and KITTI data sets.

A. Data Sets and Metrics

We evaluate the performance of our method on MOT15 [60], MOT17 [61] and KITTI [62] data sets, which are widely used to evaluate MOT performance based on the tracking-by-detection paradigm. The MOT15 data set consists of 22 video sequences divided into 11 training sets and 11 testing sets and the MOT17 data set consists of 14 video sequences divided into seven training sets and seven testing sets. In MOT15 and MOT17, videos are filmed with different lighting conditions, shooting angles, and density in both static camera and mobile devices

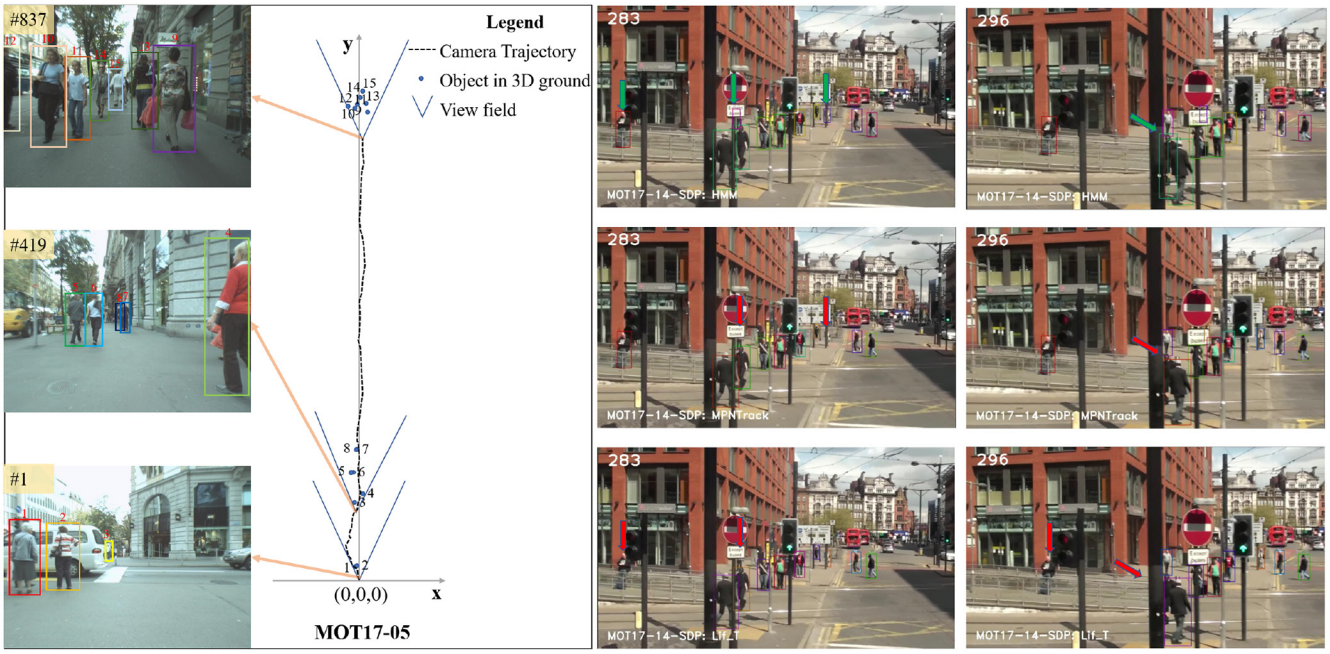

Fig. 4. Visual results of camera trajectory in MOT17-05 sequence and visual comparison between MPNTrack [27], Lif_T [25], and the proposed method in MOT17-14 sequence. The red arrow indicates the lost objects and the green arrow indicates the tracked objects.

provided with public detection, the evaluation focused on the performance of tracking algorithms. Besides, in MOT17, detections of each video are obtained by three detectors proposed in [2], [3], and [4] to balance the impact of different detectors. In the KITTI data set, videos are captured by a vehicle-mounted camera without public detections, and provides camera calibration, stereo view, laser points, and GPS data, which supports a variety of 3-D tracking methods.

The evaluation metrics include CLEAR MOT metrics [63], identity switches (ID Sw.) [64], IDF1 score [65], and higher order tracking accuracy (HOTA) [66]. The MOT accuracy (MOTA) shows the comprehensive MOT performance by combining three error sources: 1) ID Sw.; 2) FPs; and 3) missed objects (FN). The IDF1 is ratio of correctly identified detections over the average number of ground-truth and computed detections. The HOTA is geometric mean of detection accuracy and association accuracy. Averaged across localization thresholds. MT and ML are ratio of the mostly tracked (80% tracked) and the mostly lost (80% lost) objects. Frag is the total number of times a trajectory is fragmented.

B. Parameters and Visual Analysis

In this section, we analyze the parameters of the method. In addition, selected tracking visualization results are shown in Fig. 4.

Parameters: Threshold θ_1 is used to select detections for background segmentation in Section III. By evaluating the detection quality in the train set, θ_1 is set to 0.9 considering that most correct detections have the confidence higher than this threshold. In Sections III-C and IV-A, the average height of the object in real world h_0 is fixed to 1.7 m, which represents the mean height of pedestrians.

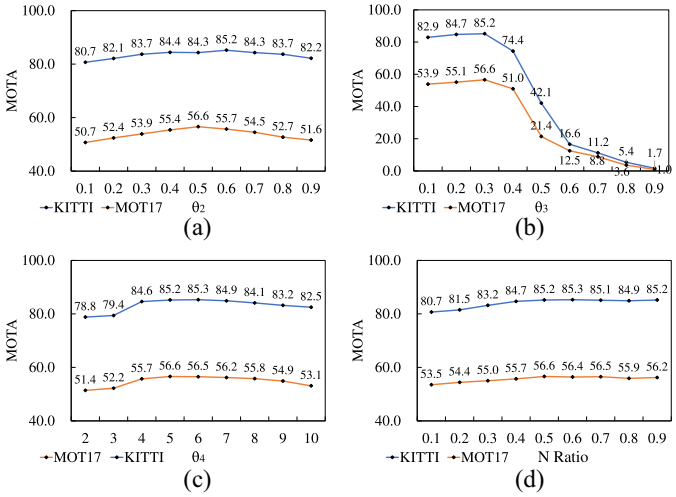

Fig. 5. Impact of each parameter on MOTA of our method in MOT17 and KITTI data sets. (a) Segmentation weight. (b) Height threshold. (c) Tracklet length. (d) N Ratio of framerate.

As shown in Fig. 5, we evaluated the effects of θ_2, θ_3, θ_4, and *NRatio* on MOTA in MOT17 and KITTI data sets. Because the parameters are relatively independent, when evaluating one parameter, other parameters are fixed as the optimal setting. Weight parameter θ_2 for background segmentation is set to 0.55 to balance the best MOTA performance on MOT17 and KITTI data sets. When θ_3 is set to 0.3, i.e., the detection with height estimation range (1.45, 1.95) is retained, MOTA of each data set achieves the highest value of 56.6% and 85.2%. With the increase of θ_3, more reliable detections are filtered out, so the value of MOTA drops rapidly. In Section IV-A, to balance the tracklet length and computational efficiency, the maximum length of tracklet θ_4 is set to 5 as used in the

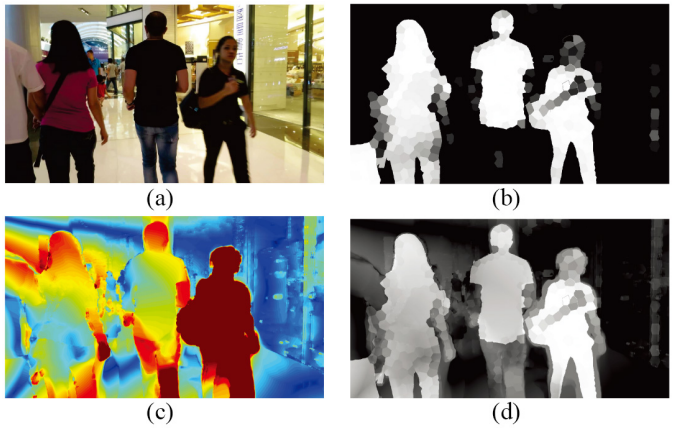

Fig. 6. Background segmentation for MOT17-12 sequence at frame 511. (a) Original image. (b) Segment (cluster). (c) Segment (optical flow). (d) Segment (fusion).

baseline method [7]. N determines the compute window in hypothesis solving, which is related to the rate of the object state change with time. Thus, we measure N according to the percentage *NRatio* of framerate. When *NRatio* is set to 50%–70%, MOTA of each data set achieves the highest value. Due to the complexity of algorithm, the memory cost and computing time increase exponentially with the increase of N. So, we set *NRatio* to 50% to get the balance between efficiency and space-time cost.

In summary, to get the best result, we use the parameters setting in the following experiments:

$$\theta_1 = 0.9, \theta_2 = 0.55, \theta_3 = 0.3, \theta_4 = 5$$
$$h_0 = 1.7, N \text{ Ratio} = 0.5.$$

Visual Qualitative Analysis: In terms of background segmentation, qualitative results are shown in Fig. 6. Darker areas in Fig. 6(b) have higher background scores by cluster measurement. Using the color features of the detection as the training set, the cluster method is more sensitive to the color. However, it fails when the object color is close to the cluster center. As the result shown in Fig. 6(b), the pants of the man in the middle is not distinguished. In Fig. 6(c), the distance of pixel velocity to the optical flow is shown in different colors. The pixels with higher distance are painted red (the person on the right side), and the pixels with lower distance are painted blue. Distance measurement based on optical flow is more sensitive to motion difference, while it fails when the objects move synchronously with the background. By combining both cluster and distance scores, more accurate fusion result is shown in Fig. 6(d).

Fig. 4 shows the visual results of the estimated camera trajectory of MOT17-05 sequence and selected frames in MOT17-14 sequence for visual comparison. As shown in MOT17-05 sequence, our method accurately restores the position of the objects to the world coordinate. As shown by the arrows in Fig. 4, compared with the MPNTrack [27] and Lif_T [25], our method can track more objects and keep continuous tracking in case of occlusion.

TABLE II
COMPARISON OF TRACKLETS GENERATION ON THE MOT17 TRAIN SET

Method	#Tracklet	w/o ID Sw.	MOTA	IDF1	HOTA
Dai. et al. [5]	45,620	92.5%	47.4	58.9	45.4
TLMHT [7]	37,700	98.9%	52.6	62.7	49.3
TT17 [8]	41,585	**99.2%**	56.5	67.0	53.7
$\text{Tracklet}_{\mathbf{s}}$	39,260	**99.2%**	**56.6**	**67.2**	**54.0**

TABLE III
VERIFICATION OF TRACKLET IN THE MOT17 TRAIN SET

Methods	MOTA	IDF1	HOTA	ID Sw.	FP	FN
Deep [19]	44.9	51.9	38.4	1,949	6,886	168,277
$\text{Deep}_{\mathbf{T}}$ [19]	**48.7**	**55.1**	**41.1**	**1,749**	**6,310**	**165,166**
Center [20]	67.4	63.0	49.0	1,822	8,066	**94,572**
$\text{Center}_{\mathbf{T}}$ [20]	**68.6**	**64.1**	**51.0**	**1,605**	**7,540**	95,012
Fair [21]	83.8	81.9	68.5	1,596	8,208	**44,625**
$\text{Fair}_{\mathbf{T}}$ [21]	**84.2**	**83.4**	**69.5**	**1,431**	**7,725**	44,901

TABLE IV
COMPARISON OF MOTION MODEL IN THE MOT17 TRAIN SET

Motion Model	MOTA	IDF1	HOTA	ID Sw.	FP	FN
Greedy [20]	49.7	58.4	45.1	1,245	10,452	157,083
K+IOU [21]	52.9	63.1	49.6	632	9,768	148,249
K+G [7]	53.2	63.9	50.2	559	10,814	146,263
LSTM+G [8]	55.0	66.2	52.1	864	11,871	138,736
MMF+STEM	**56.6**	**67.2**	**53.3**	**565**	**8,129**	**137,580**

TABLE V
COMPARISON OF MOTION MODEL IN THE KITTI TRAIN SET

Motion Model	MOTA	IDF1	HOTA	ID Sw.	FP	FN
Greedy [20]	77.5	66.8	53.0	218	759	1,637
K+IOU [21]	82.7	72.1	58.4	156	541	1,401
K+G [7]	82.8	74.9	61.1	133	537	1,382
LSTM+G [8]	83.1	75.2	61.7	142	566	1,244
MMF+STEM	**85.2**	**78.0**	**64.0**	**111**	**412**	**1,125**

C. Effect Analysis for Tracklet Generation and Motion Model

In this section, we analyze the effect of tracklet generation and motion model with MMF and STEM.

As shown in Table II, $\text{Tracklet}_{\mathbf{s}}$ represents the tracklet generation method based on smooth dynamic projection. We use different methods [5], [7], [8] to generate tracklets as input. Through the height information provided by the projection, the error detections are filtered and the confidences of detections are corrected. Therefore, the recall and the ID consistency of tracklers are both improved for the association. By evaluation on the MOT17 train set, our method achieves the highest MOTA, IDF1 and HOTA. To further verify the quality of tracklets in our method, we combine our tracklets into three open source trackers [19], [20], [21], which are widely used for online MOT. As shown in Table III, by combining $\text{Tracklet}_{\mathbf{s}}$, these methods are all improved in main metrics.

As shown in Tables IV and V, for the motion model, we compare our proposed method MMF and STEM with four mainstream motion models [7], [8], [20], [21] on MOT17 and KITTI data sets. Greedy represents distance-based greedy matching, K represents Kalman filter, IOU represents measurement based on IOU value, G represents threshold-based gating, and LSTM represents motion prediction network based on LSTM. By providing prediction probability and error estimation for different motion modes, more tracklets are associated

TABLE VI
ABLATION STUDY ON DIFFERENT MOVING SEQUENCES IN MOT17 AND KITTI TRAIN SETS

Sequence	Method	MOTA ↑	IDF1 ↑	HOTA ↑	MT1 ↑	ML1 ↓	FP ↓	FN ↓	IDS ↓	Frag ↓
MOT17-05	$\text{Model}_{\mathbf{T}}$	43.0	54.0	41.0	34	210	685	11,096	**45**	**71**
Illumination: sunny	$\text{Model}_{\mathbf{M+S}}$	51.0	63.3	50.5	79	**163**	**626**	9,485	52	108
Density: 8.3	$\text{Model}_{\mathbf{T+M+S}}$	**51.1**	**63.7**	**50.7**	**81**	**163**	631	**9,465**	48	98
MOT17-10	$\text{Model}_{\mathbf{T}}$	49.0	58.8	46.7	42	61	1,990	17,581	**82**	**249**
Illumination: night	$\text{Model}_{\mathbf{M+S}}$	54.9	**62.4**	50.1	**58**	45	1,817	15,422	150	360
Density: 8.3	$\text{Model}_{\mathbf{T+M+S}}$	**55.1**	62.3	**49.5**	**58**	**42**	**1,734**	**15,410**	151	357
MOT17-11	$\text{Model}_{\mathbf{T}}$	61.2	70.1	55.3	65	86	1,194	9,708	88	**92**
Illumination: indoor	$\text{Model}_{\mathbf{M+S}}$	64.4	72.0	57.2	71	**72**	680	9,327	71	96
Density: 8.3	$\text{Model}_{\mathbf{T+M+S}}$	**64.9**	**72.3**	**58.2**	**73**	**72**	**599**	**9,270**	**56**	95
MOT17-13	$\text{Model}_{\mathbf{T}}$	39.3	52.4	40.7	76	154	1,600	19,525	**80**	**169**
Illumination: sunny	$\text{Model}_{\mathbf{M+S}}$	**42.0**	**55.0**	**43.8**	**90**	**142**	**1,511**	**18,614**	117	222
Density: 8.3	$\text{Model}_{\mathbf{T+M+S}}$	41.7	54.8	43.2	88	**142**	1,596	18,665	110	237
KITTI-13	$\text{Model}_{\mathbf{T}}$	83.8	90.3	75.0	36	1	85	47	14	13
Illumination: sunny	$\text{Model}_{\mathbf{M+S}}$	90.9	94.6	79.1	**37**	**0**	**54**	**26**	2	**3**
Density: 2.2	$\text{Model}_{\mathbf{T+M+S}}$	**91.0**	**95.4**	**79.8**	**37**	**0**	**54**	**26**	**1**	**3**
KITTI-19	$\text{Model}_{\mathbf{T}}$	85.5	70.2	55.8	51	**1**	251	508	92	112
Illumination: shadowy	$\text{Model}_{\mathbf{M+S}}$	87.5	72.8	58.3	**54**	**1**	179	482	73	**120**
Density: 5.0	$\text{Model}_{\mathbf{T+M+S}}$	**87.8**	**74.4**	**59.7**	**54**	**1**	**176**	**479**	**59**	**120**

TABLE VII
COMPARISON ON MOVING SEQUENCES IN THE MOT15 TEST SET

Tracker	MOTA ↑	IDF1 ↑	ID Sw. ↓	FP↓	FN↓
CRFTrack [12]	35.7	54.5	168	5,474	12,683
KCF [67]	35.8	47.9	196	3,721	14,911
Lif_TsimInt [25]	41.2	59.2	195	5,170	11,971
ApLift [26]	44.7	62.4	164	6,592	8,353
mfi_tst [22]	46.4	58.6	197	4,923	9,788
MPNTrack [27]	47.5	62.5	110	4,749	10,099
Tracktor++v2 [18]	47.8	58.0	197	**2,972**	11,484
Ours	**59.7**	**71.2**	**126**	5,609	**5,718**

TABLE VIII
COMPARISON ON MOVING SEQUENCES IN THE MOT17 TEST SET

Tracker	MOTA ↑	IDF1 ↑	ID Sw. ↓	FP↓	FN↓
CTTrackPub [20]	43.0	42.2	1,689	7,738	86,083
MPNTrack [27]	45.3	53.0	670	9,322	81,625
LPC_MOT [28]	45.5	53.1	742	7,499	83,008
ApLift [26]	45.5	55.0	1,147	15,813	74,316
Lif_T [25]	45.7	55.8	**631**	**3,238**	87,034
mfi_tst [22]	48.4	52.0	1,147	6,105	79,155
TMOH [23]	48.9	54.1	1,144	4,853	79,527
Ours	**52.3**	**58.7**	869	14,075	**64,883**

with lower FP, FN, and ID Sw., which make our method achieve better results on MOT17 and KITTI data sets.

D. Ablation Study for Hybrid Motion Model

We evaluated components of our method on moving sequences in MOT17 and KITTI train sets and the results are shown in Table VI. $\text{Model}_{\mathbf{T}}$ represents only the generated tracklets are used for tracking. It is noticed that ID Sw. and Frag of trajectories are significantly reduced with tracklets generated by our method. Only use STEM in $\text{Model}_{\mathbf{M+S}}$, FP and FN of each sequence are reduced, which shows that STEM is effective in tracklets evaluation, rather than achieving the balance between FP and FN by parameter adjustment. In $\text{Model}_{\mathbf{M+S}}$, the constraints for tracklets association are stricter, but ID Sw. and frag of the trajectory do not increase significantly. To achieve the best tracking result, $\text{Model}_{\mathbf{T+sM+S}}$ gives a more accurate evaluation so that the integrity of the same object trajectory is higher, and FN is reduced. At the same time, the trajectories of different objects are correctly associated, thus reducing FP. However, compared with other sequences, the camera trajectory changes greatly in MOT17-13 sequence, which leads to a slightly lower improvement in tracking results. The total results show that our method can effectively improve the accuracy of multiobject tracking in the video of the handheld mobile camera and vehicle camera. In addition, the method has good robustness for scenes with different densities, different weather, and indoor and outdoor environment.

E. Comparison on Benchmark

To evaluate the overall performance, our method is compared with published state-of-the-art methods on MOT15, MOT17, and KITTI benchmarks. For a fair comparison, methods using public detection are compared in the MOT benchmark and we use the same private detector as [27] to get a result in the KITTI benchmark.

In moving sequences, as shown in Tables VII and VIII, our method achieves significant improvement in MOTA, IDF1, and HOTA among all the methods in MOT15 and MOT17 test sets. By generating tracklets with high confidence and motion modeling with MMF, hybrid motion model significantly improves the ID consistency within and between tracklets. Furthermore, STEM maintains higher trajectory integrity without introducing more false associations. Therefore, FN decreased while FP maintained limited growth, and finally more accurate trajectories are obtained.

In the overall benchmark result shown in Tables IX–XI, our method performs with the highest MOTA. By giving more accurate measurement and prediction for object motion, our method achieves high trajectory integrity (MT) for video sequence. Moreover, our tracker also effective for static camera. Without camera motion estimation, the model can directly use the image coordinate of the object for MMF and use STEM for tracklet association. In Table XI, in particular, we choose the tracker using three types of additional sensor data [35], [40], [51]. Compared with these methods, our

TABLE IX
COMPARISON ON THE MOT15 BENCHMARK

Tracker	MOTA ↑	IDF1 ↑	HOTA ↑	MT(%) ↑	ML(%)↓	FP ↓	FN ↓	ID Sw. ↓	Frag ↓	LocA ↑	Hz ↑
KCF [67]	38.9	44.5	33.1	16.6	31.5	7,321	29,501	720	1,440	74.7	0.3
CRFTrack_ [12]	40.0	49.6	37.3	23.0	28.6	10,295	25,917	658	1,508	76.1	3.2
Tracktor++v2 [18]	46.6	47.6	37.6	18.2	27.9	**4,624**	26,896	1,290	1,702	**79.9**	1.4
mfi_tst [22]	49.2	52.4	41.5	29.1	24.4	8,707	21,594	912	1,397	79.0	0.7
ApLift [26]	51.1	59.0	45.7	39.4	22.6	10,070	19,288	677	1,022	79.3	0.6
MPNTrack [27]	51.5	58.6	45.0	31.2	25.9	7,620	21,780	**375**	**872**	79.4	6.5
Lif_T [25]	52.5	60.0	46.0	33.8	25.8	6,837	21,610	730	1,047	79.8	1.5
Ours	**56.9**	**62.3**	**47.3**	**42.3**	**17.2**	8,450	**17,620**	400	1,277	79.4	**22.1**

TABLE X
COMPARISON ON THE MOT17 BENCHMARK

Tracker	MOTA ↑	IDF1 ↑	HOTA ↑	MT(%) ↑	ML(%)↓	FP ↓	FN ↓	ID Sw. ↓	Frag ↓	LocA ↑	Hz ↑
MPNTrack [27]	58.8	61.7	49.0	28.8	33.5	17,413	213,594	1,185	2,265	81.5	6.5
LPC_MOT [28]	59.0	**66.8**	51.5	29.9	33.9	23,102	206,948	**1,122**	**1,943**	80.9	4.8
mfi_tst [22]	60.1	58.8	47.2	26.0	29.7	13,503	209,475	2,065	3,829	81.5	2.2
ApLift [26]	60.5	65.6	51.1	33.9	30.9	30,609	190,670	1,709	2,672	80.7	1.8
Lif_T [25]	60.5	65.6	51.3	27.0	33.6	14,966	206,619	1,189	3,476	81.3	0.5
CTTrackPub [20]	61.5	59.6	48.2	26.4	31.9	14,076	200,672	2,583	4,965	81.7	17
TMOH [23]	**62.1**	62.8	50.4	26.9	31.4	**10,951**	201,195	1,897	4,622	**82.4**	0.7
Ours	62.0	65.7	**51.9**	**35.0**	**27.4**	25,628	**187,163**	1,457	6,029	81.5	**19.8**

TABLE XI
COMPARISON ON THE KITTI BENCHMARK

Tracker	MOTA ↑	MT(%) ↑	ML(%) ↓	FP ↓	FN ↓	ID Sw. ↓	Frag ↓	Sensor	Hz ↑
JCSTD [31]	43.4	18.56	34.36	11,976	**885**	236	975	-	14.3
MPNTrack [27]	46.2	**43.99**	**10.31**	**6,956**	5,096	397	1,078	-	**50.0**
MDP [17]	47.0	25.77	28.52	9,550	2,502	213	738	-	1.1
NOMT [11]	47.1	27.84	34.71	10,433	1,676	**141**	**512**	-	11.1
Mono_3D_KF [51]	45.4	33.68	26.46	8,865	3,498	267	655	GPS	3.3
Be-Track [40]	50.9	22.34	32.65	9,953	1,225	199	731	Laser Points	**50.0**
CAT [35]	52.0	34.71	24.4	9,199	1,722	201	619	Stereo Image	-
Ours	**52.6**	36.43	22.68	7,359	3,185	427	628	-	24.0

method only uses monocular video data without calibration information and obtained better MOT results. The algorithms used in our method are all designed for quasi-real time system. Only with the delay of sliding window size and using real-time detectors, our method can achieve near-online effect for mobile devices. The benchmark result can also be found in the MOT Challenge website[1] and KITTI Benchmark website.[2]

VI. CONCLUSION

In this article, a hybrid motion model was proposed to address the motion modeling problem of MOT in mobile devices. Through the motion hypothesis evaluation, the camera motion was estimated for world coordinates projection. Our method reduces the estimation error and avoid the requirement of additional information such as calibration. Using horizon perspective, smooth dynamic projection in the model extracts the world coordinate, which avoids the interference of camera motion and results in higher tracking accuracy. Meanwhile, MMF solves the motion measurement and prediction problem for different motion modes and it adapts to object motion estimation under the motion camera. In the tracking framework, STEM provides more accurate affinity measurement for tracklets. The experimental result showed that our method has simple parameter setting and high robustness. A comparison result on MOT and KITTI benchmark demonstrated a competitive performance over other state-of-the-art methods.

[1]https://motchallenge.net/results/MOT17/

[2]http://www.cvlibs.net/datasets/kitti/eval_tracking.php

REFERENCES

[1] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[2] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[3] F. Yang, W. Choi, and Y. Lin, "Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2129–2137.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Int. Conf. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[5] P. Dai, X. Wang, W. Zhang, and J. Chen, "Instance segmentation enabled hybrid data association and discriminative hashing for online multi-object tracking," *IEEE Trans. Multimedia*, vol. 21, no. 7, pp. 1709–1723, Jul. 2019.

[6] J. Shen, Z. Liang, J. Liu, H. Sun, L. Shao, and D. Tao, "Multiobject tracking by submodular optimization," *IEEE Trans. Cybern.*, vol. 49, no. 6, pp. 1990–2001, Jun. 2019.

[7] H. Sheng, J. Chen, Y. Zhang, W. Ke, Z. Xiong, and J. Yu, "Iterative multiple hypothesis tracking with tracklet-level association," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 12, pp. 3660–3672, Dec. 2019.

[8] Y. Zhang et al., "Long-term tracking with deep tracklet association," *IEEE Trans. Image Process.*, vol. 29, pp. 6694–6706, 2020.

[9] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua, "Multi-commodity network flow for tracking multiple people," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1614–1627, Aug. 2014.

[10] A. Milan, K. Schindler, and S. Roth, "Detection-and trajectory-level exclusion in multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 3682–3689.

[11] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3029–3037.

[12] J. Xiang, G. Xu, C. Ma, and J. Hou, "End-to-end learning deep CRF models for multi-object tracking deep CRF models," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 275–288, Jan. 2021.

[13] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4696–4704.

[14] G. Bourmaud and R. Mégret, "Robust large scale monocular visual SLAM," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1638–1647.

[15] M. Jaimez and J. Gonzalez-Jimenez, "Fast visual odometry for 3-D range sensors," *IEEE Trans. Robot.*, vol. 31, no. 4, pp. 809–822, Aug. 2015.

[16] R. Gao et al., "Glow in the dark: Smartphone inertial odometry for vehicle tracking in GPS blocked environments," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12955–12967, Aug. 2021.

[17] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2017, pp. 924–933.

[18] P. Bergmann, T. Meinhardt, and L. Leal-Taixé, "Tracking without bells and whistles," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 941–951.

[19] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process.*, 2017, pp. 3645–3649.

[20] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 474–490.

[21] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the fairness of detection and re-identification in multiple object tracking," *Int. J. Comput. Vis.*, vol. 129, no. 11, pp. 3069–3087, 2021.

[22] J. Yang, H. Ge, J. Yang, Y. Tong, and S. Su, "Online multi-object tracking using multi-function integration and tracking simulation training," *Appl. Intell.*, vol. 52, pp. 1268–1288, May 2021.

[23] D. Stadler and J. Beyerer, "Improving multiple pedestrian tracking by track management and occlusion handling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 10958–10967.

[24] Y. Zhang, H. Sheng, Y. Wu, S. Wang, W. Ke, and Z. Xiong, "Multiplex labeling graph for near-online tracking in crowded scenes," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 7892–7902, Sep. 2020.

[25] A. Hornakova, R. Henschel, B. Rosenhahn, and P. Swoboda, "Lifted disjoint paths with application in multiple object tracking," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 4364–4375.

[26] A. Hornakova, T. Kaiser, P. Swoboda, M. Rolinek, B. Rosenhahn, and R. Henschel, "Making higher order MOT scalable: An efficient approximate solver for lifted disjoint paths," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 6330–6340.

[27] G. Brasó and L. Leal-Taixé, "Learning a neural solver for multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6247–6257.

[28] P. Dai, R. Weng, W. Choi, C. Zhang, Z. He, and W. Ding, "Learning a proposal classifier for multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2443–2452.

[29] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. AC-24, no. 6, pp. 843–854, Dec. 1979.

[30] D. J. Papageorgiou and M. R. Salpukas, "The maximum weight independent set problem for data association in multiple hypothesis tracking," in *Optimization and Cooperative Control Strategies*. Berlin, Germany: Springer, 2009, pp. 235–255.

[31] W. Tian, M. Lauer, and L. Chen, "Online multi-object tracking using joint domain information in traffic scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 374–384, Jan. 2020.

[32] J. García, A. Gardel, I. Bravo, J. L. Lázaro, and M. Martínez, "Tracking people motion based on extended condensation algorithm," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 3, pp. 606–618, May 2013.

[33] E. Morais, A. Ferreira, S. A. Cunha, R. M. L. Barros, A. Rocha, and S. Goldenstein, "A multiple camera methodology for automatic localization and tracking of futsal players," *Pattern Recognit. Lett.*, vol. 39, pp. 21–30, Apr. 2014.

[34] S. Zhang, X. Yu, Y. Sui, S. Zhao, and L. Zhang, "Object tracking with multi-view support vector machines," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 265–278, Mar. 2015.

[35] U. Nguyen, F. Rottensteiner, and C. Heipke, "Confidence-aware pedestrian tracking using a stereo camera," *ISPRS Ann. Photogrammetry Remote Sens. Spatial Inf. Sci.*, vol. 4, pp. 53–60, May 2019.

[36] X. Liu, G. Chen, X. Sun, and A. Knoll, "Ground moving vehicle detection and movement tracking based on the neuromorphic vision sensor," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 9026–9039, Sep. 2020.

[37] M. Luber, L. Spinello, and K. O. Arras, "Learning to detect and track people in RGBD data," in *Proc. RGB-D Workshop RSS*, vol. 2, 2011, pp. 1–2.

[38] L. Cao, C. Wang, and J. Li, "Robust depth-based object tracking from a moving binocular camera," *Signal Process.*, vol. 112, pp. 154–161, Jul. 2015.

[39] F. Fang, K. Qian, B. Zhou, and X. Ma, "Real-time RGB-D based people detection and tracking system for mobile robots," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, 2017, pp. 1937–1941.

[40] M. Dimitrievski, P. Veelaert, and W. Philips, "Behavioral pedestrian tracking using a camera and lidar sensors on a moving vehicle," *Sensors*, vol. 19, no. 2, p. 391, 2019.

[41] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "Robust multiperson tracking from a mobile platform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1831–1846, Oct. 2009.

[42] C. Wojek, S. Walk, S. Roth, K. Schindler, and B. Schiele, "Monocular visual scene understanding: Understanding multi-object traffic scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 882–897, Apr. 2013.

[43] W. Choi, C. Pantofaru, and S. Savarese, "A general framework for tracking multiple people from a moving camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1577–1591, Jul. 2013.

[44] M. Wan, G. Gu, W. Qian, K. Ren, X. Maldague, and Q. Chen, "Unmanned aerial vehicle video-based target tracking algorithm using sparse representation," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9689–9706, Dec. 2019.

[45] X. Deng, J. Li, P. Guan, and L. Zhang, "Energy-efficient UAV-aided target tracking systems based on edge computing," *IEEE Internet Things J.*, vol. 9, no. 3, pp. 2207–2214, Feb. 2022.

[46] Y. Liu, J. Shen, W. Wang, H. Sun, and L. Shao, "Better dense trajectories by motion in videos," *IEEE Trans. Cybern.*, vol. 49, no. 1, pp. 159–170, Jan. 2019.

[47] J. Shen, J. Peng, and L. Shao, "Submodular trajectories for better motion segmentation in videos," *IEEE Trans. Image Process.*, vol. 27, pp. 2688–2700, 2018.

[48] H. A. P. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with Markovian switching coefficients," *IEEE Trans. Autom. Control*, vol. 33, no. 8, pp. 780–783, Aug. 1988.

[49] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation With Applications to Tracking and Navigation: Theory Algorithms and Software*. New York, NY, USA: Wiley, 2004.

[50] A. Genovesio, T. Liedl, V. Emiliani, W. J. Parak, M. Coppey-Moisan, and J.-C. Olivo-Marin, "Multiple particle tracking in 3-D+t microscopy: Method and application to the tracking of endocytosed quantum dots," *IEEE Trans. Image Process.*, vol. 15, pp. 1062–1070, 2006.

[51] A. Reich and H.-J. Wuensche, "Monocular 3D multi-object tracking with an EKF approach for long-term stable tracks," in *Proc. IEEE 24th Int. Conf. Inf. Fusion*, 2021, pp. 1–7.

[52] L.-C. Chiu, T.-S. Chang, J.-Y. Chen, and N. Y.-C. Chang, "Fast SIFT design for real-time visual feature extraction," *IEEE Trans. Image Process.*, vol. 22, pp. 3158–3167, 2013.

[53] B. Micusik and T. Pajdla, "Estimation of omnidirectional camera model from epipolar geometry," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2003, pp. 485–490.

[54] J. Shen, X. Hao, Z. Liang, Y. Liu, W. Wang, and L. Shao, "Real-time superpixel segmentation by DBSCAN clustering algorithm," *IEEE Trans. Image Process.*, vol. 25, pp. 5933–5942, 2016.

[55] J. Díaz, E. Ros, F. Pelayo, E. M. Ortigosa, and S. Mota, "FPGA-based real-time optical-flow system," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 2, pp. 274–279, Feb. 2006.

[56] J.-K. Lee and K.-J. Yoon, "Real-time joint estimation of camera orientation and vanishing points," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1866–1874.

[57] D. Fraser and J. Potter, "The optimum linear smoother as a combination of two optimum linear filters," *IEEE Trans. Autom. Control*, vol. AC-14, no. 4, pp. 387–390, Aug. 1969.

[58] H. Sheng, Y. Zhang, J. Chen, Z. Xiong, and J. Zhang, "Heterogeneous association graph fusion for target association in multiple object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 11, pp. 3269–3280, Nov. 2019.

[59] L. Chen, H. Ai, Z. Zhuang, and C. Shang, "Real-time multiple people tracking with deeply learned candidate selection and person re-identification," in *Proc. IEEE Conf. Multimedia Expo*, 2018, pp. 1–6.
[60] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "MotChallenge 2015: Towards a benchmark for multi-target tracking," 2015, *arXiv:1504.01942*.
[61] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*.
[62] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3354–3361.
[63] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image Video Process.*, vol. 2008, pp. 1–10, May 2008.
[64] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 17–35.
[65] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybridboosted multi-target tracker for crowded scene," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2953–2960.
[66] J. Luiten et al., "HOTA: A higher order metric for evaluating multi-object tracking," *Int. J. Comput. Vis.*, vol. 129, no. 2, pp. 548–578, 2021.
[67] P. Chu, H. Fan, C. C. Tan, and H. Ling, "Online multi-object tracking with instance-aware tracker and dynamic model refreshment," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, 2019, pp. 161–170.

Yubin Wu received the B.S. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2016, where he is currently pursuing the Ph.D. degree.

His research interest is computer vision, and he is particularly interested in multiple object tracking.

Hao Sheng (Member, IEEE) received the B.S. and Ph.D. degrees from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2003 and 2009, respectively.

He is currently a Professor and a Ph.D. Supervisor with the School of Computer Science and Engineering, Beihang University. He is working on computer vision, pattern recognition, and machine learning.

Yang Zhang received the B.S. and Ph.D. degrees from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2014 and 2020, respectively.

He is currently an Associate Professor with the College of Information Science and Technology, Beijing University of Chemical Technology, Beijing. He is working on computer vision, pattern recognition, and machine learning.

Shuai Wang received the B.S. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2019, where he is currently pursuing the Ph.D. degree.

His research interest is computer vision, and he is particularly interested in multiple object tracking.

Zhang Xiong (Member, IEEE) received the B.S. degree from Harbin Engineering University, Harbin, China, in 1982, and the M.S. degree from Beihang University, Beijing, China, in 1985.

He is a Professor and a Ph.D. Supervisor with the School of Computer Science and Engineering, Beihang University. He is working on computer vision, information security, and data vitalization.

Wei Ke (Member, IEEE) received the Ph.D. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2012.

He is a Professor with the Faculty of Applied Sciences, Macao Polytechnic University, Macau, China. His research interests include programming languages, image processing, computer graphics and component-based engineering and systems. His recent research focuses on the design and implementation of open platforms for applications of computer graphics and pattern recognition.