Machine Vision and Applications (2013) 24:1295–1310
DOI 10.1007/s00138-012-0469-6

ORIGINAL PAPER

Pi-Tag: a fast image-space marker design based on projective invariants

Filippo Bergamasco · Andrea Albarelli · Andrea Torsello

Received: 24 May 2012 / Revised: 1 November 2012 / Accepted: 14 November 2012 / Published online: 25 December 2012

Abstract Visual marker systems have become an ubiquitous tool to supply a reference frame onto otherwise uncontrolled scenes. Throughout the last decades, a wide range of different approaches have emerged, each with different strengths and limitations. Some tags are optimized to reach a high accuracy in the recovered camera pose, others are based on designs that aim to maximizing the detection speed or minimizing the effect of occlusion on the detection process. Most of them, however, employ a two-step procedure where an initial homography estimation is used to translate the marker from the image plane to an orthonormal world, where it is validated and recognized. In this paper, we present a general purpose fiducial marker system that performs both steps directly in image-space. Specifically, by exploiting projective invariants such as collinearity and cross-ratios, we introduce a detection and recognition algorithm that is fast, accurate and moderately robust to occlusion. The overall performance of the system is evaluated in an extensive experimental section, where a comparison with a well-known baseline technique is presented. Additionally, several real-world applications are proposed, ranging from camera calibration to projector-based augmented reality.

Keywords Fiducial markers · Projective invariants · Augmented reality · Pose estimation · Camera calibration

F. Bergamasco · A. Albarelli (✉) · A. Torsello
Dipartimento di Scienze Ambientali, Informatica e Statistica,
Università Ca' Foscari Venezia, Venice, Italy
e-mail: albarelli@unive.it

F. Bergamasco
e-mail: fbergama@dsi.unive.it

A. Torsello
e-mail: torsello@dais.unive.it

1 Introduction

A visual marker is an artificial object consistent with a known model that is placed into a scene to supply a reference frame. Currently, such artefacts are unavoidable whenever a high level of precision and repeatability in image-based measurement is required, as in the case of vision-driven dimensional assessment task such as robot navigation and SLAM [5,8,36], motion capture [2,38], pose estimation [37,39], camera calibration [7,14] and of course in field of augmented reality [6,40].

While in some scenarios, approaches based on naturally occurring features have been shown to yield satisfactory results, they still suffer from shortcomings that severely limit their usability in uncontrolled environments. Specifically, the lack of a well-known model limits their use in pose estimation. In fact, while using techniques like bundle adjustment can recover part of the pose, the estimation can be only up to an unknown scale parameter; further, the accuracy of the estimation heavily depends on the correctness of localization and matching steps.

Moreover, the availability and distinctiveness of natural features are not guaranteed at all. Indeed the smooth surfaces found in most man-made objects can easily lead to scenes that are very poor in features.

Finally, photometric inconsistencies due to reflective or translucent materials severely affect the repeatability of the point descriptors, jeopardizing the correct matching of the detected points. For this reasons, it is not surprising that artificial fiducial tags continue to be widely used and are still an active research topic.

Markers are generally designed to be easily detected and recognized in images produced by a pinhole camera. In this sense, they make heavy use of the projective invariance properties of geometrical entities such as lines, planes and conics.

One of the earliest invariance used is probably the closure of the class of ellipses to projective transformations. This implies that ellipses (and thus circles) in any pose in the 3D world appear as ellipses in the image plane. This allows both for an easy detection and a quite straightforward rectification of the plane containing any circle.

With their seminal work, Gatrell et al. [10] propose to use a set of highly contrasted concentric circles and validate a candidate marker by analyzing the compatibility between the centroids of the detected ellipses. By alternating white and black circles, a few bits of information can be encoded in the marker itself. In the work proposed in [3], the concentric circle approach is enhanced by adding colors and multiple scales. Later, in [18] and [24], dedicated "data rings" are added to the marker design.

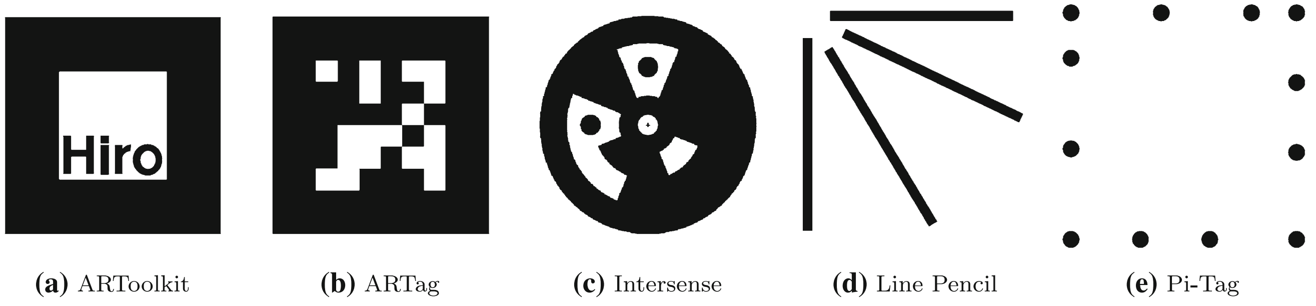
A set of four circles located at the corner of a square is adopted in [4]: in this case, an identification pattern is placed in the middle of the four dots to distinguish between different targets. This ability to recognize all the viewed markers is really important for complex scenes where more than a single fiducial is required; furthermore, the availability of a coding scheme allows for an additional validation step and thus lowers the number of false positives.

Circular features are also adopted in [31], where a set of randomly placed dots are used to define distinguishable markers that can be detected and recognized without the need for a frame. In this case, to attain robustness and to avoid wrong classification, a large number of dots are required for each marker, thus leading to a likely high number of RANSAC iterations.

Collinearity, that is the property of points that lie on a straight line of remaining aligned after any projective transformation, is another frequently used invariant. Almost invariably this property is exploited by detecting the border edges of a highly contrasted quadrilateral block. This happens, for instance, with the very well-known ARToolkit [16] system which is freely available and has been adopted in countless virtual reality applications. Thanks to the ease of detection and the high accuracy provided in pose recovery [21], this solution is adopted also in many recent marker systems, such as ARTag [9] and ARToolkitPlus [35]. The latter two methods replace the recognition technique of ARToolkit, which is based on image correlation, with a binary-coded pattern (see Fig. 1).

Finally, many papers suggest the use of the cross-ratio among detected points [33,20,28,30], or lines [32] as invariant properties around which to build marker systems. A clear advantage of the cross-ratio is that, being a projective invariant, the recognition can be made without the need of any rectification of the image. Unfortunately, the ease of detection offered by the use of the cross-ratio often comes at the price of a high sensitivity to occlusions or misdetection. In fact, spurious or missing detection completely destroy the invariant structure. Further, cross-ratios exhibit a strongly non-uniform distribution [13], which in several situation limits the overall number of distinctively recognizable patterns.

In this paper, we introduce a novel visual marker system that uses the cross-ratio and other projective invariants to perform both detection and recognition in the image plane, without requiring the estimation of an homography or any other technique of perspective correction. Further, our approach introduces some redundancy by replicating the same pattern on different sides, which can be exploited to obtain a moderated robustness to occlusion or to lower the false positive rate. In addition, the detection and recognition algorithms are both efficient and very simple to implement. In the experimental section, we validate the proposed approach by comparing its performance with two widely used marker systems under a wide range of noise sources applied to synthetically generated scenes. Finally, we also tested the effectiveness of the novel marker when dealing with real images using it to solve a number of different real-world measurement tasks and applications.



(a) ARToolkit **(b)** ARTag **(c)** Intersense **(d)** Line Pencil **(e)** Pi-Tag

Fig. 1 Some examples of fiducial markers that differ both in the detection technique and in the pattern used for recognition. The *black square border* enables detection in ARToolkit **a** and ARTag **b**, but while ARToolkit uses image correlation to differentiate markers, ARTag relies on error-correcting codes. In **c** detection happens by locating concentric ellipses, while the *eight sectors* contained in them encode some information. In **d**, the detection happens directly in image-space using the angular cross-ratio between lines, but the pose estimation requires a stereo camera. Finally, **e** shows an example of the proposed Pi-Tag which is detected and recognized in the image-space

2 Image-space fiducial markers

The proposed marker, which we named *Pi-Tag* (*Projective invariant Tag*), exhibits a very simple design. It is made up of 12 dots placed on the sides of a square: four dots per side, with the corner dots shared. There are two distinct configurations of the dots and each is repeated in two adjacent sides. See, for example, the marker in Fig. 1e: The top and left sides show the same configuration, and so do the bottom and right ones. The two different configurations are not random. In fact they are created in such a way that the cross-ratio of the two patterns is proportional via a fixed constant δ.

The interplay between the detection of these cross-ratios in the image plane and other invariants such as straight lines and conics projections allows for a simple and effective detection and recognition approach for the Pi-Tag.

2.1 Projective invariants

Our approach relies on four types of projective invariants, namely the invariance of the class of ellipses, collinearity, angular ordering (on planes facing the view direction) and cross-ratio.

The invariance of the class of ellipses has been extensively exploited in literature. Circular dots are easy to produce and, since they appear as ellipses under any projective transformation, they are also easy to detect by fitting on them a conic model with a low number of parameters. In addition, while the center of the detected ellipses is not preserved under perspective, if the original dots are small enough, the localization error has been shown to be negligible for most practical purposes [22].

Other advantages of the elliptical fitting include the ability of using the residual error to filter out false detections and to perform gradient-based refinements. For this and other reasons, dots are widely adopted also for accurate tasks such as lens distortion correction, and stereo calibration.

Given a set of points, projective geometry preserves neither distances nor the ratios between them. Fortunately, there are some interesting properties that remain invariant and can be put to use. One is the angular ordering of coplanar points. That is, if we take three points defining a triangle, once we have established an ordering on them (either clockwise or anti-clockwise), such ordering is maintained under any projective transformations that looks down to the same side of the plane.

The second invariant is collinearity and derives from the fact that straight lines remain straight under perspective transformations. Almost all rectangular fiducial markers rely on this property in the detection stage by finding lines in a scene using a wide range of different techniques.

Finally, we use the cross-ratio of four collinear points A,B,C and D, a projective invariant defined as:

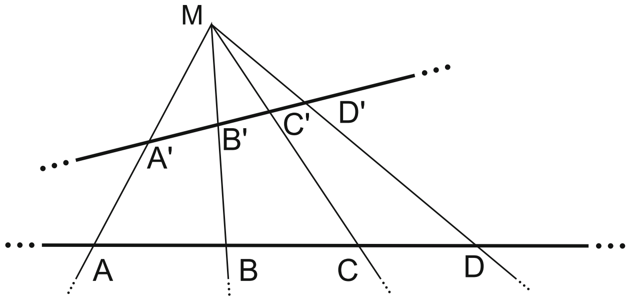


Fig. 2 The cross-ratio of four collinear points is invariant to projective transformations. $\mathrm{cr}(A, B, C, D) = \mathrm{cr}(A', B', C', D')$

$$\mathrm{cr}(A, B, C, D) = \frac{|AB|/|BD|}{|AC|/|CD|}, \tag{1}$$

where $|AB|$ denotes the Euclidean distance between points A and B (see Fig. 2).

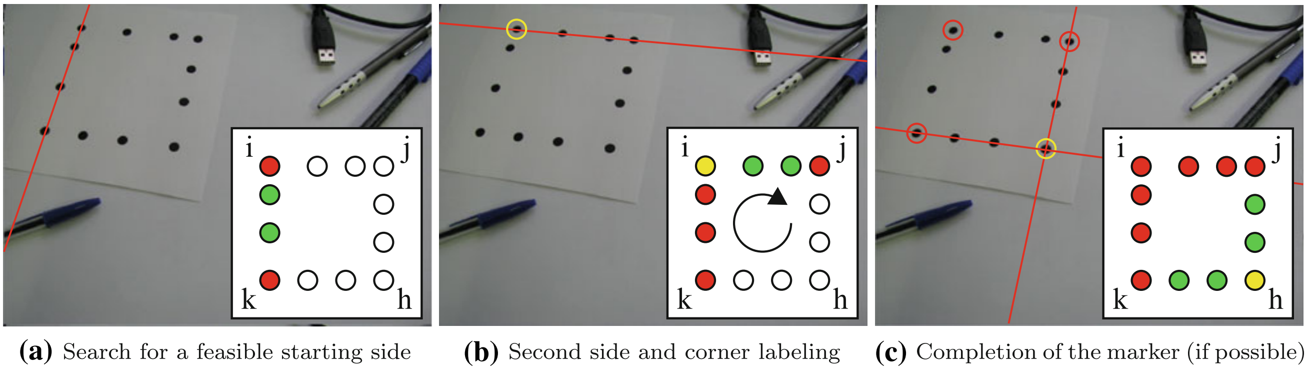
The cross-ratio does not depend on the direction of the line $ABCD$, but depends on the order and the relative positions between the points. The four points can be arranged in $4! = 24$ different orderings which yield six different cross-ratios. Due to this fact, the cross-ratio is unlikely to be used directly to match a candidate set of points against a specific model, unless some information is available to assign an unique ordering to such points.

Many fiducial marker systems use projective and permutation P^2-invariants [23] to eliminate the ambiguities of the different orderings. For example, this invariants are used to track markers or interaction devices for augmented reality in [33] and [19]. It has to be noted, however, that permutation invariance results in the inability to establish correspondences between the detected features and points in the reference model, making it impossible to fully estimate the camera pose without relying to stereo image pairs or other features in the markers.

The main idea behind the design of the proposed Pi-Tags is to combine all the afore-mentioned invariants to identify each dot without ambiguities, even in the presence of moderate occlusions, thus allowing fast and accurate pose estimation. To this end, it should be noted that we assume the imaging process to be projective. While this holds to a reasonable approximation with many computer vision devices with good lens and moderate focal length, wide angle cameras could hinder our assumption due to lens distortion. In this case, a proper distortion-correcting calibration step [29] should be performed before processing.

2.2 Marker detection and recognition

In our design, each marker is characterized by properties that are common to all tags. Specifically, each side of the marker must be made up of exactly four dots, with the corner dots being shared and labeled as in Fig. 3a. For a given constant δ,



(a) Search for a feasible starting side **(b)** Second side and corner labeling **(c)** Completion of the marker (if possible)

Fig. 3 Steps of the marker detection process: in **a**, a good candidate for a side is found by iterating through all the point pairs ($O(n^2)$). In **b**, another connected side is searched for and, if found, the resulting angular ordering is used to label the corners found ($O(n)$). Note that the labeling is unambiguous since the corner i is associated with the lowest cross ratio. Finally, in **c**, the marker is completed (if possible) by finding the missing corner among all the remaining dots. (image best viewed in colors)

a set of Pi-Tags is generated by varying the dots position constrained by the following property:

$$\mathrm{cr}_{ij} = \mathrm{cr}_{ik} = \delta \mathrm{cr}_{kh} = \delta \mathrm{cr}_{jh} \qquad (2)$$

All these properties allow to decouple the detection and recognition pipeline into two separate steps. In the detection process a set of possible marker candidates are localized in the image by exploiting the projective invariants described in the previous section.

First,the dots are located by searching for the ellipses present in the image (projective invariance of conics). To this end, we use the ellipse detector supplied by the OpenCV library [1] applied to a thresholded image. To be resilient to variations in illumination, a locally adaptive threshold is applied by [27]. Some of the ellipses found at this stage may belong to a marker in the scene (if any), others could be possibly generated by noise or clutter.

Next, we group the detected ellipses into potential marker candidates. This is done considering only the centroids of the ellipses (which are a very good approximation for original circle points). The first step to gather all the points belonging to a tag is to find a viable marker side, which can be done by exploiting the straight line invariance (collinearity). For this purpose, we iterate over all the unordered pairs of dots and then, for each pair considered, we check if they are likely to be two corner points (see Fig. 3a). This check is satisfied if exactly two other dots can be found lying within a fixed distance to the line connecting the first two candidate corners. The distance parameter is expressed in pixels and, since the accuracy of the estimated ellipse center is expected to be subpixel, a threshold of one or two pixels is usually enough to avoid false negatives without the risk of including misdetected ellipses. To obtain a better performance, this step can be accelerated using a spatial index, such as a quad-tree, rather than by testing all the ellipses found.

At this point, we have identified a candidate side of the marker. Next, we validate the candidate by finding a third corner of the marker. Again, this is done by iterating over all the dots left and, for each one, by testing if it forms a candidate side with one of the current corner points (i.e. by checking that the line connecting them passes through exactly two ellipses). If a pair of sides is found, then it is possible to test if they belong to a known marker and give a label to each corner. The test is carried on by verifying that the proportion between the cross-ratios of the sides is approximately 1 (in this case we are dealing with kij or jhk adjacent sides) or δ (in this case we are dealing with ijh or hki). The labeling happens by observing the ordering of the sides, which is conserved since always the same face of the tag is seen (see Fig. 3b).

With two sides detected and labeled, we can recognize the marker by comparing the measured cross-ratio with the database of current markers. However, to be more robust, we search for the fourth corner with the same line-based technique. Depending on the application requirements, the search for the fourth point can be mandatory (to reduce the number of false positives and get a more accurate pose) or optional (to allow for the occlusion of at most two sides of the marker).

Once the points are detected and labeled, it is possible to test if they belong to an expected marker. This final step is done by computing the average between the two or four obtained cross-ratios (divided by δ if needed) and by comparing it with all the values in the database of the tags to be searched. If the distance is below a fixed threshold, the marker is then finally recognized. Note that to avoid any ambiguity between tags, the proportion between the cross-ratios of ij sides of any pair should be different from δ.

Regarding the computation complexity of the approach, it is easy to see that finding a starting side is $O(n^2)$ with the number of ellipses, while the two subsequent steps are both $O(n)$. This means that if each detected point triggers the full chain the total complexity of the algorithm could be theoretically as high as $O(n^4)$. However, in practice, given the relatively low probability of getting four ellipses in line with the correct cross ratio, most of the starting side found lead to a correct detection. In addition, even when the starting side is not correct, it is highly probable that the cross-ratio check will stop the false matching at the second step.

While a full probabilistic study would give a more formal insight, in the experimental section we will show that even with a large number of false ellipses the recognition is accurate and it is fast enough for real-time applications.

2.3 Estimation of the camera pose

Having detected and labeled the ellipses, it is now possible to estimate the camera pose. Since the geometry of the original marker is known, any algorithm that solves the PnP problem can be used. In our tests, we used the *solvePnP* function available in OpenCV. However, it should be noted that, while the estimated ellipse centers can be good enough for the detection step, it is reasonable to refine them to recover a more accurate pose. Since this is done only when a marker is found and recognized, the computational cost is limited. In our experiments, we opted for the robust ellipse refinement approach presented in [25].

In addition, to obtain a more accurate localization, one might be tempted to correct the projective displacement of the ellipses centers. However, according to our tests, such correction in general gives little advantage and sometimes leads to a slightly reduction in accuracy. Finally, we also tried the direct method outlined in [15], but we obtained very unstable results, especially with small and skewed ellipses.

3 Experimental validation

In this section, we evaluate the accuracy and speed of the Pi-Tag fiducial markers and compare them with ARToolkit and ARToolkitPlus.

A first batch of tests is performed with synthetically generated images under different condition of viewing direction, noise, and blur. This allows us to compare the different techniques with a perfect ground truth for the camera pose, so that even slight differences in precision can be detected. The accuracy of the recovered pose is measured as the angular difference between the ground truth camera orientation and the obtained pose. While this is a subset of the whole information related to the pose, this is an important parameter in many applications and allows for a concise analysis.

A second set of experiments is aimed at characterizing the behaviour of Pi-Tags with respect to its resilience to occlusion, the presence of false positives, and the sensitivity to the threshold parameters, as well analyze computational time required by the approach.

Finally, four real-world application of the proposed tag are studied; namely, we show the effectiveness of these markers as tools for contactless measurement, camera calibration, and 3D surface alignment. In addition, we also describe a possible use of Pi-Tags with non-square aspect ratio for projected augmented reality applications.

The implementations of ARToolkit and ARToolkitPlus used are the ones freely available at the respective websites. The real images are taken with a 640×480 CMOS webcam for the occlusion test and with a higher resolution 1,280×1,024 CCD computer vision camera with a fixed focal length lens for the measurement tests.

All the experiments have been performed on a typical desktop PC equipped with a 1.6 Ghz Intel Core Duo processor and 2 GB of RAM.

3.1 Accuracy and baseline comparisons

In Fig. 4, the accuracy of our markers is evaluated. In the first set of experiments, the marker is tested at increasing grazing angles and with a minimal additive Gaussian noise. It is interesting to note that oblique angles lead to a higher accuracy for all the methods, as long as the markers are still recognizable. This is explained by the stronger reprojection constraint imposed by angled shots with respect to almost orthogonal views. Pi-Tag shows better results both when the pose is evaluated with the original thresholded ellipses and after the refinement.

In the second test, we evaluated the effects of Gaussian blur, which appears to have a limited effect on all the techniques. This is mainly related to the fact that all methods perform a preliminary edge detection step, which in turn applies a convolution kernel. Hence, it is somewhat expected that an additional blur does not affect much the marker localization. In the third test, an additive Gaussian noise was added to images with an average view angle of 0.3 radians and no artificial blur was added.

The performance of all methods decreases with increasing levels of noise and ARToolkitPlus, while in general more accurate than ARToolkit, breaks when dealing with a noise with a standard deviation greater than 80 (pixel intensities goes from 0 to 255). Finally, the effect of illumination gradient is tested only against ARToolkitPlus (since ARToolkit cannot handle this kind of noise), which, again, exhibits lower accuracy and breaks with just moderate gradients (Fig. 5).

Overall, these experiments confirm that Pi-Tag outperforms the alternative marker systems. This is probably due both to the higher number of pinpointed features and to the

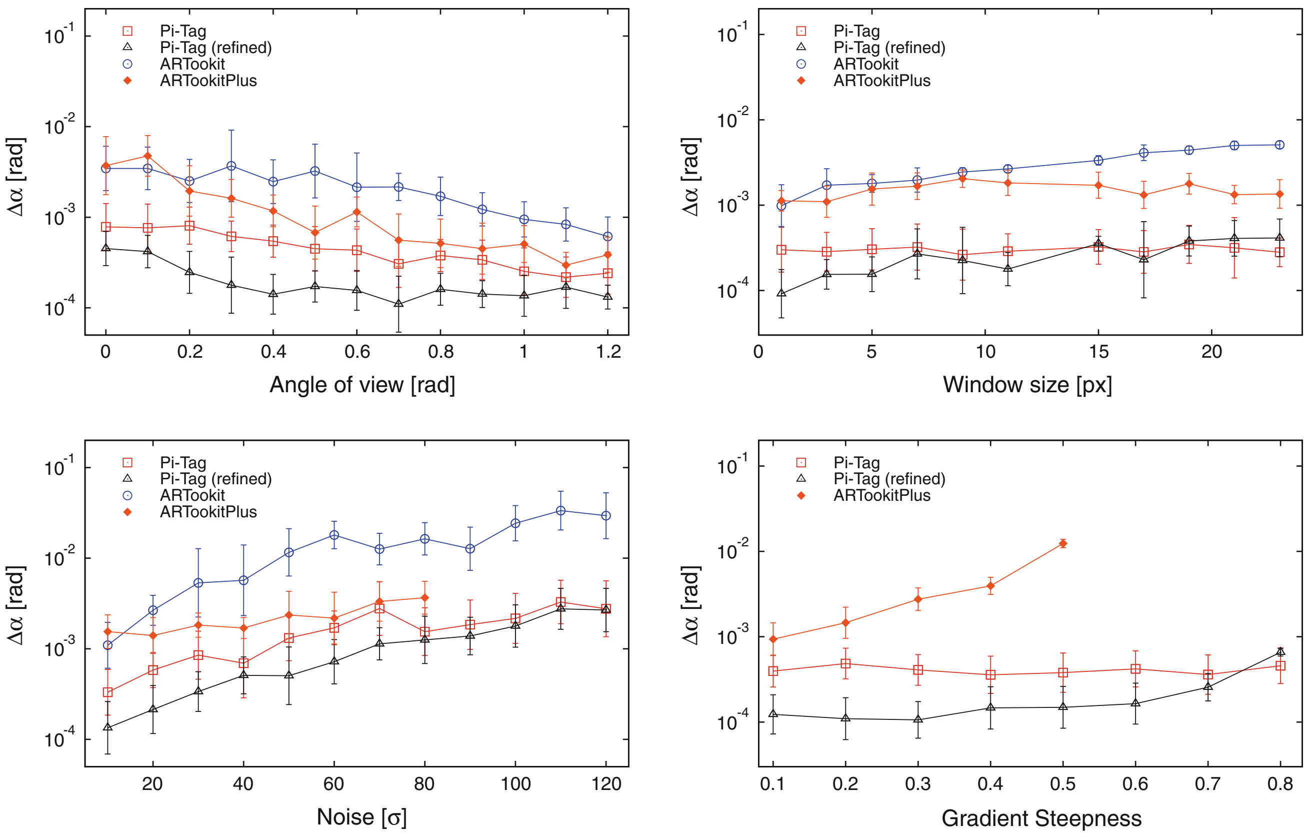


Fig. 4 Evaluation of the accuracy of camera pose estimation with respect to different scene conditions. The *first row* plots the angular error as a function of view angle and Gaussian blur respectively, while the *second row* plots the effects of Gaussian noise (*left*) and illumination gradient (*right*, measured in gray values per image pixel). The proposed method is tested both with and without refinement. Comparisons are made with ARToolkit and ARToolkit Plus

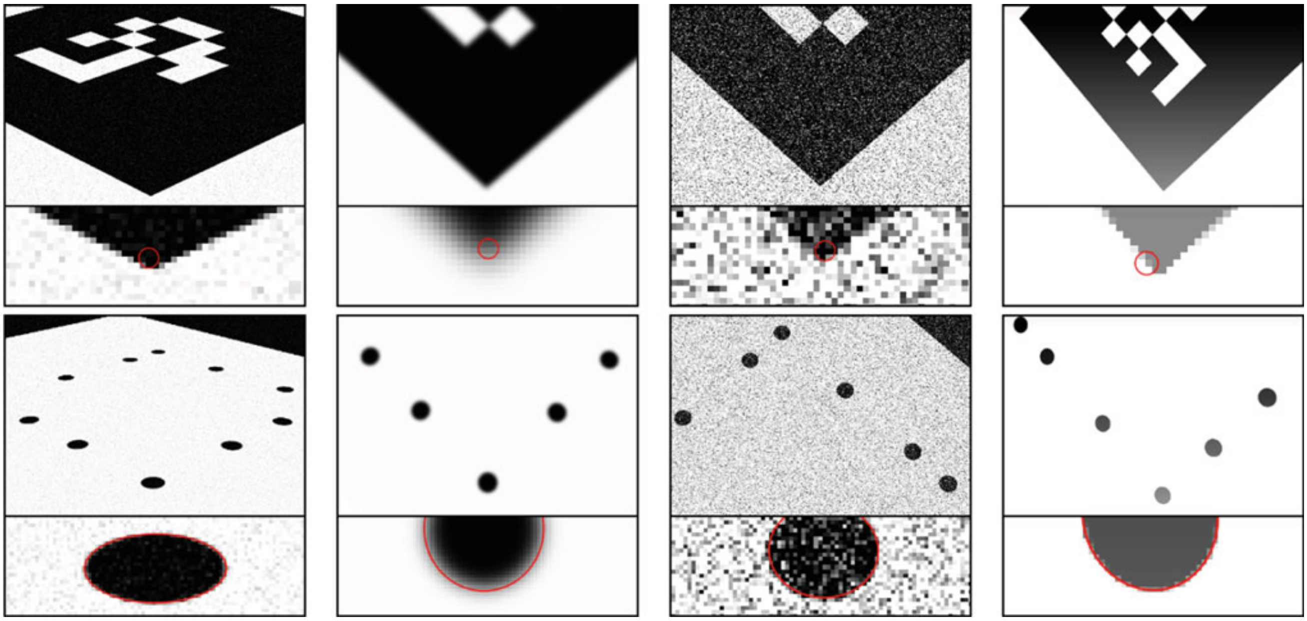


Fig. 5 Some examples of artificial noise used for synthetic evaluation. The artificial noise is, respectively, light Gaussian noise at grazing view angle (*first column*), blur (*second column*), strong Gaussian noise (*third column*) and illumination gradient (*fourth column*). The tested markers shown are ARToolkit Plus (*first row*) and Pi-Tag (*second row*)

better accuracy attainable using circular patterns rather than corners [22].

In practical terms, the improvement is not negligible. In fact an error as low as 10^{-3} radians still produces a jitter of 1 millimetre when projected over a distance of 1 meter. While this is a reasonable performance for augmented reality applications, it is unacceptable for precise contactless measurements.

3.2 Resilience to occlusion and false ellipses

One of the characteristics of Pi-Tag is that it can deal with moderate occlusion. In Fig. 6, we show how occlusion affects the accuracy of the pose estimation (i.e., how well the pose is estimated with a subset of the dots, regardless of the possibility of recognizing the marker with those dots).

While we observe a decrease in the accuracy as we increase the occlusion, the precision is still acceptable even when almost half of the dots are not visible, especially for the refined version of the tag. In Fig. 7, we evaluate the proportion of false marker detections obtained by introducing a large amount of false ellipses at random position and scale. When the threshold on the cross-ratio is kept tight, it is possible to obtain a very low rate of false positives even with a large number of random dots.

3.3 Performance evaluation

Our tag system is designed for improved accuracy and robustness to occlusion rather than for high detection speed. This is quite apparent in Fig. 8, where we can see that the recognition could require from a minimum of about 10 ms (without false ellipses) to a maximum of about 150 ms.

By comparison, ARToolkit Plus is about an order of magnitude faster [35]. However, it should be noted that, despite being slower, the frame rates reachable by Pi-Tag (from 100 to about 8/10 fps) are still sufficient for real-time applications (in particular when few markers are viewed at the same time). Further, our code is not as heavily optimized as ARToolkitPlus, which gained a factor of 10 performance gain with respect to ARToolkit. It is reasonable to assume that a similar optimization effort would result in a similar gain in performance.

3.4 Behavior on real videos

In addition to the evaluation with synthetic images, we also performed some qualitative and quantitative tests on real videos. In Fig. 11, some experiments with common occlusion scenarios are presented. Note that when at least two sides are fully visible the marker is still recognized and the correct pose is recovered.

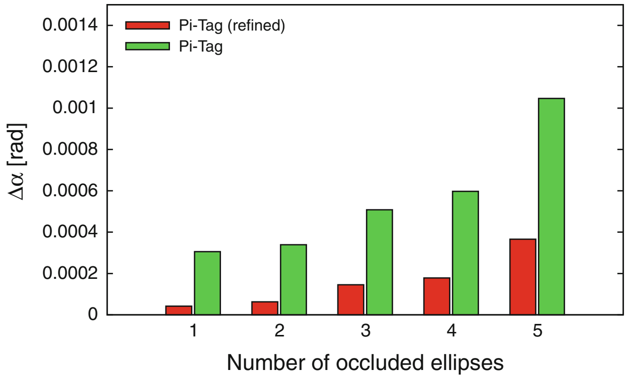


Fig. 6 Evaluation of the accuracy of the estimated camera pose when some dot of the marker are occluded (note that if more than five dots are missing the marker is not detected).

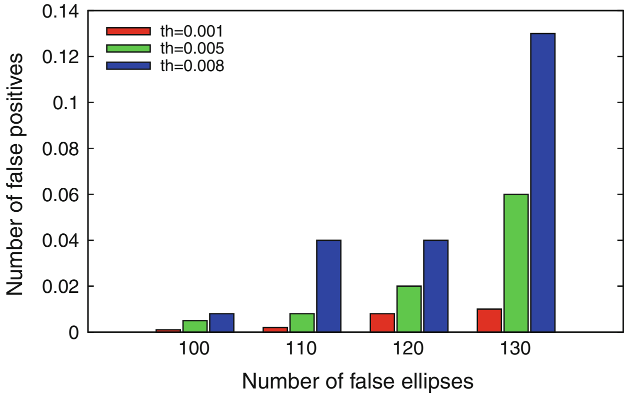


Fig. 7 Evaluation of the number of false positive markers detected as a function of the number of false ellipses introduced in the scene and the threshold applied to the cross-ratio

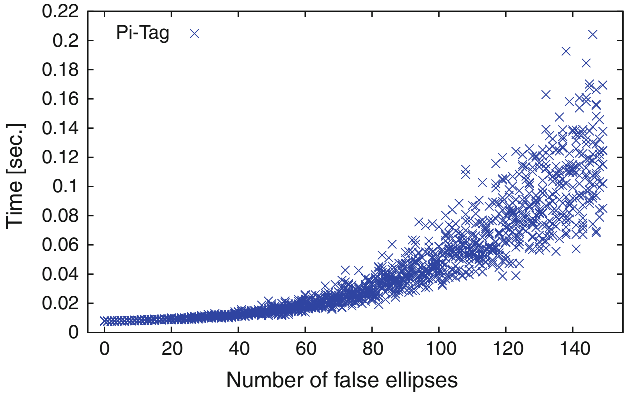


Fig. 8 Evaluation of the detection and recognition time for the proposed marker as random ellipses are artificially added to the scene

In Fig. 9, we plot the recognition rate of the markers as a function of the cross-ratio threshold. This was computed from a 10-min video presenting several different viewing conditions. It is interesting to note that even with a small threshold we can obtain a complete recall (compare this with the threshold in Fig. 7).

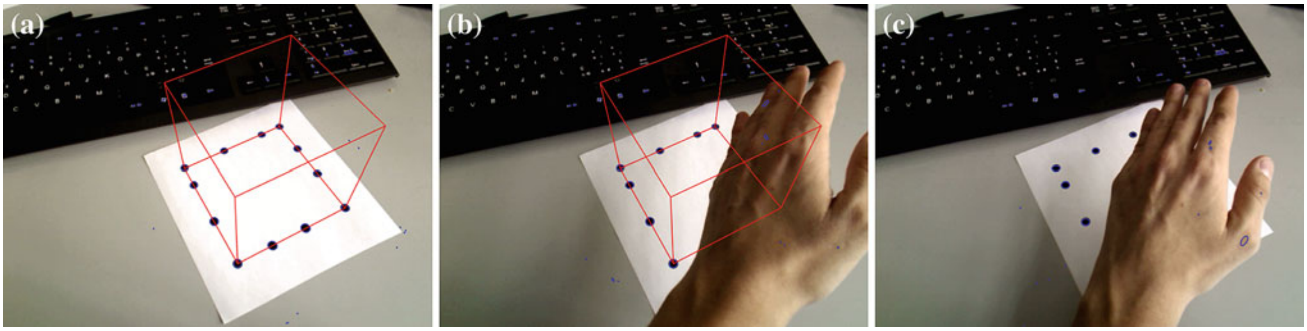


Fig. 9 Some examples of the behaviour in real videos: In **a**, the marker is not occluded and all the dots contribute to the pose estimation. In **b**, the marker is recognized even if a partial occlusion happens. In **c**, the marker cannot be detected as the occlusion is too severe and not enough ellipses are visible

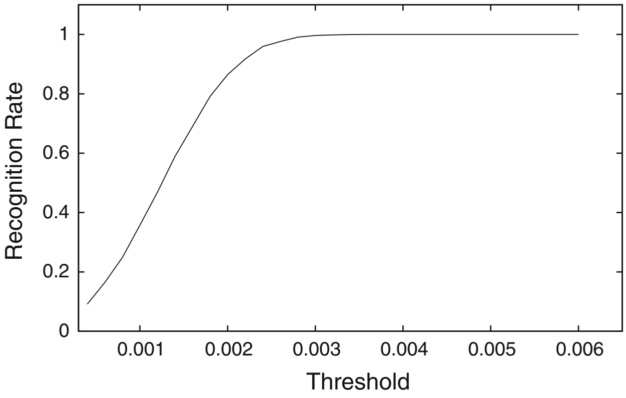


Fig. 10 Evaluation of the recognition rate achieved on a real video of about 10 min in length, with respect to different thresholds applied to the cross-ratio

Finally, Fig. 10 highlights an inherent shortcoming of our design: The relatively small size of the base features may result in a failure of the ellipse detector when the tag is far away from the camera or very angled, causing the dots to become too small or to blended together.

3.5 Using Pi-Tag for camera calibration

Camera calibration is a fundamental task whenever imaging devices are to be used in measurement applications. In fact, if the intrinsic parameters of the devices (and thus the image formation process) are not known with high accuracy, it is not possible to relate the points on the image plane to the phenomena that generated them. In the case of Pi-Tags, detection and recognition entirely happen in the image plane, for this reason calibration is not needed per se. However, a calibration procedure based solely on Pi-Tags provides a useful testbed.

There are many different image formation models and of course each one comes with a different set of parameters. In the following tests, we adopted the model proposed by [12]. In this model, the imaging device is represented as a pinhole camera whose incoming rays are displaced on the image plane through a polynomial distortion. Such distortion is parametrized by three coefficient of the polynomial usually labeled k_1, k_2 and k_3, being k_1 the most relevant (in terms of displacement) and k_3 the least relevant.

Once the distortion is factored out, the pinhole part of the model is defined through the principal point (i.e. the projection on the image plane of the projective center) labeled as (cx, cy) and the focal length (fx, fy) (which are two parameters to account for non-square pixels). Since we are dealing with low distortion cameras that are equipped with sensors with square pixels, we considered only the first distortion coefficient k_1 and we assumed $fx = fy$.

The first set of calibration experiments was performed using a target made up of Pi-Tags placed according to a known geometry and printed on an A3 sheet with a standard

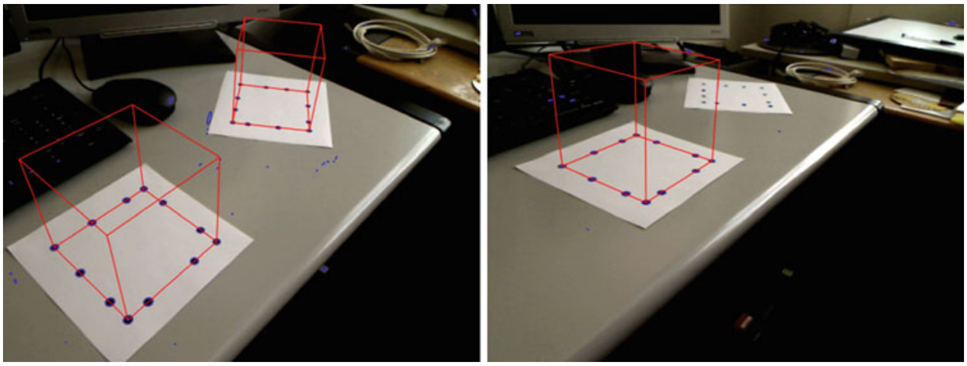


Fig. 11 Recognition fails when the marker is angled and far from the camera as the ellipses detectors cannot detect the circular features

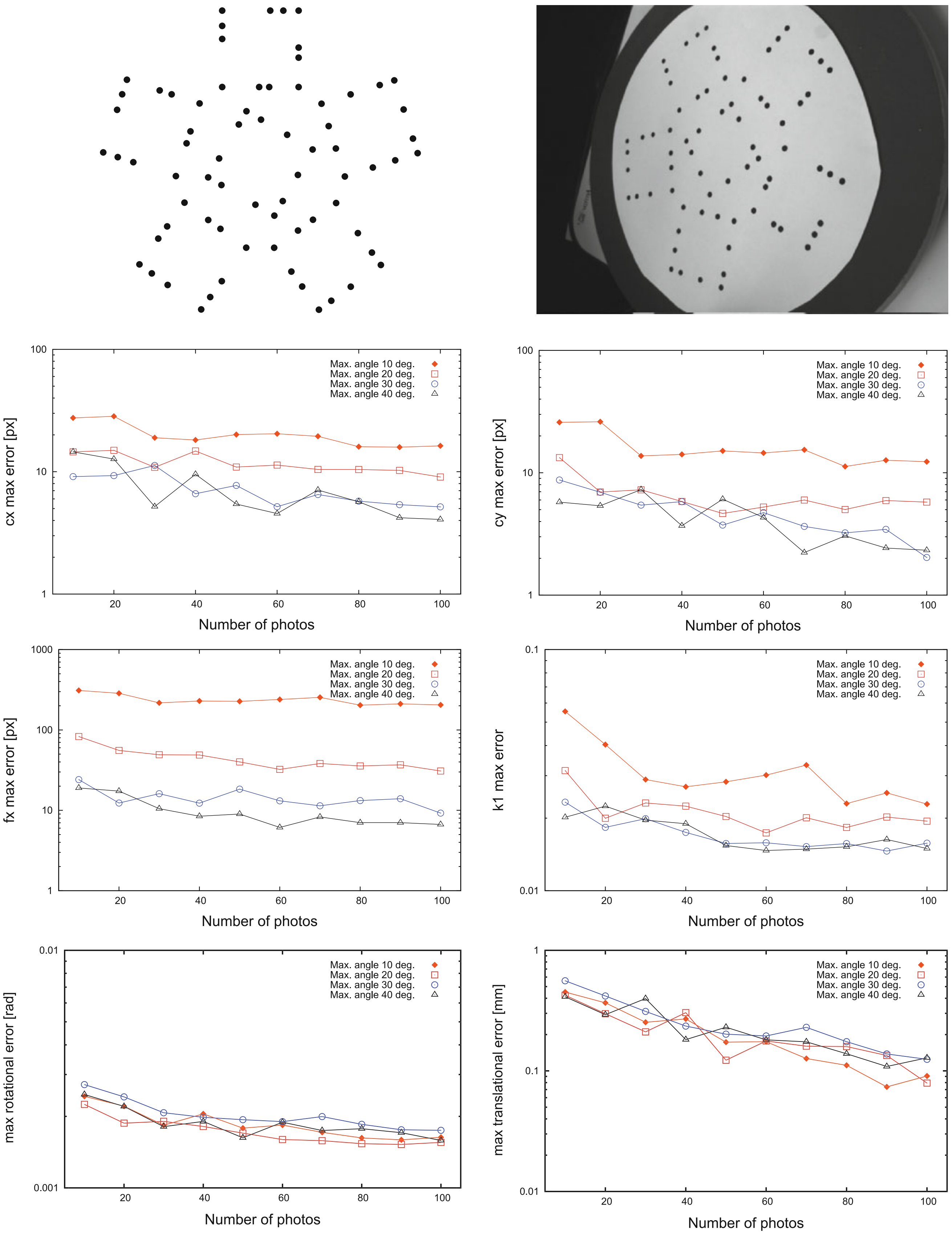


Fig. 12 Evaluation of the quality of mono and stereo calibration obtained using Pi-Tags as fiducial markers

inkjet printer (see Fig. 12). Several shots of the target were captured with different viewing angles and at different distances. For each shot the tags were detected and recognized, and an association between the 2D points on the image plane and the 3D point of the known model was stored. These data were finally fed to the camera calibration procedure available in the OpenCV library. Since both the target viewing angle and the number of shots are relevant factors for the quality of the calibration, we studied the effect of both.

In the first four graphs of Fig. 12, we show the absolute distance between the parameters recovered with the described procedure and a ground truth calibration performed with a full checkerboard pattern with about 600 reference corner and using 200 shots. Specifically, for each number of shots and maximum angle we selected 30 random set of images to be used for calibration.

The distance plotted in the graph is the maximum absolute error committed in the 30 calibrations. From these graphs, it is possible to see that taking shots with a large enough viewing angles is important. This is due both to the stronger constraint offered to pinhole parameters by angled targets, and to the more accurate pose estimation offered by Pi-Tag when the angle of view is not negligible (see Fig. 4). In addition, we can also observe that taking a large number of samples increases monotonically the accuracy obtained.

In the second set of calibration experiments, the tags were used to estimate the relative pose between two cameras of known intrinsic model. This stereo calibration is useful in many reconstruction tasks where the epipolar geometry between more than one camera can be exploited to fully localize the 3D points that are imaged (see [11]).

Again, we estimated a ground truth relative pose between a pair of identical fixed cameras using a specialized target and plotted on the bottom row of Fig. 12 the maximum absolute error between the ground truth and the values obtained in 20 calibrations performed on randomly selected shots with a given maximum viewing angle. In this condition, the viewing angle is less important, but still a large number of shots gives better results.

3.6 Contactless measurements

A calibrated camera can be used in conjunction with any detectable marker of a known size as a contactless measurement tool. To assess the precision offered for this use scenario, we printed two Pi-Tags and two ARToolkitPlus tags at a distance (center to center) of 200 millimetres (see Fig. 13). Subsequently, we took several shots and estimated such distance. In the graph displayed in Fig. 13 we plotted the difference (with sign) between the measured and real distance between tags at several viewing angles. Pi-Tag consistently exhibits smaller errors and a smaller variance. As usual, the measure improves slightly as the viewing angle increases. It is interesting to note that, according to our measurements, Pi-Tag has a tendency to underestimate the distance slightly, while ARToolkitPlus do exactly the opposite.

In Fig. 14, we show two scatter plots that depict respectively the error in localization on the *x*/*y* plane (as the norm of the displacement vector) with respect to the position of the target and the difference in depth estimation (signed) with respect to the depth of the target. In this case, the ground truth was obtained using a stereo camera pair properly calibrated. Also in this test, Pi-Tag obtains better results than ARToolkitPlus. The larger error in the localization near the image border is probably due to the inability of the polynomial distortion model to fully capture the pixel displacement far away from the principal point. Also, the spread of the error in estimating the depth is larger when the target is far from the camera, which is expected as the resolution of the detected ellipses decreases and so does the accuracy in the location of their centers.

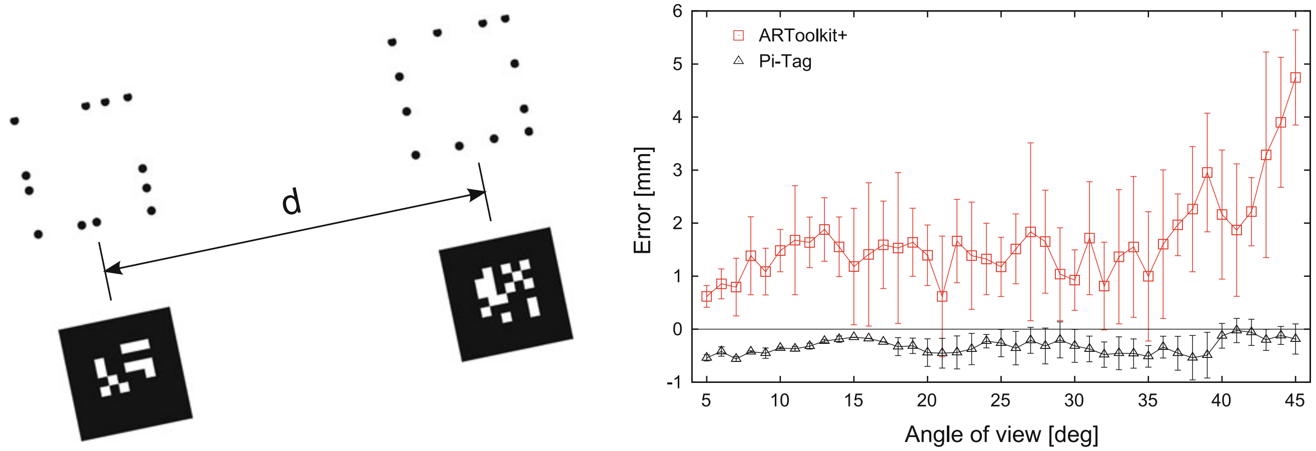


Fig. 13 Performance of the proposed fiducial marker as a tool for image-based measurement

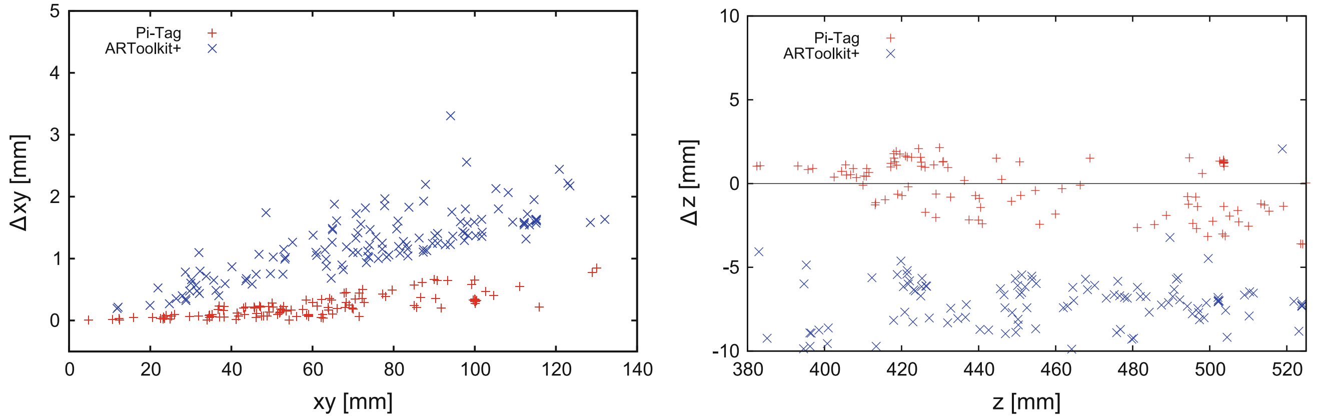


Fig. 14 Analysis of the measurement error committed with respect to different positions of the marker pair

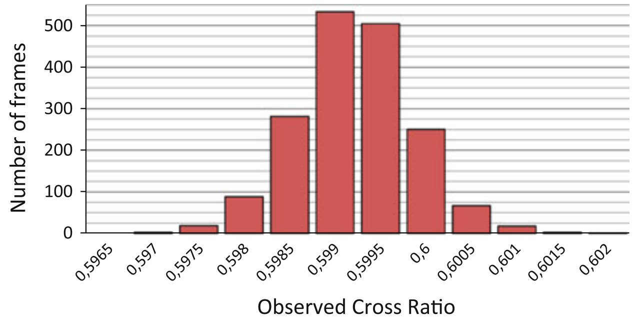


Fig. 15 Cross ratios measured in a real video sequence

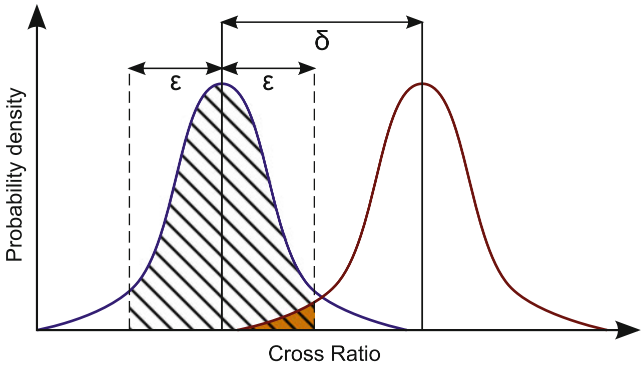


Fig. 16 Relation between recognition margin and cross-ratio separation among markers

3.7 Scalability over the number of markers

In many practical scenarios, it could be useful to place a large number of markers in the scene. For this reason, it is important to assess the ability of the proposed approach to generate markers distinctive enough to avoid wrong classifications even with big databases. To this end, we first evaluated the distribution of the measured cross-ratio in a real video sequence of about 2,000 frames showing a marker under various angles and lighting conditions. As shown in Fig. 15, the acquired cross-ratio appears to behave as a Gaussian distributed random variable.

If we deem this model as reasonable, it is easy enough to estimate both the probability of missing a marker and of a wrong classification (see Fig. 16). Given the standard deviation of the measured cross-ratio σ_{cr} (that we assume uniform over all the database) and a tolerance ϵ between the acquired value and the target cross-ratio cr, the probability of a false negative for a given marker is exactly:

$$1-\int_{\mathrm{cr}-\epsilon}^{\mathrm{cr}+\epsilon}\frac{1}{\sigma_{\mathrm{cr}}\sqrt{2\pi}}e^{-\frac{(x-\mathrm{cr})^2}{2\sigma_{\mathrm{cr}}^2}}\,\mathrm{d}x \qquad (3)$$

In addition, given a minimum separation between cross-ratios in the database of δ (assumed to be bigger that ϵ), the probability of a wrong classification is upper bounded by:

$$2\int_{-\infty}^{\mathrm{cr}-(\delta-\epsilon)}\frac{1}{\sigma_{\mathrm{cr}}\sqrt{2\pi}}e^{-\frac{(x-\mathrm{cr})^2}{2\sigma_{\mathrm{cr}}^2}}\,\mathrm{d}x \qquad (4)$$

By choosing apt values for ϵ and δ, it is possible to set the sought balance between an high recognition ability and a low number of misclassifications. For instance, as the measured standard deviation in our test video was $\sigma_{\mathrm{cr}} = 6 \times 10^{-4}$ a choice of $\epsilon = 2 \times 10^{-3}$ would grant an expected percentage of false negative lower than 0.1 %. At the same time, a choice of $\delta = 4 \times 10^{-3}$ would set the rate of wrong classifications below 0.01 %.

To translate these numbers into a feasible database size, it is necessary to account for the physical size of the marker and of the dots. In our test video, we used a square marker with a side of 10 cm and with a dot diameter of 1 cm.

Within these conditions, we were able to obtain cross-ratios from 0.026 to 1.338 keeping enough white space between dots to make them easily detectable by the camera. Assuming that the cross-ratios in the database are produced to be are evenly distributed, a span of about 1.3 grants for a total of about 300 distinct markers with the above-mentioned levels of false negatives and wrong classifications.

3.8 Registration of 3D surfaces

To further evaluate the utility of the new marker design in practical scenarios, we performed a last set of qualitative experiments. To this end, we captured several range images from different objects using a 3D scanner based on structured light. These objects were placed on a turntable and surrounded with Pi-Tags (see Fig. 17). During each acquisition step, we took an additional shot that captures the Pi-Tags in natural lighting, thus allowing us to use them to recover the pose of the object on the turntable. This is a typical

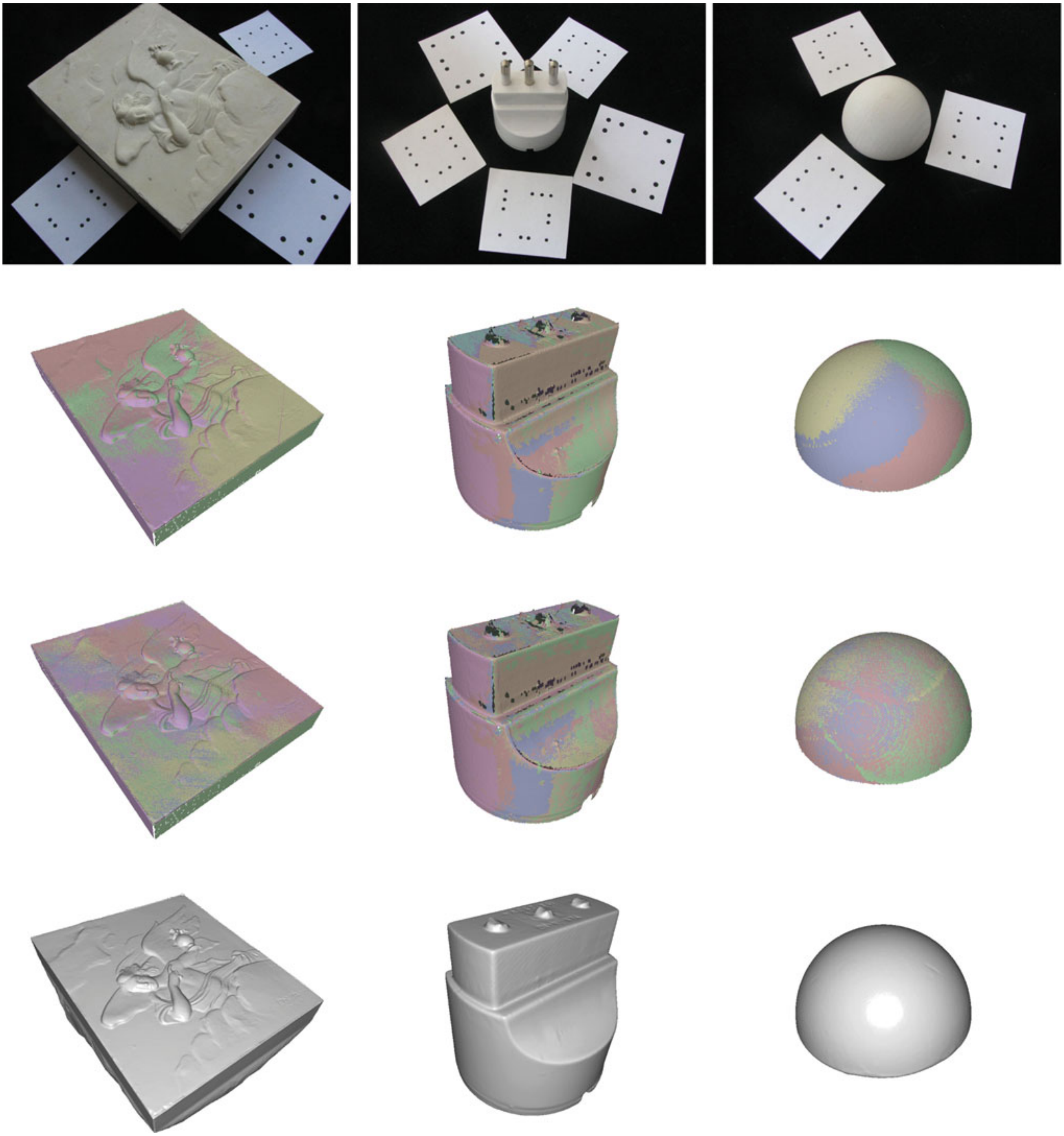


Fig. 17 Examples of surface reconstructions obtained by acquiring several ranges with a structured-light scanner and by using Pi-Tag markers to set a common reference (image best viewed in *color*)

application of artificial markers, since most algorithms used to align 3D surfaces need a good initial motion estimation to guarantee a correct convergence. In the second row of Fig. 17, we show the overlap of several ranges using the pose estimated with Pi-Tags, without any further refinement. In the third row, the initial alignment is refined using a state-of-the-art variant of the well-known ICP algorithm (see [26]). After the refinement, slight improvements in the registration can be appreciated, especially in the “hemisphere” object. The smooth blending of colors obtained means that ICP was able to obtain a good alignment, which in turn testifies the quality of the initial pose estimated using the Pi-Tags. In the last row, we present the watertight surface obtained after applying the Poisson surface reconstruction algorithm [17] to the aligned range images. Overall, the surfaces are smooth and do not exhibit the typical artefacts related to misalignment. In addition, the fine details of the first object are preserved.

3.9 Applications in projected augmented reality

An interesting property of the proposed tag design is that there is no need for it to be square. In fact, any aspect ratio can be adopted without compromising the properties of the cross ratio that are needed for the recognition of the tag and for the pose estimation. We combined this exact property with the fact that the inside of the tag is blank (as with other frame-based designs [34]) to build an additional application within the domain of the projected augmented reality.

Specifically, we built a system where a Pi-Tag is used both as a passive input device and as a display. The input device can be regarded as a 3D mouse that can be used to explore interactive content. The ability to display data on the tag is obtained using an external projector that is aimed toward the white surface internal to the tag.

A schematic representation of the setup can be seen in Fig. 19. The navigation device is basically a rectangular rigid board that exhibits a white matte projection area and a frame that contains the fiducial marker to track. Since the rigid transform that binds the camera to the projector is known and the projector frustum itself corresponds to the map area, all the parameters are available to reconstruct the position of the navigation device with respect to the map and to the projector and thus to display on the matte area some contextual data related to the location observed by the user.

The geometrical relation between the projector and the navigation device is used to rectify the displayed image so that it appears exactly as if it was formed on the screen of an active device. By printing different markers, more than one navigation device can be used simultaneously, thus allowing many users to operate on the table. Finally, since the marker position is determined in 3D, additional functions such as zooming can be controlled through the vertical position of the device. In Fig. 18, an actual implementation of the setup and the zooming effect attainable are shown.

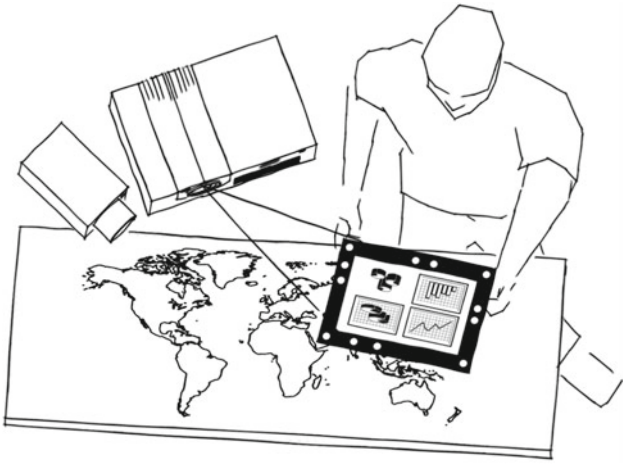


Fig. 18 A schematic representation of the setup

The main challenge of the projection calibration is to estimate its projection matrix $P = \mathbf{K_p}[R_p|T_p]$, where

$$\mathbf{K_p} = \begin{bmatrix} fx_p & 0 & cx_p \\ 0 & fy_p & cy_p \\ 0 & 0 & 1 \end{bmatrix}$$

are projector intrinsic parameters, and $[R_p|T_p]$ is the relative pose of the projector with respect to the marker, or the extrinsic parameters. Once the matrix P has been estimated, a 3D point p_m lying on the marker plane can be projected by transforming its 3D coordinates $[x_w y_w 0]^T$ to projector image-space pixel coordinates $[u_p v_p]^T$ with the following equation:

$$\mathbf{K_p} = \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} = P \begin{bmatrix} x_w \\ y_w \\ 0 \\ 1 \end{bmatrix} = Pp_w$$

Unfortunately, the projector cannot estimate the relative pose $[R_p|T_p]$ by itself because it is a pure output device. To provide that data, a camera is placed nearby ensuring that the viewing frustum of the projector is contained in the viewing frustum of the camera. As long as the relative position between the camera and projector remains unchanged, $[R_p|T_p]$ can be estimated in terms of the camera pose $[R_c|T_c]$ obtained via fiducial markers in the following way:

$$\begin{bmatrix} R_p & T_p \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} R_c p & T_c p \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} R_c & T_c \\ \mathbf{0} & 1 \end{bmatrix}$$

The estimation of $\mathbf{K_p}$ and $[R_{cp}|T_{cp}]$ can be obtained from a set of known 3D-2D correspondences as in Sect. 3.5, however, as the projector cannot “see” the markers and retrieve 3D positions of dots in the calibration target, an alternative method is used to provide this mapping.

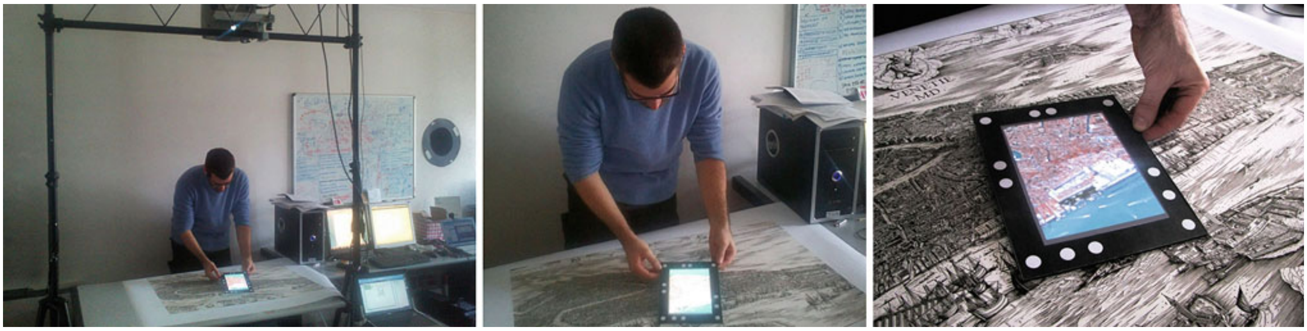


Fig. 19 Actual setup and examples of usage by moving the controller in space

A big square Pi-Tag marker is printed on a planar surface and placed under the camera/projector frustum (Fig. 20). Once the tag is placed, a snapshot is taken by the camera and used for background subtraction. This allow us to project a dot with the projector by randomizing its 2D position in projector plane, and detect its center with no ambiguity using the camera. If the camera detects that the projected dot lies inside the marker, the 3D position of the dot can be recovered because the marker plane position is known with respect to the camera via Pi-Tag pose estimator. The whole process can be summarized as follows:

1. A planar surface with a Pi-Tag marker is placed randomly under camera/projector frustum, and a snapshot is taken.
2. A dot $p_p = [u_p v_p]^T$ is projected randomly by the projector. Via background subtraction, the camera can identify the dot projected and determine its 2D position $p_c = [u_c v_c]^T$ in the camera image plane.
3. If the 2D position of the dot lies inside the marker, its 3D position $p_w = [x_w y_w z_w]^T$ (in camera world) can be recovered as the intersection of the line from the camera center of projection **0** and the point $[\frac{u_c - cx_c}{fx_c} \frac{v_c - cy_c}{fy_c} 1]^T$ and the marker plane, computed using Pi-Tag pose estimator.
4. Steps 2 and 3 are repeated to collect hundreds of 3D-2D correspondences (p_w, p_p) from this point of view.
5. Steps 1 to 4 are repeated to collect correspondences between different point of views. For our purposes, about half a dozen of different point of views is usually enough.
6. OpenCV calibrateCamera function is used to estimate $\mathbf{K_p}$ and the rigid motion $[R_{cpi}|T_{cpi}]$ between the randomly projected 3D points in camera world from each point of view and the projector. As final $[R_{cp}|T_{cp}]$, we simply choose the rigid motion with respect to the first point of view $[R_{cp0}|T_{cp0}]$ but different strategies may be used.

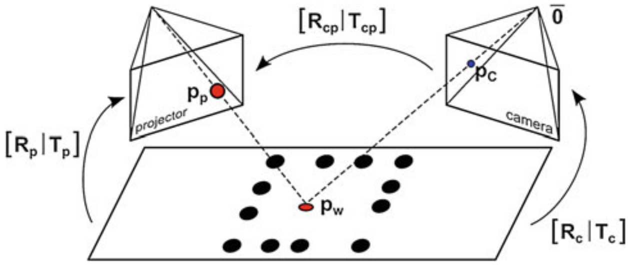


Fig. 20 Geometric relation between the entities involved in the projector calibration procedure

Only the first step requires human intervention instead of points 2 and 3 that needs to be iterated thoroughly to collect a large set of correspondences. Even if the process is automatic, steps 2 and 3 may require a very long time depending by the probability that the random dot p_p will lie inside the marker at each iteration. To speed up the calibration procedure, for each point of view, after at least four projections lying inside the marker, an homography H can be computed that maps points from camera image plane to projector image plane. With the homography H, each point p_p can be randomized directly lying inside the marker, thus eliminating the waste of time required to guess the correct set of positions. In our setup we are able to collect more than ten correspondences per second, for an average calibration time of less than 15 min.

4 Conclusions

The novel fiducial marker proposed in this paper exploits the interplay between different projective invariants to offer a simple, fast and accurate pose detection without requiring image rectification. Our experimental validations show that the precision of the recovered pose outperforms the current state-of-the-art. In fact, even if relying only on a maximum on 12 dots, the accuracy achieved using elliptical features has been proven to give very satisfactory results even in presence of heavy artificial noise, blur and extreme illumination conditions. This accuracy can be further increased using an ellipse refinement process that takes into account image gradients. The marker design is resilient to moderate occlusion without severely affecting pose estimation accuracy. The internal redundancy exhibited by its design allows to compensate the strongly non-uniform distribution of cross-ratio and also permits a good trade-off between the recognition rate and false-positives. Even taking into account the limited number of discriminable cross-ratios, the design still offers a reasonable number of distinct tags. Further, the proposed design leaves plenty of space in the marker interior for any additional payload. Since it works entirely in image-space, our method is affected by image resolution only during the ellipse detection step, and is fast enough for most real-time augmented reality applications.

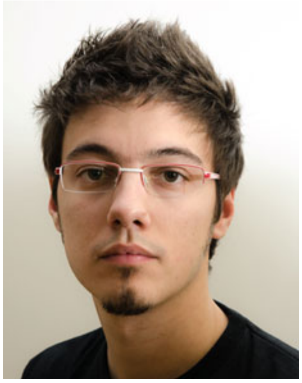
Of course, those enhancements do not come without some drawbacks. Specifically, the small size of the circular points used can lead the ellipse detector to miss them at great distance, low resolution, or if the viewing point is very angled with respect to the marker's plane. These limitations can be partially overcome by increasing the ratio between the size of the ellipses and the size of the marker itself, thus limiting the range of possible cross-ratio values and the total number of different tags that can be successfully recognized.

References

1. Bradski, G., Kaehler, A.: Learning OpenCV: Computer Vision with the OpenCV Library, 1st edn. O'Reilly Media, Inc., Cambridge (2008)
2. Cameron, J., Lasenby, J.: Estimating human skeleton parameters and configuration in real-time from markered optical motion capture. In: Conference on Articulated Motion and Deformable Objects (2008)
3. Cho, Y., Lee, J., Neumann, U.: A multi-ring color fiducial system and a rule-based detection method for scalable fiducial-tracking augmented reality. In: Proceedings of International Workshop on Augmented Reality (1998)
4. Claus, D., Fitzgibbon, A.W.: Reliable automatic calibration of a marker-based position tracking system. In: IEEE Workshop on Applications of Computer Vision (2005)
5. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: Monoslam: real-time single camera slam. IEEE Trans. Pattern Anal. Mach. Intell. **26**(6), 1052–1067 (2007)
6. Dorfmller, K.: Robust tracking for augmented reality using retroreflective markers. Comput. Graph. **23**(6), 795–800 (1999)
7. Douxchamps, D., Chihara, K.: High-accuracy and robust localization of large control markers for geometric camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. **31**, 376–383 (2009)
8. Fiala, M.: Linear markers for robot navigation with panoramic vision. In: Proceedings of the 1st Canadian Conference on Computer and Robot Vision, CRV '04, pp. 145–154. IEEE Computer Society, Washington, DC (2004)
9. Fiala, M.: Designing highly reliable fiducial markers. IEEE Trans. Pattern Anal. Mach. Intell. **32**(7), 1317–1324 (2010)
10. Gatrell, L., Hoff, W., Sklair, C.: Robust image features: concentric contrasting circles and their image extraction. In: Proceedings of Cooperative Intelligent Robotics in Space. SPIE, Washington (1991)
11. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2000)
12. Heikkilä, J.: Geometric camera calibration using circular control points. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 1066–1077 (October 2000)
13. Huynh, D.Q.: The cross ratio: a revisit to its probability density function. In: Proceedings of the British Machine Vision Conference BMVC 2000 (2000)
14. Jiang, G., Quan, L.: Detection of concentric circles for camera calibration. IEEE Int. Conf. Comput. Vis. **1**, 333–340 (2005)
15. Kannala, J., Salo, M.,: Heikkilä, J.: Algorithms for computing a planar homography from conics in correspondence. In: British Machine Vision Conference (2006)
16. Kato, H., Billinghurst, M.: Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In: Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality. IEEE Computer Society, Washington, DC (1999)
17. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the Fourth Eurographics symposium on Geometry processing, SGP '06, pp. 61–70. Aire-la-Ville, Switzerland (2006)
18. Knyaz, V.A. Head Of Group, Sibiryakov, R.V.: The development of new coded targets for automated point identification and non-contact surface measurements. In: 3D Surface Measurements, International Archives of Photogrammetry and Remote Sensing (1998)
19. Li, Y., Wang, Y.-T., Liu, Y.: Fiducial marker based on projective invariant for augmented reality. J. Comput. Sci. Technol. **22**, 890–897 (2007)
20. Loaiza, M., Raposo, A., Gattass, M.: A novel optical tracking algorithm for point-based projective invariant marker patterns. In: Proceedings of the 3rd International Conference on Advances in Visual Computing, vol. Part I, ISVC'07, pp. 160–169. Springer, Berlin (2007)
21. Maidi, M., Didier, J.-Y., Ababsa, F., Mallem, M.: A performance study for camera pose estimation using visual marker based tracking. Mach. Vis. Appl. **21** (2010)
22. Mallon, J., Whelan, P.F.: Which pattern? biasing aspects of planar calibration patterns and detection methods. Pattern Recogn. Lett. **28**(8), 921–930 (2007)
23. Meer, P., Lenz, R., Ramakrishna, S.: Efficient invariant representations. Int. J. Comput. Vis. **26**, 137–152 (1998)
24. Naimark, L., Foxlin, E.: Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. In: Proceedings of the 1st International Symposium on Mixed and Augmented Reality, ISMAR '02. IEEE Computer Society, Washington, DC (2002)
25. Ouellet, J., Hebert, P.: Precise ellipse estimation without contour point extraction. Mach. Vis. Appl. **21** (2009)
26. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: Proceedings of the Third International Conference on 3D Digital Imaging and Modeling, pp. 145–152 (2001)
27. Sauvola, J., Pietikainen, M.: Adaptive document image binarization. Pattern Recogn. **33**(2), 225–236 (2000)
28. Teixeira, L., Loaiza, M., Raposo, A., Gattass, M.: Augmented reality using projective invariant patterns. In: Advances in Visual Computing. Lecture Notes in Computer Science, vol. 5358. Springer, Berlin (2008)
29. Thormählen, T., Broszio, H.: Automatic line-based estimation of radial lens distortion. Integr. Comput. Aided Eng. **12**(2), 177–190 (2005)
30. Tsonisp, V.S., Konstantinos, V.Ch., Trahaniaslj, P.E.: Landmark-based navigation using projective invariants. In: Proceedings of the 1998 IEEE International Conference on Intelligent Robots and Systems. IEEE Computer Society, Victoria, Canada (1998)
31. Uchiyama, H., Saito, H.: Random dot markers. In: Virtual Reality Conference, IEEE, pp. 271–272 (2011)
32. van Rhijn, A., Mulder, J.D.: Optical tracking using line pencil fiducials. In: Proceedings of the Eurographics Symposium on Virtual Environments (2004)

33. Van Liere, R., Mulder, J.D.: Optical tracking using projective invariant marker pattern properties. In: Proceedings of the IEEE Virtual Reality Conference. IEEE Press, New York (2003)
34. Wagner, D., Langlotz, T., Schmalstieg, D.: Robust and unobtrusive marker tracking on mobile phones. In: Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality, ISMAR '08, pp. 121–124. IEEE Computer Society, Washington, DC (2008)
35. Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D.: Real time detection and tracking for augmented reality on mobile phones. IEEE Trans. Vis. Comput. Graph. **99**, 355–368 (2010)
36. Walthelm, A., Kluthe, R.: Active distance measurement based on robust artificial markers as a building block for a service robot architecture. In: IFAC Symposium on Artificial Intelligence in Real Time Control. Budapest Polytechnic, Budapest (2000)
37. Yoon, J.-H., Park, J.-S., Kim, C.: Increasing camera pose estimation accuracy using multiple markers. In: Advances in Artificial Reality and Tele-Existence. Lecture Notes in Computer Science, vol. 4282. pp. 239–248. Springer, Berlin (2006)
38. Yu, Q., Li, Q., Deng, Z.: Online motion capture marker labeling for multiple interacting articulated targets. Comput. Graph. Forum **26**(3), 477–483 (2007)
39. Yu, R., Yang, T., Zheng, J., Zhang, X.: Real-time camera pose estimation based on multiple planar markers. In: Proceedings of the 2009 Fifth International Conference on Image and Graphics ICIG '09, pp. 640–645. IEEE Computer Society, Washington, DC (2009)
40. Zhang, X., Fronz, S., Navab, N.: Visual marker detection and decoding in ar systems: a comparative study. In: Proceedings of the 1st International Symposium on Mixed and Augmented Reality, ISMAR '02, pp. 97. IEEE Computer Society, Washington, DC (2002)

Author Biographies



Filippo Bergamasco received MSc degree (with honors) in Computer Science from Ca'Foscari University of Venice, Venice, Italy, in 2011 and is currently a PhD candidate at the University Of Venice. His research interests are in the area of computer vision, spreading from 3D reconstruction, Game-Theoretical approaches for matching and clustering, structure from motion, augmented reality and photogrammetry. He has been involved in many commercial computer vision projects for industries and entertainment, including structured light-scanner solutions, pipe measurement system for automotive, interactive vision-based museum exhibitions and AR applications for embedded devices.

Andrea Albarelli received the PhD degree in Computer Science from the "Ca' Foscari" University of Venice in 2010, and is currently a PostDoc researcher in the same institution. His main research area is within the field of computer vision and he has published around 40 technical papers in referred international journals and conferences, mainly related to the topics of Game-Theoretical approaches for solving the matching problems and 3D data acquisition and processing. During the last academic years, he taught Information Theory and Digital Image Processing classes for both University of Padua and University of Venice. He is also involved in several technological transfer projects and he is co-founder of an academic spin-off that offers vision-based industrial solutions.

Andrea Torsello received his PhD in computer Science at the University of York, UK and is currently working as Assistant Professor at University Ca' Foscari Venice, Italy. His research interests are in the areas of computer vision and pattern recognition, in particular, the interplay between stochastic and structural approaches as well as Game-Theoretic models, with applications in 3D reconstruction and recognition. Dr. Torsello has published more than 80 technical papers in refereed journals and conference proceedings and has been in the program committees of numerous international conferences and workshops. In 2011 he has been recognized as "Distinguished alumnus" by the University of York, UK. He held the position of chairman and is currently the vice-chair of the Technical Committee 15 of the International Association for Pattern Recognition, a technical committee devoted to the promotion of research on Graph-based Representations in Pattern Recognition.