Weighted Local Bundle Adjustment and Application to Odometry and Visual SLAM Fusion

Alexandre Eudes, Sylvie Naudet-Collette, Maxime Lhuillier, Michel Dhome

Weighted Local Bundle Adjustment and Application to Odometry and Visual SLAM Fusion

Alexandre Eudes[12]
alexandre.eudes@lasmea.univ-bpclermont.fr
Sylvie Naudet-Collette[2]
sylvie.naudet@cea.fr
Maxime Lhuillier[1]
maxime.lhuillier@lasmea.univ-bpclermont.fr
Michel Dhome[1]
michel.dhome@lasmea.univ-bpclermont.fr

[1] LASMEA
UMR 6602 Blaise Pascal
University/CNRS
Aubière, France.

[2] Vision and content engineering
Laboratory
CEA, LIST
Gif-sur-Yvette, France

Abstract

Local Bundle Adjustments were recently introduced for visual SLAM (Simultaneous Localization and Mapping). In Monocular Visual SLAM, the scale factor is not observable and the reconstruction scale drifts as time goes by. On long trajectory, this problem makes absolute localisation not usable. To overcome this major problem, data fusion is a possible solution. In this paper, we describe Weighted Local Bundle Adjustment(W-LBA) for monocular visual SLAM purposes. We show that W-LBA used with local covariance gives better results than Local Bundle Adjustment especially on the scale propagation. Moreover W-LBA is well designed for sensor fusion. Since odometer is a common sensor and is reliable to obtain a scale information, we apply W-LBA to fuse visual SLAM with odometry data. The method performance is shown on a large scale sequence.

1 Introduction

Monocular Visual SLAM is a still active topic in computer vision. A first approach comes from Davison[3] in 2003. He adapts SLAM community techniques based on Kalman filter to input images in SLAM algorithms.

Next, Nister *et al*. [13] take the problem of visual SLAM with the photogrammetric community point of view and adapts Structure from Motion algorithms to be used in real time. There is no optimisation to improve accuracy in the first work of Nister *et al*. Later, in 2006, Mouragnon *et al*. [12] introduce local bundle adjustment (**LBA**) to solve this problem. Even if bundle adjustment based SLAM is more accurate than Kalman based one[14], no covariance is provided in [12]. That makes data fusion not optimal.

Recently, we introduce in [5] uncertainty on the localisation and open the possibility to make fusion from LBA-based visual SLAM. We use weighted local bundle adjustment

(**W-LBA**) only for uncertainty propagation but keep (non weighted) LBA for geometry optimisation.

In monocular SLAM, the scene structure and camera motion are theoretically reconstructed up to an unknown scale factor. This reconstruction can be updated to the real world scale by rescaling with only one extra measurement. Unfortunately, experiments show that the scale factor drifts during long trajectories. This problem is a major limitation in monocular visual SLAM.

One possible solution of this problem is to rely on another sensor to estimate the scale. It involves the fusion of monocular visual information with an other sensor. Odometer seems to be the most adapted sensor: it's a common sensor in robotics and automotive industry and it provides reliable scale information. Karlsson *et al.* (fusion algorithm[11]) and Goncalves *et al.* (vision algorithm[7]) use a particle filter to fuse odometer with distance between robot and visual landmark. Cumani *et al.* [2] reconstruct the scene and camera position by chaining triples of cameras reconstructed by standard SfM techniques. The odometer is then used to correct the scale propagation and the 3D point positions. Their method is similar to the method described in Section 5.1 but additional optimisations are performed in our work.

In this paper, we propose different applications of **W-LBA** to reduce drift in scale. In Section 2 and 3, we recall visual SLAM and local bundle adjustment process. In Section 4, we show that W-LBA can also be used for geometry optimisation and propose another weighting scheme in order to improve efficiency. In Section 5, we use the W-LBA to fuse visual SLAM with odometer measurements. Finally in Section 6, we provide experimental results and compare the proposed solution with previous work.

2 LBA based SLAM

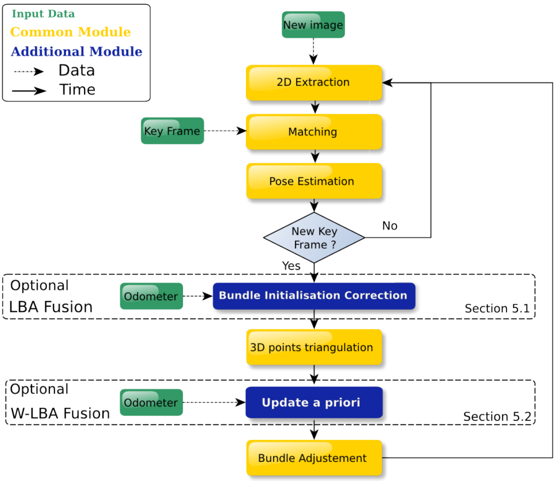

Figure 1: Visual SLAM overview and additional modules

The figure 1 presents an overview of our monocular visual SLAM. The process is similar to the Mouragnon *et al.* [12] implementation. Some additional modules are developed for odometer fusion. They are detailed in Section 5.

The aim of this algorithm is to determine the camera localisation at each image in the reference coordinate frame. In addition, a sparse 3D point cloud is reconstructed to support the iterative localisation process. We assume that the initialisation of the algorithm is done (the first three camera poses and their 3D point cloud are estimated).

Matching 2D. The first step of the method is a 2D tracking of interest points which provides 2D/2D correspondences between consecutive images. Interest points are extracted from each image using Harris detector [9]. Those points are then matched between the two images using SURF[1] descriptor.

Camera pose estimation. With this 2D/2D correspondence, we can associate a previously triangulated 3D point to an observation in the current image. Thanks to those 3D/2D correspondences, the current camera pose is estimated using Grunert's algorithm [8] applied in the RANSAC [6] framework. Since the camera intrinsic parameters are known, only 6 parameters (3 for translation, 3 for rotation) need to be estimated.

Key-frames. They are a sub-sampling of the input video. Indeed, we don't treat all images to ensure sufficient baseline between two frames for triangulation and reduce computational time. A frame is declared as key-frame, if a condition based on the number of 2D/2D correspondences is met. In those special frames, new 3D points are triangulated from the two 2D/2D correspondences between the last 3 key-frames. Last, the result of the previous steps is refined by LBA(Section 3) to optimise the geometry.

3 Bundle Adjustment

3.1 Notation

We denote $\mathscr{C}_{(\mathbf{v})}$ the covariance matrix of the random vector $\mathbf{v}$, $\mathscr{C}_{(\mathbf{u},\mathbf{v})}$ the covariance matrix of the random vector $(\mathbf{u}^\top \mathbf{v}^\top)^\top$ and $\mathscr{C}_{(\mathbf{v}/\mathbf{u})}$ the covariance between vectors $\mathbf{v}$ and $\mathbf{u}$:

$$\mathscr{C}_{(\mathbf{u},\mathbf{v})} = \begin{pmatrix} \mathscr{C}_{(\mathbf{u})} & \mathscr{C}_{(\mathbf{u}/\mathbf{v})} \\ \mathscr{C}_{(\mathbf{v}/\mathbf{u})} & \mathscr{C}_{(\mathbf{v})} \end{pmatrix} \quad (1)$$

Notation $\mathbf{r} \sim \mathscr{N}(\bar{\mathbf{r}}, \mathscr{C}_{(\mathbf{r})})$ means that the Gaussian vector $\mathbf{r}$ has mean $\bar{\mathbf{r}}$ and covariance $\mathscr{C}_{(\mathbf{r})}$.

Camera Poses The camera pose at key-frame t is refined by several LBA over time, therefore to distinguish the pose between refinements, we define for each camera pose a double timing index: $\mathbf{C}_t^{t'}$ represents the pose of key-frame t refined at time t'. We define $\mathbf{c}_t^{t'}$ as the 3D position of the camera $\mathbf{C}_t^{t'}$.

3.2 Global Bundle Adjustment

Bundle adjustment is a well known iterative method [15] designed to solve non-linear least square problems in Structure-from-Motion. It refines simultaneously the scene (3D points cloud) and the camera poses by minimising the reprojection errors:

$$\mathbf{x} \mapsto ||\mathbf{y} - P(\mathbf{x})||^2 \quad (2)$$

where vector $\mathbf{x}$ usually concatenates 6 parameters (3 for rotation,3 for translation) for each camera pose and 3 parameters for each 3D point. $\mathbf{y}$ concatenates interest points detected in images, P concatenates projection functions, and $||.||$ is the Euclidean norm.

Unfortunately, this method becomes very time consuming even with reduction and variable ordering methods [15]. To overcome this limitation, iterative methods based on Bundle Adjustment, called Local Bundle Adjustment, were introduced. Those methods optimise only a limited number of cameras and 3D points. Two versions are detailed below.

3.3 Local Bundle Adjustment (LBA)

In this Section, we recall the Local Bundle Adjustment (**LBA**) presented in [12].

Local Bundle Adjustment is a bundle adjustment such that (1) the estimated parameters $\mathbf{x}$ are the poses of the n most recent key-frames and the list of 3D points which have at least one observation in these frames, (2) the reprojection errors are only considered in the N most recent key-frames ($n < N$).

This LBA $\mathscr{B}$ is concisely written as

$$\mathbf{x} = \mathscr{B}(\mathbf{p},\mathbf{y}) \quad = \quad \text{argmin}_{\tilde{\mathbf{x}}}||\mathbf{y} - P(\mathbf{p},\tilde{\mathbf{x}})||^2. \tag{3}$$

$\mathbf{p}$ contains the poses of the $N-n$ cameras which are not estimated, $\mathbf{x}$ concatenates the n poses and the 3D points which are estimated, and $\mathbf{y}$ is the vector of detected 2D points.

Even in large sequence, LBA estimates only $6n+3s$ parameters (s is the number of 3D points in the n last key-frames). In this way the geometry of long sequence can be iteratively refined in real-time ($n = 3$ and $N = 10$ are commonly used values).

3.4 Weighted Local Bundle Adjustment (W-LBA)

Here we recall the definition of an other LBA, the Weighted Local Bundle Adjustment (**W-LBA**) presented in [5]. W-LBA provides a Maximum Likelihood Estimator (**MLE**) to estimate the structure and camera poses simultaneously. Contrary to the LBA, this estimator takes care of the uncertainty on previous estimated cameras.

As in the LBA formulation, the W-LBA estimates the structure and n camera poses contained in $\mathbf{x}$. The only difference is that N poses are now optimised. The N cameras are split in two, n cameras are freely optimised and contained in $\mathbf{x}$ and the $N-n$ old cameras are in $\mathbf{p}$. The old cameras are considered to be already well optimised, so they are less allowed to move than new one but not fully fixed as in LBA. The constraint on old poses localisation $\mathbf{p}$ is build according to the localisation at the previous step and weighted by a covariance matrix.

This local bundle assumes that $\mathbf{p} \sim \mathscr{N}(\bar{\mathbf{p}}, \mathtt{C}_\mathbf{p})$ and $\mathbf{y} \sim \mathscr{N}(\bar{\mathbf{y}}, \sigma^2\mathtt{I})$ are independent, where $\bar{\mathbf{y}} = P(\bar{\mathbf{p}}, \bar{\mathbf{x}})$. According to this Hypothesis, the MLE $\mathscr{B}^w(\mathbf{p},\mathbf{y})$ of $\bar{\mathbf{p}}$ and $\bar{\mathbf{x}}$ is:

$$\mathbf{x} = \mathscr{B}^w(\mathbf{p},\mathbf{y}) = \ \text{argmin}_{(\tilde{\mathbf{p}},\tilde{\mathbf{x}})} \frac{1}{\sigma^2}||\mathbf{y} - P(\tilde{\mathbf{p}},\tilde{\mathbf{x}})||^2 + (\mathbf{p}-\tilde{\mathbf{p}})^\top \mathtt{C}_\mathbf{p}^{-1}(\mathbf{p}-\tilde{\mathbf{p}}), \tag{4}$$

where σ is the noise on 2D points. In the W-LBA, $\mathtt{C}_\mathbf{p}$ is the "weight" on camera poses. This weighting acts on how the W-LBA takes into account of the old estimated poses $\mathbf{p}$. In previous work[5], we propagate $\mathtt{C}_\mathbf{p}$ through iterations and call it global covariance. Since the old camera poses are re-evaluated accordingly to the *a priori* constraints, this formulation allows easier integration of external information. In Section 4, we introduce the notion of local covariance.

4 Local and Global Covariance

In this Section, we explain the difference between the "global" covariance we defined in [5] and the "local" one we propose here. In the W-LBA, we defined the $\mathtt{C}_{\mathbf{p}}$ in Eq.4 as a global covariance. The global covariance has the property to increase during time and is similar to one computed with global bundle adjustment whereas the local one does not grow and represents uncertainty of camera respectively to the oldest camera in the local bundle.

This covariance is computed with the standard covariance estimation technique[10], by inverting the approximation of the Hessian $(J^{\top}J)^{-1}$ of the projection function (P in Eq.2) minimised by Bundle Adjustment on the N last cameras.

To inverse the Hessian, we need to fix the gauge on the BA [15]. This is done by removing lines of the Jacobian matrix corresponding to fixed parameters. Here, we chose the first frame pose $(\mathtt{R}^{t-N}, \mathbf{t}^{t-N})$ and the largest coordinate of the 3D location of the $(t-n)$-th pose $\mathbf{C}_{t-1}^{t-n}$.

To integrate it in the W-LBA, the local covariance is evaluated on the result of the optimisation done at the previous key-frame. By keeping only the block of $(J^{\top}J)^{-1}$ corresponding to the camera parameters, we evaluate :

$$\mathscr{C}_{(\mathbf{C}_{t-1}^{t-N},\dots,\mathbf{C}_{t-1}^{t-1})}. \tag{5}$$

Then, we remove the first and the $(n-1)$ last poses from Eq.5 to get the $\mathtt{C}_{\mathbf{p}}$:

$$\mathtt{C}_{\mathbf{p}} = \mathscr{C}_{(\mathbf{C}_{t-1}^{t-N+1},\dots,\mathbf{C}_{t-1}^{t-n})}. \tag{6}$$

This new covariance could now be used in the bundle ($\mathscr{B}^w$) to optimise the next step (at time t) using Eq.4. Since the $\mathtt{C}_{\mathbf{p}}$ in W-LBA is a local covariance, we optimise parameters with local constraints.

5 Odometer Integration

In this Section, we present two methods of fusion between odometer and visual SLAM. The first one is a simple modification of the standard visual SLAM process. The second uses the W-LBA with local covariance to integrate odometer constraints in the optimisation process.

We defined d^l the odometer distance between time l and $l+1$. Since in this Section all cameras are evaluated at time t, we omit the second index for improving notation readability.

5.1 LBA Fusion

In this method presented in [4], the idea is to change LBA initialisation by substituting the estimated pose of the current key-frame t by a corrected one.

The estimated 3D location $\mathbf{c}^t$ is replaced by $\tilde{\mathbf{c}}^t$ thanks to the scale correction provided by

$$\tilde{\mathbf{c}}^t = \mathbf{c}^{t-1} + d^{t-1}\frac{\mathbf{c}^t - \mathbf{c}^{t-1}}{||\mathbf{c}^t - \mathbf{c}^{t-1}||_2}, \tag{7}$$

where $||.||_2$ is the Euclidean norm. The camera orientation is kept unchanged.

We integrate this correction after the camera pose estimation in t and before the triangulation of new 3D points. So the new 3D points are triangulated at a corrected scale. Then this new value of pose at time t and the new 3D points are optimised by LBA as in the visual SLAM original process (figure 1).

5.2 W-LBA fusion

W-LBA with local covariance allows to estimate a more accurate position, since all poses in the sliding window are refined. Moreover, a part of the estimated camera (old cameras $\mathbf{p}$) are constrained by previous estimation weighted by covariance $\mathtt{C}_{\mathbf{p}}$. Using this property, this LBA can be used to impose constraint on old cameras ($\mathbf{p}$ in Section 3.4). In our case, we will add constraints coming from the odometer measurement by correcting the constraint on old poses. Thus we replace all the camera $\mathbf{c}$ by $\tilde{\mathbf{c}}$ in $\mathbf{p}$.

We introduce the following notations:

$norm^l = ||\mathbf{c}^{l+1} - \tilde{\mathbf{c}}^l||_2$: Euclidean distance between $\tilde{\mathbf{c}}^l$ and $\mathbf{c}^{l+1}$.
$\mathbf{dir}^l = \frac{\mathbf{c}^{l+1} - \tilde{\mathbf{c}}^l}{norm^l}$: Direction defined by the vector $\tilde{\mathbf{c}}^l\mathbf{c}^{l+1}$.

We do iteratively the same treatment on each pose in $\mathbf{p}$. We initialise the process by: $\tilde{\mathbf{c}}^{t-N+2} = \mathbf{c}^{t-N+2}$ and $\mathscr{C}_{\tilde{\mathbf{c}}^{t-N+2}} = \mathscr{C}_{\mathbf{c}^{t-N+2}}$. Then we update the position and the covariance like described below.

Position update First, we correct the position $\mathbf{c}^l$ thanks to the odometer measurement between l and $l-1$ times. $\mathbf{c}^l$ is replaced by $\tilde{\mathbf{c}}^l$ thanks to the scale correction provided by

$$\tilde{\mathbf{c}}^l = \tilde{\mathbf{c}}^{l-1} + d^{l-1}\mathbf{dir}^{l-1}. \tag{8}$$

The camera orientation is not modified.

Then we need to compute the covariance of cameras ($\tilde{\mathbf{c}}^l$) to take into account the above modification.

Covariance update The covariance matrix of the modified camera is determined thanks to the covariance of the previous pose, the odometer covariance and the transformation previously defined (Eq.8). We define $J_{\tilde{\mathbf{c}}^{l-1}}$,$J_{\mathbf{c}^l}$,$J_{d^{l-1}}$ the Jacobian of Eq.8 for $\tilde{\mathbf{c}}^{l-1}$,$\mathbf{c}^l$,d^{l-1} respectively. They are given by:

$$J_{\tilde{\mathbf{c}}^{l-1}} = I - \frac{d^{l-1}}{norm^{l-1}}(I - \mathbf{dir}^{l-1}\mathbf{dir}^{l-1^\top}) \tag{9}$$

$$J_{\mathbf{c}^l} = \frac{d^{l-1}}{norm^{l-1}}(I - \mathbf{dir}^{l-1}\mathbf{dir}^{l-1^\top}) \tag{10}$$

$$J_{d^{l-1}} = \mathbf{dir}^{l-1} \tag{11}$$

We compute the covariance $\mathscr{C}_{\tilde{\mathbf{c}}^l}$:

$$\mathscr{C}_{\tilde{\mathbf{c}}^l} = J_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1})}\mathscr{C}_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1})}J^\top_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1})} \tag{12}$$

The camera $\mathbf{c}^l$ is not independent of other camera in $\mathbf{p}$. Thus we also need to evaluate the correlation between $\tilde{\mathbf{c}}^l$ with the i-th camera in $\mathbf{p}$. The correlation is computed as follow:

$$i \in [t-N+1...l-1], \mathscr{C}_{(\tilde{\mathbf{c}}^l/\tilde{\mathbf{c}}^i)} = \mathscr{C}^\top_{(\tilde{\mathbf{c}}^i/\tilde{\mathbf{c}}^l)} = J_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1})}\mathscr{C}_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1}/\tilde{\mathbf{c}}^i)} \tag{13}$$

$$i \in [l+1...t-n], \mathscr{C}_{(\tilde{\mathbf{c}}^l/\mathbf{c}^i)} = \mathscr{C}^\top_{(\tilde{\mathbf{c}}^i/\mathbf{c}^l)} = J_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1})}\mathscr{C}_{(\tilde{\mathbf{c}}^{l-1},\mathbf{c}^l,d^{l-1}/\mathbf{c}^i)} \tag{14}$$

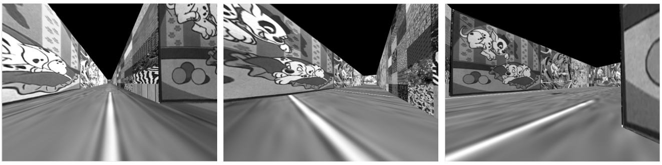

Figure 2: Three images of the synthetic sequence.

When the pose ($\mathbf{p}$) and covariance ($\mathtt{C}_\mathbf{p}$) of all the $(N-n)$ old cameras are updated, we just apply the W-LBA with local covariance instead of LBA in the visual SLAM (figure 1).

6 Experiments

First we demonstrate that the W-LBA is more accurate than LBA and more particularly when local covariance is used. Then, we show that data fusion with W-LBA is possible and that W-LBA fusion outperforms LBA fusion.

The LBA (Section 3.3) is the method used by Mouragnon *et al.* for poses optimisation. Two W-LBA have been presented here: The global W-LBA, W-LBA with global covariance (Section 3.4) is our method ([5]) used to propagate error but not for geometry optimisation. The local W-LBA (Section 4) is our proposal to integrate local covariance in W-LBA.

The LBA fusion (Section 5.1) is the modification of the bundle initialisation([4]). The W-LBA fusion (Section 5.2) is our proposal to fuse odometer with visual SLAM using local W-LBA.

First, trajectories are registered to ground truth, then different errors are computed to compare trajectories ($\mathbf{c}$) to ground truth (GT). We refer to position error as $||\mathbf{c}_t - GT_t||_2$, angular error as $angle(\mathbf{c}_{t-1} - \mathbf{c}_t, GT_{t-1} - GT_t)$ and inter-camera ratio as $\frac{||\mathbf{c}_{t-1} - \mathbf{c}_t||_2}{||GT_{t-1} - GT_t||_2}$. The latter indicates the scale accuracy of the reconstruction. For each error, we give the mean, standard deviation and minimum or maximum values.

6.1 Compare W-LBA and LBA

In this Section, we make a comparison between LBA, global W-LBA and local W-LBA.

Experiments are made on a synthetic sequence of a corridor with textured walls generated by 3D Studio Max. This sequence is composed of 2900 frames and is 365m long with a 1 m/s camera speed. Some images of the sequence images are provided in figure 2.

In figure 3, we compare the trajectories given by LBA (in purple) , local (in blue) and global (in black) W-LBA to the ground truth (in red). The table 1 presents the statistical results of the different methods.

In [5], there is no performance evaluation of the W-LBA bundle, we only give the mathematical formulation and derive the propagation of covariance. We observe that the global W-LBA gives quite similar results than LBA on the reconstruction on figure 3. However, the local W-LBA is more appropriate and gives more accurate results than global W-LBA. It is more accurate in position and has less scale propagation error but we note a minor angular error increase.

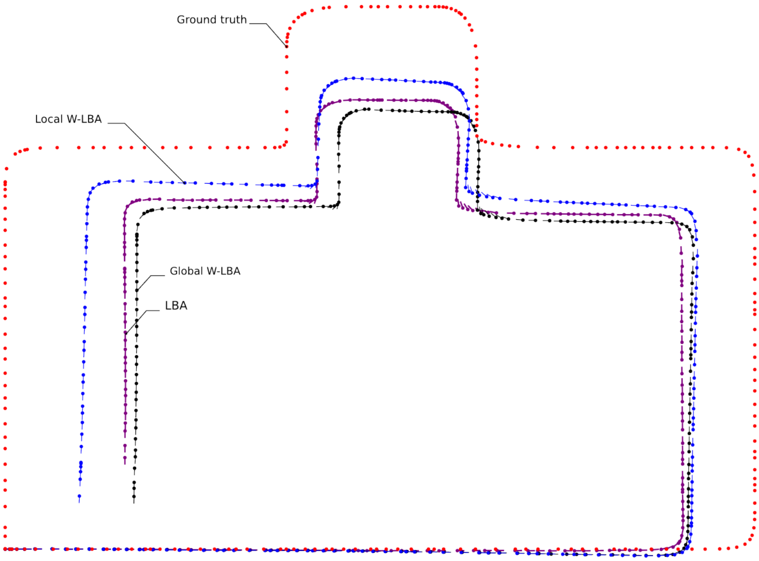

Figure 3: Top view of the trajectories reconstructed by LBA , global and local W-LBA and the ground truth of the synthetic sequence.

	Inter-camera ratio			Position error(m)			Angular error(deg)		
	mean	std	min	mean	std	max	mean	std	max
LBA	0.793	0.108	0.797	10.558	5.148	21.248	1.104	1.013	6.801
global W-LBA	0.799	0.099	0.656	11.324	5.777	20.332	1.868	1.276	9.687
local W-LBA	0.848	0.076	0.848	8.288	3.431	12.394	1.906	1.220	9.074

Table 1: Statistics on each reconstruction of figure 3 compared with ground truth

6.2 Odometry fusion

The odometry fusion has been tested in a real 4 km long sequence taken in urban area by a car in normal driving conditions (50 km/h maximum, in standard traffic). The camera provides 640 × 480 grey-scale images at 30 fps. The ground truth is given by a trajectometer(IXSEA landins) and we use the standard odometer of the car. Fig. 5 shows some images of the sequence. Fig. 4 provides a top view of the four trajectories (ground truth, LBA, LBA fusion,W-LBA fusion).

We observe that the scale of the trajectory is not constant all over the sequence and the scale factor drift is important at the end of the sequence. Thus fusion is a necessity to provide a correct trajectory reconstruction. Indeed, the two fusion methods, contrary to the standard monocular SLAM (LBA), achieved to reconstruct the 4 km sequences with less that 40m error. Statistics (in table 2) show that our method is more compliant with ground truth scale and has a lower error in position with a mean error of only 18 meters (about 50% error reduction compared to LBA fusion).

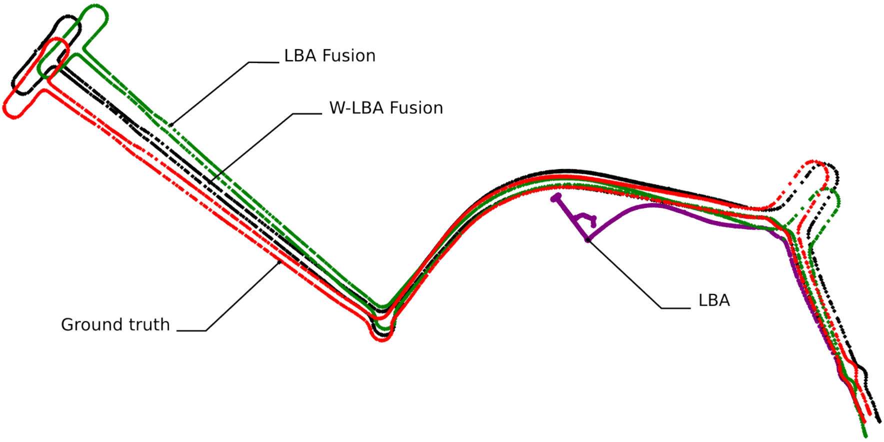

Figure 4: Top view of the reconstruction of the different methods of fusion with odometer (LBA fusion and W-LBA fusion) and the ground truth of the real sequence.

	Intercamera ratio			Position error(m)			Angular error(deg)		
	mean	std	max	mean	std	max	mean	std	max
LBA	0.177	0.256	2.071	348.980	271.22	894.79	15.382	8.243	161.47
LBA fusion	0.985	0.074	2.117	33.690	23.144	85.471	5.086	5.797	17.650
W-LBA fusion	1.016	0.151	5.350	18.012	11.286	43.235	3.397	12.018	17.938

Table 2: Statistics on each reconstruction of figure 4 compared with ground truth

7 Conclusion

In this paper, we highlight the use of Weighted Local Bundle Adjustment with local covariance to increase the accuracy of reconstruction and reduce the drift of scale factor. We demonstrate and prove with experiments, that W-LBA used with local covariance is more appropriate for data fusion. The fusion with odometer data allows to reduce the scale factor drift due to lack of constraint in the monocular visual SLAM.

Future works include the fusion between visual SLAM and GPS, loop closing using W-LBA and check global uncertainty propagation with fusion as in [5].

Figure 5: Three images of the real sequence.

References

[1] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *ECCV'06*.

[2] A. Cumani, S. Denasi, A. Guiducci, and G. Quaglia. Integrating monocular vision and odometry for SLAM. *WSEAS Transactions on Computers*, 3(3):625–630, 2004.

[3] A.J. Davison. Real-time simultaneous localisation and mapping with a single camera. In *ICCV'03*, 2003.

[4] A. Eudes, Maxime Lhuillier, Sylvie Naudet-Collette, and Michel Dhome. Fast odometry integration in local bundle adjustment-based visual slam. In *ICPR'10*.

[5] Alexandre Eudes and Maxime Lhuillier. Error propagations for local bundle adjustment. In *CVPR'09*.

[6] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communication of the ACM*, 24(6), 1981.

[7] L. Goncalves, E. Di Bernardo, D. Benson, M. Svedman, J. Ostrowski, N. Karlsson, and P. Pirjanian. A visual front-end for simultaneous localization and mapping. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 44–49, 2005.

[8] R.M. Haralick, C.N. Lee, K. Ottenberg, and M. Nolle. Review and analysis of solutions of the three point perspective pose estimation problem. *IJCV*, 1994.

[9] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1988.

[10] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. CUP, 2003.

[11] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and ME Munich. The vSLAM algorithm for robust localization and mapping. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 24–29, 2005.

[12] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd. Real time localization and 3d reconstruction. In *CVPR'06*.

[13] D. Nister, O. Naroditsky, and J. Bergen. Visual odometry. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1.

[14] H. Strasdat, JMM Montiel, and A.J. Davison. Real-Time Monocular SLAM: Why Filter? In *Int. Conf. on Robotics and Automation, Anckorage, AK*, 2010.

[15] B. Triggs, P.F. McLauchlan, R.I. Hartley, and A.W. Fitzgibbon. Bundle adjustment-a modern synthesis. In *Vision Algorithm: Theory and Practice*, 1999.