Preprints of the 15th IFAC Symposium on
Information Control Problems in Manufacturing
May 11-13, 2015. Ottawa, Canada

Smartphone based video-telemetry logger for remote maintenance services

A. L. Cologni, L. Fasanotti, E. Dovere, F. Previdi, S. Bonfanti*
F. C. Owen**

* *Department of management, information and production engineering*
Università degli Studi di Bergamo, Dalmine (BG),24044 Italy
(alberto.cologni@unibg.it)
** *Mechanical Engineering Department*
California Polytechnic State University, San Luis Obispo, CA 93407 USA

Abstract: **The strong growth of mobile devices of increasing quality and data transfer rates allows the introduction of these devices in diverse fields. In recent years, with the introduction of the "Smart factory" concept, the use of MIDs (Mobile Internet Devices) in manufacturing and assembly processes is increasingly common. The use of smart devices began at the managerial level, but it is now moving down to the factory floor. This evolution unlocks a large number of high-value services: from the time reduction of normal activities to implementing augmented reality. This research project incorporates a real-world example and defines a first operational paradigm for the acquisition, transmittal, and display of multi-source information.**

The aim of this paper is to describe, with clear references to a real case, the basic principles which should guide the implementation on a mobile platform of a remote maintenance service based on a hybrid (video and telemetry) logger.

Keywords: **smart factory; smartphone; remote maintenance; human in the loop.**

1. INTRODUCTION

In today's highly globalized market, where the companies must have high productivity, high quality and low cost for their products and services, maintenance equipment has become a crucial factor to achieve these requirements, as pointed out in (Lee, Ni, Djurdjanovic, Qiu, & Liao, 2006), (Lu & Sy, 2009), (Espindola, Pereira, Henriques, & Botelho, 2010).

These topics are more relevant for vendor companies who sell industrial machines in different countries. More and more often, the vendor companies stipulate service contracts with the users; for this reason they must manage maintenance service and guarantee the maximum availability of their equipment. Moreover, the increasing needs of production flexibility and efficiency on the factory floor are causing an increase in the devices' complexity, with a widespread adoption of programmable electronics, that required highly specialized technicians to diagnose and repair a fault and to reduce down time. These skills are very difficult to find in the customer's maintenance personnel, and, often, the support or a direct intervention by vendor's technician is necessary to solve a problem.

These requirements lead to a growing adoption of advanced maintenance systems. Among this tools remote monitoring systems are playing a greater and greater role. These allow manufacturing and assembly problems on the factory floor to be solved without the on-site intervention of skilled personnel (Mori, Fujishima, Komatsu, Zhao, & Liu, 2008), (Ong & Lee, 2004), (B., 2003), (Wang, Tse, & Lee, 2007).

These kinds of remote monitoring systems are well suited for use with automated machines. They have, however, strong limitations in the case of manual or semi-automatic machines, also known as "human-in-the-loop" machines (Ermidoro, Formentin, Cologni, Previdi, & Savaresi, 2014). In these cases it is very hard to discriminate remotely between a real machine failure and one caused by misuse of the operator. In order to solve these problems, a remote monitoring system needs to take into account not only the machine behaviour but also the actions of the machine operator. An ideal methodology is to capture machine operations on a video while acquiring simultaneously data from the machine operation.

In order to implement these features, it is necessary to find a diffuse and flexible technology that allows video capture and the acquisition of signals from a network. These data can then be transmitted the global network. Today's smart devices provide a cost-effective solution to this problem. They have a wide set of communication interfaces, concentrated computational power and a friendly and well known HMI. Their flexibility allows them to be adapted to diverse manufacturing and assembly processes appropriate to the human's role with the machine (maintenance, operations, corporate oversight, etc…).

The aim of this paper is to present the development of an innovative remote monitoring system, specialized for a semi-automated machine, based on smart devices, that demonstrates all the benefits of the mobile technologies in terms of HMI, communication capabilities and context awareness, in order to increase the efficiency of service technicians during maintenance operations.

The machine involved in developing this system is a lifting system that is an assembler's helper in handling heavy or awkward parts that previously would have been mounted manually with more than one worker. An electric motor basically removes the weight of the part to be assembled. Specialized end-effectors are used for assembling various parts on an assembly line. Additionally the system was implemented on an industrial bridge crane (see Section 4).

The system will allow normal operators, who are not maintenance experts, to test and check the status of the machine during their normal work. The main technological innovation is the management and the synchronization of the two different information streams (video-audio and data).

The paper is organized as follows. In Section 2, the experimental setup of the prototype is described, with some details related to the layout and the structure of the single components. In Section 3, the synchronization problem is detailed and, a solution is presented and tested. Sections 4 and 5 are devoted to the description of the real system and final assessments.

2. SYSTEM ARCHITECTURE AND EXPERIMENTAL SETUP

The system architecture is designed with the aim of defining a standard and reusable methodology.

Figure 1 - Paradigm of use

The architecture (depicted in Figure 1) consists of the following components:

- The machine that, obviously, must have on-board, an own electronics;
- The gateway, a small (but very important) component that routes the information from the machine to the smart device (and vice versa, if needed);
- A smart device, whose purpose is:
    - To acquire the information from the gateway and, also, from all the other sensors on-board;
    - To process the data;
    - To route the information from the "field" to the global network;
    - To set-up and control the machine (when the smartphone is used to change the behaviour of the machine);
- The manager of the service that collects all the information from the devices and applies corrective actions if necessary.

The real case, described in this document, fits perfectly with the general paradigm that is defined. The layout of the system has the following characteristics. The machine is an industrial manipulator (Previdi, et al., 2010) (Previdi, Fico, Savaresi, Belloli, & Pesenti, 2012) actuated by operator command. The role of the smartphone is to acquire, simultaneously, the data stream coming from the gateway and the video, recorded directly from its camera. Each acquired log can be reviewed directly on the device or sent instead to the manufacturer of the machine.

This section of the paper is devoted to the detailed description of each component of the case study.

2.1 Machine architecture

The electronic architecture of the machine is based on three electronic boards that exchange data information through a CAN backbone, represented in the following figure.

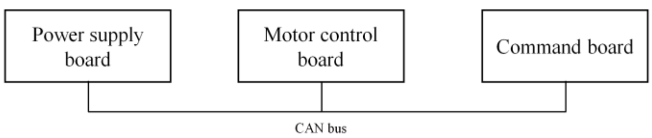

Figure 2 - Machine architecture

The power supply board manages the different power supply voltages; the motor control board controls and monitors the machine motor; the command board implements all the algorithms to implement the HMI and controls all machine operations.

2.2 Structure of the gateway

In order to develop the communication gateway, the chosen architecture is the following. The microcontroller that manages the CAN communication, interacts with a semi-autonomous radio module, based on the communication standard IEEE 802.11b / g, through a local communication channel on the board based on a UART.

The choice of using the Wi-Fi protocol for the connection with the smartphone has been implemented so as to maximize the gateway compatibility with the largest possible number of available smartphones, while ensuring a high standard of performance in the communication.

In order to improve the development and debugging of the system, the gateway has been equipped with a high number of interfaces, in addition to the CAN bus interface. Particularly the gateway has the following wired connections:

- CAN interface;
- Analogical input (0-3.3V);
- Digital input (24V);
- Digital output (24V);
- Serial communication interface (RS-232).

In terms of wireless connections, a Wi-Fi Roving Networks RN-131G/C module was chosen, directly connected to the controller through a serial port (as previously described). This module integrates a proper CPU that allows it to manage the overall TCP/IP stack. The possible configurations are:

- Infrastructure;
- Ad Hoc;
- Soft-AP.

In this work, the soft-AP mode is used.

2.3 Machine / Gateway Interface

The machine-gateway interface is configured as described in Figure 3. As can be seen, the information backbone of the machine is a CAN bus. Thus the gateway has access to the entire data stream in the machine and can acquire all the messages sent between the different boards.

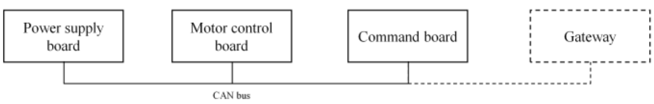

Figure 3 - Gateway connection scheme

2.4 Firmware architecture

The firmware developed for the gateway is made of three main sections for setting up, managing the different communication interfaces and linking the smartphone with the operating data.

Details related to the protocol, in particular the data packet exchanged between the gateway and the smart device, are described in Table I.

Table I - Layout of the data message

0xF0	0xF0	Var_1	Var_2	Var_3	Var_4
Var_5	Var_6	…	…	…	Var_{N-1}
Var_N	0x2A	0x2A	chksm	0x0F	0x0F

2.5 Smart-device application

The layout of the smart device application is depicted in Figure 4.

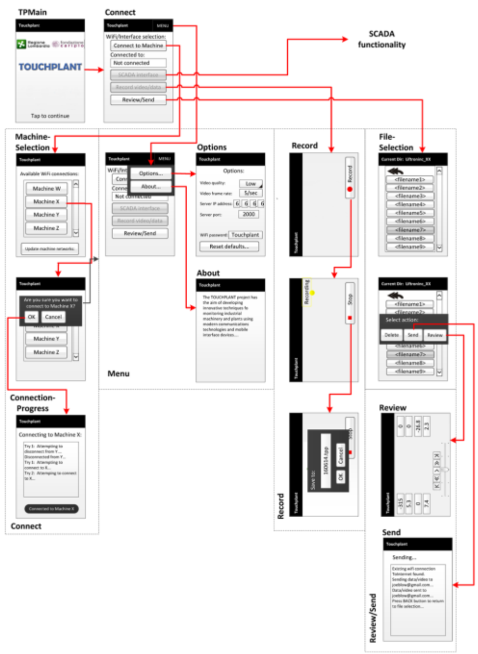

Figure 4 - Layout of the smart device application

The application is Android-based and thus written in Java.

In the Android architecture each screen of the application is a so-called “activity”. A user moves normally from one screen to another, performing various operations - e.g., connecting to a wireless network, recording a video, sending data files further on for further review and analysis, etc. Thus the app is organized by activities. How the app moves from one activity to another and what it does in each activity essentially describes the operation of the app.

In the application, there are ten user interface screens (and, obviously, ten activity classes). The passages between different activities are made using buttons that invoke the code described above. The typical workflow is defined in Figure 5.

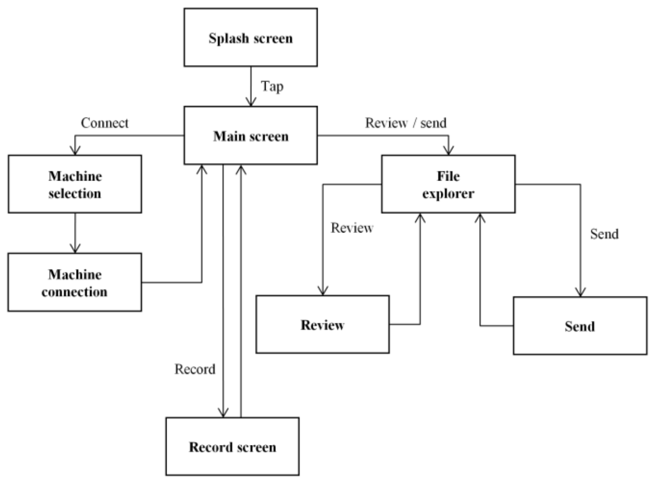

Figure 5 - Scheme of the application use

The remote transmission of video and operational data is currently via e-mail with a log attached.

2.6 *Useful data to manage maintenance interventions*

The data acquired from the machine are necessary to determine the real functioning status and, consequently, to define the maintenance interventions and how to manage these operations remotely. The logging system is developed for use by a normal operator of the machine. All the logs can be sent to the manufacturer to being analyzed by skilled maintenance personnel who determine, through the monitoring of threshold values, if the machine is in good operating condition or, in the opposite case, which components have degraded and need replacement or repair.

Particularly, the data acquired simultaneously are related to:

- Motor current;
- Motor voltage;
- Motor speeds;
- Operator commands;
- Handling times;
- General I/O;
- Alarms and warning messages.

Currents and voltages can be analyzed to see if the electrical components are functioning properly. Through these signals and the speeds, it is possible to diagnose the motor – the main actuator of the machine – to determine its condition and whether it is malfunctioning and, if so, what type of failure is present.

The movement speed and the handling time can be used to detect problems in the motion system, especially regarding the lift cables, the system of pulleys and the grasp system.

The I/O signals and the alarms and warning messages can notify the operator of the status of electromechanical components and if the control logic is working properly.

3. SYNCHRONIZATION

The synchronization of video and machine operating data is one of the most critical part of the application. As it is described above, the main challenge of this system is to obtain two synchronized streams of video and data information.

The video stream is acquired with a sample rate of 30 Hz, while the rate of the signals is 10 Hz. These two streams are acquired, as previously described, with the support of:

- The digital camera of the smartphone and an audio stream from the microphone;
- A communication channel (Wi-Fi) between the terminal and the machine.

In order to minimize the response time between the gateway and the MID (Mobile Internet Device), the layout of the network is with a single hop (from the Can / Wi-Fi gateway to the mobile device).

The application is developed in order to guarantee the minimum time delay between the activation of video recording and the start of data acquisition. When the user touches the Record button, it is necessary to start a peripheral request to the operating system (in particular the camera). The request and the response time depend on the hardware of the device and the version of the operating system; for this reason the acquisition start to the gateway is sent only after the positive response from the peripheral request.

During the acquisition, the video and the data streams are managed by two different threads: the video manager writes an mp4 file, while the data-stream manager writes a simple ASCII file (the file layout is described in the previous section).

In order to manage the synchronization of these streams, the data file contains the information related to the elapsed time of the acquisition for each sampling cycle.

In order to evaluate the synchronization performance, it is important to define some measures (related to the scheme represented in Figure 6):

- *TOF* (Time Of Flight) The time that is to send a packet from one point to another in the network (in this case a simple two-point network is created), considered constant and symmetric (this approximation was validated experimentally)
- *CT* (Computation Time) The time for the elaboration of the received packet
- T_s (Sample Time) The time between two consecutive samples

It is easy to understand that, if *TOF* and *CT* tend to a value lower than T_s, the delay between the request and the response could be approximated as *0 ms*. In order to verify the value of *TOF* and *CT*, an experiment was performed.

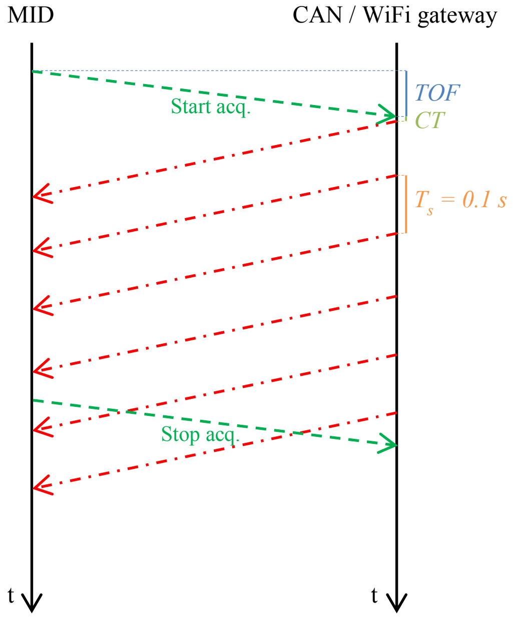

Figure 6 - Transmission graph

It is possible to estimate the sum of *TOF* and *CT* by sending a packet from the MID and configuring the gateway to respond to each packet with the same one. The value of *CT* can be obtained by changing the value of a digital output during the evaluation of the algorithm. The scheme of the test is shown in Figure 7.

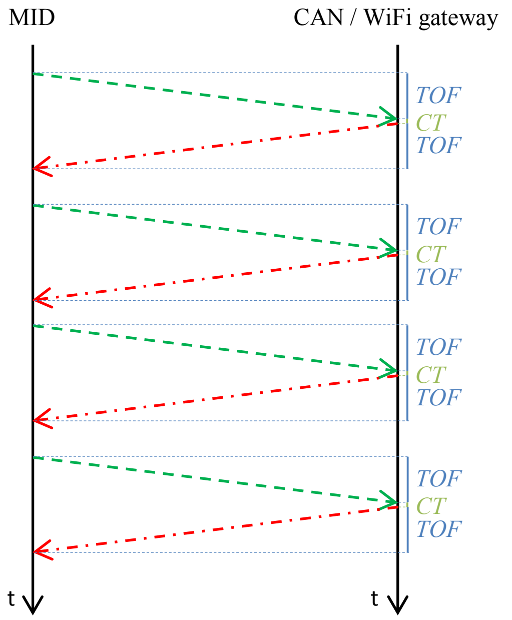

Figure 7 - Test scheme

The results of the experiment are detailed in Table II. The mean *TOF* for a complete round-trip, compared with the standard T_s, has a very small value (approximately 9%; with the computation time <10%). For this reason it is possible to neglect the delay between sending and receiving the packet.

Table II - Tests results

	Average value [ms]	**Standard deviation [ms]**
TOF	4.51	1.7
CT	0.73	0.012

The advantages of this solution are manifold:

- Reduction of the mobile device computation time;
- Absence of complicated applicative protocols (the re-transmissions are handled directly by the TCP protocol);
- Guarantee of the defined T_s (because it is managed by the gateway).

On the other side, the limitations of this solution are:

- The presence of a residual delay (estimated to be lower than 10 ms);
- The possibility of receiving a packet with a high delay (it is not important because it is not necessary to show data during the acquisition; the synchronization is made using the gateway timestamp).

4. EXPERIMENTAL RESULTS

The overall system has been tested on a real plant. The steps that have been performed are the following:

- Acquisition of different complete working cycles (for more information see Figure 9);
- Review of the acquired data on the MID (depicted in Figure 10);
- Transmission of data to the manufacturer;
- Remote evaluation of the logs.

Figure 8 - Acquisition screen

These tests are provided, as simple example, on a bridge crane; obviously, the result are extendible to different machines with a similar structure (the only requirement is the presence of the CAN backbone).

These tests are executed, as a simple example, on an industrial bridge crane; obviously, the result are extendible to different machines with a similar structure (the only requirement is the presence of the CAN backbone).

Figure 9 - Review screen

5. CONCLUSIONS AND FUTURE DEVELOPMENTS

In this paper a multi-source logger for remote maintenance is presented. The main problem of the solution is the synchronization between the gateway and the smart device. The solution is a simple, low-computation synchronization protocol. This solution has been tested in a real industrial setting.

Future development will be as follows:

- Implementation of a standard "industrial" synchronization protocol in order to evaluate the performance of this system with respect to other, common solutions;
- Coexistence test of the complete system in order to verify the robustness of the solution in harsh industrial environments;
- Development of a multi-client gateway, in order to allow simultaneous communication with more than one smart device;
- Implementation of a server application. This activity will be split into two different steps:
    - Implementation of a supervised maintenance solution (with the support of skilled technicians);
    - Implementation of an unmanned maintenance system, able to perform fault detection and isolation without the support of personnel.

ACKNOWLEDGMENT

This research activity has been carried out in the context of the "Touchplant project", a project funded with the joint support of "Regione Lombardia" and "Fondazione Cariplo".

REFERENCES

B. I. (2003). From remote maintenance to MAS-based e-maintenance of an industrial process. *Journal of Intelligent Manufacturing, 14*(1), 59-82.

Ermidoro, M., Formentin, S., Cologni, A. L., Previdi, F., & Savaresi, S. M. (2014). On time-optimal anti-sway controller design for bridge cranes. *Proceeding of the 33rd American Control Conference.*

Espindola, D. B., Pereira, C. E., Henriques, R. V., & Botelho, S. S. (2010). Using mixed reality in the visualization of maintenance processes. *Prceeding of the 1st IFAC Workshop on Advanced Maintenance Engineering, Services and Technology.*

Lee, J., Ni, J., Djurdjanovic, D., Qiu, H., & Liao, H. (2006). Intelligent prognostics tools and e-maintenance. *Computers in Industry, 57*, 476-489.

Lu, K. Y., & Sy, C. C. (2009). A real-time decision-making of maintenance using fuzzy agent. *Expert Systems with Applications: An International Journal, 36*(2), 2691-2698.

Mori, M., Fujishima, M., Komatsu, M., Zhao, B., & Liu, Y. (2008). Development of remote monitoring and maintenance system for machine tools. *CIRP Annals - Manufacturing Technology, 57*(1), 433-436.

Ong, M. H., & Lee, S. M. (2004). Evaluating the use of multimedia tool in remote maintenance of production machinery in the automotive sector. *Prceeding of the IEEE Conference on Robotics, Automation and Mechatronics.*

Previdi, F., Fico, F., Belloli, D., Savaresi, S. M., Pesenti, I., & Spelta, C. (2010). Virtual Reference Feedback Tuning (VRFT)of velocity controller in self-balancing industrial manual manipulators. *Prceeding of the 29th American Control Conference.*

Previdi, F., Fico, F., Savaresi, S. M., Belloli, D., & Pesenti, I. (2012). Direct Design of a Velocity Controller and Load Disturbance Estimation for a Self-Balancing Industrial Manual Manipulator. *Mechatronics, 22*(8), 1177-1186.

Wang, W., Tse, P. W., & Lee, j. (2007). Remote machine maintenance system throught Internet and mobile communication. *The International Journal of Advanced Manufacturing Technology, 31*(7-8), 783-789.