General Robot-Camera Synchronization Based on Reprojection Error Minimization

Kenji Koide[1] and Emanuele Menegatti[1]

Abstract— **This paper describes a synchronization method to estimate the time offset between a robot arm and a camera mounted on the robot (i.e., robot-camera synchronization) based on reprojection error minimization. In this method, we detect a calibration pattern (e.g., checkerboard) from camera images while projecting the pattern onto the image space with robot hand poses and forward kinematics. Then, we estimate the delay of the camera data by finding the robot-camera time offset which minimizes the reprojection error between the visually detected and the projected patterns. Since the proposed method does not rely on any camera-specific algorithms, it can be easily applied to any new camera models, such as RGB, infrared, and X-ray cameras, by changing only the projection model. Through experiments on a real system, we confirmed that the proposed method shows a good synchronization accuracy and contributes to the accuracy of a continuous scan data mapping task.**

I. Introduction

Vision based inspection systems are widely considered for industrial applications [1]. To respond to the increasing demand for rapid and efficient production, a lot of automated vision-based inspection systems have been exploited for real systems. However, to our knowledge, most of the existing inspection systems are highly dependent on specific production systems, and their reusability is very limited.

SPIRIT [1], an industrial robot project, aims to develop a "general" inspection robot framework. In this framework, the product to be inspected, the robot, and the camera for inspection can be easily replaced with ones for new inspection tasks. For this purpose, all the components which compose the framework have to be independent of the specific product, robot, and camera model. In particular, the generality with respect to the camera model is important. The framework has to be able to handle various imaging sensors in a unified system, to name a few: RGB-D, thermographic, and X-ray cameras.

Robot-Camera synchronization is one of the essential tasks for visual inspection systems. Usually, there is a delay on images acquired with a camera due to encoding, decoding, and buffered communication [2]. In scenarios where a robot performs a continuous scan motion while mapping images in the robot space, the synchronization accuracy has a significant impact on the final mapping accuracy. If the camera is not synchronized with the robot, we cannot refer the correct robot pose at the moment an image was acquired, and cannot map the image in the correct position in the robot space.

Hardware-based synchronization is the most reliable and accurate way to synchronize two or more devices. Typically, to synchronize data from multiple devices, the data acquisition of each device is triggered by using an external hardware signal [3]. With such a hardware mechanism, data from different devices can be synchronized in the order of 100 nanoseconds [4]. However, hardware synchronization mechanisms require the devices to have a hardware interface and a special data acquisition mode for triggering the data acquisition using the external signal. In case devices (in our case, camera and robot) do not have such synchronization interfaces, we need a software-based method, which uses only data acquired by the devices, to synchronize them.

Several methods for software-based robot-camera synchronization have been proposed [5], [6], [7]. They first detect a calibration pattern using the camera while moving the robot hand where the camera is mounted, and then estimate the camera poses (i.e., camera motion) with respect to the calibration pattern. The time delay between the camera and robot motion sequences is estimated using, for instance, cross-correlation techniques [6]. There are also several techniques to perform spatial and temporal calibration simultaneously [8]. The problem here is that we need to explicitly estimate the camera pose with respect to the calibration pattern for each input image. For pinhole cameras, typically a PnP algorithm [9] is used to estimate the camera pose from an image. However, in the case with a non-pinhole camera, usual PnP algorithms cannot be used to estimate the camera pose. For instance, with some special camera models (e.g., X-ray source-detector camera model), it is not possible to estimate the camera pose from a single image.

The idea to use the reprojection error term for robot-camera calibration is introduced by [10]. They estimate the robot-camera transformation by minimizing the reprojection error. Since this method does not require to explicitly estimate the camera pose with respect to the pattern for each image, it does not rely on any camera model-specific algorithms, and it can be applied to different camera models by changing the projection model. It has been shown that the reprojection error minimization-based method works well on both pinhole and source-detector camera models on real systems with RGB and X-ray cameras. Following [10], in this work, we introduce a reprojection error minimization-based method to achieve a general robot-camera synchronization method.

In this paper, we propose a general robot-camera synchro-

*This work was supported by EU Horizon 2020 grant No. 779431, the SPIRIT project.

[1]Kenji Koide and Emanuele Menegatti are with the Department of Information Engineering, University of Padova, via Gradenigo 6/B, 35131, Padova, Italy, koide@dei.unipd.it

[1]http://spirit-h2020.eu/

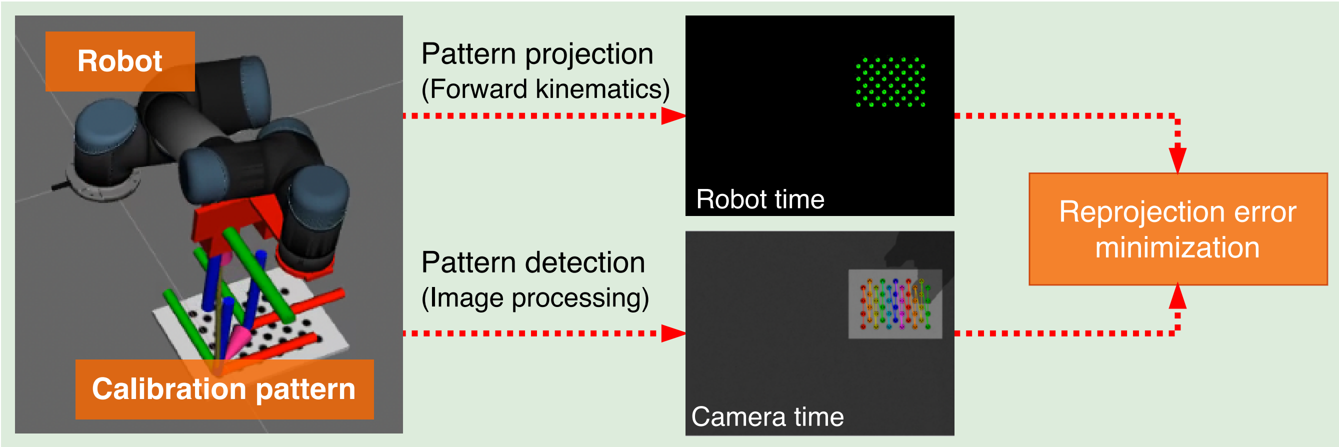


Fig. 1: The proposed robot-camera synchronization method.

nization method based on reprojection error minimization (see Fig. 1). We first collect a sequence of robot hand poses and a sequence of images of a calibration pattern (e.g., checkerboard). Then, we synchronize the robot and the camera by finding the time offset which minimizes the reprojection error between the visually observed calibration patterns and the patterns reprojected with the robot hand poses and forward kinematics. Since the proposed method does not rely on any camera-specific algorithms, like PnP algorithm, it can be applied to any camera models by changing only the projection model.

II. Methodology

Fig. 1 shows an overview of the proposed method. We estimate the time offset Δt by comparing sequences of non-synchronized robot hand poses and camera images. To compare them, we first move the robot hand along a certain path (e.g., sine curve above the calibration pattern), and record a sequence of robot hand poses and a sequence of images of the calibration pattern. Let $\mathcal{R}$ be the robot hand pose sequence, and $\mathcal{I}$ be the calibration pattern image sequence. Let us assume that we collected N images and K hand poses, and the calibration pattern consists of M points.

Let $\hat{\boldsymbol{p}}_t^i$ be the i-th point of the calibration pattern detected from the j-th image with timestamp $\hat{t}$ (in the camera time). Given a robot-camera time offset Δt, we calculate the robot hand pose $\boldsymbol{R}_t$ at the corresponding robot time $t = \hat{t} + \Delta t$ by interpolating the discrete robot hand poses $\mathcal{R}$. We use Slerp (Spherical linear interpolation) [11] to interpolate the robot hand poses:

$$\boldsymbol{R}_t = Slerp(\mathcal{R}, t). \tag{1}$$

With the interpolated robot hand pose $\boldsymbol{R}_t$, we project each point of the calibration pattern $\boldsymbol{p}^i$ into the camera space.

$$\boldsymbol{p}_t^i = Proj(\boldsymbol{R}_t, \boldsymbol{p}^i), \tag{2}$$

where, $\boldsymbol{p}_t^i$ is the projected i-th point of the pattern at robot time t. We can use any projection function suitable to describe the camera. For instance, we use a pinhole camera model for RGB cameras [12], and a source-detector camera model for X-ray cameras [13].

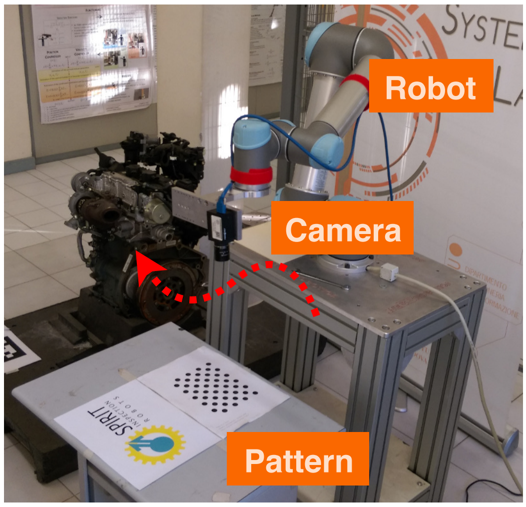


Fig. 2: A snapshot of the synchronization experiment.

The reprojection error is defined by:

$$E(\Delta t) = \sum_{\hat{t}}^{N} \sum_{i}^{M} \|\hat{p}_t^i - Proj(Slerp(\mathcal{R}, \hat{t} + \Delta t), \boldsymbol{p}^i)\|. \tag{3}$$

We estimate the robot-camera time offset $\Delta \tilde{t}$ which minimizes the reprojection error:

$$\Delta \tilde{t} = \arg \min_{\Delta t} E(\Delta t). \tag{4}$$

In this work, we find the optimal time offset $\Delta \tilde{t}$ using exhaustive search in the range $[-0.2s, 0.2s]$. Since the reprojection error shows a good convexity as shown in Fig. 3, this search process can be improved with a line search method [14].

III. Experiments

A. Robot camera synchronization

To validate the proposed method, we conducted an experiment on a real system. Fig. 2 shows a snapshot of the

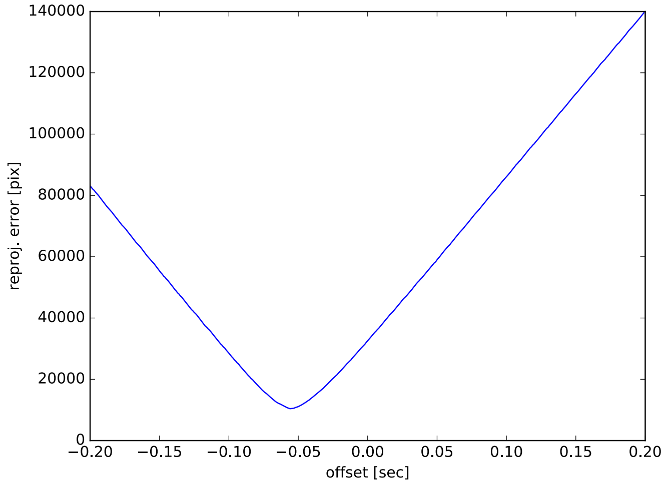


Fig. 3: The reprojection error with different time offset Δt. The reprojection error is minimized at $\Delta t = -0.056$ [sec].

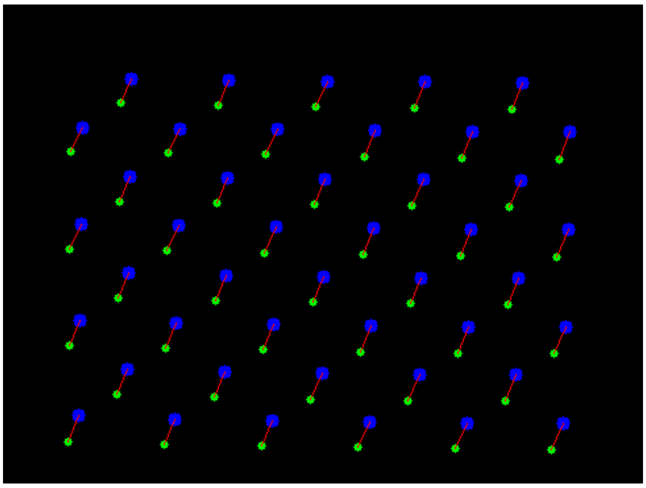
experimental environment. We mounted a camera (Pointgrey Flea3) on a robot arm (Universal robot UR5), and placed a calibration pattern so that the camera can see it. The robot-camera transformation and the calibration pattern pose are estimated with an automatic calibration technique [15]. We moved the robot along a sine curve above the calibration pattern (width=0.4[m], height=0.3[m], altitude=0.5[m]), and recorded images and robot hand poses while the robot was moving. We fed these sequences of images and robot hand poses to the proposed method and estimated the time offset between the camera time and the robot time.

Fig. 3 shows the plot of the reprojection errors E versus the time offset Δt. We can see that the reprojection error is minimized at the point $\Delta t = -0.056$ [sec]. This means that the images (camera time) are delayed from the robot hand poses (robot time) 0.056 [sec], and by adding this offset to the camera timestamp, we can obtain the corresponding timestamp in the robot time.

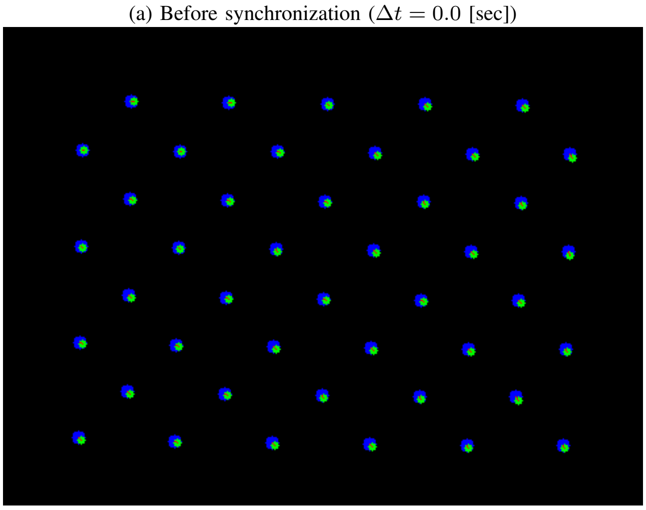
Fig. 4 shows the reprojected and the visually observed patterns at the frame when the robot was moving at the maximum speed ($\sin(\text{phase} = \pi)$, about 0.25 [m/sec]). The green points are the reprojected pattern with the robot hand pose and forward kinematics, and the blue points are the pattern visually detected from the image. Before synchronization, the pattern reprojected based on the robot hand are obviously precede the visually detected pattern. With the estimated time offset, the reprojected pattern is well synchronized with the visually detected pattern, and we observe very small reprojection errors between them.

B. Continuous scan data mapping

To demonstrate that the camera and the robot are well synchronized with the proposed method, we conducted a continuous scan data mapping experiment. With the same setting as Sec. III-A, we recorded images and robot hand poses. We estimate the plane of the calibration pattern from the very first frame of the images, and then project all the images acquired with the continuous robot motion onto the



(a) Before synchronization ($\Delta t = 0.0$ [sec])



(b) After synchronization ($\Delta t = -0.056$ [sec])

Fig. 4: The visually observed (blue) and the projected (green) points before and after the synchronization. The robot was moving at about 0.25 [m/sec]

calibration pattern plane with the robot hand poses (without any image processing, such as image stitching and alignment). Fig. 6 shows the images projected and accumulated on the calibration pattern plane. 47 images are accumulated in total. Without synchronization, the positions of the projected images deviate due to the delay of the camera images, and as a result, the accumulation image is blurred. With the estimated time offset, the images are synchronized with the robot hand poses and precisely projected on the plane, and we observe a clear accumulation image with the continuous scan.

IV. Conclusions

This paper proposed a robot-camera synchronization method which estimates the robot-camera time offset by minimizing the reprojection error. Since the proposed method does not rely on any camera-specific algorithms, it can be applied to any imaging sensors by changing only the projection model. The experimental results show that the proposed method can be applied to real systems, and it

(a) (b) (c) (d) (e)

Fig. 5: The image sequence recorded while the robot was drawing a sine curve above the calibration pattern.

(a) Before synchronization ($\Delta t = 0.0$ [sec]) (b) After synchronization ($\Delta t = -0.056$ [sec])

Fig. 6: The continuous scan data mapping results. 47 images are accumulated in total.

contributes to the accuracy of continuous scan data mapping tasks.

We are planning to apply the proposed method to real X-ray imaging systems to show that the method can be applied to new imaging sensors with a small modification of the projection model. We are also planning to conduct further assessment of the synchronization accuracy, and how it affects the continuous scan data mapping accuracy.

ACKNOWLEDGMENT

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 779431.

REFERENCES

[1] J. Ghosh, *Vision based inspection*. Springer Netherlands, 1994.

[2] "Latency in live network video surveillance," Axis communications, Tech. Rep., 2015. [Online]. Available: https://www.axis.com/files/whitepaper/wp_latency_live_netvid_63380_external_en_1504_lo.pdf

[3] A. Vibeck, "Synchronization of a multi camera system," 2015.

[4] P. Loschmidt, R. Exel, A. Nagy, and G. Gaderer, "Limits of synchronization accuracy using hardware support in IEEE 1588," in *IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication*. IEEE, sep 2008.

[5] D. Rosen, L. Carlone, A. Bandeira, and J. Leonard, "A certifiably correct algorithm for synchronization over the special Euclidean group," in *Intl. Workshop on the Algorithmic Foundations of Robotics*, San Francisco, CA, Dec. 2016.

[6] K. Pachtrachai, F. Vasconcelos, G. Dwyer, V. Pawar, S. Hailes, and D. Stoyanov, "CHESS—calibrating the hand-eye matrix with screw constraints and synchronization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2000–2007, jul 2018.

[7] F. Furrer, M. Fehr, T. Novkovic, H. Sommer, I. Gilitschenski, and R. Siegwart, "Evaluation of combined time-offset estimation and hand-eye calibration on robotic datasets," in *Field and Service Robotics*. Springer, 2018, pp. 145–159.

[8] P. Furgale, J. Rehder, and R. Siegwart, "Unified temporal and spatial calibration for multi-sensor systems," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, nov 2013.

[9] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o(n) solution to the pnp problem," *International Journal of Computer Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[10] K. Koide and E. Menegatti, "General hand–eye calibration based on reprojection error minimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1021–1028, apr 2019.

[11] K. Shoemake, "Animating rotation with quaternion curves," 1985.

[12] D. A. Forsyth and J. Ponce, "A modern approach," *Computer vision: a modern approach*, vol. 17, pp. 21–48, 2003.

[13] M. Mitschke and N. Navab, "Recovering the x-ray projection geometry for three-dimensional tomographic reconstruction with additional sensors: Attached camera versus external navigation system," *Medical Image Analysis*, vol. 7, no. 1, pp. 65–78, 2003.

[14] M. J. B. D. D. W. H. Swann, *"Non-linear optimization techniques"*. Oliver & Boyd,, 1969.

[15] M. Antonello, A. Gobbi, S. Michieletto, S. Ghidoni, and E. Menegatti, "A fully automatic hand-eye calibration system," in *European Conference on Mobile Robots*. IEEE, 2017, pp. 1–6.