



Camera Localization by Single View Query Using One Circular Target

Damien Mariyanayagam, Pierre Gurdjos, Sylvie Chambon, Vincent Charvillat

► To cite this version:

Damien Mariyanayagam, Pierre Gurdjos, Sylvie Chambon, Vincent Charvillat. Camera Localization by Single View Query Using One Circular Target. Michael Felsberg; Per-Erik Forssén; Ida-Maria Sintorn; Jonas Unger. Image Analysis, 11482, Springer, pp.275-286, 2019, Lecture Notes in Computer Science, 978-3-030-20204-0. 10.1007/978-3-030-20205-7_23 . hal-02353730

HAL Id: hal-02353730

<https://hal.science/hal-02353730>

Submitted on 7 Nov 2019

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.

Camera localization by single view query using one circular target

Damien Mariyanayagam, Pierre Gurdjos, Sylvie Chambon, Vincent Charvillat

IRIT, Toulouse, France

Abstract. We are concerned with the problem of computing the pose of a (so-called query) camera, from the sole contour of a circular marker in a single view, assuming that we have at one's disposal a set of reference views seeing circular markers, from which the full camera pose and calibration have been precisely estimated. In the calibrated case, regarding the query image alone, there is also a twofold ambiguity in the pose as well as an unknown rotation in the supporting plane of the marker that cannot be fixed. The key idea of this paper is to show that with the additional information of one pair of matched points in both query and reference views, an exact solution for the query pose can be obtained. In order to answer the question whether a given reference view corresponds to the query view, another pair of matched points in both views is then necessary. With multiple matched point pairs, it is possible to deploy a RANSAC-based scheme to assess accurately the best configuration while maintaining robustness to mismatched point pairs. This method shows some promising results especially when the set of point correspondences contains a lot of outliers (erroneous matches).

1 Introduction

The problem of estimating the pose of a calibrated camera in a single view has been widely studied in the computer vision literature e.g., see in [1]. The general case consists in solving a so-called PnP (Perspective- n -Point) problem, using n correspondences of 2D-3D points. The “minimal” problem i.e., which requires the minimal amount of information, is known as $P3P$ and consists in recovering the pose of a calibrated camera from $n = 3$ correspondences with up to 16 real solutions. On the other hand, when using a pair of views, the relative pose of one calibrated camera to another calibrated one can be recovered by using only 2D-2D correspondences. The problem is then that of estimating the essential matrix which has 10 exact solutions given 5 points [2], where the decomposition of such a matrix delivers the pose parameters up to an unknown scale factor. Recently, it is worth noting that a hybrid approach proposes to both use 2D-2D and 2D-3D correspondences to combine the strengths of each method [3].

In a controlled environment, artificial markers can be laid out into the scene to offer reliable features which are easier to detect and to be matched with others than natural scene features. The most popular artificial markers are probably planar markers [4]. The mapping between a planar marker and its image is a 2D

projective transformation known as (world-to-image) homography and can be estimated from at least four world-to-image correspondences. The simplest planar marker is a square, and, consequently, square markers are commonly used [4, 5]. Once the camera is calibrated, the decomposition of the homography matrix allows to recover the pose of the marker relative to the camera [6]. Other well-known artificial markers, recently investigated, consist of planar circles [7–10]. In the calibrated case, the image of only one circle suffices to recover the camera pose because but there are two solutions for it. In [11], a theoretical framework has been proposed where the pose two-fold ambiguity was investigated. Moreover, the rotation around the normal to the circle (i.e., to its supporting plane) remains unknown. Additional features can be introduced on the circular marker to fix the rotation but it can be really tricky to detect them, in particular when the marker is far from the camera.

In this work, we are concerned with the problem of computing the pose of a circular marker, given a reference view when only its (elliptical) contour is detected in a so-called query view. To our knowledge, the issue of using multiple views of circles in combination with natural point 2D correspondences has been marginally studied. Kahl *et al.* proposed to estimate the fundamental matrix by using correspondences of 5 points and one conic, leading to 10 solutions [12]. Alvarez *et al.* studied the estimation of the world-to-image homography in sport scenarios by using only one circle with some additional information (e.g., like the central line and the central point or a touch line) [13]. Huang et al. proposed a general fusion frame of circles and points to estimate the pose. Starting from an initial guess computed from geometric constraints, they optimized the reprojection error by solving a non-linear least squares problem [14]. However in the case of only one circle in combination with only one additional point as initial guess, their proposed approach lacks of robustness by not considering outliers, which limits its practicability.

We state the problem in the following way. We assume to have at one’s disposal a (reference) view of a circular marker from which the full camera pose and calibration could be precisely estimated. Given a second (query) calibrated view in which only the circle is seen, our contribution is to estimate the pose of the second camera w.r.t. the same 3D frame as the first by using additional 2D-2D matches of natural 3D points between the reference and query views. Hence, the 2D arbitrary rotation on the supporting plane can be fixed. First, the pose of the circle is estimated in both images with the previous method published in [11]. It is important to note that any additional point constrains the remaining unknown rotation angle. Consequently, the key idea is to use matched points in a RANSAC-based approach to fix the unknown 2D rotation and, so, the essential matrix. As only one point correspondence is required to compute the essential matrix, we show that our proposed method can be robust to an important number of outliers (erroneous matches) with very few inliers (correct matches).

In the section 2, the problem and its formulation are introduced. Then, in the section 3, the two proposed approaches for solving this problem are described. Finally, results using both synthetic and real data are presented in section 4.

2 Problem statement

2.1 Context

Suppose a scenario where a collection of circular markers has been placed in some environment, as seen in Fig. 1. Assume that a set of reference views of all

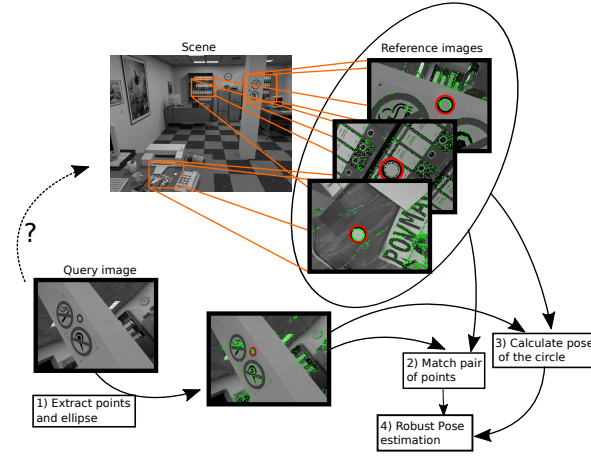


Fig. 1: Illustration of the pose estimation problem. The circle (in red) and points of interest (in green) are first detected and then matched with their counterparts in the reference views. These correspondences are exploited in the proposed approach for pose estimation.

these circular markers are given. Moreover, for each view, the local pose of the camera has been pre-computed, possibly using additional information. Indeed, a 3D frame can be attached to the target support plane such that this plane has equation $Z = 0$ and the circle of the target is centered at the origin with its radius equals 1.

Given a new view, so-called query view, the aim is to locate the camera associated with the query view i.e. to determine the camera pose, by finding within the dataset the “best” view in which the same marker is seen at the same 3D position. It is already known that using only one image of a circle provides an infinity of solutions with only one degree of freedom for the (calibrated) camera poses. This is due to the ambiguity in rotation around the circle axis which prevents to set a local coordinate system on the marker’s support plane. Moreover,

each reference view is associated with a unique camera pose w.r.t. the marker's support plane. Consequently, given a reference view and a query view, we get a similar infinity of solutions for the epipolar geometry (in terms of the fundamental or essential matrices) of the view pair. The goal of this paper is to show that with the additional information of one pair of matched points in both views, an exact solution for the query pose (and so for the fundamental/essential matrix) can be obtained. In order to answer the question whether a given reference view corresponds to the query view, another pair of matched points in both views is then necessary. And, with multiple matched point pairs, it is possible to deploy a RANSAC-based scheme to assess accurately the best configuration while maintaining robustness to mismatched point pairs.

2.2 Formulation of the problem

We consider a pinhole camera model whose intrinsic parameters are known, or, equivalently, without any loss of generality, whose calibration matrix [1] K satisfies $K = I$. For each query view, the ellipse, corresponding to the projection of the circle, is first detected. The two solutions for the pose of the circle relative to the query camera are then estimated using the method provided in [11]. The steps for one solution consist in: (i) recovering the vanishing line of the support plane; (ii) calculating the homography using a parameterization based on the image of the circle and the vanishing line; (iii) decomposing the homography to find the pose in the camera frame. The resulting pose consists of an orientation of the supporting plane and the location of the origin of this plane, fixed to be the circle center. The orientation is given by a matrix $R_q \in SO(3)$ whose first two columns can be randomly fixed and third column is the vector of the normal to this plane. Whereas the location of the origin is given by the Cartesian vector $t_q \in \mathbb{R}^3$ of the circle center.

2.3 Parametrization with the unknown rotation

Denote by $P = [R \mid t]$ the projection matrix of any camera where the rotation matrix $R \in SO(3)$ and the translation vector $t \in \mathbb{R}^3$ define the camera pose w.r.t. the object frame. On the one hand, the projection matrix of a reference camera $P_{\text{ref}} = [R_{\text{ref}} \mid t_{\text{ref}}]$ is completely known. On the other hand, the projection matrix of a query camera seeing a circle can be determined up to an unknown rotation around the circle axis. This projection writes $P_q = [R_q R(\theta) \mid t_q]$ where (R_q, t_q) has been computed from the image of a circle and the 2D rotation parameterized by an unknown angle θ :

$$R(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Suppose that a point is seen on both the reference image and the query image, with their respective augmented vectors $u_1 = [u_1, v_1, 1]^T$ and $u_2 = [u_2, v_2, 1]^T$.

If we assume that these two image points are in correspondence i.e., they are the images of the same 3D point with augmented Cartesian vector $\mathbf{X} = [X, Y, Z, 1]^\top$, then the unknown rotation $\mathbf{R}(\theta)$ can be solved using the system of two equations $\mathbf{u}_1 \sim [\mathbf{R}_{\text{ref}} \mid \mathbf{t}_{\text{ref}}]\mathbf{X}$ and $\mathbf{u}_2 \sim [\mathbf{R}_q\mathbf{R}(\theta) \mid \mathbf{t}_q]\mathbf{X}$. To get rid of the 3D point $\mathbf{X}$, this hypothesis is used: the rotation $\mathbf{R}(\theta)$ of the second camera (corresponding to query image) has to be estimated so that the back-projection rays (obtained by back-projecting the matched image points $\mathbf{u}_1$ and $\mathbf{u}_2$) intersect in 3D space. For a camera with projection matrix $\mathbf{P}$, the 3D line corresponding to the back projection of an image point $\mathbf{u} = [u, v, 1]^\top$ can be expressed in matrix form using its Plücker representation as:

$$\mathbf{L} = \mathbf{P}^\top [\mathbf{u}]_\times \mathbf{P}^{-1} \text{ and } \mathbf{L} = \begin{pmatrix} [\mathbf{m}]_\times \mathbf{d} \\ -\mathbf{d}^\top 0 \end{pmatrix} \quad (1)$$

with $\mathbf{d}, \mathbf{m} \in \mathbb{R}^3$ where $(\mathbf{d}, \mathbf{m})$ is the so-called Plücker coordinates of the line.

Call the reference camera the camera number 1: the Plücker coordinates $(\mathbf{d}_1, \mathbf{m}_1)$ of the line obtained by back-projecting a reference point $\mathbf{u}_1$ are

$$(\mathbf{d}_1, \mathbf{m}_1) = ((\mathbf{R}_{\text{ref}})^\top \mathbf{u}_1, \mathbf{c}_1 \times \mathbf{d}_1)$$

where $\mathbf{c}_1 = -(\mathbf{R}_{\text{ref}})^\top \mathbf{t}_{\text{ref}}$. Call the query camera the camera number 2: we have almost the same equation for the Plücker coordinates $(\mathbf{d}_2, \mathbf{m}_2)$ of the back-projection of a reference point $\mathbf{u}_2$ with the difference that there is an unknown rotation matrix $\mathbf{R}(\theta)$

$$(\mathbf{d}_2, \mathbf{m}_2) = (\mathbf{R}(\theta)^\top \mathbf{d}'_2, \mathbf{R}(\theta)^\top \mathbf{m}'_2)$$

where $\mathbf{d}'_2 = (\mathbf{R}_q)^\top \mathbf{u}_2$, $\mathbf{m}'_2 = \mathbf{c}'_2 \times \mathbf{d}'_2$ and $\mathbf{c}'_2 = -(\mathbf{R}_q)^\top \mathbf{t}_q$.

Now, it is necessary to determine θ . We seek a value of θ such that the distances between the two 3D lines obtained by back-projections from the reference and query cameras is zero i.e., such that the two lines intersect. For that, the so-called reciprocal product of the Plücker coordinates [15] is introduced. Any two 3D lines have a common perpendicular which intersects both at right angles at two points (one on each line) called the feet of the common perpendicular. The distance between the two lines equals the distance between these two feet which are the closest points in space on the respective lines. Its expression involves the *reciprocal product* of the Plücker coordinates of the two lines which is defined as

$$(\mathbf{d}_1, \mathbf{m}_1) * (\mathbf{d}_2, \mathbf{m}_2) = \mathbf{d}_1^\top \mathbf{R}(\theta) \mathbf{m}'_2 + \mathbf{d}_2^\top \mathbf{R}(\theta) \mathbf{m}_1 \quad (2)$$

and equals zero when the two 3D lines intersect.

It is worth noting that, by varying θ , and thus rotating the back-projection ray of the second camera (query) around the axis defined by the normal of the

¹ $[\mathbf{a}]_\times = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ a_2 & a_1 & 0 \end{pmatrix}$ is the order-3 skew-symmetric matrix of $\mathbf{a} \in \mathbb{R}^3$ such that $[\mathbf{a}]_\times \mathbf{b} = \mathbf{a} \times \mathbf{b}$, for all $\mathbf{b} \in \mathbb{R}^3$.

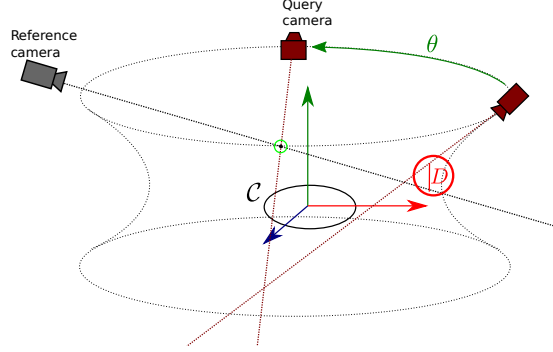


Fig. 2: Which angle θ makes the back-projection of a matched point from the query camera intercept that of the reference camera?

circle $\mathcal{C}$ and its center, we generate a hyperboloid, see Fig. 2. As the problem becomes that of finding the intersection of the hyperboloid of one sheet with a line, there are exactly zero, one or two real solutions. The ambiguity cannot be solved directly without some other considerations. As an example, in some theoretical cases, the intersections of the hyperboloid with the back-projection ray are separated with the principal plane whereas, in practical cases, both solutions are in front of the camera and most of the time in a small sphere centered around the marker. The correct solutions can be identified only if several points are involved. In this context, we propose two methods to solve the problem. First, several points are used to obtain a linear system of equations in cosine and sinus with the circular constraint. Second, the minimal case with only one correspondence can be considered, the equation is directly solved (two exact solutions are estimated) and a consensus approach using all the other correspondences are then used.

3 Pose estimation from one circle and several correspondences

We explained previously that, to estimate the full pose of the query camera relative to the reference camera, we need to determine a 2D rotation in the supporting plane of the circle. To that end, each point correspondence brings a new equation, depending on the rotation angle θ , in which the reciprocal product (2) vanishes. In this section, we will explain how to calculate this angle in a robust way using several correspondences of points. Here, only one reference image is considered but this step must be carried out for every candidate reference image. The correct reference image can then be picked up as that of providing the greatest number of correspondences (inliers in robust case).

3.1 Linear formulation with quadratic constrains

As already said, the revolution of a back-projection ray from a query point, around the axis defined by the normal of the circle and its center, forms a hyperboloid of one sheet in 3-space, see Fig. 2. Solving the rotation consists in finding its intersection with the back-projection ray of a corresponding reference point. Several cases are then possible, there are zero, one or two real solutions to this problem. It is easy to see that the equation obtained by vanishing (2) can be written as a linear combination of $x_1 = \cos \theta$ and $x_2 = \sin \theta$ with the constraint $x_1^2 + x_2^2 = 1$. In fact, it can be interpreted in 2D as the problem of intersecting a circle ($x_1^2 + x_2^2 = 1$) with a line ($ax_1 + bx_2 + c = 0$), where a, b, c are parameters only depending on the Plücker coordinates of the back-projection lines.

In this part, we want to study if several pairs of matched points can be used jointly to solve the problem of estimating the missing rotation. Instead of having one line to intersect a circle, we will have a bunch of lines intersecting each others and the circle, see Fig. 3. It can be interesting to normalize the equation

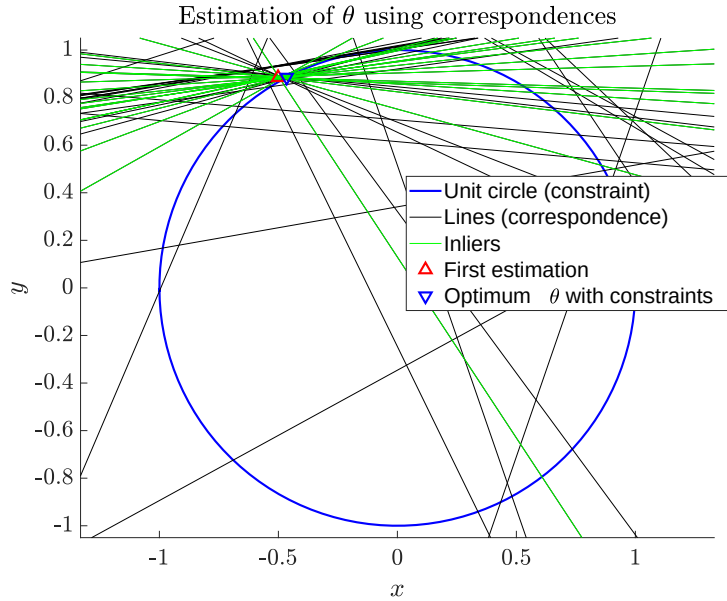


Fig. 3: Example of lines, solution of the equation 2 in x_1 and x_2 .

vanishing (2) before using them with a solver. Indeed, if we normalize the Plücker coordinates of the 3D lines then the reciprocal product (2) corresponds to the moment between the lines. We can write [15]

$$\mathbf{d}_1 \mathbf{R}(\theta)^T \mathbf{m}'_2 + \mathbf{d}'_2 \mathbf{R}(\theta) \mathbf{m}_1 = D \sin \gamma \quad (3)$$

where γ is the angle between the two lines and D is their relative distance. Thus, minimizing the set of equations, with norm L_1 for robustness to erroneous data, is then minimizing the sum of the distances between the back-projection rays of image features.

First, the solution is initialized using a linear least squares solution without the circular constraint. We can see, in Fig. 3, that the linear least squares solution, represented by a red triangle point, does not lie necessary on the circle but tries to fit a maximum of lines. Then, we use a non linear squares solver ensuring the circular constraint, the blue triangle on Fig. 3 now corresponds to this solution.

3.2 Robust solution using minimal number of correspondences

The first proposition implies to simultaneously use all the correspondences to calculate a solution. However, in practice there are many erroneous correspondences of points. We have seen that only one correspondence is sufficient to calculate an angle. Indeed, we can directly solve the cosine equation which can lead up to 2 solutions for the angle. We propose to use a random sample consensus approach, MSAC (M-estimator SAmple and Consensus) [16], to deal with cases where many points are incorrectly matched. The method consists in selecting the minimal number of correspondences (only one in our case) to estimate the angle and, then, to evaluate among all the other points, the inliers which fit the pose model (estimated with this correspondence). The process is then repeated iteratively to minimize a fitting function while ensuring that enough points are selected as inliers. The inlier points can then be used at the end of the process to estimate the angle more precisely.

The equation vanishing (2) can be solved directly but this direct estimation can lead from 0 to 2 real solutions for the angle. If there is at least one solution for the angle, the hypothesis is then evaluated with the others points. Once an angle has been estimated we can evaluate which points validate the model. As the real location of the 3D point is unknown, we propose to evaluate the distance in pixel of a point to its epipolar line in the query image. The fundamental matrix can be constructed as follows using the projection matrix of the reference camera P_{ref} and the projection matrix of the query camera P_q :

$$\begin{aligned} \mathbf{O}_{\text{ref}} &\sim P_q [\mathbf{t}_{\text{ref}} \ 1]^\top \\ \mathbf{F} &\sim [\mathbf{O}_{\text{ref}}] \times P_q P_{\text{ref}}^+ \end{aligned}$$

where P_{ref}^+ is the pseudo-inverse of the reference camera projection matrix (see [17, p. 244]). We can note that $\mathbf{O}_{\text{ref}}$ is the coordinates of the center of the reference camera in the query camera frame. Then, to calculate the distance of a point to the model, we first calculate its epipolar line in the query image, $\mathbf{l}_{\text{ref}} \sim \mathbf{F} \mathbf{u}_1$ and calculate its distance to this line. Consequently, the greater the distance, the more likely it is that the point is an outlier. To distinguish outliers from inliers, we can apply a threshold to this distance².

² Experimentally, this threshold has been set to 5 pixels.

4 Results

We have evaluated our method on both synthetic and real images. We compute the accuracy of the relative pose of the camera related to the marker and assess the correctness of the reference image identification, i.e. how the approach is able to identify the corresponding reference image.

4.1 Test description

Image dataset. For the synthetic tests, we used the virtual office room from the ICL-NUIM dataset³ to generate a virtual scene by adding circular markers on different elements of the scene, and also some synthetic images from different points of views using PovRay. More precisely, 8 circular markers have been projected on the scene, at different locations. We have first generated one reference image for each marker. Then, 25 new images have been generated from 25 different points of view, far enough from the marker so it can not be recognized by its code. These 25 views are used as query images, i.e. images where we want to recognize the marker, based on the detection of the contour of the ellipse of the marker and the points of interest detected around this marker.

For the real images we have placed 14 markers in an office room. Our “ground truth” for the camera poses has been computed using a chessboard around each marker. Finally, 44 images have been taken using a OnePlus 5 smartphone camera with at least 2 different views for each marker. Some examples of the synthetic and real images used in our test can be seen in Fig. 4.



Fig. 4: Samples of the images used in our test: real at left, synthetic at right

Points of interest. They have been detected and matched for each pair of reference/query images by using the AliceVision⁴ library which is based on SIFT for detection and for matching. We have limited the number of points to 1000 per image. This restriction assures that the best points are kept and that they cover an equally distributed grid on the image. Points have been matched and

³ <https://www.doc.ic.ac.uk/~ahanda/VaFRIC/iclnuim.html>

⁴ <https://alicevision.github.io/>

filtered with Lowe ratio criteria [18] set at 0.8. Ellipses in both synthetic and real images are detected by following these steps: first ellipse points are detected using a Sobel filter, second, these points are connected using propagation of an elliptical arc and then the parameters of the ellipse are estimated with [19].

Criteria. For each query image, using the proposed method, we compute the identification of the marker and the pose of the camera relative to the plane. Regarding the pose, what is evaluated is the angle, θ , which is the sole unknown dof of the 2D rotation in the support plane of the circle. Additionally, the errors on the relative orientation and the position of the camera have been computed. The first criterion evaluates how the addition of new point correspondences allows the calculation of the missing dof (rotation). The last criteria evaluate the global quality of the pose estimation given by both the circle image and points.

4.2 Results on the pose estimation

The table 1 displays the results for the recognition rate of the marker and the Fig. 5 shows the error on the estimated pose. We distinguish the two proposed P1CnP (Perspective with 1 Circle and n Points) methods: “P1CnP least-square” refers to that described in section 3.1 which solves the problem directly with all correspondences and “P1CnP MSAC” to that introduced in section 3.2 which solves the problem using a RANSAC-based scheme.

Table 1: Percentage of recognition, we distinguish real and synthetic data.

Method	True Positive	
	Real	Synthetic
P1CnP least-square	70.45 %	88 %
P1CnP MSAC	84.09 %	84 %

A first general comment is that the results on the synthetic images are worst than on real images. This can be explained because there are more untextured regions or repetitive textures in our synthetic images than in real images, so, we have obtained a lower number of correct matches and, therefore, the pose estimation contains more errors. On Fig. 5, we can see that the method using only 1 point with the random consensus is more accurate than the direct least-square method with the constraint. The first line of figures are the most interesting, we can see that additional correspondences of points give an accurate estimation of the rotation around the normal of the plane, the median is below 2 degree for real images. The other figures show the results on the complete method to estimate the pose: the orientation and the position. The results are quite good for the pose estimation for real images especially if we consider the apparent small size of the marker in comparison to the image as we can see in the image

samples (the marker is at the center of the chessboard pattern). Moreover the markers are generally well identified using the reference images as seen in table 1. However there are some cases on both synthetic and real images where the identification fails. The identification only relies on the number of inlier correspondences. So when some markers are close to each other and share common points of interest, the angle may not be discriminant enough to distinguish the correct reference. Additionally, we think that the initial pose obtained can be easily improved if we suppose that we can use this initial estimation in a further optimization process using the estimated coordinates of the inliers points. Concerning the execution time for the pose estimation, the MSAC algorithm converges rather quickly. For instance with a request image with 1000 correspondences to a reference image, the algorithm takes about 30 ms using Matlab and without any particular optimization.

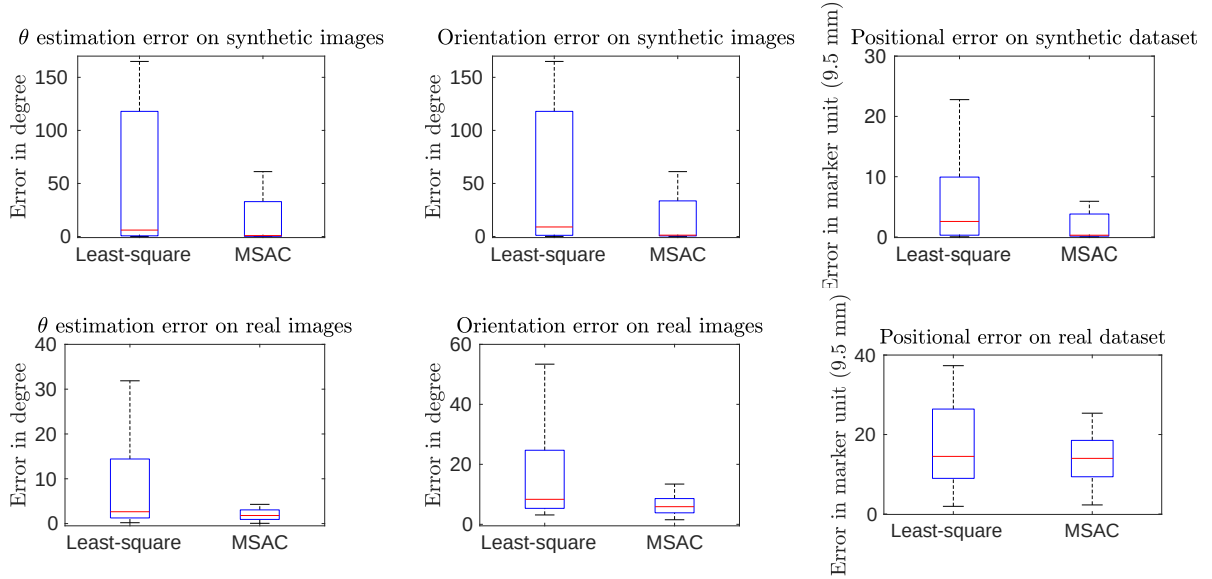


Fig. 5: Results on synthetic (top) and real (bottom) images: Error on the rotation $R(\theta)$ (left), orientation error (middle) and position error (right)

5 Conclusion

In this paper, using circular markers to estimate the pose, we have introduced a new method that is able to produce an accurate solution, assuming that only the contour of this marker can be reliably detected. The originality of the approach is to combine the detection of the contour of the marker with the use of several points of interest in correspondence. We have shown that this method can provide a good estimation even when there are many erroneous matches.

References

1. Szeliski, R.: Computer vision: algorithms and applications. Springer Science & Business Media (2010)
2. Nistér, D.: An efficient solution to the five-point relative pose problem. *PAMI* **26** (2004)
3. Camposco, F., Cohen, A., Pollefeys, M., Sattler, T.: Hybrid camera pose estimation. In: *CVPR*. (2018)
4. Fiala, M.: Designing highly reliable fiducial markers. *PAMI* **32** (2010)
5. Wagner, D., Schmalstieg, D.: Artoolkitplus for pose tracking on mobile devices. In: *Computer Vision Winter Workshop*. (2007)
6. Sturm, P.: Algorithms for plane-based pose estimation. In: *CVPR*. Volume 1., IEEE (2000)
7. Wu, H., Chen, Q., Wada, T.: Conic-based algorithm for visual line estimation from one image. In: *Conference on Automatic Face and Gesture Recognition*, IEEE (2004)
8. Gurdjos, P., Sturm, P., Wu, Y.: Euclidean structure from $n \geq 2$ parallel circles: theory and algorithms. In: *ECCV*, Springer (2006) 238–252
9. Calvet, L., Gurdjos, P., Griwodz, C., Gasparini, S.: Detection and accurate localization of circular fiducials under highly challenging conditions. In: *CVPR*. (2016)
10. Kim, J.S., Gurdjos, P., Kweon, I.S.: Geometric and algebraic constraints of projected concentric circles and their applications to camera calibration. *PAMI* **27** (2005)
11. Mariyanayagam, D., Gurdjos, P., Chambon, S., Brunet, F., Charvillat, V.: Pose estimation of a single circle using default intrinsic calibration. In: *ACCV*. (2018)
12. Kahl, F., Heyden, A.: Using conic correspondences in two images to estimate the epipolar geometry. In: *ICCV*, IEEE (1998)
13. Alvarez, L., Caselles, V.: Homography estimation using one ellipse correspondence and minimal additional information. In: *ICIP*, IEEE (2014)
14. Huang, B., Sun, Y., Zeng, Q.: General fusion frame of circles and points in vision pose estimation. *Optik* **154** (2018)
15. Mason, M.T.: Mechanics of robotic manipulation. MIT press (2001)
16. Torr, P.H., Zisserman, A.: Mlesac: A new robust estimator with application to estimating image geometry. *CVIU* **78** (2000)
17. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)
18. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *IJCV* **60** (2004)
19. Szpak, Z.L., Chojnacki, W., van den Hengel, A.: Guaranteed ellipse fitting with a confidence region and an uncertainty measure for centre, axes, and orientation. *JMIV* **52** (2015)