Enhanced Initialization for Track-before-Detection based Multibody Motion Segmentation from a Moving Camera

Hernan Gonzalez, Arjun Balakrishnan, Sergio Alberto Rodriguez Florez, Abdelhafid Elouardi

Enhanced Initialization for Track-before-Detection based Multibody Motion Segmentation from a Moving Camera

Hernan Gonzalez *, Arjun Balakrishnan†, Sergio A. Rodríguez F.‡, Abdelhafid Elouardi§
*†‡§ ENS Paris-Saclay. Paris-Sud University. Paris-Saclay University, 91405, Orsay, France
*†‡§ SATIE Laboratory CNRS Joint research unit - UMR 8029
Email: *hernan.gonzalez@u-psud.fr,† arjun.balakrishnan@u-psud.fr,‡sergio.rodriguez@u-psud.fr,
§abdelhafid.elouardi@u-psud.fr

***Abstract*—Vision-based motion segmentation provides key information for dynamic scene understanding and decision making in autonomous navigation. In this paper, we propose an enhanced initialization strategy which is tightly coupled to a multibody Structure from Motion (SfM) segmentation employing a monocular camera. The method relies on epipolar geometry, RANSAC formulation and motion estimation for segmenting ego-motion and eoru-motions. The proposed strategy is intended to enhance the initialization procedure employed in [1] obtaining a 50 times speed-up factor. This result is used as input for the Track before Detection (TbD) methodology which efficiently simplifies the existing multibody Structure from Motion implementation approaches. An evaluation using images from a publicly available dataset with dynamic traffic scenarios and large camera motions confirms the effectiveness of the method.**

I. Introduction

Scene understanding is an essential topic in the development of fully Autonomous Vehicles (SAE *Level* 5). In this context, multiple sensors systems as radar, LiDAR and cameras are used to provide redundant and reliable information of the scene [2]. Vision sensor approaches are intended to retrieve dynamic information of the vehicle surroundings [3], [4], [5].

Image motion segmentation has been widely studied because of the complexity of uncontrolled conditions in real scenes, due to illumination and planar ground changes. Multiple motion segmentation methods have been studied as it is surveyed in [6]. The first methods that introduced motion factorization were [7], [8]. These methods were founded on algebraic and geometric formulations, however, they are sensitive to noise. Motion clustering methods in [9], [10], [11] can achieve precise results but they were tested in dataset scenes including tracked image features. Such kind of tests neglects the impact of feature tracking errors. Multibody Structure from Motion (SfM) formulation was used in [12] for segmenting scene motions, computing 3D structures of objects and inferring camera motion. In [13], a rough segmentation of moving objects is improved by means of a bi-linear optimization procedure which takes advantage of their 3D shape and metric constraints. In [14], the authors propose a multibody segmentation by introducing hypotheses generation with precise results, however, this approach entails a high computational cost. This study was enhanced in [15] by coupling the kinematic constraints of ground vehicles to improve overall performance.

Multibody Visual Simultaneous Localization and Mapping (VSLAM) also addresses multiple dynamic objects segmentation. In [16] was introduced the Bearing only Tracking (BOT) to segment moving objects as a complement of the VSLAM approach. In [17] the authors present a stereo-vision based multibody visual SLAM to segment objects by estimating their relative distances to the camera.

This work presents an enhanced variant of the Track-before-Detect-SfM (TbD-SfM) method introduced in [1]. It is worth noting that the approach introduced in [14] was employed in [1] for initializing motion segmentation. This paper outlines a closed-form approach to segment six degree of freedom (6DoF) simultaneous motion based on geometric constraints and RANSAC formulation. The contribution of this paper is a new initialization approach for TbD-SfM method intended to reduce computational complexity. The proposed initialization begins with the ego-motion segmentation. Epipolar geometry is applied between the first and last image of a temporal sliding window to estimate static feature points. Next, bucketing technique is implemented over these features to select k points and compute the motion between the consecutive image pair of a sliding window. The estimated motion is applied to all feature points and ego-motion points are segmented. Then, a 6DoF motion estimation method that uses the hypotheses generation is used to find eoru-motions in the remaining features set. Results of the initial segmentation are employed as the input of the TbD-SfM approach. TbD-SfM implements a Bayesian estimation of the dynamic object area position and then a motion estimation by using feature points of the area. Our method was tested on KITTI dataset [18] and the results are evaluated following three criteria: execution time, segmentation statistics and reprojection errors as proposed in [9], [14].

This paper is structured as follows: Sec. II introduces the

fundamentals of single and multiple motions analysis in the SfM formulation, motion and structure recovery, and motion hypotheses generation. Sec. III details the initialization step and the TbD-SfM methodology for multibody motion segmentation. Sec. IV presents an application under full scale dynamic scenes and the corresponding motion segmentation results. Finally, the paper is concluded and perspectives are outlined in Sec. V.

II. Structure from Motion Factorization Fundamentals

A. Structure from Motion Factorization Fundamentals

A scene composed of static and dynamic rigid objects is analyzed by means of image features points. Such a set of 2D feature points are tracked and matched over a sequence composed of f consecutive images. The 2D feature points set cardinality is denoted by p. Considering these assumptions, the factorization approach in [7] addresses two problems: (i) recovering the 3D structure of the scene (up to a scale factor) and (ii) estimating inter-frame camera motion (namely ego-motion). The trajectory matrix represented by $W \in \mathbb{R}^{3f\times p}$ is composed of feature coordinates on the images along the sequence. Feature coordinates are defined as $w_p = [u_p, v_p, 1]^T \in \mathbb{R}^{3f\times 1}$ and tracked features coordinates along a frame sequence are arranged in a column vector of W as $w_p = [w_{1p}, w_{2p}, ..., w_{fp}]^T$. Inter-frame camera motion is represented as a 6DoF 3D rigid transformation, $M = [R\,|\,t]$ with $M \in \mathbb{R}^{3f\times 4}$. Rotation is denoted as $R \in \mathbb{R}^{3\times 3}$ and translation by $t \in \mathbb{R}^{3\times 1}$. Finally, the recovered structure, $S \in \mathbb{R}^{4\times p}$, is defined as a set 3D homogeneous coordinates vectors $s_p = [s_x, s_y, s_z, 1]^T$ as stated in Eq. (1):

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1p} \\ w_{21} & w_{22} & \cdots & w_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ w_{f1} & w_{f2} & \cdots & w_{fp} \end{bmatrix}, M = \begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_f \end{bmatrix} \quad (1)$$

$$S = \begin{bmatrix} s_1 & s_2 & \cdots & s_p \end{bmatrix}$$

Thus, single motion general formulation of SfM is as follows:

$$W_{3f\times p} = M_{3f\times 4} \cdot S_{4\times p} \quad (2)$$

The factorization of trajectory matrix W provides a direct estimation of motion M and structure S. As a solution, Eq. (3) states $\widetilde{W}$ representing a rank-4 trajectory matrix estimation obtained from camera motion, $\widetilde{M}$, and structure, $\widetilde{S}$, estimates:

$$\widetilde{W}_{3f\times p} \approx \widetilde{M}_{3f\times 4}\widetilde{S}_{4\times p} \quad (3)$$

B. Structure from Motion Factorization for Multiple motions

Multibody motion segmentation approach in [8] simplifies the camera motion and structure estimation of multiple dynamic objects as is formulated in Eq. (4). The trajectory matrix of each n independent body motion is represented by $W_n \in \mathbb{R}^{3f\times p}$, and the trajectories matrices union constitute the multibody trajectory matrix W. The motion of the n independent dynamic objects are denoted as $M_n \in \mathbb{R}^{3f\times 4}$, and their union defines the multibody camera motion described by $M \in \mathbb{R}^{3f\times 4n}$. Finally, multibody 3D structure, $S \in \mathbb{R}^{4n\times p}$, encloses in a sparse form the structure of each body, $S_n \in \mathbb{R}^{4\times p}$, in a diagonal matrix. Eq. (4) is solved by factorizing W_n of each trajectory matrix.

$$[W_1|...|W_n] = [M_1|...|M_n] \cdot \begin{bmatrix} S_1 & 0 & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & S_n \end{bmatrix} \quad (4)$$

C. Motion and Structure Recovering

$\overline{W}$ represents the trajectory matrix normalized byusing the 8-point algorithm. Then,k points are sampled from $\overline{W}$ in a pair of consecutive frames. The sampled points are represented in the first and second frame respectively by $\overline{w}_f = [\overline{w}_1, \overline{w}_2, ..., \overline{w}_k]^T$ and $\overline{w}'_f = [\overline{w}'_1, \overline{w}'_2, ..., \overline{w}'_k]^T$. A first feature point $\overline{w}_i$ is randomly selected and more features, $\overline{w}'_i$, are assigned to it using a nearest neighbor criterion [19] weighted by the probability distribution of Eq. (5). ζ and ρ values are heuristically selected according to the probability scale.

$$Pr(\overline{w}_i|\overline{w}'_i) = \begin{cases} \frac{1}{\zeta} exp - \frac{\|\overline{w}_i - \overline{w}'_i\|^2}{\rho^2} & if\, \overline{w}_i \neq \overline{w}'_i \\ 0 & if\, \overline{w}_i = \overline{w}'_i \end{cases} \quad (5)$$

Then, essential matrix, E, is computed in a least square form, $A \cdot x = 0$, where A includes the set of points $\overline{w}_f$ and $\overline{w}'_f$ to enforce the epipolar constraints over matrix E as is detailed in Eq. (6).

$$\overline{w}'^T_f \cdot E \cdot \overline{w}_f = 0 \quad (6)$$

The inter-frame camera motion, $\widetilde{M} = [R|t]$, is computed from the singular-value decomposition (SVD) of E as follows:

$$UDV^T = SVD\,(E) \quad (7)$$

where four possible solutions $\left[UQV^T \pm U_{3c}\right]$ and $\left[UQ^TV^T \pm U_{3c}\right]$ are tested so as to find the correct combination in order to rebuild the scene in front of the camera. Finally, the structure $\widetilde{S}_k \in \mathbb{R}^{4\times k}$ is calculated by the SVD of the camera projection matrix between an image pair.

D. Generation of motion hypotheses

In order to segment the scene, motion $\widetilde{M}^h$ and structure $\widetilde{S}^h$ hypotheses are generated along the sliding window as explained in Sec. II-C. The hypotheses are evaluated by considering the features reprojection error and the number of inliers associated. Reprojection error is defined as the average difference between the trajectory matrix W and its estimation $\widetilde{W}$ according to Eq. (8). Inliers are defined as features points with a reprojection error lower than the maximum error allowed ϵ_{pto}. The hypothesis obtaining the highest number of inliers is selected as the best one to describe the observed motion.

$$\sum_{f=1}^{\Gamma} \frac{1}{\Gamma} \left\| W - \left(\widetilde{M}^h \cdot \widetilde{S}^h \right) \right\| \leq \epsilon_{pto} \quad (8)$$

III. Track-before-Detect Framework

The multibody SfM based approach introduced in [14] provides a suitable solution for scene motion segmentation. However, its computational cost increases exponentially with the amount of dynamic objects in the scene. The authors proposed an accelerated variant in [15] by constraining observed motions to evolve over a ground plane. The variant limits the motion model estimation from 6-DOF to 2-DOF. TbD-SfM was proposed as 6-DOF scene motion segmentation tightly coupling motion detection and temporal filtering of multiple dynamic image regions. Dynamic regions are tracked to restrict the searching area, to preserve a high feature points density in the regions, and to reduce the computational cost without kinematics constraint in the motion estimation.

A. Track-before-Detect Initialization

In order to implement the TbD-SfM approach, it is necessary an initial segmentation of the dynamic objects of the scene. The observed motions in the scene are the ego-motion (set of static feature points) and the eoru-motions (groups of dynamic feature points). The ego-motion is represented by the set with the largest amount of feature points (dominant motion assumption). The eoru-motions are dynamic objects and they will be assigned to the inputted dynamic regions. This stage outputs a first estimation of the size, location, and number of dynamic objects in the scene. At the end of this section, the initialization procedure is summarized in Alg. 1.

1) Ego-motion segmentation: The proposed method to segment the ego-motion feature points relies on epipolar geometry of two subsequent views. Epipolar geometry is independent of scene structure, only depends on the intrinsic camera parameters and the relative camera pose [20]. Let consider a uncalibrated monocular moving camera whose center is denoted C_t at time t and its next position at time $t+1$, C_{t+1}, as show in Fig. 1.

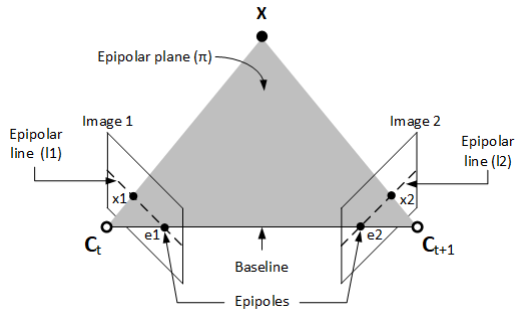

Fig. 1. Two views Epipolar Geometry representation

X is a 3D point projected onto image 1 and image 2, and its homogeneous image coordinates are represented by $x1 = [u, v, 1]^T$ and $x2 = [u', v', 1]^T$ in both views respectively. In order to infer which feature points lie onto static objects, the fundamental matrix, F, is robustly estimated. F is the algebraic representation of the epipolar geometry and satisfies epipolar constraints presented in Eq. (9).

$$x_2^T \cdot F \cdot x_1 = 0 \quad (9)$$

Fundamental matrix is computed using the RANSAC (RANdom SAmple Consensus) strategy between the first frame and the last frame of the temporal sliding window in trajectory matrix W. It is necessary a minimum sample set of $k = 8$ feature points to estimate F. Feature points with an error distance to the epipolar line lower than a threshold are classified as inliers or potential static features. In order to determine which feature points are static, a stage of hypotheses generation is implemented using inliers features.

The hypotheses generation begins applying a bucketing clustering of inliers features. Then, k feature points of each bucket are sampled along the temporal sliding window. To this end, the image is divided in 15x10 buckets. The bigger the amount of features in a bucket, the higher the probability of sampling features of such a bucket. Giving a set of k feature points represented by W_k, the motion, $\widetilde{M}^h$, and structure, $\widetilde{S}_k^h$, of the hypothesis are computed between a consecutive pair of frames along the sliding window as is presented in Sec. II-C.

Relative motion between frames, $\widetilde{M}^h$, and trajectory matrix, W, are used to calculate structure, $\widetilde{S}^h$, of the set of inliers points. The trajectory matrix is estimated $\widetilde{W}^h$ with the motion $\widetilde{M}^h$ and structure $\widetilde{S}^h$ hypothesis as Eq. (3).

Hypotheses are evaluated by comparing W and $\widetilde{W}^h$ as was presented in Sec. II-D.The hypotheses generation is repeated until finding a hypothesis with a percentage of inliers greater than a established value (e.g. 90 %). In this case, the hypothesis will become the best estimation and it will be defined by $\widetilde{W}$, $\widetilde{M}$ and $\widetilde{S}$. The structure is determined taking into account that 3D points must be located in front of the camera, that is deep coordinates (Z) are positive values. Structure points not satisfying such a constraint are classified as outliers and are discarded. Later, feature points with a reprojection error greater than ϵ_{pto} are considered outliers and removed as shown in Eq. (8), and then, the $\widetilde{S}$ is updated.

Up to this point, an initial group of ego-motion feature points is segmented. This group is used to find the ego-motion features remaining in the points classified as oultiers by the epipolar geometry. To this end, a trajectory matrix with these points is created. Then, a structure is calculated by using the trajectory matrix and the motion hypothesis $\widetilde{M}$. Next, the outliers are removed by checking the structure and the reprojection error as it was described before, and finally the structure is updated.

2) Eoru-motion segmentation: After the ego-motion segmentation, the remaining feature points can be classified as dynamics objects or outliers. In order to classify these points, the generation of motion hypotheses is implemented to find one or more sets of features that represent the dynamic objects (see Sec. II-D). In order to avoid false positive segmentation, the sets of features points considered as dynamic objects must be observed in previous sliding windows. An ROI image correlation is carried out between

frames so as to find previously observed dynamic objects. Finally image features lying on dynamic objects are enclosed in dynamic regions.

3) Representation of dynamic regions: Dynamic regions frame potential moving objects and associate their corresponding features along a temporal sliding window. This dynamic region model is not suited for ego-motion features since they are usually spread over a large image area. Thus, dynamic regions are only employed to frame potential eoru-motions in the tracking scheme. A dynamic region is modeled by a bounding box centered at (u, v) with height, h, and width, w in pixel units.

Algorithm 1 Initialization procedure

procedure
Input: Trajectory Matrix W
Output: $W_{1,\dots,n}, M_{1,\dots,n}, S_{1,\dots,n}$ ▷ n: number of motions
  Segment ego-motion from W using epipolar geometry
  while percentage of inliers $\leqslant$ threshold **do**
    Select k features using bucketing technique
    Compute $\widetilde{M}^h$ between pair of frames with W
    Compute $\widetilde{S}^h$ with $\widetilde{M}^h$ and W
    Compute $\widetilde{W}^h$ with $\widetilde{S}^h$ and $\widetilde{M}^h$
    Compute the reprojection error $\left\|W - \widetilde{W^h}\right\|$
  end
  Remove outliers from $\widetilde{W}$, $\widetilde{S}$ and Update $\widetilde{S}$
  Compute $\widetilde{S}$ with $\widetilde{M}$ and W
  Remove outliers from $\widetilde{W}$, $\widetilde{S}$ and Update $\widetilde{S}$
  $W = [W_1, W_{outliers}]$ with W_1 as ego-motion
  Generate motion hyp. from $W_{outliers}$as in Sec.III-A2
  return $W_{1,\dots,n}, M_{1,\dots,n}, S_{1,\dots,n}$

B. Track-before-Detection for Scene Analysis

In this section TbD approach is introduced for motion segmentation. The method begins by finding ego-motion feature points, named as W_1. Thus, features points inside dynamic regions $(W_2, ..., W_n)$ are removed from the trajectory matrix as:

$$W_1 = W - [W_2|W_3|...|W_n] \tag{10}$$

It is worth noting that the set of ego-motion features W_1 could include outliers or misclassified points. To cope with this, a robust estimation of the fundamental matrix F is computed with RANSAC on the trajectory matrix W_1. Then, the motion, $\widetilde{M}_1$, and structure, $\widetilde{S}_1$, are computed as in Sec. III-A1. The set of segmented features must satisfy the minimum amount of features k required to compute a motion. This constraint is evaluated by checking the number of columns of the trajectory matrix W_n as follows: $m = col(W_n) - k$.

In case of multiple motion solutions with equal number of inliers, the hypothesis achieving the smallest mean reprojection error is selected. The remaining features represent outliers or a new observed motion in the scene.

1) Motion Factorization on Dynamic Regions: Motion estimation of dynamic objects is carried out using features enclosed in each dynamic region $[W_2|W_3|...|W_n]$. The trajectory matrices $\widetilde{W}_n$ are composed of points that follow the n^{th} dynamic object . Since some features in dynamics regions might be missed classified, the first step consists in detecting features belonging to ego-motion group. To this end, a structure is computed with the dominant motion $\widetilde{M}_1$ and each dynamic points group W_n. Feature points with a positive Z-coordinate value and a reprojection error lower than ϵ_{pto} are re-assigned as ego-motion feature points and the ego-motion structure $\widetilde{S}_1$ is updated. Then, remaining feature points in each group are used to calculate the eoru-motion $\widetilde{M}_n$, structure $\widetilde{S}_n$ and the estimated trajectory matrix $\widetilde{W}_n$ as detailed in Sec. II-D. The eoru-motion estimation is repeated by using the remaining feature points in order to find a new motion or classify such points as outliers.

2) Image region tracking: Multiple-Target Tracking (MTT) strategy is applied by using a set of Kalman filters (KF). It predicts the location of each dynamic region. This approach assumes that the dynamic objects perform smooth motion changes along the sequence. The position of the dynamic region on the image is tracked by a 8D state vector. Each region track state is represented by x_f as shown in Eq. (11), and it is composed by the image centroïd coordinates (x_c, y_c), in pixel, the width w, the height h and their first derivatives $(v_x, v_y, \delta_w, \delta_h)$ respectively:

$$x_f = [x_c, y_c, w, h, v_x, v_y, \delta_w, \delta_h]^T \tag{11}$$

A linear Gaussian model is used to track the regions by assuming a smooth-linear inter-frame motion as formalized in Eq. (12):

$$\begin{cases} x_f = A \cdot \ x_{f-1} + \alpha_f & \alpha_f \sim N(\alpha_f; 0, \Lambda_f) \\ y_f = C \cdot \ x_f + \beta_f & \beta_f \sim N(\beta_f; 0, \Gamma_f) \end{cases} \tag{12}$$

where A and C represent transition and observation models, respectively. x_{f-1} is the state vector in a previous sample frame and y_f the multivariate observations. α_f and β_f are the state and observation noise following zero-centered normal distributions with known variances.

3) Track-to-Motion Association: Each dynamic region is linked to a Kalman filter. The state is predicted by each filter and its prediction constraints the perimeter of the set of features points W_n for estimating an eoru-motion. Features obtained by the factorized motion are employed to update the position and size of the state vector if the dynamic region satisfies an association criterion. This criterion correlates the tracked dynamic region and the factorized one regarding their appearance and the uncertainty-weighted state given by the inverse of the mean reprojection error.

4) Track Creation and Deletion: In order to avoid false detections, a dynamic region must be confirmed along the temporal sliding window so as to validate the detection of a new dynamic object. When, the new object is detected, a filter is initialized to track it. Low-confidence tracks that are non-updated and non-associated are disposed. It worth to highlight that new objects can be detected if there are at least 8 feature points following such a motion and meet the reprojection error criterion ϵ_{pto} (as stated in Sec. II-D). Otherwise, such features will be rejected as outliers.

IV. Results

KITTI dataset [18] was employed to validate our new initialization approach for TbD-SfM motion segmentation. The database contains sequences of 1392x512 images sampled in uncontrolled light conditions from a camera embedded on a moving car. The dataset does not provide ground truth features for the scenes, so that there exists the possibility of feature tracking errors. Feature points are acquired by means of the Libviso2 extractor [21]. The scenes are processed in a temporal sliding window of 4 frames. The results obtained per sliding window are processed and the mean value is reported as a frame result.

The TbD-SfM method is intended to segment objects in a dynamic scene. The evaluation of the method is done by means of the execution time gain, the segmentation error, reprojection error and the outliers ratio.

The mean reprojection error scores the average difference between trajectory matrix W and its corresponding estimate $\widetilde{W}$ in the sliding window as was detailed in Sec.II-D.

The execution time gain is the ratio between the time obtained with the initialization method (A) [14] and the time obtained with our initialization method (B).

$$ratio = \frac{Time\,Method\,(A)}{Time\,Method\,(B)} \tag{13}$$

Segmentation error [9] quantifies the number of misclassified points. It is calculated as follows:

$$Seg.\,Error = 100\frac{\#\,of\,misclassified\,points}{Total\,\#\,of\,points} \tag{14}$$

Feature points that are not segmented by a motion are defined as outliers. These points do not achieve a reprojection error lower than the established threshold (ϵ_p). It is calculated as:

$$Outliers\,Ratio = 100\frac{\#\,of\,unclassified\,points}{Total\,\#\,of\,points} \tag{15}$$

In the considered scene, there are two simultaneous motion until frame 18, the moving camera (ego-motion) and a car passing from the back to the front with high speed. Then, a third car appears in the same direction from frame 19 until frame 43 (ego-motion and 2 eoru-motions), and ends, with two motions from frame 44 until frame 100 (ego-motion and eoru-motion). This scene is challenging because the three vehicles drive in the same direction.

We have processed a total of 100 frames obtaining a segmentation error of 0.25% in the scene. This error was obtained with Eq. (14) and the average number of 1600 feature points per frame.

Fig. 2 shows the TbD-SfM results obtained in each step of the process. Fig. 2(a) presents the ego-motion features points (red markers) and their outlier points. In Fig. 2(b) the box indicates the tracking area where the dynamic objects must be segmented and the features points in the area (blue markers). Feature points belonging to the ego-motion are removed by comparing their structure with the ego-motion structure. Finally, Fig. 2(c) shows the final motion segmentation of the frame. The cyan markers represent outliers of the scene segmentation.

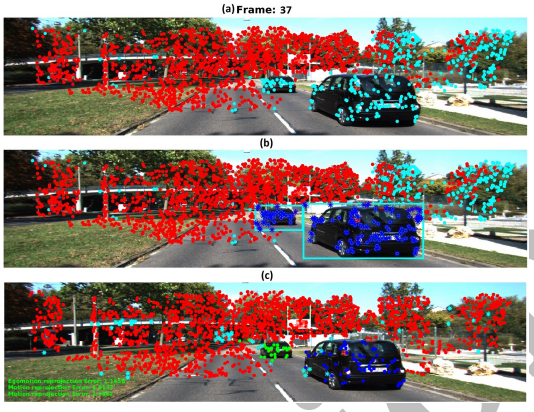

Fig. 2. Motion segmentation results in frame 37 with TbD-SfM: (a) Top figure shows segmented ego-motion features in red markers and outliers features in cyan. (b) Tracked regions in each dynamic object are illustrated. (c) Figure presents the motion segmented in each tracked area. Feature points of moving objects are detailed in blue and green color.

Fig. 3 presents the mean reprojection error of segmented motions in the temporal window. The ego-motion reprojection error is less than 1.9 pixels along the sequence (red color). The first dynamic object (green color) was segmented from the frame 1 until the frame 40 with a reprojection error less than 2 pixels. The second dynamic object was segmented from frame 21 until the last frame with a reprojection error less than 2.7 pixels. However, the eoru-motions are segmented as one dynamic object in the frame 41 because they are close.

In Fig. 3(b), zero values with a red cross mark represent frames where it was not possible to estimate the reprojection error because it is necessary at least 8 points to estimate the motion and the segmented group has less than this amount.

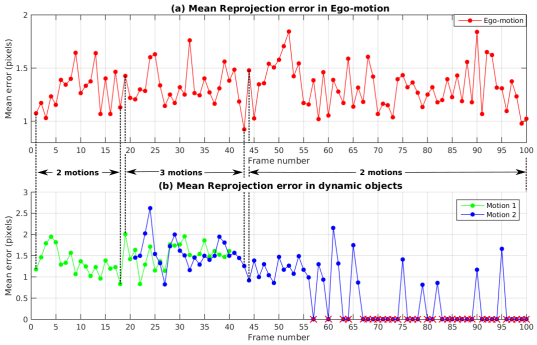

Fig. 3. Mean reprojection error in estimated motions: (a) Ego-motion, (b) Dynamic objects

Fig. 4 shows the outliers percentage along the sequence with a highest value of 14% in frame 90. The mean value of outliers was 2.88%, this proves that TbD-SfM method can estimate multiple motions with a low reprojection error, preserving the quantity of features points as is shown in Fig. 4. The oultiers percentage could increase the computational costs when a group of point are close enough. In this case the group is considered as a new possible object and it is analyzed with the method presented in Sec. II-D.

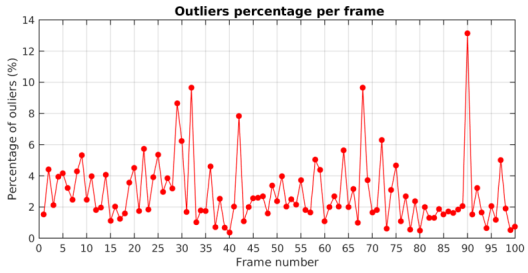

Fig. 4. Outliers percentage along the sequence.

Fig. 5 shows the execution time results for an initialization every 10 frames and 20 frames. It details the execution time per frame along the sequence. Other initializations are carried out in frames where the TbD-SfM method does not find any dynamic object as shown frames 71, 80 and from frame 92 until frame 95. The red dashed lines show frames in which initializations were executed. The results show that the initialization step does not have any influence over the execution time of the TbD-SfM motion segmentation method. The highest value obtained was $96\,s$ in the frame 51 because after ego-motion computation , motion estimation method (Sec. II-D) is performed over the remaining features points so as to find a new motion.

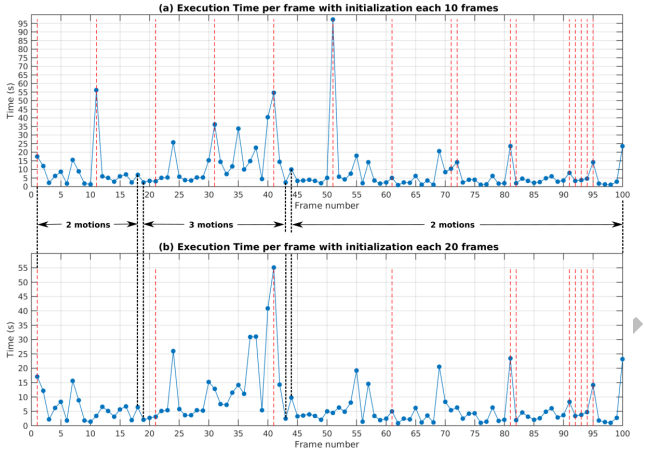

Fig. 5. Execution time along the sequence with initialization: (a) each 10 frames, (b) each 20 frames

Table I presents the execution time obtained every 20 frames with two different initializations.

The method proposed in [14] (Method A) with 300 hypotheses to segment the motions and our proposed approach (Method B). Our approach obtained a speed gain between 50 and 800 times.

V. Conclusions

In this paper, we have presented an enhanced initialization process integrated to a TbD-SfM method in order to segment motions from a moving monocular camera. Experiments show that the initialization proposed speeds up segmentation of dynamic objects without affecting the TbD-SfM segmentation process and any performance loss. Our method achieved a low segmentation error with a high amount of feature points segmented as shown by the outliers percentage. The closed-form approach preserves the density of feature points reducing the probability of losing dynamic objects. The enhanced execution time allows this method be a scalable when the number of simultaneous motions is increased. Future works will be devoted to 3D representation and reconstruction of the dynamic object trajectories. In addition, semantic information can be used to improve motion model of the tracked moving objects in a monocular system.

TABLE I
Execution time comparison each 20 frames

Frame	Method (A) presented in (min)	(B) Ours Method (s)	Ratio A/B
1	40	17	141
21	44	3.2	825
41	53	55	57
61	50	5	600
81	38	23.4	99

References

[1] H. Gonzalez, S. Rodriguez, and A. Elouardi, "Track-before-detect framework-based vehicle monocular vision sensors," *Sensors*, 2019.

[2] G. Zhao, X. Xiao, J. Yuan, and G. W. Ng, "Fusion of 3d-lidar and camera data for scene parsing," *Journal of Visual Communication and Image Representation*, 2014.

[3] K. Lim, Y. Hong, M. Ki, Y. Choi, and H. Byun, "Vision-based recognition of road regulation for intelligent vehicle," in *IEEE Intelligent Vehicles Symposium*, 2018.

[4] S. Sivaraman and M. M. Trivedi, "A review of recent developments in vision-based vehicle detection," in *IEEE Intelligent Vehicles Symposium*, 2013.

[5] L. Gao, W. Liu, R. Niu, Y. Sun, Y. Xin, and H. Liang, "Moving vehicle detection in dynamical scene using vector quantization," in *IEEE Intelligent Vehicles Symposium Proceedings*, 2014.

[6] L. Zappella, X. Lladó, and J. Salvi, "Motion segmentation: A review," in *Conf. on Artificial Intelligence Research and Development*, 2008.

[7] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: a factorization method," *Int. Journal of Computer Vision*, 1992.

[8] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *Int. Journal of Computer Vision*, 1998.

[9] R. Vidal, Y. Ma, S. Soatto, and S. Sastry, "Two-view multibody structure from motion," *Int. Journal of Computer Vision*, 2006.

[10] A. Goh and R. Vidal, "Segmenting motions of different types by unsupervised manifold clustering," in *Conf. on Computer Vision and Pattern Recognition*, 2007.

[11] R. Vidal and R. Hartley, "Three-view multibody structure from motion," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, 2007.

[12] K. E. Ozden, K. Schindler, and L. J. V. Gool, "Multibody structure-from-motion in practice." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2010.

[13] L. Zappella, A. Del Bue, X. Lladó, and J. Salvi, "Simultaneous motion segmentation and structure from motion," in *IEEE Workshop on Applications of Computer Vision*, 2011.

[14] R.Sabzevari and D.Scaramuzza, "Monocular simultaneous multi-body motion segmentation and reconstruction from perspective views," in *IEEE Int. Conf. on Robotics and Automation*, 2014.

[15] R. Sabzevari and D. Scaramuzza, "Multi-body motion estimation from monocular vehicle-mounted cameras," *IEEE Trans. on Robotics*, 2016.

[16] A. Kundu, K. M. Krishna, and C. V. Jawahar, "Realtime multibody visual slam with a smoothly moving monocular camera," in *Int. Conf. on Computer Vision*, 2011.

[17] N. D. Reddy, I. Abbasnejad, S. Reddy, A. K. Mondal, and V. Devalla, "Incremental real-time multibody vslam with trajectory optimization using stereo camera," in *IEEE Int. Conf. on Intelligent Robots and Systems*, 2016.

[18] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *Int. Journal of Robotics Research*, 2013.

[19] M. Zuliani, C. S. Kenney, and B. S. Manjunath, "The multiransac algorithm and its application to detect planar homographies," in *IEEE Int. Conf. on Image Processing*, 2005.

[20] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[21] A. Geiger, J. Ziegler, and C. Stiller, "Stereoscan: Dense 3d reconstruction in real-time," in *Intelligent Vehicles Symposium*, 2011.