# Supplementary Material: Online Variational Bayesian Motion Averaging

Guillaume Bourmaud

Toshiba Research Europe
guillaume.bourmaud@crl.toshiba.co.uk

## 1 The case of a single loop using the absolute parametrization

### 1.1 Absolute parametrization and measurement models

The absolute parametrization consists in estimating transformations of the form $T_{kW}$ where $W$ is the "world" coordinate system and $k$ is the local coordinate system of the camera at time instant $k$. Thus at time instant $k$, the set of estimated transformations is $\{T_{iW}\}_{i=1,\dots,k}$. In practice, $T_{1W}$ is usually fixed to the identity matrix since the "world" coordinate system is also unknown. The likelihood for an odometry measurement and a loop closure are assumed to be Gaussian and are given in Table 1 eq.(1) and eq.(2), respectively.

**Table 1.** Odometry measurement model and loop closure measurement model using the absolute parametrization

| **Odometry likelihood** | **Loop closure likelihood** |
|---|---|
| $p\left(Z_{n(n+1)}\middle\vert T_{nW}, T_{(n+1)W}\right) = \mathcal{N}_G\left(Z_{n(n+1)}; T_{nW}T^{-1}_{(n+1)W}, \Sigma_{n(n+1)}\right)$ (1) | $p\left(Z_{mn}\middle\vert T_{mW}, T_{nW}\right) = \mathcal{N}_G\left(Z_{mn}; T_{mW}T^{-1}_{nW}, \Sigma_{mn}\right)$ (2) |

### 1.2 Inference using the absolute parametrization

Using the likelihoods defined in eq.(1) and eq.(2), we wish to minimize the following criterion w.r.t the absolute transformations $\{T_{iW}\}_{i=1,\dots,N_L}$:

$$-2\ln\left(p\left(\{T_{iW}\}_{i=1,\dots,N_L}\middle\vert Z_{1N_L}, \{Z_{i(i+1)}\}_{i=1,\dots,N_L-1}\right)\right) =$$
$$\left\|\log^{\vee}_G\left(Z_{1N_L}T_{N_LW}T^{-1}_{1W}\right)\right\|^2_{\Sigma_{1N_L}} + \sum_{i=1}^{N_L-1}\left\|\log^{\vee}_G\left(Z_{i(i+1)}T_{(i+1)W}T^{-1}_{iW}\right)\right\|^2_{\Sigma_{i(i+1)}} + \text{cst} \tag{3}$$

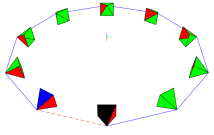
The classical way to minimize this criterion is to apply a Gauss-Newton algorithm where the absolute transformations are jointly refined iteratively as follows (the superscript stands for the iteration):

$$T^{(l)}_{iW} = \exp^{\wedge}_G\left(\delta^{(l/l-1)}_{iW}\right)T^{(l-1)}_{iW} \quad \text{for} \quad i = 1...N_L. \tag{4}$$

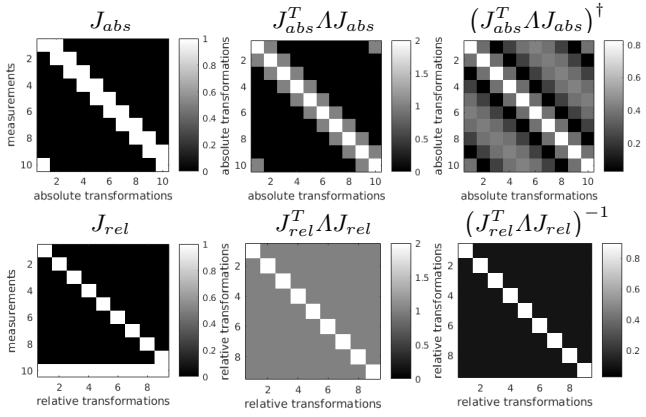
The increments $\left\{\delta_{iW}^{(l/l-1)}\right\}_{i=1,\dots,N_L}$ are obtained at each iteration by solving the following (sparse) linear system of size $p \times N_L$:

$$\begin{bmatrix} \delta_{1W}^{(l/l-1)} \\ \vdots \\ \delta_{N_L W}^{(l/l-1)} \end{bmatrix} = \left( \left(J_{abs}^{(l)}\right)^T \Lambda J_{abs}^{(l)} \right)^{\dagger} \left(J_{abs}^{(l)}\right)^T \Lambda \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \\ r_{1N_L}^{(l-1)} \end{bmatrix} \tag{5}$$

where $J_{abs}^{(l)}$ is the Jacobian matrix of the system, $\Lambda$ is a block diagonal matrix concatenating the inverse of the covariance matrices of the measurements and $\dagger$ stands for the pseudo-inverse of a matrix (since $J_{abs}^T \Lambda J_{abs}$ has $p$ null eigenvalues). Moreover $r_{1N_L}^{(l-1)} = \log_G^{\vee}\left(Z_{1N_L} T_{N_L W}^{(l-1)} \left(T_{1W}^{(l-1)}\right)^{-1}\right)$ and $r_{i(i+1)}^{(l-1)} = \log_G^{\vee}\left(Z_{i(i+1)} T_{(i+1)W}^{(l-1)} \left(T_{iW}^{(l-1)}\right)^{-1}\right)$.



(a) Illustration of a perfect loop of length 10, where a cone represents a camera pose (camera 1 is black camera 10 is blue). The (noiseless) odometry measurements are plotted as solid blue lines while the (noiseless) loop closure measurement is shown as a dashed red line.




(b) Jacobian, pseudo-Hessian and inverse pseudo-Hessian for absolute and relative parametrizations (only the magnitude of the coefficients is shown).

**Fig. 1.** Illustration of the motion averaging problem on $SE(3)$ for a single loop. Using an absolute parametrization, the inverse pseudo-Hessian exhibits very strong correlations between the absolute transformations. On the contrary, using a relative parametrization, the inverse pseudo-Hessian has very small correlations (not null but close to zero) between the relative transformations, motivating our variational Bayesian approximation of the posterior distribution which assumes independent relative transformations.

As can be seen in Fig. 1b, the pseudo-Hessian $J_{abs}^T \Lambda J_{abs}$ is extremely sparse, which allows to employ efficient sparse solvers. However, the pseudo-inverse

of the pseudo-Hessian $\left(J_{abs}^T \Lambda J_{abs}\right)^\dagger$, which represents (once the algorithm has reached convergence) the covariance matrix of the posterior distribution under a linear approximation, manifests very strong correlations between the transformations. This makes any approximation of that covariance matrix very difficult and thus prevents us from being able to derive a filter operating on large scale problems. In fact, this approach corresponds to what was proposed in [2] where the full covariance matrix is maintained. As it is acknowledged by the authors of [2], in practice their filter can only efficiently estimate less than a thousand absolute transformations.

## 2 Proof of eq.(7) of the submitted version of the paper

At iteration $l$, the Gauss-Newton algorithm consists in linearizing the terms inside the norms of the following cost function

$$\begin{aligned}&\left\|\log_G^\vee\left(Z_{1N_L}\left(\prod_{i=1}^{N_L-1}\exp_G^\wedge\left(\delta_{i(i+1)}^{(l/l-1)}\right)T_{i(i+1)}^{(l-1)}\right)^{-1}\right)\right\|^2_{\Sigma_{1N_L}}\\&+\sum_{i=1}^{N_L-1}\left\|\log_G^\vee\left(\exp_G^\wedge\left(\delta_{i(i+1)}^{(l/l-1)}\right)T_{i(i+1)}^{(l-1)}Z_{i(i+1)}^{-1}\right)\right\|^2_{\Sigma_{i(i+1)}}\end{aligned}\tag{6}$$

around $\delta_{i(i+1)}=\mathbf{0}$ for $i=1...N_L-1$. Thus we obtain:

$$\begin{aligned}&\left\|r_{1N_L}^{(l-1)}-\sum_{i=1}^{N_L-1}J_{1i}^{(l)}\delta_{i(i+1)}^{(l/l-1)}\right\|^2_{\Sigma_{1N_L}}+\sum_{i=1}^{N_L-1}\left\|r_{i(i+1)}^{(l-1)}+\delta_{i(i+1)}^{(l/l-1)}\right\|^2_{\Sigma_{i(i+1)}}\\&=\left\|r^{(l-1)}-J_{rel}^{(l)}\delta^{(l/l-1)}\right\|^2_{\Lambda^{-1}}\end{aligned}\tag{7}$$

where we approximated the Jacobian of $\log_G^\vee$ by the identity, $J_{11}^{(l)}\simeq Id$, $J_{1n}^{(l)}\simeq \mathrm{Ad}_G\left(\prod_{i=1}^{n-1}T_{i(i+1)}^{(l-1)}\right)$ for $n=2...N_L-1$, $r_{1N_L}^{(l-1)}=\log_G^\vee\left(Z_{1N_L}\left(\prod_{i=1}^{N_L-1}T_{i(i+1)}^{(l-1)}\right)^{-1}\right)$ and $r_{i(i+1)}^{(l-1)}=\log_G^\vee\left(T_{i(i+1)}^{(l-1)}Z_{i(i+1)}^{-1}\right)$.

Moreover, $r^{(l-1)}=\begin{bmatrix}r_{12}^{(l-1)}\\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)}\\ r_{1N_L}^{(l-1)}\end{bmatrix}$, $\delta^{(l/l-1)}=\begin{bmatrix}\delta_{12}^{(l/l-1)}\\ \vdots \\ \delta_{(N_L-1)N_L}^{(l/l-1)}\end{bmatrix}$, $J_{rel}^{(l)}=\begin{bmatrix}-Id & & \mathbf{0}\\ & \ddots & \\ \mathbf{0} & & -Id\\ J_{11}^{(l)} & \cdots & J_{1(N_L-1)}^{(l)}\end{bmatrix}$ and $\Lambda$ is a block diagonal matrix concatenating the inverse of the covariance matrices of the measurements.

Minimizing (7) w.r.t $\delta^{(l/l-1)}$ gives

$$\delta^{(l/l-1)} = \left(\left(J_{rel}^{(l)}\right)^T \Lambda J_{rel}^{(l)}\right)^{-1} \left(J_{rel}^{(l)}\right)^T \Lambda r^{(l-1)} \tag{8}$$

$$= \left(C^{-1} + \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} J_{LC}^{(l)}\right)^{-1} \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} r_{1N_L}^{(l-1)}$$

$$- \left(C^{-1} + \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} J_{LC}^{(l)}\right)^{-1} C^{-1} \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \end{bmatrix} \tag{9}$$

where $J_{LC}^{(l)} = \begin{bmatrix} J_{11}^{(l)} \cdots J_{1(N_L-1)}^{(l)} \end{bmatrix}$ and $C = \begin{bmatrix} \Sigma_{12} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \Sigma_{(N_L-1)N_L} \end{bmatrix}$. However, using the Woodbury formula:

$$(A+UBV)^{-1} = A^{-1} - A^{-1}U\left(B^{-1} + VA^{-1}U\right)^{-1} VA^{-1} \tag{10}$$

the second term in (9) becomes

$$- \left(C^{-1} + \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} J_{LC}^{(l)}\right)^{-1} C^{-1} \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \end{bmatrix}$$

$$= - \left(C - C\left(J_{LC}^{(l)}\right)^T \left(\Sigma_{1N_L} + J_{LC}^{(l)} C \left(J_{LC}^{(l)}\right)^T\right)^{-1} J_{LC}^{(l)} C\right) C^{-1} \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \end{bmatrix}$$

$$= - \left(Id - C\left(J_{LC}^{(l)}\right)^T \left(\Sigma_{1N_L} + J_{LC}^{(l)} C \left(J_{LC}^{(l)}\right)^T\right)^{-1} J_{LC}^{(l)}\right) \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \end{bmatrix} \tag{11}$$

Moreover, using the following identity

$$\left(C^{-1} + \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} J_{LC}^{(l)}\right)^{-1} \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1}$$

$$= C\left(J_{LC}^{(l)}\right)^T \left(J_{LC}^{(l)} C \left(J_{LC}^{(l)}\right)^T + \Sigma_{1N_L}\right)^{-1} \tag{12}$$

the first term in (9) becomes

$$\left(C^{-1} + \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} J_{LC}^{(l)}\right)^{-1} \left(J_{LC}^{(l)}\right)^T \Sigma_{1N_L}^{-1} r_{1N_L}^{(l-1)}$$

$$= C\left(J_{LC}^{(l)}\right)^T \left(J_{LC}^{(l)} C \left(J_{LC}^{(l)}\right)^T + \Sigma_{1N_L}\right)^{-1} r_{1N_L}^{(l-1)} \tag{13}$$

From (9), (11) and (13), we obtain

$$\delta^{(l/l-1)} = C\left(J_{LC}^{(l)}\right)^T \left(J_{LC}^{(l)} C \left(J_{LC}^{(l)}\right)^T + \Sigma_{1N_L}\right)^{-1} \cdot \cdot \left(\begin{bmatrix} r_{1N_L}^{(l-1)} \end{bmatrix} + J_{LC}^{(l)} \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \end{bmatrix}\right) - \begin{bmatrix} r_{12}^{(l-1)} \\ \vdots \\ r_{(N_L-1)N_L}^{(l-1)} \end{bmatrix} \tag{14}$$

In order to reduce the number of calls to $\log_G^\vee$, which can be time consuming (especially for $SL(3)$ since there exists no analytical formula for $\log_{SL(3)}^\vee$), we use the following approximation (assuming $T_{i(i+1)}^{(0)} = Z_{i(i+1)}$)

$$\begin{aligned} r_{i(i+1)}^{(l-1)} &= \log_G^\vee\left(T_{i(i+1)}^{(l-1)} Z_{i(i+1)}^{-1}\right) \\ &= \log_G^\vee\left(T_{i(i+1)}^{(l-1)} \left(T_{i(i+1)}^{(0)}\right)^{-1}\right) \\ &= \log_G^\vee\left(T_{i(i+1)}^{(l-1)} \left(T_{i(i+1)}^{(l-2)}\right)^{-1} T_{i(i+1)}^{(l-2)} \cdots \left(T_{i(i+1)}^{(1)}\right)^{-1} T_{i(i+1)}^{(1)} \left(T_{i(i+1)}^{(0)}\right)^{-1}\right) \\ &= \log_G^\vee\left(\exp_G^\wedge\left(\delta_{i(i+1)}^{(l-1/l-2)}\right) \exp_G^\wedge\left(\delta_{i(i+1)}^{(l-2/l-3)}\right) \cdots \exp_G^\wedge\left(\delta_{i(i+1)}^{(1/0)}\right)\right) \\ &\simeq \sum_{n=1}^{l-1} \delta_{i(i+1)}^{(n/n-1)} \end{aligned} \tag{15}$$

Finally, from (14) and (15) we obtain

$$\delta^{(l/l-1)} \simeq \begin{bmatrix} \Sigma_{12} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \Sigma_{(N_L-1)N_L} \end{bmatrix} \left(J_{LC}^{(l)}\right)^T \left(\Sigma_{1N_L} + \sum_{i=1}^{N_L-1} J_{1i}^{(l)} \Sigma_{i(i+1)} \left(J_{1i}^{(l)}\right)^T\right)^{-1} \cdot \cdot \left(\begin{bmatrix} r_{1N_L}^{(l-1)} \end{bmatrix} + J_{LC}^{(l)} \sum_{n=1}^{l-1} \begin{bmatrix} \delta_{12}^{(n/n-1)} \\ \vdots \\ \delta_{(N_L-1)N_L}^{(n/n-1)} \end{bmatrix}\right) - \sum_{n=1}^{l-1} \begin{bmatrix} \delta_{12}^{(n/n-1)} \\ \vdots \\ \delta_{(N_L-1)N_L}^{(n/n-1)} \end{bmatrix} \tag{16}$$

## 3 Online Variational Bayesian Motion Averaging algorithm

A pseudo-code of our online motion averaging algorithm is presented in Alg.1.

**Algorithm 1** Online variational Bayesian motion averaging algorithm

**Inputs:** $\left\{Z_{(k-1)k}, \Sigma_{(k-1)k}\right\}_{k=2,\dots}$ (stream of odometry measurements), $\{Z_{mn}, \Sigma_{mn}\}_{m<n}$ (stream of loop closure measurements), $t$ (gating threshold)

**Outputs:** The algorithm continuously provides an estimate of the absolute transformations (w.r.t the first reference frame) $\left\{\overline{T}_{1k}^{abs}\right\}_{k=1,\dots}$ by composing the estimated relative transformations

$k=1$, $\overline{T}_{1k}^{abs} = Id$
**while** 1 *//infinite loop processing new odometry and loop closure measurements*
 $k = k+1$

 *//process new odometry measurement* $\left(Z_{(k-1)k}, \Sigma_{(k-1)k}\right)$
 $\overline{T}_{(k-1)k} = Z_{(k-1)k}$, $P_{(k-1)k} = \Sigma_{(k-1)k}$ *//initialize mean and covariance of* $T_{(k-1)k}$
 $\overline{T}_{1k}^{abs} = \overline{T}_{1k-1}^{abs}\overline{T}_{(k-1)k}$ *//initialize absolute transformation* $\overline{T}_{1k}^{abs}$

 *//process new loop closure measurements* $\{Z_{lk}, \Sigma_{lk}\}_{l<k}$
 **for** each new loop closure measurement $(Z_{lk}, \Sigma_{lk})$ where $l < k$ **do**
  **if** $\left\|\log_G^{\vee}\left(Z_{lk}\left(\prod_{i=l}^{k-1}\overline{T}_{i(i+1)}\right)^{-1}\right)\right\|^2_{\Sigma_{lk}+\sum_{i=l}^{k-1} J_{li}P_{i(i+1)}J_{li}^T} < t$ **then** *//gating*

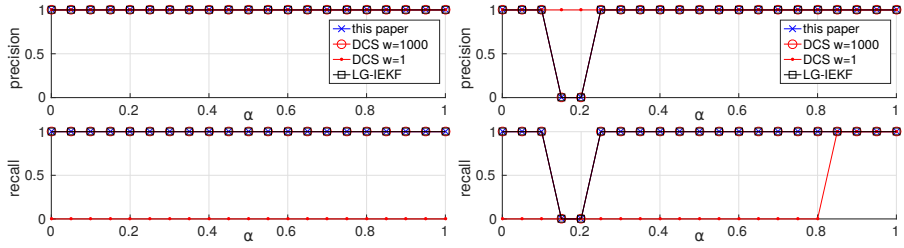
   *//apply Gauss-Newton*
   **for** $i = l, ..., k-1$ **do** $\breve{T}_{i(i+1)} = \overline{T}_{i(i+1)}$, $r_{i(i+1)} = \mathbf{0}$, $\delta_{i(i+1)} = \mathbf{0}$ **end for**
   **while** not converged
    $r_{lk} = \log_G^{\vee}\left(Z_{lk}\left(\prod_{i=l}^{k-1}\breve{T}_{i(i+1)}\right)^{-1}\right)$ *//compute loop error*
    $r_{l(l+1)}$ += $\delta_{l(l+1)}$, $J_{ll} = Id$ *//update relative errors and compute Jacobians*
    **for** $i = l+1, ..., k-1$ **do** $r_{i(i+1)}$ += $\delta_{i(i+1)}$, $J_{li} = \mathrm{Ad}_G\left(\prod_{j=l}^{i-1}\breve{T}_{j(j+1)}\right)$ **end for**
    $r_{lk,cum} = r_{lk}$, $C_{lk,cum} = \Sigma_{lk}$ *//compute cumulated error and covariance*
    **for** $i = l, ..., k-1$ **do**
     $r_{lk,cum}$ += $J_{li}r_{i(i+1)}$, $C_{lk,cum}$ += $J_{li}P_{i(i+1)}J_{li}^T$
    **end for**

    **solve** $C_{lk,cum}x = r_{lk,cum}$

    *//compute increments and update relative transformations*
    **for** $i = l, ..., k-1$ **do**
     $\delta_{i(i+1)} = P_{i(i+1)}J_{li}^T x - r_{i(i+1)}$, $\breve{T}_{i(i+1)} = \exp_G^{\wedge}\left(\delta_{i(i+1)}\right)\breve{T}_{i(i+1)}$
    **end for**
   **end while**

   *//variational Bayesian approximation of the posterior*
   **for** $i = l, ..., k-1$ **do** $\overline{T}_{i(i+1)} = \breve{T}_{i(i+1)}$, $P_{i(i+1)} = \left(J_{li}^T\Sigma_{lk}^{-1}J_{li} + P_{i(i+1)}^{-1}\right)^{-1}$ **end for**

   *//update absolute transformations modified by the loop closure*
   **for** $i = l, ..., k-1$ **do** $\overline{T}_{1(i+1)}^{abs} = \overline{T}_{1i}^{abs}\overline{T}_{i(i+1)}$ **end for**
  **end if**
 **end for**
**end while**




(a) KITTI 00 (b) KITTI 07

**Fig. 2.** Results for planar visual SLAM ($SE(2)$) with wrong loop closures (the results of DCS are taken from Fig.4 in [4]): Using its default parameter value ($w = 1$), DCS [1] rejects all (correct and wrong) the loop closures. Using a higher value ($w = 1000$), DCS manages to reject wrong loop closures. Contrary to DCS, both our approach and LG-IEKF [2] do not need parameter tuning (we simply use the $\mathcal{X}^2$ value with $p$ degrees of freedom given by a p-value of 0.001) and produces the same results as DCS when its parameter is optimized. *Remark: results for COP-SLAM* [3] *could not be provided since it is not able to detect wrong loop closures.*

## 4 Evaluation of the robustness

In this experiment, we employ the dataset provided by the authors of [4] which allows to evaluate the robustness of an approach to wrong loop closures on a planar visual SLAM application (Lie group $SE(2)$). For two sequences of the KITTI dataset, both a visual odometry module and a loop closure module have been employed in order to obtain the odometry measurements and loop closures. The loop closure module that has been used has a threshold $\alpha$ that allows to tune the algorithm output ($\alpha = 1$: only highly confident loop closures, $\alpha = 0$: all the loop closures). Synthetic wrong loop closures are also generated and added to the set of loop closure measurements. In order to evaluate the robustness of an algorithm, the experiment consists, for different values of $\alpha$, in applying the algorithm and computing the loop closure detection precision and recall. The results for this experiment are given in Fig.2 where we reported the precision and recall for our approach, LG-IEKF [2] and DCS [1] (that uses g$^2$o).

One can see that our approach surprisingly achieved exactly the same precision and recall as both LG-IEKF and DCS. This is a remarkable result since these two algorithms are not designed to perform online large scale estimation and are consequently much slower than our approach.

Let us note that for $\alpha = 0.15$ and $\alpha = 0.2$ we observe a failure case for all the algorithms. Indeed, both precision and recall drop to zero because an ambiguous wrong loop closure was classified as inlier. This led to the distortion of the estimated trajectory and, as a consequence, to the rejection of all the subsequent loop closures.

# 5 Additional experiments

## 5.1 Monocular Visual SLAM

In this experiment, we demonstrate the ability of our approach to deal with the Lie group $Sim(3)$ by considering a monocular visual SLAM problem. In order to obtain the measurements, a local SLAM approach performs sliding-window bundle adjustment in order to reconstruct small overlapping submaps. Here a "submap" corresponds to a fragment of the camera trajectory and a small 3D point cloud. The odometry measurements are obtained by computing the 3D similarities between the temporally consecutive submaps. In order to detect a loop closure between the current submap and a previous submap, we first use a place recognition algorithm. Then, for each candidate past submap, we match the descriptors of its 3D points with the 3D points of the current submap and then compute a 3D similarity using a RANSAC algorithm.

The objective of a motion averaging algorithm in this application is to align all the submaps, resulting in a global map of the environment, i.e a global 3D point cloud as well as the complete camera trajectory.

In Fig. 3, we present results for monocular visual SLAM (Lie group $Sim(3)$) on sequence 15 of the KITTI dataset. One can see that the trajectory estimated with our approach is visually much closer to the result of [5] (which employs a Lidar) than the trajectory estimated with COP-SLAM. Results on sequence 13 of the KITTI dataset are provided in the submitted version of paper.

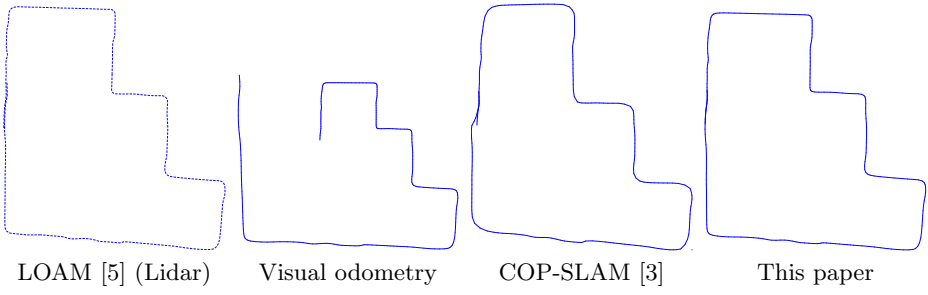



**Fig. 3.** Results for monocular visual SLAM (Lie group $Sim(3)$) on sequence KITTI 15. The ground truth is not available for that sequence. Thus, we reported the best result obtained using a Lidar [5].

## 5.2 Video mosaicking

In this experiment, we demonstrate the ability of our approach to deal with the Lie group $SL(3)$ by considering a video mosaicking problem. The odometry measurements are obtained by computing the homographies between temporally consecutive frames of the video. In order to detect the loop closures (i.e when

the current video frame overlaps with a past video frame), we employ an image recognition algorithm and then apply a RANSAC algorithm to compute the homography between the current video frame and the retrieved past video frame.

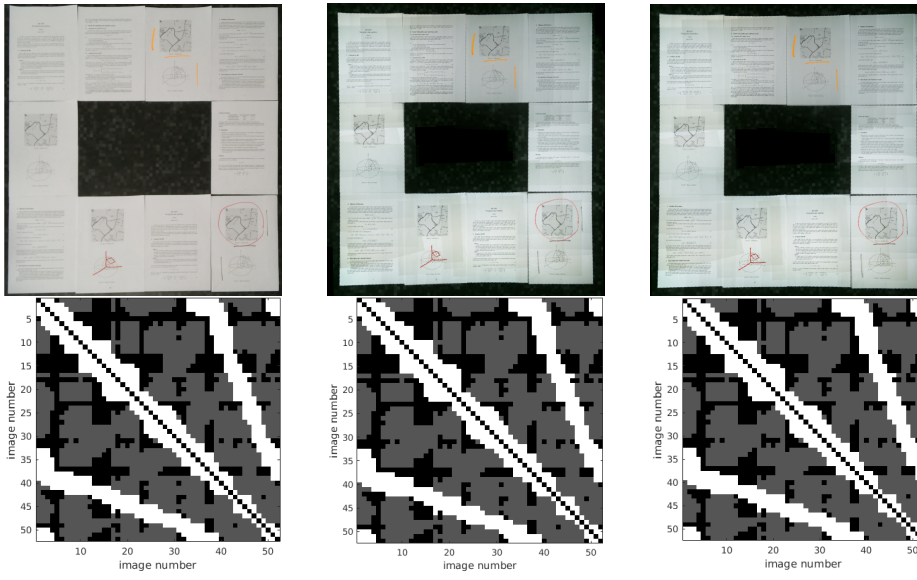
The objective of a motion averaging algorithm in this application is to align all the video frames, assuming a planar scene, resulting in a global (planar) mosaic of the entire scene.

In Fig. 4, we present results for video mosaicking (Lie group $SL(3)$) on the dataset provided by the authors of [2].

In theory, LG-IEKF [2] should outperform our approach since our approach approximates the posterior distribution with few parameters in order to efficiently deal with large scale problems.

However, one can see that, like LG-IEKF, our approach produces a mosaic visually very close to the ground truth and also perfectly detects the wrong loop closures.

(a) Examples of input images




Ground truth LG-IEKF [2] Ours

(b) Top row: mosaic, Bottom row: labeling matrices (a white pixel is an inlier, a black pixel corresponds to an unavailable measurement, a gray pixel corresponds to an outlier)

**Fig. 4.** Results for video mosaicking (Lie group $SL(3)$).

## References

1. Agarwal, P., Tipaldi, G.D., Spinello, L., Stachniss, C., Burgard, W.: Robust map optimization using dynamic covariance scaling. In: Robotics and Automation (ICRA), 2013 IEEE International Conference on. pp. 62–69. IEEE (2013)
2. Bourmaud, G., Mégret, R., Giremus, A., Berthoumieu, Y.: Global motion estimation from relative measurements in the presence of outliers. In: Computer Vision–ACCV 2014, pp. 366–381. Springer (2014)
3. Dubbelman, G., Browning, B.: Cop-slam: Closed-form online pose-chain optimization for visual slam. Robotics, IEEE Transactions on 31(5), 1194–1213 (2015)
4. Latif, Y., Cadena, C., Neira, J.: Robust graph slam back-ends: A comparative analysis. In: Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on. pp. 2683–2690. IEEE (2014)
5. Zhang, J., Singh, S.: LOAM: Lidar odometry and mapping in real-time. In: Robotics: Science and Systems Conference (RSS). Berkeley, CA (July 2014)