WIFI BASED REMOTE CONTROL SYSTEM WITH VIDEO FEEDBACK FOR INTELLIGENT VEHICLES

Sergio Nogueira, Jean-Michel Contet, Yassine Ruichek, Franck Gechter

University of Technology of Belfort-Montbéliard
Systems and Transportation Laboratory
90010 Belfort France

ABSTRACT

The proposed work lies within the framework of a program concerning the concept of intelligent vehicles and their integration in the city of the future. This paper presents a real-time remote control system with video feedback using wireless technologies. The system consists in controlling an automated vehicle by a distant operator through a computer or pocket PC. This operation is possible since the operator perceive the environment of the vehicle thanks to a video feedback from the camera installed on the vehicle. The vehicle controlling concerns the checking, validation and converting the information transmitted by the operator into vehicle commands (speed, brake, steering). The images provided by the embedded camera are compressed using specific algorithms and sent to the operator in real-time conditions in order to ensure synchronous operations between the operator and the vehicle. The system is tested and evaluated using our automated vehicle platform. The experiments show that the system matches the expected results.

1. INTRODUCTION

Today, public transportation systems in urban environment become less and less competitive. They are not adapted to the new needs of disperse transportation in city center. Moreover the current situation into urban cities increases air and noise pollution. The objective of our program is to provide new solutions in terms of urban mobility in order to improve user life (accessibility, security, comfort, rapidity, air preservation, noise reduction, etc.). Considering the existing public transportation systems, the goal is to develop a complementary transportation system designed for short trips everywhere in a city.

This new transportation system is composed with a fleet of proper and intelligent compact vehicles[1] that could provide automated assistances (speed and distance control, obstacles detection, platooning, automatic car parking, accessibility to embedded and distant services). In addition, this system offers new possibilities such as car sharing or public open access to vehicle units. These advantages should incite users for limiting the usage of private cars in the city. This will reduce efficiently the traffic with a better management of intelligent vehicle fleet, offering three driving modes: manual (for car-sharing…), semi-automated (driving assistance…) and fully automated [3].

This paper focuses on a real-time remote control system with a video feedback using wireless technologies. This system allows an operator to control a distant intelligent vehicle by receiving the vehicle environment thanks to a video feedback from embedded cameras installed on the vehicle.

The paper is structured as follow. Section 2 describes the video feedback procedure. The video data compression is defined in section 3. Section 4 presents the intelligent vehicle control. Before concluding, section 5 gives experimental results.

2. VIDEO FEEDBACK

In order to remote control the vehicle, an operator needs a real time perception of the vehicle environment. Consequently a video feedback from embedded cameras is needed. To ensure this phase, the algorithm is decomposed by 4 main steps: (i) data acquisition; (ii) image processing; (iii) image compression; (iv) video transmission.

The video transmission function can be viewed as a video stream server. A specific client application (computer, pocket PC…) can be connected with the embedded software to receive video data. The data can be compressed or not depending on the video client configuration.

The image transmission is based on the socket programming using the UDP (User Data Protocol) internet protocol. Therefore, an image represents so much data to be transmitted on this form. Each image must be split into several blocks and finally sent as a plenty of short messages.

2.1. Image subdivision

Generally the image subdivision is made line per line. In this paper, we propose to split the image into several square blocks called “macroblock”. A macroblock is piece of image of 8x8 pixels.

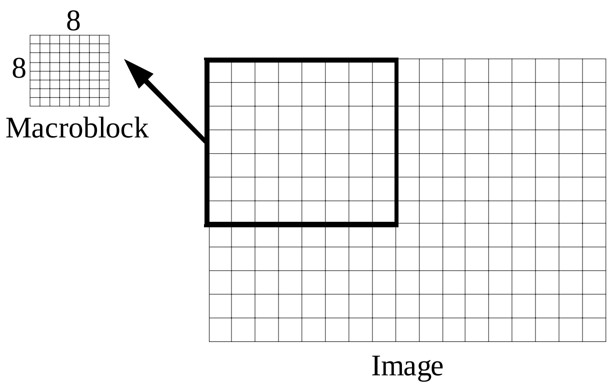

Figure 1 : Image subdivision

2.2. Image transmission

The video transmission is made by socket programming using the UDP internet protocol. To receive a video a client needs the “UDP ip” and “UDP port” of the embedded computer. This architecture allows all platforms (WinCE, Linux, XP…) and programming languages (C++, Java, c#…) to receive the video and control the intelligent vehicle. Consequently all devices (pocket pc, Smartphone, laptop…) supporting network can be used as a remote control operator.

Considering an image I with a size equal to 320x240, sending I, macroblock per macroblock, represents 1200 access to the network layer. In order to accelerate the network transmission, a specific pack, which contains a few macroblocks, has been created. As a consequence, the macroblocks must be identified in order to reconstitute the image for the client application. A macroblock pack is composed as shown in figure 2.

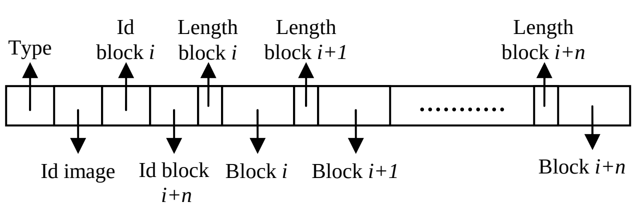

Figure 2 : Macroblock pack

The parameters of a macroblock pack are:

- Macroblock pack type, which identifies the composition of the pack blocks. This parameter is used for optimizing the macroblock pack depending on the data size.
- Image identifier;
- Block identifier in the image;
- Length of a block;
- Data of a block;

In spite of optimizing the image transmission via the network, it is necessary to reduce the data quantity using compression algorithms in order to respect real time constraint.

3. COMPRESSION

An image sequence takes a huge memory storage space. For example, in a video stream with a resolution of 640x480 pixels in grey scale (8 bits) at 25 images per seconds, the bit rate is:

$$640 \times 480 \times 8 \times 25 = 58.59 \text{ Mbits/sec}.$$

In the condition of an outdoor application, the wireless network rate is in theory 54 Mbits/sec. Therefore, it’s necessary to compress data to transmit in an efficient form. Video streams are particular data. They contain both redundancy data:

- Space: they are a strong correlation between neighborhood pixels of the same frame
- Temporal: A strong correlation exists between two consecutive frames.

Redundancy data are used during prediction. In order to compress the video stream, a specific process pipe has been implemented using both lossy and lossless algorithms (see figure 3). Each process works on macroblocks of 8x8 pixels. The compression process [2] pipe consists of six main operations as shown on figure 3.

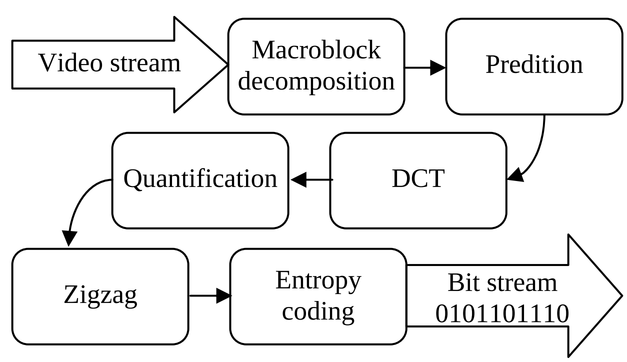

Figure 3 : Compression process pipe

4. CONTROL

The specific embedded software in the intelligent vehicle has two main tasks: transmit the video to the vehicle controller (sections 2 and 3) and receive the control commands. This second task is composed of three functionalities: (i) checking for new control command; (ii) validating the control command; (iii) converting the control command in vehicle command (speed, brake and steering).

The control command transmission is based on socket programming as video transmission, but it uses TCP (Transmission Control Protocol) internet protocol. This is important in order to guarantee the security needed for the navigation. Indeed, this communication protocol certifies data emission/reception (The TCP module sends back an acknowledgement for packets, which have been successfully received). In addition the TCP checks that no bytes are corrupted by using a checksum. The speed and angle commands are normalized into the range $\left[-400,+400\right]$ to simplify the control from external devices. All values outside from this range will not be took in consideration by the vehicle. Moreover, to avoid some control clumsiness, like changing instantly speed from 400 to 0, ramp has been employed.

4.1. Acceleration/Deceleration ramp

Acceleration/deceleration ramp is used in order to control the vehicle dynamic with more comfort and security. The ramp has been specifically defined for the intelligent vehicle dynamic. Logarithmic function (cf. continuous line of figure 4), offers more flexibility and is more similar to standard private vehicles. The car speed s is computed as:

$$s = s + k \cdot \log\left(\left|s-d\right|\right) \text{ where } k = \begin{cases} -2 & \text{if } \; s-d>1 \\ 1 & \text{if } \; s-d<0 \\ 0 & \text{else} \end{cases}$$

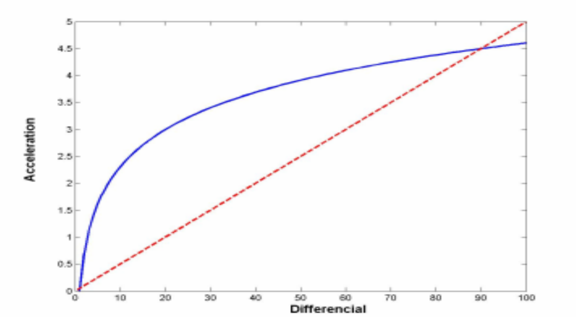

Figure 4 : Acceleration ramp

The vehicle deceleration is two times stronger than acceleration. However, sometimes it is necessary to stop immediately the vehicle. A procedure is especially implemented both in the controller and vehicle software in order to take into account urgent situations. If the software controller sends quickly zero values or values out of range, an emergency stop is activated. In addition, if no values are received from the controller during one second, the emergency stop is also applied. To validate the effectiveness and robustness of the remote control, controller software has been implemented and tested on two devices: Pocket PC and laptop.

4.2. Pocket PC Controller

The controller software for palmtop gives a simple interface to control the vehicle. After establishing the connection between the vehicle embedded software and the controller software, video from the camera (installed on the vehicle) appears on the palmtop screen (cf. figure 5).

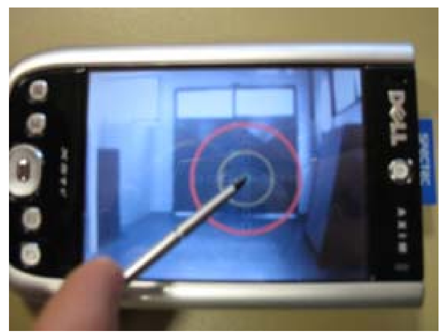

Figure 5 : Target on Pocket PC screen

The vehicle remote navigation is simply controlled by the palmtop software controller using the stylus. The speed and wheel commands are computed from stylus movement on the screen and normalized in the range $\left[-400,+400\right]$. The speed commands are transformed using the acceleration/deceleration ramp. Finally, commands are sent to the vehicle and converted in vehicle commands.

It is possible to activate an emergency stop on the pocket PC by removing the contact between the stylus and the screen of the palmtop. If needed, pressing a palmtop button will disconnect the TCP connection with the vehicle embedded software causing, thus, an emergency stop. In this case, the vehicle/palmtop connections must be reinitialized to control again the intelligent vehicle.

5. EXPERIMENTAL RESULTS

All experiments have been performed on an intelligent vehicle platform called RobuCAB (cf. figure 6). This vehicle has four driving and steering wheels. An electrical motor is on each wheel allowing the different configuration of motion: dual, single and “crab”. An embedded PC (with an X86 processor) linked, via CAN bus, to two MPC555 (Motorola PowerPC Processor) control cards provide the intelligent vehicle control. The vehicle is equipped with several sensors (laser range finder, sonar, Magnetical Field, cameras) and different communication abilities (Wifi, GSM/GPRS).

Figure 6 : RobuCAB platform

The palmtop used for testing the remote control and the video transmission is a Dell Axim X51v working with Windows Mobile 5.0. This Pocket PC has an Intel PXA270 624MHz processor with an Intel 2700G 3D accelerator. A Wifi SD Card (Spectec Card) is added for performing connection in 802.11g (54 Mbps).

The experiment shows a significant difference between the theoretical possibilities of the Wifi technology (802.11g) and practical uses. Indeed, the maximal rate obtained for data transmission was 21 Mbps of the 54 Mbps, which represents only 40% theoretical rate. This is due to the performance of the used hardware.

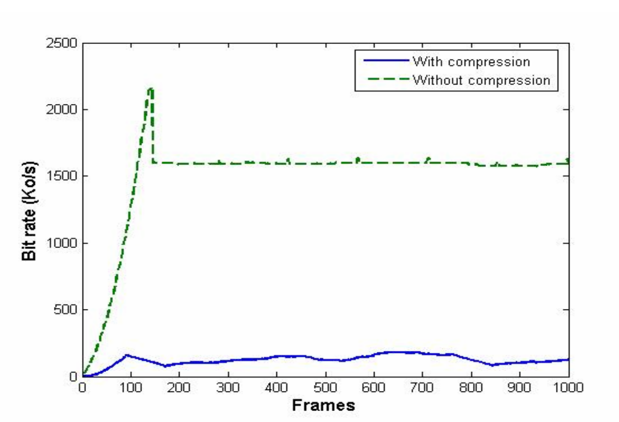

Figure 7 : Bit rates with and without compression

The compression performance was tested using a video composed with images having a resolution of 320x240 at 20 frames per second. This represents rate of 1500 Kbps. The quantitative result (cf. figure 7) shows that the average compression rate is about 1000%. As shown in figure 8, which represents a zoom of the continuous curve of figure 7, the average bit rate for compressed data is about 150 Kbps.

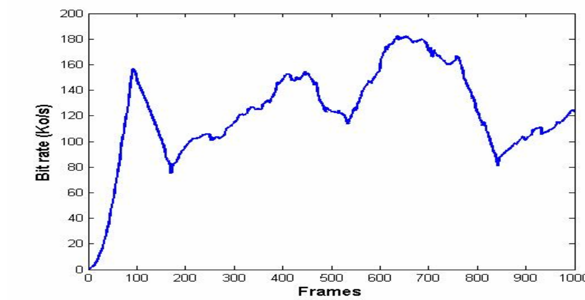

Figure 8 : Bit rate with compression

Concerning the vehicle control, the commands are performed in real time: each command is achieved in less than 2ms. Thanks to the compression rate and command response time, the distant operator is capable to control the vehicle with an effective reactivity and without any difficulty. The compression benefit allows adding more cameras views in order to give the operator the possibilities to perceive the entire vehicle environment.

6. CONCLUSION

In this paper we proposed a Wifi based system for intelligent vehicle remote control with a video feedback. This system is composed with two parts. The first one concerns the vehicle embedded software destined to send video data to an external device using Wifi network communication. The second part represents a software, installed in the external device, that allows to send navigation commands to the vehicle using the wireless connection. To reach an effective reactivity between the vehicle and the operator, the video data is compressed and the transmission procedure is optimized. Experimental results show the effectiveness and robustness of the vehicle remote control system. Future works will concern the analysis of the long wifi range.

7. ACKNOWLEDGE

Many thanks to Fabien Houlmann and Stéphane Metz for their contribution to the implementation.

8. REFERENCES

[1], Parent M. and Fauconnier S., “Design of an electric vehicle specific for urban transport,” in Congrès EVT’95. Paris, 1995.

[2] S. Nogueira, Y. Ruichek, F. Gechter, A. Koukam, F. Charpillet, « An artificial vision based environment perception system », pour le livre « DSP for In-Vehicle and Mobile Systems 2 », Springer.

[3] D. Salomon, "Data Compression: The Complete Reference", Springer, 3 edition, February 2004.