Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 50-1 (2017) 331–336

Tracking Algorithms for Cooperative Telemaintenance Repair Operations ⋆

Michael Rojkov * **Doris Aschenbrenner** * **Klaus Schilling** ** **Marc Latoschik** ***

* *Zentrum für Telematik e.V. (ZfT) 97070 Würzburg (Germany) (Tel: +49 (0)931 615633-63; e-mail: name.surname@telematik-zentrum.de).*
** *University of Würzburg Computer science Chair Robotics and Telematics 97070 Würzburg (e-mail: schi@informatik.uni-wuerzburg.de)*
*** *University of Würzburg HCI Group 97070 Würzburg (e-mail: marc.latoschik@uni-wuerzburg.de)*

Abstract: This paper broaches the issue of using 2D tracking algorithm frameworks on mobile devices in order to enable cooperative repair tasks in robot-operated industrial production. A local service technician can be supervised remotely by an external expert in the context of telemaintenance. At first we introduce several tracking libraries and test them on low-performance mobile devices. After that we conduct a test run on a large amount of pictures derived from video streams with different characteristics of the application area of industrial maintenance and repair operations in order to find the best suited algorithm. At the end we describe the implementation on several mobile devices and delay considerations.

Keywords: Telematics, Telemaintenance, Human-machine interface, Human-centered design, User interfaces, Computer vision, Computer-aided work, Computer-aided instruction, Computer-aided diagnosis, Industrial robots

1. INTRODUCTION

Complicated production lines with industrial robots and the surrounding plant components are in need of being remotely accessible. This is especially critical if an unexpected problem cannot be solved by the local specialists. Special machine manufacturers, especially robotics companies, sell their technology worldwide. Some factories, for example in emerging economies, lack qualified personnel for repair and maintenance tasks. When a severe failure occurs, an expert of the manufacturer needs to be sent on siteon site, which leads to long down times of the machine or even the whole production line. With the development of data networks, a huge part of those travels can be omitted, if appropriate telemaintenance equipment is provided.

Previous research on cooperative repair by Aschenbrenner et al. (2016) and teleoperation in an active production line, presented in Aschenbrenner and et al. (2015), have strongly indicated further need for the investigation of video interaction methods.

Figure 1 displays the concept of the "video interaction" principle. The communication application should provide a possibility for service technicians and experts to build a mutual base of knowledge and to exchange information about the repair task. This is often referred to as shared view and the underlying psychological concept is "common ground" introduced by Clark and Carlson (1981).

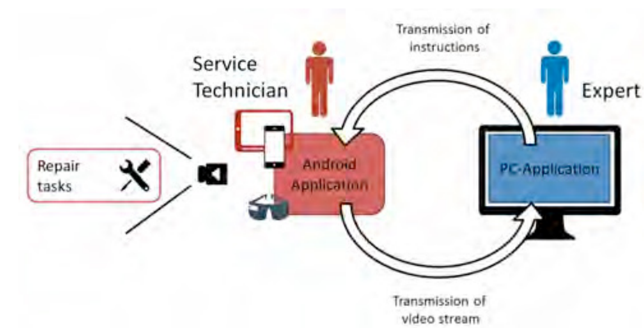

Fig. 1. Concept of video interaction

Our specific applications are external supervised maintenance and repair operations on an industrial manipulator. A mobile device is handed to a service technician who has to execute a repair task. This can either be a tablet pc, a smartphone, or AR glasses. The service technician is able to record his working environment with the installed camera in the mobile device. These recordings then are transferred to the expert's desktop application via the network. The remote expert can draw instructions like lines and circles into the video frame by using the mouse. These repair instructions are then sent to the service technician's application via the network and are displayed on his or her mobile device.

The display of the repair instructions can be supported by the use of Computer Vision, specifically tracking algorithms. The idea was to select objects in the video frame and track them. Tracked objects serve as supporting points

⋆ funded by the Bavarian Ministry of Economic Affairs, Infrastructure, Transport and Technology in its R&D program 'Bayern digital'.

10.1016/j.ifacol.2017.08.058

for the display of repair instructions. These are shifted relatively to the chosen supporting points and are thereby always displayed at the correct position within the frame.

This paper broaches the issue of using Computer Vision tracking algorithms for a remotely supervised repair operation. Different tracking frameworks and their algorithms will be compared and tested on off-the-shelf mobile devices along common metrics.

2. RELATED WORK

The first studies in the area of collaborative work provided by Flor (1998) investigated the conversations and information exchange of two workers who worked side by side on a repair task. The identification of goal objects, the instruction for tasks and the confirmation of finished tasks were evaluated. Further observation studies from Tang (1991) introduced a common work field, which can be used for sketches and writings. It has been shown that this common view helped to support the work process and to convey information. For a remote constellation, research of Kuzuoka (1992) showed, that collaborative groups on the same workplace performed better than spatially separated coworkers, but also showed the potential of multimedia systems for supporting collaborative work.

Another study focused on the question, which kind of visual information provides a benefit for a coworking team. Kraut et al. (2003) measured the performance of a person working alone on a bicycle repair task with a group of a remote mechanic and a local worker on the same task. The experiment provided evidence for the importance of shared visual context for remote collaborative work. Further investigations by Gergle et al. (2013) on puzzle tasks showed, that shared visual context was also important for situation awareness and grounding.

3. FRAMEWORK COMPARISON

Several possibilities to implement object tracking were taken into account when developing a communication system to support cooperative telemaintenance repair operations. JMyron (Nimoy (2016)), OpenCV (Project (2016b)) and BoofCV (Project (2016a)) libraries were evaluated for the use in Java and Processing on Android applications.

We chose those libraries according to the following criteria:

- The libraries should run on a computer with Windows and on a mobile device with Android.
- To lower the programming effort, the libraries should run with Processing (Fry and Reas (2001)).
- The application should run on off-the-shelf components and still provide a sufficient resolution and frame rate.

We wanted to facilitate software development and also needed the implemented software to run without special high-performance hardware. In order to verify the criteria, for each Computer Vision library a Windows and an Android application were implemented and tested. We tested latency and the tracking quality of the applications in order to decide on framework.

This fist evaluation was performed by implementing an application running on a ASUS MeMo ME302C tablet using a 640x480 video stream as visual input. Frame-per-seconds measurements were conducted using software calculations.

3.1 JMyron

JMyron provides basic functions of color and motion tracking. It is able to detect color clusters in a picture and track them over the course of subsequent video frames. Tests with JMyron revealed that the framework was not able to perform a stable tracking of a selected object. Sudden shifts and misalignments of the selected tracking area could be frequently observed. Because of that, the functions of JMyron were deemed insufficient, and JMyron was dismissed from further development.

3.2 OpenCV

OpenCV provides many techniques for processing and recognition of visual information. In this study specifically the algorithms 'Template matching', 'Camshift' and the 'ORB-tracker' were selected for evaluation.

Template Matching The algorithm searches a picture for a pre-selected template. Every pixel in the picture is compared to the pixels of the template, trying to find an area with the best matching criteria. The application performes with an average of 1.95 fps on a picture of a size of 640x480 pixels. The algorithm delivers good tracking results and can track a selected object over a period of time. If the viewing angle of the picture is changed too much in comparison to the viewing angle of the template picture, the tracking is starting to become inaccurate.

Camshift The algorithm locates objects in a picture by analyzing the maxima of a density spread. HSV and RGB density spreads were used in the implementation. The application performed with an average of 13.56 fps on a picture of the size 640x480. The algorithm provides good results in tracking objects with a dominant single color. Objects that were multicolored were harder to locate.

ORB-Tracker The algorithm is a feature tracker, that derives keypoints from a template picture and compares them with keypoints in a search picture. In the evaluated implementation, the settings ORB-detector, ORB-descriptor and BruteforceHamming were used. The application performed with an average of 1.05 fps on a picture of the size 640x480. ORB was successfully able to track a selected object despite its rotation or visual features.

3.3 BoofCV

The third visual framework to be evaluated was BoofCV. It provides a couple of methods for visual recognition and processing. The performances of the algorithms 'Circulant', 'Mean-Shift', 'Sparce Flow' and 'TLD' were tested.

Circulant-Tracker This algorithm uses the theory of circulant matrices, discrete Fourier transform (DCF), and linear classifiers to track a target and learn its changes in appearance described in Henriques (2012). The target is assumed to be rectangular and has fixed size and location. A dense local search is performed around the most recent