Non Functional requirement analysis in IoT based Smart Traffic Management System

Shubham N. Mahalank
Research Student
School of Electronics and Communication Engineering
KLE Technological University
Hubli, India
Email: shubhammahalank@gmail.com

Keertikumar B. Malagund
Research Student
School of Electronics and Communication Engineering
KLE Technological University
Hubli, India
Email: kbmalagund@gmail.com

R. M. Banakar
Professor, Member IEEE
School of Electronics and Communication Engineering
KLE Technological University
Hubli, India
Email: banakar@bvb.edu

***Abstract*—Non Functional requirement analysis is done for the Internet of Things (IoT) based traffic management unit which indicates the traffic density. Few quality characteristics of the design are analyzed during the development process. Design model decisions are governed by these Non Functional Requirement (NFR) design parameters. These quality characteristics considered are cost, sensitivity, design complexity, storage capacity, development process, response criteria and environmental impact. Having analyzed the quality attributes the design components are selected to deploy the design unit. Considerable effort needs to be put at the system design level to streamline the Internet of Things based design process. A Non Functional Requirement Analysis template documentation and checklist form is generated in this approach.**

Keywords: ***Non Functional Requirement Analysis, Traffic Management, Internet of Things, Traffic Density***

1. Introduction

There are two views with which any IoT based system design commences. The design starts with the Requirement Analysis. The two views are: Functional Requirement and Non-Functional Requirement. Functional Requirement is the action that the IoT based Traffic Density indicator as a system can perform. Non-Functional Requirements are the qualities possessed by the design unit which assists in the brand building of the design unit.

Understanding and studying the non-functional requirements (NFRs) and further implementing the solutions in a design model is a key part of involving NFRs in the development process. Due to lot of variance in basic non-functional requirement, it is a tough task to bridge the gap between requirement analysis and software design. In [1] authors have proposed a pattern-based approach to design NFRs and further integrate the design result into existing functional models. Non-functional requirements are emerging as one of the most important factors for successful information system development in service oriented organizations. Non-functional requirements involve not only software characteristics but also number of elements which govern the design module. In [2] authors proposes a conflict analysis method for non-functional requirements of information systems. This method includes ontologies, metadata, and rules.

In [3] the main purpose is to focus on the lack of software non-functional requirement describing methods during software design phase. A software non-functional requirements describing model is built up. The model is made up of three templates, namely attributes, constraints and operations templates. On this basis, the software non-functional requirements data model structure is defined. Requirements Analysis is considered as the most important phase for the development of quality software because errors caused by poor and inadequate requirement analysis are likely to slow down the design process and implementation phase. These errors are observed as time-consuming and most expensive to repair. Thus, Requirement Analysis is the phase, which determines the success or failure of a software project. Poor requirement analysis increases the cost of development by 70-85 percent, which requires reworking on all phases of software project. It is observed that non-functional properties are often ignored while focusing on the functionality of the software. Many software systems have failed because of negligence of non-functional requirements. Therefore, it is necessary to measure the satisfaction level of non-functional requirement during software development process as mentioned in [4].

In Web Service research, providing methods and tools for automatic composition of services based on functional requirements has gained a lot of attention in present world. However less effort have focused on integrating Non-Functional requirement in web service composition process and some solutions have been proposed to deal with this challenge. In [5] the authors have proposed a way for user's non-functional requirements with functional requirements while composing a user's desired web service design model. For large software projects, system designers have to stick to a significant number of functional and non-functional requirements, which makes software development a complex engineering task. If these requirements change during the development process, complexity increases. In [6] authors suggest systems based on context-aware composition to enable a system designer to postpone and automate decisions regarding efficiency of non-functional requirements, such as performance, and focus on the design of the core functionality of the system instead.

Non-Functional (NF) requirement is very important for the success of a software service. Considering this into account there could be multiple services implementing a same function, hence it will be important for software providers to understand the real Non Functional demands from consumers so that they can meet these demands and attract the users [7]. The use of Model Driven Development (MDD) is increasing in industry day-by-day. When a Non Functional Requirement (NFR) is not considered in the development must be added meta models, models and also transformations are affected [8]. Although the term 'non-functional requirement' has been in use for more than 20 years, there is still no any agreement in the requirements engineering community what non-functional requirements are and how we should document and validate them. In [9] authors survey the existing definitions of the term, highlights and discusses the problems with the current definitions, and contributes concepts for overcoming these problems.

In [10] author introduces an information retrieval based approach for automating the detection and classification of non-functional requirements (NFRs). Early detection of NFRs is useful because it enables system level constraints to be considered and incorporated into early architectural designs as opposed to being considered in at a later time. NFRs can be detected in both structured and unstructured documents, including requirements specifications that contain scattered and non-categorized NFRs, and free form documents such as meeting minutes, interview notes, and memos containing stakeholder comments documenting their NFR related needs. Authors in [10] describe about the classification algorithm and then evaluate its efficiency in an experiment based on fifteen requirements specifications developed as term projects by MS students at DePaul University. In [11], authors have proposed a framework for quantifying non-functional requirements (NFRs). This framework uses quality characteristics of the execution domain, application domain and component architectures to refine qualitative requirements into quantifiable ones. Conflicts are resolved during the refinement process and more concrete non-functional requirements are produced.

One of the main challenges in handling Non-Functional Requirements (NFRs) in designing systems is to take into account their dependency on each other. For this reason, they cannot be considered in isolation or as a single entity and a careful balance among them should be established for an effective framework. This makes it a difficult task to select design decisions and features that lead to the satisfaction of all different NFRs in the system, which becomes even more difficult when the complexity of a system grows. In [13] authors have introduced an approach based on fuzzy logic and decision support systems that helps to identify different design alternatives that lead to higher satisfaction of NFRs in the system. This is achieved by constructing a model of the NFRs and then performing analysis on the model.

Although the non functional requirement analysis has often been mentioned in embedded research, the context of IoT based system has not been sufficiently examined. Thus some important differences among embedded design and IoT have not been recognised or investigated. The main contribution of a work is the usage of survey method framework to develop a modular approach for IoT systems. Non functional requirement attributes based on the data collected show that these attributes can have a different influence on the benefits of IoT data acquisition infrastructure development. Our results also contribute to the system design knowledge of the unit in early stages of the model development.

2. Non Functional Requirement Analysis Framework in IoT based System

The requirement of NFRA in IoT system design is distinct, since the design relies on physical components, network protocols and software integration. An important design requirement analysis in IoT is an organised approach to examine the IoT infrastructure requirements very clearly, since any changes in the later stages incurs huge cost. System level design changes increase the design and implementation time to a larger extent. The response and the behavior of the system needs detained planning. The IoT infrastructure mainly consists of the data acquisition unit in the IoT based Traffic Density Indication (TDI). Fig. 1 represent the data acquisition platform for IoT based systems.

Successful choice of a good IoT data acquisition system is critical, since the design has to deliver reliable communication. The features considered in our IoT design process using NFRA are, Performance, Storage Capacity and Maintainability Constraints. The discussion illustrates that the non functional requirement analysis play a vital role in selecting the appropriate system components. The success of the IoT system design choice depends on the careful investigation of these requirements.

Considering the multi-disciplinary need in the IoT system, it is crucial for the IoT system designers to meet the user demand of performance and attract customer community. In this work, NFRA is done and a recommendation response in the form of system level design sheet is developed. The experimental results shows the feasibility of this approach, since a prototype of the Smart Traffic density identification system is deployed and tested for functionality.

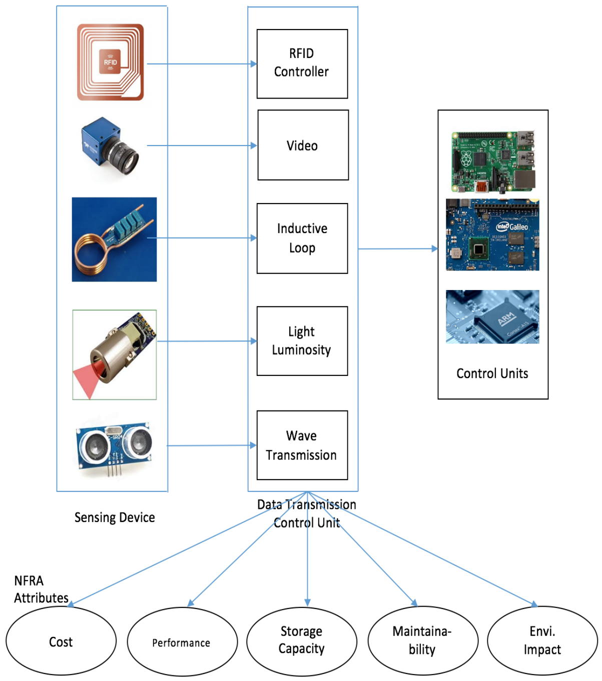

Figure 1. Data Acquisition platform for IoT based designs

3. Performance and Maintainability Constraint

In IoT based systems designing the system, integrating the modules and further implementing a system in real time environment together constitute to form a development process. The development process depends on number of factors like, design complexity, components used in data acquisition system, transmission medium, front end notification system for users, etc. The development process involved in different systems like Infrared Sensors, Video Analysis, Inductive Loop Detection System, Ultrasonic Sensors and RFID System is discussed briefly.

Infrared sensors which form the part of data acquisition model in a system are used to detect vehicles, road surfaces and other objects. These sensors identify the obstacle by transmitting the infrared wave which uon reception are converted to electric signals. These electric signals are further sent to the processing unit. Prior to that, these sensors are mounted beside the road near pavement at every intersection. They can be used to transmit information regarding signal control, detection of pedestrians in crosswalks and transmission of traffic information. These signals are transfered to the processing unit where it is stored, processed and further transmitted to the users to notify the information. The main disadvantage of a system is, operation of system can be affected by fog and also the implementation and maintenance factor are non-reliable. The basic disadvantages of infrared sensors are that the operation of the system may be affected due to fog, also installation and maintenance of the system is tedious.

Inductive loop detection system works on the principle of electromagnetic induction, where one or more turns of wire are placed in a cutout near roadway. In this system, there is a lead in wire connecting the roadside pullbox to the controller. Whenever a vehicle passes over or stops near the loop, there is change in induction with respect to the coil. If there is change in induction, the value of electric signal is altered i.e., the value of frequency is changed. This deviation in the value of frequency from the normal values causes the control unit to send a signal to the controller which processes and transmits the information to the users. It can be used to know the vehicle presence, number of vehicles passing in a particular area. The main disadvantage of a system is a poor durability due to improper connections and due to application of sealant over the cutout of the road. In some cases, where the system is implemented in poor pavement or where digging of roads is more often, the problem of reliability comes into existence.

Video analysis consists of a smart camera placed near the traffic signals at the top along with a processing unit and transmission unit. In this system the traffic is monitored continuously using a camera which records the video. The video captured is compressed in order to reduce its size for efficient transmission. The video analysis is performed using the techniques of digital image processing and computer graphics. The analysed data obtained contains the information about frequency of vehicles, average speed of vehicles, lane occupancy, etc. This data obtained can be stored and processed in control unit which is further sent to users to convey the information. The problems associated with video analysis are (a) the overall cost of the system is quite high (b) the system gets affected in case of heavy fog or rains (c) night time surveillance requires proper street lighting.

RFID Systems stand for Radio Frequency Identification System. It mainly consists of two parts, RFID Controller and RFID Tag. The main part of RFID controller is Interrogator, which acts as communication medium between RFID Controller and RFID Tag. The main controller part is present inside the RFID Controller. The RFID Controller works in accordance with the Interrogator where the processing depends on configuration. The controller part can perform read or write operations or both depending upon the configuration of the controller with respect to tag. The RFID controller further sends the data to IoT infrastructure where the data is transmitted with internet as a medium. RFID tags are the wireless modules that transmit the electromagnetic radio waves which are used for identifying and tracking of vehicles. Usually there are two types of RFID tags, Active and Passive. Active RFID Tags have an inbuilt power source, where as passive RFID Tags have to be provided with an external power supply. RFID Tag has a transmitter and receiver end which transmit the radio waves and upon receptance calculate the time duration. Each RFID has a serial number for unique identification.

In a system where Ultrasonic Sensors are the part of Data Acquisition System are used to detect the Traffic Density. Ultrasonic Sensors are placed beside roadways especially near the junction where three to four roads intersect. These sensors have both the transmitting and receiving end on the same side unlike that of infrared sensors where the transmitter and receiver are at the opposite extreme ends. The sensors transmit the ultrasonic wave, upon hit by an obstacle, the wave reflects back and the waves are accepted by the receiver end. If the vehicle obstacles the normal transmission transmission of ultrasonic wave in the prescribed range, than there is change in time duration of transmission . This change is considered as a presence of an obstacle in front of it. Upon detection the sensor transmits values to the processor unit. After processing the information is transmitted to the users.

The factor of scalability is also important in order to implement a system in large scale. Ultrasonic Sensors and its system can be produced in large scale to meet the requirements, hence there is no issue with scalability with this system.

4. Storage Unit (Capacity)

1. Infrared Sensor:

The input data from the sensor in the data acquisition system design is a binary data. The binary data '0' indicates that there is no traffic or vehicle as an obstruction. The binary value '1' indicates that a vehicle is moving. This data is sent to the processing module. Hence the storage model is given by,

$$Meminfra = num(bits) \tag{1}$$

Where num is in terms of bits and only a register is enough to store the data value. The processing time is very less when the input data is sent to the IoT infrastructure.

2. Inductive Loop Detection:

In inductive loop data is in the form of inductive electric signal where in turn there is change in frequency. Here the data is obtained from the inductive loops placed beside the roads which form the part of data acquisition system. In this system, change in frequency due to change in induction is identified which causes the electronic unit to send a signal indicating the presence of a vehicle [18]. This method is helpful in knowing the vehicle presence and even the number of vehicles passing through a particular area [19]. Here the data is transmitted in the form of signals with particular frequency which has to be stored in a procesing unit for further operations. Hence, the storage model equation is given by,

$$Meminfra = "num"(bits/bytes) \tag{2}$$

Where num is in terms of bits or bytes. Here the processing time is moderate when the input data is sent to IoT infrastructure. Here memory storage with a register is sufficient to store the data values.

3. Ultrasonic Sensor:

The input data is from the Ultrasonic Sensors which form the part of data acquisition model. Here the sensors transmit the Ultrasonic wave and upon reception at the receiver end, sensors calculate the time taken for transmission and hence evaluate it. Like Infrared sensors, in Ultrasonic sensors the data is in the form of binary values '0' and '1' which becomes more convenient for processing in control unit. The value obtained can be employed to get the information about Traffic Density, detection of pedestrians on footpath and also transmission of traffic information. The data values obtained are further sent to processing unit where the data is stored temporarily or permanently as per the requirements. Hence the storage model equation is given by,

$$Meminfra = num(bits) \tag{3}$$

Here num is in terms of bits, hence a register is sufficient to store the data value. The processing time is very less when the input is sent to IoT infrastructure in case of Ultrasonic Sensors.

The main advantage of employing Ultrasonic Sensor and its system is the characteristic of portability i.e., the system can also use Infrared Sensors or RFID Tags to detect the traffic density.

4. Video Analysis:

Here the input data is from the smart camera in the data acquisition model, where data is in the form of video which can be compressed to reduce the transmission bandwidth. Here the Video Analysis is performed to get the description and hence to compute traffic statistics like number of vehicles passing, average speed of vehicles in that particular lane, lane occupancy, etc. This data is further sent to the processing unit where it is stored and transmitted to required units as and when required. Hence the storage model equation is given by,

$$Meminfra = num(KB/MB) \tag{4}$$

Where num is in tems of KBs or MBs which usually depend on the length and quality of a video. Here processing time is very high when the input data is sent to IoT infrastructure. Usually memory storage units like sRAM and dRAM are required to store the data values.

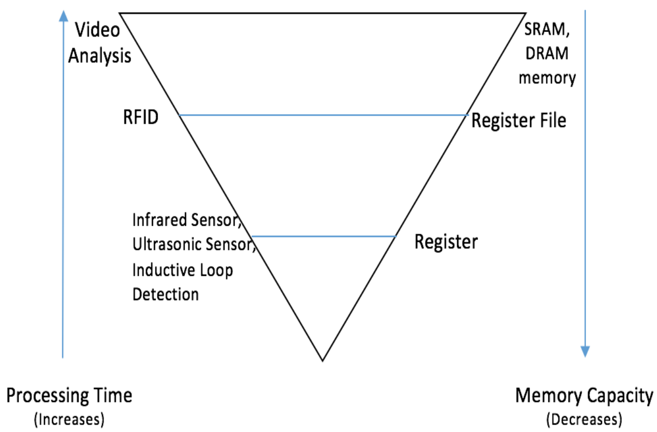

Figure 2. Memory Model

NON-FUNCTIOANAL REQUIREMENT MAPPING TEMPLATE

Design Title	Mapping of Non-Functional Requirement #1
Designer Name	
Affiliation	
Design Date	05-01-2016

Name: Non Functional requirement #1 in Data Acquisition System for Traffic Density Identification.

Summary: The Ultrasonic sensing in the form of wave provide better signal strength.

Rationale: If the signal strength is not good, then the traffic density indicator performance may be weak.

Requirements: A cost effective ultrasonic sensor is required at each cross road section with a IoT based unit to transmit a vehicular movement at the traffic management junction.

Figure 3. Template 1 for Performance

5. RFID System:

In RFID System the input data is from the combination of RFID Controller and RFID Tag. Here RFID tags are the wireless devices which make use of radio frequency waves to detect the vehicles passing. Hence RFID tag form the part of data acquisition model. In this system the value of radio frequency wave is transmitted to the processing unit where the data is stored. Hence the storage equation is given by,

$$Meminfra = num(bits) \tag{5}$$

Where num is in terms of bits, hence only a register is sufficient to store the data values. Here the processing time is moderate when the input data is sent ti IoT infrastructure.

Fig. 2 represents the memory model with respect to different different data acquisition alternatives and their comparative variance in terms of Processing time (Performance) and Memory capacity.

5. NFRA Output Response

The NFRA Output Response is a practical Checklist which describes how well the IoT based Traffic Management System works. An elegant way to describe the non functional quality attribute is to quantify it. Normally portability is one of the features which is addressed in NFRA process. In the design of IoT based Traffic Management Unit the design portability is more suited from Ultrasonic Sensor based unit to either Infrared sensor Unit design RFID based design unit.

The other minimal non-functional checklist obtained from NFRA as an output response are as follows:
1. Performance: Data Acquisition Model in accordance with Raspberry Pi provide better information about Traffic Density
2. Capacity: Data Value to be transmitted is in terms of bits.
3. Deployment: Implementation of Data Acquisition model on field level is more convenient as more reliable results can be obtained
4. Environmental Protection: Aesthetic Robust design for Data Acquisition Unit to protect especially from road accidents and pedestrians.
5. Maintainability Constraint: The wired connection between the Data Acquisition Model and Control Unit is the major constraint to maintain.
6. Signal Strength (Frequency):Data Acquisition Unit should have sensors with higher frequency waves for better sensitivity and detection.
7. Design Consideration: Careful attention should be given for packaging the design unit

Using NFRA facilitates in building better hardware design model. One can own a lot of success in the deployment process to build IoT based system.

An example of how to arrive at a functional requirement from NFRA is given below. For our design model in [6] Andrew Stellman in 2010 has described a sample NFRA to functional requirement template as in Fig. 3 and Fig. 4.

6. Conclusion

The process followed in selecting the design model components provides a general well integrated framework within which the deployment details can be distinctly featured. By employing Non Functional Requirement Analysis the quality attributes the quality attributes for the Internet of Things based Traffic Management Unit may be specified in terms of its developmental, learning and hardware modeling context. The description arrived at is sufficient to provide design framework boundary, detailed enough to be used as a ready reckoner within the deployment context. The

NON-FUNCTIOANAL REQUIREMENT MAPPING TEMPLATE

Design Title	Mapping of Non-Functional Requirement #5
Designer Name	
Affiliation	
Design Date	05-01-2016

Name: Non Functional requirement #5 in Data Acquisition System for Traffic Density Identification .

Summary: The wired connection between the data acquisition unit and control unit is the major constraint to be taken care of after deployment .

Rationale: If there is no protection for the wired connection, then there will be no proper connection between the units or sometimes null .

Requirements: An aesthetic design to protect the wires along with the data acquisition unit from various external factors .

Figure 4. Template 2 for Maintainability Constraint

emphasis is to answer the basic question "Is this the right prototype/product?". An elegant approach which can be used professionally is discussed. It is the methodical approach, which is a unique specifiable design path that assists to reliably choose a course of action during the deployment phase. The minimal checklist obtained using Non Functional Requirement Analysis serves as a guideline to implement the design.

References

[1] Yi Liu, Zhiyi Ma, Rui Qiu, Hongjie Chen, Weizhong Shao, " An Approach to Integrating Non-functional Requirements into UML Design Models Based on NFR-Specific Patterns," 2012 12th International Conference on Quality Software (QSIC), pp. 132 - 135, Aug. 2012.

[2] Chi-Lun Liu, "Ontology-Based Conflict Analysis Method in Non-functional Requirements," 2010 IEEE/ACIS 9th International Conference on Computer and Information Science (ICIS), pp. 491 - 496, Aug. 2010.

[3] Ye Fei, Zhu Xiaodong, "An XML-Based Software Non-Functional Requirements Modeling Method," ICEMI '07. 8th International Conference on Electronic Measurement and Instruments, 2007, pp. 2-375 - 2-380, July 2007.

[4] Khatter. K, Kalia. A, "Quantification of non-functional requirements," 2014 Seventh International Conference on Contemporary Computing (IC3), pp. 224-229, Aug. 2014.

[5] El Kassmi. I, Jarir. Z, "An approach to weave seamlessly non-functional requirements with functional requirements in Web Service composition," 2015 International Conference on Protocol Engineering (ICPE) and International Conference on New Technologies of Distributed Systems (NTDS), pp. 1 - 6, July 2015.

[6] Danylenko A., Lowe W., "Context-aware recommender systems for non-functional requirements," 2012 Third International Workshop on Recommendation Systems for Software Engineering (RSSE), pp. 80 - 84, June 2012.

[7] Xiao-Lin Zhang, Chi-Hung Chi, Chen Ding, Wong R.K., "Non-functional Requirement Analysis and Recommendation for Software Services," 2013 IEEE 20th International Conference on Web Services (ICWS), pp. 555 - 562, July 2013.

[8] Agirre J.A., Sagardui G., Etxeberria, L, "Evolving legacy model transformations to aggregate non functional requirements of the domain," 2015 3rd International Conference on Model-Driven Engineering and Software Development (MODELSWARD), pp. 437 - 448, Feb 2015.

[9] Bajpai V. Gorthi R.P., "On Non-Functional Requirements," 2012 IEEE Students' Conference on Electrical, Electronics and Computer Science (SCEECS), pp. 1 - 4, Mar. 2012.

[10] Cleland-Huang J., Settimi R., Xuchang Zou, Solc P, "The Detection and Classification of Non-Functional Requirements with Application to Early Aspects, 14th IEEE International Conference Engineering, pp. 39 - 48, Sept. 2006.

[11] Hill R., Jun Wang, Nahrstedt K., "Quantifying non-functional requirements: a process oriented approach, 12th IEEE International Requirements Engineering Conference, pp. 352 - 353, Sept. 2004.

[12] Tabassum M. R., Siddik M. S., Shoyaib M., Khaled S. M., "Determining interdependency among non-functional requirements to reduce conflict, 2014 International Conference on Informatics, Electronics and Vision (ICIEV), pp. 1 - 6, May 2014.

[13] Saadatmand M., Tahvili S., "A Fuzzy Decision Support Approach for Model-Based Tradeoff Analysis of Non-functional Requirements, 2015 12th International Conference on Information Technology - New Generations (ITNG), pp. 112 - 121, April 2015.

[14] Ganesan S., Sugumaran V., "Non Functional Requirements for IT Convergence and Infrastructure, 2011 First ACIS/JNU International Conference on Computers, Networks, Systems and Industrial Engineering (CNSI), pp. 452 - 457, May 2011.

[15] Abdukalykov R., Hussain I., Kassab M., Ormandjieva O., "Quantifying the Impact of Different Non-functional Requirements and Problem Domains on Software Effort Estimation, 2011 9th International Conference on Software Engineering Research, Management and Applications (SERA), pp. 158 - 165, Aug. 2011.

[16] Rohleder C., "Representing Non-functional Requirements on Services - A Case Study, 2012 IEEE 36th Annual Computer Software and Applications Conference Workshops (COMPSACW), pp. 359 - 366, July 2012.

[17] Bouain A., El Fazziki A., Sadgal M., "Integration of non-functional requirements in a service-oriented and model-driven approach, 2014 IEEE Eighth International Conference on Research Challenges in Information Science (RCIS), pp. 1 - 8, May 2014.

[18] US7245220 B2, "Radio frequency identification (RFID) controller", http://www.google.com/patents/US7245220, Jul. 2007.

[19] Ali, George B., Vanajakshi L., "A simple multiple loop sensor configuration for vehicle detection in an undisciplined traffic Sensing Technology," 2011 Fifth International Conference on Sensing Technology (ICST), pp. 644 - 649, Dec. 2011.

[20] Bing-Fei Wu, "A new Approach to Video-based Traffic surveillance using fuzzy hybrid Information Inference Mechanism," IEEE Intelligent Traffic Society, March 2013.

[21] Xianbin Cao, Changxia Wu, Jinhe Lan, Pingkun Yan, "Vehicle Detection and Motion Analysis in Low-Altitude Airborne Video Under Urban Environment Circuits and Systems for Video Technology," IEEE Transactions on Circuits and Systems for Video Technology, pp. 1522 - 1533, July 2011.

[22] "Detection Technology," IVHS Volume-1, http://ntl.bts.gov/DOCS/96100/index.html, July 1995.

[23] Hussain, T.M., Saadawi T. N., Ahmed S. A., "Overhead infrared sensor for monitoring vehicular traffic," IEEE Transactions on Vehicular Technology, pp. 477 - 483, Nov. 1993.

[24] Ninad L., Sheetal K., "Smart Traffic Management System," International Journal of Computer Applications, Vol. 75, No. 7, Aug 2013.