AndroCar – Mobile Powered Car Security System

HASD Thissera, HMT Paranagama, SC Gajanayaka#, TT Sugathadasa,
HKK Hewasinghe, DMR Kulasekara and AS Karunananda
Faculty of Computing, General Sir John Kotelawala Defence University, Ratmalana, Sri Lanka
#sinhalokaya@gmail.com

Abstract— *As smart phones, tablets and many other portable devices are getting closer and closer to the humankind; many technologies are being released day by day. Among them, security has been one major aspect where much research has been done. Though modern vehicle security systems have been developed, there are certain limitations which are experienced in the practical usage of such applications. We realized that though many such systems were available, not many had the capability of providing a video feed via an IP along with triggering an alarm and so forth using a mobile device. The present systems in the market are quite expensive and use too advanced technologies which require specialist installation and maintenance. Mobile integration is also not as optimized as it should be. Therefore our main goal in this research was to "integrate" all these features and build a coherent mobile car security system with a centralized administration. The basic underlying idea of this project therefore is to provide a "practical" and a "cost effective" car security system to the general public. In this project, we provide a camera feed (via IP) of the vehicle to the vehicle owner to his / her mobile device and then enable the user to control the system remotely. It also can detect any human activity, using the switches and report it to the owner via an auto generated sms notification. We have used a web camera which is connected to a Raspberry Pi chipset and powered using a power bank, switches, LED bulbs, 2 speakers and a Wi-Fi adaptor to build the system. When someone tends to open the door lock of the vehicle, a switch which is situated under the door lock of the vehicle, will trigger giving a signal to activate the chipset and send a sms notification to the owner. Then the owner can get an instant view of the situation (on his mobile), and if he wishes to, he can take necessary steps to protect his vehicle from any danger. We have tested the implementation under lab conditions and the results have been quite satisfactory. We were able to achieve the main goals which we wanted to achieve. However it needs to be noted that if this is to be implemented commercially we would have to develop our system to use "port forwarding" (so that the owner may view the video from anywhere in the world), and a city-wide Wi-Fi network or a high-bandwidth data connection would be necessary (HSDPA / WCDMA) for the system to work smoothly.*

Keywords— *Car Security System, Raspberry Pi, Mobile Security System*

I. INTRODUCTION

The number of motor vehicles stolen increase (INTERPOL, 2014) day by day but there is no proper system to avoid such crimes and identify the thefts. In 2013, 715,373 vehicles were stolen with a value amounting to $4.3 billion. Therefore much still needs to be researched in this area for the development of effective security systems using cutting edge technology (Borazjani, et al., n.d.).

A. Background

Vehicle related security is one key aspect associated with recent trends in the development of technology. Most of the systems in use are able to detect any "collision" or "strong impact" on the body of the car using sensors and fire up an alarm. This is not effective at all times as the alarm goes on even for a strong wind or even when an animal hits the body of the car. This system proposed is expected to solve that problem, as a switch is installed on every door lock and it is only when someone tries to open the door using the door lock, that it would be detected. Furthermore, a video feed of the car would enable, at any given time, the owner to take necessary action and to control the whole system using a simple smartphone app. With growing inter-linked "smart-cities", a system of this sort may further enable state-level police departments and others concerned to track vehicles real-time.

B. Problem definition

On research, we realized that car security systems available in the local market did not meet the intended objectives of a gadget of that sort – as aforesaid. It is in this light that we decided that we should base our project in this sphere and try and develop a system which would solve the aforesaid lapses in existing applications.

C. The objectives

As identified below, the main objectives that were expected to be achieved via this research were as follows:

- Provide a camera feed (via IP) of the car to the car owner to his / her mobile device.
- Remotely control the alarm.
- Detect any human activity (Using the switches) and report it to the owner – via an auto generated notification.

II. RELATED WORK

It is apt to give regard to the research done in this area. It should be noted that this area has inspired many a scientist to actively research and develop mechanisms to implement security systems to be more efficient and user friendly.

The site called "Instructables" seemed to have many projects done by various people with the Raspberry Pi. The project titled, "Raspberry Pi as low-cost HD surveillance camera" caught our attention immediately (Scavix Software Ltd. & Co. KG, 2011). On the face of it, this looked quite identical to what we wanted achieve.

However, upon close inspection, we realised that there were differences to what we actually wanted to do. This project was referring to the video camera being fixed to a stationary position, so that the video feed of a "place" may be obtained. We wanted to fix the video camera to a car, thereby making the camera "mobile". Furthermore, this project had the feed viewable as a web page on a PC

In this project, they have used a Linux distribution package called "motion" (KennethLavrsen, 2014) to stream the video to a remote client. We tried out the whole procedure as mentioned in the "steps", but realised that we faced a few problems:

- The camera feed was viewable on a PC web browser, but had "noticeable" lapses in the feed received. The quality of the feed received was altogether of a low quality. Therefore we had to look at a different solution than "motion".
- The web page only accommodated the feed. We wanted to include the servo controls, alarm and night mode controls as well…
- Furthermore, the storage option never worked as specified in the project, where the videos would be stored on a remote server connected to the same network. We realised that the reason to this was the fact that we didn't use the official RPi camera module. We had to find a solution that worked with the custom camera we used. However, later we realised that we had no intention to store the videos captured… A sms notification would anyway be sent to the user if there was an intrusion, and thus there was no need to have the videos stored, accumulating extra disk space.

From the other projects we read about, almost all of them involved streaming a video. However there was no project where the mobile phone interface, i.e. Android and such an OS, was geared to "communicate" with the camera while running a servo motor and an alarm simultaneously. Furthermore, it should be noted that there were many domestic level projects done by amateurs, where they had used the Pi and other such chipsets to control the room lights, electronic devices, monitor temperature levels (Isa & Sklavos , 2015), open doors via a sms (Elfasakhany, et al., 2011) and so on. Therefore it is this requirement that was the core concern of this research – to integrate all above components in to the research.

When referring to academic research, we first studied the "Smart home automation" (Isa & Sklavos , 2015) which attracted the interest of the research community during the last decade, in a great manner. The introduced system operation is supported by a GSM module, which enables the alert messages transmission to both mobile devices of end users, and central security offices. Each time, the involved end users and the security officers, can be informed of attacks, operation modes changes etc. through SMS.

We used this system to get an idea as how to develop a connection between the vehicle's security camera & a mobile. A study was done as to whether we may directly apply the techniques used in this as our system was a "mobile" system and not a "stationary" one like smart home automation.

Next we studied the Open Car Testbed And Network Experiments (OCTANE) (Borazjani, et al., n.d.) which reduces the barrier of entry into the security research and teaching of automotive networks by providing a software package and a hardware framework for the reverse engineering and testing of automotive networks. OCTANE provides a platform for security research and teaching by replicating the interactions between the hardware components and control software of the systems so that the user can focus on the security aspects of the automotive network instead of the tool configuration and setup.

By studying various researches what we understood was that image-recognition techniques (Sunitha, et al., 2014) can provide the important functions to an advanced intelligent Car Security system, to avoid vehicle theft and protect the usage of unauthenticated users. Thus if an unauthenticated person tries to steal / damage the car, the embedded controller platform stops the car immediately. GSM module will send the indication to the user as soon as the car gets stopped.

The car security system that locks / unlocks the devices by using password is the system which we studied next (Anbazhagan, et al., 2014). This technique is not only applicable to vehicle but quite applicable to control the various appliances in home. But after studying their project we understood that it's not a cost effective though it is highly confidential.

The next project we studied was the project which the ATMEGA 2560 controller (Dudhale , et al., 2015) was being used with three proximity switches below the accelerator, the brake and the clutch. There is a sequence which transmitted to the microcontroller, which compares the entered sequence with the pre - stored sequence. The car would ignite only if these sequences matched, if not, system uses a GSM module to send SMS to the owner of the car and a GPS to broadcast the location of the car as well as buzzer will switch on and doors will be locked. So the owner of the car is able to secure the vehicle as the status of the car can be sent to registered mobile numbers through system's GSM module. Also with the help of ATMEGA2560 controller, processing of data can be done easily. We looked at how this chipset itself may be used for our concerns, but realized that the Raspberry Pi chipset was even more cost effective.

We also found a complete project on a site called makezine.com which had quite an identical project to what we were going to do (Stultz, n.d.).

This project had a web camera feed, two servo motors to control the same and a complete web interface where both the motor can be controlled and the camera feed be viewed.

Thanks to the creator of this project, Matt Stultz, we were able to download the exact image of his configurations of the Raspbian image, which made things quite easier for us.

Upon flashing the image given, we were successfully able to view the camera feed in the web application created by him. This web view contained four buttons to control the 2 servo motors, which would control the angle the camera is positioned as shown in Fig. 1:

Figure 1: Camera feed and the buttons (web interface)

III. EXPERIMENTAL DESIGN

When considering the current context the vehicle security systems, we see that most of them lack accuracy i.e. even if an object / animal strikes the vehicle the alarm will be triggered, which is inappropriate and if the owner of the vehicle is not in the range where he could hear where the security alarm the system is inadequate.

We also need to realize that almost all these systems provide "strong" and "long-lasting" solutions, which need a more complex setup and technician help. What we wanted to build was a simpler version of the same, where anyone would be able to fix it up himself and make things work. Furthermore, our solution would be geared for more "short-term" needs such as catering to an instance where the vehicle owner may want to have a quick look at his / her car while in a meeting and so on.

However we need to note that there are highly sophisticated systems too, which can even detect human behavior through motion detection and recognition and "turn" the camera accordingly. These are high-end gadgets and tend to be too costly for general-purpose work. Since our main objective was to achieve these in a more cost-friendly and a “tidier” way, we believe that our research would fit in to merge the existent gaps in this arena.

Can we get rid of these limitations? Yes we can. The solution which we have proposed provides a more advanced, cost-effective and simple (user friendly) equipment which will wipe off these limitations to a great extent.

There will of course be issues where our system would tend to be irresponsive, for e.g. when internet is down etc. but we hope that in the majority of general cases, our system would prove to be enough and sufficient.

Fig. 2 below explains the design of the system concisely (Sunitha, et al., 2014). Initially the system would be temporally powered off until a “collision” is detected via the switches fixed to the car.

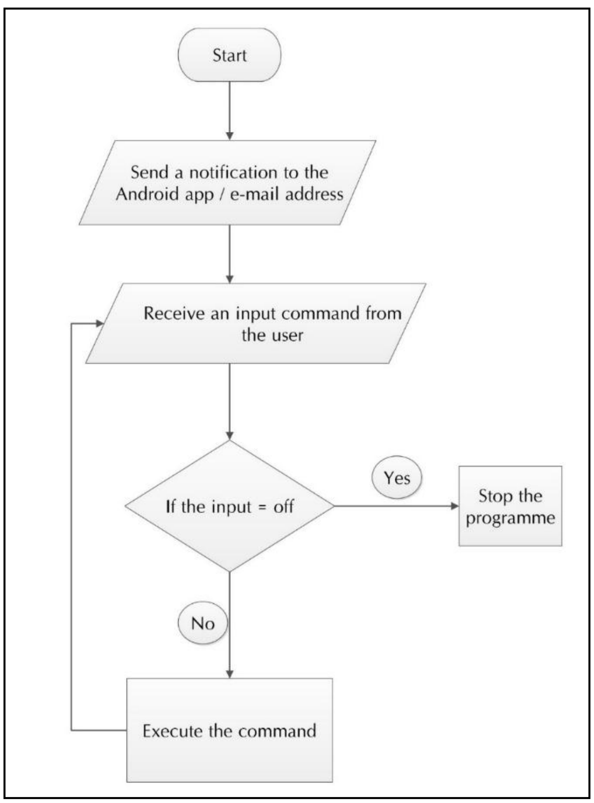

Figure 2: Experimental design of the system

After this is detected, the system would power on, sending a sms notification to the car owner, to his mobile device. The owner may then decide to view the video feed from the car via a web browser or take an action such as control the lights of the car or fire up an alarm. For experiment sake a trial version of the Twilio sms API (Twilio, n.d.) has been used.

IV. IMPLEMENTATION

The Raspberry Pi chipset was used to construct a model of the proposed security system. The major items planned to be used for this project were Raspberry Pi (Model B preferred), USB internet dongle / Wi-Fi dongle (preferred), 4 switches, strong LED bulb/s, smartphone, a siren (we used a pair of multimedia speakers), web camera, servo motor, mobile USB charger (230 V to 5 V power adapter), powered USB hub, SD card of at least 4 GB capacity and so forth. The devices were connected as depicted in Fig. 3.

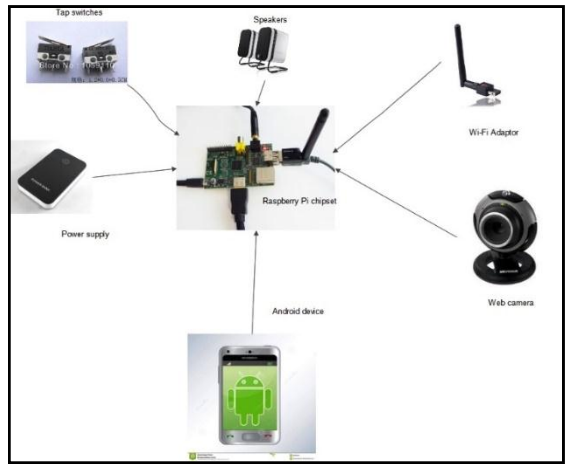

Figure 3: Overall connectivity of devices to the Pi

We followed the “formal” procedure of setting up the Pi as depicted in the official Raspberry Pi documentation (Raspberry Pi Foundation, n.d.):

- We connected an external USB keyboard and a mouse to a powered USB hub and plugged them to the Pi. *Note: It is advised to use a powered USB hub as the Pi is unable to power up extra devices using the 5 V power available to itself. If a powered hub is not used, the Pi might tend to restart on and off causing malfunctions in the file system.*
- The Pi was powered using a USB charger (230 V to 5 V power adapter), most commonly used among all mobile phones today.
- Ethernet cable was connected to the Ethernet port to obtain internet connectivity.
- Then to set up the Raspbian OS, we flashed the latest “Raspbian Debian-Wheezy 3.12” version available in the above web site (Raspberry Pi Foundation, 2014), to a 4 GB SD memory card and

inserted it to the card slot in the Pi, using "SD Formatter 4.0" (SD Association, n.d.). *Note: At least a "Class 6" SD card is recommended for optimal performance. The chip might tend to cause malfunctions in the long run, otherwise.*

- A CRT monitor was connected via a "VGA to HDMI" converter to the HDMI port of the Pi.

Upon switching on the system for the first time, the OS installation ran perfectly fine, and the system was ready to use in about 10 minutes. We then logged into the RPi interface using the default credentials.

We needed an easier and a quicker way to communicate with the Pi, and execute complex commands remotely. The best solution to this was to use SSH, as explained in an Instructable (Scavix Software Ltd. & Co. KG, 2011). A small app available for Windows 8 in this regard is "PuTTY" (Tatham, n.d.). Via PuTTY we were able to run almost any command on the Pi, just by connecting to the same local area network where the Pi is connected to.

A lorry-like structure was then designed, to house the necessary devices and equipment. The servo motor setup and the Raspberry Pi chipset are fitted to the "hood" of the car, usually to where the lamp comes. It would only be practical to do so, as a complete (360^0) view could be obtained in that manner, when the motor is turned back and forth. The speakers would have to be fitted to the back or the bonnet of the car, as appropriate.

An image of Raspbian (Stultz, 2014) was used to work with the Pi. The servo motor, LED bulbs and switches were connected to the pins in the Pi. For this we have used the BCM numbering (Matt, 2012). The package titled "RPi.GPIO" (compiled by Chris Hager (Hager, 2013)) for Python has been used to code for these devices. The camera was connected via a USB port.

All these devices are then linked to a web page (using the Flask framework for Python) where it is possible to operate them. For instance it is possible to (while viewing the web video feed of the car), fire up the alarm or the LED bulbs. When the user is not viewing the feed on the phone, the system (upon detection of an impact to the switches), send an automatic sms notification to the user's phone (Dudhale , et al., 2015).

V. RESULTS & DISCUSSION

We list below (Figure 4) a discussion of the fulfilment of objectives of the research.

Objective	Was it fulfilled? Or not?
1. To gain the knowledge about modern security systems and to research into what areas might need further development.	Yes. In our research, we researched about the modern security systems which are in the market, its pros and cons and the applicability of such systems to the present sophisticated society.
2. To provide a solution to an existing socio-technological problem existent by developing an effective car security system.	Yes. We were able to provide a flexible, cost-effective, less sophisticated system to a real-world problem concerning our everyday security systems. In the introduction, we identified certain lapses in the present systems, and were able to provide a solution to the same by our implementation and approach.
3. To develop the model car security system in a structured environment using a microcontroller.	Yes. The system was developed using the Raspberry Pi chipset.
4. Implement an app using the Android API (Application Programming Interface) or a web interface as appropriate.	Not as expected. We did not have to use the Android API in our project for any development work. We did, however, research about Android at length, but only resorted to making the final interface to be a web-based application, rather than an app.
5. To implement car security system using hardware and software.	Yes. We have justified the hypothesis we set at the start, through the final solution that has been presented. Both hardware and software have been involved, and the result has indeed been an embedded system altogether.

Figure 4: Table of the fulfilment of objectives

VI. CONCLUSION

While we have looked at what we have already done, it is important to note the improvements that can be done to this system. We have noted a few of these below:

- Fingerprint identification – You can develop this system with some other features like fingerprint identification. This would enhance the security of a vehicle and make it possible only for some selected people to start the car. Thus by implementing this system on a vehicle one can ensure much greater security and exclusivity than that offered by a conventional lock and key (Anbazhagan, et al., 2014).

The reliability of any automatic fingerprint system strongly relies on the precision obtained in the minutia extraction process. A number of factors are detrimental to the correct location of minutia.

- Motion detection – Furthermore you can develop the system with software aspects relating to motion detection. Efficiency of the system is high if it is developed with the motion detection for animals. For e.g., if such a system is developed, it would be possible to distinguish more distinctly between a human and an animal intrusion, and the Pi may be programmed to act "automatically" without even the owner getting involved.

While we have conducted our research to achieve a day-today objective and to provide a solution, emphasis should be made that much research awaits to be done in this arena by students.

ACKNOWLEDGMENT

We would like to express our great appreciation to Prof. AS Karunananda for his valuable and constructive suggestions during the planning and development of this research work. His willingness to give his time so generously is very much appreciated. We would also like to thank Mr. DMR Kulasekara for his advice and guidance that helped us to make this possible. We would also like to thank Lt. Col. SPP Pakshaweera, Mr. A Gunasekera and staff of the Department of Computer Science at KDU, for their valuable and precious time, which is generously and highly admired.

REFERENCES

Anbazhagan, R. et al., 2014. Design of Intelligent Car Security System. *Middle-East Journal of Scientific Research,* pp. 1229-1233.

Borazjani, P. N., Everett, C. E. & McCoy, D., n.d. *OCTANE: An Extensible Open Source Car Security Testbed,* s.l.: George Mason University.

Dudhale , A., Rasal , S., Sabale , A. & Thakkar , N., 2015. *Car Security Using A-B-C Analysis,* s.l.: s.n.

Elfasakhany, A. et al., 2011. *Design and Development of a House-Mobile Security System,* s.l.: s.n.

Hager, C., 2013. *Welcome to RPIO's documentation!.* [Online]
Available at: http://pythonhosted.org/RPIO/
[Accessed 2 December 2014].

INTERPOL, 2014. *Analytical Report - Motor Vehicle Crime in Global Perspective ,* s.l.: INTERPOL.

Isa , E. & Sklavos , N., 2015. *Smart Home Automation: GSM Security System Design & Implementation,* s.l.: s.n.

KennethLavrsen, 2014. *Motion - Download Files.* [Online]
Available at:
http://www.lavrsen.dk/foswiki/bin/view/Motion/DownloadFiles
[Accessed 1 November 2014].

Matt, 2012. *Simple Guide to the RPi GPIO Header and Pins.* [Online]
Available at: http://www.raspberrypi-spy.co.uk/2012/06/simple-guide-to-the-rpi-gpio-header-and-pins/
[Accessed 25 December 2014].

Raspberry Pi Foundation, 2014. *Downloads.* [Online]
Available at: http://www.raspberrypi.org/downloads/
[Accessed 15 October 2014].

Raspberry Pi Foundation, n.d. *Raspberry Pi documentation.* [Online]
Available at: http://www.raspberrypi.org/help/noobs-setup/
[Accessed 30 September 2014].

Scavix Software Ltd. & Co. KG, 2011. *Raspberry Pi as low-cost HD surveillance camera.* [Online]
Available at:
http://www.instructables.com/id/Raspberry-Pi-as-low-cost-HD-surveillance-camera/
[Accessed 20 October 2014].

SD Association, n.d. *SD Formatter 4.0 for SD/SDHC/SDXC.* [Online]
Available at:
https://www.sdcard.org/downloads/formatter_4/
[Accessed 14 October 2014].

Stultz, M., 2014. *Make:.* [Online]
Available at: http://makezine.com/projects/raspberry-

eye-remote-servo-cam/
[Accessed 3 January 2015].

Stultz, M., n.d. *"Raspberry Eye" Remote Servo Cam.* [Online]
Available at: http://makezine.com/projects/raspberry-eye-remote-servo-cam/
[Accessed 20 December 2014].

Sunitha, M., Kumar, M. V. & Raghu, G., 2014. *Embedded Car Security System,* s.l.: s.n.

Tatham, S., n.d. *PuTTY Download Page.* [Online] Available at: http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
[Accessed 15 August 2014].

Twilio, n.d. *API Documentation.* [Online] Available at: https://www.twilio.com/docs/api
[Accessed 10 January 2015].

BIOGRAPHY OF AUTHORS

Mr. HASD Thisera is a Computer Science undergraduate at General Sir John Kotelawala Defence University – Ratmalana, Sri Lanka. He is in his second year. He has learnt languages such as C, HTML and PHP. His research interests include Embedded Systems, Computer Security and Human Computer Interactions.

Mr. HMT Paranagama is a Computer Science undergraduate at General Sir John Kotelawala Defence University – Ratmalana, Sri Lanka. He is in his second year. He is proficient in C and Java. His research interests include Robotics, Artificial Intelligence and Wireless Communications.

Mr. SC Gajanayaka is a Computer Science undergraduate at General Sir John Kotelawala Defence University – Ratmalana, Sri Lanka. He is in his second year. He has mastered languages such as C, Java, C#, Visual Basic .NET, PHP and JavaScript. His research interests include Embedded Systems, Artificial Intelligence, Advanced Algorithms and Human Computer Interactions.

Ms. HKK Hewasinghe is a Computer Science undergraduate at General Sir John Kotelawala Defence University – Ratmalana, Sri Lanka. She is in her second year. She is proficient in Java and C#. Her research interests include Artificial Intelligence and Human Computer Interactions.

Ms. TT Sugathadasa is a Computer Science undergraduate at General Sir John Kotelawala Defence University – Ratmalana, Sri Lanka. She is in her second year. She is proficient in HTML and Python. Her research interests include Robotics and Embedded Systems.

Mr. DMR Kulasekara is an instructor at General Sir John Kotelawala Defence University – Ratmalana, Sri Lanka. His research interests include Robotics, Embedded Systems, Computer Graphics and Vision and Advanced Algorithm Designing.

Prof AS Karunanandanda is a senior professor of information technology at the University of Moratuwa, Sri Lanka. At present he is the dean of research and development at General Sir John Kotelawala Defence University. His Research interests include ontological modelling, multi-agent systems, intelligent systems, machine learning, local language technologies, complexity science and theory of computing