3D Res (2019)10:11
https://doi.org/10.1007/s13319-019-0220-4

3DR EXPRESS

Nonparametric Statistical and Clustering Based RGB-D Dense Visual Odometry in a Dynamic Environment

使用RGBD数据在dynamic环境中的dense VO方法

Wugen Zhou · Xiaodong Peng · Haijiao Wang · Bo Liu

Received: 26 September 2018 / Revised: 17 February 2019 / Accepted: 18 February 2019

Abstract The robustness of dense-visual-odometry is still a challenging problem if moving objects appear in the scene. In this paper, we propose a form of dense-visual-odometry to handle a highly dynamic environment by using RGB-D data. Firstly, to find dynamic objects, we propose a multi-frame based residual computing model, which takes a far time difference frame into consideration to achieve the temporal consistency motion segmentation. Then the proposed method combines a scene clustering model and a nonparametric statistical model to obtain weighted cluster-wise residuals, as the weight describes how importantly a cluster residual is considered. Afterward, the motion segmentation labeling and clusters' weights are added to the energy function optimization of dense-visual-odometry to reduce the influence of moving objects. Finally, the experimental results demonstrate that the proposed method has better performance than the state-of-the-art methods on many challenging sequences from a benchmark dataset, especially on highly dynamic sequences.

Keywords Visual odometry · Dynamic environment · SLAM · Nonparametric statistical · Motion segmentation

1 Introduction

Visual odometry estimation plays a key role in simultaneous localization and mapping (SLAM) systems for navigation of robots and autonomous vehicles [1]. Especially, visual odometry can provide 3D motion estimation for legged robots and aerial vehicles, whereas traditional encoder based wheel odometry can give 2D motion estimations only. Moreover, many robotic applications such as object detection, obstacle avoidance, semantic segmentation and place recognition could benefit from rich visual information, which has advantages for joint vision tasks [2–4].

Recently, many visual odometry methods have emerged and achieved successful results. There are two main groups of visual odometry methods. One is

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s13319-019-0220-4) contains supplementary material, which is available to authorized users.

W. Zhou · X. Peng (✉) · H. Wang · B. Liu
National Space Science Centre, Chinese Academy of Sciences, Beijing 100190, China
e-mail: zhouwugen2006@126.com

X. Peng
e-mail: pxd@nssc.ac.cn

H. Wang
e-mail: wanghaijiao@nssc.ac.cn

B. Liu
e-mail: boliu@nssc.ac.cn

W. Zhou · H. Wang
University of Chinese Academy of Sciences, Beijing 100049, China

Published online: 07 March 2019

feature-based visual odometry [5–7]. These methods estimate 6 degree-of-freedom poses by solving a closed-form optimal problem using matched visual correspondence features from source to target frames. The other is the dense-visual-odometry [8–11] approach, which takes the pose estimation as an energy minimization by using the entire dense intensity or depth difference between target and warped source images.

Generally, dense-visual-odometry is more stable and robust than the visual feature-based method in static environments, especially in low-texture environments [12]. However, the performance of most of the state-of-the-art dense methods deteriorates when dynamic objects appear in the scene because of the assumption of a static world. Usually, the non-static parts in most quasi-static environments, which include small dynamic objects, can be viewed as noise. This noise can be reduced by using some probabilistic methods, such as the random sample consensus (RANSAC) [13] or robust Huber function [14]. Unfortunately, most of these dense methods cannot work correctly if the dynamic parts become significant. Some other dense methods use segmentation based on the cluster-wise [15] or nonparametric [16] model to deal with dynamic scenes. However, they cannot effectively remove dynamic objects and precisely estimate the ego-motion in highly dynamic environments.

In this paper, we propose a dense approach based on the nonparametric statistical model and the clustering model to maintain the robustness of dense-visual-odometry in highly dynamic scenes by using RGD-B data. The main contributions of this work are as follows: (i) the multi-frame-based residual computing model is proposed to achieve temporal consistency motion segmentation, leading to an improvement in the precision of camera motion estimation; and (ii) the clustering model and the nonparametric statistics model are combined to obtain the weighted clusters and then prevent dynamic clusters from pose estimation. The experiments show that our approach outperforms the state-of-the-art dense-visual-odometry methods that are designed to handle dynamic scenes on many sequences from the RGB-D benchmark dataset.

2 Related Work

Currently, the handling of dynamic scenes remains a challenging problem for visual SLAM. Motion segmentation is regarded as a key step to deal with this problem, which finds the key points or pixels on dynamic objects and then removes them from the optimization process.

Kundu et al. [17] proposed a monocular visual SLAM with motion segmentation based on multi-view geometry constraints. The method improves the robustness of pose estimation by filtering feature points that do not conform to constraints on the driving road. However, the assumption is so strong that it limits the application to rigid motion scenes. Tan et al. [18] considered the difference in appearance and structure between key frames and current frames to filter out the effects of dynamic objects, but their method is limited to small scenes. With the assumption that all static background motions are equally likely, scene motions are clustered by using sparse 3D flow [13] or scene flow [19], and then camera motion can be calculated by iterative refinement schemes such as RANSAC. However, these methods may fail when dynamic key points outnumber static key points. Other approaches consider using multiple cameras to explicitly compensate for occlusion of dynamic objects [20], or add an additional IMU sensor [21, 22] to alleviate this problem. In addition, Li et al. [23] proposed a static point weighting method for sparse 3D edge point clouds. Their approach can achieve the state-of-the-art accuracy. However, the methods mentioned above are all based on sparse feature points, and therefore they cannot perform dense motion segmentation and dense mapping thereafter.

Different from these works, our approach is a dense-visual-odometry method. Some related works are discussed here. A solution for the joint estimation of visual odometry and dense scene flow was proposed by Jaimez et al. [15]. They used the background segmentation and energy function optimization to divide the scene into moving, stationary, and intermediate state parts. However, some intermediate state parts deteriorate the method's performance since all clusters are treated equally in the optimization of energy function. Our approach is also related to the work of Kim et al. [16], who proposed to leverage accumulated depth residuals from multiple previous frames to model a static background by the

nonparametric method. The method is based on pixel-wise segmentation and the statistical model for pose estimation, which is susceptible to independent non-rigid body motion of past frames, and therefore some dynamic pixels may be included in the energy function minimization. Meanwhile, Scona et al. [24] proposed StaticFusion. They maintain a static background environment mapping used for pose estimation in a model-to-frame way. However, StaticFusion cannot work on scenes with fast camera motion due to not having enough time for mapping. Finally, Sun et al. [25] proposed a motion removal approach as a pre-processing step and integrated it into the front end of RGB-D SLAM. However, the disadvantage of this method is that they can deal with only one motion, instead of multiple moving objects in dynamic scenes.

Another line of related works are off-line methods. Roussos et al. [26] proposed an approach of multi-body motion segmentation and reconstruction based on the energy function. The algorithm effectively gives the camera pose, scene depth, and 3D reconstruction in dynamic scenes. Unfortunately, the method processes RGB-D data in a batch way and hence can be seen as an off-line system. Wang et al. [27] estimated dense optical flow from frames, where dynamic objects can be excluded by clustering motion patterns based on optical flow. However, due to the large amount of calculations, they could not achieve real-time performance.

Regarding motion segmentation, Stückler et al. [28] proposed an efficient real-time dense motion segmentation, whose weakness is that it is only applicable to rigid body segmentation. Although some unsupervised learning based methods [29–31] were recently proposed and achieved good results, they cannot always perform well in other special dynamic scenes since they need a large dataset for training a network; thus, they suffer from the generalization problem.

3 Dense Visual Odometry Approach

3.1 Overview

In this paper, we proposed a visual odometry approach based on the nonparametric statistical model and the clustering model. The overview is shown in Fig. 1. First, K-means clustering was used to segment each frame into N clusters based on depth and intensity. Each cluster was considered to be a rigid body, and thus the pixel-wise motion segmentation problem was simplified into a cluster-wise segmentation. Second, the initial camera pose was calculated by minimized photometric and depth residuals in a Cauchy M-estimator, and then the estimated poses were used to warp previous frames to the current frame coordinate. After regularization, these warped frames were used to compute temporal consistency residuals for each cluster, which ensured the continuity of clusters' motion. Third, temporal consistent residuals were used to build a nonparametric statistical model based on the t-distribution and to find moving objects by utilizing a dynamic threshold condition. Finally, the probability confidence of each static cluster based on the statistical model was regarded as weight that would be incorporated into the energy function optimization for obtaining a more accurate camera pose estimation. Afterward, the warp function was updated based on a new estimated transformation for the next iteration.

3.2 Preliminaries

Since the RGB-D sensor simultaneously provides a color image and depth image, a pair of frames (I_{k-1}, Z_{k-1}) and (I_k, Z_k) is given as input, where $I(\mathbf{x}) \in \mathbb{R}$ and $Z(\mathbf{x}) \in \mathbb{R}$ represent the intensity and depth, respectively, of pixel $\mathbf{x} = (u, v)^T \in \mathbb{R}^2$. Intensity is converted from the color image (0.299R + 0.587G + 0.114B). In the homogeneous coordinate, given a 3D point $\mathbf{p} = (X_k, Y_k, Z_k, 1)^T$, the projection function and its inverse function between the 3D point and its pixel on the image is as follows:

$$\mathbf{x} = \pi(\mathbf{p}_k) = \left(\frac{X_k f_x}{Z_k} + o_x, \frac{Y_k f_y}{Z_k} + o_y\right) \tag{1}$$

$$\mathbf{p}_k = \pi^{-1}(\mathbf{x}, Z_k) = \left(\frac{u - o_x}{f_x} Z_k, \frac{v - o_y}{f_y} Z_k, Z_k, 1\right) \tag{2}$$

where f_x and f_y are the focal lengths and (o_x, o_y) is the principal point.

As the camera moves, the 3D point $\mathbf{p}$ in the preview frame's camera coordinate can be transformed rigidly to the current frame with the transformation matrix $\mathbf{T}_{k-1}^{k} \in SE(3)$. The new coordinate of the 3D point in the current camera coordinate can be obtained by the following function:

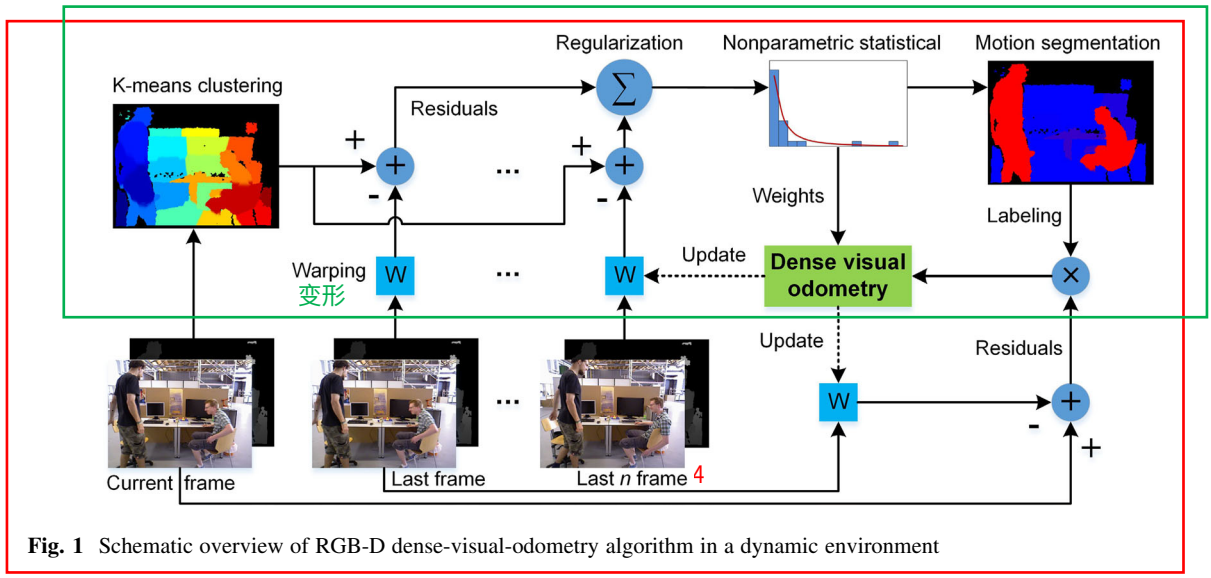

Fig. 1 Schematic overview of RGB-D dense-visual-odometry algorithm in a dynamic environment

$$\mathbf{p}_k = \mathbf{T}_{k-1}^k \mathbf{p}_{k-1} = \begin{bmatrix} \mathbf{R}_{k-1}^k & \mathbf{t}_{k-1}^k \\ 0 & 1 \end{bmatrix} \mathbf{p}_{k-1} \tag{3}$$

where $\mathbf{R}_{k-1}^k \in SO(3)$ and $\mathbf{t}_{k-1}^k \in \mathbb{R}^3$ stand for the rotation matrix and the translation matrix, respectively.

3.3 Scene Clustering

In dynamic scenes, a rigid moving object is an ideal assumption and facilitates fast motion segmentation. However, real-life scenes often include many non-rigid moving objects, such as people. To obtain dense motion segmentation, many current approaches perform pixel-wise segmentation directly. Without having regard for the spatial relationship between pixels, pixel-wise segmentation would have inevitable noise. Therefore, cluster-wise segmentation was used instead of pixel-wise segmentation to achieve robust performance. Since the clustering is prone to local wrong segmentation based on depth information only, intensity-assisted geometry clustering was proposed to alleviate this problem. Compared with the usage of depth only, this clustering could maintain a good level of detail with additional intensity. The distance function is defined as follows:

$$d_i = \alpha_c I(x) + (1 - \alpha_c) Z(x) \tag{4}$$

where α_c is the weight to balance the dimensions of intensity and depth. Then, K-means clustering was performed based on Eq. (4) to implement scene pre-segmentation. Since the number of clusters should be provided in advance in K-means clustering, we set the cluster number to 24 empirically.

The advantages of this clustering method are twofold. (i) Each cluster can be considered as a rigid body. Therefore, the pixel-wise non-rigid scene can be transformed into a cluster-wise rigid scene, which greatly simplifies the motion segmentation. (ii) This method can effectively boost dense motion segmentation and thus potentially support scene flow estimation and obstacle avoidance.

3.4 Residual Model

The traditional background modeling for motion segmentation in surveillance scenarios assumes that the camera is static, but when the camera moves, the static or dynamic objects become hard to distinguish. To identify dynamic objects, the motion segmentation should be performed with regard to the cluster's residuals, which are computed between the warped previous frame and the current frame by using the calculated transformation matrix $\mathbf{T}_{k-n}^k$.

According to the dense-visual-odometry method, the residual between consecutive frames is originally used to estimate camera pose. With taking both depth and intensity into consideration, the residual formulations between the last n frame and the current frame are defined as follows:

$$\boldsymbol{r}_Z^p(\xi) = \boldsymbol{Z}_k\left(\mathcal{W}\left(\boldsymbol{x}_{k-n}^p, \xi_{k-n}^k\right)\right) - \left|\boldsymbol{T}_{k-n}^k \pi^{-1}\left(\boldsymbol{x}_{k-n}^p, \boldsymbol{Z}_{k-n}\left(\boldsymbol{x}_{k-n}^p\right)\right)\right|_Z, \tag{5}$$

$$\boldsymbol{r}_I^p(\xi) = \boldsymbol{I}_k\left(\mathcal{W}\left(\boldsymbol{x}_{k-n}^p, \xi_{k-n}^k\right)\right) - \boldsymbol{I}_{k-n}\left(\boldsymbol{x}_{k-n}^p\right) \tag{6}$$

where $\boldsymbol{r}_Z^p$ and $\boldsymbol{r}_I^p$ represent the residual of pixel depth and intensity, respectively. With the camera moving between the last n frame and the current frame, the corresponding image warping function is given by

$$\mathcal{W}\left(\boldsymbol{x}_{k-n}^p, \xi_{k-n}^k\right) = \pi\left(\boldsymbol{T}_{k-n}^k \pi^{-1}\left(\boldsymbol{x}_{k-n}^p, \boldsymbol{Z}_{k-n}\left(\boldsymbol{x}_{k-n}^p\right)\right)\right). \tag{7}$$

同样对齐的前提下，理论上，背景的残差很低 实际上，背景的残差较高

In theory, if the image alignment is perfect, a cluster that belongs to a static background will have a low residual. In practice, the occlusion area of the background often tends to give a high residual even with better image alignment. To obtain a precise static segmentation related to camera motion, the occlusion area needs to be excluded. Similar to Jaimez's work [15], the cluster residual formulation between the current frame and the previous n frame is calculated as follows:

别人的残差计算方式，curr与前几帧做归一化

$$\boldsymbol{\delta}_{k-n}^{k,i} = \frac{\sum_{p=1}^{\boldsymbol{S}_i - \boldsymbol{O}_i} \alpha_I \boldsymbol{r}_I^p + \boldsymbol{r}_Z^p / \bar{\boldsymbol{Z}}_i}{\boldsymbol{S}_i - \boldsymbol{O}_i} \tag{8}$$

where $\boldsymbol{S}_i$ is the pixel size of cluster i, and $\boldsymbol{O}_i$ is the occluded pixel size of cluster i, which is considered to be with depth residuals that are higher than a threshold. $\bar{\boldsymbol{Z}}_i$ is the cluster's average depth, and α_I is the weight to balance depth and intensity.

遮挡像素

归一化残差很有必要

The calculated cluster residual is rough and often inaccurate. In practice, the performance of scene motion segmentation is always poor if original residuals are used directly. Therefore, residual regularization was performed according to Jaimez's method [15]; the regularized residual had a higher correlation with the motion of dynamic objects.

However, tiny movements are difficult to be detected if only two consecutive frames are taken into consideration since non-rigid dynamic objects move continuously. These tiny motions also slightly deteriorate the camera pose estimation, thus leading to a significantly cumulative error over a long time. Therefore, more previous frames should be considered to achieve the clusters' temporal motion consistency. Here, we consider residual computing between the far time difference n frame and the current frame to obtain temporal consistency.

为了避免微小运动无法检测出来使用了前n帧来进行残差计算

$$\boldsymbol{\delta}_i' = (1 - \alpha_t)\boldsymbol{\delta}_{k-1}'^{k,i} + \alpha_t \boldsymbol{\delta}_{k-n}'^{k,i} \tag{9}$$

where $\boldsymbol{\delta}_{k-1}'^{k,i}$ is computed by two adjacent frames that consider the large-amplitude motion in the scene, and $\boldsymbol{\delta}_{k-n}'^{k,i}$ is computed by setting previous n frame as the target frame, taking small motion into account. These two residuals are both regularized. In our experiments, the number n is set to 4, and the weight item α_t is 0.6.

Considering the residual fusion by multiple frames with larger time difference n, the scene motion maintains better temporal continuity and consistency, and the slightly moving parts become more distinguishable, compared with methods involving two consecutive frames. Therefore, the cumulative error influenced by dynamic objects is reduced.

考虑多帧比考虑2帧具有更好的精确性

3.5 Nonparametric Statistical Model

Generally, the motion of the static background part always has a dual relationship with the camera motion. Therefore, the static clusters' residuals are often small or close to zero since they are aligned perfectly, and residuals of those dynamic clusters are usually large values that deviate significantly from zero due to the independent motion of dynamic objects.

With different scenes, residual distributions are not always the same. The distribution characteristics of cluster residuals should be explored. Figure 2 is an example statistical residual histogram of a highly dynamic scene. Inspired by [14, 23], the

有创造力的

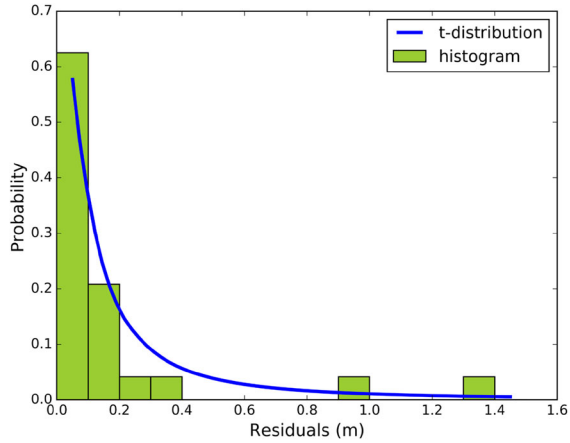

Fig. 2 Histogram of residuals and t-distribution fitting

nonparametric statistical model based on the t-distribution is constructed as follows: t分布

$$\boldsymbol{w}_i = \frac{v_0 + 1}{v_0 + \left((\boldsymbol{\delta}'_i - \mu)/\sigma\right)^2} \tag{10}$$

$$\sigma = 1.4826\ \text{Median}\left\{\left|\boldsymbol{\delta}'_i - \mu\right|\right\} \tag{11}$$

where v_0 is the freedom degree of the t-distribution, which determines the steep degree of the distribution curve. In our experiment, v_0 is empirically set to 10. μ is the statistical mean value, which is set to zero. σ is the statistical variance, which is the nonparametric statistic based on the median absolute deviation.绝对中位差

As shown in Fig. 2, the nonparametric statistical model can fit the histogram of the actual experimental residuals nicely. Furthermore, this model can adaptively fit other low-dynamic or static scene distributions as well. Since the probability of the statistical model represents the cluster's motion likelihood, it can guide scene motion segmentation and provide weights for every cluster to estimate ego-motion.

3.6 Camera Pose Estimation

Unlike other methods that model motion segmentation as a classification problem based on a fixed threshold, we used an adaptive threshold to perform motion segmentation due to the following reasons. (i) During experiments, an appropriate threshold always changes with different residual distributions due to scene diversity. (ii) The optimal threshold is also affected by the velocity of camera motion, and couples with the number of dynamic objects. (iii) Due to the complexity of the scene, the threshold may dramatically change and easily go beyond the normal value, resulting in the failure of motion segmentation. Thus, the adaptive threshold should be truncated.

分割参数adaptive

Based on the reasons mentioned above, the adaptive threshold is calculated by statistical variance, camera velocity, and the number of dynamic objects with truncation. The scene motion segmentation labeling formulation is defined as follows:

$$\mathcal{B}_i = \begin{cases} 1, & \boldsymbol{\delta}'_i < 1/(\min(10,\ \max(3, \alpha_B v_c \sigma))) \\ 0, & \boldsymbol{\delta}'_i \geq 1/(\min(10,\ \max(3, \alpha_B v_c \sigma))) \end{cases} \tag{12}$$

静态 动态 label

where $\mathcal{B}_i$ represents the motion labeling of cluster i; the label value 1 indicates that the cluster belongs to the static background, and the label value 0 indicates that it belongs to dynamic parts. α_B is a coefficient used to adjust the dimension between dependent variables and residuals, and this coefficient is calculated by $10^3 \times \min(10^3, 10^{N_d})$, which takes the number of dynamic clusters into consideration. In this coefficient formulation, N_d is the number of clusters that are marked as moving objects.

Although the statistical model can well fit residual distributions, as well as provide clusters' weights, in static environments or scenes with few dynamic objects, clusters' weights given by the nonparametric model usually suppress some static clusters to participate in the pose estimation, which may decrease pose accuracy. Consequently, the weight model should be refined by taking the number of dynamic objects and residual distribution as distinguishable criteria. If the scene has a large number of dynamic objects with a high level of residuals, then the statistical weight model should be used to reduce the influence of dynamic clusters. Meanwhile in a static scene or quasi-static scene, the weight should be linearly dependent on cluster residuals. Considering the compatibility between high dynamic and low dynamic scenarios, the weight model is built as follows:

weight的几种情况

Nd-number of dynamic

$$\boldsymbol{\omega}_i^p = \begin{cases} \boldsymbol{w}_i, & \text{Median}\ \left(\boldsymbol{\delta}'_i\right) > 0.02 \cup N_d > 5 \\ 1 - \boldsymbol{\delta}'_i, & \text{Others} \end{cases} \tag{13}$$

where $\boldsymbol{\omega}_i^p$ is the weight of cluster i, and Median $\left(\boldsymbol{\delta}'_i\right)$ is the median of cluster residuals.

Finally, we add motion labeling and the weight model into the energy function optimization of dense-visual-odometry. The energy function optimization is constructed based on depth and intensity according to Kerl's work [9] for high stability, and the method of Jaimez's work [15] shows that the optimization process can obtain good results in the Cauchy M-estimator. The ego-motion is estimated by the following equations: 能量函数使用Kerl的，并已被Jaimez验证

$$\xi = \arg\min_{\xi} \left\{ \sum_{m=1}^{M} \mathcal{B}_i [F(\boldsymbol{\omega}_i^p \boldsymbol{r}_Z^p(\xi)) + F(\alpha_I \boldsymbol{\omega}_i^p \boldsymbol{r}_I^p(\xi))] \right\} \tag{14}$$

$$F(r) = \frac{c^2}{2} \log\left(1 + \left(\frac{r}{c}\right)^2\right). \tag{15}$$

In addition, the energy-function-based visual odometry method can converge to the true value only when the motion is small, and large motion often guides the convergence to a local minimum. Thus, we used the pyramid model to solve this optimization problem for obtaining more accurate ego-motion estimation.

此外，基于能量函数的视觉节理方法只有在运动较小的情况下才能收敛到真实值，而大的运动往往引导收敛到局部极小值。
因此，我们利用金字塔模型求解该优化问题，得到更精确的自我运动估计。

4 Experiments

4.1 Experimental Setting

The proposed method was tested on the TUM RGB-D dataset according to the RGB-D SLAM benchmark method [32]. The dataset has many challenging dynamic scenes, and some sequences even contain moving objects covering more than half of the image. In contrast to most previous methods, our approach is more focused on challenging scenes containing two or more moving objects, which often cause large drift.

For the convenient evaluation of the proposed method, these RGB-D sequences were divided into three categories: (i) static scenes, which have no moving objects; (ii) low-dynamic scenes (most scenes of these sequences have only small parts of moving objects with slight motion); and (iii) highly dynamic scenes, which have much more or even more than half of the whole image containing two or more independently moving people or chairs.

The proposed visual odometry method was evaluated and compared with previous dense-visual-odometry methods on these RGB-D scenarios. All of the experiments were performed on a desktop computer with an Intel Core i7-4790 CPU at 3.6 Hz and 15 GB RAM. The proposed algorithm was implemented with C++ language on Ubuntu 16.04. Our visual odometry method used multiple CPU cores for acceleration.

4.2 Motion Segmentation

Motion segmentation is an extremely important pretreatment for visual odometry methods in dynamic scenes. The performance of motion segmentation has direct influence on the accuracy and robustness of pose estimation. Generally, if motion segmentation is insufficient, some dynamic pixels will participate in pose optimization, which may reduce accuracy. In contrast, if too much over-segmentation occurs, the valid number of static pixels will become too small to calculate the camera pose in the pyramid model.

In the experiments, the fast odometry and scene flow estimation (VO-SF) method [15] was used as a comparison algorithm. We used different scenes from low dynamic "*fr3/sitting*" and highly dynamic "*fr3/walking*" to perform motion segmentation. As illustrated in Fig. 3c, d, red parts represent moving objects and blue parts belong to static background. Results show that our method outperformed VO-SF for segmenting micro motion scenes (cases (C) and (D)), one person walking scenes (cases (B) and (G)), and highly dynamic scenes (cases (E) and (F)). This is especially so in case (F), which included dynamic clusters in more than half of the valid depth area. In case (H), the camera is moving fast and the valid depth pixels are less than 50% of the whole image, and thus our method fails to segment the scene properly, leading to large drift in camera pose estimation. Case (A) shows another inappropriate result due to slight over-segmentation that prevents some static pixels from pose estimation.

The improvements on motion segmentation mainly result from the following. (i) The intensity is taken into consideration in the scene segmentation, which helps geometric clustering. (ii) The residual model considers the far time difference n frame, which supports the detection of some small movements. (iii) The labeling threshold is adaptively adjusted with the change in the type of scene.

4.3 Ego-Motion Estimation

To evaluate our method, the relative pose error (RPE) metric is adopted to give a quantitative assessment of the camera pose. We compared our method with three previous approaches: (i) robust dense visual odometry (DVO) [14]; (ii) the background model-based dense-visual-odometry (BaMVO) method [16]; and (iii) the fast visual odometry and scene flow (VO-SF) method [15]. All of these methods are dense-visual-odometry with RGB-D inputs. Among them, DVO is the state-of-the-art method in static or quasi-static environments. Meanwhile, BaMVO and VO-SF are designed to handle a dynamic environment.

As shown in Table 1, the comparison results illustrates that our method outperformed other methods in most highly dynamic sequences. Especially in

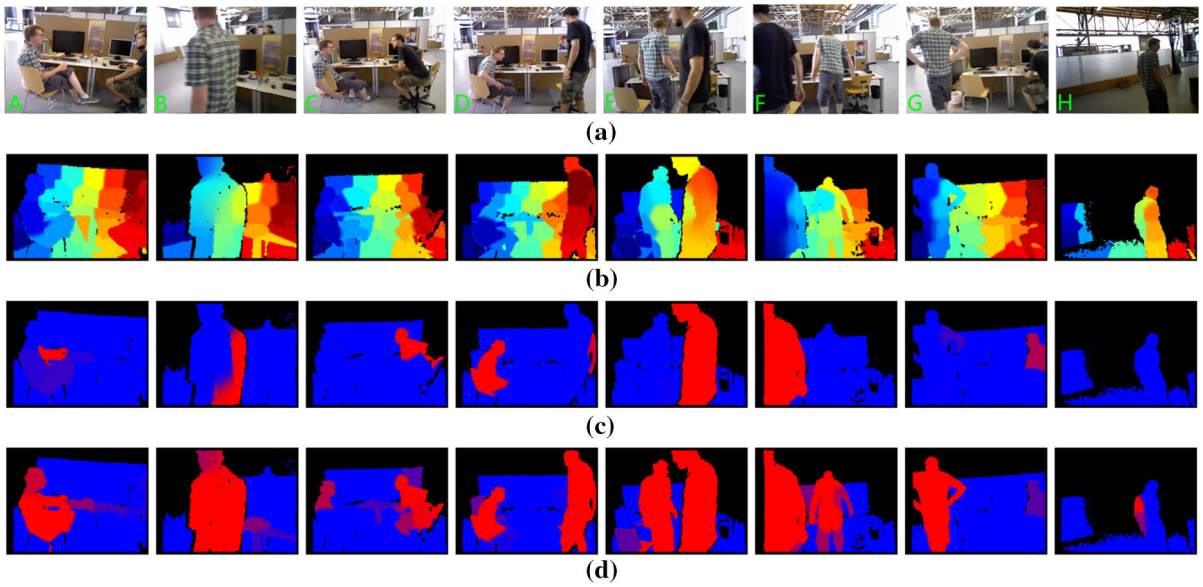

Fig. 3 Examples of motion segmentation results. **a** Original input RGB data; **b** K-mean clustering results; **c** motion segmentation results of VO-SF; **d** motion segmentation results of our method. (Color figure online)

Table 1 RPE results of dense-visual-odometry methods on TUM RGB-D dataset

Sequences		RMSE of translational drift (m/s)				RMSE of rotational drift (deg/s)			
		DVO	BaMVO	VO-SF	**Ours**	DVO	BaMVO	VO-SF	**Ours**
Static	*fr2/desk*	0.0296	0.0299	0.0291	**0.0249**	1.3920	1.1167	1.1743	**1.0557**
	fr3/long-office	**0.0231**	0.0332	0.0341	0.0394	1.5689	2.1583	**1.1968**	1.5259
Low dynamic	*fr2/desk-with-person*	0.0354	**0.0352**	0.0368	0.0432	1.5683	**1.2159**	1.2354	1.3834
	fr3/sitting-static	**0.0157**	0.0248	0.0242	0.0257	**0.6084**	0.6977	0.7069	0.7324
	fr3/sitting-xyz	**0.0453**	0.0482	0.0566	0.0650	1.4980	**1.3885**	1.4319	1.6000
	fr3/sitting-rpy	0.1735	0.1872	0.1086	**0.1052**	6.0164	5.9834	**2.9141**	3.0110
	fr3/sitting-halfsphere	0.1005	**0.0589**	0.0798	0.0997	4.6490	**2.8804**	3.0011	3.8675
Highly dynamic	*fr3/walking-static*	0.3818	0.1339	0.1110	**0.0460**	6.3502	2.0833	1.8300	**0.9555**
	fr3/walking-xyz	0.4360	0.2326	0.3040	**0.1270**	7.6669	4.3911	5.6900	**2.6152**
	fr3/walking-rpy	0.4038	0.3584	0.3956	**0.2690**	7.0662	6.3398	6.5010	**4.8014**
	fr3/walking-halfsphere	0.4120	**0.1738**	0.3410	0.2736	7.2200	**4.2863**	6.7700	5.6182

Bold means the best results that our method and other compared approaches could achieve

the first three sequences, our method improved the translation by an average of 45% and the rotation by an average of 40% compared with BaMVO, and 50% and 43%, respectively, compared with VO-SF. In static scenes, our algorithm detected mismatched pixels caused by occlusions, and had comparable results or a slight advantage with the other methods. In low-dynamic scenes, which are close to static scenes most of the time, our method achieved results comparable to those of VO-SF, but performed slightly poorer than BaMVO. The possible reason for this is that fewer static pixels are used for ego-motion estimation due to over-segmentation in low-dynamic scenes, therefore causing a slightly poor accuracy. We will further investigate this problem in future work.

The performance of our method can also be proved by a comparison of estimated trajectories. As shown in Fig. 4, trajectory errors are obviously reduced on

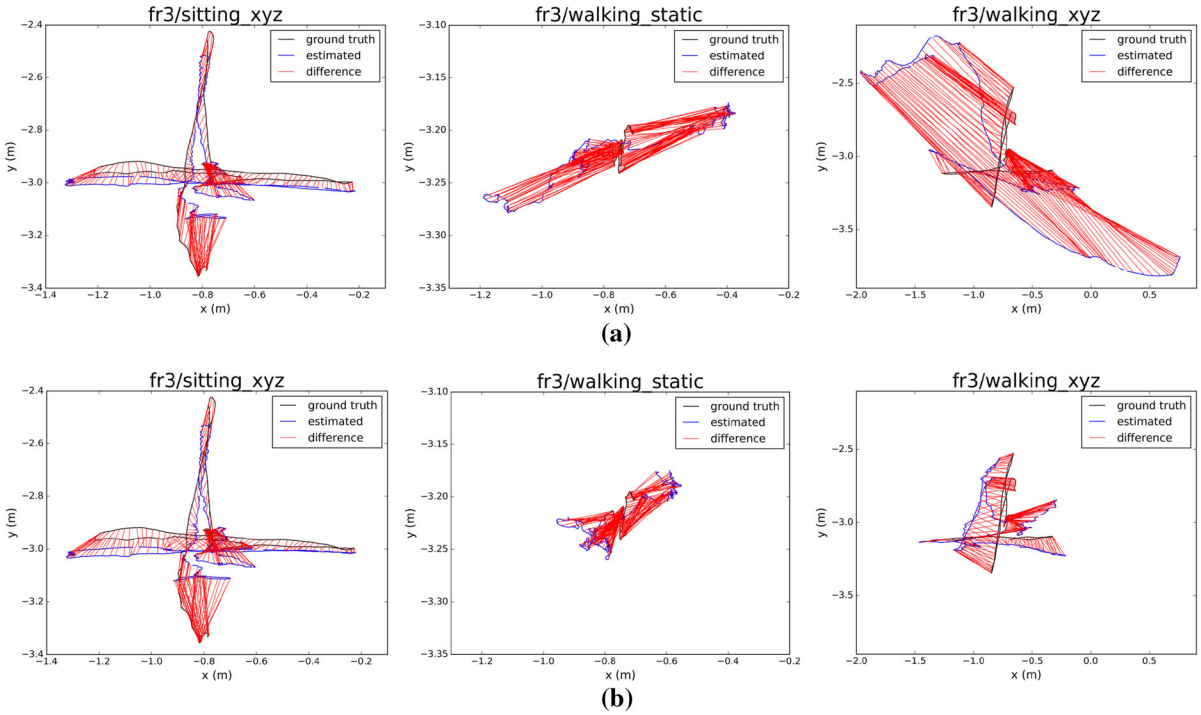

Fig. 4 Examples of estimated trajectories. **a** Results of VO-SF; **b** results of the proposed method

highly dynamic sequences by our method, but our result is slightly poorer than that of VO-SF in low-dynamic scenes.

The improvements are mainly based on the following aspects. (i) Clustering based on depth and intensity can provide better scene segmentation, which ensures more distinguishable motion for moving object details. (ii) Considering the constraint of temporal consistency of moving objects, most dynamic objects can be removed. (iii) Clusters' weights are given based on the statistical model, which decreases the weight of high-level motion likelihood clusters and increases the importance of static clusters in energy function optimization for ego-motion estimation.

The DVO method is designed for static scenes, which cannot obtain good results in dynamic sequences. Meanwhile, the BaMVO method only considers depth information to perform pixel-wise motion segmentation, and cannot remove motion pixels perfectly, whereas VO-SF takes only two consecutive frames into consideration for residual calculations, which is not good when detecting more moving parts. Conversely, our method takes the multi-frame strategy, and combines the nonparametric model and clustering model to achieve robustness in highly dynamic scenes. This is also highlighted by the results presented in Fig. 5, which illustrates a comparison of the relative pose errors of DVO, VO-SF and our method on the "*fr3/walking-static*" sequence from the TUM RGB-D dataset. As shown in the figure, our method was less influenced by the moving person than DVO and VO-SF.

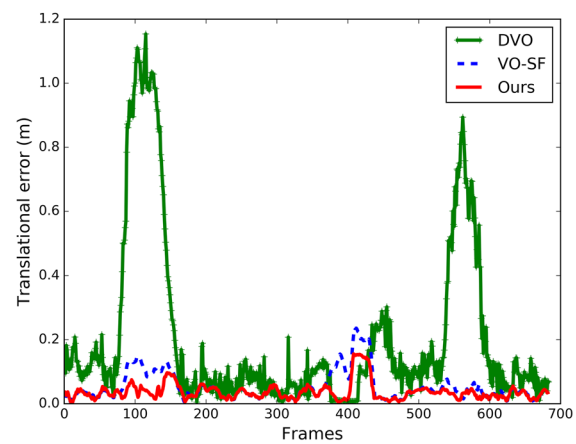

Fig. 5 Translational error of RPE comparison on "*fr3/walking-static*" sequence

To make a running time comparison, we used the same environment with the BaMVO method. The proposed method and other contrasting algorithms all performed on the QVGA image (320 × 240). The average computation time of our method was 57.5 ms per frame. In comparison, DVO required an average running time of 35.4 ms per frame, BaMVO required 43.8 ms per frame, and VO-SF required 44.5 ms per frame. Our method utilized a multi-frame strategy that included another warping and regularization process. Thus, the running time was relatively longer than that of the other methods. Due to depth dependence of our method, it could achieve about 45 ms per frame in "*fr3/walking*" sequences which include quite a bit of invalid depth pixels.

5 Conclusion

In this paper, a dense-visual-odometry method by using RGB-D data was proposed to handle highly dynamic environments. The proposed approach employed the multi-frame strategy to keep the consistency of motion segmentation. Afterward, a robust cluster weight model was proposed based on nonparametric statistics model and the clustering model to prevent highly dynamic clusters from the energy-function-based camera motion estimation. Results showed that our method can achieve better performance than the state-of-the-art dense-visual-odometry method BaMVO on most highly dynamic sequences. Compared with BaMVO in terms of translational and rotational drift error, our method improved the accuracy by an average of 45% and 40%, respectively, in the three challenging "*fr3/walking*" sequences from the TUM RGB-D dataset. These results demonstrated the robustness of our method in handling highly dynamic scenes.

In future work, we plan to introduce a deep learning-based motion segmentation method to boost the accuracy of pose estimation in a dynamic environment. Moreover, the proposed visual odometry method also needs loop closure to reduce the cumulative error of visual odometry, as well as static environment mapping to construct a dense visual SLAM system that can work in dynamic scenes.

Acknowledgements This work was supported by Pre-Research Foundation under Grant 6140001010216ZK24001 funded by Chinese Equipment Development Department.

References

1. Yousif, K., Bab-Hadiashar, A., & Hoseinnezhad, R. (2015). An overview to visual odometry and visual SLAM: Applications to mobile robotics. *Intelligent Industrial Systems, 1*(4), 289–311.
2. Fioraio N & Stefano LD (2013) Joint detection, tracking and mapping by semantic bundle adjustment. In *IEEE conference on computer vision and pattern recognition* (pp. 1538–1545).
3. Reddy ND, Singhal P et al. (2015) Dynamic body VSLAM with semantic constraints. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 1897–1904).
4. Mousavian A, Kosecka J & Lien JM (2015) Semantically guided location recognition for outdoors scenes. In *IEEE international conference on robotics and automation* (pp. 4882–4889).
5. Klein G & Murray D (2007) Parallel tracking and mapping for small AR workspaces. In *IEEE and ACM international symposium on mixed and augmented reality* (pp. 1–10).
6. Mur-Artal, R., Montiel, J. M. M., & Tardós, J. D. (2015). ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics, 31*(5), 1147–1163.
7. Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(6), 1052–1067.
8. Newcombe RA, Lovegrove SJ & Davison AJ (2011) DTAM: Dense tracking and mapping in real-time. In *IEEE international conference on computer vision* (pp. 2320–2327).
9. Kerl C, Sturm J & Cremers D (2014) Dense visual SLAM for RGB-D cameras. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 2100–2106).
10. Steinbrücker F, Sturm J & Cremers D (2011) Real-time visual odometry from dense RGB-D images. In *IEEE international conference on computer vision workshops* (pp. 719–722).
11. Engel J, Schöps T & Cremers D (2014) LSD-SLAM: Large-scale direct monocular SLAM. In *European conference on computer vision* (pp. 834–849).
12. Taketomi, T., Uchiyama, H., & Ikeda, S. (2017). Visual SLAM algorithms: A survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications, 9*(1), 16.
13. Wang Y & Huang S (2014) Motion segmentation based robust RGB-D SLAM. In *World congress on intelligent control and automation* (pp. 3122–3127).
14. Whelan T, Johannsson H et al. (2013) Robust real-time visual odometry for dense RGB-D mapping. In *IEEE international conference on robotics and automation* (pp. 5724–5731).
15. Jaimez M, Kerl C, Gonzalez-Jimenez J & Cremers D (2017) Fast odometry and scene flow from RGB-D cameras based

on geometric clustering. In *IEEE international conference on robotics and automation* (pp. 3992–3999).
16. Kim, D. H., & Kim, J. H. (2017). Effective background model-based RGB-D dense visual odometry in a dynamic environment. *IEEE Transactions on Robotics, 32*(6), 1565–1573.
17. Kundu A, Krishna KM & Jawahar CV (2011) Realtime multibody visual SLAM with a smoothly moving monocular camera. In *IEEE international conference on computer vision* (pp. 2080–2087).
18. Tan W, Liu H et al. (2013) Robust monocular SLAM in dynamic environments. In *IEEE international symposium on mixed and augmented reality* (pp. 209–218).
19. Alcantarilla PF, Yebes JJ, Almazán J & Bergasa LM (2012) On combining visual SLAM and dense scene flow to increase the robustness of localization and mapping in dynamic environments. In *IEEE international conference on robotics and automation* (pp. 1290–1297).
20. Zou, D., & Tan, P. (2013). CoSLAM: Collaborative visual SLAM in dynamic environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(2), 354–366.
21. Leung TS & Medioni G (2014) Visual navigation aid for the blind in dynamic environments. In *IEEE conference on computer vision and pattern recognition workshops* (pp. 579–586).
22. Kim DH, Han SB & Kim JH (2015) Visual odometry algorithm using an RGB-D sensor and IMU in a highly dynamic environment. In *Robot intelligence technology and applications 3*, *advances in intelligent systems and computing*, (vol. 345, pp. 11–26).
23. Li, S., & Lee, D. (2017). RGB-D SLAM in dynamic environments using static point weighting. *IEEE Robotics & Automation Letters, 2*(4), 2263–2270.
24. Scona R, Jaimez M et al. (2018) StaticFusion: Background reconstruction for dense RGB-D SLAM in dynamic environments. In *IEEE international conference on robotics and automation* (pp. 3849–3856).
25. Sun, Y., Liu, M., & Meng, Q. H. (2017). Improving RGB-D SLAM in dynamic environments: A motion removal approach. *Robotics & Autonomous Systems, 89,* 110–122.
26. Roussos A, Russell C, Garg R & Agapito L (2012) Dense multibody motion estimation and reconstruction from a handheld camera. In *IEEE international symposium on mixed and augmented reality* (pp. 31–40).
27. Wang Y & Huang S (2014) Towards dense moving object segmentation based robust dense RGB-D SLAM in dynamic scenarios. In *International conference on control automation robotics & vision* (pp. 1841–1846).
28. Stückler, J., & Behnke, S. (2015). Efficient dense rigid-body motion segmentation and estimation in RGB-D video. *International Journal of Computer Vision, 113*(3), 233–245.
29. Ranjan A, Jampani V et al. (2018) Adversarial collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. *arXiv preprint* arXiv:1805.09806.
30. Kruger, B., Vogele, A., et al. (2017). Efficient unsupervised temporal segmentation of motion data. *IEEE Transactions on Multimedia, 19*(4), 797–812.
31. Ho, H. W., Wagter, C. D., Remes, B. D. W., & Croon, G. C. H. E. D. (2015). Optical-flow based self-supervised learning of obstacle appearance applied to MAV landing. *Robotics & Autonomous Systems, 100,* 78–94.
32. Sturm J, Engelhard N et al. (2012) A benchmark for the evaluation of RGB-D SLAM systems. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 573–580).