DOI 10.4010/2015.444
ISSN2321 3361 © 2015 IJESC

Research Article September 2015 Issue

Smart Vehicle Security and Defending Against Collaborative Attacks by Malware

A.Siva Krishna[1], Dr.S.Asif Hussain[2]
M.Tech Scholar[1], Associate Professor[2]
Department of ECE
Annamachaya Institute of Technology, Rajampet
smartshivakrishna@gmail.com[1], sah.ssk@gmail.com[2]

Abstract:
Present, there is a huge growth in vulnerabilities. So, Security systems are becoming the unavoidable systems in today's life because of the increasing criminal activities. In the proposed work a real time vehicle security system and malware detection is implemented. If any person starts the car, the security system will check the person's authentication. The proposed system allows only the authorized user to use the vehicle. If it finds any unauthorized person, the proposed Person Authentication System (PAS) will block the person to operate the car and it will send the alert information image to the system controller. Also the malware system detects any spyware in the image which has been implemented in java platform.

Keywords: GSM module, Person Authentication System (PAS), ARM LPC 2148

I. INTRODUCTION

Now a day's thefting is increased in the field of vehicle steal. Mainly in the luxurious system Cars are expensive. Other than a house, perhaps, few purchases we make will compare to a new car. And just like any other expensive asset, a car brings with it a secondary cost. Most of the people are using cars today to make their life well. According to this requirement the manufactures also bring the cost of the car too low. So even a middle class family can also purchase this freely. Increased car usage turns increase the theft also. In this project work a proposed security system is introduced in order to overcome this problem. The Person Authentication System block representation will illustrate the function of the entire project work.

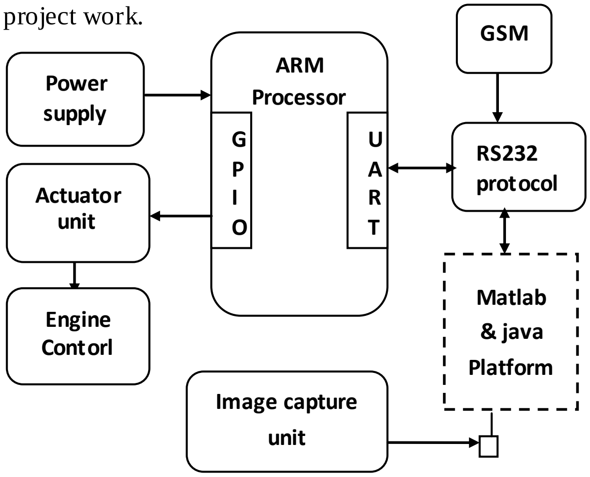


Fig 1: Proposed PAS block representation

This project consists of a Processor using ARM core, GSM unit as hardware parts and an effective face recognition system using Matlab platform and malware identification using JAVA platform. In this project initially the owner's image or else the driver's image should be stored in the database. Whenever a person is starting the car, the face detection recognition unit will takes the image and it will compare with the database image. If the image is matched then the car will move with out any problem. In other case if the image is unmatched, then the alert will be transmitted to the stored Security Mobile Number (SMN).

II . DESIGN AND IMPLEMENTATION

This project uses two important platforms.

1. Software Platform and
2. Hardware Platform.

These platforms are discussed below

i) Software Platform

In this project a face recognition system is used which will do the key role in the entire operation. For the face recognition system, we are using the MATLAB and malware detection java Platform is used. The image recognition will process in the following way.

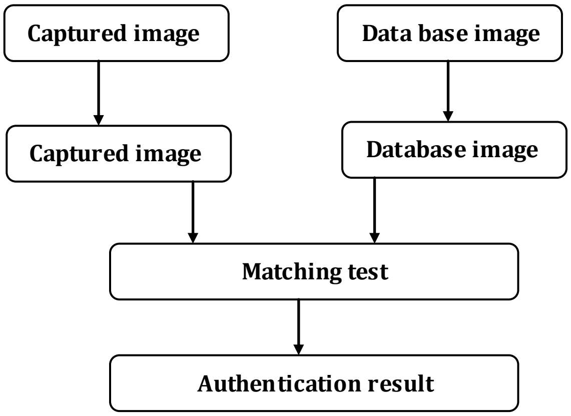


Fig 2: PAS Software architecture

For capturing the image we need to use a camera which should support a YUY2_640x480 format. Initially we have to take the data base image and should store in the project folder. The supportable camera configuration in the MATLAB is given in the form of the following data function vid=video input ('winvideo',1,'YUY2_640x480'); Initially ten data base sample image have to be stored in the project folder to get the effective feature extraction. Once the camera captured the image means it will be send to MATLAB. Whenever MATLAB reads an image it will convert into grey scale format because for recognition purpose the image should be a single plane. After capturing the image, we need to click on the database. As an acknowledgement we will get the following help dialogue.

The Command 'helpdlg' ('database successfully added). Then pre-processing will be done with in the captured image and the database image which involves, Similarity checks and probability finding. Here similarity checking is nothing but the comparison between two images by calculating the distance between the input and data base image.

We can do this by an effective edge analysis and pixel analysis. Using the function value = Euclidean Distance(X, Y*)*, we can find the similarities between the input image and data base image and also the changes in the same input after a particular time period.

Finally, pixel value result will be compared with the mean and median value to find the authentication. Then the result will be shown on the MATLAB.

```
if (cnt1>2 || cnt2>2)

if cnt1>cnt2
    warndlg('Authorised Person is A');
else
    errordlg('Person not Autherized');
```

If the image is matched then there will not be any response to the SMN. But if the image is not authorized then, the alert will be send to the SMN through GSM modem.

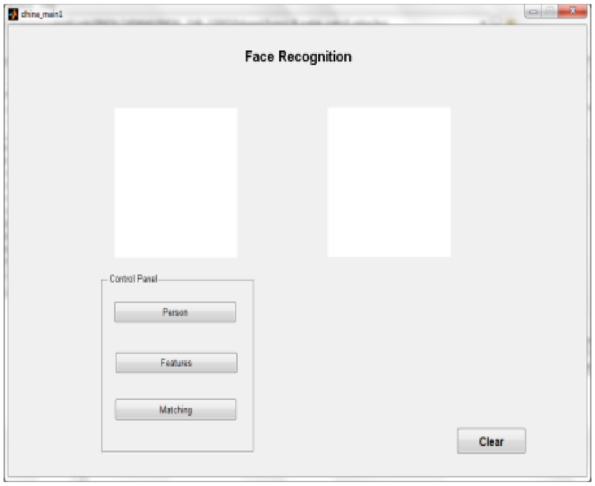


Fig 3: GUI Dialog representing Personal Authentication System (PAS) Screen

ii) Malware detection:

In malware detection system we have a tendency to area unit victimisation navie Bayes formula for classification with accuracy. a plus of naive Bayes is that it solely needs a little quantity of coaching information to estimate the parameters (means and variances of the variables) necessary for classification. as a result of freelance variables area unit assumed, solely the variances of the variables for every category have to be compelled to be determined and not the complete variance matrix

III. SYSTEM ARCHITECTURE

The proposed work describes the architecture in various modules which are

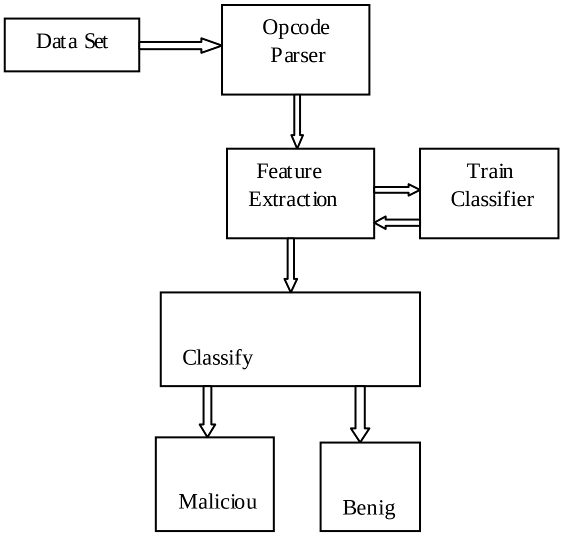


Fig.4: shows the architecture of the system

1. Data Collection
2. Dataset Creation
3. Feature Extraction
4. Classification0

1. Data Collection

In this stage, data set consists of 100 binaries out of which 90 are benign and 10 are spyware binaries. The benign files were collected from Download.com, which certifies the files to be free from spyware. The spyware files were downloaded from the links provided by SpywareGuide.com. This hosts information about different types of spyware and other types of malicious software

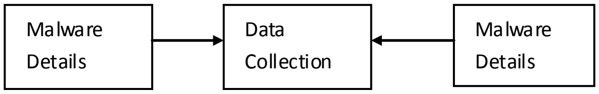


Fig.5: Collection of data from various sources

2. *Dataset Creation*

In which byte sequences represent fragments of machine code from an executable file. We use xxd, which is a UNIX-based utility for generating hexadecimal dumps of the binary files. From these hexadecimal dumps we may then extract byte sequences, in terms of *n*-grams of different sizes. ARFF databases based on frequency and common features were generated. All input attributes in the data set are represented by Booleans. These ranges are represented by either 1 or 0.

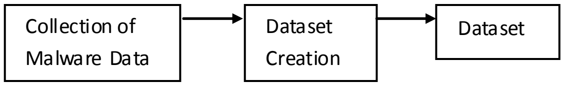


Fig.6: Creation of data with attributes

3. *Feature Extraction*

In this stage output from the parsing is further subjected to feature extraction. We extract the features by using following approaches, the Common Feature-based Extraction (CFBE) and Frequency-based Feature Extraction. The occurrence of a feature and the frequency of a feature. Both methods are used to obtain Reduced Feature Sets (RFSs) which are then used to generate the ARFF files.

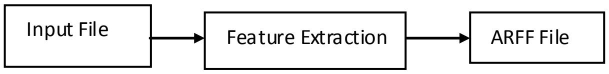


Fig.7: Reduced Feature Sets

4. *Classification*

Naive Baye's classifier is a probabilistic classifier based on Bayes theorem with independence assumptions, i.e., the different features in the data set are assumed not to be dependent of each other. This of course, is seldom true for real-life applications. Nevertheless, the algorithm has shown good performance for a wide variety of complex problems. J 48 is a decision tree-based learning algorithm. During classification, it adopts a top-down approach and traverses a tree for classification of any instance. Moreover, Random Forest is an ensemble learner. In this ensemble, a collection of decision trees are generated to obtain a model that may give better predictions than a single decision tree.

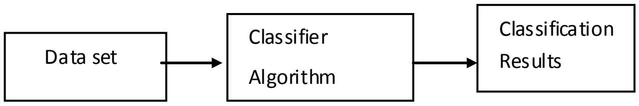
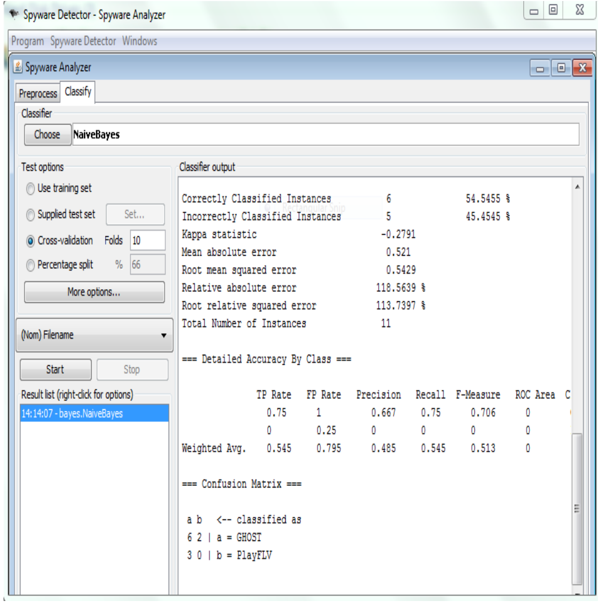


Fig.8: Top-down approach for classification


Spyware Detector - Spyware Analyzer
Program Spyware Detector Windows
Spyware Analyzer
Preprocess
Classify
Choose
Start
Stop
More options...

Classifier: NaiveBayes

Test options:
○ Use training set
○ Supplied test set Set...
● Cross-validation Folds 10
○ Percentage split % 66

(Nom) Filename

Result list (right-click for options): 14:14:07 - bayes.NaiveBayes

Classifier output

```
Correctly Classified Instances           6               54.5455 %
Incorrectly Classified Instances         5               45.4545 %
Kappa statistic                         -0.2791
Mean absolute error                      0.521
Root mean squared error                  0.5429
Relative absolute error                118.5639 %
Root relative squared error            113.7397 %
Total Number of Instances               11

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  C
                 0.75      1          0.667     0.75      0.706      0
                 0         0.25       0         0         0          0
Weighted Avg.    0.545     0.795      0.485     0.545     0.513      0

=== Confusion Matrix ===

 a b   <-- classified as
 6 2 | a = GHOST
 3 0 | b = PlayFLV
```

Fig.9: Results for Naive Bayes

iii) Hardware Platform

This part consists of ARM core processor as a main unit, GSM system, Ignition unit , PC and a camera. This module with designing and implementation technique is given below. ARM processor is used for controlling the overall system. Here we are using the LPC2148 series, which has two UART. In UART0 we will interface the PC for image processing. Then the ignition driver circuit is connected to the GPIO pin of ARM. Interrupt routine code is used to check whether we are getting any serial interrupt from matlab platform or not. UART is connected to GSM modem.

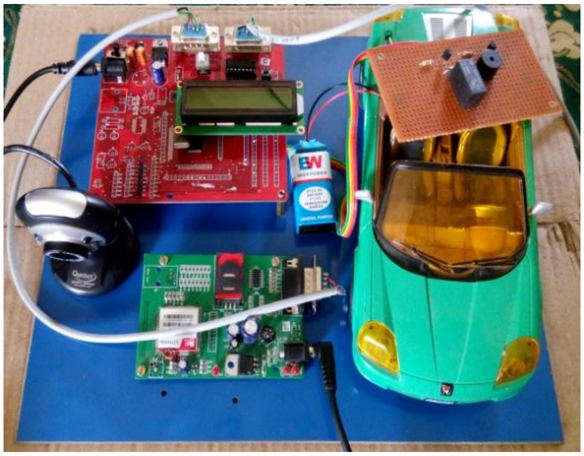


Fig 10: hardware platform of the project.

These interrupt checking method needs to configure the vector address. So the vector address configurations for both UART are given below. The Vectored Interrupt Controller (VIC) takes 32 interrupt request inputs and directly programmable assigns them vectored IRQ. VICIntSelect is a register which have the control of all interrupt registers. As we are using the UART0 interrupt and UART1 interrupt we have to just enable the 6th and 7th bit of the VICIntSelect register. After enabling for each interrupts separate slot have to be enabled for processing. So whenever an interrupt is coming from the Owner , then ARM processor can directly jumb to the interrupt routine to processing the command.. Because of this facility ARM can handle the different interrupts from the Owner and can do the respective functions without any fault.

In this project the engine unit will be controlled by a driver circuit. The driver circuit consists of a relay, resistor and a transistor. If the car is started, the engine will be turned ON which means ARM processor will give the bias voltage to the transistor to switch on the relay which inturn switch on the car engine. Meanwhile the processor will check the interrupt routine. Once if it receives the interrupt 'S' through UART then the processor will cut the bias voltage to the transistor. So that, the engine gets turned off.

IV. WIRELESS PLATFORM

1. GSM Overview:

A GSM modem is a wireless modem that works with a GSM wireless network. Global system for mobile communication (GSM) is a globally accepted standard for digital cellular communication.

GSM is the name of a standardization group established in 1982 to create a common European mobile telephone standard that would formulate specifications for a pan-European mobile cellular radio system operating at 900 mhz.

GSM modems support an extended set of AT commands. These extended AT commands are defined in the GSM standards.

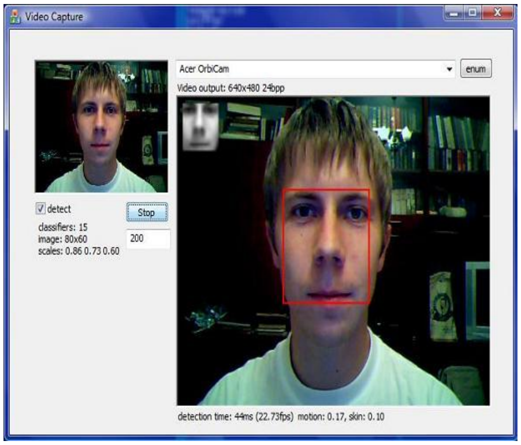
To send the SMS message, type the following command:AT+CMGS="+31638740161" <ENTER>

Replace the above phone number with your own cell phone number. The modem will respond with:

You can now type the message text and send the message using the <CTRL>-<Z> key combination: TEST GSM ! <CTRL-Z>

Here CTRL-Z is keyword for sending an sms through the mobile device.After some seconds the modem will respond with the message ID of the message, indicating that the message was sent correctly: +CMGS: 62

V. EXPERMENTAL RESULTS



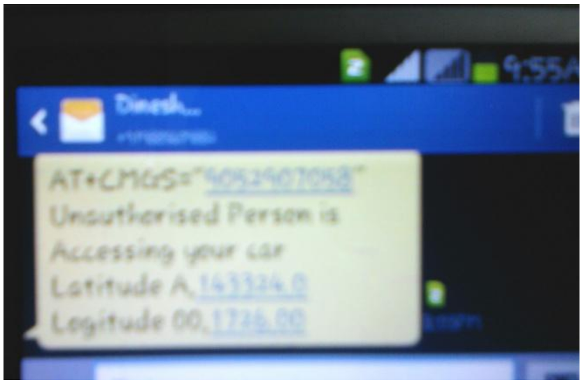
Acer OrbiCam
Video output: 640x480 24bpp
classifiers: 15
image: 80x60
scales: 0.86 0.73 0.60
200
detection time: 44ms (22.73fps) motion: 0.17, skin: 0.10

Fig 11: Face Detection Module

The connected modules correctly by use RS-232 cables and wires. In this hardware part we can see mainly three parts ARM processor, GSM module, motor & webcam. So first we are taking images by using webcam and these can be stored in database.

Database capacity is depend upon owner convenience, these images are authorized. When ever person enter into the car the webcam take the snap automatically which should be fixed infornt of the driver seat. The captured image should verify the database images, if it is matched car will star.

If any unauthorized person get into the car,and want to strat the car. The unauthorized person's image did not match the database images.so car will not be started. And at the same time gsm module activated and sends a message to the owner shown below.



AT+CMGS="9052907058"
Unsutherised Person is
Accessing your car
Latitude A,143324.0
Logitude 00,1736.00

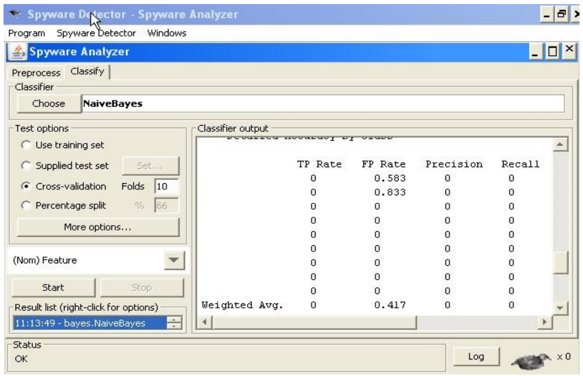
Fig 12: Owner get the massage from the GSM module.

MALWARE RESULTS:

This malware results is purely software dependent. In this project we are using weka tool to detect the malware. In this weka tool the results are totally probability based. We have data mining mechanism in weka tool. Data mining is a future extraction from the given files. For example we have thousand of files in your database, at that we extract the infected files from this we are using data mining technique.

In this weka tool we are having more than 256 algorithms. In our project we mainly concentrate on three algorithms namely navey bias, j48 and random forest. Based on the performance these are very useful to detect malware with in a fraction of seconds. The below figures shows us true positive rate and false positive rate of three different algorithms.

	NAÏVE BIAS		J48		RANDOM FOREST	
	F.N	F.F	F.N	F.F	F.N	F.F
T.P(true positive) Rate	0.353	0	0.471	0	0.353	0
F.P(false positive) Rate	0.416	0.417	0.417	0.417	0.283	0.25
Exicution time(in sec)	0	0	0.03	0	0	0
Correctly classified instanses	6	0	8	0	6	0
InCorrectly classified instanses	11	17	9	17	11	17



Classifier: NaiveBayes

Test options: ○ Use training set ○ Supplied test set Set... ● Cross-validation Folds 10 ○ Percentage split % 66

(Nom) Feature

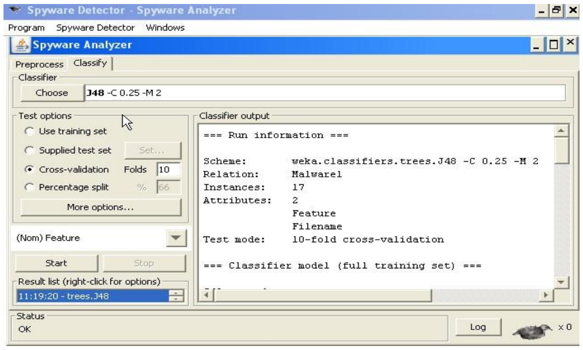
Result list (right-click for options): 11:13:49 - bayes.NaiveBayes

Status: OK

Classifier output:

```
                 TP Rate   FP Rate   Precision   Recall
                   0        0.583      0          0
                   0        0.833      0          0
                   0        0          0          0
                   0        0          0          0
                   0        0          0          0
                   0        0          0          0
                   0        0          0          0
                   0        0          0          0
                   0        0          0          0
Weighted Avg.      0        0.417      0          0
```

Fig.13: TP and FP rates of naïve bias algorithm.



Classifier: J48 -C 0.25 -M 2

Test options: ○ Use training set ○ Supplied test set Set... ● Cross-validation Folds 10 ○ Percentage split % 66

(Nom) Feature

Result list (right-click for options): 11:19:20 - trees.J48

Status: OK

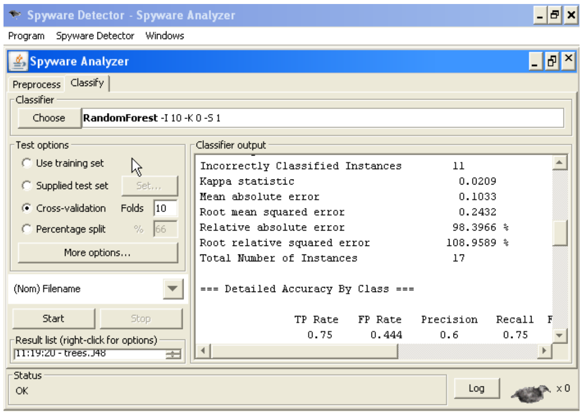
Classifier output:

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     Malware1
Instances:    17
Attributes:   2
              Feature
              Filename
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===
```

Fig. 14: malware detection of J48 algorithm.



Classifier: RandomForest -I 10 -K 0 -S 1

Test options: ○ Use training set ○ Supplied test set Set... ● Cross-validation Folds 10 ○ Percentage split % 66

(Nom) Filename

Result list (right-click for options): 11:19:20 - trees.J48

Status: OK

Classifier output:

```
Incorrectly Classified Instances        11
Kappa statistic                          0.0209
Mean absolute error                      0.1033
Root mean squared error                  0.2432
Relative absolute error                 98.3966 %
Root relative squared error            108.9589 %
Total Number of Instances               17

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall   F
                0.75      0.444      0.6        0.75
```

Fig 15: malware detection of random forest algorithm.

In every time we check the malware infection in two ways that is file name and file feature. So that the values of TP and FP rates also changed. The values are shown below.

F.N – FILE NAME, F.F – FILE FEATURE

Table 1: T.P, F.P, execution time, correctly and incorrectly classified instances of different algorithms

VI. CONCLUSION

As Security systems are becoming the unavoidable systems, this proposed project can bring a solution in system malware identification and image based authentication. Person authentication system can provide the important functions required by advanced intelligent Car Security, to avoid vehicle theft and protect the usage of unauthenticated users. Malware detection will helps to avoid the system crash. This project will help to reduce the complexity and improve security, also much cheaper and smarter than traditional ones.

VII. REFERENCES

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wirelessmultimedia sensor networks," *Computer Networks*, vol. 51, no. 4, pp.921–960, 2007.

[2] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S.Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, "Comprehensiveexperimental analyses of automotive attack surfaces," in *Proc.20th USENIX Conf. Security*, Berkeley, CA, USA, 2011, pp. 6–6.

[3] J. H. Lee and I. B. Jung, "Reliable asynchronous image transfer protocolin wireless multimedia sensor networks," *Sensors*, vol. 10, pp. 1486–1510, 2010.

[4] R. Pon, M. Batalin, M. Rahimi, Y. Yu, D. Estrin, G. Pottie,M. Srivastava,G. Sukhatme, and W. Kaiser, "Self-aware distributed embeddedsystems," in *Proc. IEEE Int.Workshop Future Trends Distrib. Comput.Syst.*, 2004, pp. 102–107.

[5] N. Bird, S. Atev, N. Caramelli, R. Martin, O. Masoud, and N. Papanikolopoulos,"Real time, online detection of abandoned objectsin public areas," in *Proc. IEEE Conf. Robot. Automat.*, 2006, pp.3775–3780.

[6] D. Li, Y. Jiang, and G. Chen, "A low cost embedded color visionsystem based on SX52," in *Proc. IEEE Int. Conf. Inf. Acquisit.*, Aug. 2006, pp. 883–887.