Low-latency Vision-based Fiducial Detection and Localisation For Object Tracking

Ansu Man Singh[1], Quang P. Ha[1], David K. Wood[2], Mark Bishop[2]

[1.]Faculty of Engineering and Information Technology, University of Technology Sydney, Australia
[2.] Ocular Robotics Pty Ltd, NSW, Australia

Abstract

Real-time vision systems are widely-used in construction and manufacturing industries. A significant proportion of computational resources of such systems is used in fiducial identification and localisation for motion tracking of moving targets. The requirement is to localise a pattern in an image captured by the vision system precisely, accurately, and with a minimum available computation time. As such, this paper presents a class of patterns and, accordingly, proposes an algorithm to fulfil the requirement. Here, the patterns are designed using circular patches of concentric circles to increase the probability of detection and reduce cases of false detection. In the detection algorithm, the image captured by the vision system is first scaled down for computationally-effective processing. The scaled image is then separated by filtering only the colour components, which are made up of outer circular patches in the proposed pattern. A blob detection algorithm is then implemented for identifying inner circular patches. The inner circles are then localised in the image by using the colour information obtained. Finally, the localised pattern, along with the camera and distortion matrix of the vision system, is applied in a perspective-n-point solving algorithm to estimate the marker orientation and position in the global coordinate system. Our system shows significant enhancement in performance of fiducial detection and identification and achieves the required latency of less than ten milliseconds. Thus, it can be used for infrastructure monitoring in many applications that involve high-speed real-time vision systems.

Keywords – Fiducial tracking, marker detection

1   Introduction

Vision-based systems have numerous applications in robotics and automation systems used in industry, including construction and manufacturing. For instance, in [1], a vision-based system is presented for 3D terrain surface reconstruction of a construction site by using stereo cameras. The vision systems can also be integrated with unmanned vehicles to monitor and inspect a disaster-prone location, as reported in [2].

One of the purposes of the vision-based systems is to localise and track moving targets, for example, for colour tracking of a multiple robot system [3], or monitoring of equipment in construction and surface mining [4]. To this end, markers are widely used in these systems to precisely estimate the position and orientation of moving targets. In construction, such systems can be helpful in estimaton of the pose and position of the end effector of excavators to analyse the workflow during the earthmoving process [5]. Furthermore, the developed fiducial system will be used in thermal stress analysis (TSA) of vibrating mechanical systems using the vision camera, thermal sensors and the state-of-the-art optical sensor pointing technology available at Ocular such as the Robot Eye [6] shown in Figure 1.

One of the challenges in fiducial tracking is the least computation time required to accurately detect and localise the marker. This is especially important in high-speed applications where computational resources are used for other critical processes as well, such as control and management. In these scenarios, if too much computation time is spent in fiducial localisation, then it may affect the overall performance of the system.

To address the problem mentioned, we present a marker system and the method to detect the markers with a latency for detection of less than 10 ms. The proposed marker consists of concentric circles located at four corners of a square. The purpose of the outer circles is to reduce the possibility of false detection of the inner circles. In the detection algorithm, computational efficiency is achieved by downscaling the images captured by the vision sensor. Then, to detect the inner circles we segment the scaled image into binary images which represent only the colour components made up of the outer circles. The inner circles are then detected and localised in the binary images. Finally, to estimate the position and orientation of the marker in the 3D space, we use the Levenberg-Marquardt optimisation algorithm iteratively.

The organisation of this paper is as follows. First, the paper discusses the fiducial design in the Section 2,

followed by, the description of the algorithm for its detection and localisation, Section 3. Then, results of the comparative study of our marker system with standard markers are presented in Section 4. Finally, the paper concludes with a conclusion.

Figure 1. Ocular Robot Eye

2   Fiducial Design

There are many marker systems that make uses of circular patches. For instance in [7], colour-coded circular markers are arranged in various shapes such as circle, triangle, square, etc. However, such a marker system can have a higher possibility of false detection if the circles outside the region of interest are incorrectly detected. A typical pattern with circular patches is presented in Figure 2, where the circle is detected outside the region of interest. This may reduce the performance of a detection algorithm for single-circle markers.
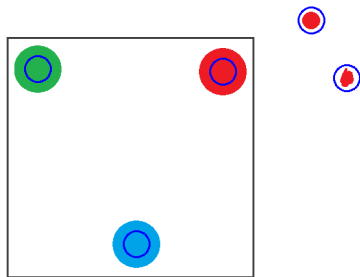

Figure 2: False detection of circles outside the marker region

To overcome the drawback of a fiducial design with single circular patches mentioned above, we propose to use markers with concentric circles, where the outer circles are made up of two distinct colours, blue and green in this case. The proposed marker is shown in Figure 3, where the rationale for its choice is that if the inner circles can be searched for by using only the pixels consisting of the colour of outer circles, then the incorrect detection of circles outside the pattern region of

interest is significantly reduced. Notably, the circle at the center of the pattern is not used in the detection algorithm. Nevertheless, this circle can have a special usage in tracking the thermal targets in motion. Particularly, the pattern can be attached to the target with the area of interest located within the central circle.
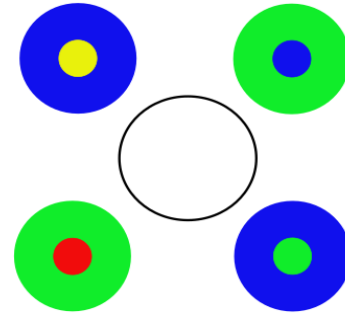

Figure 3. Proposed marker

3   Marker Detection and Pose Estimation

In the following section, we present the process of the detection and localisation of the inner circles of the marker, followed by, the procedure for estimation of its position in 3D space.

3.1   Image segmentation

In this step, the images captured by the camera is first scaled down by a factor of 4:1. The scaling effect reduces the computational latency of the detection algorithm. After scaling, two binary images are produced to represent the presence or absence of colour components of the outer circles. In order to obtain the binary images, we apply a threshold on every pixel of the scaled image. The threshold depends on the conventional Hue-Saturation-Value (HSV) model of the colour of the outer circles, where H corresponds a pure colour with H=0 refered to Red, S describes the whiteness with S=0 for White, and V is for darkness with V=0 for Black. Figure 4 represents the segmentation step.

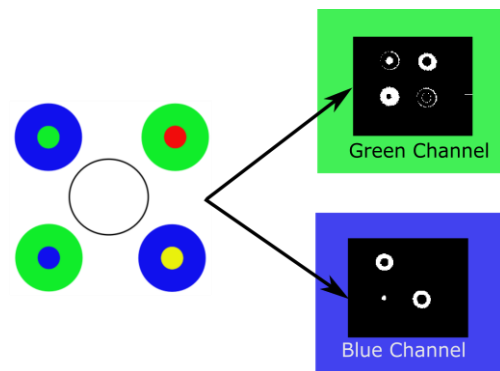Figure 4: Segmentation of captured image into binary images

3.2 Inner circle detection

In order to detect the inner circles two methods can be applied here, namely the Hough detection method and blob detection method.

3.2.1 Hough circle detection

The Hough detection method uses the circular Hough transform to find circles in an image. If an image contains many points, some of which fall on perimeters of circles, then the search objective is to find parameter triplets (a, b, r) to describe each circle of centre (a, b) and radius R:

$$\begin{cases} x = a + r\cos\theta \\ y = b + r\sin\theta, \end{cases}$$

where angle θ sweeps through the full 360 degree range that the points (x, y) trace the perimeter of a 2D circle. The fact that the parameter space is 3D makes a direct implementation of the Hough circle technique (HCT) more expensive in computer memory and time. Indeed, the fiducial marker, however, goes through transformations, i.e. rotational and translation which produces an ellipse in the image plane of the camera. In this scenario, the Hough detection method becomes ineffective. This is clearly demonstrated in Figure 5(a), where the algorithm fails to detect inner circles of the fiducial. In addition, for better performance, HCT requires filtering for smoothing, which further increases the computational load. Thus, for the inner circle detection in the marker, this method is not a suitable.
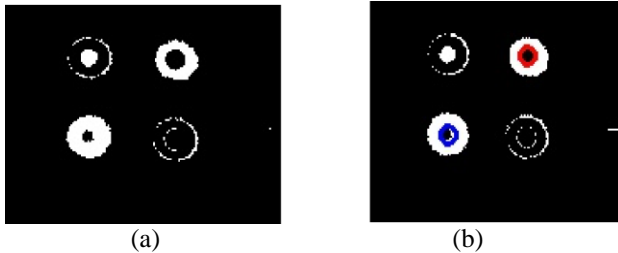

(a)	(b)

Figure 5: Comparison of (a) Hough circle detection, and (b) blob detection

3.2.2 Blob detection algorithm

Based on the general concept that blobs of an image are meaningful regions to interpret its main features. Here, the blob detection algorithm (BDA) is based on the contour detection for extracting distinct regions from the background. This algorithm first detects a contour in the provided image, which is not necessary a circle but can be elliptical patches. Then, the BDA connects the overlapped contours and may be rotated in 3D. As a result, this method is suitable for inner circle

identification of the marker. The detection of the inner circles of the marker by the blob detection method is presented in Figure 5 (b), where the inner circles present more meaningful features. Similarly, the identification of the inner coloured circles using the blob detection algorithm is presented in Figure 6.
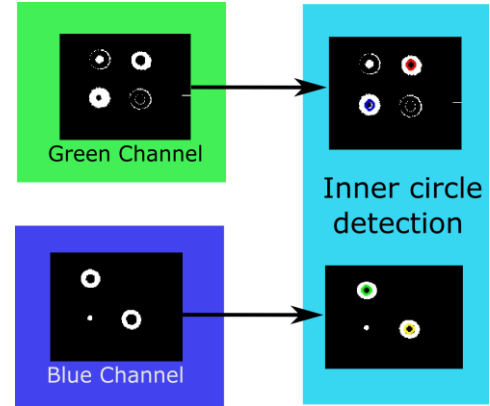

Figure 6: Inner circle detection using the blob detection algorithm

3.3 Circle localisation and pose estimation

After detecting circles in the binary images, the inner circles are identified by checking the H-S-V value of the center of the circles in the original image. The inner circles are identified in the image plane if the H-S-V values lies within the pre-defined colour threshold for the inner circles. In other words, suppose x_i is the centre of a detected circle in the binary images and $h() : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ is the function mapping a pixel to the H-S-V value, then the detected centre is identified as the kth inner circle if the condition $\lambda_{k(min)} \leq h(x_i) \leq \lambda_{k(max)}$ (k =1,2,3,4) is satisfied, where and $\lambda_{k(\min)}$ and $\lambda_{k(max)}$ are the minimum and maximum colour threshold for the kth inner circle, respectively.

After the localisation of the inner circles in the image plane, the orientation and position of the marker are estimated by solving the Levenberg-Marquardt optimisation for the non-linear least square minimisation problem:

$$\arg\min_j \sum_{j=1}^{N} d(\mathbf{x}_j, \mathbf{P}\mathbf{X}_j)^2,$$

where $\mathbf{x}_j$ and $\mathbf{X}_j$ are respectively the image point in the image plane and the scene point in the world coordinate system, and $d(\mathbf{x},\mathbf{y})$ is the Euclidean distance between two vectors $\mathbf{x}$ and $\mathbf{y}$, and $\mathbf{P}$ is the camera projection matrix. The projection matrix $\mathbf{P}$ is given by

$$\mathbf{P} = \mathbf{KRt} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \mathbf{Rt} \,,$$

where f_x and f_y are respectively the focal length of the camera in x and y direction, c_x and c_y are the principal axes, $\mathbf{R}$ is the rotational matrix and $\mathbf{t}$ is the translational vector of the camera with respect to the pattern coordinate system. The camera matrix $\mathbf{K}$ is obtained from the camera calibration and should be considered in the optimisation, as described in OpenCV [8]. The library of OpenCV provides the interfaces for the optimisation. The solver solves for $\mathbf{R}$ and $\mathbf{t}$ after the recursive optimisation. Figure 7 summarises the processes of marker detection and pose estimation.
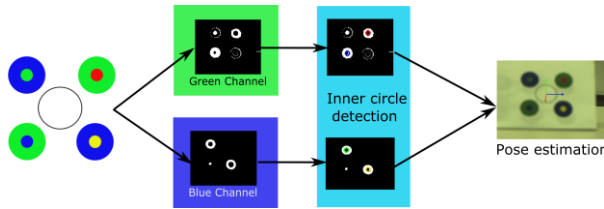

Figure 7: Orientation and position estimation of the fiducial marker in 3D space

The flowchart depicted in Figure 8 illustrates the marker detection procedure. Even though colour markers are used in our fiducial system, they are detected and localised in binary images in order to consume less computational load. In addition, the images are also scaled before the detection method is applied. As a result of these operations, the expected low latency can be achieved for detection of markers.

4 Experiment and results

In this section, we present the benchmarking of our proposed marker detection system with standard marker detection systems, such as the Aruco and AprilTags markers.

4.1 Aruco Marker system

Aruco marker system uses the binary-coded markers, as shown in Figure 9. To correct the binary codes in the detected image, hamming codes can be used as reported in [9]. The detection algorithm for a marker system consist of three stages, namely (i) image segmentation, (ii) contour extraction and (iii) marker identification. The Aruco marker is one of the standard marker systems for camera calibration. Its interfaces are included in the OpenCV 3.1.0 library [8].
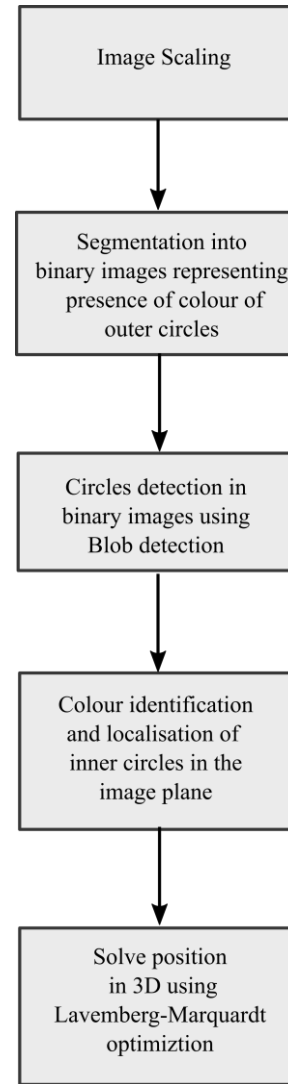

Figure 8: Flowchart for the marker detection procedure

4.2 AprilTags marker system

AprilTags is also one of the widely used marker systems. A marker of Apriltags consists of binary codes, as depicted in Figure 10. In the detection algorithms of the marker system: first, the line segments are detected, followed by quad detection, and finally, the binary codes are extracted and corrected [10].

The size of markers used in the experiment is listed in Table 1 below.

Table 1. Marker size used in experiment

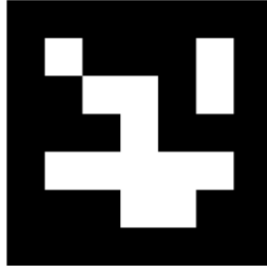| Marker	Size
Aruco	5.5×5.5 cm
AprilTags	5×5 cm
Proposed marker	5×5 cm

Figure 9: Aruco marker

Figure 10: AprilTag marker

In our experiments, a high frame rate camera was used to take images of the markers. The camera was operated at 100 frames per seconds, and the depth of field for the camera was set to 50 cm with the lower limit of 50 cm and the upper limit of 100 cm. The specifications of the camera are listed in Table 2. Similarly, the algorithms for markers detection were tested in a PC with Intel ® Core ™ i7 CPU operating at the clock frequency of 3.4 GHz, and 16 GB RAM.

Table 2. Specifications of camera

Resolution	658×492
Focal length	16 mm
Aperture	F/2-22

The experiments were performed by varying the distance of the markers from the camera. The markers were placed at the distance of 40 cm, 50 cm, 90 cm, and 120 cm, respectively.

The results of the experiments are presented in Figures 11 and 12, respectively. Figure 11 shows the detection ratio of the fiducial systems. The detection ratio for the individual marker systems is calculated by taking the ratio of the number of detected frames over total frames during an experiment. From the figure it is clear that the detection ratio for our proposed marker system and the Aruco marker system is better than AprilTags, when the markers are placed from 40 cm to 90 cm from the camera. However, the performance of the detection algorithms decreases when the markers are placed beyond the far limit of the depth of field of camera, i.e. 100 cm. Even in this case, the detection ratio of the proposed system is still better than Aruco markers.
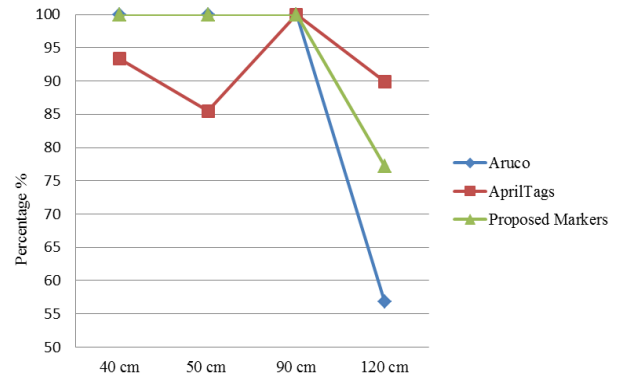

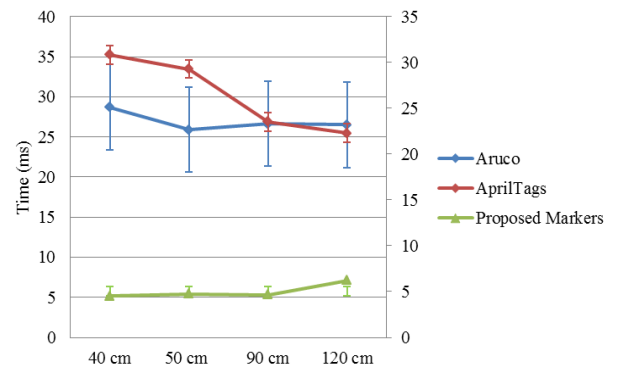Figure 11: Detection rate at various distance from camera

Figure 12: Average computation time for all markers

Figure 12 shows the average computation time for all the fiducial tracking systems. The computational time is evaluated in milliseconds while the vertical bars in the graph shows the standard deviation in the marker detection time. It is evident from the figure that the proposed marker detection system performs much better compared to the other standard markers. For instance, the detection time for the proposed algorithm is 5 ms, compared to 35 ms and 28 ms for the Apriltags and Aruco marker systems, respectively. In addition, the standard deviation for the computation time is also small compared to other standard methods. Although the computation time increases slightly when the marker is beyond the upper limit of the depth of field of the camera, i.e. 100 cm from the camera, the low latency in terms of computation time is still upheld. Notably, the computation time for the Aruco marker is less than that of the AprilTags, but the standard deviation is much greater. From this experiment, it can be concluded that our proposed marker detection system can detect and localise marker in less than 10 ms to meet the requirement of high-speed real-time vision-based systems .

Conclusion

This paper presents a fiducial tracking system which is designed to track moving targets. The proposed fiducial in our tracking system consists of concentric circular patches. A comparison study of our proposed system with other standard marker detection systems shows that our method can detect the marker accurately and have much lower computational latency compared to the standard markers. Our system can detect fiducials in less than 10 milliseconds which make it suitable for high-speed applications such as moving object tracking.

Acknowledgement

References

[1] Sung, C., Lee, S. H., Kwon, Y. M., & Kim, P. Y. (2016, January). Fast and Robust 3D Terrain Surface Reconstruction of Construction Site Using Stereo Camera. In *The 33rd International Symposium on Automation and Robotics in Construction*, pages 19-27, Auburn, USA, 2016.

[2] Wen, M. C., Yang, C. H., Chen, Y., Sung, E. X., & Kang, S. C. A Stereo Vision-Based Support System for Tele-Operation of Unmanned Vehicle. In *The 33rd International Symposium on Automation and Robotics in Construction*, pages 19-27, Auburn, USA, 2016.

[3] Yu, Y. -H., Kwok, N. M., and Ha, Q. P. Color tracking for multiple robot control using a system-on-programmable-chip. *Automation in Construction*, 20(6): 669 – 676, 2011.

[4] Azar, E. R. Active Control of a Pan-Tilt-Zoom Camera for Vision-Based Monitoring of Equipment in Construction and Surface Mining Jobsites. In *The 33rd International Symposium on Automation and Robotics in Construction*, pages 121-131, Auburn, USA, 2016.

[5] Feng, C., Dong, S., Lundeen, K. M., Xiao, Y., & Kamat, V. R. (2015, January). Vision-based articulated machine pose estimation for excavation monitoring and guidance. In *The 32rd International Symposium on Automation and Robotics in Construction*. pages 1-9, Oulu, Finland, 2015.

[6] http://www.ocularrobotics.com/technology/

[7] Bruce, J., & Veloso, M. Fast and accurate vision-based pattern detection and identification. In *The 2003 IEEE International Conference on Robotics and Automation*, pages 1277-1282, Taipei, Taiwan, 2003.

[8] http://docs.opencv.org/2.4/doc/tutorials/calib3d/camera_calibration/camera_calibration.html

[9] Garrido-Jurado, S., Muñoz-Salinas, R., Madrid-Cuevas, F. J., & Marín-Jiménez, M. J. (2014). Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6): 2280-2292, 2014.

[10] Olson, E. AprilTag: A robust and flexible visual fiducial system. In *The 2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3400-3407, Sanghai, China, 2011.