# Robust and Accurate Deterministic Visual Odometry

Pierre Bénet, *SBG Systems, Carrière sur Seine, France*
Alexis Guinamard, *SBG Systems, Carrière sur Seine, France*

**ABSTRACT**

Finding alternative technologies for GNSS-denied environments is a key to extend the capability and robustness of autonomous vehicle and mapping application. A solution to the problem is vision-simultaneous-localization and mapping. Since cameras are light weights, robust and passive sensors, they are leading candidates for GNSS-denied environment technology. Accuracy and robustness are the two main concerns regarding these technologies. While high accuracy is achieved thanks to loop-closing (correct position when crossing places that where already visited) [1], robustness is achieved thanks to an accurate short-term visual odometry. Hence SOFT-SLAM [2] the currently top-ranking stereo vision methods on KITTI benchmark [3] focused on pure visual odometry [4] before to deal with Simultaneous Localization And Mapping.

In this paper we present a novel algorithm for fast and robust stereo odometry based on hybrid stereo and monocular algorithm. First, interest points in the images are selected using circular matching of features between left and right, current and next images, using a sparse feature descriptor described in Stereoscan [5]. Then rotation and translation between two consecutive poses are estimated separately. A mean square is used for translation estimation whereas a parametrization of epipolar constraint similar to [6] is used for rotation estimation. Experimental results show that the proposed algorithm achieves state of the art translation error on KITTI benchmark using the KITTI evaluation metric [3]. According to this metric, it has already lower mean translational and rotational error than state of the art SLAM algorithm such as ORB-SLAM2[1] while our algorithm is a pure visual odometry algorithm. We also tested our algorithm in inertial aided situation using the EuRoC MAV dataset[7] where we also achieved competitive results. Our algorithm processes a frame in 0.07s on average on a single core at 3Ghz. This allows real time odometry outputs.

**OUTLINE**

First, we will present the feature detection and matching algorithms, next we will set the equations of stereo odometry and monocular odometry. Then from optimal state estimation theory, using a general formulation of Extended Kalman Filter [8], we show how an optimal non-linear least square is obtained, for both problems of epipolar constraint (monocular motion estimation) and stereo reprojection constraint (for stereo estimation). This optimal formulation gives us at the same time a metric to filter outlier using the chi-square test and a suitable covariance estimation to couple the system with other sensors such as inertial data or GNSS data in difficult GNSS situations[9]. We also present or visual inertial coupling strategy and further visual odometry improvements.

Regarding integrity and robustness, the total uncertainty of our odometry system is propagated during the vehicle travel thanks to an Extended Kalman filter. Hence the predicted accuracy is well estimated. Confidence ellipses are obtained during the whole path and correspond to the observed error relative to the ground truth. Our main contributions are:

- An overall deterministic algorithm that do not uses randomized process to converge: Outlier rejection is based on a chi2 test using prior estimate and a constant 30% outlier removal using chi2 test on posterior estimate.
- A New monocular algorithm based on an epipolar constraint parametrization and accurate statistical formulation, leading to a precise, robust and fast monocular algorithm that boosts overall stereo visual odometry accuracy.

**RELATED WORK**

During the past year, several algorithms were developed to solve the problem of visual Odometry or visual Simultaneous Localization and Mapping.

- Frame to Frame odometry

A simple frame to frame algorithm is enough to give good results in Stereo Odometry. Here only the delta pose (a rotation and a translation) between two frame pair is computed in a sequential manner. The least squares with Random sampling Consensus [10] (RANSAC) is usually used to solve this problem. Several minimal sets of features are tested, and the best set is chosen. For stereo

odometry, 3 pair of points are sufficient. Several improvements can be found in the literature such as feature tracking for outlier removal [4]. An important improvement is to use a separate estimator for rotation based on monocular odometry[11][12]. Monocular frame to frame algorithms only gives the rotation and the direction of translation but not the scale. In practice these algorithms are very precise for rotation estimation and can be used to improve the angular estimation in stereo visual odometry system. Monocular frame to frame algorithms serve also for initialization in more complex SLAM system[1]. Most of these frame to frame algorithms use RANSAC to find inliers. There are the 8 points algorithm[13] for monocular which relies on singular value decomposition; and the 5 points algorithm [14] which relies on solving a $10^{th}$ degree polynomial. However we will show that a parametrization of epipolar constraint based on [6] yields a state-of-the-art precision at a minimal computation cost without relying on randomized process.

- Sliding window filter

Another class of algorithm are Sliding window filter [15]. These were initially developed for monocular odometry since frame to frame monocular algorithms cannot track the scale. Here the position of the features (~2000 for DSO[16]) enters the minimization system along with a few frames (~10) which gives a big optimization system. Hopefully, since the system remains sparse it can be easily solved using block inversion. Several parametrizations are available for feature position: either a 3d position in space[1], or only the inverse depth of the feature in a reference image[17]. Again, since monocular algorithm gives often better angular estimate than stereo algorithm, their extension to stereo odometry yield good results [18][1].

- Graph SLAM

Graph Slam [1] system is a way of solving the full system without neglecting correlation between variables. This system is used for relocalization and loop closing. It corrects the estimate when the camera sees a place that has already been visited. These systems yield best precision in closed environments where the camera revisits a lot of place. Here, since more correlation between states are kept than for windowed SLAM, the system is less sparse, but still sparse. The Preconditioned Conjugate gradient method is used to solve this problem [19] which is a little slower than sliding window systems.

- Inertial coupling

In more recent years, more research on Visual Inertial coupling have been done[20][21][22][23]. Indeed, the inertial coupling extension of the latter three SLAM methods have been done. Inertial coupling often gives more robustness and a better precision while being lightweight. Moreover, this extension is mandatory for monocular slam system since scale cannot be observed (except learning depth from previous data [24]). There are also several visual odometry dataset available such as the EUROC-MAV dataset [7]

## FEATURE MANAGEMENT

Our visual odometry algorithm relies on detecting and associating interests points in the image such as corners and blobs. This is a well-known starting point for a lot of algorithms. First one detects high gradient points in the images, then tries to associate them on the other images using a similarity or dissimilarity measure between these points.

### Feature detection

To detect these points, one can use corner and blob convolution[5] or Harris Corner detector. The two methods are based on gradient image. Whereas convolution tries to find a matching pattern, Harris Corner detector will detect if a feature is not a pure line and can be safely located in two dimensions. Harris Corner simply computes the image gradient covariance around the interest pixel, and if the determinant of the covariance is big enough, the feature will be safely recognizable. These detectors give a quality value for each pixel of the image. Then one can divide the image in small blocks and take only the best pixel in each blocks. This process is commonly referred as non-maximum suppression. Then, having one potential feature in each blocks of one image and another, one will try to match them between the image using a similarity or dissimilarity measure.

### Dissimilarity measure

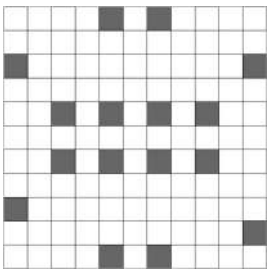

*Figure 1: position of the pixels used for the descriptor. There are 16 pixels distributed in a sparse manner in a 11x11 window around the feature center. The descriptor is the concatenation of the intensity of the smoothed x gradient and y gradient (ie. sobel transform) at these pixel positions.*

We use the same feature descriptor as Stereoscan [5]: To measure similarity between pixels for matching, a descriptor vector is defined. The descriptor vector is composed of 32 pixels of the spatial derivative of the image situated in a sparse manner around the interest point. Then to measure the distance between two descriptors, we compute their normalized Sum of squared differences (NSSD). We keep the best match among the feature tested.

$$NSSD(d_1, d_2) = \frac{\|d_1 - d_2\|^2}{\|d_1\| \|d_2\|} \quad (1)$$

**Circular matching**

By searching the best match in a circular manner between the left and right and the current and next frame, reliable stereo features are detected. From the left and right image feature association, one can compute a first feature depth, then one can reproject the feature position using the motion prior to have a more precise search zone for matching between different times. This search zone depends on the motion prior state and covariance. The same process is done backward. If the last pixel found is the same that the first, the circle is closed, and the circular matching succeeded.


*Figure 2: Circular matching.*

**KALMAN FILTER AND LEAST SQUARE**

**Stereo motion estimation**

Here we focus on the update part of the Kalman filter. For this part, the only model we need is the observation function. For the stereo problem, we write an observation as the observation of the feature position from one stereo couple to another stereo couple. We define the observation function $h$ by the successive operation of triangulation, transformation, and projection:

$$h = \begin{pmatrix} f\,(x' + b/2)/z' \\ f\,y'/z' \\ f\,(x' - b/2)/z' \\ f\,y'/z' \end{pmatrix} \leftarrow \begin{matrix} x' \\ y' \\ z' \end{matrix} \text{ o } (\mathrm{R\,X + t} \leftarrow \mathrm{X}) \text{ o } \begin{pmatrix} (u_{x,l}/2 + u_{x,r}/2)b/(u_{x,r} - u_{x,l}) \\ (u_{y,l}/2 + u_{y,r}/2)b/(u_{x,r} - u_{x,l}) \\ b\,f/(u_{x,r} - u_{x,l}) \end{pmatrix} \leftarrow \begin{matrix} u_{x,l} \\ u_{y,l} \\ u_{x,r} \\ u_{y,r} \end{matrix} \quad (2)$$

$b$ is the baseline, ie. the distance between the left and right cameras in meter. $f$ is the focal length of the cameras in pixels. $u$ is composed of the left and right pixel coordinates relative to the images centers. Exponential map is used to parameterize the rotation

R. We can rewrite this observation function from (2) in a more concise way using $x$ as the state variable that includes the rotation parameters and the translation t. We also show the noise that are added on the pixel coordinates.

$$\mathrm{y} = (y_l + \gamma_1, y_r + \gamma_2) = \mathrm{h}(u_l + \gamma_3, u_r + \gamma_4, x) = \mathrm{h}(u, x) \tag{3}$$

The observation dimension is 4 and the state variable dimension is 6. We see that we have both additive noise $\gamma_1$ and $\gamma_2$ and non-additive noise $\gamma_3$ and $\gamma_4$. This is quite rare in Kalman filtering. However, this is not very complicated. Since Kalman equations are available for both additive and non-additive noise and the noises are independent, one can easily find the solution which is summing the contribution of the additive and non-additive noise. To this end, the gradients of the observation function needs to be computed:

$$H = \frac{\partial h(u,x)}{\partial x} \quad \text{and} \quad M = \frac{\partial h(u,x)}{\partial u} \tag{4}$$

and we assume that the noise on the pixel coordinates are independent and of same intensity, $\Gamma = \alpha\, I_{4x4}$ then the Kalman update equations are:

$$\begin{gathered} \mathrm{K} = \mathrm{P}^{-}\mathrm{H}^{\mathrm{T}}(\mathrm{H}.\mathrm{P}^{-}.\mathrm{H}^{\mathrm{T}} + \mathrm{M}.\Gamma.\mathrm{M}^{\mathrm{T}} + \Gamma)^{-1} \\ \mathrm{x}^{+} = \mathrm{x}^{-} + \mathrm{K}\big(\mathrm{y} - \mathrm{h}(\mathrm{u}, \mathrm{x})\big) \\ P^{+} = (\mathrm{I} - \mathrm{KH})\mathrm{P}^{-} \end{gathered} \tag{5}$$

After applying these equations, one can re-linearize with a better estimate and recompute the update. This is an Iterated Extended Kalman filter. However, to have a little speed-up, we convert the equations (5) to an information filter [25]:

$$\begin{gathered} P^{+} = \left(\Sigma(H_i^T W_i H_i)\right)^{-1} \\ \mathrm{x}^{+} = \mathrm{P}^{+}\left(\Sigma\left(H_i^T W_i (y_i - h_i)\right)\right) \\ \text{with } W_i = (M_i.\Gamma.\mathrm{M}_i^T + \Gamma)^{-1} \end{gathered} \tag{6}$$

Using this composite noise covariance for $W_i$ instead of an identity noise covariance showed significant improvements.

**Monocular motion estimation**

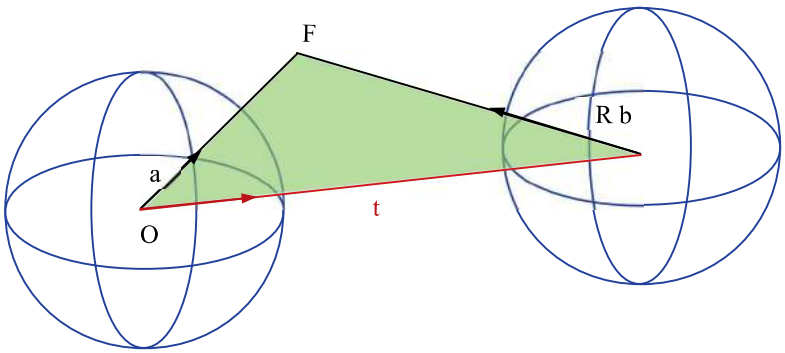


*Figure 3: Epipolar constraint.*

Here we will use a parametrization of the epipolar constraint. The epipolar constraint says that when a point is observed from two different cameras, the three vectors made of the directions of the features in the two cameras and the translation between the cameras must lie on the same plane. So let $a^T = (u_{x,1}, u_{y,1}, f)$ and $b^T = (u_{x,2}, u_{y,2}, f)$ be the feature directions in each camera frame.
Then the constraint can be written with $\times$ the cross product operator, or [] the transformation matrix of the cross product operation:

$$a.(t \times (R\, b)) = a^T\, [t] R\, b = 0 \tag{7}$$

Since the scale cannot be observed, another parametrization of (7) with less freedom is used:

$$h(u_1 + \gamma_1, u_2 + \gamma_2, x) = a^T R_1^T [e_z] R_2 b = h(u, x) = 0 \quad (8)$$

$R_1$ and $R_2$ are parametrized using temporary matrix rotation and exponential map and $R_2$ is only parametrized around $x$ and $y$ [6]. This way the system is of five degrees. This corresponds to a six degrees of freedom pose without the scale. At each Gauss Newton iteration, the exponential maps are cumulated by left multiplication on the temporary rotation matrix and the exponential maps are reset to 0. This time the observation function is scalar but has the same number of inputs as previously. There are only non-additive noises, so the following weight must be used in equation (6):

$$W_i = (M_i . \Gamma . \mathrm{M}_i^T)^{-1} \quad (9)$$

While in stereo method approximating the noise covariance by the Identity only degrade results, for epipolar constraint, this approximation makes the system fail. In [6], this problem was treated by modifying the parametrization to weight more equally the observations. However these formulations remain suboptimal since the noise covariance was still neglected. Different parametrization tests showed that a simple parametrization with an optimal formulation remains the best solution.

**OUTLIER REMOVAL**

A big part of odometry is to find the good features and reject the outliers. As explained previously, a lot of frame to frame algorithms use randomized process to find inliers. The other solution is to use loss functions[26].
First, at the matching stage, computing a prediction zone based on the a priori covariance to restrict the matching area reduces the outlier number. Then, several loss functions were tested such as the Huber loss, the Cauchy loss, or a constant outlier ratio. For the stereo stage, a constant 30% ratio of outlier removal showed better results. For the epipolar constraint, the Cauchy loss showed a highest precision. This loss is computed relatively to the current median chi square test. However using only the features classified inliers from the stereo algorithm yields even better results for the epipolar constraint. This is an interesting result, that shows that even if a frame to frame stereo algorithm can be less precise on the rotation than a monocular algorithm, it can be better for outlier removal.

*Figure 4: Outlier removal, blue: 70% inliers, red: 30% outliers.*

**INERTIAL COUPLING**

We also implemented coupling with inertial measurement from an accelerometer and a gyrometer. Thanks to the Euroc[24] dataset we successfully tested the benefits of imu coupling. Our coupling strategy is close to [4] where we do not take the full state of the Kalman filter as output but only the rotation. We believe that with higher grade imu, the translation output from the Kalman filter will be more precise than the direct sum of the odometry translations estimation. Here we use a full Kalman filter with gyro bias and accelerometers bias that can be extended to any other sensors such as GPS. At start the full biases from the gyroscope and the accelerometer are estimated without help, as well as the initial roll and pitch without performance loss as long as the initial roll and pitch are under 10 degrees, which is the case on all Euroc dataset. The Kalman filter also gives accurate uncertainty estimation (Fig. 6 to 9), even when no inertial momentum unit is available. In such case the IMU is replaced with a general motion model. The same motion model for drone and automotive scenario is used.

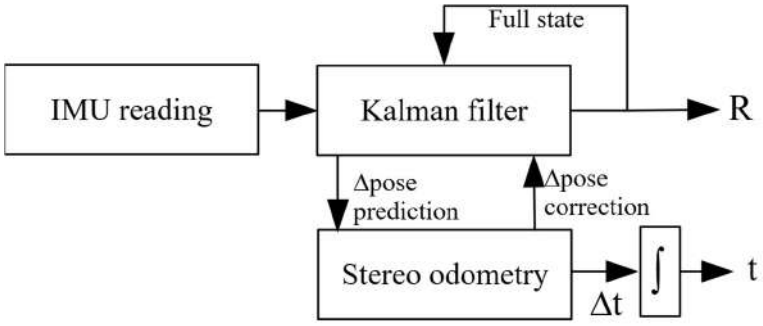


*Figure 5: IMU and Stereo odometry coupling.*

## FURTHER IMPROVEMENTS

### Rotation merging

A significant improvement can be done using the method described in [4], That relies on computing the rotation evolution on three frames and more instead of two frames, this gives another value for the current delta rotation, and the different values for the delta rotation are merged using spherical interpolation:

$$R'^{k}_{k-1} = (R^{k}_{k-1})^{1/2}(R^{k}_{k-2}(R^{k-1}_{k-2})^{T})^{1/2} \tag{10}$$

Computing the rotation over three frames requires to redo the whole process of feature matching and outlier removal, increasing the computation time by two, for an improvement of about 10%. SOFT-SLAM[2] pushed the principle using one more frame but we found that it gives an insignificant improvement in our case.

### Keyframing

Keyframing is a well-known component in visual SLAM that consists in selecting only the frames that are not redundant. In particular in monocular odometry or for relocalization this is a critical point. For frame to frame odometry, interest is limited, and the process of keyframe is simply to wait for a significant displacement to change the reference frame. For automotive scenario such as in KITTI, waiting for a significant displacement gives no real improvement. However for the drone dataset, this becomes more interesting, since the drone has long periods of rest and its displacement can be very slow sometimes. In this case, keyframing cuts the error about half.

## RESULTS

We are currently at the second place on the KITTI stereo odometry benchmark. The first place is occupied by SOFT-SLAM which is a Simultaneous localization and mapping system. We currently have the best algorithm among the stereo visual odometry without loop closing on the KITTI benchmark. We also obtain our results with the smallest computing power.

*Table 2: Extract of the current KITTI Stereo odometry leaderboard.*

| Stereo rank | Method | Translation | Rotation (deg/m) | Runtime | Environment |
|---|---|---|---|---|---|
| **1** | SOFT-SLAM | 0.65% | 0.0014 | 0.1s | 2 core @ 2.5Ghz |
| **2** | **RADVO** | **0.82%** | **0.0018** | **0.07s** | **1 core @ 3.0Ghz** |
| **3** | LGSLAM | 0.82% | 0.0020 | 0.2s | 4 core @ 2.5Ghz |
| **4** | ROTROCC+ | 0.83% | 0.0026 | 0.25s | 2 core @ 2.0Ghz |
| **5** | GDVO | 0.86% | 0.0031 | 0.09s | 1 core @ >3.5Ghz |


*Figure 6: Results on KITTI dataset 00 (orange: ours , blue: ground truth, green : ORB-SLAM2, , black: uncertainty ellipses)*

We compare the output of our algorithm on the KITTI test dataset and the EuRoC dataset. Typical metric used for comparing performance are:

- the relative metric which is a measure of percent of translation drift used in KITTI benchmark. This metric requires the ground truth to have precise rotations. The procedure for computing such a metric is freely available on the KITTI website.
- The absolute metric, which is the global translation error after rotation and translation fitting between the ground truth and the odometry result. We used a package called evo for evaluating this metric which is freely available.

We compare in Table 1 our result to state-of-the-art methods. The results from the different algorithms were retrieved from their respective paper. We checked anyway the evaluation metrics on ORBSLAM2 and obtained the expected results. We compare our visual odometry algorithm with two SLAM methods (ORB-SLAM2 and SOFT-SLAM) and a direct visual odometry method, GDVO[27].

*Table 1: comparison of RADVO with SOFT-SLAM, ORB-SLAM2 and GDVO on the KITTI dataset using the relative translational metric in percent and the absolute metric in meters*. lc refers to dataset with loop closing available

| Relative and absolute error | RADVO | RADVO | SOFT-SLAM | SOFT-SLAM | ORB-SLAM2 | ORB-SLAM2 | GDVO | GDVO |
|---|---|---|---|---|---|---|---|---|
| KIT.00 lc | 0.52% | 2.1m | 0.66% | 1.2m | 0.70% | 1.3m | 0.71% | 4.9m |
| KIT.01 | 0.74% | 3.8m | 0.96% | 3.0m | **1.39%** | **10.4m** | **1.00%** | 5.2m |
| KIT.02 lc | 0.59% | **4.3m** | **1.36%** | **5.1m** | 0.76% | 5.7m | 0.70% | **6.1m** |
| KIT.03 | **0.92%** | 1.2m | 0.70% | 0.5m | 0.71% | 0.6m | 0.75% | 0.3m |
| KIT.04 | 0.44% | 0.3m | 0.50% | 0.4m | 0.48% | 0.2m | 0.42% | 0.2m |
| KIT.05 lc | 0.50% | 1.4m | 0.43% | 0.8m | 0.40% | 0.8m | 0.47% | 1.8m |
| KIT.06 lc | 0.56% | 1.4m | 0.41% | 0.5m | 0.51% | 0.8m | 0.41% | 1.5m |
| KIT.07 lc | 0.50% | 0.8m | 0.36% | 0.3m | 0.50% | 0.5m | 0.40% | 0.8m |
| KIT.08 | 0.88% | 2.7m | 0.78% | 2.3m | 1.05% | 3.6m | 0.88% | 2.4m |
| KIT.09 lc | 0.82% | 2.5m | 0.59% | 1.3m | 0.87% | 3.2m | 0.77% | 2.2m |
| KIT.10 | 0.85% | 1.0m | 0.68% | 0.9m | 0.60% | 1.0m | 0.63% | 1.1m |

In Table 1 we have the minimum worst-case error in both relative and absolute error metric. We can see that SOFT-SLAM and ORB-SLAM2 have a lower absolute error on datasets with loop closing since they correct position on loop closing, but the ability to close loop does not influence a lot the relative metric.

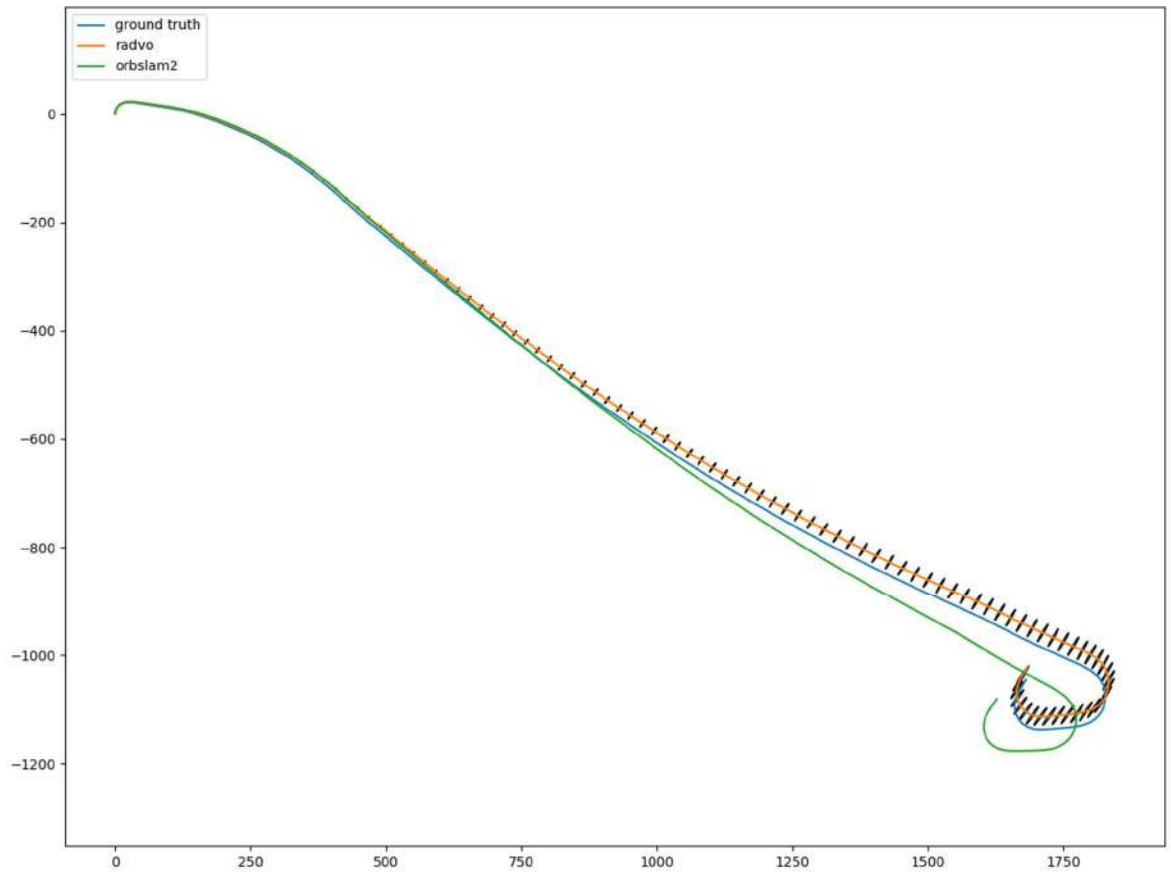


*Figure 7: Results on KITTI dataset 01 (orange : ours , blue :ground truth, green : ORB-SLAM2 , black : uncertainty ellipses)*

*Table 3: Results of RADVO compared with SOFT-SLAM, ORB-SLAM2, VIDSO and VI ORB-SLAM on the EuRoC dataset using the absolute metric in centimeters. Datasets marked with * indicate a needed stereo calibration correction remarked by SOFT-SLAM.*

| Absolute error (cm) | RADVO | SOFT-VO | SOFT-SLAM | ORB-SLAM2 | VIDSO | VI ORB-SLAM |
|---|---|---|---|---|---|---|
| V1_01 | 7.4 | 8.9 | 4.2 | 3.5 | 5.9 | 2.7 |
| V1_02 | 8.9 | 9.7 | 3.4 | 2.0 | 6.7 | 2.8 |
| V1_03 | 10.1 | 10.1 | 5.7 | 4.8 | 9.6 | - |
| V2_01* | 7.0 | 13.6 | 7.2 | 3.7 | 4.0 | 3.2 |
| V2_02* | 4.8 | 27.6 | 6.9 | 3.5 | 6.2 | 4.1 |
| V2_03* | 67.5 | 72.4 | 17.3 | - | 17.4 | 7.4 |
| MH_01 | 7.4 | 10.0 | 2.8 | 3.5 | 6.2 | 7.5 |
| MH_02 | 7.4 | 5.6 | 4.2 | 1.8 | 4.4 | 8.4 |
| MH_03 | 9.2 | 17.0 | 3.8 | 2.8 | 11.7 | 8.7 |
| MH_04 | 8.6 | 28.1 | 9.6 | 11.9 | 13.2 | 21.7 |
| MH_05 | 8.7 | 20.3 | 5.8 | 6.0 | 12.1 | 8.2 |

Next evaluation on Table 3 were done using the absolute evaluation metric only as it is almost the only one used on Euroc dataset. Here we compare:

- Three inertial aided stereo vision algorithm: RADVO (ours), SOFT-VO and SOFT-SLAM
- Two inertial aided monocular visual system, VIDSO and VI ORB-SLAM
- One stereo odometry algorithm: ORB-SLAM2

We see again that the three methods that are able to close loops are the best performing (ORB-SLAM2, VI ORB-SLAM and SOFT-SLAM). We also see that the Euroc dataset is slightly more challenging. It seems that to process the most difficult logs inertial unit are needed, and SLAM seems important to obtain convenient results on difficult dataset such as V2_03.

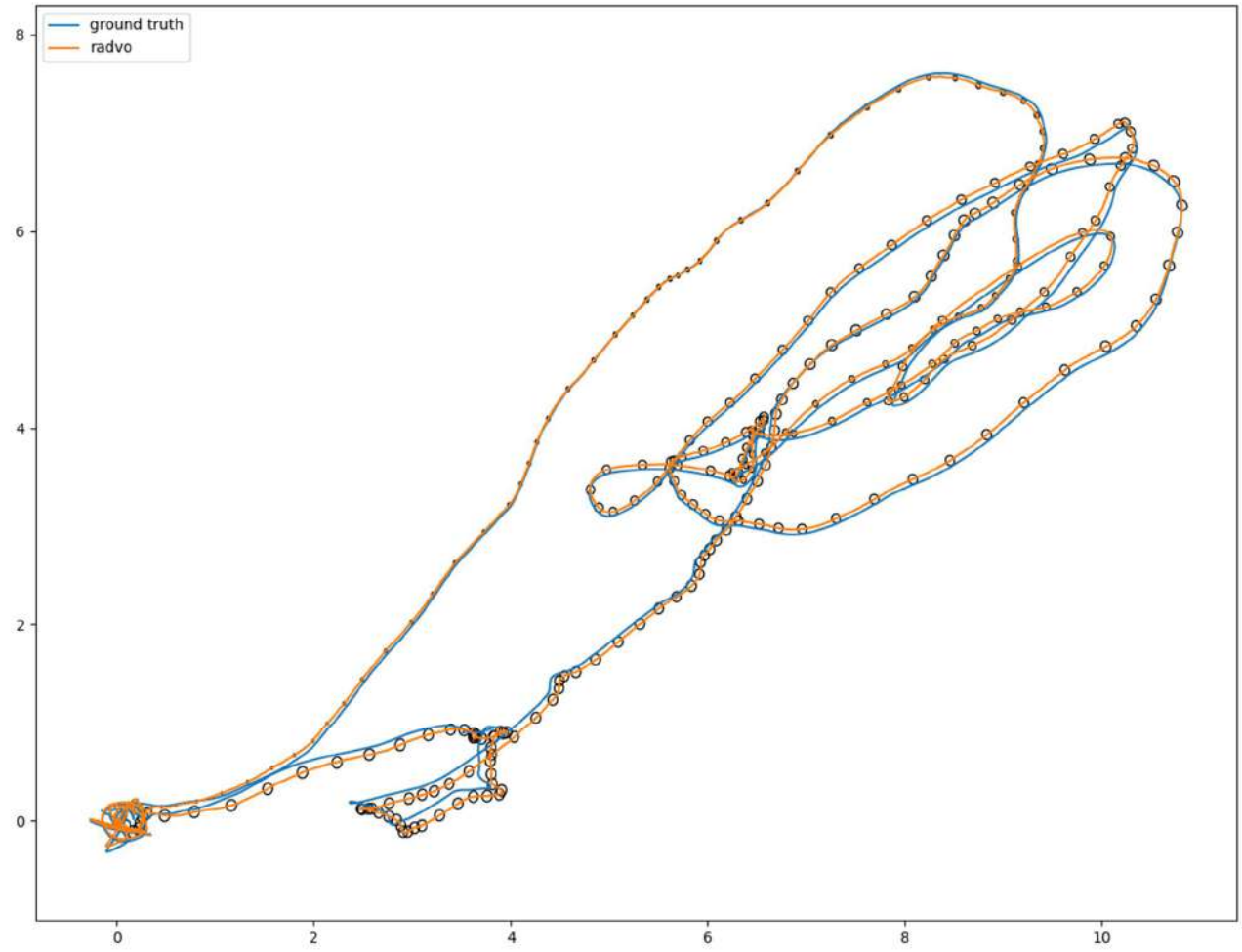


*Figure 8: Results on MH_01 dataset (orange : ours , blue :ground truth, black : uncertainty ellipses)*

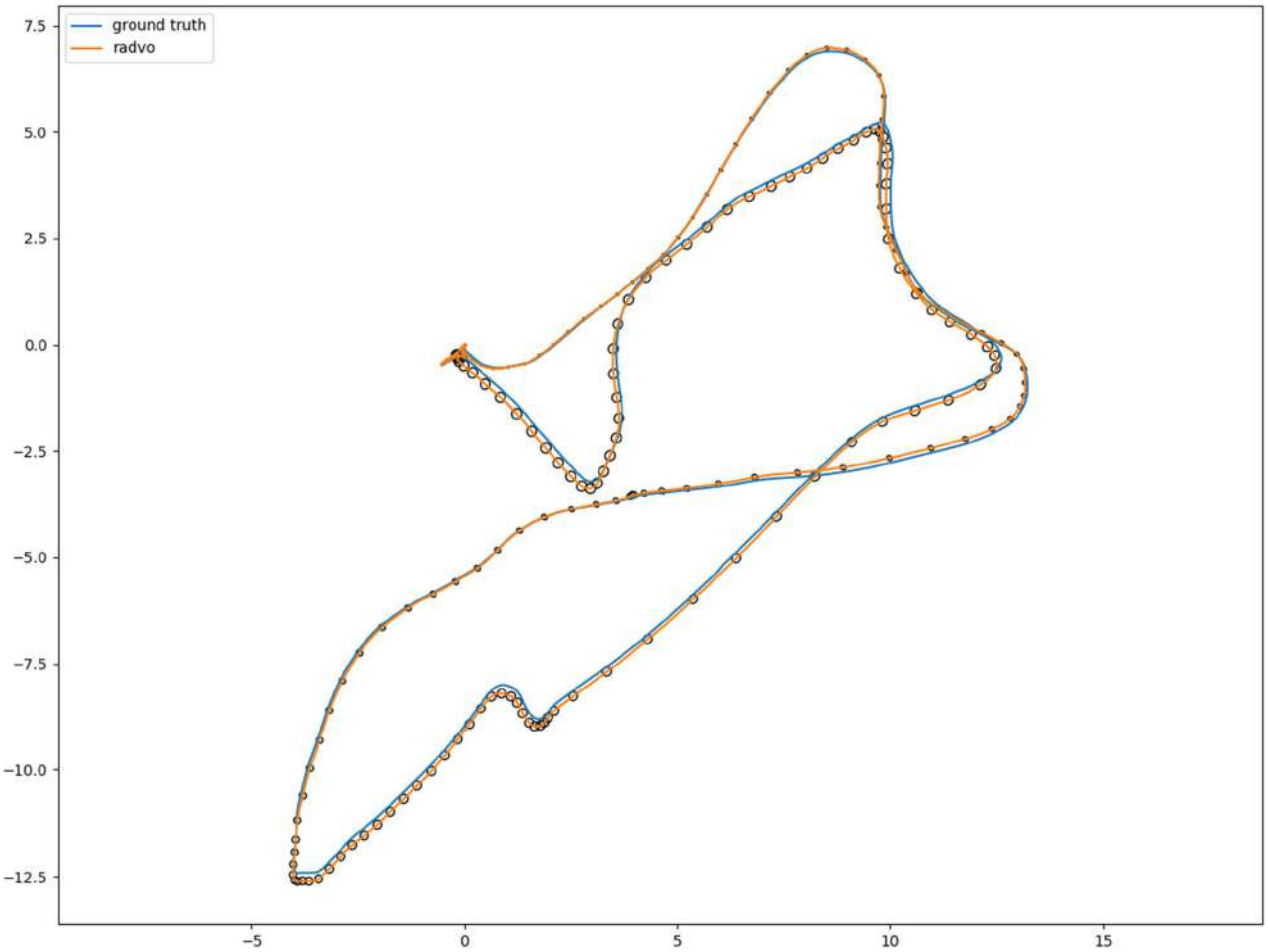


*Figure 9: Results on MH_05 dataset (orange : ours , blue : ground truth, black : uncertainty ellipses)*

## CONCLUSION

We developed a robust and accurate light weights visual odometry algorithm that have the interesting property of being deterministic. Experiments showed that the algorithm is general enough to produce state of the art results on automotive scenario and drone scenario. We also found the benefits of using an inertial momentum unit to improve results. We developed a strong and simple visual odometry algorithm that relies mainly on a direct derivation of probabilistic theory (ie. The Kalman filter) instead of dedicated algorithm for computer vision such as RANSAC. This has the overall consequence of making our algorithm robust.
However our algorithm still have a margin of improvement since it is not able to relocalize. And it has some performance loss on most difficult dataset V2-03 of Euroc. Depending on the application of our proposed algorithm this may not be a problem. If our algorithm is used with higher grade imu like an Ellipse or less chaotic motion profile such as typical survey.

## REFERENCES

1. Mur-Artal, R. and Tardós, J.D., "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras." *IEEE Transactions on Robotics*, 33(5), 2017, pp.1255-1262
2. Cvišić, I., Ćesić, J., Marković, I. and Petrović, I., "SOFT-SLAM: Computationally efficient stereo visual simultaneous localization and mapping for autonomous unmanned aerial vehicles." *Journal of field robotics*, 35(4), 2018, pp.578-595.

3. Geiger, A., Lenz, P. and Urtasun, R., "Are we ready for autonomous driving? the kitti vision benchmark suite." *IEEE Conference on Computer Vision and Pattern Recognition,* 2012, June. pp. 3354-3361.
4. Cvišić, I. and Petrović, I., "Stereo odometry based on careful feature selection and tracking." *IEEE European Conference on Mobile Robots (ECMR),* September 2015*,* pp. 1-6.
5. Geiger, A., Ziegler, J. and Stiller, C., "Stereoscan: Dense 3d reconstruction in real-time." *IEEE intelligent vehicles symposium (IV)* June 2011, (pp. 963-968).
6. Lui, V. and Drummond, T., "An Iterative 5-pt Algorithm for Fast and Robust Essential Matrix Estimation." *BMVC,* 2013.
7. M. Burri, J. Nikolic, P. Gohl, T. Schneider, J. Rehder, S. Omari, M. W. Achtelik, and R. Siegwart, "The euroc micro aerial vehicle datasets," *IJRR*, 2016
8. Simon, D., *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006
9. Zhu, N., Marais, J., Bétaille, D. and Berbineau, M., "GNSS position integrity in urban environments: A review of literature." *IEEE Transactions on Intelligent Transportation Systems*, *19*(9), 2018, pp. 2762-2778.
10. Fischler, M. A., & Bolles, R. C. "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography." *Communications of the ACM, 1981*, *24*(6), pp. 381-395.
11. Persson, M., Piccini, T., Felsberg, M., & Mester, R. "Robust stereo visual odometry from monocular techniques." *IEEE Intelligent Vehicles Symposium (IV),* June 2015*,* pp. 686-691
12. Buczko, M., & Willert, V., Flow-decoupled normalized reprojection error for visual odometry. *IEEE 19th International Conference on Intelligent Transportation Systems (ITSC),* November 2016, pp. 1161-1167
13. Hartley, R. I., "In defense of the eight-point algorithm." *IEEE Transactions on pattern analysis and machine intelligence*, 1997, *19*(6), pp. 580-593.
14. Nistér, D., "An efficient solution to the five-point relative pose problem." *IEEE transactions on pattern analysis and machine intelligence*, 2004, *26*(6), 756-770.
15. Sibley, G., Matthies, L., Sukhatme, G. "Sliding window filter with application to planetary landing." *Journal of Field of Robotics.* 2010, 27, pp. 587-608.
16. Engel, J., Koltun, V., & Cremers, D. "Direct sparse odometry." *IEEE transactions on pattern analysis and machine intelligence, 40(3), 2017, pp. 611-625.*
17. Civera, J., Davison, A. J., & Montiel, J. M. "Inverse depth parametrization for monocular SLAM." *Transactions on Robotics*, 24(5), pp. 932-945, 2008.
18. Wang, R., Schworer, M., & Cremers, D. "Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras.", *Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 3903-3911.*
19. Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., & Burgard, W. "g 2 o: A general framework for graph optimization." *IEEE International Conference on Robotics and Automation,* May 2011, pp. 3607-3613.
20. Mur-Artal, R.; Tardós, J.D. "Visual-inertial monocular SLAM with map reuse." *IEEE Robotics and Automation Letters*, 2017, 2, 796–803
21. Von Stumberg, L., Usenko, V., Cremers, D. "Direct sparse visual-inertial odometry using dynamic marginalization." *International Conference on Robotics and Automation*, May 2018, pp. 2510-2517
22. Qin, T., Pan, J., Cao, S., & Shen, S. "A general optimization-based framework for local odometry estimation with multiple sensors." *arXiv preprint arXiv:1901.03638*. 2019
23. Jiang, J., Niu, X., Guo, R., & Liu, J. "A hybrid sliding window optimizer for tightly-coupled vision-aided inertial navigation system." *Sensors*, 2019, 19(15), 3418.
24. Yang, N., Stumberg, L. V., Wang, R., & Cremers, D. "D3VO: Deep Depth, Deep Pose and Deep Uncertainty for Monocular Visual Odometry" *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1281-1292.*
25. Terejanu, G. A. "Discrete kalman filter tutorial." *University at Buffalo, Department of Computer Science and Engineering, NY*, *14260*. 2013
26. Barron, J. T. "A general and adaptive robust loss function." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019,* pp. 4331-4339.
27. Zhu J. "Image Gradient-based Joint Direct Visual Odometry for Stereo Camera." *IJCAI* August 2017 (pp. 4558-4564).