2016 IEEE International Conference on Robotics and Automation (ICRA)
Stockholm, Sweden, May 16-21, 2016

Robust Stereo Visual Odometry through a Probabilistic Combination of Points and Line Segments

Ruben Gomez-Ojeda, Javier Gonzalez-Jimenez

Abstract— **Most approaches to stereo visual odometry reconstruct the motion based on the tracking of point features along a sequence of images. However, in low-textured scenes it is often difficult to encounter a large set of point features, or it may happen that they are not well distributed over the image, so that the behavior of these algorithms deteriorates. This paper proposes a probabilistic approach to stereo visual odometry based on the combination of both point and line segment that works robustly in a wide variety of scenarios. The camera motion is recovered through non-linear minimization of the projection errors of both point and line segment features. In order to effectively combine both types of features, their associated errors are weighted according to their covariance matrices, computed from the propagation of Gaussian distribution errors in the sensor measurements. The method, of course, is computationally more expensive that using only one type of feature, but still can run in real-time on a standard computer and provides interesting advantages, including a straightforward integration into any probabilistic framework commonly employed in mobile robotics.**

I. Introduction

In recent years, visual odometry (VO) has gained importance in robotics applications such as ground vehicles moving on uneven terrains, or unmanned aerial vehicles (UAVs). An alternative to VO in these cases is the use of inertial measurement units (IMUs), but they are not able to cancel the gravity effects precisely, accumulating large errors over time. Traditional solutions also include wheel odometry, which cannot replace VO since it only works with smooth and planar movements, and GPS-based navigation systems, which are limited to open outdoor environments and are unable to estimate the orientation of the device they are attached to. An additional advantage of VO is that the information required (provided by cameras) can be exploited for other navigation-related tasks such SLAM [1] and scene recognition. Visual odometry can be addressed with a single camera [2] [3] [4], stereo cameras [5], or RGB-D sensors [6] [7]. Moreover, two methodologies have been considered in the literature: appearance-based and feature-based. The first group, known as dense approach, works on the whole image assuming some kind of photo-consistency between the successive frames [8] [9]. An alternative strategy consists of matching some relevant features (either points or lines) in the images, and then estimates the pose increments by establishing some rigid-body constraints between those features. Most visual odometry systems are based on feature points, since they are easily detectable and matchable. Some remarkable works following this approach are [10] and [11]. In the former, the authors report a stereo visual odometry (SVO) system based on an iterative estimation of the 6DoF camera motion. The point features are detected with a variant of the Harris corner detector and matched according to their normalized correlation. In the latter, the authors propose an algorithm which employs point features in combination with a sparse feature matcher to reconstruct the 3D pose of a stereo camera given a sequence of images. Those methods have proven to work fast and robustly in many environments, but their behavior in low textured scenes, such as the one in Figure 1(a), deteriorates since it is difficult to find a large set of reliable points. In contrast, line segments are usually abundant in any human-made environment, even in low textured scenes, but these methods are not so common in literature since the detection of lines involves a high computational cost. In that context, Witt and Weltin [12] proposed the Iterative Closest Multiple Lines (ICML) algorithm, where the Iterative Closest Point (ICP) al-

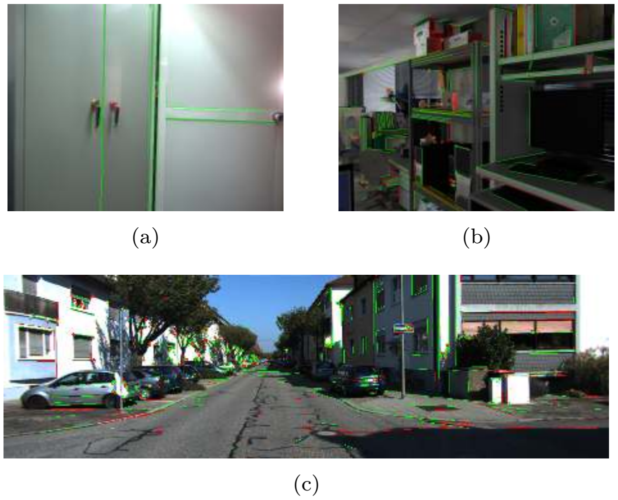

(a)
(b)
(c)

Fig. 1. (a) Texture-less scenes are challenging for traditional point-based SVO approaches.(b) Synthetic frame extracted from the Tsukuba dataset. (c) Frame extracted from the KITTI dataset, where both point and line features are abundant.

Mapir Group. Universidad de Málaga. E.T.S. Ingeniería de Informática-Telecomunicación. Campus de Teatinos, 29071, Málaga, Spain. E-mail: rubengooj@gmail.es

This work has been supported by the project "PROMOVE: Advances in mobile robotics for promoting independent life of elders", funded by the Spanish Government and the "European Regional Development Fund ERDF" under contract DPI2014-55826-R.

978-1-4673-8025-6/16/$31.00 ©2016 IEEE

gorithm is adapted to the case of line segments. This approach estimates simultaneously the correct matches and the pose increment by considering one-to-many line matches inside a non-linear optimization process, which works well under the assumption of small rotations. While this proposal yields a good performance in fast video sequences, it has certain tendency to fall into local minima, thus the authors also propose a robust hypothesize-and-test algorithm as a failure detection step. However, in highly textured environments (e.g. outdoor scenes) the number and quality of the detected lines decreases, and, in consequence, the performance of the algorithm. This problem is addressed by Koletschka et al. [13] with a strategy that efficiently combines point and line features, and hence it can work in different environments. They also propose an algorithm for the stereo matching of the line segments which computes the sub-pixel disparity of the endpoints of the line and deals with partial occlusions. None of the above-mentioned proposals takes into account the probabilistic entity of the features employed since they face the SVO problem in a deterministic way. While this alternative has the advantage of being more efficient computationally, the probabilistic treatment of the variables reduces the undesirable effect of noisy measurements in the optimization, and allows the estimated variables (poses and landmarks) to be easily integrated in probabilistic frameworks which are commonly used in mobile robotics. In this paper we propose a complete probabilistic SVO system that works robustly in different environments thanks to the combination of both points and line segments, which usually provide complementary information. The incremental pose of the stereo camera is recovered iteratively through probabilistic on-manifold optimization of the projection errors, which are computed between the projected features from the first frame and those detected in the second frame. We estimate the uncertainty of all the variables involved in the stereo process, which are assumed to follow Gaussian distributions, and then introduce them as weights in the cost function minimization, increasing the robustness to noise and yielding more accurate results. The source code of the developed C++ Stereo Visual Odometry library is available online, and will be updated as research progresses. An illustrative video of our SVO system and the source code can be found here: http://mapir.isa.uma.es

II. System Overview

In a nutshell, we track the features (points and segments) in a sequence of stereo frames and compute their 3D position and their associated uncertainty. The 3D landmarks are then projected to the new camera pose, where an error function is minimized in order to end up with both the pose increment of the camera and the uncertainty of this estimation. In the following we introduce each step of the SVO system and describe the most important details of its implementation.

1) Point Features: For dealing with feature points, we employ the ORB [14] detector and descriptor due to its efficiency and good performance. In order to reduce the number of outliers, we only consider the measurements that are mutual best matches, i.e. the best match in the left image corresponds to the best match in the right one. To ensure that the correspondences are meaningful enough, we also check that the distance in the description space between the two closest matches is above certain threshold, which is set to the double of the distance of the best match. We also ensure a fair distributions of points over the input images with a bucketing approach that divides the image in 16 buckets, and tries to add at least 20 features in each.

2) Line Segment Features: The line segments are detected with the Line Segment Detector (LSD) [15], which has a high precision and repeatability. However, it is time consuming, which is its major weakness for real-time applications. To mitigate this, we detect the line segments in a parallelized framework in both stereo images. For the stereo matching and frame-to-frame tracking we first compute the LBD descriptors [16] for each line, and match them based on their local appearance features. Similarly to the case of points, we check that both features are mutually best matches, and also that the best two matches are sufficiently separated in the description space. We have not applied a bucketing strategy in the line segments detection, since it provides less reliable features and hence yields poorer results.

3) Motion Estimation: Once the features have been tracked from a stereo frame to the next one, the line segment endpoints and the feature points are back-projected. Then, the motion is estimated iteratively through a probabilistic Gauss-Newton minimization of the line and point projection errors. The negative effect of incorrect correspondences is reduced by employing a Pseudo-Huber loss function to detect and remove the outliers, as proposed in [17]. The complete process will be detailed in Section III.

4) Uncertainty Propagation: In order to improve the precision of the incremental pose estimation, we weight them with the inverse of the error covariance matrix. This covariance matrix is obtained by propagating the feature errors which are assumed to be zero-mean Gaussian distributed (a common hypothesis in computer vision). Ultimately, this propagation process ends up with the uncertainty of the estimated pose, which makes our system suitable to be easily integrated in any probabilistic robotic algorithm. The error distributions will be described and validated in Section IV.

III. Combined Stereo Visual Odometry

The straightforward approach to compute the camera motion in a SVO system minimizes the error of the 3D features reconstructed from two consecutive stereo frames (i.e. 3D error minimization as in [18]). This procedure has the advantage of a closed form solution but, in practice, it is not the best option since it is strongly affected by the euclidean errors induced by the noisy measurements of the features, which may lead to large motion error in the estimated odometry. Instead, a more precise approach is that of projecting the 3D points (the endpoints in the case of line features) from the first frame to the second one, thus the motion is obtained by 2D error minimization of the features in the image.

A. Problem Statement

Let C and C' be the stereo coordinate systems (typically placed at the left camera) at two consecutive poses, related by the relative transformation $\mathbf{T}(\boldsymbol{\xi}) \in SE(3)$, where $\boldsymbol{\xi} \in \mathfrak{se}(3)$ is the 6-vector of coordinates in the Lie algebra $\mathfrak{se}(3)$. The problem we face is that of estimating the optimal $\mathbf{T}(\boldsymbol{\xi}^*)$ that minimizes the projection error for points and line segments (expressions (1) and (2) below, respectively) under the hypothesis that the measurements are affected by unbiased Gaussian noise (as modeled in Section IV). The stereo camera is assumed to be in an ideal configuration with a baseline b, and the calibration parameters $\mathbf{K}$ are either provided by the manufacturer or known from previous calibration. The point projection error $\Delta\mathbf{p}_i(\boldsymbol{\xi})$ is given by:

$$\Delta\mathbf{p}_i(\boldsymbol{\xi}) = \hat{\mathbf{p}}_i(\boldsymbol{\xi}) - \mathbf{p}'_i \tag{1}$$

with $\mathbf{p}'_i$ being the i-th detected point in the second frame, and $\hat{\mathbf{p}}_i(\boldsymbol{\xi})$ the projected point from the first frame to the second one, both in homogeneous coordinates. With that notation, we define the line equation in the second image $\mathbf{l}'_j$ as the cross product between the endpoints of the line in homogeneous coordinates, denoted as $\mathbf{p}'_j$ and $\mathbf{q}'_j$ respectively. The line projection error is defined as a vector formed by the euclidean distances from the projected endpoints of the line segments in the first frame and the line detected in the second frame, i.e.:

$$\Delta\mathbf{l}_j(\boldsymbol{\xi}) = \left[\mathbf{l}'^{\top}_j \cdot [\hat{\mathbf{p}}_j(\boldsymbol{\xi}) \;\; \hat{\mathbf{q}}_j(\boldsymbol{\xi})]\right]^{\top} \tag{2}$$

where $\hat{\mathbf{p}}_j(\boldsymbol{\xi})$ and $\hat{\mathbf{q}}_j(\boldsymbol{\xi})$ refer to the projected endpoints, and $\mathbf{l}'_j$ is the j-th infinite line detected in the second frame.

B. On-Manifold Optimization

The optimal pose increment $\mathbf{T}(\boldsymbol{\xi}^*)$ is computed through an iterative minimization of the Maximum Likelihood Estimator (MLE) which selects the model $\boldsymbol{\xi}^*$ for which the probability of the observed data becomes maximum. Under the assumption that the data is corrupted by unbiased Gaussian noise, the MLE coincides with the following non-linear least-squares estimator:

$$\begin{aligned}\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}}{\operatorname{argmin}}\Bigg\{ &\sum_{i}^{N_p} \Delta\mathbf{p}_i(\boldsymbol{\xi})^{\top}\boldsymbol{\Sigma}^{-1}_{\Delta\mathbf{p}_i}\Delta\mathbf{p}_i(\boldsymbol{\xi}) \\ &+ \sum_{j}^{N_l} \Delta\mathbf{l}_j(\boldsymbol{\xi})^{\top}\boldsymbol{\Sigma}^{-1}_{\Delta\mathbf{l}_j}\Delta\mathbf{l}_j(\boldsymbol{\xi})^{\top}\Bigg\}\end{aligned} \tag{3}$$

where N_p and N_l corresponds to the number of point and line correspondences respectively, and the matrices $\boldsymbol{\Sigma}^{-1}_{\Delta\mathbf{p}_i}$ and $\boldsymbol{\Sigma}^{-1}_{\Delta\mathbf{l}_i}$ are the 2×2 inverse of the covariance matrices for each type of feature. We calculate the optimal solution through iterative Gauss-Newton optimization on the manifold tangent space $\mathfrak{se}(3)$. In this case, the Jacobian matrix is expressed as follows:

$$\mathbf{J}(\boldsymbol{\xi}) = \left.\frac{\partial\mathbf{E}(\boldsymbol{\xi}\oplus\boldsymbol{\varepsilon})}{\partial\boldsymbol{\varepsilon}}\right|_{\boldsymbol{\varepsilon}=\mathbf{0}} \tag{4}$$

where the vector $\mathbf{E}$ contains both line and point projection errors, and the operator $\oplus : \mathfrak{se}(3)\times\mathfrak{se}(3)\mapsto\mathfrak{se}(3)$ is a generalization of the normal addition operator for Euclidean spaces. For further details on the mathematics, please refer to [19].

C. Fast Outlier Rejection

Due to inaccuracies in the feature detection and tracking process the presence of outliers in the observed data is unavoidable, which leads the optimization process to unreliable results. Besides, the assumption of Gaussian distribution errors renders the system to be highly vulnerable to outliers. In order to deal with this phenomena we have implemented a variant of the ERODE outlier detector [17], which performs a fast and efficient outlier removal based on radial distributions, due to its computational performance. Concretely, we have employed the Cauchy loss function to robustify the MLE:

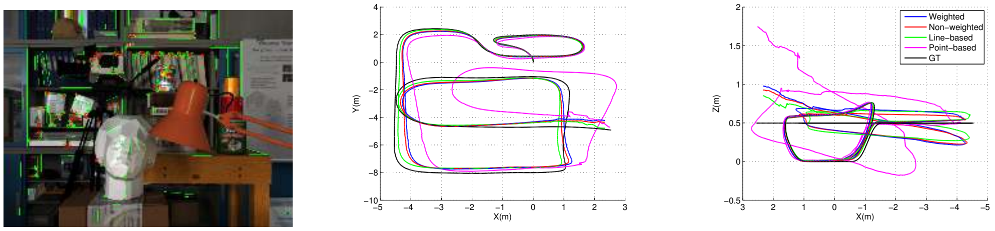
$$\rho(s) = log(1+s) \tag{5}$$

where the input s corresponds to each component of the error vector in (3). With this function, the minimization process converges to the true solution, and after a few iterations, the outliers can be easily detected and removed as they present large residues. Finally, the minimization process is relaunched with the inliers to obtain the optimal solution.

IV. Uncertainty of the Error Functions

The advantage of combining different types of features of the scene, namely points and line segments, relies on their proper weighting in the cost function, which in turn comes from their observation errors. Specifically, this is implemented in the optimization process by weighting the measurements with the inverse of the uncertainty of the projection error from each feature, as expressed in (3) with the matrices $\boldsymbol{\Sigma}^{-1}_{\Delta\mathbf{p}_i}$ and $\boldsymbol{\Sigma}^{-1}_{\Delta\mathbf{l}_j}$. These matrices, which are intended to account for errors in image quantization and in the detection process, are obtained by estimating the Jacobians of the error functions (equations (1) and (2)) with respect to the observations $\mathbf{x}$, which includes both point and line segment observations $\mathbf{p}_i$ and $\mathbf{l}_j$ respectively, i.e.:

$$\boldsymbol{\Sigma}_{\Delta k_i} \approx \frac{\partial\Delta k_i}{\partial\mathbf{x}}\boldsymbol{\Sigma}_{\mathbf{x}}\frac{\partial\Delta k_i}{\partial\mathbf{x}}^{\top} \tag{6}$$



(a)

(b)

(c)

Fig. 2. New University of Tsukuba Stereo Dataset. (a) Line and point matches after visual odometry estimation (the outliers and inliers are plotted in red and green respectively). (b) Top view of the trajectory. (c) Side view of the trajectory.

where the subindex $k \in \{\mathbf{p}, \mathbf{l}\}$ refers to the type of the error function (points or lines). The observation uncertainties $\mathbf{\Sigma_x}$ are modeled as bi-dimensional Gaussians with standard deviations $\sigma_x = \sigma_y = 1$ pixel in the image plane, for both points and endpoints of the line segments. As stated in [20], the uncertainty of the optimal pose is approximated by the inverse of the Hessian of the cost function in (3), expressed as

$$\mathbf{\Sigma}_{\boldsymbol{\xi}^*} \approx \left(\mathbf{J}(\boldsymbol{\xi}^*)^\top \mathbf{W}(\boldsymbol{\xi}^*) \mathbf{J}(\boldsymbol{\xi}^*)\right)^{-1} \tag{7}$$

where $\mathbf{J}(\boldsymbol{\xi}^*)$ is the full Jacobian in (4) that contains both point and line error functions, and $\mathbf{W}(\boldsymbol{\xi}^*)$ is a block-diagonal matrix containing the uncertainty of each projection error for each type of feature. Then, the camera pose increment follows a 6D normal distribution with mean the optimal pose $\boldsymbol{\xi}^*$, and covariance matrix $\mathbf{\Sigma}_{\boldsymbol{\xi}^*}$

$$\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{\xi}^*, \mathbf{\Sigma}_{\boldsymbol{\xi}^*}) \tag{8}$$

A. *Detecting ill-Pose Configurations*

For some spatial distributions the problem may be ill-posed. Such situations can not be detected before the optimization process, since it also depends on the relative motion of the camera. However, this information can be derived from the covariance matrix $\mathbf{\Sigma}_{\boldsymbol{\xi}^*}$. If we express this matrix in diagonal form, their elements give us the variance of the estimated motion parameters $\boldsymbol{\xi}^*$ in the space of the eigenvectors. This information can be employed to neglect those motion terms whose uncertainty is too high. This strategy is very useful when data from other sources, such as IMUs, GPS, wheel odometry, etc., are available, and that information can be fused with our SVO estimation leading to more robust solutions.

V. Experimental Validation

In this section we illustrate the benefits of the weighted combination of points and segments. For that, we estimate the trajectory of a stereo camera in several video sequences acquired in different environments.

A. *Video Sequences*

1) Tsukuba dataset: In this experiment we employ the New University of Tsukuba Stereo Dataset [21] (Figure 2(a)), which contains 1800 stereo pairs from a synthetic laboratory scenario for different illumination conditions. The stereo camera performs a 3D trajectory (about 50 meters length) over the laboratory scene, with several changes in orientation. We compare the accuracy of our SVO system, with the one that does not weight the measurements, and also check the advantages of combining point and line features by representing the trajectories obtained with one type of feature. Figures 2(b) and 2(c) depict both the top and side view of the estimated trajectories with the ground truth. During most of the sequence both estimations (the weighted and non-weighted) show a high accuracy, however, the non-weighted method presents a small superiority since this is a noise-free synthetic dataset, and therefore the uncertainty of the measurements is almost negligible. In the final part of the scene there is a door which induces a lateral drift into the non-weighted method, while the weighted trajectory keeps smooth. The reason for that is an increase in the number of bad measurements, whose negative effect in the quality of the estimation is avoided thanks to the employment of the uncertainty and also the Pseudo-Huber loss function in the process of detecting and removing the outliers. We also observe a superior behavior of the line-based algorithm with respect to the point-based, but obviously the solution which employs both features presents a better performance since that combination provides more information to the system which increases its accuracy.

2) KITTI dataset: We also have used the KITTI benchmark [22], which provides accurate ground truth based on a Velodyne laser scanner and a GPS localization system. The stereo camera rig is formed by two gray-scale Point Grey Flea®2 video cameras separated with a baseline of 54 cm attached to the top of a car. As already mentioned, the introduction of proper weights for the different features in the optimization improves the accuracy of the estimated trajectory, since it limits the influence of those landmarks with high uncertainty. For checking that, we compare the results obtained with our strategy, that weights the features with their uncertainty, with one that does not employ this information (non-weighted approach). We also compare it with implementations that consider only one type of feature: either points or line segments. Figure 3 plots the distributions of the relative pose errors (RPE) [23] of the rotation and translation components between all camera pose increments for the test sequences of the KITTI dataset. This chart confirms

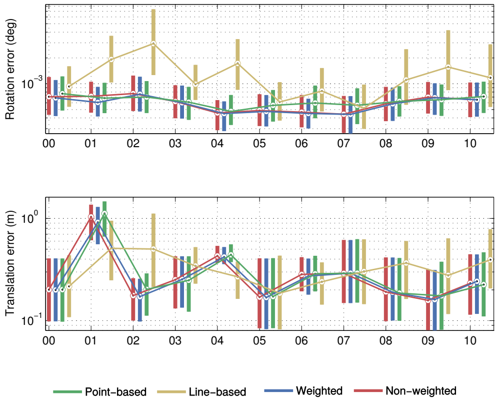


Fig. 3. RPE distributions of the SVO system in the KITTI dataset sequences, comparing the performance of the combined weighted (in blue) and non-weighted (in red) methods, and also the point (in green) and line-based (in yellow) methods.

the superior performance of both combined methods, which works well in most scenes while the behavior of both point and line-based systems is irregular since they are more influenced by the structure of each environment. In general, the point-based approach is superior to the line-based in the KITTI dataset, since it is a highly textured dataset where most lines found do not provide enough information to recover the 6D pose. It also can be noticed a slight out-performance of the weighted method. However, the major benefits of including the uncertainty in the optimization process can be observed visually in Figure 4. The top views of the 3D estimated trajectories and the ground truth are represented in Figures 4(b) and 4(c)Figure 4(d), while a frame from the scene is plotted in Figure 4(a). We observe a good performance of the weighted method during the three sequences, while the non-weighted algorithm suffers from a big error in the rotation estimation of the sequence KITTI-01 that deviates the trajectory from the ground truth. This is caused by a series of noisy measurements during the medium part of the sequence, that induces the non-weighted method to a poor estimation of the camera motion while the weighted methods, even those which employ only one type of feature, are capable of inhibit the influence of these bad landmarks thanks to the uncertainty weighting.

B. Comparison in the KITTI Vision Benchmark

In this section we compare the performance of our SVO method with several state-of-art algorithms in the evaluation sequences from the KITTI dataset. A deeper comparison would test the performance of our method against those in [12] and [13] in various environments, since both approaches employ line segment features in the pose estimation, however, we have not found any public implementation of them. Table I shows the results of several feature-based VO algorithms, as reported in the KITTI benchmark website, which unlike previous experiments it measures the accumulated trajectory error. Although the performance of our method is slightly inferior in terms of relative translation errors, with an error of 3.26% against the 2.44% of VISO2S [24], and relative rotation errors, with errors of 0.0095 deg/m in comparison with the 0.0077 deg/m of TGVO [25], its main advantage is the robust performance in noisy and low-textured scenarios, when most point-based methods usually fails [12].

TABLE I

Comparison of several VO systems in the KITTI Vision Benchmark.

Method	Tran.(%)	Rot.(deg/m)	Time(s)
Ours	3.26	0.0095	0.20
VISO2S [24]	2.44	0.0114	0.05
TGVO [25]	2.94	0.0077	0.06
VO3ptLBA [26]	3.13	0.0104	0.57
VISO2M+GP [24] [27]	7.46	0.0245	0.15
VISO2M [24]	11.94	0.0234	0.10

C. Processing Time

In this section we first compare the average execution times of both weighted and non-weighted approaches. Table II shows the average computation times and number of correspondences per frame, for the stereo sequences with different resolutions. It can be noticed a slight increment in the execution time of the weighted approach due to the computation of the weights, which can be perfectly assumed by most applications in mobile robotics. We also analyze the influence of the image resolution in the processing time. First, we observe that our stereo visual odometry system runs in average with frequencies of 12 Hz for 640 × 480 images, with a high number of correspondences processed (an average number of 96 lines and 78 points). The proposed SVO system can work with frequencies superior to 30 Hz when the resolution is set to lower values (320×240). In that case, since the average number of detected line correspondences decreases, the accuracy of the camera pose estimation may drop.

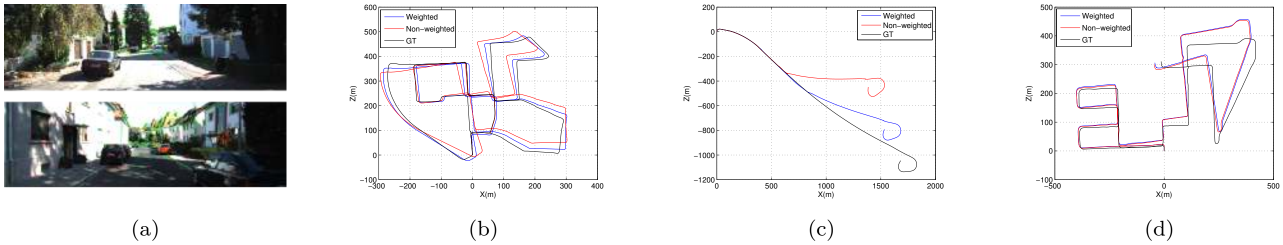
VI. Conclusions

In this paper we have introduced a novel stereo visual odometry system based on points and line features, thus capable of working in different environments. For effectively combining them we take into account the uncertainty of the measurements, which improves the accuracy of the estimation. Besides, the probabilistic distribution of the poses provided by our algorithm can be implemented in any probabilistic framework commonly adopted in robotic applications. In addition, we have confirmed the theoretical results through a series of real experiments in both synthetic and real environments, by estimating the trajectory of different stereo cameras. Future work will focus on improving the performance of our SVO system by introducing a different weighting that will reduce the impact that bad measurements or mobile objects has in the algorithm.

TABLE II

Average number of correspondences and processing times per frame.

				Weighted		Non-weighted	
Dataset	Resolution	Lines	Points	Frequency	Runtime	Frequency	Runtime
Tsukuba	320 × 240	45 lines	40 points	31.40 Hz	31.85 ms	32.01 Hz	31.24 ms
Tsukuba	640 × 480	96 lines	78 points	12.04 Hz	83.05 ms	12.38 Hz	80.76 ms
KITTI	613 × 185	54 lines	60 points	21.47 Hz	46.57 ms	22.00 Hz	45.46 ms
KITTI	1226 × 370	75 lines	188 points	4.54 Hz	220.03 ms	4.57 Hz	218.99 ms



(a) (b) (c) (d)

Fig. 4. Sequences extracted from the KITTI benchmark. (a) Line and point correspondences from two different frames after filtering geometrically inconsistent matches. (b) Top view of the KITTI-00 trajectory. The ground truth is represented with black lines, the estimation weighted with the uncertainty with blue lines, and the non-weighted estimation is plotted with red lines. (c) Top view of the KITTI-01 trajectory. (d) Top view of the KITTI-08 trajectory.

References

[1] J. L. Blanco, F. A. Moreno, and J. González-Jiménez, “A constant-time SLAM back-end in the continuum between global mapping and submapping: application to visual stereo SLAM,” *The International Journal of Robotics Research (To appear)*, 2015.

[2] E. Eade and T. Drummond, “Edge landmarks in monocular slam.,” in *BMVC*, pp. 7–16, 2006.

[3] G. Klein and D. Murray, “Improving the agility of keyframe-based slam,” in *Computer Vision–ECCV 2008*, pp. 802–815, Springer, 2008.

[4] J. Engel, J. Sturm, and D. Cremers, “Semi-Dense Visual Odometry for a Monocular Camera,” in *Computer Vision (ICCV), 2013 IEEE International Conference on*, pp. 1449–1456, IEEE, 2013.

[5] B. Kitt, A. Geiger, and H. Lategahn, “Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme,” in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pp. 486–492, IEEE, 2010.

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

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

[8] C. Forster, M. Pizzoli, and D. Scaramuzza, “SVO : Fast Semi-Direct Monocular Visual Odometry,” *IEEE International Conference on Robotics and Automation*, 2014.

[9] J. Engel, T. Sch, and D. Cremers, “LSD-SLAM: Large-Scale Direct Monocular SLAM,” *Eccv*, pp. 1–16, 2014.

[10] D. Nistér, O. Naroditsky, and J. Bergen, “Visual odometry for ground vehicle applications,” *Journal of Field Robotics*, vol. 23, no. 1, pp. 3–20, 2006.

[11] A. Geiger, J. Ziegler, and C. Stiller, “Stereoscan: Dense 3d reconstruction in real-time,” in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pp. 963–968, IEEE, 2011.

[12] J. Witt and U. Weltin, “Robust Stereo Visual Odometry Using Iterative Closest Multiple Lines,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4164—-4171, IEEE/RSJ, 2013.

[13] T. Koletschka, L. Puig, and K. Daniilidis, “MEVO: Multi-environment stereo visual odometry,” *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4981—-4988, sep 2014.

[14] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: an efficient alternative to SIFT or SURF,” in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2564–2571, IEEE, 2011.

[15] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, “LSD: A fast line segment detector with a false detection control,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 4, pp. 722–732, 2010.

[16] L. Zhang and R. Koch, “An efficient and robust line segment matching approach based on LBD descriptor and pairwise geometric consistency,” *Journal of Visual Communication and Image Representation*, vol. 24, no. 7, pp. 794–805, 2013.

[17] F.-A. Moreno, J.-L. Blanco, and J. González-Jiménez, “ERODE: An efficient and robust outlier detector and its application to stereovisual odometry,” in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pp. 4691–4697, IEEE, 2013.

[18] F. A. Moreno, J. L. Blanco, and J. Gonzalez, “Stereo vision specific models for particle filter-based {SLAM},” *Robotics and Autonomous Systems*, vol. 57, no. 9, pp. 955–970, 2009.

[19] J.-l. Blanco, “A tutorial on SE (3) transformation parameterizations and on-manifold optimization,” no. 3, 2013.

[20] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[21] M. Peris, A. Maki, S. Martull, Y. Ohkawa, and K. Fukui, “Towards a simulation driven stereo vision system,” in *Pattern Recognition (ICPR), 2012 21st International Conference on*, pp. 1038–1042, IEEE, 2012.

[22] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? The KITTI vision benchmark suite,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 3354–3361, IEEE, 2012.

[23] 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)*, 2012.

[24] A. Geiger, J. Ziegler, and C. Stiller, “Stereoscan: Dense 3d reconstruction in real-time,” in *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pp. 963–968, IEEE, 2011.

[25] B. Kitt, A. Geiger, and H. Lategahn, “Visual odometry based on stereo image sequences with ransac-based outlier rejection scheme,” in *Intelligent Vehicles Symposium (IV), 2010 IEEE*, pp. 486–492, IEEE, 2010.

[26] P. F. Alcantarilla, J. J. Yebes, J. Almazán, and L. M. Bergasa, “On combining visual slam and dense scene flow to increase the robustness of localization and mapping in dynamic environments,” in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pp. 1290–1297, IEEE, 2012.

[27] S. Song and M. Chandraker, “Robust scale estimation in real-time monocular sfm for autonomous driving,” in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 1566–1573, IEEE, 2014.