Large-Scale Direct SLAM with Stereo Cameras

Jakob Engel, Jörg Stückler, Daniel Cremers

***Abstract*— We propose a novel Large-Scale Direct SLAM algorithm for stereo cameras (Stereo LSD-SLAM) that runs in real-time at high frame rate on standard CPUs. In contrast to sparse interest-point based methods, our approach aligns images directly based on the photoconsistency of all high-contrast pixels, including corners, edges and high texture areas. It concurrently estimates the depth at these pixels from two types of stereo cues: Static stereo through the fixed-baseline stereo camera setup as well as temporal multi-view stereo exploiting the camera motion. By incorporating both disparity sources, our algorithm can even estimate depth of pixels that are under-constrained when only using fixed-baseline stereo. Using a fixed baseline, on the other hand, avoids scale-drift that typically occurs in pure monocular SLAM. We furthermore propose a robust approach to enforce illumination invariance, capable of handling aggressive brightness changes between frames – greatly improving the performance in realistic settings. In experiments, we demonstrate state-of-the-art results on stereo SLAM benchmarks such as Kitti or challenging datasets from the EuRoC Challenge 3 for micro aerial vehicles.**

I. INTRODUCTION

Visual simultaneous localization and mapping (SLAM) under real-time constraints has traditionally been tackled using sparse interest points, since they reduce the large amount of pixels in images to a small amount of features. Only recently, real-time capable direct methods have been proposed that avoid the reliance on interest points, but instead perform image alignment and 3D reconstruction directly on pixels using photoconsistency constraints. The premise of direct approaches over interest-point based methods is that image information can be used densely. No manual design of interest point detectors, descriptors, and matching procedures is required, which would also restrict the SLAM algorithm to a specific type of feature – typically only image corners are used. Instead in direct SLAM methods, a rich set of pixels contributes to depth estimation and mapping.

In this paper, we propose the first large-scale direct visual SLAM approach for stereo cameras that is real-time capable on CPUs. Our method estimates depth with uncertainty estimates at pixels with high intensity gradient, reconstructing a semi-dense depth map online. It concurrently tracks the rigid-body motion through photometric alignment of images based on the depth maps.

In our previous work on large-scale direct monocular SLAM (LSD-SLAM), we obtain depth in keyframes by pixel-wise stereo between the current and the keyframe.

This work has been partially supported by grant CR 250/9-2 (Mapping on Demand) of German Research Foundation (DFG) and grant 16SV6394 (AuRoRoll) of BMBF.

J. Engel, J. Stückler and D. Cremers are with the Department of Computer Science, Technical University of Munich, Germany {engelj,stueckle,cremers}@in.tum.de

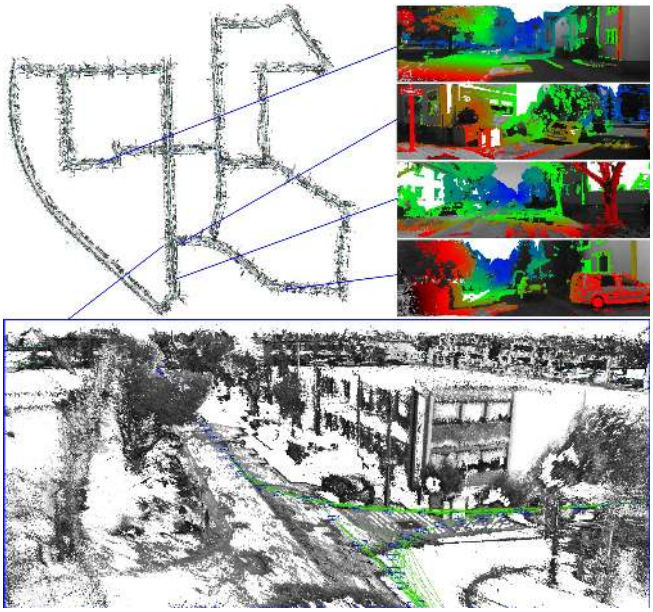


Fig. 1. Stereo LSD-SLAM is a fully direct SLAM method for stereo cameras. It runs at 30Hz on a CPU, computing accurate camera movement as well as semi-dense probabilistic depth maps. We exploit both static and temporal stereo and correct for affine lightning changes, making the method both accurate and robust in real-world scenarios. Some examples are shown in the attached video.

Camera motion is tracked towards a keyframe through photometric image alignment. For SLAM on the global scale, keyframes are aligned towards each other and their poses are optimized by graph optimization. Since reconstruction scale is not observable in monocular SLAM, we additionally optimize for the scale in direct image alignment as well as in pose graph optimization.

In this work, we couple *temporal stereo* of monocular LSD-SLAM with *static stereo* from a fixed-baseline stereo camera setup. At each pixel, our Stereo LSD-SLAM method integrates static as well as temporal stereo cues into the estimate depending on availability. This combines the properties of monocular structure from motion with fixed-baseline stereo depth estimation in a single SLAM method. While static stereo effectively removes scale as a free parameter, temporal stereo cues allow for estimating the depth from baselines beyond the small baseline of the stereo camera. Temporal stereo is not restricted to one specific (e.g. horizontal) direction like static stereo. Rather its baseline corresponds to the translational motion between frames. We furthermore propose a method for handling illumination changes in direct image alignment which significantly improves the robustness of our algorithm in realistic settings.

We evaluate Stereo LSD-SLAM on the popular Kitti

benchmark and datasets from the EuRoC Challenge 3 for micro aerial vehicles (MAVs), demonstrating the state-of-the-art performance of our approach.

II. RELATED WORK

Sparse interest-point-based approaches to visual odometry and SLAM have been extensively investigated in recent years. The term visual odometry has been coined in the seminal work of Nister et al. [1] who proposed sparse methods for estimating the motion of monocular as well as stereo cameras by sequential frame-to-frame matching. Chiuso et al. [2] proposed one of the first real-time capable monocular SLAM methods based on non-linear filtering. Davison [3] proposed MonoSLAM, a real-time capable, EKF-based method that demonstrated SLAM in small workspaces. Sparse interest points are tracked in an EKF-SLAM formulation in order to recover camera motion and the (global) 3D position of the interest points. Another example of sparse monocular SLAM is Parallel Tracking and Mapping (PTAM [4]) which separates and parallelizes optimization for tracking and mapping in a bundle adjustment framework. More recently, Strasdat et al. [5] included scale as a parameter in a key-frame-based optimization approach to sparse monocular SLAM.

Using a fixed-baseline stereo camera setup, scale becomes directly observable. One early work applies EKF-SLAM on a sparse set of interest points [6]. Paz et al. [7] combine monocular stereo cues with fixed-baseline stereo in a sparse hierarchical EKF-SLAM framework.

Direct methods that avoid the detection of sparse interest points have recently attracted attention for visual SLAM. One major advantage of direct over sparse methods is that they do not rely on manually designed image features which constrain the type of information that can be used in subsequent processing stages. In the RGB-D domain [8], [9], [10], direct methods have become the state-of-the-art for their high accuracy and efficiency. LSD-SLAM [11] has been the first large-scale direct monocular SLAM method. In LSD-SLAM, camera motion is tracked towards keyframes for which semi-dense depth maps are estimated using probabilistic filtering. Pose graph optimization aligns the keyframes in a globally consistent arrangement. LSD-SLAM explicity considers scale drift in pose graph optimization and finds a single consistent scale. For stereo cameras, a direct visual odometry approach has been proposed by Comport et al. [12]. Their approach does not explicitly recover depth, but uses quadrifocal constraints on pixels which are in stereo correspondence for camera motion estimation. In the direct stereo method in [13], a disparity map is integrated over time, while the motion of the stereo camera is tracked through direct image alignment using the estimated depth. The keyframes in our approach also integrate depth, while we employ probabilistic filtering instead. Our approach combines fixed-baseline stereo cues from the static camera setup with temporal stereo from varying baselines caused by the moving camera. We combine this with a pose-graph-based SLAM system that globally optimizes the poses of the keyframes. A further important contribution of our work is the correction for affine lighting changes to enable direct image alignment in realistic settings. Differently to previous methods [14], [15], we optimize for affine lighting correction parameters in an alternating fashion, which allows for different outlier rejections schemes to be applied in image alignment and lighting correction.

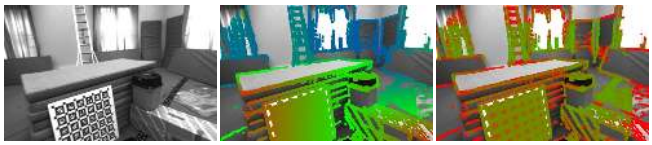


Fig. 3. Each keyframe maintains a Gaussian probability distribution on the inverse depth for all pixels that have sufficient image gradient such that the depth can be estimated. From left to right: Intensity image, semi-dense inverse depth map, inverse depth variance map.

III. LSD-SLAM WITH STEREO CAMERAS

LSD-SLAM [11] is a key-frame based localization and mapping approach which uses the following main steps:

- The motion of the camera is tracked towards a reference keyframe in the map. New keyframes are generated if the camera moved too far from existing keyframes in the map.
- Depth in the current reference keyframe is estimated from stereo correspondences based on the tracked motion (temporal stereo).
- The poses of the keyframes are made globally consistent by mutual direct image alignment and pose graph optimization.

In Stereo LSD-SLAM, the depth in keyframes is in addition directly estimated from static stereo (see Fig. 2). There is a number of advantages of this approach to relying solely on temporal or solely on static stereo. Static stereo allows for estimating the absolute scale of the world and is independent of the camera movement. However, static stereo is constrained to a constant baseline (with, in many cases, a fixed direction), which effectively limits the performance to a specific range. Temporal stereo does not limit the performance to a specific range as demonstrated in [11]. The same sensor can be used in very small and very large environments, and seamlessly transits between the two. On the other hand, it does not provide scale and requires non-degenerate camera movement. An additional benefit of combining temporal and static stereo is, that multiple baseline directions are available: while static stereo typically has a horizontal baseline – which does not allow for estimating depth along horizontal edges, temporal stereo allows for completing the depth map by providing other motion directions.

In detail, we make the following key contributions:

- We generalize LSD-SLAM to stereo cameras, combining temporal and static stereo in a direct, real-time capable SLAM method.
- We explicitly model illumination changes during direct image alignment, thereby making the method highly robust even in challenging real-world conditions.
- We perform a systematic evaluation on two benchmark

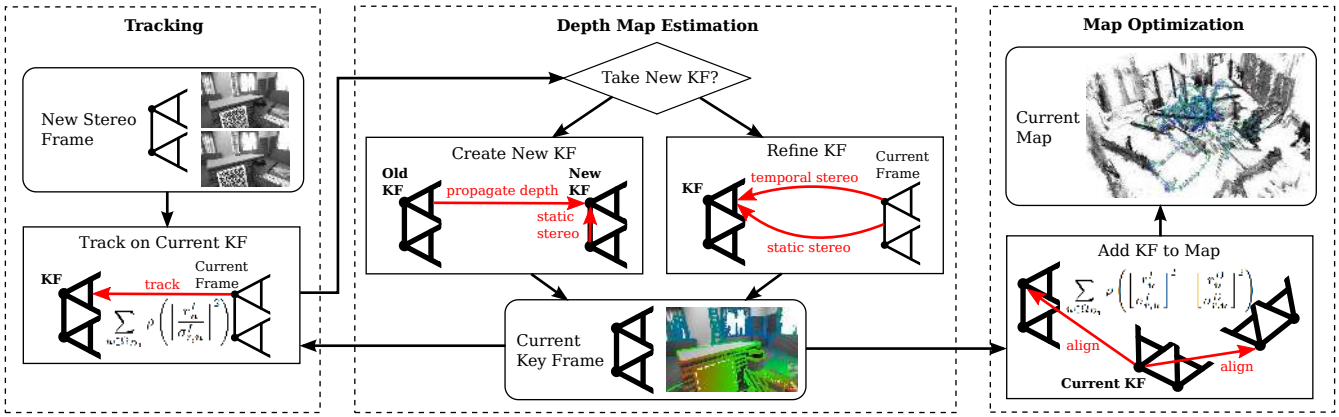


Fig. 2. Overview on the Stereo LSD-SLAM system.

datasets from realistic robotics applications, demonstrating the state-of-the-art performance of our approach.

A. Notation

We use bold capital letters for matrices (such as $\mathbf{R}$) and bold lower case letter for vectors (such as $\boldsymbol{\xi}$). The operator $[\cdot]_n$ selects the n-th row of a matrix. Throughout the paper we use d to denote the *inverse* of the depth z of a point, i.e., $d = z^{-1}$.

In Stereo LSD-SLAM, a map is maintained as a set of keyframes $\mathcal{K}_i = \{I_i^l, I_i^r, D_i, V_i\}$. Each keyframe consists of the left and right image $I_i^{l/r} \colon \Omega \to \mathbb{R}$ of the stereo camera, an inverse depth map $D_i \colon \Omega_{D_i} \to \mathbb{R}^+$ and its variance map $V_i \colon \Omega_{D_i} \to \mathbb{R}^+$. Depth and variance are only maintained for one of the images in the stereo pair, we always use the left image as reference frame. We assume the image domain $\Omega \subset \mathbb{R}^2$ to be given in stereo-rectified image coordinates, i.e., the intrinsic and extrinsic camera parameters are known a-priori. The domain $\Omega_{D_i} \subset \Omega$ is the semi-dense restriction to the pixels which are selected for depth estimation.

We denote pixel coordinates by $\boldsymbol{u} = (u_x\, u_y\, 1)^T$. A 3D position $\boldsymbol{p} = (p_x\, p_y\, p_z\, 1)^T$ is projected into the image plane through the mapping $\boldsymbol{u} = \pi(\boldsymbol{p}) := \mathbf{K}\,((p_x/p_z)\,(p_y/p_z)\,1)^T$, where $\mathbf{K}$ is the camera matrix. The mapping $\boldsymbol{p} = \pi^{-1}(\boldsymbol{u}, d) := \left(\left(d^{-1}\mathbf{K}^{-1}\boldsymbol{u}\right)^T 1\right)^T$ inverts the projection with the inverse depth d.

B. Depth Estimation

We estimate the geometry of the scene in keyframes. Each keyframe maintains Gaussian probability distributions on the inverse depth of a subset of pixels. This subset is chosen as the pixels with high image gradient magnitude, since these pixels provide rich structural information and more robust disparity estimates than pixels in textureless areas. Figure 3 shows an example of such a semi-dense depth map and associated variance map. We initialize the depth map by propagating depth hypothesis from the previous keyframe. The depth map is subsequently updated with new observations in a pixel-wise depth-filtering framework. We also regularize the depth maps spatially and remove outliers.

In contrast to monocular SLAM, depth is estimated both from *static stereo* (i.e., using images from different physical cameras, but taken at the same point in time) as well as from *temporal stereo* (i.e., using images from the same physical camera, taken at different points in time).

a) Static Stereo: We determine the static stereo disparity at a pixel by a correspondence search along its epipolar line in the other stereo image. In our case of stereo-rectified images, this search can be performed very efficiently along horizontal lines.

As correspondence measure we use the SSD photometric error over five pixels along the scanline. After subpixel accurate refinement of the disparity, its variance is estimated through the geometric and photometric error identified in [16]. If a Gaussian prior with mean d and standard deviation σ_d on the inverse depth is available, we constrain the search to $[d - 2\sigma_d, d + 2\sigma_d]$. In practice, the search interval consists of only very few pixels for all but newly initialized hypothesis, greatly accelerating the search and reducing the probability of finding an incorrect or ambiguous match. According to the two error sources, we expect that pixels with image gradients close to vertical, or with low image gradient along the horizontal direction do not provide accurate disparity estimates. Hence, we neglect these pixels for static stereo.

When a new keyframe is initialized, we immediately perform static stereo to update and prune the propagated depth map. In particular, pruning removes pixels that became occluded, and we fill in holes arising from forward-warping the depth map. Subsequently, we also make use of static stereo from tracked non-keyframes, and integrate the obtained disparity information into the keyframe they were tracked on: In a first step, the inverse depth hypothesis at a pixel $\boldsymbol{u}$ in the keyframe is transformed into the new frame,

$$\boldsymbol{u}' = \pi\left(\mathbf{T}_{\boldsymbol{\xi}}\pi^{-1}\left(\boldsymbol{u}, d\right)\right) \tag{1}$$

$$d' = \left[\mathbf{T}_{\boldsymbol{\xi}}\pi^{-1}\left(\boldsymbol{u}, d\right)\right]_3^{-1} \tag{2}$$

$$\sigma_{d'}^2 = \left(\frac{d}{d'}\right)^4 \sigma_d^2, \tag{3}$$

according to the pose estimate $\boldsymbol{\xi}$. The propagated hypothesis is used as prior for a stereo search, and the respective observed depth d'_{obs} and observation variance $\sigma^2_{d',\text{obs}}$ is determined. Finally, the observation is transformed back into

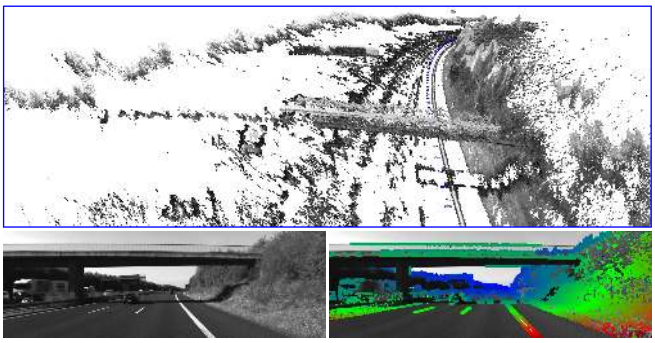


Fig. 4. Temporal vs. Static Stereo: Example of a scene where both temporal stereo (epipolar lines are parallel to the lane-markings on the road) and static stereo (epipolar lines are parallel to the horizontal bridge) alone fail to capture all information present. Our combined approach fuses information from both, and hence can reconstruct everything in the scene.

the keyframe using

$$d_{\text{obs}} = \left[\mathbf{T}_{\boldsymbol{\xi}}^{-1}(\pi^{-1}(\boldsymbol{u}', d'_{\text{obs}}))\right]_3^{-1} \tag{4}$$

$$\sigma_{\text{obs}}^2 = \left(\frac{d'_{\text{obs}}}{d_{\text{obs}}}\right)^4 \sigma_{d',\text{obs}}^2, \tag{5}$$

and fused into the depth map. Note that observations from non-keyframes can only be generated for pixels with an existing prior hypothesis – new hypothesis are only generated during stereo on the keyframe, or from temporal stereo. This process is schematically shown in Fig. 2.

b) Temporal Stereo: After tracking, we estimate disparity between the current frame and the reference keyframe and fuse it in the keyframe. Again, we only use pixels for which the expected inverse depth error is sufficiently small. We determine this uncertainty from several criteria: the image gradient should be sufficiently large, not be parallel to the epipolar line and the pixel should not be close to the epipole. We kindly refer to [16] for further details on this method. While we use a simple 5-pixel SSD error, we correct for affine lighting changes with the affine mapping found during tracking, as will be described in Sec. III-C. Note that for temporal stereo, the geometric error typically is higher than for static stereo, as relative camera pose stems from direct image alignment. This pose estimate often is less accurate than the offline calibrated extrinsic calibration between the stereo camera pair.

C. Direct Image Alignment with Affine Lighting Correction

We determine the camera motion between two images using direct image alignment. We use this method to track camera motion towards a reference keyframe. It is also used for estimating relative pose constraints between keyframes for pose graph optimization. Finally, we propose a robust method to compensate for affine lighting changes.

1) Direct Image Alignment: The relative pose between two images I_1^l and I_2^l is estimated by minimizing the photometric residuals

$$r_{\boldsymbol{u}}^I(\boldsymbol{\xi}) := I_1^l(\boldsymbol{u}) - I_2^l(\pi(\boldsymbol{p}')) \tag{6}$$

where $\boldsymbol{p}' := \mathbf{T}_{\boldsymbol{\xi}}\pi^{-1}(\boldsymbol{u}, D_1(\boldsymbol{u}))$ and $\boldsymbol{\xi}$ transforms from image frame I_2^l to I_1^l. We also determine the uncertainty $\sigma_{r,\boldsymbol{u}}^I$ of this residual [11]. The optimization objective for tracking a current frame towards a keyframe is

$$E^{\text{track}}(\boldsymbol{\xi}) := \sum_{u\in\Omega_{D_1}} \rho\left(\frac{r_{\boldsymbol{u}}^I(\boldsymbol{\xi})}{\sigma_{r,\boldsymbol{u}}^I}\right), \tag{7}$$

where ρ is a robust weighting function; we choose ρ as the Huber norm. Note that in contrast to [12], we only align I_1^l to I_2^l. While one could choose to add photometric constraints to the new right image I_2^r, we observed that this can decrease accuracy in practice: typically, the baseline from I_1^l to I_2^r is much larger than to I_2^l, leading to more outliers from occlusions and reflections.

Since fused depth is available in keyframes, we add geometric residuals for keyframe-to-keyframe alignment,

$$r_{\boldsymbol{u}}^D(\boldsymbol{\xi}) := [\boldsymbol{p}']_3 - D_2(\pi(\boldsymbol{p}')) \tag{8}$$

providing additional information that is not available when initially tracking new frames, since these not have associated depth estimates yet. The combined objective is

$$E^{\text{keyframes}}(\boldsymbol{\xi}) := \sum_{u\in\Omega_{D_1}} \left[\rho\left(\frac{r_{\boldsymbol{u}}^I(\boldsymbol{\xi})}{\sigma_{r,\boldsymbol{u}}^I}\right) + \rho\left(\frac{r_{\boldsymbol{u}}^D(\boldsymbol{\xi})}{\sigma_{r,\boldsymbol{u}}^D}\right)\right] \tag{9}$$

Note that this formulation exploits the full depth information available for both frames, including propagated and fused observations from other stereo pairs (see Sec. III-B). This is in contrast to an implicit quadrifocal approach as e.g. in [12].

We minimize these objectives using the iteratively re-weighted Levenberg-Marquardt algorithm in a left-compositional formulation: Starting with an initial estimate $\boldsymbol{\xi}^{(0)}$, in each iteration a left-multiplied increment $\delta\boldsymbol{\xi}^{(n)}$ is computed by solving for the minimum of a second-order approximation of E, with fixed weights:

$$\delta\boldsymbol{\xi}^{(n)} = -(\mathbf{J}^T\mathbf{W}\mathbf{J} + \lambda\text{diag}(\mathbf{J}^T\mathbf{W}\mathbf{J}))^{-1}\mathbf{J}^T\mathbf{W}\mathbf{r} \tag{10}$$

where

$$\mathbf{J} = \left.\frac{\partial\mathbf{r}(\boldsymbol{\epsilon}\circ\boldsymbol{\xi}^{(n)})}{\partial\boldsymbol{\epsilon}}\right|_{\boldsymbol{\epsilon}=0} \tag{11}$$

is the derivative of the stacked vector of residuals $\mathbf{r}(\boldsymbol{\xi})$ with respect to a left-multiplied increment $\boldsymbol{\epsilon}$, $\mathbf{J}^T\mathbf{W}\mathbf{J}$ the Gauss-Newton approximation of the Hessian of E, and $\mathbf{W}$ a diagonal matrix containing the weights. The new estimate is then obtained by multiplication with the computed update

$$\boldsymbol{\xi}^{(n+1)} = \delta\boldsymbol{\xi}^{(n)} \circ \boldsymbol{\xi}^{(n)}. \tag{12}$$

We use a coarse-to-fine scheme to improve efficiency and basin of convergence of the optimization.

Assuming the residuals to be statistically independent, the inverse of the Hessian from the last iteration $(\mathbf{J}^T\mathbf{W}\mathbf{J})^{-1}$ is an estimate for the covariance $\boldsymbol{\Sigma}_{\boldsymbol{\xi}}$ of a left-multiplied increment $\boldsymbol{\epsilon}$ onto the final minimum, that is

$$\boldsymbol{\xi}^{(n)} = \boldsymbol{\epsilon}\circ\boldsymbol{\xi}_{\text{true}} \quad \text{with} \quad \boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\xi}}). \tag{13}$$

In practice, the residuals are highly correlated, such that $\boldsymbol{\Sigma}_{\boldsymbol{\xi}}$ is only a lower bound - yet it contains valuable information about the correlation between noise on the different degrees of freedom.

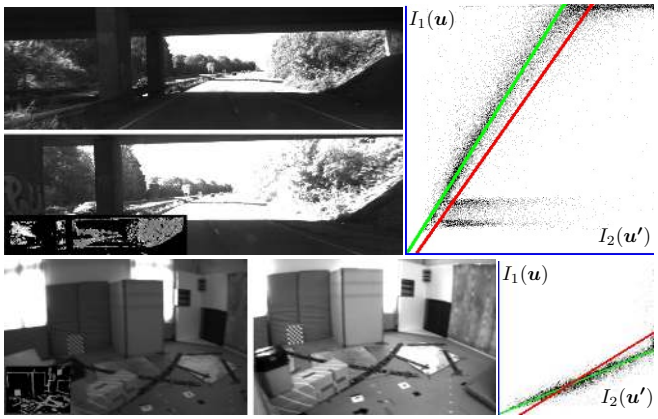


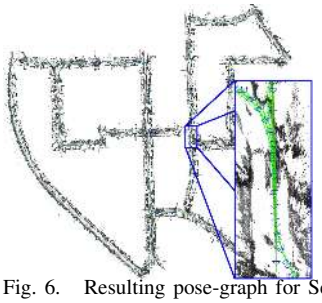
Fig. 5. Affine Lighting Correction: Two scenes with strong lighting changes. On the right, we show a the scatter-plot of all residuals *after* direct image alignment; The green line shows the best fit from our approach, while the red line shows the best fit for all pixel. Note how it is heavily affected by outliers caused by occlusions and over-exposed pixels, which are easily recognizable in the scatter-plot.

2) Affine Lighting Correction: Direct image alignment is fundamentally based on the brightness constancy assumption, which is heavily violated e.g. when the cameras exposure time is adjusted to better fit the average brightness of the scene. A well-known countermeasure is to use a cost function that is invariant to *affine lighting changes*, e.g. using the normalized cross correlation (NCC) instead of a simple sum of squared differences (SSD) for matching. Here, we propose a similar approach, and modify the photometric residuals (6) to be invariant to affine lighting changes:

$$r^I_{\boldsymbol{u}}(\boldsymbol{\xi}) := aI^l_1(\boldsymbol{u}) + b - I^l_2(\boldsymbol{p}'). \tag{14}$$

Instead of a joint optimization for a, b and $\boldsymbol{\xi}$ in a common error formulation, we alternate between (1) a single Levenberg-Marquardt update step in $\boldsymbol{\xi}$ (fixing a, b) and (2) a full minimization over a, b (fixing $\boldsymbol{\xi}$), using different weighting schemes. This is motivated by the observation that $\boldsymbol{\xi}$ and a, b react very differently to outliers:

- The minimum in a, b is heavily affected by occluded and over-exposed pixels, as these tend to "pull" in the same wrong direction. On the other hand, it typically is well-constrained already by only a small number of inlier-residuals – we therefore employ a simple, aggressive cut-off SSD error, i.e. $\rho_{a,b}(r) := \min\{\delta_{\text{max}}, r^2\}$. Fig. 5 shows two example scenes, and the resulting affine mapping with and without outlier rejection.
- The minimum in $\boldsymbol{\xi}$ is much less affected by outliers, as they tend to "pull" in different directions, cancelling each other out. In turn, it may happen that some dimensions of $\boldsymbol{\xi}$ are only constrained by a small amount of pixels, which initially have a high residual – removing these as outliers will cause the estimate to converge to a wrong local minimum. We therefore employ the weighting scheme proposed in [11], which only down-weights but does not remove residuals.



Level	Num	Time
46×30	10.5k	0.6 ms
92×60	6.1k	2 ms
184×120	4.6k	8 ms
368×240	4.1k	24 ms

Fig. 6. Resulting pose-graph for Sequence 00 from the Kitti benchmark, containing 1227 keyframes and 3719 constraints. The table shows how many constraints have been attempted to track down to which pyramid level, as well as the average time required for reciprocal image alignment on that pyramid level. Note how most incorrect loop-closures candidates are discarded at very coarse resolution already, which is very fast. Over the whole sequence, only 43 large loop-closure attempts were required, to find all loop-closures in the sequence.

Minimization in a, b is done by iteratively minimizing

$$E_{a,b}(a,b) := \sum_{\boldsymbol{u}\in\Omega_{D_1}} \rho_{a,b}\left(\left(aI^l_1(\boldsymbol{u}) + b\right) - I^l_2(\boldsymbol{u}')\right) \tag{15}$$

with $\boldsymbol{u}' := \pi\left(\boldsymbol{p}'\right)$, which can be done in closed-form:

$$a^* = \frac{\sum_{\boldsymbol{u}\in\Omega_L} I^l_1(\boldsymbol{u}) I^l_2(\boldsymbol{u}')}{\sum_{\boldsymbol{u}\in\Omega_L} I^l_2(\boldsymbol{u}') I^l_2(\boldsymbol{u}')} \tag{16}$$

$$b^* = \frac{1}{|\Omega_L|} \sum_i \left(I^l_1(\boldsymbol{u}') - a^* I^l_2(\boldsymbol{u})\right), \tag{17}$$

with the set of inliers

$$\Omega_L := \left\{\boldsymbol{u} \in \Omega_{D_1} \mid \rho_{a,b}\left(\left(aI^l_1(\boldsymbol{u}) + b\right) - I^l_2(\boldsymbol{u}')\right) < \delta_{\text{max}}\right\}.$$

The found affine parameters a, b are then used during temporal stereo and during the consistency check on depth propagation.

D. Key-Frame-Based SLAM

Once a keyframe $\mathcal{K}_i$ is finalized – that is, after it is replaced as tracking reference and will not receive any further depth updates – it is added to the pose-graph, which is continuously optimized in the background. Constraints are obtained by performing SE(3) alignment with depth residual and affine lighting correction to a set of possible loop-closure candidates: Tracking is attempted on all keyframes $\mathcal{K}_{j_1}, ..., \mathcal{K}_{j_n}$, which

- are at a physical distance of less than $(60 + p \cdot 0.05)\,\text{m}$.
- have a difference in viewing direction of less than $(35 + p \cdot 0.01)^\circ$.

where p is the length of the shortest connecting path in the keyframe graph between the two keyframes in meters, which serves as a conservative approximation to the accumulated relative pose error. For very large maps, additional loop-closures can be found by exploiting appearance-based image-retrieval techniques like FAB-MAP [17]. However in our experiments we did not find this to be necessary. For keyframes with $p \leq 100\,\text{m}$, we use the relative pose obtained by composing edges along this path as initialization for direct image alignment, otherwise the identity is used.

TABLE I

RESULTS ON KITTI BENCHARK

Seq.	SLAM t_{rel}	SLAM r_{rel}	SLAM t_{abs}	SLAM time	VO t_{rel}	VO r_{rel}	VO time
00	0.63	0.26	1.0	82	1.09	0.42	21
01	2.36	0.36	9.0	37	2.13	0.37	24
02	0.79	0.23	2.6	64	1.09	0.37	28
03	1.01	0.28	1.2	72	1.16	0.32	27
04	0.38	0.31	0.2	51	0.42	0.34	28
05	0.64	0.18	1.5	77	0.90	0.34	29
06	0.71	0.18	1.3	72	1.28	0.43	29
07	0.56	0.29	0.5	74	1.25	0.79	31
08	1.11	0.31	3.9	73	1.24	0.38	29
09	1.14	0.25	5.6	61	1.22	0.28	30
10	0.72	0.33	1.5	70	0.75	0.34	21
mean 00-10	0.91	0.27	2.6	67	1.14	0.40	29
mean 11-21	1.21	0.35	–	69	1.40	0.36	28

- t_{rel}: translational RMSE drift (%), av. over 100 m to 800 m intervals.
- r_{rel}: rotational RMSE drift (deg per 100 m), av. over 100 m to 800 m intervals.
- t_{abs}: absolute RMSE after 6DoF alignment, in meters.
- time: single-threaded computation time per frame, in milliseconds.

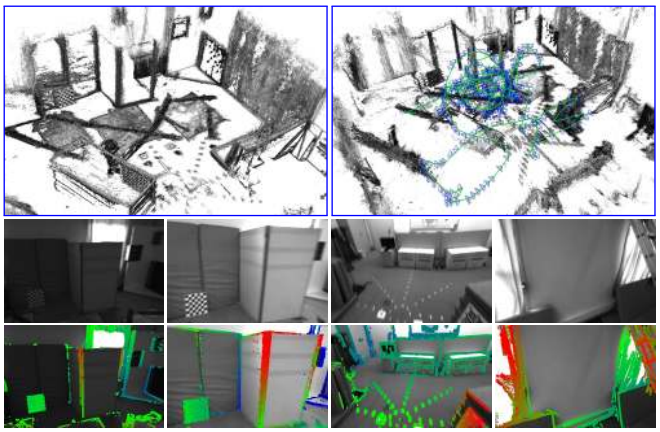


Fig. 7. EuRoC Datasets from a micro aerial vehicle. Top: reconstruction from the first (left) and third (right) trajectory. Bottom: Selection of images from the third trajectory, displaying strong lightning changes (first to second image), motion blur (third image) and views with little texture (fourth image).

For each candidate $\mathcal{K}_{j_k}$ we independently compute $\boldsymbol{\xi}_{j_k i}$ and $\boldsymbol{\xi}_{i j_k}$ by minimizing (9). Only if the two estimates are statistically similar, i.e., if

$$e(\boldsymbol{\xi}_{j_k i}, \boldsymbol{\xi}_{i j_k}) := (\boldsymbol{\xi}_{j_k i} \circ \boldsymbol{\xi}_{i j_k})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{\xi}_{j_k i} \circ \boldsymbol{\xi}_{i j_k}) \quad (18)$$

$$\text{with} \quad \boldsymbol{\Sigma} := \boldsymbol{\Sigma}_{j_k i} + \text{Adj}_{j_k i} \boldsymbol{\Sigma}_{i j_k} \text{Adj}_{j_k i}^T \quad (19)$$

is sufficiently small, they are added as constraints to the pose-graph. Here, $\text{Adj}_{j_k i}$ is the adjoint of $\boldsymbol{\xi}_{j_k i}$ in SE(3). To speed up the removal of incorrect loop-closure candidates, we apply this consistency check after each pyramid level. Only if it passes, direct image alignment is continued on the next higher resolution. This allows to discard most incorrect candidates with only very little wasted computational resources: Figure 6 shows how many constraints where tracked on which pyramid level for one of the longest sequences in the Kitti dataset.

IV. RESULTS

We present the results obtained by Stereo LSD-SLAM (1) on the well-known Kitti dataset, and (2) on three sequences recorded from a micro aerial vehicle (MAV) flying indoors, taken from the EuRoC Challenge 3. We evaluate both the runtime and accuracy, for different parameter settings. Although our implementation makes heavy use of multiple CPU cores, all timings given in this chapter refer to single-threaded execution on an Intel i7-4900MQ CPU running at 2.8 Ghz.

A. EuRoC Dataset

We run Stereo LSD-SLAM on the EuRoC dataset, taken from a MAV flying around a room which is equipped with a motion capture system for ground truth acquisition. The dataset contains 3 trajectories, with increasingly aggressive motion. Fig. 7 shows the reconstruction obtained. The absolute translational RMSE is 6.6 cm, 7.4 cm and 8.9 cm for the first, second and third trajectory respectively. In this dataset we removed the first and last 150 images for each trajectory, as in some of them only the ground surface is visible.

B. Kitti Dataset

We evaluated our method on the well-known Kitti dataset. Table I summarizes the results both for Stereo LSD-SLAM with, and without loop-closures (VO). The results given are for half resolution, as we feel this is a better trade-off between accuracy and computational speed – see also Sec. IV-D. On the evaluation sequences 11-21, we achieve a mean translational RMSE of 1.21% for full SLAM, which currently ranks second amongst stereo methods. Stereo LSD-SLAM is however much faster than methods achieving similar accuracy. The increased error compared to the test sequences 00-10 is due to the presence of many moving objects in 20 and 21, which cause direct image alignment to occasionally fail (Sec. IV-F). Furthermore, the Kitti benchmark only provides images captured at 10 Hz while driving at speeds of up to 80 km/h – which is challenging for direct methods, as these are good at exploiting small intra-frame motions.

C. Visual Odometry vs. SLAM

Here, we evaluate the capability to perform large-scale loop-closures when running the full SLAM system, as well as the effect of only performing loop-closures in a small window of the last l frames – effectively turning Stereo LSD-SLAM into a Visual Odometry. For $l = 0$, no image alignment with geometric error is performed, and only the pose from the initial frame alignment is used. For this comparison, we only consider Kitti sequences which contain significant loop-closures, i.e. 00, 02, 05, 06 and 07. Figure 8 summarizes the result: It can clearly be seen that performing full SLAM greatly decreases long-term drift, which is little surprising. However, this comes at increased computational cost: when performing full SLAM, the overall computational budget required more than doubles (also see Tab. I), as the full pose-graph has to be optimized and many loop-closure constraints have to be tracked. All numbers in this Section refer to running Stereo LSD-SLAM at half resolution.

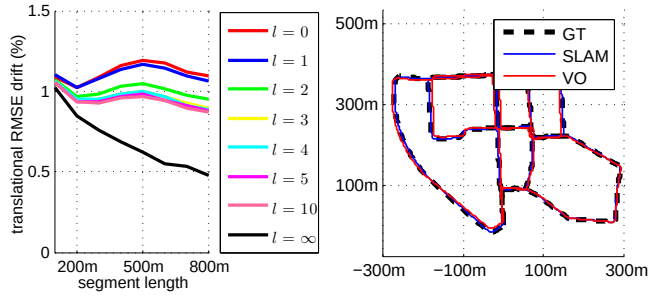


Fig. 8. Visual Odometry vs. SLAM: Left: translational drift over different evaluation segment lengths, for different sizes of the pose-graph optimization window l. For $l = \infty$, our method performs full SLAM; hence the translational drift decreases when evaluating over longer segments (down to 0.5%). Right: 6DoF-aligned trajectories of the Kitti 00 sequence. While performing local pose-graph optimization slightly increases the local accuracy, it cannot remove drift over long segments.

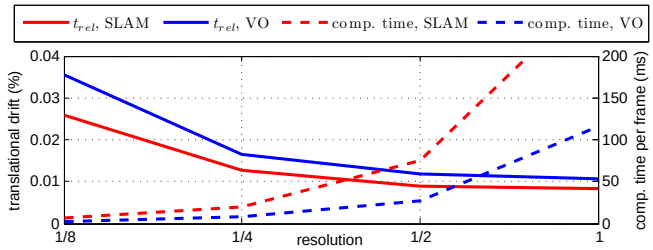


Fig. 9. Image Resolutions: The plot shows the mean translational RMSE t_{rel} for different image resolutions, as well as the required computation time. Stereo LSD-SLAM allows to smoothly trade-off one for the other – for an image resolution of one eight of the original, it runs at 400 Hz (VO) / 145 Hz (SLAM) in a single thread, still achieving a mean drift of only 3.5% (VO) and 2.5% (SLAM).

TABLE II
COMPUTATIONAL TIME REQUIRED

	154×46	310×92	620×184	1240×368
Tracking	1.2 ms	4.2 ms	16.0 ms	61.0 ms
Mapping	0.8 ms	2.9 ms	13.1 ms	62.8 ms
Constr. Search	3.7 ms	10.5 ms	40.0 ms	143.1 ms
Pose-Graph Opt.	1.2 ms	1.3 ms	1.4 ms	1.3 ms
Total (SLAM)	6.9 ms	18.9 ms	70.5 ms	268.2 ms

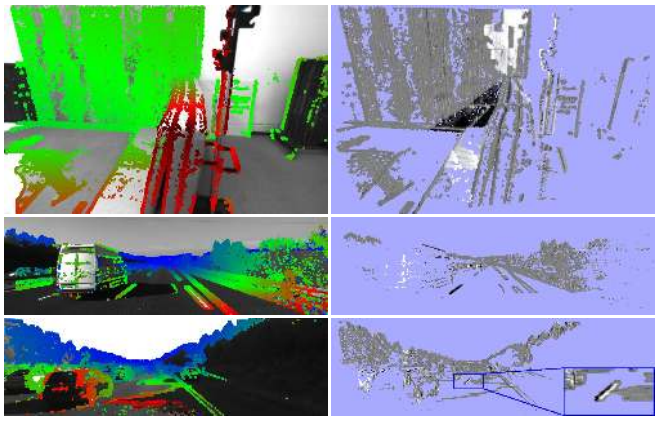


Fig. 10. Examples for scenes with moving objects & strong occlusions. On the right, we show the intensity residual after direct image alignment (small values are shown in gray; large negative / positive residuals are shown in black / white). While in the first two examples direct image alignment locks onto the correct motion, in the last one, it latches onto the wrong motion in the scene – the moving cars – and fails to align the two images correctly. This can be seen by the residual around the lane marking.

D. Effect of Image Resolution

A beautiful property of Stereo LSD-SLAM is that the achieved accuracy degrades very gracefully with decreasing image resolution, while the computational budget required shrinks rapidly. In fact, we were able to run both full SLAM as well as VO on the Kitti dataset at down to one eight of the original resolution, i.e., 154×46 pixels, and still achieve a reasonable mean translational drift of 2.5% (SLAM) and 3.5% (VO) – at greatly reduced computational cost, running in 15× real-time (SLAM) and 40× real-time (VO). The result is summarized in Fig. 9.

E. Performance analysis

In Table II, we summarize the computational time required for each part of the algorithm. All timings are given in milliseconds per frame. For lower resolutions, images are downsampled in a pre-processing step, as this typically can be done at no additional cost in hardware (pixel binning). It can clearly be observed that all parts of the algorithm – except for pose-graph optimization – directly scale with the number of pixels in the image. Only at very low resolution, resolution-independent operations – like inverting the Hessian during LM minimization – start to have a visual impact.

F. Moving Objects & Occlusions

A remarkable property of direct image alignment approaches is the "locking property" [18]: In the presence of multiple motions or outliers, the coarse-to-fine approach causes direct methods to lock onto the most dominant motion within the validity radius of the linearisation. A robust weighting function then allows to minimize the effect of pixels not belonging to this motion. Figure 10 shows three examples in which large parts of the image are moving or become occluded: In the first two examples the dominant motion is correctly identified, whereas in the third example image alignment locks onto the moving cars in the foreground. We observed this problem only in Sequence 20 of the Kitti benchmark as there are many cars moving at the same speed – arguably making the dominant motion in the scene that of the cars. For the on-line evaluation, we resolve this by removing all points in a certain volume in front of the car for this sequence only. Nevertheless, future work could take advantage of our approach, for example by segmenting the scene motion into a number of rigid-body motions ([18], [19], [20]).

G. Qualitative Results

We show in Fig. 11 some qualitative results of the estimated semi-dense depth maps, and the resulting point-clouds. Note how depth is estimated in almost all areas that have gradient information, and how many fine details (signs, lamp posts) are recovered. Also, the inclusion of temporal stereo allows to estimate depth for strictly horizontal structures, like the power transmission lines visible in some of the images.

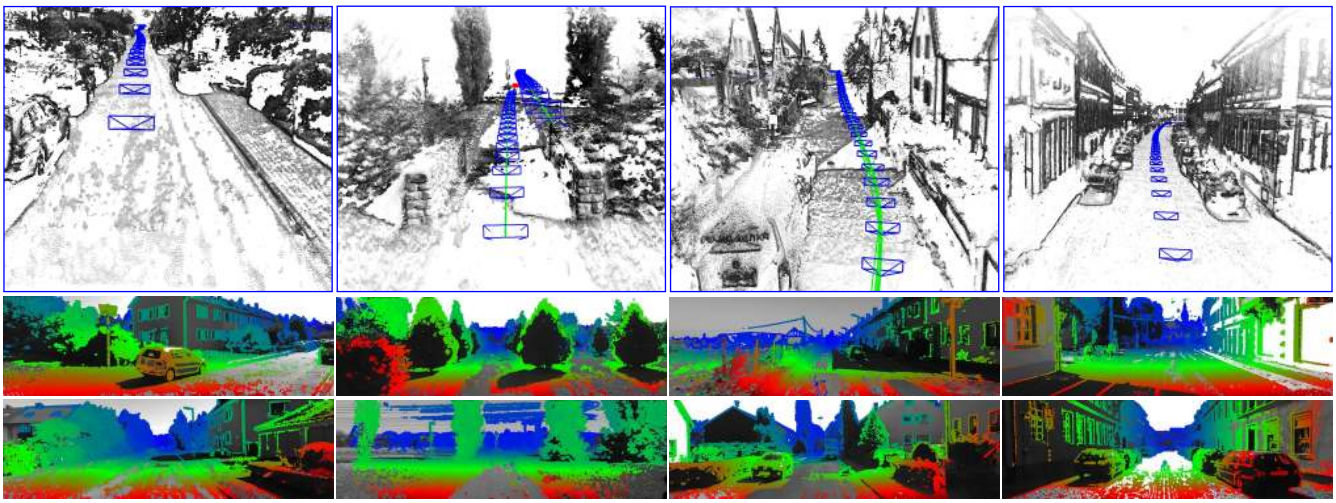


Fig. 11. Point clouds and depth maps for the Kitti dataset (sequences 08,14,15,18), running at full resolution. Also see the attached video.

V. CONCLUSIONS

We proposed Stereo LSD-SLAM, a novel direct approach to SLAM with stereo cameras. Our method leverages static, fixed-baseline stereo as well as temporal, variable-baseline stereo cues. Static stereo provides accurate depth within the effective operating range of the stereo camera. It also removes scale ambiguities and difficulties with degenerate motion along the line of sight, a problem inherent to monocular SLAM that only uses temporal stereo. With temporal stereo on the other hand, depth can be estimated in variable baseline directions that correspond to the translational motion between frames.

Our method directly aligns images using photometric and geometric residuals at a semi-dense set of pixels. We choose pixels where there is sufficient information for static or temporal stereo estimation. In contrast to sparse interest-point-based methods, our approach is not restricted to a specific type of image features that are extracted in a decoupled processing stage prior to image alignment.

In our experiments, Stereo LSD-SLAM demonstrates state-of-the-art results on the popular Kitti benchmark dataset for stereo odometry and SLAM on autonomous cars. Stereo LSD-SLAM also performs very accurate on challenging sequences recorded with a micro aerial vehicle (MAV) for the EuRoC Challenge 3. Both datasets are very challenging for a purely monocular SLAM approach, since motion is mainly along the line of sight (cars), or can mainly consist of rotations (MAVs).

In future work, we consider extending our approach to multi-camera setups beyond binocular stereo cameras. Sensor fusion with inertial or GPS information could further enhance accuracy and robustness on the local and the global scale. Finally, we plan to address multi-body motion segmentation and estimation. This way, our method would not only recover the dominant motion in the images, but also the motion of further independent moving objects.

REFERENCES

[1] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[2] A. Chiuso, P. Favaro, H. Jin, and S. Soatto, "Structure from motion causally integrated over time," vol. 24, no. 4, pp. 523–535, Apr 2002.

[3] A. Davison, I. Reid, N. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 29, 2007.

[4] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Int. Symp. on Mixed and Augmented Reality (ISMAR)*, 2007.

[5] H. Strasdat, J. Montiel, and A. Davison, "Scale drift-aware large scale monocular slam." in *Robotics: Science and Systems (RSS)*, 2010.

[6] A. J. Davison and D. W. Murray, "Simultaneous localization and map-building using active vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 865–880, July 2002.

[7] L. M. Paz, P. Pinies, J. Tardos, and J. Neira, "Large-scale 6-dof slam with stereo-in-hand," *Transaction on Robotics*, Oct 2008.

[8] C. Kerl, J. Sturm, and D. Cremers, "Robust odometry estimation for RGB-D cameras," in *Int. Conf. on Robotics and Automation (ICRA)*, 2013.

[9] ——, "Dense visual SLAM for RGB-D cameras," in *Int. Conf. on Intelligent Robot Systems (IROS)*, 2013.

[10] M. Meilland and A. Comport, "On unifying key-frame and voxel-based dense visual SLAM at large scales," in *Int. Conf. on Intelligent Robot Systems (IROS)*, 2013.

[11] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: Large-scale direct monocular SLAM," in *European Conference on Computer Vision (ECCV)*, 2014.

[12] A. Comport, E. Malis, and P. Rives, "Accurate quadri-focal tracking for robust 3d visual odometry," in *Int. Conf. on Robotics and Automation (ICRA)*, 2007.

[13] T. Tykkala and A. Comport, "A dense structure model for image based stereo SLAM," in *Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[14] S. Klose, P. Heise, and A. Knoll, "Efficient compositional approaches for real-time robust direct visual odometry from RGB-D data," in *Int. Conf. on Intelligent Robot Systems (IROS)*, 2013.

[15] T. Goncalves and A. Comport, "Real-time direct tracking of color images in the presence of illumination variation,," in *Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[16] J. Engel, J. Sturm, and D. Cremers, "Semi-dense visual odometry for a monocular camera," in *Int. Conf. on Computer Vision (ICCV)*, 2013.

[17] M. Cummins and P. Newman, "Appearance-only SLAM at large scale with FAB-MAP 2.0," *Int. J. Robotics Research*, 2010.

[18] M. Irani and P. Anandan, "All about direct methods," 1999.

[19] G. Zhang, J. Jia, and H. Bao, "Simultaneous multi-body stereo and segmentation," in *Int. Conf. on Computer Vision (ICCV)*, 2011.

[20] J. Stückler and S. Behnke, "Efficient dense rigid-body motion segmentation and estimation in RGB-D video," *Int. J. Comput. Vision (IJCV)*, Jan. 2015.