2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Continuous Scale-Space Direct Image Alignment for Visual Odometry from RGB-D Images

Yassine Ahmine[1,2], Guillaume Caron[2,3], Fatima Chouireb[1], and El Mustapha Mouaddib[2]

***Abstract*— In this paper, we propose a novel dense 3D image alignment algorithm that estimates the Euclidean transformation between pairs of camera poses from pixel intensities. The novelty consists in the automatic scale adaptation within each level of a multi-resolution image pyramid, using the scale-space representation of images. This is done through the continuous optimization of a scale parameter along with camera pose parameters in the same optimization framework. The proposed approach permits to significantly improve the robustness of the direct image alignment to large inter-frame motion. Various experiments on the TUM RGB-D dataset show that the proposed algorithm outperforms a fixed scale pyramid-based state-of-the-art alignment method.**

I. INTRODUCTION

VO (Visual Odometry) and SLAM (Simultaneous Localization and Mapping) techniques typically consider a probabilistic model that takes as input noisy measurements Z (image stream) and as output an estimation of the model parameters X (pose of the camera and position of map points). The objective is consequently to find the model parameters that maximize the probability $P(Z|X)$ of observing the measurements. These techniques can be divided into indirect vs direct methods and sparse vs dense methods.

Indirect methods tackle the estimation problem in a two-step scheme. The first step consists in a feature detection/matching step, which provides geometric clues that are used in the second step to estimate the model parameters. Direct methods [1], [2], on the other hand, use directly the pixel intensities provided by the camera, in order to estimate the parameters of the probabilistic model, generally by considering a least square approach. Sparse methods [3] for their part use a subset of selected image pixels (most commonly corners), for the estimation process. Conversely, dense methods try to make use of all the image pixels in the probabilistic framework.

Historically, sparse indirect methods were the first VO and SLAM solutions to be developed, principally because of the limited computation capabilities of the available platforms at that time. Examples of this type of solutions are the system proposed by Klein and Murray [4] and the one proposed by Mur-Artal *et al.* [5]. Other methods that considered a dense direct approach can be found in the literature, such as the work of Kerl *et al.* [6] and the work of Engel *et al.* [7]. Forster *et al.* [8] developed a semi-direct system based on a KLT tracker that tried to take the best of both worlds. In the category of sparse direct methods, one can consider the algorithm proposed by Engel *et al.* [9]. The latter coupled the optimization of the geometric parameters with the optimization of photometric parameters that model the changes in the scene illumination.

VO and SLAM algorithms can also be classified according to the sensor they consider. Pumarola *et al.* [10] considered RGB images as inputs to propose a SLAM system that used

Fig. 1: The considered algorithms aim at aligning the image pair $\{I_1, I_2\}$, where the images resulting from the alignment are $G_{I_2}(\mathcal{W}, \lambda)$ and $I_2(\mathcal{W})$. Here the fixed scale pyramid-based method was unable to align the image pair because it fell in a local minimum, while the proposed optimized scale pyramid-based method succeeded because it is able to automatically adapt the image smoothing to suppress local minima.

[1]Yassine Ahmine and Fatima Chouireb are with Laboratoire de Télécommunication, Signaux et Systèmes, Université Amar Telidji de Laghouat, Algeria yassine.ahmine@etud.u-picardie.fr, f.chouireb@lagh-univ.dz

[2]Yassine Ahmine, Guillaume Caron, and El Mustapha Mouaddib are with Laboratoire de Modélisation, Information et Systèmes, Université de Picardie Jules Verne, France mouaddib@u-picardie.fr

[2]Guillaume Caron is with CNRS/AIST Joint Robotics Laboratory UMI 3218 / RL, Tsukuba, Japan guillaume.caron@u-picardie.fr

*This work was partly supported by the PROFAS B+ scholarship

lines and point features. In their work, Bryner *et al*. [11] considered an event camera for the 3D tracking of the camera using a 3D model of the scene. Other works considered RGB-D images, such as the work of Le *et al*. [12], which proposed a dense piece-wise planar algorithm robust to low-textured scenes.

In their paper, Schöps *et al*. [13] proposed the BAD-SLAM algorithm and presented a benchmark for the evaluation of SLAM solutions. They showed that dense approaches, e.g. the quadrifocal VO [14] or the DVO (Dense Visual Odometry) algorithm [15], performed better, using their benchmark, than indirect SLAM methods (ORB-SLAM2) [16]. The fundamental building block of the DVO algorithm is the frame to frame tracker, which estimates the 3D Euclidean transformation between two successive frames. As presented in Steinbruecker *et al*. [17], the performance of this tracker drops when the displacement between frames is too important.

This paper aims at overcoming the latter drawback by proposing a new alignment method more robust to large inter-frame motion. We introduce a coarse-to-fine alignment using the continuous scale-space representation of images [18] as opposed to discrete scale levels of multi-resolution pyramids [19]. The main contribution is to optimize the scale parameter along with the camera pose at each resolution level of the pyramid to enlarge the convergence domain of the direct alignment while being precise. The approach is similar to [20] except that, first, we consider the full scale 6 degrees of freedom camera pose instead of the projective motion of planar objects and, second, we consider the multi-resolution pyramid for computational efficiency. Another work to consider a continuous formulation of the RGB-D image alignment is Continuous Visual Odometry (CVO) [21]. Exploiting an Hilbert space modeling, CVO increases the estimation precision whereas this paper increases the robustness of the direct alignment to large displacements.

The remaining of the paper is organized as follows: first, the method integrating the image scale-space representation to the 3D pose estimation algorithm will be described. Then, a presentation of the results and a discussion about the performance of the proposed algorithm will be presented. Finally, a conclusion about this work and a discussion about its perspectives will be discussed.

II. Scale-Space Dense Alignment

We consider that our algorithm has at each time-stamp t two undistorted input images (I_t, I_{t-1}) and the corresponding depth images (D_t, D_{t-1}). The objective is to estimate the relative camera 3D motion (rigid body transformation) between two consecutive frames, while considering the scale-space representation of the input images. This is done through the joint optimization of the scale and motion parameters within each level of a multi-resolution image pyramid. Fig. 1 shows an example where our method was able to successfully align a pair of images that exhibit a large displacement.

A. Notation

The following notation is used throughout this paper. Scalars are represented using light letters s. Bold lower-case notation $\mathbf{v}$ indicates a vector. When representing matrices, we use bold upper-case $\mathbf{M}$, and images using light upper-case I. The rigid body transformation is written $\mathbf{T} \in SE(3)$, and with a slight abuse of notation we use the twist vector $\boldsymbol{\xi} \in \mathbb{R}^6$ to represent its associated tangent space $\mathfrak{se}(3)$.

B. Camera Model

$\mathbf{p} = (x, y, z, 1)^T$ is a 3D point in the camera coordinate frame represented using homogeneous coordinates. The projection $\mathbf{x} = (u, v)^T$ of this point in the image coordinate frame is modeled using a pinhole camera model:

$$\mathbf{x} = \Pi(\mathbf{p}) = \begin{pmatrix} f_u \frac{x}{z} + c_u \\ f_v \frac{y}{z} + c_v \end{pmatrix}, \tag{1}$$

where (f_u, f_v) represent the focal lengths and (c_u, c_v) the coordinates of the principal point of the camera all in pixel units. The inverse projection function $\Pi^{-1}(.)$ permits to reconstruct the 3D point $\mathbf{p}$ from its projection $\mathbf{x}$ and the corresponding depth $z = D(\mathbf{x})$:

$$\mathbf{p} = \Pi^{-1}(\mathbf{x}, z) = \begin{pmatrix} z\frac{u-c_u}{f_u} \\ z\frac{v-c_v}{f_v} \\ z \end{pmatrix}. \tag{2}$$

C. Warping Function

The warping function permits to transform a pixel location $\mathbf{x}$ from the first image I_1 to the second image I_2. Therefore, $\mathbf{x}$ and the corresponding depth $D_1(\mathbf{x})$ are first used to reconstruct a 3D point $\mathbf{p}$ (in the coordinate frame of the first camera) through the inverse projection function $\Pi^{-1}(\mathbf{x}, D_1(\mathbf{x}))$. The reconstructed 3D point is then transformed to the coordinate frame of the second camera by applying to it the rigid body transformation $\mathbf{T}$:

$$\mathbf{p}' = \mathbf{T}.\mathbf{p} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}, \tag{3}$$

where $\mathbf{R} \in SO(3)$ is the rotation matrix and $\mathbf{t} \in \mathbb{R}^3$ is the translation vector. In order to have a minimal parameterization of $\mathbf{T}$, we use twist coordinates $\boldsymbol{\xi}$:

$$\boldsymbol{\xi} = \begin{pmatrix} \mathbf{v}^T & \mathbf{w}^T \end{pmatrix}^T = \begin{pmatrix} v_1 & v_2 & v_3 & w_1 & w_2 & w_3 \end{pmatrix}^T, \tag{4}$$

where $\mathbf{v}$ and $\mathbf{w}$ are the 3-vectors of linear and angular velocities respectively. The matrix exponential is used to relate the twist coordinates $\boldsymbol{\xi}$ with the transformation matrix $\mathbf{T}$:

$$\mathbf{T} = g(\boldsymbol{\xi}) = \exp(\hat{\boldsymbol{\xi}}), \tag{5}$$

where $\hat{\boldsymbol{\xi}} = \begin{pmatrix} \hat{\mathbf{w}} & \mathbf{v} \\ 0 & 0 \end{pmatrix} \in \mathfrak{se}(3)$ and the hat operator $\hat{(.)}$ is defined as follows [22]:

$$\hat{\mathbf{w}} = \begin{pmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{pmatrix} \tag{6}$$

The $\boxplus$ operator for pose composition can be consequently defined, as follows:

$$\boldsymbol{\xi} \boxplus \mathbf{T} = \exp(\hat{\boldsymbol{\xi}}).\mathbf{T}. \tag{7}$$

The transformed point $\mathbf{p}'$ is then projected in the second image by applying the projection function $\Pi(\mathbf{p}')$. The warping function $\mathcal{W}(\mathbf{x}, \boldsymbol{\xi})$ can hence be written as follows:

$$\mathbf{x}' = \mathcal{W}(\mathbf{x}, \boldsymbol{\xi}) = \Pi\left(g(\boldsymbol{\xi}).\Pi^{-1}(\mathbf{x}, D_1(\mathbf{x}))\right). \tag{8}$$

D. Probabilistic Optimization

Considering the photo-consistency assumption and independent and identically distributed measurement noise, the maximum likelihood estimator for the estimation problem of $\boldsymbol{\xi}$ can be expressed in the following manner:

$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}}{\operatorname{argmin}} \sum_i w_i r_i(\boldsymbol{\xi})^2, \tag{9}$$

where w_i and $r_i(\boldsymbol{\xi}) = I_2(\mathcal{W}(\mathbf{x}_i, \boldsymbol{\xi})) - I_1(\mathbf{x}_i)$ are respectively the weight and the residual of the ith pixel of the image. Since we assume a Gaussian likelihood of the residual $p(r_i) \propto exp(r_i^2/\sigma^2)$ where σ is the standard deviation of the Gaussian, w_i is constant [6]. After linearization, this results in the following expression to minimize:

$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}}{\operatorname{argmin}} \sum_i \left[r_i(0) + J_i \Delta\boldsymbol{\xi}\right]^2, \tag{10}$$

where $J_i = \frac{\partial r_i(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}}$ is the Jacobian matrix of the ith pixel. Considering the Gauss-Newton optimization method, the following expression of the twist increments can be deduced:

$$\Delta\boldsymbol{\xi} = -H^{-1} \sum_i J_i^T r_i, \tag{11}$$

where $H = \sum_i J_i^T J_i$ represents the approximation of the Hessian matrix of the ith pixel. Then, the geometric parameters can be updated as follows:

$$\mathbf{T}^{\textbf{new}} \leftarrow \Delta\boldsymbol{\xi} \boxplus \mathbf{T} \tag{12}$$

E. Scale-Space Image Alignment

Instead of estimating the parameters $\boldsymbol{\xi}$ using the images $I_1(\mathcal{W}(\mathbf{x}, \boldsymbol{\xi}))$ and $I_2(\mathbf{x})$, we propose to use a scale-space representation [18] of the images $G_{I_2}(\mathcal{W}(\mathbf{x}, \boldsymbol{\xi}); \lambda)$ and $G_{I_1}(\mathbf{x}; \lambda_{ref})$:

$$G_{I_2}(\mathcal{W}(\mathbf{x}, \boldsymbol{\xi}); \lambda) = I_2(\mathcal{W}(\mathbf{x}, \boldsymbol{\xi})) * g(\mathbf{x}; \lambda), \tag{13}$$

and

$$G_{I_1}(\mathbf{x}; \lambda_{ref}) = I_1(\mathbf{x}) * g(\mathbf{x}; \lambda_{ref}), \tag{14}$$

where $*$ represents the product of convolution, $g(\mathbf{x}; \lambda)$ is a Gaussian kernel (computed for a width of $2\,\mathrm{ceil}(2\,\lambda)+1$), and λ and λ_{ref} are the scale-space parameters (standard deviation of the Gaussian kernel in pixels) of I_2 and I_1 respectively. The advantage of such a representation is that we can directly control the degree of smoothing applied to the images. In an optimization scheme, it permits to automatically go from a coarse representation to a fine one and vice versa. From equations 13 and 14 the new cost is expressed as:

$$G_{r_i}(\boldsymbol{\xi}) = G_{I_2}(\mathcal{W}(\mathbf{x}, \boldsymbol{\xi}); \lambda) - G_{I_1}(\mathbf{x}; \lambda_{ref}). \tag{15}$$

Then, by stacking the geometric parameters and the scale parameter $\tilde{\boldsymbol{\xi}} = (\boldsymbol{\xi}^T, \lambda)^T$, and after the linearization of Eq. 15, we obtain the following expression:

$$\tilde{\boldsymbol{\xi}}^* = \underset{\tilde{\boldsymbol{\xi}}}{\operatorname{argmin}} \sum_i \left[G_{r_i}(0) + \tilde{J}_i \Delta\tilde{\boldsymbol{\xi}}\right]^2, \tag{16}$$

where $G_{r_i}(0) = G_{I_2}(\mathcal{W}(\mathbf{x}, 0); \lambda) - G_{I_1}(\mathbf{x}; \lambda_{ref})$ and the augmented Jacobian of the ith pixel is computed as follows:

$$\tilde{J}_i = \left[\frac{\partial G_{I_2}}{\partial \xi}, \frac{\partial G_{I_2}}{\partial \lambda}\right]. \tag{17}$$

where $\frac{\partial G_{I_2}}{\partial \lambda}$ is computed using finite differences as in [20]. This results in the following increment expression:

$$\Delta\tilde{\boldsymbol{\xi}} = \begin{pmatrix} \Delta\boldsymbol{\xi} \\ \Delta\lambda \end{pmatrix} = -\tilde{H}^{-1} \sum_i \tilde{J}_i^{\,T} G_{r_i}, \tag{18}$$

where $\tilde{H} = \sum_i \tilde{J}_i^{\,T} \tilde{J}_i$. This results in the following update rule for the scale parameter:

$$\lambda^{\textbf{new}} \leftarrow \lambda + \Delta\lambda. \tag{19}$$

The geometric parameters are updated as in Eq. 12.

F. Parameters Initialization

Considering the use of pyramids we set the value of λ_{ref} for each level to 1, except for the finest level where $\lambda_{ref} = 0.1$ (to gain precision). The value of λ is initialized to a greater value in order to obtain a coarse-to-fine approach [20], we found in the experiments that $\lambda_{init} = 3$ gives good results. During the optimization, the algorithm switches to the next level, when the convergence criteria or the maximum number of 40 iterations is attained. Furthermore, the total number of used pyramid levels is 4, as [23]. During all the experiments, the same set of parameters was used.

III. Results

In this section, the proposed algorithm is compared to the algorithm proposed by [17] implemented with image pyramids to deal with large displacements, which represents the basic elements of state of the art SLAM systems such as [6]. We refer to this approach as fixed scale Pyramidal Photometric-Based (PP-B), while our approach is referred to as Optimized scale pyramid Photometric-Based (OP-B). The results of the various experiments that were conducted are presented and discussed. We considered for the validation the

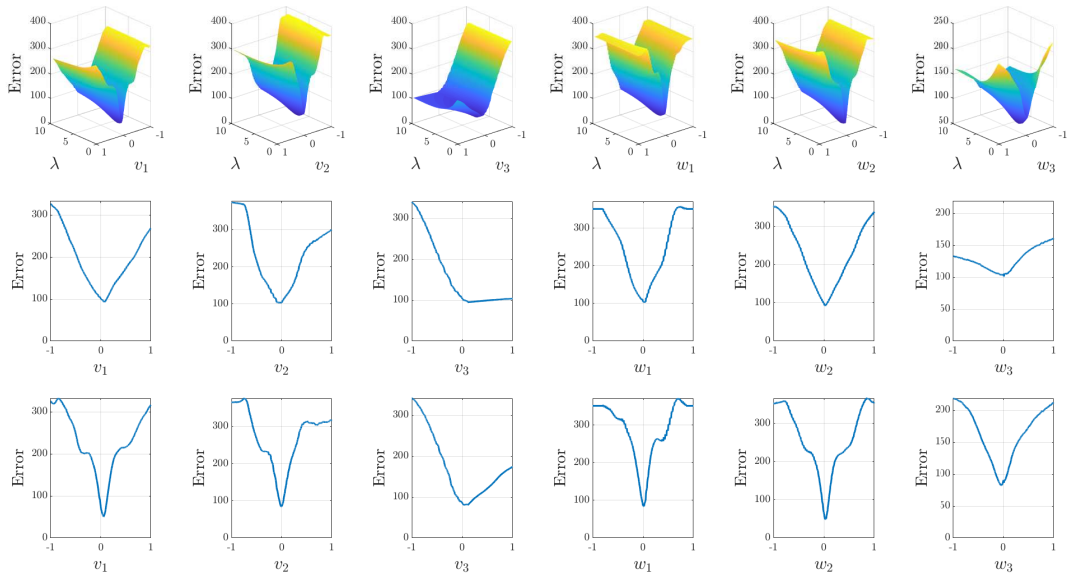

Fig. 2: Top: visualization of the cost function for each geometric DoF combined with the scale-space DoF for the coarsest pyramid level. Middle: section view with $\lambda = 5$ and $\lambda_{ref} = 1$. Bottom: section view for $\lambda = 0.1$ and $\lambda_{ref} = 1$ (equivalent to the fixed scale case).

dataset of TUM [24], which provides sequences of RGB-D images from a Microsoft Kinect and the corresponding ground-truth data.

The maximum number of iterations considered for the PP-B method is set to 40 iterations, while we used an image pyramid with an additional level in comparison to the OP-B to reach 5 levels.

In the first part of the evaluation, we present qualitative results about the proposed method. We show and discuss the effect of the scale parameter on the shape of the cost function, then we present the evolution of λ during the optimization process.

The second part presents quantitative results about the comparison between the proposed and the PP-B method through the experiments that were conducted using different sequences (fr1/room, fr1/desk, and fr1/floor). It is worth noting that the mean execution time of the proposed method is 1.2 times the mean execution time of the PP-B method.

A. *Cost Function Qualitative Analysis*

The first line of Fig. 2 presents a 3D representation of the cost function for each geometric Degree Of Freedom (DoF) combined with the scale-space parameter for the coarsest level of the multi-resolution pyramid. The values of the scale parameter range from 0.1 to 10, and the values of each geometric parameter range from -1.0 to 1.0, while the others are fixed to their reference value. λ_{ref} is set to 1. The middle and bottom lines of Fig. 2 show a section view for two values of the scale parameter $\lambda = \{5, 0.1\}$. The bottom line of Fig. 2 is equivalent to the usual fixed scale within a pyramid level.

Fig. 2 shows that the scale parameter has a smoothing

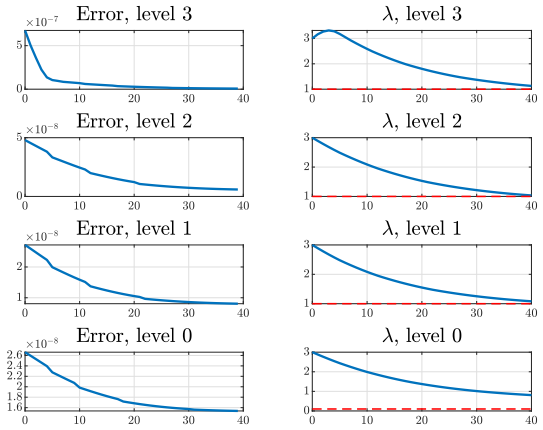

Fig. 3: Evolution of the normalized photometric error during the optimization (for each pyramid level) and the corresponding scale parameter. The blue line represents the value of λ at each iteration (x-axis) and the red dashed line represents λ_{ref}.

effect on the cost function, i.e. for large values of this parameter the local minima of the cost function are suppressed, inducing an enlargement of the basin of convergence around the global minimum. On the other hand, when the value of the scale parameter is small, which is equivalent to the purely photometric case, the global minimum is sharp but the local minima are not suppressed. It is worth noting that the shape of the cost function in this case corresponds to

the coarsest level of the PP-B that is not enough to suppress local minima. During optimization, we can see that the scale parameter (λ) first increases as shown in the pyramid level 3 of Fig. 3, which shows that the proposed method can increase the value of the scale parameter if its initial value is too low. After that, λ continuously tends towards the reference (λ_{ref}) in each pyramid level resulting in a coarse to fine approach. An interesting aspect of the proposed method is that λ_{ref} acts as an implicit objective value for λ that permits the automatic adaptation of the scale within each pyramid level.

B. Quantitative Results

In this part, we present a quantitative comparison between the proposed OP-B and PP-B methods. We consider for the evaluation the freiburg1/room, freiburg1/desk, freiburg2/desk, and freiburg1/floor sequences. The fr1/room, fr1/desk, and fr2/desk were recorded in an office environment and contain translational and rotational motion with varying speeds. The fr1/floor is a sequence containing images of a sweep over a wooden floor.

*1) **Validation in Office Environments**:* First, to evaluate the drift of the algorithms, the RPE (relative pose error) per second as in [15] was computed, based on the ground truth data, using the validation tools proposed by [24]. Tables I, II, and III present the statistics of the RPE for the fr1/room, fr1/desk, and the fr2/desk sequences respectively.

TABLE I: Statistics of the RPE of the proposed and PP-B methods on the fr1/room sequence.

	rms (m/s)	mean (m/s)	median (m/s)
OP-B	**0.4348**	**0.3767**	0.3502
PP-B	0.4351	0.3768	**0.3495**

TABLE II: Statistics of the RPE of the proposed and PP-B methods on the fr1/desk sequence.

	rms (m/s)	mean (m/s)	median (m/s)
OP-B	**0.5531**	**0.4987**	0.5139
PP-B	0.5545	0.4991	**0.5136**

TABLE III: Statistics of the RPE of the proposed and PP-B methods on the fr2/desk sequence.

	rms (m/s)	mean (m/s)	median (m/s)
OP-B	**0.2698**	**0.2327**	**0.2174**
PP-B	0.2718	0.2338	0.2178

We can see that both methods have similar error statistics in terms of rms, mean, and median for the different sequences. This is consistent, since the proposed approach provides enlarged basin of convergence and does not aim at augmenting the precision of the estimation in the nominal case of the PP-B.

In order to simulate larger displacements and study the behavior of each algorithm in such conditions, we followed the methodology used in [17]. Hence, we estimate the motion between pairs of images $I(n)$ and $I(n+k)$ for $k = 1, 2, 3, 4$ (image steps). Example images are shown in Fig. 4. The evolution of the mean and the rms errors of the RPE w.r.t. image steps for the fr2/desk, fr1/desk, and fr1/room sequences is shown in tables IV, V, and VI respectively.

As can be seen in Table IV the difference between the methods is not significant for the fr2/desk. This is due to the fact that the camera movement in this sequence is slow, especially in comparison with the fr1/room and fr1/desk sequences. On both latter sequences, methods show similar results in terms of accuracy when the image steps are small ($k \leq 2$). However, when the displacement is more important ($k > 2$), our method have less drift than the PP-B. Indeed, the rms of the RPE for the proposed method is smaller by around 4 cm and 6 cm for the fr1/room and fr1/desk respectively.

TABLE IV: The rms and mean values of the RPE (m/s) for the proposed and the PP-B methods in the fr2/desk sequence.

		Every image	1 out of 2	1 out of 3	1 out of 4
rms	OP-B	**0.26975**	**0.27401**	**0.27587**	**0.26971**
	PP-B	0.27179	0.27702	0.27908	0.27305
mean	OP-B	**0.2327**	**0.23547**	**0.2366**	**0.23046**
	PP-B	0.23381	0.2371	0.2381	0.23194

TABLE V: The rms and mean values of the RPE (m/s) for the proposed and the PP-B methods in the fr1/desk sequence.

		Every image	1 out of 2	1 out of 3	1 out of 4
rms	OP-B	**0.55309**	**0.55362**	**0.55529**	**0.54373**
	PP-B	0.55453	0.55652	0.57628	0.60778
mean	OP-B	**0.49873**	**0.50028**	**0.50059**	**0.4887**
	PP-B	0.49911	0.50195	0.51479	0.53708

TABLE VI: The rms and mean values of the RPE (m/s) for the proposed and the PP-B methods in the fr1/room sequence.

		Every image	1 out of 2	1 out of 3	1 out of 4
rms	OP-B	**0.43476**	**0.43424**	**0.42911**	**0.48324**
	PP-B	0.43506	0.43441	0.44336	0.52348
mean	OP-B	**0.37666**	**0.37627**	**0.37223**	**0.39377**
	PP-B	0.3768	0.37629	0.38479	0.42346

In order to evaluate the consistency of the estimated trajectories w.r.t. the ground-truth data, we computed the Absolute Trajectory Error (ATE) shown in tables VII, VIII, and IX in addition to figures 8, 9, and 10. Furthermore, the estimated trajectories along with the ground-truth trajectory are presented in figures 8, 9, and 10. We can see from these results that the proposed method is able to maintain a better level of consistency (ATE) compared to the photometric-based method for image steps of 4, which are equivalent to a mean displacement of 4.0 cm and 4.9 cm for the fr/room and fr1/desk sequences respectively.

TABLE VII: The rms values of the ATE (cm) for the proposed and PP-B methods in the fr1/room sequence.

	Every image	1 out of 2	1 out of 3	1 out of 4
OP-B	36.4620	37.2552	**37.8434**	**66.1154**
PP-B	**36.1006**	**36.8521**	54.8574	90.3661

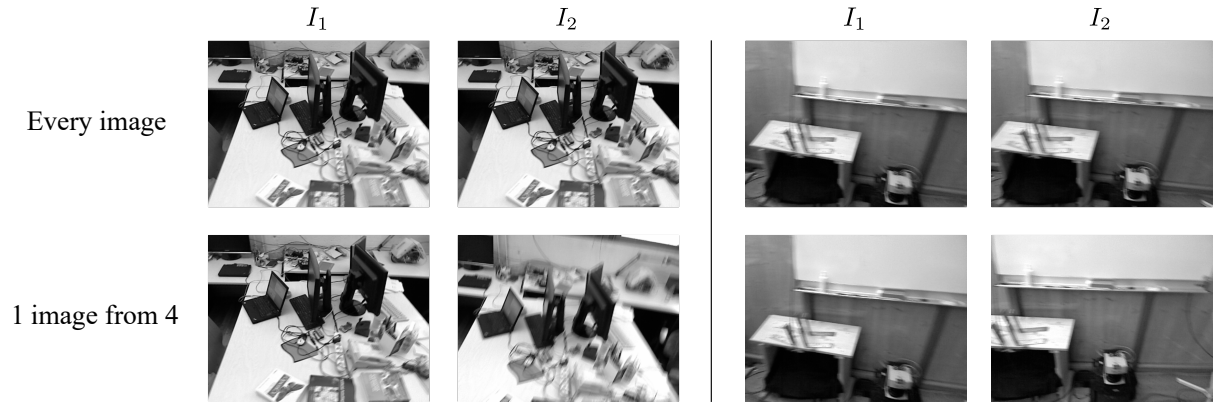

Fig. 4: Example image pairs for different images steps from the fr1/desk and fr1/room sequences.

TABLE VIII: The rms values of the ATE (cm) for the proposed and PP-B methods in the fr1/desk sequence.

	Every image	1 out of 2	1 out of 3	1 out of 4
OP-B	10.4134	10.9150	**11.0837**	**13.0052**
PP-B	**9.9718**	**10.1357**	35.7624	56.2765

TABLE IX: The rms values of the ATE (cm) for the proposed and PP-B methods in the fr2/desk sequence.

	Every image	1 out of 2	1 out of 3	1 out of 4
OP-B	59.105	58.609	58.295	58.76
PP-B	**58.593**	**58.017**	**57.582**	**57.73**

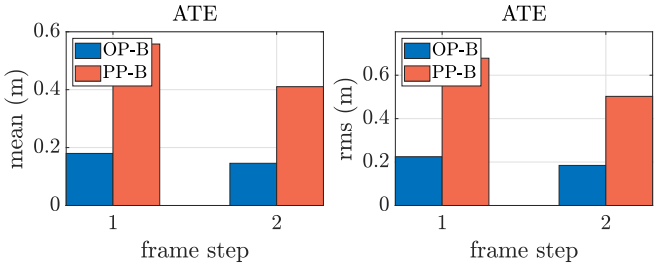

Fig. 5: Evolution of the mean and the rms of the ATE in meters w.r.t. image steps for the fr1/floor sequence.

When the image step is small ($k \leq 2$), the the difference in the ATE obtained from the estimates of the PP-B and the OP-B is not significant. This slight difference results from the fact that the scale parameter of the input images does not always attain its reference, notably due to the blur present in the image pairs. However, when the image step is important ($k > 2$) the error increases more for the pyramidal photometric-based than the proposed method. The latter shows, thus, more consistency of the estimates w.r.t. the ground-truth trajectory. Thereby, the proposed method is more accurate and more consistent than the PP-B one, for large displacements.

*2) **Fr1/Floor Sequence**:* In this part we consider the fr1/floor sequence, which is a challenging sequence (with rapid camera movement and a lot of reflections on the wooden floor) that the PP-B method was unable to estimate even at frame rate. Fig. 5 presents the evolution of the mean and rms values of the ATE w.r.t. image steps. The estimated trajectories and their corresponding errors are shown in Figure 11. We can see that the proposed method is able to estimate a trajectory consistent with the ground-truth at frame rate and considering 1 out of 2 images. Both methods failed, when considering more image steps.

*3) **Robustness to Camera Velocity**:* To evaluate the impact of camera velocity changes on the alignment precision, we focus on the fr1/desk sequence as it features faster velocities than the others. Figures 6 shows the ground truth velocity norm and acceleration along with the inter-frame RPE, for 1 image step. One may note that, contrary to Section III-B.1, we do not consider the RPE in m/s in order to emphasize the impact of camera velocity variations that last less than 1 second. Fig. 6 shows OP-B and PP-B still reach the same results. Both experience the maximum error near 5.5s where the high camera velocity and its variations lead to both large motion blur and important camera displacement between successive images.

In order to evaluate the behavior of both methods for important displacements (high velocities) we follow the approach of [25]. We accordingly simulated fast camera motions by time-decimating the input data of the sequence (using an image step of 4 images). The corresponding simulated velocity norm and acceleration along with the inter-frame RPE are shown in Fig. 7. We can see that the drift of both methods increases with respect to a step of 1 image. However, the drift of PP-B can locally reach an error of 0.6m near 5.5s (Fig. 7, (c)), about 6 times the error of OP-B at the same time. In comparison on the entire sequence, OP-B maximum error is below 0.2 m.

IV. Conclusion and Future Works

In this paper, we presented a novel 3D image alignment algorithm that takes advantage of the scale-space representation of the input images. The proposed method permits to jointly optimize in the same framework, the geometric parameters (of the rigid body motion) and the scale parameter (of the scale-space representation of the image). Throughout the various experiments that were conducted, we saw that the state-of-art method that uses pyramids with fixed scale

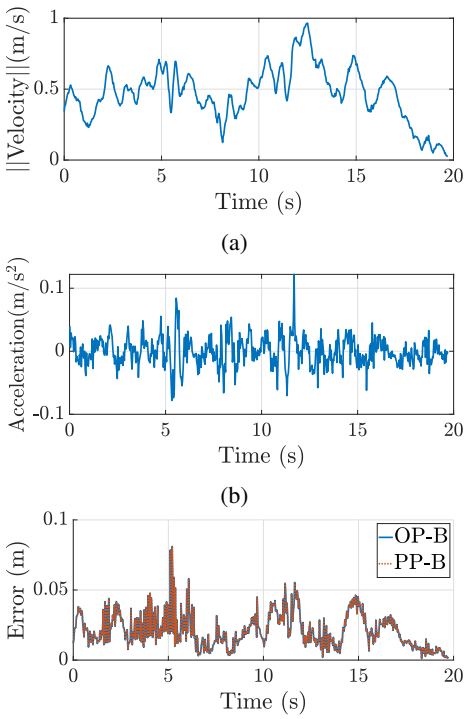

Fig. 6: Evolution of the camera velocity (a), camera acceleration (b), and estimation error (c) according to time for 1 image step on the fr1/desk sequence.

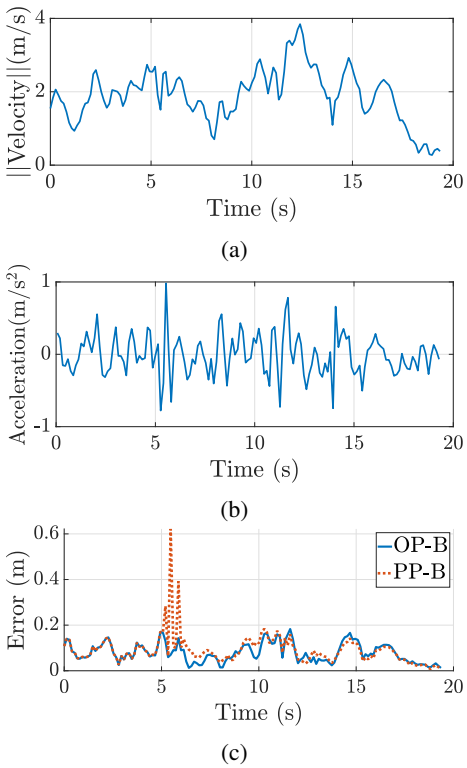

Fig. 7: Evolution of the camera velocity (a), camera acceleration (b), and estimation error (c) according to time for 4 image steps on the fr1/desk sequence.

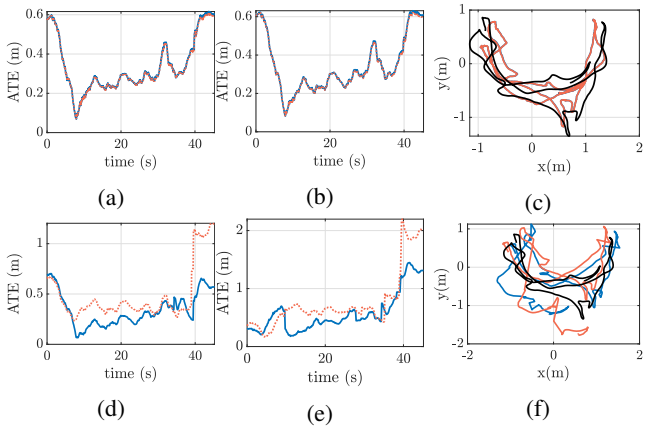

Fig. 8: Visualization of the absolute errors of each trajectory, for the fr1/room sequence, considering the different image steps: 1 (a), 2 (b), 3 (d), and 4 image steps (e). The estimated trajectories along with ground-truth (on the xy plane) for 1 and 4 image steps are shown in (c) and (f) respectively. The PP-B and the new OP-B methods are, respectively, in red and blue. The ground-truth trajectory is in black.

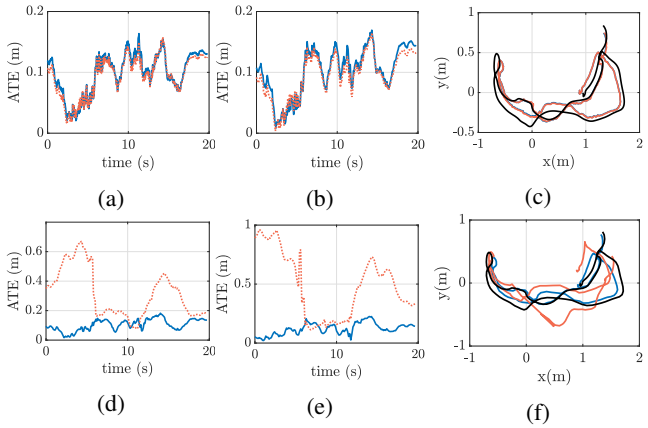

Fig. 9: Visualization of the absolute errors of each trajectory, for the fr1/desk sequence, considering the different image steps: 1 (a), 2 (b), 3 (d), and 4 image steps (e). The estimated trajectories along with ground-truth (on the xy plane) for 1 and 4 image steps are shown in (c) and (f) respectively. The PP-B and the new OP-B methods are, respectively, in red and blue. The ground-truth trajectory is in black.

to handle large motions were outperformed by the proposed method, when considering large camera displacements. This shows the relevance of the proposed image alignment algorithm to handle important camera displacements

Various perspectives arise from the results of this work, notably the consideration of robust estimators and pose graph optimization to increase the estimation precision of the proposed method. This would permit to build a complete SLAM system, as [6]. Moreover, area-based resampling methods represent an attractive solution to build a fully continuous pyramid-based alignment, instead of common discrete pyramids considered in this work.

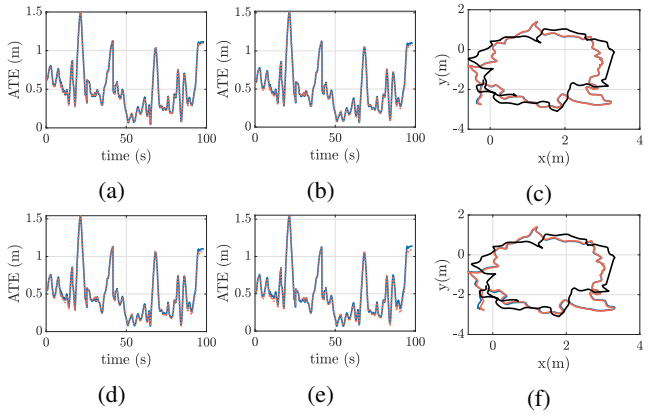

Fig. 10: Visualization of the absolute errors of each trajectory, for the fr2/desk sequence, considering the different image steps: 1 (a), 2 (b), 3 (d), and 4 image steps (e). The estimated trajectories along with ground-truth (on the xy plane) for 1 and 4 image steps are shown in (c) and (f) respectively. The PP-B and the new OP-B methods are, respectively, in red and blue. The ground-truth trajectory is in black.

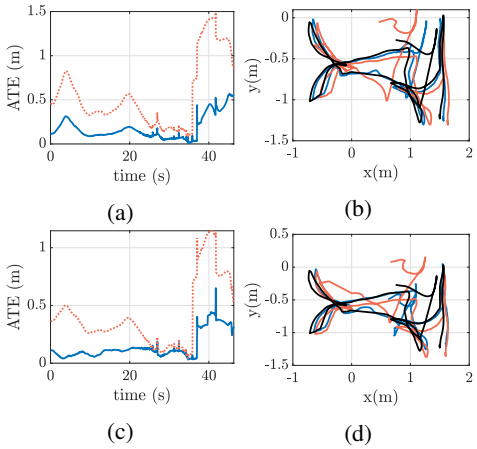

Fig. 11: Visualization of the absolute errors of each trajectory for the fr1/floor sequence considering every image (a) and 1 out 2 images (c). The estimated trajectories along with ground-truth (on the xy plane) for 1 and 2 image steps are shown in (b) and (d) respectively. The PP-B and the new OP-B methods are, respectively, in red and blue. The ground-truth trajectory is in black.

REFERENCES

[1] G. Silveira, E. Malis, and P. Rives, "An Efficient Direct Approach to Visual SLAM," *IEEE Transactions on Robotics*, no. 5, pp. 969–979, oct 2008.

[2] A. Concha, G. Loianno, V. Kumar, and J. Civera, "Visual-inertial direct slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 1331–1338.

[3] X. Gao, R. Wang, N. Demmel, and D. Cremers, "Ldso: Direct sparse odometry with loop closure," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 2198–2204.

[4] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, Nov 2007, pp. 225–234.

[5] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "Orb-slam: A versatile and accurate monocular slam system," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, Oct 2015.

[6] C. Kerl, J. Sturm, and D. Cremers, "Dense visual slam for rgb-d cameras," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 2100–2106.

[7] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European Conference on Computer Vision – ECCV 2014*. Springer International Publishing, 2014, pp. 834–849.

[8] C. Forster, M. Pizzoli, and D. Scaramuzza, "Svo: Fast semi-direct monocular visual odometry," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 15–22.

[9] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 611–625, March 2018.

[10] A. Pumarola, A. Vakhitov, A. Agudo, A. Sanfeliu, and F. Moreno-Noguer, "Pl-slam: Real-time monocular visual slam with points and lines," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 4503–4508.

[11] S. Bryner, G. Gallego, H. Rebecq, and D. Scaramuzza, "Event-based, direct camera tracking from a photometric 3d map using nonlinear optimization," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 325–331.

[12] P. Le and J. Košecka, "Dense piecewise planar rgb-d slam for indoor environments," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 4944–4949.

[13] T. Schöps, T. Sattler, and M. Pollefeys, "Bad slam: Bundle adjusted direct rgb-d slam," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 134–144.

[14] A. Comport, E. Malis, and P. Rives, "Real-time quadrifocal visual odometry," *The International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 245–266, 2010.

[15] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for rgb-d cameras," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 3748–3754.

[16] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[17] F. Steinbruecker, J. Sturm, and D. Cremers, "Real-time visual odometry from dense rgb-d images," in *Workshop on Live Dense Reconstruction with Moving Cameras at the Intl. Conf. on Computer Vision (ICCV)*, 2011.

[18] T. Lindeberg, *Scale Selection*. Boston, MA: Springer US, 2014, pp. 701–713.

[19] J.-Y. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker," *Intel Corporation, Microprocessor Research Labs*, 2000.

[20] Y. Ahmine, G. Caron, E. Mouaddib, and F. Chouireb, "Adaptive lucas-kanade tracking," *Image and Vision Computing*, vol. 88, pp. 1 – 8, 2019.

[21] M. G. Jadidi, W. Clark, A. Bloch, R. Eustice, and J. W. Grizzle, "Continuous direct sparse visual odometry from rgb-d images," in *Proceedings of Robotics: Science and Systems*, June 2019.

[22] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[23] K. M. Han and Y. J. Kim, "Robust rgb-d camera tracking using optimal key-frame selection," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 6275–6281.

[24] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *Proc. of the International Conference on Intelligent Robot Systems (IROS)*, Oct. 2012.

[25] M. Jaimez and J. Gonzalez-Jimenez, "Fast visual odometry for 3-d range sensors," *IEEE Transactions on Robotics*, vol. 31, no. 4, pp. 809–822, 2015.