



HAL
open science

Parallax beam: a vision-based motion estimation method robust to nearly planar scenes

Martin Rebert, David Monnin, Stephane Bazeille, Christophe Cudel

► To cite this version:

Martin Rebert, David Monnin, Stephane Bazeille, Christophe Cudel. Parallax beam: a vision-based motion estimation method robust to nearly planar scenes. *Journal of Electronic Imaging*, 2019, 28 (02), pp.023030. 10.1117/1.JEI.28.2.023030 . hal-03605185

HAL Id: hal-03605185

<https://hal.science/hal-03605185v1>

Submitted on 10 Mar 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Parallax beam: a vision-based motion estimation method robust to nearly planar scenes

Martin Rebert^{a,b,*}, David Monnin^a, Stéphane Bazeille^b, Christophe Cudél^b

^aFrench-German Research Institute of Saint-Louis, AVP, 5 rue du Général Cassagnou, 68301 Saint-Louis, France

^bUniversity of Haute-Alsace, IRIMAS, EA 7499, 12 rue des Frères Lumière, 68093 Mulhouse, France

Abstract. In computer vision, the epipolar geometry embeds the geometrical relationship between two views of a scene. This geometry is degenerated for planar scenes as they do not provide enough constraints to estimate it without ambiguity. Nearly planar scenes can provide the necessary constraints to resolve the ambiguity. But classic estimators such as the 5-point or 8-point algorithm combined with a random sampling strategy are likely to fail in this case because a large part of the scene is planar and it requires lots of trials to get a non-degenerated sample. However, the planar part can be associated with a homographic model and several links exist between the epipolar geometry and homographies. The epipolar geometry can indeed be recovered from at least two homographies or one homography and two non-coplanar points. The latter fits a wider variety of scenes, as it is unsure to be able to find a second homography in the non-coplanar points. This method is called plane-and-parallax. The equivalence between the parallax and the epipolar lines allows to recover the epipole as their common intersection and so the epipolar geometry. Robust implementations of the method are rarely given and we encounter several limitations in our implementation. Noisy image features and outliers make the lines not to be concurrent in a common point. Also, off-plane features are unequally influenced by the noise level. We noticed that the bigger the parallax, the lesser the noise influence. We therefore propose a new model for the parallax, that takes into account the noise on the features location to cope with the previous limitations. We call our method the *parallax beam*. The method is validated on the KITTI vision benchmark and n synthetic scenes with strong planar degeneracy. The results show that the parallax beam improves the estimation of the camera motion in scene with planar degeneracy and remains usable when there is not any particular planar structure in the scene.

Keywords: Camera motion, homography, parallax, epipolar geometry.

*Martin Rebert, martin.rebert@isl.eu

1 Introduction

Autonomous platforms have been studied extensively in the last two decades and with it their area of application also increased. Grass-cutting or vacuum cleaning robot are already at work in modern households. Today trials are taking place into supermarkets for shopping assistants, like TwinswHeel or wiiGO. In another field, farming is concerned with the new autonomous tractor prototype of Case IH. In the security area Segway has unveiled its Nimbo model, a patrol robot detecting suspicious human behavior. The most well-known subject concerns self-driving cars, which are already in action in California. The military is also active on the subject with autonomous reconnaissance drones or self-driving vehicle for logistic convoys or road-clearance. We target applications belonging to the domain of defense such as an extension of our Change Detection project presented in Ref. 1. It helps detecting Improvised Explosive Devices (IEDs) for military convoys. The aim is to facilitate and to speed up the road clearance procedure while ensuring a very high detection rate. Nowadays steps are made in making the system able to navigate autonomously. It would improve its performance, while reducing the risk of human casualties. As extensions, we are also targeting patrolling and mule robots.

These applications have in common their faculty to locate themselves in an open-world and navigate in it. For that purpose, different types of sensors are available to them. The Change Detection

system already comprises a Global Navigation Satellite Systems (GNSS) chip and a monocular camera. While the imaging sensor is a must-have for our system to detect IEDs, the GNSS data proved itself not to be precise enough for the image registration and lacks reliability due to potential jamming or due to weak signal reception in difficult conditions (urban, forest, indoor). The other mentioned systems also make use of LiDAR sensors and Inertial Measurement Units (IMUs). LiDARs perform well in structured environment, however its added value in flat environments, where no structure is available not even an asphalt road, seems little. IMUs monitor acceleration and angular rate to integrate the position and orientation of the vehicle. It requires other sensors to compensate the accumulated drift as it is completely unaware of its surroundings. For the moment, we focus on a single sensor application using a monocular camera, which has the advantage of providing a lot of information at a low price. Our camera is tilted towards the ground to maximize its visibility since threats are usually located on the ground level. This orientation also makes sense in a navigation context as the information lies below the skyline and it also reduces the risk of being blinded by the sun during sunrise or sunset.

Visual odometry is the process of recovering the trajectory of a camera from the video sequence it records. Bundle adjustment is a widely used technique to do so.²⁻⁵ While these methods operate on several views, they rely on two-view geometry for (re-)initialization. The initialization often consists in estimating the epipolar geometry between the two frames.²⁻⁴ Ref. 2, 3 estimate the essential matrix thanks to the 5-point algorithm⁶ or a variant of it. Ref. 4 uses the 8-point algorithm to compute the fundamental matrix.⁷

In our conditions the ground plane is assumed to be dominant plane in the image. The motion of the camera is then obtained by decomposing a homography and using the plane assumption to select the correct decomposition. However, when there is a strong presence of vertical structures in the scene, a wall for example, the assumption does not hold and the motion estimation fails. Examples of such scenes are provided in Fig. 1 (bottom row), while the top row shows typical scenes we encounter, hence justifying the need for a homography. To overcome the problem, the epipolar geometry can be estimated through the fundamental or essential matrix. Their estimation algorithms mentioned previously have the inconvenience of not handling well scenes with a strong planar structure, because coplanar points do not provide enough constraint to uniquely determine the epipolar geometry.⁷ Therefore, specific techniques are required for two-view geometry in nearly planar or planar scenes.

The paper starts with a review of the strategies that deal with planar degeneracies in two-view geometry. Sec. 3 continues on situating the contributions within our algorithmic chain and details them. Finally, results validating our method are given in Sec. 4. The results are obtained on synthetic and real data from the KITTI vision benchmark.⁸ Finally in Sec. 5 we conclude and give the perspectives of the this work.

2 Two-view geometry and planar degeneracies

In the literature the presence of nearly planar scenes leads to two possible mathematical models to describe the geometry between two views: the epipolar geometry and the homography. The first one is the two-view model which is used in general and the homography is specifically used for planar structures or when the point of view remains unchanged, i.e. only the orientation is varying. The methods dealing with the nearly planar scenes are often making use of both models and we could classify them in three categories depending on their use of the models.



Fig 1 Encountered scenes. Top row: scenes where the use of a homography is necessary. Bottom row: a homography can be used but our assumption to estimate the motion from it is incorrect.

1. Estimate both models independently and select the one which fits the scene better, Sec. 2.1.
2. Avoid the epipolar geometry to be estimated on a planar structure by using an homography to assess the planarity of the data used for the estimation, Sec. 2.2.
3. Estimate the epipolar geometry using one or several homographies, Sec. 2.3.

2.1 Concurrent models estimation

The first category of methods use a competition between the two models, they are both estimated, then compared according to a metric and only the best one is kept. An example of metrics for model selection is the quantity of information needed to encode the model⁹ or a ratio between the inliers subsets of the two models associated with a threshold to choose between the two.⁴ The main critic that can be made for these methods is that both models are estimated in parallel and from scratch to finally reject one, so half of the computation effort is wasted.

2.2 Detecting and avoiding planar degeneracies

The second category of methods tries to detect and avoid degenerate configurations while estimating the fundamental matrix. The BEEM algorithm¹⁰ relies on an efficient data exploration scheme to escape degeneracies. It relies on the SIFT¹¹ features and its extension to features like Harris corners¹² is not straightforward. We mention Harris corners because they appeared to be the best approach for our application.¹³ Moreover, SIFT features are also not the best choice for real-time systems. The QDEGSAC¹⁴ method is a variant of RANSAC¹⁵ testing the number of constraints provided by the data with successive RANSAC loops, if less constraints are given and the same result is obtained, then the initial data did not provide enough constraint, hence it is likely degenerated. It is however failing, when the data is indeed degenerated. In Ref. 16, the degeneracy is

tested by successively testing motion models with decreasing degrees of freedom (DOF). If almost the same inlier support is obtained by the model with less DOF, then the data is degenerated for the previous. It has the advantage of dealing with the special cases of the pure translation, the pure rotation and the no-motion, but four different models might need to be tested for a single image pair. The DEGENSAC¹⁷ is also a RANSAC based approach, but instead of successive model estimations, it detects the degeneracies within the RANSAC sample test. If one part of the sample test is coherent with a homography, then the fundamental matrix is recalculated using the plane-and-parallax principle,⁷ compared to the one obtained by the 7-point algorithm and finally the best one is kept. A critic can be made regarding the computational effort, choosing the 7-point algorithm as the entry point leads to minimum three fundamental matrices to test in each loop and four plus five homographies in the worst case. The methods in this category do not allow to recover the camera motion when the data are actually planar. Only the method in Ref. 16 comes near as it deals with the pure rotation motion, which implies a homography. Unfortunately, it does not deal with motion and planar case.

2.3 Exploiting planar degeneracies

The third category of methods recovers the epipolar geometry by exploiting the degeneracies from scenes. They usually concentrate on scenes, in which several planes could be found. It would mean that we have dominant plane in the image and the remaining part of the image is also planar (Fig. 1 bottom right image). These methods do not perform any detection or avoidance step, the degeneracy is in contrary plainly used. Among these methods a distinction can be made on whether they rely on at least two homographies or only one. The most straightforward method consists in a robust estimation of at least two different homographies from an image pair¹⁸ and use their combined inlier sets to estimate the fundamental matrix. Alternatively, one can use the compatibility equation⁷ between the homographies and the fundamental matrix to estimate its parameters. The latter needs a proper normalization,¹⁹ otherwise it is unstable.²⁰ Moreover, the two independently estimated homographies are not necessarily compatible with the epipolar geometry²¹ leading to inconsistent motions, i.e. two homographies can lead to different sets of epipolar lines, hence different motions. From the two homographies, a homology transferring the epipole onto itself can also be built.²² The epipole corresponds then to the eigenvector associated with the unique eigenvalue, the two others being identical. However, with real noisy data, it is not guaranteed to have two identical eigenvalues, then choosing the unique one becomes problematic. If it is not possible to rely on finding two independent homographies, e.g. there is only one plane in the scene, the epipolar geometry can still be recovered from a single homography and two off-plane points. In Ref. 22 the author lists different least squares problems to recover the fundamental matrix from a single homography, however nothing is said regarding outliers rejection and initial values for the minimization. A closer look indicates that the epipole is needed to recover the fundamental matrix from a single homography. The epipole is found using the plane-and-parallax approach,^{7,19,23} which in the given form presents several drawbacks. It treats all the parallax vectors identically, but intuitively points that lie closest to the plane provide less reliable information. Additionally, it is sensitive to noise, when off-plane points are scarce leading to incorrect epipole locations.

2.4 Toward a new approach dealing with planar degeneracies

Our application makes the presence of planar scenes recurrent. Therefore, the third category of methods seems the most adapted as often an homography would suffice and we would like to recover the epipolar geometry only when necessary. Most of the methods in this category rely on finding several homographies associated with different planar structures, which is not a reliable assumption in our case. We want to rely only on the homography that would normally be estimated by our system and to reuse it as much as possible if it appears not to be sufficient. The plane-and-parallax approach appears to be our candidate.

To overcome its limitations previously established, we introduce a new model for the parallax, that we call the parallax beam. The parallax beam method provides a way to recover the epipolar geometry when the off-plane data is scarce and it also has the advantage of taking into account the noise level on the data combined with a reliability measure for the off-plane data. We will present our method from the feature detection to the motion estimation with a focus on our parallax beam model to recover the epipolar geometry and on the refinement step we use after it.

3 From a homography and parallax to motion parameters

3.1 System overview

To place the work within our processing chain, an overview of it is given in Fig. 2. The input data are two images from the same scene and the camera has slightly moved between the two shots. Then image features are detected and described in each image. The feature detection uses a modified Harris corner detector,¹³ which ensures a homogeneous feature distribution over the whole image. The description is made by the BRIEF descriptor and the features are matched based on a Hamming distance between descriptors, while preserving the homogeneous distribution over the image. A homography is robustly estimated over the feature correspondences using a modified RANSAC procedure. Each RANSAC sample is tested for folding.²⁴ If the test is passed, then the standard Direct Linear Transformation algorithm is used to estimate the homography from four correspondences.⁷ The validity of the model is tested by the symmetric projection error. The homography is then refined on the biggest set of inliers and normalized.²⁵ Afterwards a test on the size of the set of inliers is performed. If the size is bigger than 90 % of the number of feature correspondences, then the scene is considered planar. The motion of the camera is recovered from the homography matrix.²⁶ The correct solution is selected using the assumption of the ground plane.

Otherwise, the scene is likely not to be planar and the homographic model does probably not embed all the information from the image features. In that case, recovering the epipolar geometry is of interest. To that end, we reuse as much as possible the results previously obtained and use the estimated homography in a plane-and-parallax method to get an approximate position of the epipole. The parallax model to locate the epipole is our first contribution and is referred as the *parallax beam model*. From the epipole and the homography matrix, the fundamental matrix can be estimated and decomposed to recover the camera motion. Once a rough translation and rotation are available, they are refined using a nonlinear least-squares procedure.

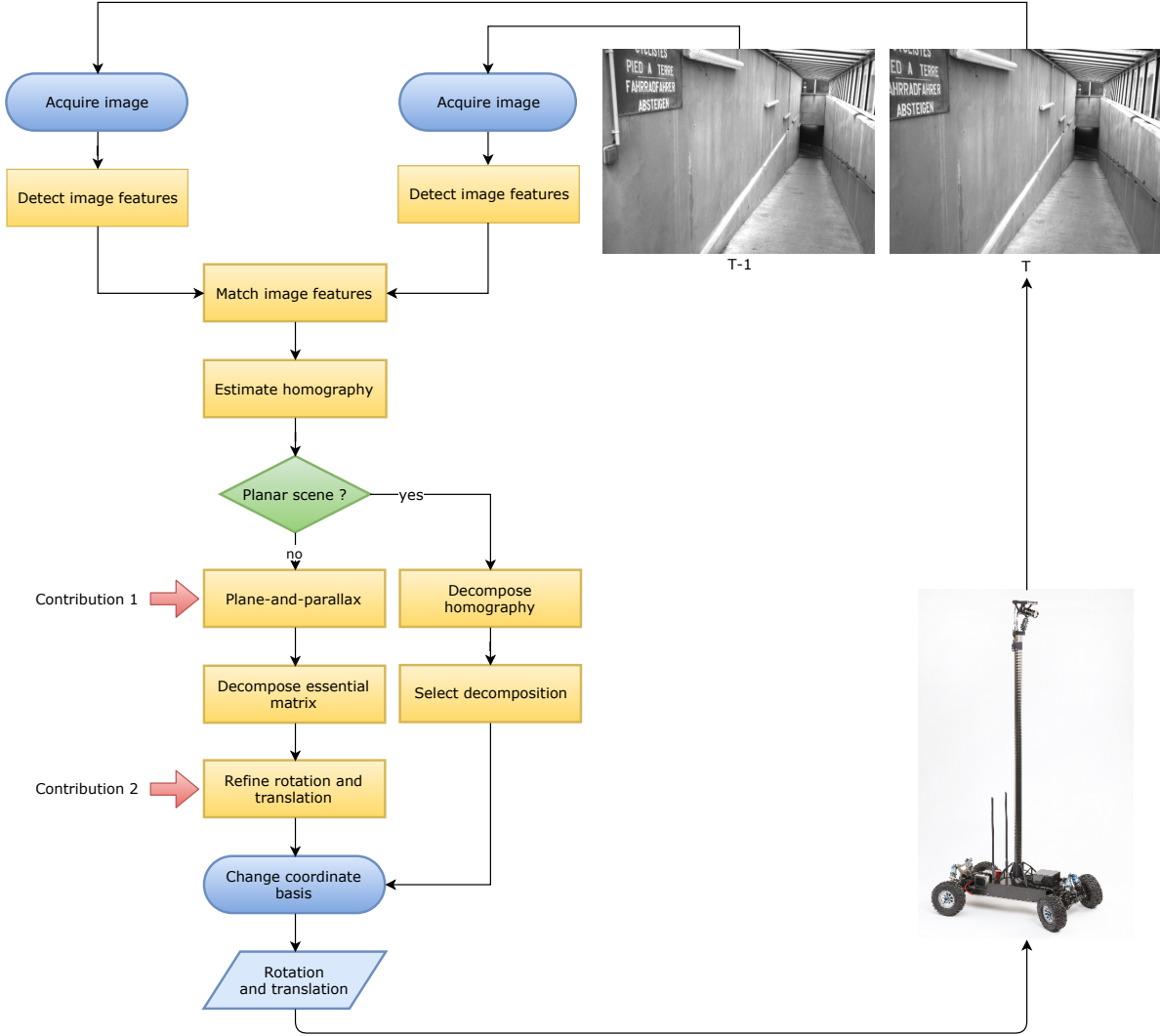


Fig 2 Flow chart of our method. This method robustly estimates the camera motion in the presence of planar degeneracies.

3.2 Parallax beam: a new parallax paradigm

The plane-and-parallax formulation⁷ estimates the fundamental matrix from image features, which are partially coplanar. The coplanar ones are transferred from one image plane to the other by a homography \mathbf{H} . For the corresponding image features \mathbf{x}_i and \mathbf{x}'_i , we have $\mathbf{x}'_i = \mathbf{H}\mathbf{x}_i$ if they are on the homographic plane. For the ones not in the support of the homography, $\mathbf{x}'_i \neq \mathbf{H}\mathbf{x}_i$ and the line joining \mathbf{x}'_i and $\mathbf{H}\mathbf{x}_i$ is designated here as the parallax line. This line and the epipolar line through the feature \mathbf{x}'_i are theoretically the same.⁷ From the definition of epipolar lines, if at least two of them are available, the epipole is at their intersection, as in Fig. 3(a). Given the epipole and a homography matrix, we know from Ref. 7 that the fundamental matrix is recovered from:

$$\mathbf{F} = [\mathbf{e}']_{\times} \mathbf{H}. \quad (1)$$

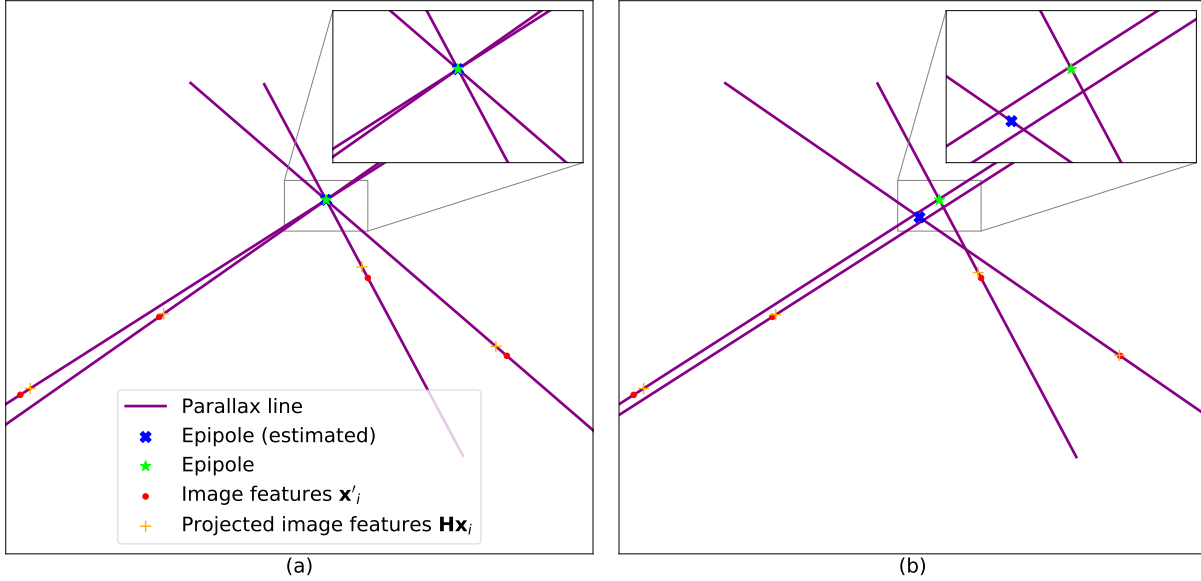


Fig 3 (a) In a noise-free situation, the parallax lines all intersect at the position of the epipole. (b) The noise from the corner detection and the estimation of the homography disrupt lines intersection.

3.2.1 Epipole from parallax lines: the standard approach

Using the plane-and-parallax principle on real data is less straightforward, imprecisions on the image features detection and on the homography estimation result in parallax lines not having a common intersection, as in Fig. 3(b). In the figure, the image features and the projected image features have disrupted by a Gaussian noise of zero mean and a standard deviation of 0.17. The presence of outliers in the data also adds difficulty to the task. The problem of finding the epipole becomes the problem of finding the point, where most lines seem to converge. Besides the description of the plane-and-parallax in Ref. 7, we did not find a detailed version of the algorithm dealing with outliers, so we propose the one in Algorithm 1, which is suggested by common sense and uses standard techniques. From our experience, the method presents several limitations. First,

Algorithm 1 Epipole from parallax lines

```

while  $iteration < maxIteration$  do
  Select 2 lines
  Compute their intersection point
  Count how many lines pass, within a given threshold, near the intersection point
  if count > biggestCountSoFar then
    Remember the intersection point and the lines passing near it
  end if
end while
Apply a least-squares to find the closest point to the remembered lines

```

the epipole might not be estimated correctly, when there is an even number of lines that seems to intersect in two different points, as in Fig. 3(b). Second, the parallax lines are not equally influenced by the noise. The smaller the parallax size, the more the noise can change the orientation of

the line, but the lines are all considered in the same way to estimate the epipole. Third, the method relies on a threshold defining the closeness to the intersection point and its value is difficult to set meaningfully. The values between 10 and 100 pixels are usually good guesses, but it is quite scene dependent to select an optimal value within these boundaries.

3.2.2 Parallax beam: Robust epipole estimation

To overcome these limitations, we propose our parallax beam paradigm, which relies on the following considerations:

1. the position of image features is corrupted with noise,
2. the projection of image features by \mathbf{H} is corrupted with noise,
3. image features close to the homographic plane are less reliable than the farther ones.

When the position of the image features is corrupted with noise, the true position lies in the vicinity of the detected point. Usually the noise is assumed Gaussian with a standard deviation σ . For the sake of simplicity, we differ by considering a uniform circular noise around the detected image feature depicted in Fig. 4(a). The uniform circular noise is defined by its radius r . Even though the inner distribution is different, by taking $r = 3\sigma$ we ensure that 99 % of the Gaussian values fall within the circle boundaries. Since the method is intended to work on consecutive images, we consider the distortion induced by the homography to be small enough so that the circular noise can still apply on the projected feature. The circles around the image feature and its homographic projection define a set of possible parallax lines, Fig. 4. The orientation of these lines is contained between the two diagonal common tangents of the two circles. Therefore, by considering the area between the tangents, that contains both circles, we have all the parallax lines that could be obtained from our noisy feature points. This area is referred hereafter as the *parallax beam*. When comparing with Fig. 4(b), we note that the parallax distance influences the opening angle of the beam. The noise has more influence on an epipolar line obtained if the points \mathbf{x}'_i and $\mathbf{H}\mathbf{x}_i$ are close together.

The parallax beam is completely defined using the points \mathbf{m} , \mathbf{t}_1 and \mathbf{t}_2 depicted in Fig. 5(a). \mathbf{m} is the middle of the segment $[\mathbf{x}'_i, \mathbf{H}\mathbf{x}_i]$, \mathbf{t}_1 and \mathbf{t}_2 are the points of tangency through \mathbf{m} . To get \mathbf{t}_1 and \mathbf{t}_2 , the origin is shifted to $\mathbf{H}\mathbf{x}_i$. In this coordinate frame, we write $\mathbf{m} = (x_m, y_m)$ and $\mathbf{t}_1 = (x, y)$. Because \mathbf{t}_1 is on the circle centered on $\mathbf{H}\mathbf{x}_i$ and of radius r we have,

$$r^2 = x^2 + y^2 \tag{2}$$

From the Pythagorean theorem $l^2 = r^2 + (x - x_m)^2 + (y - y_m)^2$ and since $x_m^2 + y_m^2 = l^2$ and Eq. 2 it induces,

$$x = \frac{r^2 - y_m y}{x_m} \tag{3}$$

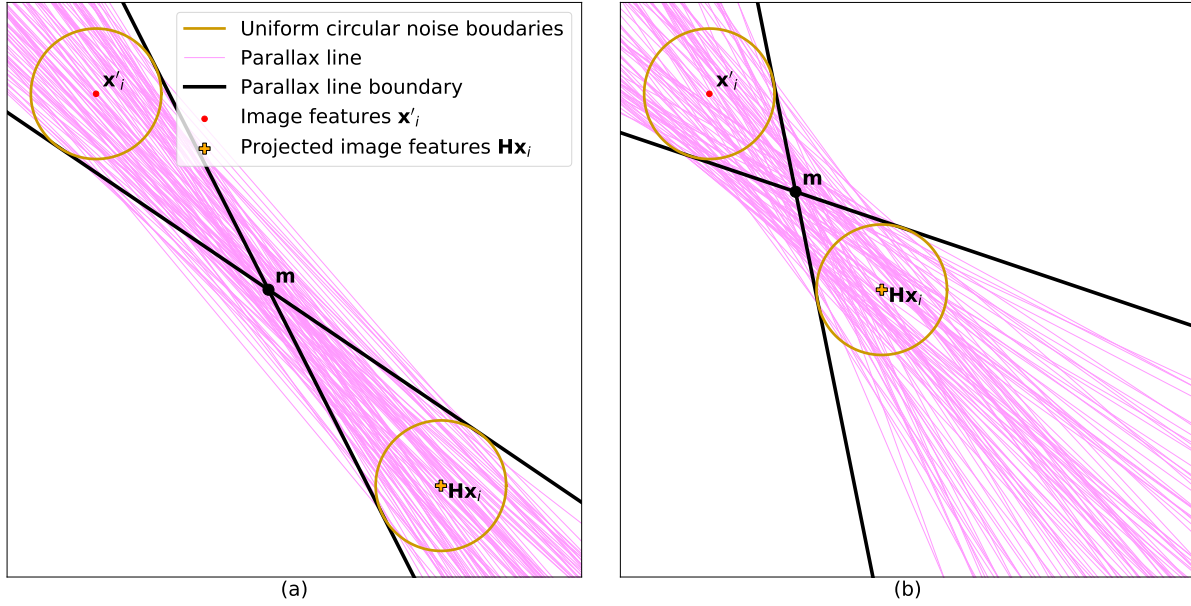


Fig 4 A set of parallax lines can be generated from points within the circles, the black lines represent the boundaries for the orientation of the parallax lines. (a) The distance between x'_i and Hx_i is 4 pixels. (b) The distance is 2 pixels, i.e. closer to the homographic plane.

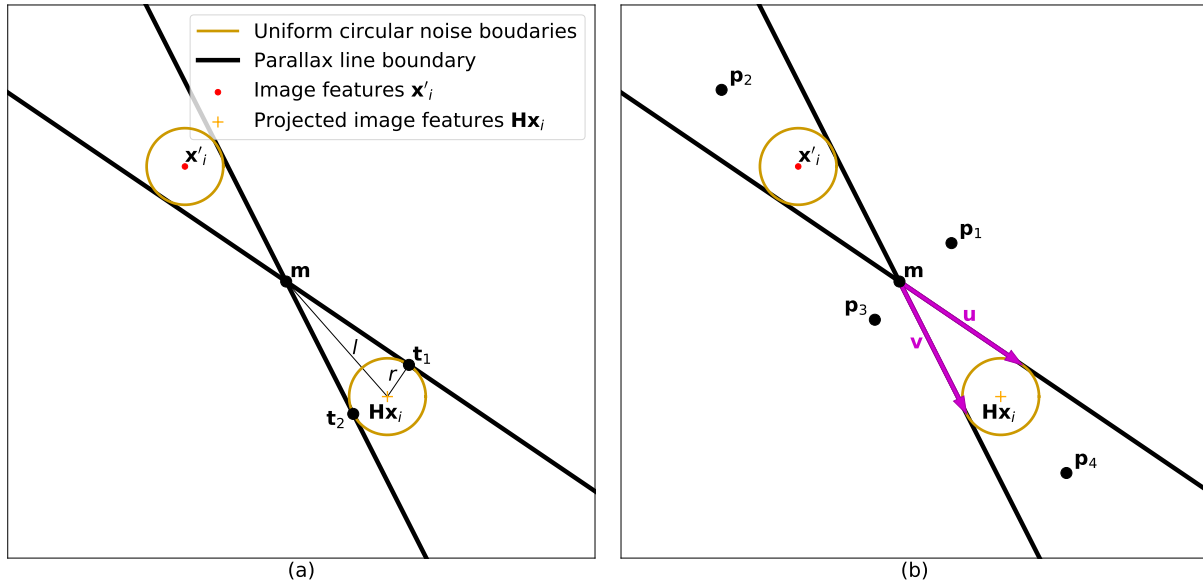


Fig 5 (a) Computing the boundaries of the parallax lines contained in the parallax beam. (b) The points p_i correspond to the four cases to test if a point lies within the parallax beam.

Eq. 3 is substituted into Eq. 2 to obtain a 2^{nd} degree polynomial in y , whose roots are the y -

coordinates of \mathbf{t}_1 and \mathbf{t}_2 provided that \mathbf{m} lies outside the circle, i.e. $l > r$,

$$0 = y^2 l^2 - y(2r^2 y_m) + r^4 - x_m^2 r^2$$

$$y = \frac{r \left(r y_m \pm \sqrt{(r y_m)^2 + (l x_m)^2 - (l r)^2} \right)}{l^2}. \quad (4)$$

The x -coordinate is obtained from y using Eq. 3. Finally, \mathbf{t}_1 and \mathbf{t}_2 must be shifted back to the image origin by adding $\mathbf{H}\mathbf{x}_i$. Having \mathbf{m} , \mathbf{t}_1 and \mathbf{t}_2 , the two lines are defined and are now referred as the beam boundaries.

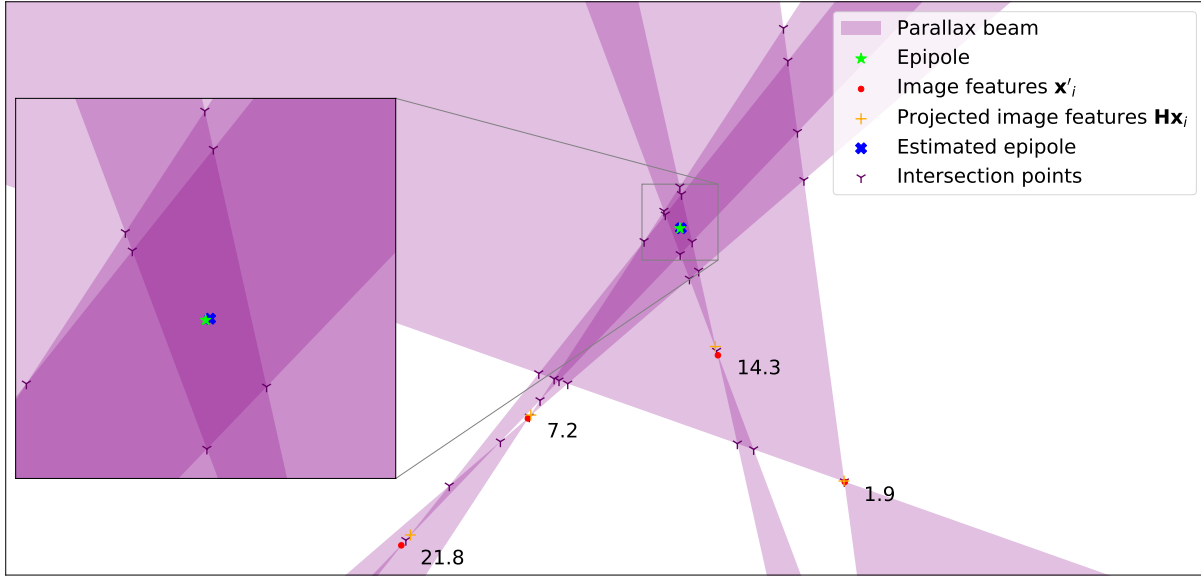


Fig 6 Intersection of the boundaries of the parallax beams to generate a mesh of intersection points. The epipole is estimated as the centroid of the vertices defining the area of maximum overlap. The indicated values represent the size of the parallax.

Finding the epipole from a set of parallax beams consists in finding where most of them overlap. To do so, we intersect all the boundaries of the parallax beams, which generates a mesh of intersection points, see Fig. 6. The intersection points define the vertices of polygons over which a fix number of beams overlap. Finding the area of maximum overlap is done by finding the vertices, which are in a maximum number of beams. Fig. 5(b) shows four points \mathbf{p}_i . They represent the four partitions of the 2D-space made by the beam. The coordinates of the points \mathbf{p}_i are transformed in the basis defined by \mathbf{u} and \mathbf{v} , which are straightforwardly obtained from \mathbf{m} , \mathbf{t}_1 and \mathbf{t}_2 . In this basis, a point lying in the beam has its coordinates of equal sign. A test on the sign of the coordinates of the vertices indicates, whether a point lies inside or outside the beam, see Table 1. The epipole is finally defined as the centroid of the vertices of the polygon defining the area of maximum overlap.

At this step, we have defined our parallax beam model and how to recover the epipole from it. The method is also working in the presence of outliers, in that case and once the epipole is estimated, the inliers are defined as the image features, whose beam contains the epipole. Otherwise they are

Table 1 Sign of a point coordinates in the $(\mathbf{m}, \mathbf{u}, \mathbf{v})$ basis, if it is in or out of the parallax beam.

Point in Fig. 5(b)	In/Out	\mathbf{u} -coordinate	\mathbf{v} -coordinate
\mathbf{p}_1	Out	+	-
\mathbf{p}_2	In	-	-
\mathbf{p}_3	Out	-	+
\mathbf{p}_4	In	+	+

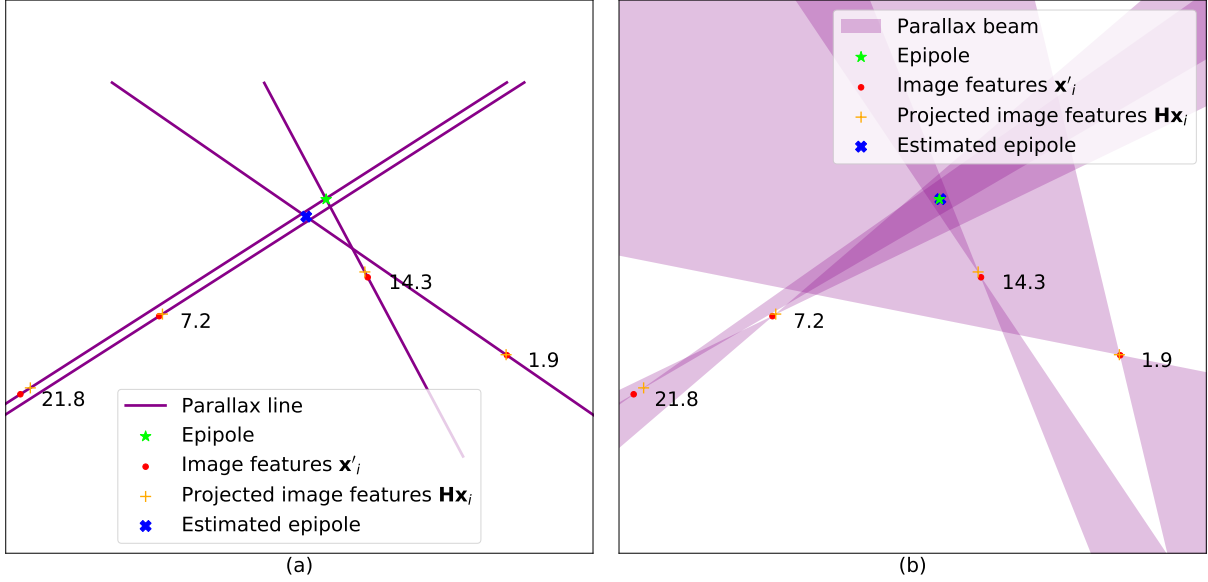


Fig 7 Image features locations are corrupted with a white Gaussian noise with $\sigma = 0.17$. (a) The standard parallax method fails to estimate the epipole correctly. (b) Our parallax beam successfully retrieves an estimation of the epipole. The number indicates the size of the parallax.

considered as outliers. Different size of parallax were used to generate Fig. 6 to highlight another interest of the method. Since the diagonal tangents of both circles define the envelop of the parallax beam, the opening of the beam, i.e. the angle between the tangents, also depends on the size of the parallax. The smaller the size of the parallax, the more open is the parallax beam. This is in accordance with the intuition that in the presence of noise, the points close to the homographic plane are less reliable to estimate the epipole. We can see in Fig. 6 that the parallax beam generated from a parallax of 1.9 pixels has a wider opening than the beam generated from the parallax of 21.8 pixels. Thus it provides a weaker constraint on the position of the epipole due to the bigger area covered by the beam. Our new parallax beam paradigm brings then the following improvements:

- It models the noise on the image features by considering an uncertainty area around them. This area is a uniform circular distribution of points from which a set of likely parallax lines is derived. This set is represented by our parallax beam. We gave a key to relate the uniform circular distribution to the classic Gaussian noise model.
- The reliability of the image features, depending on the distance to the homographic plane, is included in the approach. The further, the more reliable because the noise has less influence.

- It relies on an estimation of the standard deviation of the noise, which has more physical meaning than the notion of nearness for the intersection of lines.

Finally, Fig. 7 illustrates the capacity of our approach compared to the standard line intersection in the presence of noise. In that case the plane-and-parallax from the intersection of lines fails. It actually has a 50 % chance of finding the correct epipole location depending on the order the RANSAC trials. One could also filter the parallax lines to use based on the length of the parallax, which would solve the problem in Fig. 7. However, such threshold would also remove small but correct parallax, that our method is able to use but it grants it a smaller influence in the estimation process.

3.3 Motion Refinement

The procedure proposed in the previous section to distinguish inliers from outliers does not guarantee a perfectly outlier-proof result. To refine either the essential matrix or the motion once decomposed, it is common to apply a non-linear refinement on the parameters to estimate. However, if performed naively, this procedure performs poorly in the presence of outliers in the dataset. We therefore propose a non-linear refinement step, which operates on the rotation and translation parameters and that is robust to the presence of outliers in the observations.

The motion is composed of a rotation \mathbf{R} and a translation \mathbf{t} . Once the fundamental matrix has been estimated, the motion between the two views can be recovered up-to-scale by the decomposition of the essential matrix \mathbf{E} .⁷ The essential matrix is the algebraic representation of the epipolar geometry in the calibrated case and is related to \mathbf{F} by

$$\mathbf{E} = \mathbf{K}^T \mathbf{F} \mathbf{K}, \quad (5)$$

where \mathbf{K} is the calibration matrix containing the intrinsic parameters of the camera. The decomposition of \mathbf{E} give four solutions, among which only one satisfies the positive depth constraint, requiring the scene to be visible from both camera poses. At this point a first estimation of the camera motion depicted by the matrix \mathbf{R}_{ini} and the vector \mathbf{t}_{ini} is available. To increase the precision on the motion recovery, the rotation and translation components need to be refined using non-linear least-squares.

A standard approach to constraint the norm of the quaternion during the least-squares would make use of the Lagrangian multipliers,²⁷ but it has the drawback of increasing the number parameters to estimate. On the opposite, the work of Ref. 28 makes use of the tangent hyperplan to the hypersphere in \mathbb{R}^4 containing all the unit quaternions. It reduces the number of parameters to be estimated from four to three while ensuring the unit norm constraint. More recently Ref. 29 achieved the same results by using the intersection of a ray from a reference pole to the quaternion on the hypersphere. The quaternion is then represented by the intersection of the ray with the equatorial plane. It has the advantage of avoiding the use of a SVD and having simple partial derivatives. Concerning the translation, it has only two degrees of freedom, since the motion is recovered up-to-scale. Therefore, the translation vector can be constrained to unit norm and the parametrization of Ref. 29 is adapted to a 3D sphere and a 2D equatorial plane. Thus, we have a minimal parametrization of the motion, three parameters for the rotation and two for the translation.

The Sampson distance is the function to be minimized because of its accurate estimation of the re-projection error and its small runtime.³⁰

$$\epsilon_{Sampson} = \frac{\mathbf{x}'_i{}^T \mathbf{E} \mathbf{x}_i}{\sqrt{(\mathbf{E} \mathbf{x}_i)_1^2 + (\mathbf{E} \mathbf{x}_i)_2^2 + (\mathbf{E}^T \mathbf{x}'_i)_1^2 + (\mathbf{E}^T \mathbf{x}'_i)_2^2}}, \quad (6)$$

with $\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R}$ and $(\mathbf{E} \mathbf{x}_i)_j$ standing for the j^{th} coordinate of the vector $(\mathbf{E} \mathbf{x}_i)$. Since the inliers selection from Sec. 3.2.2 is a bit permissive, the data used as input for the nonlinear least-squares estimation might still contain outliers. Therefore, a robust cost function should be used. Following the conclusions of Ref.31, the Geman-McClure function is chosen for its good convergence capability even with poor initial conditions,

$$\rho(\epsilon) = \frac{1}{2} \frac{\epsilon^2}{1 + \epsilon^2}. \quad (7)$$

Although the Levenberg-Marquardt algorithm is the classic non-linear least squares solver in the field of computer vision, the work of Ref. 32 reached the conclusion that the Powell's dogleg algorithm could be used to achieved the same precision with a reduced computational cost. The algorithm used is described in Ref. 33 and the Jacobian matrices are computed analytically using automatic differentiation to avoid the approximation and computational cost of finite differences.

4 Evaluation of the parallax beam method

4.1 Conditions of evaluation

The parallax beam method is compared with the plane-and-parallax algorithm given Alg. 1, the 5-point algorithm⁶ and the 8-point algorithm.³⁴ They are evaluated in a two-view configuration with the camera 1 having the projection matrix $\mathbf{P}_1 = \mathbf{K} [\mathbf{I}_3 | \mathbf{0}]$ and the camera 2 having the projection matrix $\mathbf{P}_2 = \mathbf{K} [\mathbf{R} | \mathbf{t}]$, where \mathbf{K} is the intrinsic matrix, \mathbf{R} is the rotation of the camera between the two views and \mathbf{t} is the translation of the camera between the two views. The methods are evaluated on their ability to recover \mathbf{R} and \mathbf{t} . We evaluate them twice; the first evaluation is done on the output of the above-mentioned methods and the second evaluation takes place after a refinement step. For the parallax beam the refinement is the one proposed in the previous section. For the others, we use the Levenberg-Marquardt, which is well known in computer vision. The two-step evaluation allows use to evaluate the parallax beam independently from its refinement step. The values of \mathbf{R} and \mathbf{t} recovered at each step are compared the ground truth values being \mathbf{R}_{gt} and \mathbf{t}_{gt} . The rotation error ϵ_R expresses the error between \mathbf{R} and \mathbf{R}_{gt} . We express the difference rotation matrix $\mathbf{R}_{diff} = \mathbf{R}^T \mathbf{R}_{gt}$, which is the identity matrix if the two rotation matrices are equal. Using the property of $SO(3)$ that $tr(\mathbf{R}) = 1 + 2 \cos(\theta)$, ϵ_R is defined as:

$$\epsilon_R = \arccos \left(\frac{tr(\mathbf{R}_{diff}) - 1}{2} \right).$$

ϵ_R corresponds physically to the angle of \mathbf{R}_{diff} of the axis-angle representation of a rotation in 3D.

The translation error ϵ_t expressed the error between \mathbf{t} and \mathbf{t}_{gt} . In two-view geometry the translation vector can only be recovered up-to-scale, therefore a metric including the norm is not suited. We

are rather interested to evaluate if \mathbf{t} points in the same direction as \mathbf{t}_{gt} . ϵ_t is then defined as:

$$\epsilon_t = \arccos \left(\frac{\mathbf{t} \cdot \mathbf{t}_{gt}}{\|\mathbf{t}\| \|\mathbf{t}_{gt}\|} \right)$$

and corresponds to the angle between the two vectors.

4.2 On general scenes from the KITTI Vision Benchmark

We use the KITTI vision benchmark⁸ sequences for visual odometry as the first 11 sequences have a frame per frame ground truth. For each image of these sequences the camera pose is known, it is therefore straightforward to have the motion of the camera between two images. The comparison of the methods is performed on the sequences 00, 02, 03 and 10 and we use the first 500 images of each sequences. The statistics are presented in Table. 2. Figure 8 presents graphically the maximum error obtained for each method and for each sequence. We present the maximum error instead of the mean because our main concern is the robustness.

From the curve representing the parallax beam, we can see that the method exhibits slightly higher error than the 5-point algorithm and the parallax line method. However, the 8-point algorithm appears to be more unstable than the other methods, as high errors on both translation and rotation are occurring. When estimating the rotation component of the motion all the three methods behave almost identically. On the other hand, the translation component the methods perform in a less similar way, indeed the 5-point algorithm appears to be the best. Nevertheless, the parallax beam and the parallax line lie close behind and can also provide reliable results for motion estimation.

However, the KITTI sequences do not provide scenes with strong planar degeneracy to prove the interest of the parallax beam method. Nevertheless, the KITTI sequences proved our method to be usable in generic scenes, where there is not any particular plane dominating the image content. After analyzing the images, we notice that most of the time the homography used in the parallax beam fits a virtual plane, i.e. it does not correspond to any physical plane in the scene. It relaxes the constraint of having a well identify plane in the scene to use the parallax beam method, which extends its scenarios of usage. Even though we still need to prove its interest compared to the parallax line and the 5-point algorithm.

4.2.1 On synthetic scenes with strong planar degeneracies

In order to challenge the methods in scenes with strong planar degeneracies, we decided to use a synthetic scene to be able to generate an accurate ground truth. The scene is composed of two planes, one plane being the ground and the other one being a wall as shown in Fig. 9. The motion between the camera P_1 and P_2 is taken from successive poses of a KITTI ground truth file. The camera P_1 has always the expression written in Sec. 4.1 and is rotated by 20° towards the ground. We use the ground truth files of the KITTI vision benchmark to ensure that the motion is coherent for a vehicle using the Ackermann steering geometry (car-like). We make four sequences, which differ by the distance from the wall for P_1 :

d = 2.5 m : the camera P_1 is close to the wall. There is a strong planar degeneracy as only a small part of the ground is visible.

d = 5.0 m : in this position the wall and the ground are equally represented in the image, the planar degeneracy is the weakest of the four sequences.

Table 2 Statistics on ϵ_R and ϵ_t when estimating the camera motion on the KITTI sequences. The values are given in degrees.

Method	mean(ϵ_t)	std(ϵ_t)	max(ϵ_t)	mean(ϵ_R)	std(ϵ_R)	max(ϵ_R)
Sequence 00						
Before Refinement						
Beam	2.17	2.07	20.97	1.52	2.0	7.9
5-point	2.24	1.72	12.94	1.48	2.02	7.83
8-point	7.75	29.69	179.03	1.89	8.22	179.57
Line	2.34	2.02	13.91	1.53	1.99	8.0
After Refinement						
Beam	1.44	1.4	16.75	1.48	2.01	7.87
5-point	1.52	1.16	6.52	1.48	2.02	7.87
8-point	9.21	29.55	178.71	4.11	9.82	178.37
Line	1.38	1.19	7.88	1.48	2.02	7.88
Sequence 02						
Before Refinement						
Beam	2.16	2.07	20.97	1.52	2.0	7.9
5-point	2.25	1.72	12.94	1.48	2.02	7.83
8-point	7.77	29.72	179.03	1.9	8.23	179.57
Line	2.34	2.02	13.91	1.53	2.0	8.0
After Refinement						
Beam	1.44	1.4	16.75	1.48	2.02	7.87
5-point	1.52	1.16	6.52	1.48	2.02	7.87
8-point	9.23	29.58	178.71	4.11	9.83	178.37
Line	1.38	1.19	7.88	1.48	2.02	7.88
Sequence 03						
Before Refinement						
Beam	1.6	1.35	10.88	0.96	1.09	6.51
5-point	1.91	1.49	13.85	0.92	1.1	6.66
8-point	9.25	33.24	179.71	2.08	13.87	179.9
Line	2.19	2.11	26.81	0.98	1.09	6.52
After Refinement						
Beam	0.87	0.65	5.33	0.93	1.1	6.5
5-point	1.03	0.88	6.25	0.92	1.1	6.54
8-point	10.86	32.89	179.59	4.17	14.73	179.21
Line	0.88	0.65	5.25	0.93	1.1	6.51
Sequence 10						
Before Refinement						
Beam	2.16	2.07	20.97	1.52	2.0	7.9
5-point	2.25	1.72	12.94	1.48	2.02	7.83
8-point	7.77	29.72	179.03	1.9	8.23	179.57
Line	2.34	2.02	13.91	1.53	2.0	8.0
After Refinement						
Beam	1.44	1.4	16.75	1.48	2.02	7.87
5-point	1.52	1.16	6.52	1.48	2.02	7.87
8-point	9.23	29.58	178.71	4.11	9.83	178.37
Line	1.38	1.19	7.88	1.48	2.02	7.88

$d = 10.0 \text{ m}$: in this position the ground is the dominant plane, but the wall still represents around 25 % of the image.

$d = 15.0 \text{ m}$: the camera P_1 is relatively far from the wall. There is a strong planar degeneracy as

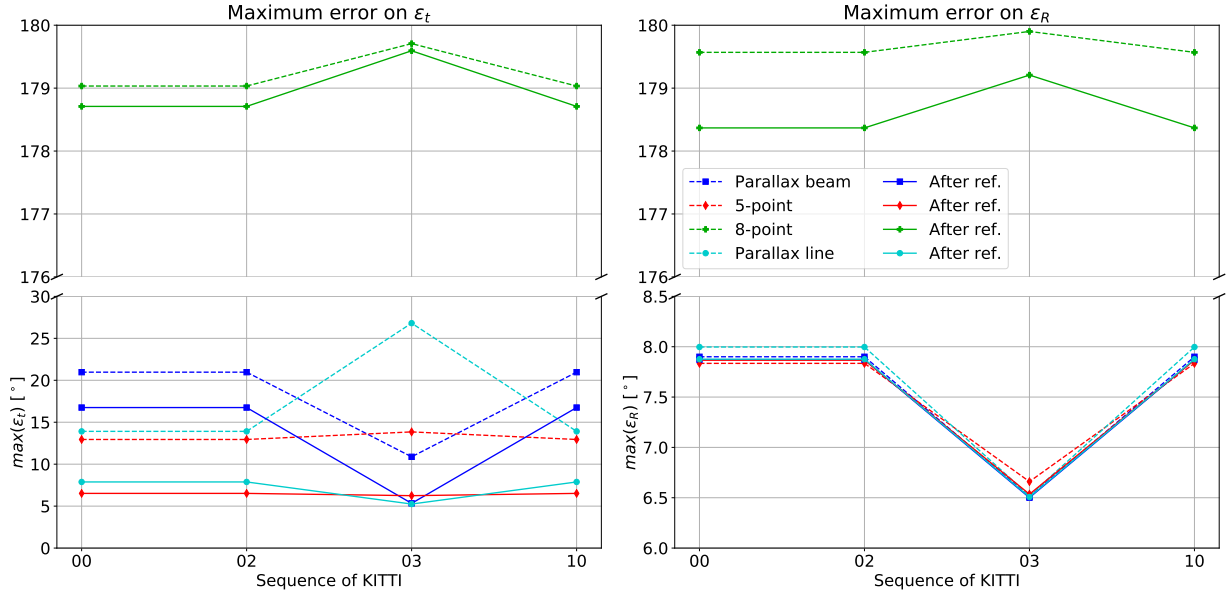


Fig 8 Maximum error measured for the sequences of KITTI. The values are taken from the maximum error columns of Table 2.

only a small part of the wall is visible.

For each distance, we run 500 motions. The data are corrupted to include 20 % of outliers and Gaussian noise is added on the feature points location.

The statistics on ϵ_R and ϵ_t are given in Table 3. When the camera is close to the wall, the parallax beam and the 5-point algorithm exhibit the best performances of the four methods. There is a slight advantage for the parallax beam method but the results of the 5-point would be usable in a real system. The parallax line and the 8-point algorithm fail to estimate the translation component correctly, whereas the rotation is estimated with a precision which is similar to the other methods. When the planar degeneracy is not strong and the planes are equally represented, all of the methods exhibit similar overall performance. However, the parallax line fails sometimes and falls quite far from the true translation. This behavior is not surprising as the method is usually said to be in trouble, when there are several well represented planes in the image. The interest of the parallax beam emerges as the camera is placed further from the wall and the planar degeneracy gets stronger. In the $d = 10.0$ m, the parallax beam and the parallax line have the best performance, with a slight advantage for the parallax beam. Even though the 5-point is providing a correct estimation most of the time, it sometimes fails as shown by the maximum error on the translation. At the furthest from the wall, $d = 15.0$ m, shown in Fig. 10, the parallax beam clearly detaches from the other methods and shows to be the most robust. The difference of performance for the 5-point between the $d = 2.5$ m and the $d = 15.0$ m cases is explained in Ref. 6, the algorithm is essentially unaffected by planar degeneracy if the baseline created by the two views is perpendicular to the plane, which is the case in the $d = 2.5$ m. The slightly better performance of the parallax beam in the $d = 2.5$ m compared to the $d = 15.0$ m can be explained by the distance of the off-plane points to the camera. When the wall is closed and only few ground points are visible, the ground points are also close to the camera and therefore have a relatively bigger apparent motion than in the $d = 15.0$ m case. In this case the off-plane points are the furthest from the camera and using

the only the parallax information they might appear as closer to the homographic plane. We could say that the $d = 2.5$ m is better conditioned than the $d = 15.0$ m.

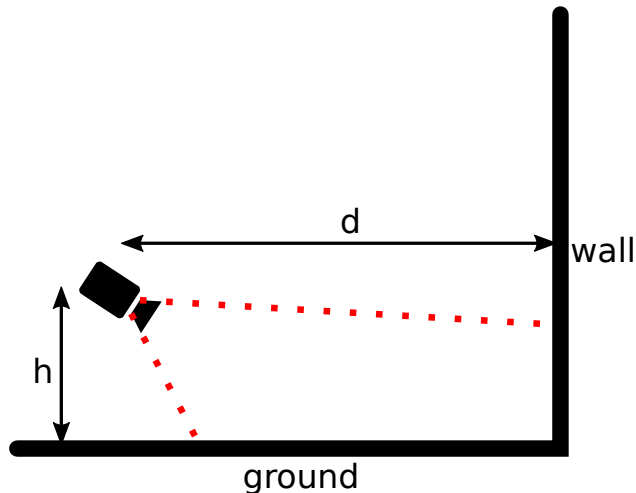


Fig 9 Synthetic scene and camera pose for generating synthetic image features with a strong planar degeneracy; *ref.* stands for refinement.

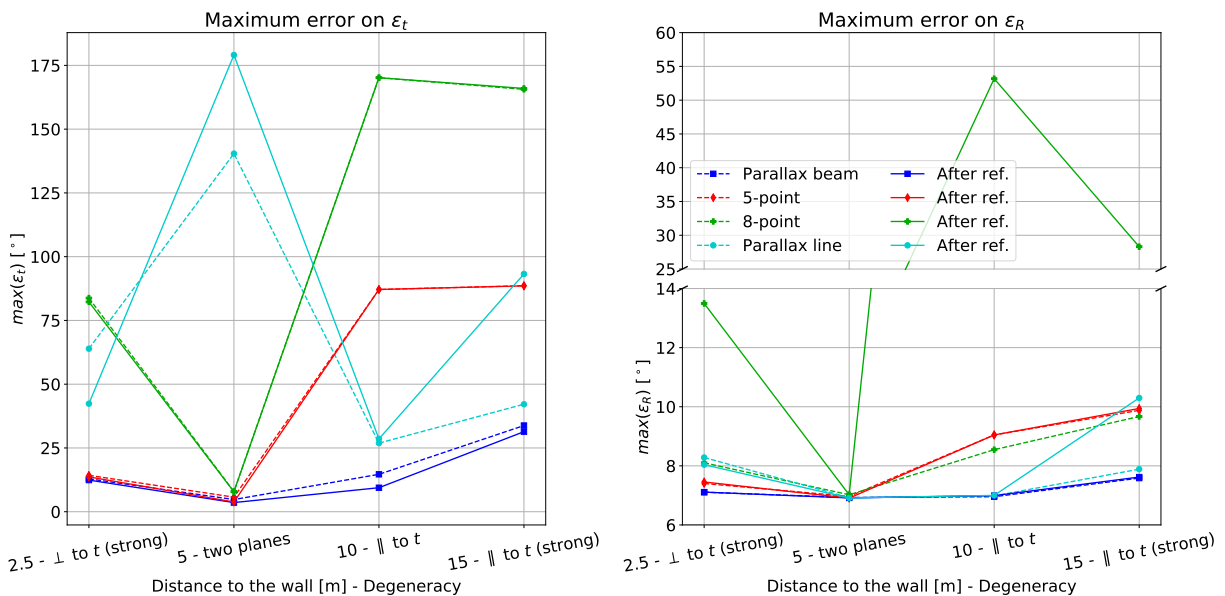


Fig 10 Maximum error measured for the tested planar degeneracies. The values are taken from the maximum error columns of Table 3. \perp stands for perpendicular, i.e. \perp to t means that the plane responsible for the degeneracy is perpendicular to the translation vector of the camera. \parallel stands for parallel; *ref.* stands for refinement.

4.3 Summary of the results

The 5-point algorithm, the 8-point algorithm, the parallax line method and our proposed parallax beam method were tested on the KITTI vision benchmark and on synthetic data with strong planar degeneracies. In our results, the 8-point algorithm appeared to be the least stable algorithm. The

Table 3 Statistics on ϵ_R and ϵ_t when estimating the camera motion on synthetic data. The values are given in degrees.

Method	mean(ϵ_t)	std(ϵ_t)	max(ϵ_t)	mean(ϵ_R)	std(ϵ_R)	max(ϵ_R)
Distance $d = 2.5$ m						
Before Refinement						
Beam	4.51	2.12	12.9	1.87	2.0	7.11
5-point	6.18	2.38	14.28	1.97	2.02	7.4
8-point	11.48	10.15	83.76	2.19	1.99	8.1
Line	14.52	9.21	63.92	2.45	1.97	8.28
After Refinement						
Beam	3.81	2.2	12.4	1.84	2.01	7.11
5-point	3.91	2.27	13.65	1.89	2.07	7.45
8-point	11.41	9.87	82.27	2.28	2.09	13.5
Line	8.26	7.33	42.36	2.16	2.07	8.04
Distance $d = 5.0$ m						
Before Refinement						
Beam	1.3	1.03	4.77	1.77	2.05	6.93
5-point	1.52	1.08	5.71	1.77	2.05	6.96
8-point	2.2	1.3	7.91	1.77	2.05	7.02
Line	1.72	6.31	140.42	1.77	2.05	6.92
After Refinement						
Beam	0.91	1.02	3.59	1.76	2.05	6.92
5-point	0.99	1.02	3.81	1.76	2.05	6.91
8-point	2.2	1.3	7.82	1.78	2.05	7.02
Line	1.27	8.03	179.07	1.76	2.05	6.91
Distance $d = 10.0$ m						
Before Refinement						
Beam	3.23	1.66	14.66	1.8	2.04	6.96
5-point	1.43	3.97	87.13	1.77	2.07	9.05
8-point	32.42	47.26	170.17	2.39	2.12	8.55
Line	2.36	1.82	26.9	1.79	2.06	6.98
After Refinement						
Beam	1.7	1.47	9.41	1.78	2.05	6.99
5-point	1.07	3.99	87.13	1.77	2.07	9.05
8-point	31.72	47.01	170.17	3.7	6.15	53.18
Line	1.2	1.71	28.55	1.77	2.06	7.0
Distance $d = 15.0$ m						
Before Refinement						
Beam	8.92	4.92	33.79	1.93	1.97	7.59
5-point	5.69	19.14	88.6	2.07	2.4	10.16
8-point	38.8	45.4	165.6	2.63	2.1	8.76
Line	21.81	7.2	42.14	2.46	1.99	7.96
After Refinement						
Beam	2.7	4.1	31.38	1.8	2.02	6.92
5-point	5.4	19.21	88.56	2.07	2.4	10.16
8-point	38.53	45.7	165.96	3.64	4.66	30.95
Line	16.47	21.44	93.22	2.52	2.44	10.75

parallax line as given in Alg. 1 appeared to perform correctly most of the time, but we were able to verify one recurring critic of the literature saying that it might not perform correctly in the presence of several well distributed planes. It also appeared not to be very robust, when the planar degeneracy is strong. In that case, the 5-point algorithm performed better, especially if the camera is going towards the dominant plane of the scene. However, the 5-point algorithm proved to fail,

when the planar degeneracy occurs otherwise. The parallax beam appeared to work correctly in generic scenes, even though a bit behind the 5-point algorithm but still usable. In these scenes, the constraint of having a planar structure for the homography appeared not to be important as it is often a purely mathematical plane that is used. When the planar degeneracy is strong, the parallax beam surpasses the other methods.

To conclude the comparison of the four methods, we propose trajectories made using only two-view geometry to illustrate once more the interest of the parallax beam, they are presented in Fig. 11. The moving platform starts in a car park and turn into an underground tunnel. The parallax beam is the only one estimating a correct trajectory. The 8-point is the worst of the four, the 5-point fails few times when the scene is very degenerated. The parallax line fails also, when the scene is very degenerated because it has too little parallax to exploit.



Fig 11 Trajectories obtained by using two-view geometry to illustrate the behavior of the four methods on a real image sequence.

5 Conclusion

We proposed the parallax beam to estimate the camera motion in two-view geometry. The method is designed to cope with the drawbacks of the standard plane-and-parallax approach. We have shown, that the parallax beam handles well scenes with strong parallax degeneracy and by extension when little parallax is present. In addition, it did not exhibit the instability the standard approach in the presence of equally distributed planes. It has even shown acceptable performances in general scenes (from KITTI), where planes are not dominant. Compared to most of the existing methods based on the homography model, the parallax beam only requires to estimate one and to have few off-plane points, which is less limiting than estimating two homographies.

With the parallax beam method we are confident that we can initialize our coming bundle adjustment processing in all the situations we encounter. This will allow the bundle adjustment process to be initialized whenever our system has to perform autonomously.

References

- 1 D. Monnin, A. L. Schneider, and E. Bieber, “Detecting suspicious objects along frequently used itineraries,” in *Security+ Defence*, 78340F–78340F, International Society for Optics and Photonics (2010).
- 2 E. Royer, M. Lhuillier, M. Dhome, *et al.*, “Monocular Vision for Mobile Robot Localization and Autonomous Navigation,” *International Journal of Computer Vision* **74**(3), 237–260 (2007).
- 3 C. Forster, M. Pizzoli, and D. Scaramuzza, “SVO: Fast semi-direct monocular visual odometry,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 15–22 (2014).
- 4 R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “ORB-SLAM: a Versatile and Accurate Monocular SLAM System,” *IEEE Transactions on Robotics* **31**, 1147–1163 (2015). arXiv: 1502.00956.
- 5 J. Engel, V. Koltun, and D. Cremers, “Direct Sparse Odometry,” *arXiv:1607.02565 [cs]* (2016). arXiv: 1607.02565.
- 6 D. Nistér, “An efficient solution to the five-point relative pose problem,” *IEEE transactions on pattern analysis and machine intelligence* **26**(6), 756–770 (2004).
- 7 R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, ISBN: 0521540518, second ed. (2004).
- 8 A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, (2012).
- 9 P. H. Torr, A. Zisserman, and S. J. Maybank, “Robust detection of degenerate configurations while estimating the fundamental matrix,” *Computer Vision and Image Understanding* **71**(3), 312–333 (1998).
- 10 L. Goshen and I. Shimshoni, “Balanced Exploration and Exploitation Model Search for Efficient Epipolar Geometry Estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**, 1230–1242 (2008).
- 11 D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” *International Journal of Computer Vision* **60**, 91–110 (2004).
- 12 C. Harris and M. Stephens, “A Combined Corner and Edge Detector,” 23.1–23.6, Alvey Vision Club (1988).
- 13 L. Gond, D. Monnin, and A. Schneider, “Optimized feature-detection for on-board vision-based surveillance,” in *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XVII*, **8357**, 83571N, International Society for Optics and Photonics (2012).
- 14 J.-M. Frahm and M. Pollefeys, “RANSAC for (quasi-) degenerate data (QDEGSAC),” in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, **1**, 453–460, IEEE (2006).
- 15 M. A. Fischler and R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM* **24**(6), 381–395 (1981).
- 16 P. Decker, D. Paulus, and T. Feldmann, “Dealing with degeneracy in essential matrix estimation,” in *Image Processing, 2008. ICIIP 2008. 15th IEEE International Conference on*, 1964–1967, IEEE (2008).

- 17 O. Chum, T. Werner, and J. Matas, "Two-view geometry estimation unaffected by a dominant plane," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, **1**, 772–779, IEEE (2005).
- 18 E. Vincent and R. Laganire, "Detecting planar homographies in an image pair," in *Image and Signal Processing and Analysis, 2001. ISPA 2001. Proceedings of the 2nd International Symposium on*, 182–187, IEEE (2001).
- 19 Y. Zhou, L. Kneip, and H. Li, "A Revisit of Methods for Determining the Fundamental Matrix with Planes," in *Digital Image Computing: Techniques and Applications (DICTA), 2015 International Conference on*, 1–7, IEEE (2015).
- 20 Q. T. Luong and O. D. Faugeras, "Determining the fundamental matrix with planes: instability and new algorithms," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 489–494 (1993).
- 21 Z. L. Szpak, W. Chojnacki, and A. v. d. Hengel, "Robust multiple homography estimation: An ill-solved problem," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2132–2141 (2015).
- 22 D. Sinclair, H. Christensen, and C. Rothwell, "Using the relation between a plane projectivity and the fundamental matrix," in *Proc. SCIA*, 181–188 (1995).
- 23 R. Szeliski and P. H. Torr, "Geometrically constrained structure from motion: Points on planes," in *European Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, 171–186, Springer (1998).
- 24 D. Monnin, E. Bieber, G. Schmitt, *et al.*, "An effective rigidity constraint for improving RANSAC in homography estimation," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, 203–214, Springer (2010).
- 25 Y. Ma, S. Soatto, J. Kosecka, *et al.*, *An invitation to 3-d vision: from images to geometric models*, vol. 26, Springer Science & Business Media (2012).
- 26 E. Malis and M. Vargas, "Deeper understanding of the homography decomposition for vision-based control," Research Report RR-6303, INRIA (2007).
- 27 Z. Zhang, "Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting," Research Report RR-2676, INRIA (1995).
- 28 J. Schmidt and H. Niemann, "Using Quaternions for Parametrizing 3-D Rotations in Unconstrained Nonlinear Optimization," in *Proceedings of the Vision Modeling and Visualization Conference 2001, VMV '01*, 399–406, Aka GmbH (2001).
- 29 G. Terzakis, P. Culverhouse, G. Bugmann, *et al.*, "On quaternion based parameterization of orientation in computer vision and robotics," *Journal of Engineering Science and Technology Review (JESTR)* **7**(1), 82–93 (2014).
- 30 M. E. Fathy, A. S. Hussein, and M. F. Tolba, "Fundamental matrix estimation: A study of error criteria," *Pattern Recognition Letters* **32**(2), 383 – 391 (2011).
- 31 K. MacTavish and T. D. Barfoot, "At all Costs: A Comparison of Robust Cost Functions for Camera Correspondence Outliers," in *2015 12th Conference on Computer and Robot Vision*, 62–69 (2015).
- 32 M. L. A. Lourakis and A. A. Argyros, "Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment?," in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, 2*, 1526–1531 Vol. 2 (2005).

- 33 C. Voglis and I. E. Lagaris, “A Rectangular Trust Region Dogleg Approach for Unconstrained and Bound Constrained Nonlinear Optimization,” 7 (2004).
- 34 R. I. Hartley, “In defence of the 8-point algorithm,” in *Proceedings of IEEE International Conference on Computer Vision*, 1064–1070 (1995).

Martin Rebert was born in Colmar, France, in 1991. He received his Master’s Degree in engineering from Télécom Bretagne in 2015. He is now preparing his Ph.D. with the collaboration of the IRIMAS laboratory of the University of Haute Alsace and the French-German Research Institute of Saint-Louis. His work is focused on computer vision and autonomous navigation.

Stéphane Bazeille is an Assistant Professor at Université Haute-Alsace in France. He received his Master’s Degree in mathematics and computer science from the University of La Rochelle in 2005 and his Ph.D. degree in the field of robots perception from the University of Western Britany in 2008. He is the author of more than 30 scientific papers in international journal and conferences. His current research interests include perception systems, computer vision, artificial intelligence and robotics.

David Monnin is a researcher in the field of information processing and computer vision at the French-German Research Institute of Saint-Louis since 1997. He received his Ph.D. in signal, image and speech processing from the University of Grenoble in 2003. His work focuses on image registration, change detection and vision-based navigation.

Christophe Cudel was born at Troyes, France, in 1967. He received his Ph.D. from the University of Reims Champagne Ardenne in 1995. He joined the University of Haute Alsace and the laboratory MIPS in 1997, first as assistant professor and now as professor. His researches are focused on computer vision, eye-tracking, light-field and multi-view cameras. He teaches computer vision for industrial control, robotic and signal processing.