Probabilistic Semi-Dense Mapping from Highly Accurate Feature-Based Monocular SLAM

Raúl Mur-Artal and Juan D. Tardós
Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, Spain
{raulmur, tardos}@unizar.es

***Abstract*—In the last years several direct (i.e. featureless) monocular SLAM approaches have appeared showing impressive semi-dense or dense scene reconstructions. These works have questioned the need of features, in which consolidated SLAM techniques of the last decade were based. In this paper we present a novel feature-based monocular SLAM system that is more robust, gives more accurate camera poses, and obtains comparable or better semi-dense reconstructions than the current state of the art. Our semi-dense mapping operates over keyframes, optimized by local bundle adjustment, allowing to obtain accurate triangulations from wide baselines. Our novel method to search correspondences, the measurement fusion and the inter-keyframe depth consistency tests allow to obtain clean reconstructions with very few outliers. Against the current trend in direct SLAM, our experiments show that by decoupling the semi-dense reconstruction from the trajectory computation, the results obtained are better. This opens the discussion on the benefits of features even if a semi-dense reconstruction is desired.**

I. Introduction

The problem of Visual Simultaneous Localisation and Mapping (Visual SLAM) has attracted the attention of the robotics community for more than a decade. Solving this problem can provide a robot the desirable information of self-localisation and a model of its environment to interact with it. Most consolidated techniques have relied on features [8, 1], while recent approaches make use of direct methods [14, 3].

A. Feature-based SLAM

Modern feature-based techniques [8, 19, 12] are based on keyframes [20] and bundle adjustment (BA) optimization [23]. These techniques extract features on the images, typically keypoints selected by their repeatability and distinctiveness from different viewpoints. Camera poses and map features are jointly optimized by BA, which minimizes the reprojection error. The main strengths are the following:

- Due to their good illumination and viewpoint invariance, features provide wide baseline matches, which in conjunction with large loop closures, give a strong camera network for bundle adjustment or pose graph optimization, resulting in very accurate solutions.
- Bags of words and binary features [5] allow to perform place recognition in real time in large scale environments and, depending on the features, with a high invariance to viewpoint [11].
- As features are triangulated from spatially and temporally distant keyframes, moving objects are typically successfully ignored. This characteristic and the use of RANSAC and robust cost functions make the SLAM system robust in the presence of dynamic elements.

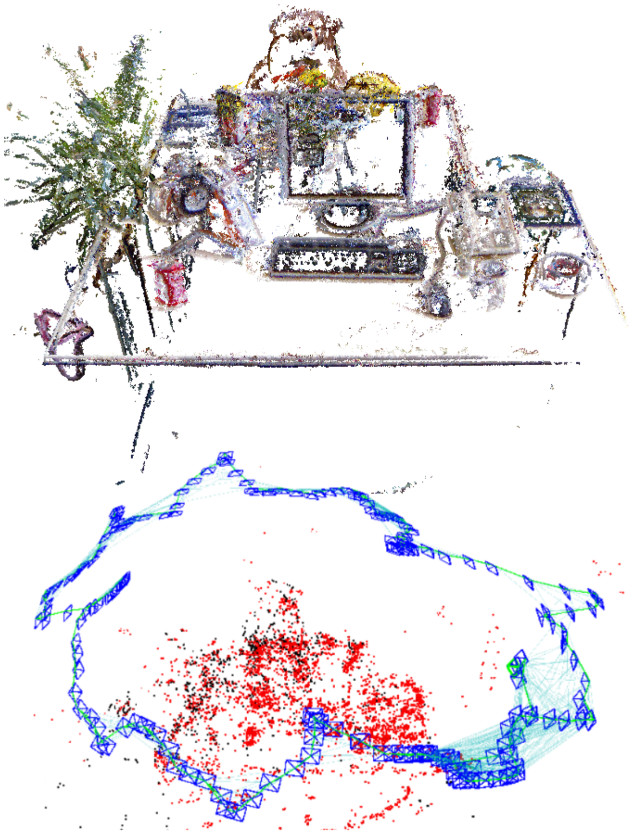


Fig. 1. Example of a semi-dense reconstruction (top, best seen in color) of the *fr2_desk* sequence from the TUM RGB-D Benchmark [22], performed in real-time by our system. Only points with small inverse depth uncertainty are shown. Our highly accurate feature-based monocular SLAM system [12] provides a stream of accurately localised keyframes (bottom) from which we compute the semi-dense reconstruction.

The main inconvenient of feature-based SLAM is that the map is very sparse, being of little use for robotic tasks such as navigation or object interaction. However the map has excellent camera localisation capabilities (see Fig. 1, bottom).

B. Direct SLAM and Semi-Dense/Dense Mapping

Direct SLAM approaches [3, 14] localise the camera optimising the pose directly over pixel intensities, minimizing the photometric error. These approaches perform a dense (all pixels in the image) or a semi-dense (only high gradient areas) reconstruction. While dense reconstruction methods [21, 13, 14, 15] reconstruct surfaces and require GPU acceleration due to the computational cost involved, semi dense approaches [3] recover object contours and textured surfaces, requiring no GPU but multi-threading optimization. The main strengths of these approaches are:

- Rich scene representation useful for object or scene recognition, navigation or augmented reality.
- Robust tracking under defocus or motion blur, provided the area is first mapped under favorable conditions [14].
- As not using features, tracking and mapping is still reliable in scenes where few features could be detected.

The impressive results of these approaches, have questioned the need of features and seem to suggest an evolution from feature-based methods to direct SLAM.

C. Semi-Dense Mapping over Feature-Based SLAM

Building on excellent feature-based algorithms developed in the last years [8, 18, 19, 16, 9, 5], we have designed ORB-SLAM, a new feature-based monocular SLAM system [12], whose source code is online available[1]. ORB-SLAM operates in real-time in indoors and outdoors environments, being able to relocalise and close loops from very different viewpoints and with a robust map bootstrapping. We evaluated exhaustively our system in 27 public sequences from the exigent KITTI [6], TUM RGB-D [22] and NewCollege [17] datasets, demonstrating unprecedented accuracy and robustness, superior to those shown by direct approaches. Some examples will be presented in section IV. For the reader convenience we summarize our feature-based SLAM in section II.

Some previous works have proposed dense reconstruction methods using GPUs, built over feature-based SLAM [13, 21] or visual odometry algorithms [15]. Following a similar approach, in this paper we propose a novel system incorporating to ORB-SLAM an especially designed probabilistic semi-dense mapping module, to perform in real-time, without GPU acceleration, rich semi-dense reconstructions. One of the main novelties of our semi-dense mapping method is that instead of using many subsequent frames to filter the inverse depth of a reference frame [2, 3, 15], we perform the reconstruction over keyframes, which are very well localised by local bundle adjustment, and pose graph optimization after a loop closure. This allows to obtain high quality and accurate reconstructions. If the highest accuracy is desired, the reconstruction can also be performed at the end of the session in few seconds after a full bundle adjustment. Fig. 1 shows an example of a semi-dense reconstruction obtained by our system.

Our stereo correspondence search and inverse depth uncertainty derivation is based on [2]. However as searching on keyframes (wider baselines) we have to deal with potentially more outliers, due to occlusions or multiple similar pixels. To gain robustness, in addition to the photometric similarity, we compare the modulo and orientation of the image gradient, and propose a novel measurement fusion. We also propose an *inter*-keyframe depth consistency check that discards most of the outliers, see an example in Fig. 4. In contrast to [2], our formulation do not make assumptions of small rotations in the derivation of the inverse depth uncertainty. We describe our semi-dense mapping approach in section III.

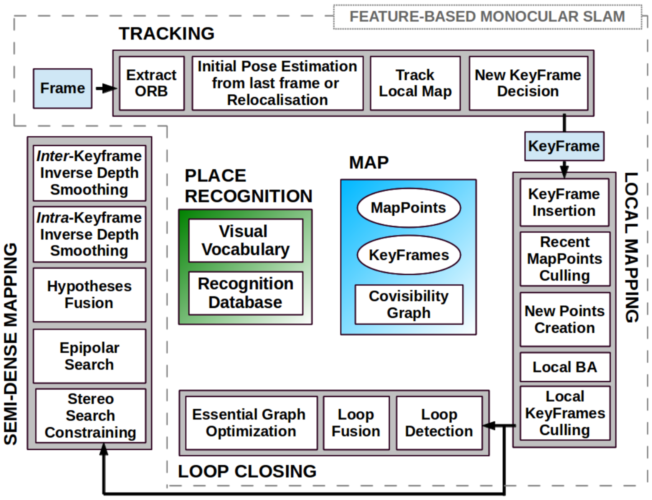


Fig. 2. Our whole system including the three threads of the feature-based monocular SLAM [12], tracking, local mapping and loop closing, and the semi-dense mapping thread proposed in this work

After the excellent recent works [14, 3], there is the extended believe in the community that direct methods are more robust because they do not need features, and are more accurate because they use more information from the images. Surprisingly, our results in section IV show the opposite.

II. Underlying Monocular SLAM

In this section we review ORB-SLAM [12], the monocular SLAM that we use to compute the pose of selected keyframes. One important property of the system is that the same ORB features [16] used for the tracking and mapping are used by the bags of words place recognition module, based on DBoW2 [5], to perform global relocalisation and loop closing. ORB features are oriented multi-scale FAST corners with a 256 bit descriptor. They are very fast to extract and match while they are invariant to any rotation and to scale in a range. These properties allow us to extract 1000 features per image at frame-rate and get matches from different viewpoints and under illumination changes. An overview of the system is shown in Fig. 2. We review next each of the three system threads.

A. The Tracking Thread

The goal of the tracking is to localise the camera with every frame and to decide when to insert a new keyframe. We use first a constant velocity motion model to guess the current camera pose and perform an initial matching with the previous frame. In case the motion model is violated, e.g. due to abrupt movements, a coarse window search is performed centered in the feature positions on the last frame. If the tracking is lost, e.g. due to a big occlusion, the place recognition module is

[1] http://webdiis.unizar.es/~raulmur/orbslam

used to relocalise the camera. Once we have initial matchings we can select a reference keyframe in the map which share most points with the current frame. We then retrieve a local covisible map from the keyframes connected to the reference one in the covisibility graph [19, 10]. Points in the local map are then projected in the current frame and matched. The camera pose is finally optimized by motion-only bundle adjustment (i.e. points are fixed) using the Huber cost function.

One of the main novelties of our SLAM is a *survival of the fittest* approach to keyframe selection: keyframe insertion policy is generous as there exists a keyframe culling procedure in the local mapping thread that will later discard redundant keyframes. This boosts tracking robustness under hard exploring conditions (i.e. rotations, fast movements) as keyframes are inserted every few frames, while the map is maintained compact by the culling procedure.

B. The Local Mapping Thread

The local mapping thread processes new keyframes and performs local bundle adjustment. Firstly it performs an epipolar search of unmatched ORB features in connected keyframes in the covisibility graph. Those successfully matched are triangulated generating new map points. An exigent point culling policy is applied to those points some time after creation, based on the information gathered during the tracking, in order to retain only high quality points. The local mapping is also in charge of culling redundant keyframes, based on the amount of points that are also seen by other keyframes.

C. The Loop Closing Thread

The loop closing thread queries the keyframe database and retrieves loop candidate keyframes. Loop candidates are geometrically validated, computing a similarity transformation that informs about the drift accumulated in the loop. To correct the loop, firstly both sides of the loop are aligned and duplicated points are fused. Finally a pose graph optimization over similarity constraints [18] is performed to achieve global consistency. To reduce the complexity, the optimization is performed over what we call the *Essential Graph*, a subgraph of the covisibility graph that retains all keyframes but includes less edges, still preserving a strong network.

III. Probabilistic Semi-Dense Mapping

Our probabilistic semi-dense mapping technique processes the keyframes provided by the monocular SLAM system to reconstruct textured surfaces and object contours. The outline of our method is the following:

1) Each keyframe K_i is processed from scratch. Every pixel in a high gradient area is searched along the epipolar line on N neighbor keyframes, yielding N inverse depth hypotheses.
2) Each inverse depth hypothesis is represented by a gaussian distribution that takes into account the image noise, the parallax and the ambiguity in the matching. We consider that the keyframe poses are well localised and do not take into account their uncertainty.
3) Because the baseline is wide between keyframes the search range along the epipolar line is large. To deal with outlier measurements, due to similar pixels or occlusions, we fuse the maximum subset of the N hypotheses that are mutually compatible. Each pixel p of the inverse depth map is then characterized with a gaussian distribution $\mathcal{N}(\rho_p, \sigma^2_{\rho_p})$.
4) As proposed in [2], a smoothing step is then applied to the inverse depth map so that a pixel is averaged with its neighbors. If a pixel inverse depth distribution is not compatible with its neighbors it is discarded.
5) After the neighbor keyframes have also computed their respective inverse depth maps, consistency in the per-pixel depths is checked across neighbor keyframes to discard outliers and the final depth is refined by optimization.

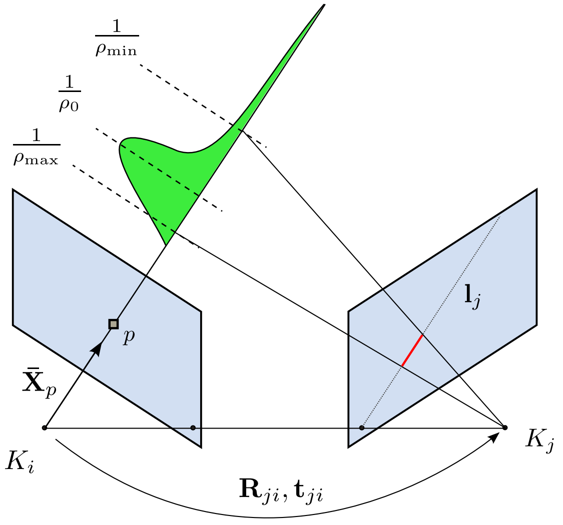


Fig. 3. Epipolar constrained search of a pixel in a neighbor keyframe given a prior inverse depth distribution.

Next we describe in detail each step.

A. Stereo Search Constraints

Our feature-based SLAM system provides useful information to constraint the search of pixel correspondences to compute the inverse depth map. On one hand keyframes have associated tracked ORB features with known depth, which renders us the maximum $\rho_{\max}$ and minimum $\rho_{\min}$ expected inverse depths of the scene. This provides a prior $\mathcal{N}(\rho_0, \sigma^2_{\rho_0})$, with $\rho_{\max} = \rho_0 + 2\sigma_{\rho_0}$ and $\rho_{\min} = \rho_0 - 2\sigma_{\rho_0}$, for the inverse depth search, as illustrated in Fig. 3.

In addition using the covisibility graph we can retrieve the set of N keyframes $\mathcal{K}$, which share most map point observations with K_i, and focus the stereo search in those keyframes. Keyframes are processed with a small delay (around 10 keyframes) so that they can be reconstructed using also *future* keyframes to get a better reconstruction. This is also convenient as local BA optimizes recent keyframes, potentially interfering with this semi-dense mapping thread.

B. Epipolar Search

Each pixel p of K_i with gradient modulo greater than a threshold λ_G is searched along the epipolar line $\mathbf{l}_j$ on each keyframe $K_j \in \mathcal{K}$, constrained to the segment between $\rho_{\min}$ and $\rho_{\max}$. The epipolar line is computed from the fundamental matrix $\mathbf{F}_{ji}$ [7], and for the sake of simplicity in the rest we parametrize it as a function of the horizontal coordinate u_j:

$$\mathbf{x}_j^\top \mathbf{F}_{ji}\mathbf{x}_p = \mathbf{x}_j^\top \mathbf{l}_j = 0 \quad \rightarrow \quad v_j = m \cdot u_j + n \tag{1}$$

In contrast to [2] (narrow baseline frames), our search along the epipolar line is larger (wider baseline keyframes) and we need to take special care of outliers. Therefore, in addition to comparing the intensity I, we propose to compare the modulo G and orientation Θ of the image gradient.

The pixel p is characterized by an intensity value I_p, a gradient modulo G_p and orientation Θ_p, and the goal is to find its best correspondence on $\mathbf{l}_j$. Firstly the pixels p_j not fulfilling the following conditions are not considered:

- p_j must lie in a high gradient area, that is $G(u_j) > \lambda_G$.
- The ambiguity of a match is related to the intensity gradient along the epipolar line [2]. Therefore the gradient direction must not be perpendicular to the epipolar line, that is $|\Theta(u_j) - \Theta_L \pm \pi| < \lambda_L$, with Θ_L the epipolar line angle (considering both directions).
- The gradient orientation of p_j must be similar, that is $|\Theta(u_j) - (\Theta_{p_i} + \Delta\theta_{j,i})| < \lambda_\theta$, where $\Delta\theta_{j,i}$ is the in-plane rotation between keyframe images, which is computed from the median rotation of corresponding ORB between both keyframes.

These conditions discard most of the points of the epipolar line, reducing potential mismatches. To compare the remaining points we define a similarity error $e(u_j)$:

$$e(u_j) = \frac{r_I^2}{\sigma_I^2} + \frac{r_G^2}{\sigma_G^2}, \quad r_I = I_p - I(u_j), \quad r_G = G_p - G(u_j) \tag{2}$$

where r_I is the photometric error and r_G is the gradient modulo error; σ_I and σ_G are the standard deviation of the intensity and gradient respectively. Because the gradient is a function of the intensity their noise are related $\sigma_G^2 = \theta\sigma_I^2$ with $\theta = 0.23$ if using the Scharr operator to compute the image derivatives ($\theta < 1$ as the Scharr operator performs an average reducing the noise). With this relation the similarity error is:

$$e(u_j) = (r_I^2 + \frac{1}{\theta} r_G^2)\frac{1}{\sigma_I^2} \tag{3}$$

We select the pixel at coordinate u_0 that minimizes this error, with residuals r_{I_0} and r_{G_0}. We can then compute the derivate of the error:

$$\frac{\partial e}{\partial u_j} = \frac{-2(r_I\, g + \frac{1}{\theta} r_G\, q)}{\sigma_I^2} \tag{4}$$

where g is the intensity gradient and q is the derivate of the intensity gradient modulo, both along the epipolar line:

$$g \approx \frac{I(u_j+1) - I(u_j-1)}{2}, \quad q \approx \frac{G(u_j+1) - G(u_j-1)}{2} \tag{5}$$

Performing a first order taylor approximation of the residuals (2) and equaling to zero the similarity error derivate (4) we can retrieve the pixel correspondence with subpixel precision:

$$u_0^* = u_0 + \frac{g(u_0) r_I(u_0) + \frac{1}{\theta} q(u_0) r_G(u_0)}{g^2(u_0) + \frac{1}{\theta} q^2(u_0)} \tag{6}$$

Now we can derive the uncertainty of u_0^* from the intensity noise σ_I^2 by error propagation, for simplicity considering only the noise in the residuals $r_I(u_0)$ and $r_G(u_0)$:

$$\sigma_{u_0^*}^2 = \frac{2\sigma_I^2}{g^2(u_0) + \frac{1}{\theta} q^2(u_0)} \tag{7}$$

This uncertainty tell us that a match is more reliable as higher is the gradient along the epipolar of the quantities involved in the similarity measure (2), in our case the intensity and the image gradient modulo.

Now we need to propagate the uncertainty of the match $\sigma_{u_0^*}^2$ to the uncertainty in the inverse depth $\sigma_{\rho_p}^2$. The inverse depth ρ_p of pixel p in K_i is a function of the position in the epipolar line u_j (which can be derived from the formula of the projection of a 3D world point into a camera image [7]):

$$\rho_p(u_j) = \frac{\mathbf{r}_z^{ji}\,\bar{\mathbf{X}}_p(u_j - c_x) - f_x\,\mathbf{r}_x^{ji}\bar{\mathbf{X}}_p}{-\mathbf{t}_z^{ji}\,(u_j - c_x) + f_x\,\mathbf{t}_x^{ji}} \tag{8}$$

where $\mathbf{r}_z^{ji}$ and $\mathbf{r}_x^{ji}$ are the third and first row of the rotation $\mathbf{R}_{ji}$, $\mathbf{t}_z^{ji}$ and $\mathbf{t}_x^{ji}$ are the third and first elements of the translation $\mathbf{t}_{ij}$, $\bar{\mathbf{X}}_p = \mathbf{K}^{-1}\,\mathbf{x_p}$ is the unary ray trough pixel p as seen in Fig. 3, being $\mathbf{K}$ the calibration matrix, and f_x and c_x are the focal and the principal point. Using equation (8) we form the inverse depth hypothesis $\mathcal{N}(\rho_j, \sigma_{\rho_j}^2)$, as follows:

$$\begin{gathered}\rho_j = \rho_p(u_0^*) \\ \sigma_{\rho_j} = \max(|\rho_p(u_0^* + \sigma_{u_0^*}) - \rho_j|, |\rho_p(u_0^* - \sigma_{u_0^*}) - \rho_j|)\end{gathered} \tag{9}$$

Note that our uncertainity propagation is general, in contrast to the assumption of small rotations in [2].

C. Inverse Depth Hypothesis Fusion

At this point we have a set of inverse depth hypotheses for the pixel p. The number of hypotheses can be less than N as the epipolar line segment between $\rho_{\min}$ and $\rho_{\max}$ could lie entirely out of some of the keyframes or no pixel fulfills the conditions described in section III-B. In addition some of the hypotheses can be outliers due to several similar pixels or occlusions. We therefore search for at least λ_N compatible hypotheses. The compatibility between two hypotheses a, b is tested with the χ^2 test at 95%:

$$\frac{(\rho_a - \rho_b)^2}{\sigma_a^2} + \frac{(\rho_a - \rho_b)^2}{\sigma_b^2} < 5.99 \tag{10}$$

Selecting at each time a hypothesis we check the compatibility with the rest of hypotheses. If the best combination gives $n > \lambda_N$ compatible measures, they are fused, yielding the inverse depth distribution $\mathcal{N}(\rho_p, \sigma_{\rho_p}^2)$ for the pixel p:

$$\rho_p = \frac{\sum_n \frac{1}{\sigma_{\rho_j}^2}\rho_j}{\sum_n \frac{1}{\sigma_{\rho_j}^2}}, \quad \sigma_{\rho_p}^2 = \frac{1}{\sum_n \frac{1}{\sigma_{\rho_j}^2}} \tag{11}$$

D. Intra-Keyframe Depth Checking, Smoothing and Growing

After we have computed the semi-dense inverse depth map of the keyframe we perform an outlier removal, smoothing and growing step as proposed in [2]. To retain the inverse depth measurement of a pixel its inverse depth distribution must be supported by at least 2 of its 8 pixel neighbors $p_{i,n}$ as described in (10). The inverse depth of those retained pixels is averaged by their compatible neighbors using (11), but fixing the standard deviation to the minimum of the neighbors. This step smooths the reconstruction, while preserving edges, as only compatible measurements are averaged. Those pixels, in a high gradient area that do not have an inverse depth measurement but are surrounded by at least two pixels with compatible distributions, are also assigned an average inverse depth (with the minimum standard deviation). This grows the reconstruction getting more density.

E. Inter-Keyframe Depth Checking and Smoothing

Once the inverse depth maps of the neighbors of K_i have been computed, we check with them the consistency of each inverse depth distribution in the inverse depth map of K_i. For each pixel p of K_i with an associated inverse depth ρ_p, we project the corresponding 3D point into each neighbor keyframe K_j and propagate the inverse depth as follows:

$$\begin{aligned} \mathbf{x}_j &= \mathbf{K}\,\mathbf{R}_{ji}\,\frac{1}{\rho_p}\,\bar{\mathbf{X}}_p + \mathbf{K}\,\mathbf{t}_{ji} \\ \rho_j &= \frac{\rho_p}{\mathbf{r}_z^{ji}\bar{\mathbf{X}}_p + \rho_p\,\mathbf{t}_z^{ji}} \end{aligned} \tag{12}$$

As the projection $\mathbf{x}_j$ will not coincide with an integer pixel coordinate, we look in the 4 neighbor pixel $p_{j,n}$ around $\mathbf{x}_j$ for a compatible inverse depth as follows:

$$\frac{(\rho_j - \rho_{j,n})^2}{\sigma^2_{\rho_{j,n}}} < 3.84 \tag{13}$$

To retain the inverse depth distribution of a pixel p, at least one compatible pixel $p_{j,n}$ must be found in at least λ_N neighbor keyframes.

Finally we perform a gauss-newton step that minimizes the depth difference in all compatible pixels:

$$d_p^* = \min_{d_p} \sum_{j,n} (d_{j,n} - d_p\,\mathbf{r}_z^{ji}\bar{\mathbf{X}}_p - \mathbf{t}_z^{ji})^2 \frac{1}{d_{j,n}^4 \sigma^2_{\rho_{j,n}}} \tag{14}$$

We optimize the depth instead of its inverse because the propagation equation (12) is linear in depth an the optimal d_p^* is reached in one iteration. The denominator of (14) comes from uncertainity propagation of the inverse depth to depth.

IV. Experimental Evaluation

In this section we present several experiments to show the performance of our system and specifically our semi-dense mapping approach. An accompanying video[2] shows the system operating in real time and several reconstructions.

A. Implementation details

We have performed all experiments in a laptop with an Intel i7-4700MQ processor, which allows to run simultaneously 8 threads. Our feature-based Monocular SLAM and the semi-dense module are implemented in C++ with ROS. The feature-based SLAM system uses 3 threads (the tracking, local mapping and loop closing), while ROS will probably make use of at least 1 additional thread. Therefore in the *online* setting, the semi-dense mapping module makes use of 4 threads for multi-threading optimization. All operations described in section III are independent for each pixel and therefore can be parallelized. The values for the parameters of the semi-dense module were set as follows: $N = 7$, $\sigma_I = 20$, $\lambda_G = 8$, $\lambda_L = 80°$, $\lambda_\theta = 45°$ and $\lambda_N = 3$.

B. The importance of removing outliers

One of the main characteristics of our semi-dense mapping method is that stereo correspondences are searched between keyframes with wide baseline. This can produce the appearance of outliers due to occlusions or multiple similar pixels. Although each inverse depth value has an associated uncertainty, imposing a restrictive variance threshold is not enough to remove outliers that could have similar uncertainty than inliers. This motivated the inclusion of an inter-keyframe depth consistency checking, see section III-E, to detect and remove outliers. Fig. 4 shows a semi-dense reconstruction of a planar scene that consists of several posters on the floor. First row of the figure shows the top and side views of the reconstruction without applying any outlier detection, which results in a solution with many outliers. Second row is the same reconstruction, but retaining only the pixels that have an inverse depth variance below a threshold, which reduces the number of outliers but not completely. Third row is the original reconstruction with the inter-keyframe outlier removal, but without a variance threshold, and almost all outliers have been removed. The best solution is shown in the fourth row with both a variance threshold and the inter-keyframe checking.

C. Accuracy

In our previous work [12] we performed an extensive evaluation of our feature-based monocular SLAM, in terms of keyframe pose accuracy. We used the TUM RGB-D Benchmark [22] as it provided several sequences with accurate ground-truth camera localization from an external motion capture system. Despite the dataset is quite exigent for monocular SLAM (e.g. limited field of view, blur, strong rotations), we achieved a typical RMSE error in the keyframe position around 1cm, and in some sequences as the *fr2_xyz* only 3 mm. We compared our results in 16 sequences with the state-of-the-art direct SLAM, LSD-SLAM [3], and surprisingly we obtained higher accuracy (around 5 times better) and higher robustness, as LSD-SLAM was not able to process all sequences.

To test our semi-dense mapping method we have selected 4 of the sequences in which the camera motion allows to recover a detailed reconstruction. Still those sequences where not recorded with care for semi-dense/dense reconstruction as

[2] https://youtu.be/HlBmq70LKrQ

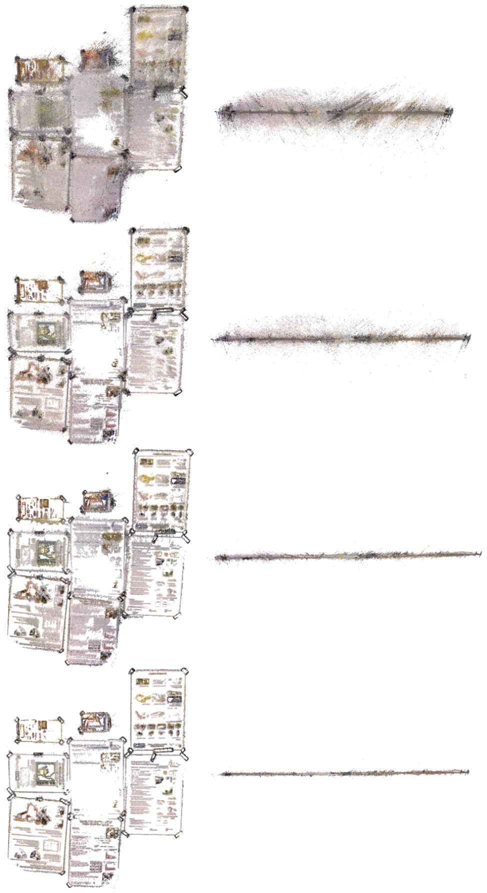


Fig. 4. Example of outlier removal in the sequence *fr2_nostructure_texture_near_with_loop* (TUM RGB-D Benchmark [22]). Description in text.

it was not the goal. An analysis of suitable camera movements for this kind of reconstructions can be found in [4]. Left and middle columns of Fig. 5 shows different viewpoints of each reconstruction (it is recommended to zoom this figure to see the details). It can be seen how the reconstruction contains very few outliers, while the point density is enough to recognize different objects as seen in Fig.6. The accuracy of the reconstruction can be noticed in the straight contours of the the desk in *fr2_desk* (Fig. 5(a)), the scene planarity in *fr3_nostructure_texture_near_with_loop* (Fig. 5(b)), and the readable text in sequence *fr3_structure_texture_near* (Fig. 5(d)). In the sequence *fr3_long_office_household* (Fig. 5(c)) there is a close approximation of the camera to the teddy bear that introduce more error than normal drift accumulation. Therefore the pose graph optimization performed at the loop closure at the end of the sequence cannot completely compensate this error. The result is that the desk contours do not completely align, despite in general the reconstruction being quite accurate. Running the reconstruction offline after full BA yields a perfectly aligned and accurate solution.

Table I shows the median times per keyframe and total reconstruction times for each sequence. It can be seen that the system operates in real-time as the total time spent by the semi-dense mapping module is less than the total sequence length. However it is important to remind that the reconstruction is always with some seconds delay to permit the reconstruction of a keyframe with *future* keyframes and to avoid interferences of the local BA. Variations in time depend mainly on the amount of high gradient pixels in the keyframes.

TABLE I
ONLINE RECONSTRUCTION TIMES FOR THE SEQUENCES OF FIG. 5

Time per Keyframe	5(a)	5(b)	5(c)	5(d)
Inverse Depth Map Estimation (ms)	234	232	268	170
Intra-Keyframe Smoothing (ms)	28	25	31	24
Inter-Keyframe Smoothing (ms)	151	128	154	119
Total (ms)	425	376	451	308
Reconstruction Time (s)	65.2	37.0	67.1	14.3
Sequence Length (s)	98.8	56.6	87.1	37.0

TABLE II
KEYFRAME POSITION ACCURACY COMPARISON

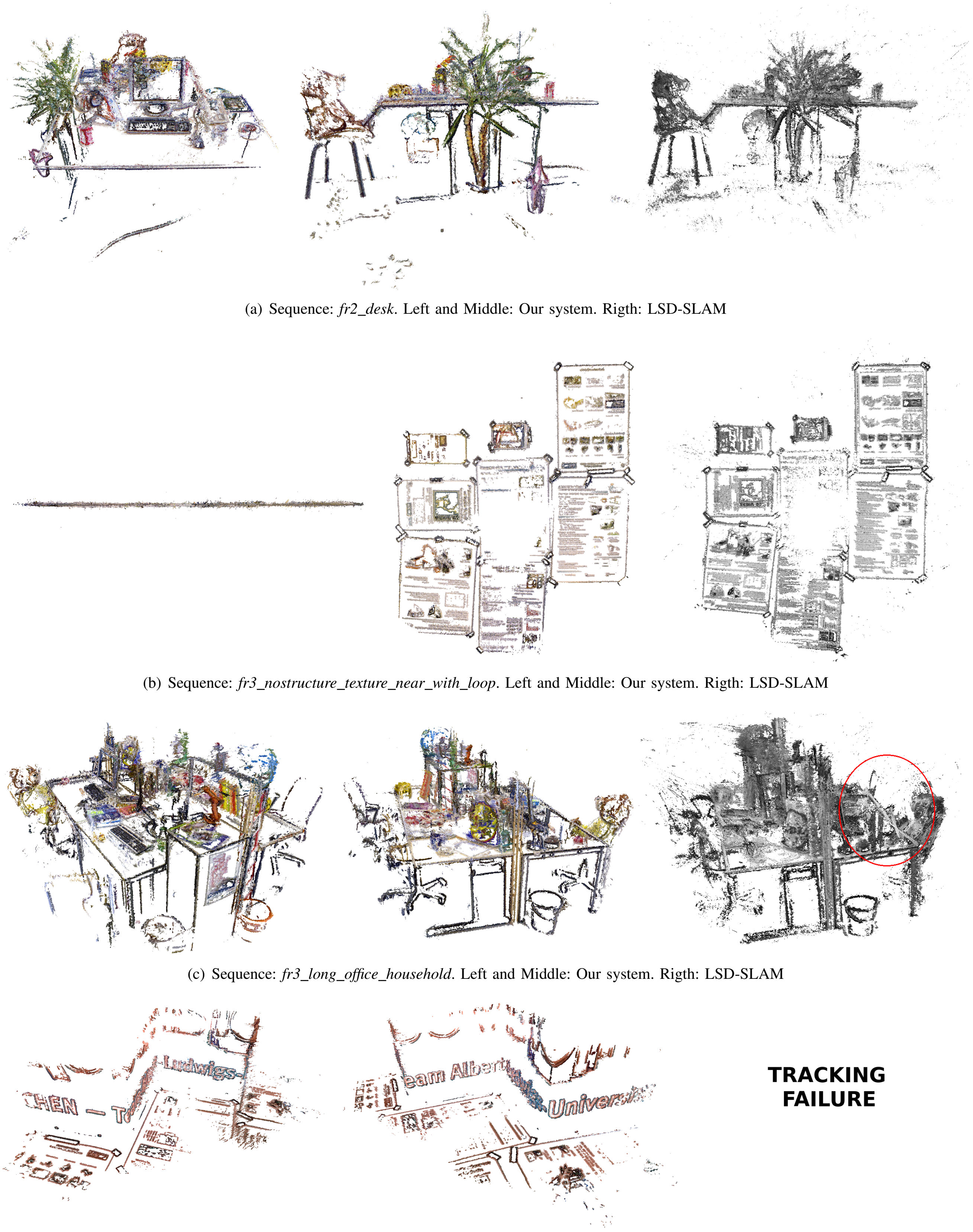
	Absolute Keyframe Trajectory RMSE (cm)		
Sequence of Fig.	ORB-SLAM	PTAM [8]	LSD-SLAM [3]
5(a)	0.88	X	4.57
5(b)	1.39	2.74	7.54
5(c)	3.45	X	38.53
5(d)	1.58	1.04	X
7	0.63	X	31.73

All results correspond to the median over 5 executions. Keyframes and ground truth have been aligned by 7 DoF, as all systems are monocular and the scale is arbitrary. For LSD-SLAM we have cut off the 10 first keyframes as their initialization takes some time to converge. *X* means tracking failure.

To compare we have executed LSD-SLAM in the same sequences (right column of Fig. 5). Table II shows a comparison of keyframe position error, which shows clearly our better accuracy. The reconstruction of *fr2_desk* is similar to ours. In *fr3_nostructure_texture_near_with_loop* the loop at the end of the sequence is not closed and posters do not perfectly align. The reconstruction of *fr3_long_office_household* is broken in one of the sides of the desk as it is highlighted, because the reconstruction is severely corrupted after the close camera approximation to the teddy bear and the loop closure can only partially mitigate the error. Finally in sequence *fr3_structure_texture_near* the tracking fails after a rotation at the beginning of the sequence.

D. Dynamic Scenes

In this experiment we have run our system in the sequence *fr2_desk_with_person*. This is a desk sequence where a person is moving and changes some object positions. Our SLAM system is robust under those dynamic elements, achieving a RMSE error in the keyframe positions of 6.3mm. Because the semi-dense mapping operates over the keyframes, only objects that have remained static in several keyframes are reconstructed. The whole reconstruction is shown in Fig. 7.



(a) Sequence: *fr2_desk*. Left and Middle: Our system. Rigth: LSD-SLAM

(b) Sequence: *fr3_nostructure_texture_near_with_loop*. Left and Middle: Our system. Rigth: LSD-SLAM

(c) Sequence: *fr3_long_office_household*. Left and Middle: Our system. Rigth: LSD-SLAM

(d) Sequence: *fr3_structure_texture_near*. Left and Middle: Our system. Rigth: LSD-SLAM

Fig. 5. Left and middle columns: semi-dense reconstructions performed by our system in four sequences from the TUM RGB-D Dataset [22]. Rigth: the reconstruction of the state of the art LSD-SLAM [3]. Variance thresholds have been adapted in both systems trying to show the reconstructions as clean as possible from outliers. Reconstructions for LSD-SLAM have been taken from its grayscale visualizer subtracting the background color.

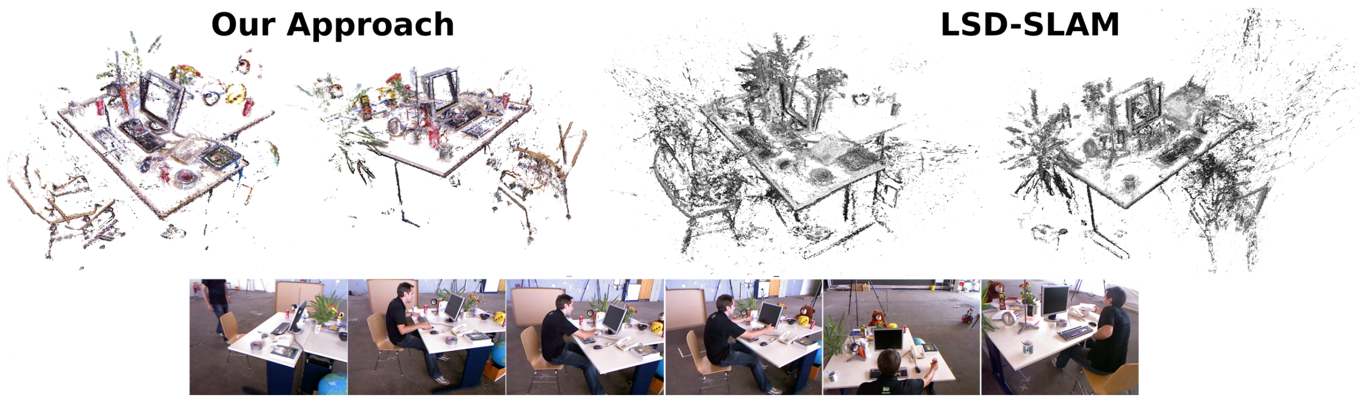


Fig. 7. Example of a dynamic scene. At the bottom it can be seen that there is a person changing object positions. Both our reconstruction and LSD-SLAM are shown for comparison. It is recommended to zoom the images.

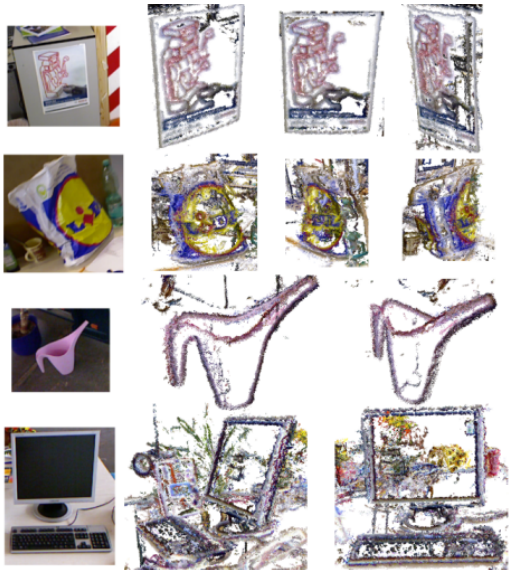


Fig. 6. Example of reconstructed objects that are easily recognizable.

It can be seen that low dynamic changes (final positions of an object) are present in the reconstruction while static elements (e.g. the desk contour) are well defined.

We have also executed LSD-SLAM in this sequence to compare. It can be seen in Table II the low accuracy achieved in this sequence. The reconstruction is also shown in Fig. 7, where the point clouds of the first 30 keyframes of the sequence are not shown as they were very wrongly positioned. Still the overall reconstruction contains many outliers.

V. Discussion

We have presented a novel feature-based monocular SLAM system, which incorporates a probabilistic semi-dense mapping module to perform in real-time, in a conventional computer and without GPU, rich semi-dense reconstructions. The semi-dense mapping operates over keyframes, which are very well localised due to local BA and pose graph optimization at loop closing, allowing to obtain high quality reconstructions. The search of pixel correspondences in wide baseline keyframes motivated a novel inverse-depth hypothesis fusion and an *inter*-keyframe outlier detection mechanism, which checks the depth consistency across keyframes, resulting in clean reconstructions with very few outliers. Our correspondence search and inverse depth uncertainty derivation is based on [2], adding the image gradient modulo and orientation in the comparison, and deriving the equations without narrow baseline assumptions, as we operate on keyframes. Figure 4 showed that our probabilistic uncertainty model and the novel inter-keyframe outlier detection significantly improves the reconstruction quality, irrespective of the keyframe poses, which is one of the main contributions of this paper.

Supported by our experimental evaluation, one of the main claims of this work is that using features, our system is more robust and accurate in the keyframe localisation than direct approaches (using as baseline the recent LSD-SLAM [3]). Features have good invariance to illumination and viewpoint, while direct matching is limited by photometric consistency. Wide baseline matches and large loop closures provide a strong camera network which is essential for BA and pose graph optimization to obtain accurate solutions. In addition features are less affected by auto-gain, auto-exposure and rolling-shutter artifacts. Another key difference is that BA is able to jointly optimize keyframe poses and map reconstruction, while in direct SLAM, due to the computational complexity, keyframe poses are never re-optimized [14], or map optimization is reduced to a pose graph optimization [3].

The main limitation of our approach is that the semi-dense reconstruction is obtained with a few keyframes delay, and it is not used for camera tracking. Future research could focus on densifying the semi-dense reconstruction and incoporating direct (e.g. photometric) terms in our feature-based tracking.

Acknowledgments

This work was supported by the Dirección General de Investigación of Spain under Project DPI2012-32168 and the Ministerio de Educación Scholarship FPU13/04175.

References

[1] Andrew J. Davison, Ian D. Reid, Nicholas D. Molton, and Olivier Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1052–1067, 2007.

[2] Jakob Engel, Jürgen Sturm, and Daniel Cremers. Semi-dense visual odometry for a monocular camera. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1449–1456, 2013.

[3] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *European Conference on Computer Vision (ECCV)*, pages 834–849. Zurich, Switzerland, September 2014.

[4] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. Appearance-based active, monocular, dense reconstruction for micro aerial vehicle. In *Robotics: Science and Systems (RSS)*, Berkeley, USA, July 2014.

[5] Dorian Gálvez-López and Juan D. Tardós. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5):1188–1197, 2012.

[6] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[7] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[8] Georg Klein and David Murray. Parallel tracking and mapping for small AR workspaces. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 225–234, Nara, Japan, November 2007.

[9] Rainer Kuemmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g2o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3607–3613, Shanghai, China, May 2011.

[10] Christopher Mei, Gabe Sibley, and Paul Newman. Closing loops without places. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3738–3744, Taipei, Taiwan, October 2010.

[11] Raúl Mur-Artal and Juan D. Tardós. Fast relocalisation and loop closing in keyframe-based SLAM. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 846–853, Hong Kong, China, June 2014.

[12] Raúl Mur-Artal, J. M. M. Montiel, and Juan D. Tardós. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 2015. To appear.

[13] Richard A. Newcombe and Andrew J. Davison. Live dense reconstruction with a single moving camera. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1498–1505, San Francisco, USA, June 2010.

[14] Richard A. Newcombe, Steven J. Lovegrove, and Andrew J. Davison. DTAM: Dense tracking and mapping in real-time. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2320–2327, Barcelona, Spain, November 2011.

[15] Matia Pizzoli, Christian Forster, and Davide Scaramuzza. REMODE: Probabilistic, monocular dense reconstruction in real time. In *International Conference on Robotics and Automation (ICRA)*, pages 2609–2616, Hong Kong, China, June 2014.

[16] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. ORB: an efficient alternative to SIFT or SURF. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2564–2571, Barcelona, Spain, November 2011.

[17] Mike Smith, Ian Baldwin, Winston Churchill, Rohan Paul, and Paul Newman. The new college vision and laser data set. *The International Journal of Robotics Research*, 28(5):595–599, 2009.

[18] Hauke Strasdat, J. M. M. Montiel, and Andrew J. Davison. Scale drift-aware large scale monocular SLAM. In *Robotics: Science and Systems (RSS)*, Zaragoza, Spain, June 2010.

[19] Hauke Strasdat, Andrew J. Davison, J. M. M. Montiel, and Kurt Konolige. Double window optimisation for constant time visual SLAM. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2352–2359, Barcelona, Spain, November 2011.

[20] Hauke Strasdat, J. M. M. Montiel, and Andrew J. Davison. Visual SLAM: Why filter? *Image and Vision Computing*, 30(2):65–77, 2012.

[21] Jan Stühmer, Stefan Gumhold, and Daniel Cremers. Real-time dense geometry from a handheld camera. In *Proc. 32nd Annual Symp. German Association for Pattern Recognition (DAGM)*, pages 11–20, Darmstadt, Germany, September 2010.

[22] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 573–580, Vilamoura, Portugal, October 2012.

[23] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment a modern synthesis. In *Vision algorithms: theory and practice*, pages 298–372. 2000.