A QoS Control Method for Camera Network based People Detection Systems

Toru Abe[1], Adrian Agusta[2], Yuto Mitsuhashi[3] and Takuo Suganuma[1]
[1] *Cyberscience Center, Tohoku University, 2-1-1 Katahira, Aoba-ku, Sendai, 980-8577 Japan*
[2] *Electronics System Development Division, Yamaha Motor Co., Ltd., 2500 Shingai, Iwata, Shizuoka, 438-8501 Japan*
[3] *Graduate School of Information Sciences, Tohoku University, 2-1-1 Katahira, Aoba-ku, Sendai, 980-8577 Japan*

Keywords: Camera Network, People Detection System, QoS Control.

Abstract: Various people detection systems based on camera networks have been developed, and their services (output of users' locations) are utilized in a variety of applications. Usually, each application requires a people detection system to keep its quality-of-service (QoS) at a certain level. However, required system QoS levels vary widely among different applications, and the QoS requirements of each application range over various QoS factors, such as the coverage area, resolution, and frequency of users' locations. Moreover, the trade-off between QoS factors arises from limitations on the system resources, which fluctuate due to changes in circumstances. Consequently, it is difficult for such systems to stably fulfill the diverse QoS requirements of individual applications. To deal with these difficulties, we propose a QoS control method for camera network based people detection systems. Taking into account the trade-off between several QoS factors under limited and varied system resources, our method dynamically adjusts system parameters and controls system QoS to provide each application with users' locations at a required QoS level. Experimental results indicate that our method well maintain system QoS for the changes in application requirements and system resources.

1 INTRODUCTION

Recently, various applications which utilize users' locations obtained through sensor networks have been proposed for a variety of fields including security surveillance, smart home care, environment monitoring, etc. For these purposes, camera network based people detection systems are widely used (Valera and Velastin, 2005; Song et al., 2011; Wang, 2013).

In a people detection system based on a camera network, images captured by cameras are transmitted via a network, users' locations are estimated on a server from these images, and then the estimation results are provided to an application as the system service. Usually, the available system resources for communications and computing are limited and varied, which affect the quality-of-service (QoS) of the people detection system, whereas each application utilizing users' locations requires the people detection system to keep its QoS at a certain level.

To fulfill the various QoS requirements of individual applications under limited and varied system resources, several approaches have been proposed to camera network based people detection systems. One of the most popular approaches is the introduction of hierarchical architecture, which aims at the effective utilization of limited system resources by localizing the communications and computing of lower-level image data (Micheloni et al., 2008; Karuppiah et al., 2010). Several approaches reduce consumption of system resources by selecting part of cameras in the network and assigning tasks to them (Casares and Velipasalar, 2011; Dieber et al., 2011). Another approach adaptively adjusts system parameters taking into account the trade-off between consumption of the system resources and QoS requirements of the applications (Hengstler and Aghajan, 2007; Micheloni et al., 2008; Wang et al., 2010).

However, required QoS levels for the camera network based people detection system vary widely among different applications, and the QoS requirements of each application range over various QoS factors, such as the coverage area, output resolution, and output frequency of users' locations. Moreover, the trade-off between these QoS factors arises from limitations on the system resources, which fluctuate due to changes in circumstances. Consequently, it is difficult for such people detection systems to stably fulfill

Abe T., Agusta A., Mitsuhashi Y. and Suganuma T..
A QoS Control Method for Camera Network based People Detection Systems.
DOI: 10.5220/0004845202420248
In *Proceedings of the 9th International Conference on Computer Vision Theory and Applications* (VISAPP-2014), pages 242-248
ISBN: 978-989-758-009-3

the diverse QoS requirements of individual applications under limited and varied system resources.

To deal with these difficulties, we propose a QoS control method for camera network based people detection systems. Taking into account the trade-off between QoS factors under limited and varied system resources, our proposed method dynamically adjusts system parameters and control system QoS to provide each application with users' locations at a required QoS level. Experimental results indicate that our method can well adapt system QoS to the changes in QoS requirements and system resources.

2 CAMERA NETWORK BASED PEOPLE DETECTION SYSTEM

2.1 System Configuration

Figure 1 shows the supposed configuration of a people detection system, which consists of a camera network and a main server. In this system, images captured by the cameras are transmitted via the network, users' locations are estimated on the main server from these images, and then the estimation results are provided to an application as services of the system.

Usually, each application requires such a people detection system to keep its QoS at a certain level. Those required system QoS levels vary widely among different applications, and the QoS requirements of each application range over various QoS factors, such as the coverage area, accuracy, resolution, delay, and frequency of users' locations (Hengstler and Aghajan, 2007). For example, in security surveillance applications, the coverage area of users' locations is an important QoS factor, although the resolution is rarely a key issue (Moeslund and Granum, 2001). On the contrary, in smart home care applications which control something by users' locations, the delay and frequency are critical issues, however, the coverage area is less important.

In addition, the trade-off between QoS factors arises from limitations on the system resources for communications and computing. Since the available system resources vary with disturbances (e.g. miscellaneous traffic and processes), several QoS factors are affected by changes in circumstances.

Accordingly, to make people detection systems more serviceable, they need a QoS control method which takes into account the trade-off between QoS factors under limited and varied system resources while adjusting system parameters, such as the number of cameras, size of image, frame rate of image, and network bandwidth.

2.2 Relation Model between QoS Factors and System Parameters

For the people detection system, there are many QoS factors and system parameters, which are intricately interrelated. For example, a QoS factor "coverage area" is determined by various system parameters (e.g., the number, placement, and specifications of cameras), and "output frequency" is affected by the trade-off between QoS factors and the fluctuation in available system resources due to not only disturbances but also parameter adjustments themselves.

In this paper, we focus on the coverage area, output resolution, and output frequency of users' locations as the QoS factors, and the number of cameras, size of image, frame rate of image as the system parameters. Furthermore, we simplify the relations between these QoS factors and system parameters as summarized in Figure 2. This relation model supposes that the number of cameras and size of image directly control (i.e. correspond one-to-one with) the coverage area and output resolution, respectively. On the other hand, the output frequency is supposed not

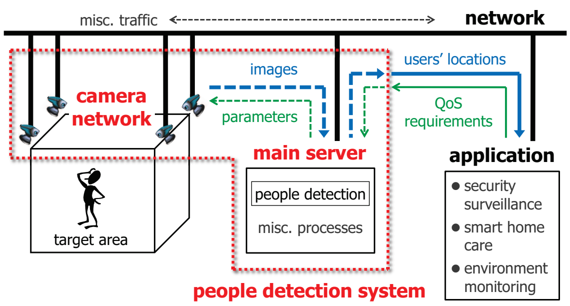

Figure 1: Supposed configuration of a people detection system based on a camera network.

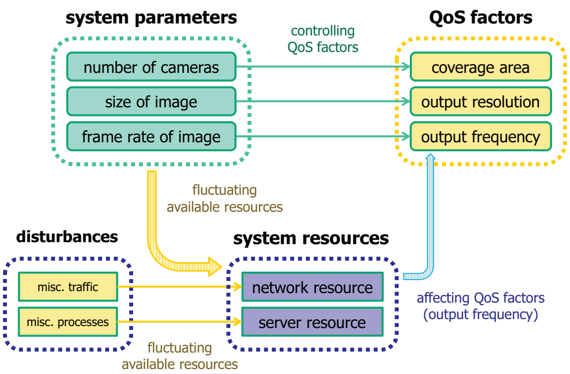

Figure 2: Simplified relation model between QoS factors and system parameters of people detection systems.

only to be controlled by the frame rate of image but also to be affected by the trade-off between QoS factors and the system resource fluctuation due to disturbances and parameter adjustments.

3 QoS CONTROL OF CAMERA NETWORK BASED PEOPLE DETECTION SYSTEMS

3.1 QoS Control Method

Figure 3 shows the structure of our proposed QoS control method for camera network based people detection systems. This method is implemented as cameras, people detection part, and QoS control part.

QoS requirements, which specify required levels and priority order for QoS factors, are supplied from an application in advance. Required levels for the coverage area, output resolution, and output frequency are measured by the number of cameras, size of image, and times per unit time, respectively.

With the initial system parameters, the cameras capture images of the target area, and transmit them to the people detection part on the main server. The people detection part estimates users' locations from the received images, and provides the application with the estimation results as the system output. The QoS control part compares the QoS requirements of the application to the states of the people detection part, which contain observed levels of the QoS factors. According to the differences in the QoS factors between the QoS requirements and the people detection states, the system parameters are adjusted, and then sent to the cameras and the people detection part.

With the adjusted system parameters, the cameras and the people detection part change their own settings (i.e., activating or deactivating each individual camera, setting image size, and setting image frame rate). Thus, the next series of the processes starts.

3.2 System Parameter Adjustment

Figure 4 shows the flow of system parameter adjustment in our method. Based on the relation model in 2.2, our method adjusts the system parameters in priority order of the QoS factors. For example, when the highest priority is assigned to a QoS factor "output frequency," a system parameter "frame rate of image" is iteratively modified to reduce the difference in "output frequency" between the QoS requirements and the people detection states. Following this, remaining adjustments are carried out for lower priority QoS factors.

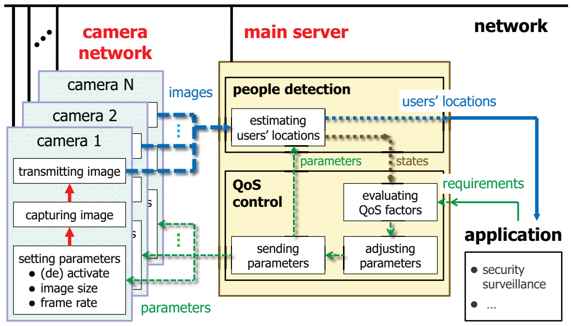

Figure 3: Structure of the proposed QoS control method for camera network based people detection systems.

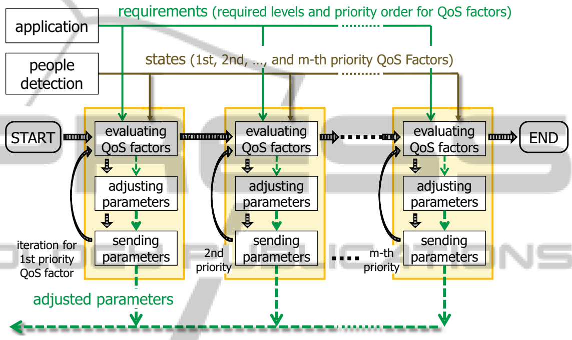

Figure 4: Flow of system parameter adjusting.

tors. Every adjustment is iterated until the difference in its target QoS factor does not decrease or the differences in higher priority QoS factors increase.

As described in 2.2, we suppose that the QoS factors are controlled directly by the system parameters. However, as the number of cameras and/or the size of image increase, the output frequency is affected by the QoS factor trade-off and the system resource fluctuation. Therefore, it is difficult to control the QoS factors directly in those cases.

To achieve appropriate QoS control, we approximate the relation between the QoS factors. Suppose that all cameras are set to the same size and the same frame rate of image. If the people detection part outputs a set of users' locations after receiving the images from all activated cameras, then the maximum output frequency (OF) is inversely proportional to the cover area (CA) controlled by the number of cameras and the output resolution (OR) controlled by the size of image. Consequently, the relation between these QoS factors is expressed in

$$\mathrm{OF} = \frac{\alpha}{\mathrm{CA} \times \mathrm{OR}}, \tag{1}$$

where α is a coefficient depending on the system performance. By substituting observed levels of the QoS factors into Eq. (1) in order of their priorities, target levels can be determined for lower priority QoS factors, and amounts for system parameter adjustments can be estimated from these results. Since α varies with the available system resources, its initial value is set by the calibration of a target people detection system, and then updated iteratively by the observed levels of CA, OR, and OF.

4 EXPERIMENTS

4.1 Experimental Environment

To demonstrate the effectiveness of our proposed method, we conducted QoS control experiments on a camera network based people detection system.

Figure 5 shows the structure of the prototype system used in the experiments. This system consists of a main server and an image server, which are connected with each other through 100Mbps Ethernet. The specifications of those servers are listed in Table 1. The main server has a QoS control module and a people detection module, while the image server has camera simulation modules. Those modules are implemented in C++. To manage experimental conditions, instead of an actual camera network, the image server (each camera simulation module) transmits images to the main server (people detection module).

The QoS control module is manually provided with QoS requirements (the required levels and priority order for the coverage area (CA), output resolution (OR), and output frequency (OF)). According to the states obtained from the people detection module, system parameters (the number of cameras, size of image, and frame rate of image) to fulfill the QoS requirements are computed in the QoS control module and sent to other modules. The people detection module is based on "`pedestrian_detect`" in "OpenCV-2.4.3 GPU demos pack" (Bradski et al., 2012) which is modified to receive multiple image sequences, estimate users' locations from them, and change its own setting with parameters from the QoS control module. In the camera simulation modules, videos captured

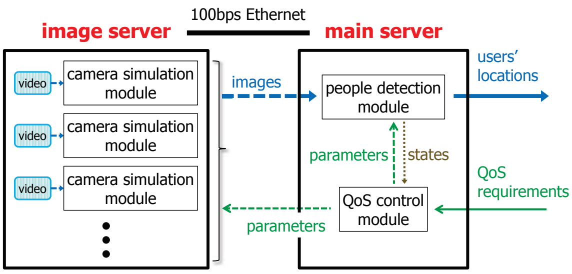

Figure 5: Structure of the prototype system.

Table 1: Specifications of servers.

main server	
CPU	Intel Core i7-870 (2.93GHz)
GPU	NVIDIA GeForce GT 430
memory	8GB
OS	Windows 7 Professional x64
image server	
CPU	Intel Core i5-560M (2.66GHz)
memory	8GB
OS	Windows 7 Professional x64

Figure 6: Examples of people detection results by the prototype system (green rectangles represent detected locations).

with multi-camera (CVLab-EPFL, 2012) are used for transmission, where the size and frame rate of images are changed with parameters from the QoS control module (the original size and frame rate of each video are 360×288 pixels and 25 fps, respectively).

Examples of people detection results (users' locations) by the prototype system are shown in Figure 6.

4.2 Experimental Results

4.2.1 Baseline Performance

Firstly, we carried out experiments to evaluate the baseline performance of the prototype system.

In the experiments, output frequency (OF) is evaluated by varying the number of cameras and the size of image. The number of activated cameras (camera simulation modules transmitting images) is varied from one to eight, and the size of transmitted image is varied from 144×115 to 720×576 pixels (i.e., the magnification for the original size 360×288 pixels is varied from 0.4 to 2). The frame rate of each camera is fixed at 25 fps. Experimental results are summarized in Figure 7. Since the people detection module

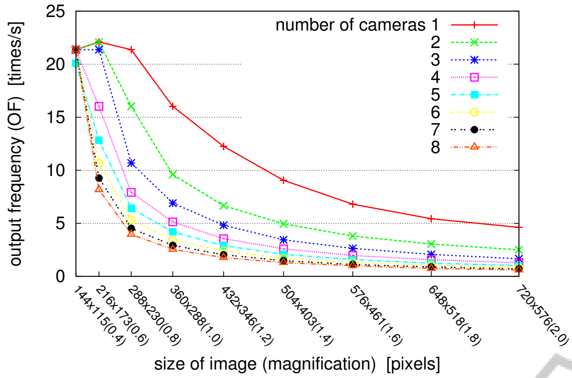

Figure 7: Output frequency (OF) by varying the number of cameras and the size of image.

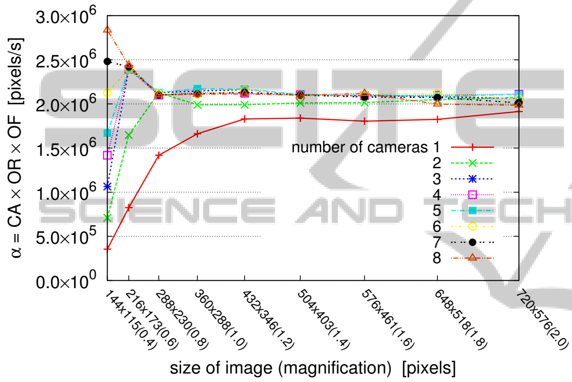

Figure 8: Coefficient α in Eq. (1) estimated from observed levels of the QoS factors.

outputs a set of users' locations after receiving the images from all activated cameras, OF is not more than a frame rate of 25 fps. Naturally, because of limitations on the system resources for communications and computing, OF decreases as the number of cameras and/or the size of image increase.

Multiplying the individual results by the number of cameras and the size of image, we compute the number of pixels processed on the main server per unit time, which correspond $\alpha = CA \times OR \times OF$ in Eq. (1). Obtained results are summarized in Figure 8. Except for cases where the number of cameras and the size of image are small, obtained $CA \times OR \times OF$ are fairly constant regardless of the number of cameras or the size of image. Consequently, by calibrating the system for a certain number of cameras in advance, the obtained $CA \times OR \times OF$ can be used for adjusting system parameters as the initial value of α in Eq. (1).

4.2.2 Adaptation to QoS Requirement Changes

To demonstrate the effectiveness of our proposed method in adapting to QoS requirement changes, we

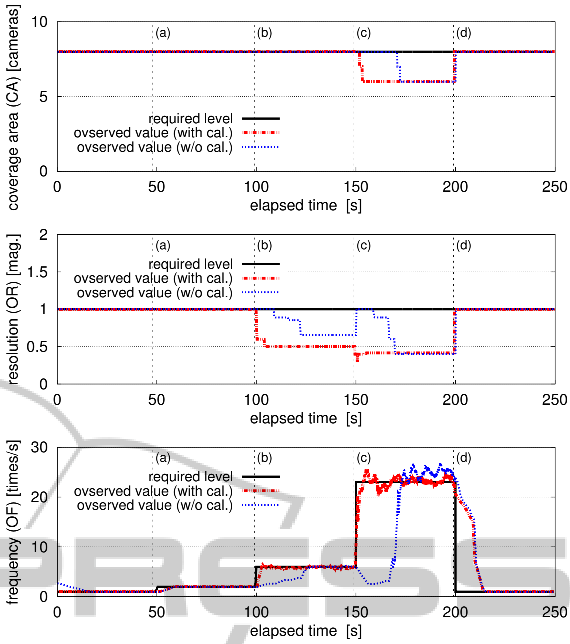

Figure 9: Adaptation to QoS requirement changes (required level for OF is changed at (a), (b), (c), and (d)).

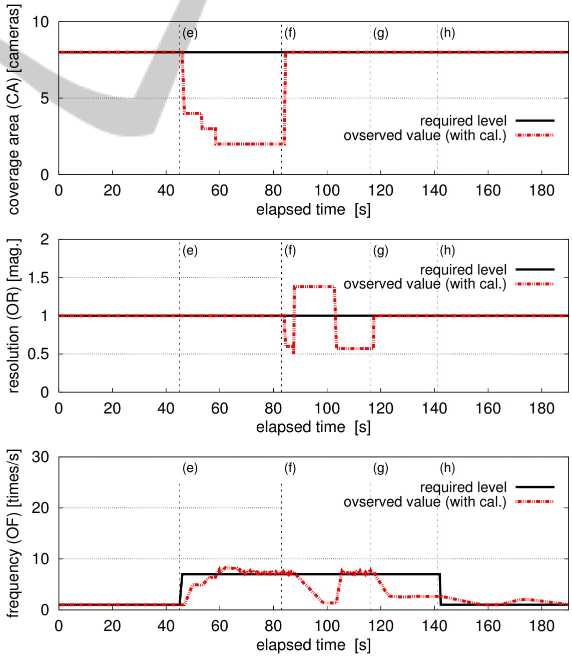

Figure 10: Adaptation to changes in required level for OF and priority order for QoS factors at (e), (f), (g), and (h).

conducted QoS control experiments.

Observed levels of CA, OR, and OF by changing QoS requirements are shown in Figure 9. In this experiment, the required level for OF is changed from 1 time/s to (a) 2 times/s at 48s → (b) 6 times/s at 99s →

(c) 23 times/s at 149s $\rightarrow$ (d) 1 time/s at 199s, whereas the required levels for CA and OR are fixed at 8 cameras and 1 magnification, respectively, and the priority order for QoS factors is fixed to OF>CA>OR. The QoS control is carried out with or without the prototype system calibration where CA $\times$ OR $\times$ OF $=$ 2.07×10^6 pixels/s is obtained for a set of eight cameras as the initial value of α in Eq. (1).

From (a) to (b) in Figure 9, since the required level for OF is rather low, the system parameters can be adjusted to its increase without the influence on CA or OR, and consequently all requirements for the QoS factors are fulfilled. From (b) to (c), the system parameters can be adjusted to an increase in the required level for OF, however, the lowest priority QoS factor OR is lowered due to limitations on the system resources. From (c) to (d), the system parameters are adjusted to a further increase in the required level for OF by decreasing the second priority QoS factor CA in addition to OR. As can be seen from Figure 9, the QoS control with the calibration adapts to the QoS requirement changes more quickly than the QoS control without the calibration.

Figure 10 shows the experimental result of the QoS control with the calibration by changing the priority order for QoS factors. In this experiment, the priority order is changed from OR>OF>CA to (f) OF>CA>OR at 83s $\rightarrow$ (g) CA>OR>OF at 116s. Meanwhile, the required levels for CA and OR are fixed at 8 cameras and 1 magnification, respectively, and the required level for OF is changed from 1 time/s to (e) 7 times/s at 45s $\rightarrow$ (h) 1 time/s at 141s.

From (e) to (f), by decreasing the lowest priority QoS factor CA, the QoS control adapts to an increase in the required level for the highest priority QoS factor OF. From (f) to (g), because the lowest priority QoS factor changes from CA to OR, the QoS control keeps OF at its required level by decreasing OR instead of CA. From (g) to (h), since OF is given the lowest priority, the QoS control keeps CA and OR at their required levels by decreasing OF.

These results indicate that our proposed method can keep the QoS factors of the people detection system at specified QoS levels in specified priority order.

4.2.3 Adaptation to System Resource Changes

To demonstrate the effectiveness of our proposed method in adapting to system resource changes, QoS control experiments were conducted.

In the experiments, the required levels for CA, OR, and OF are fixed at 8 cameras, 1 magnification, and 6 times/s, respectively, and, the priority order for QoS factors is fixed to OF>CA>OR.

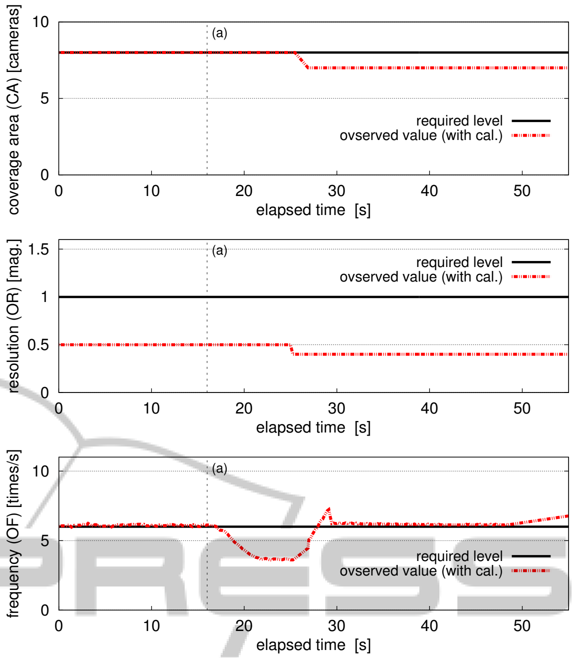

Figure 11: Adaptation to a server resource change at (a).

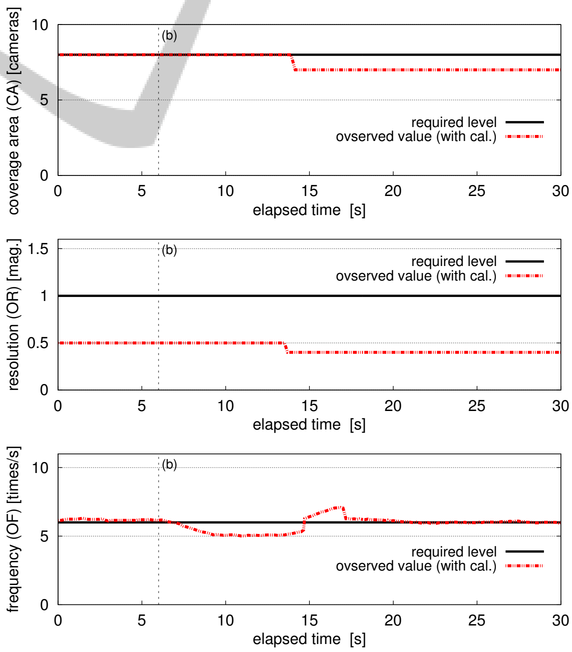

Figure 12: Adaptation to a network resource change at (b).

Figure 11 shows the experimental result, where a CPU load of 50% is imposed on the main server by a CPU load generator program in addition to loads of the QoS control module and the people detection module. On the other hand, Figure 12 shows the experimental result, where 66.6Mbps of traffic from a traffic generator program is imposed on the network in addition to the traffic between the QoS control, people detection, and camera simulation modules.

Currently, since neither the CPU load nor the network traffic is monitored directly by the prototype system, the server resource fluctuation due to an additional CPU load ((a) at 16s in Figure 11) or the network resource fluctuation due to additional network traffic ((b) at 6s in Figure 12) is detected as a decrease in OF. To keep the highest priority QoS factor OF at its required level, the QoS control decreases the lowest priority QoS factor OR in both the experiments. However, as OF cannot reach its required level in both cases, the QoS control decreases the second priority QoS factor CA in addition to OR.

These results indicate that our proposed method can adapt to the fluctuation in the available system resources for communications and computing.

5 CONCLUSIONS

In this paper, we have proposed a QoS control method for camera network based people detection systems. Taking into account the trade-off between several QoS factors under limited and varied system resources, our proposed method dynamically adjusts system parameters and controls system QoS. Through the experiments, we illustrated the effectiveness of our method in maintaining individual QoS factors for the changes in QoS requirements and system resources. Those results demonstrate that our method can keep the QoS factors of the people detection system at specified QoS levels in specified priority order. Consequently, our method can be expected to make the people detection system more serviceable for various applications utilizing users' locations.

Currently, our proposed method controls the coverage area, output resolution, and output frequency of users' locations as the QoS factors by adjusting the number of cameras, size of image, and frame rate of image as the system parameters through simplified their relation model. In future work, we would like to investigate extending our method to various other QoS factors (e.g., output accuracy, output delay, and power consumption), system parameters (e.g., camera placement, image coding, and network bandwidth), and more precise models of their relations.

ACKNOWLEDGEMENTS

This work was supported in part by the Japan Society for the Promotion of Science (JSPS) under a Grant-in-Aid for Scientific Research (C) (No.23500201).

REFERENCES

Bradski, G. et al. (2012). OpenCV | Free Science & Engineering software downloads at SourceForge.net. `http://sourceforge.net/projects/opencvlibrary/files/opencv-win/2.4.3/`. Online; accessed 1-Aug.-2013.

Casares, M. and Velipasalar, S. (2011). Adaptive methodologies for energy-efficient object detection and tracking with battery-powered embedded smart cameras. *IEEE Trans. Circuits Syst. Video Tec hnol.*, 21(10):1438–1452.

CVLab-EPFL (2012). Multi-camera pedestrians video | CVLAB. `http://cvlab.epfl.ch/data/pom`. Online; accessed 1-Aug.-2013.

Dieber, B., Micheloni, C., and Rinner, B. (2011). Resource-aware coverage and task assignment in visual sensor networks. *IEEE Trans. Circuits Syst. Video Tec hnol.*, 21(10):1424–1437.

Hengstler, S. and Aghajan, H. (2007). Application-oriented design of smart camera networks. In *First ACM/IEEE Int. Conf. Distrib. Smart Cameras*, pages 12–19.

Karuppiah, D. R., Grupen, R. A., Zhu, Z., and Hanson, A. R. (2010). Automatic resource allocation in a distributed camera network. *Mach. Vision Appl.*, 21(4):517–528.

Micheloni, C., Lestuzzi, M., and Foresti, G. L. (2008). Adaptive video communication for an intelligent distributed system: Tuning sensors parameters for surveillance purposes. *Mach. Vision Appl.*, 19(5-6):359–3733.

Moeslund, T. B. and Granum, E. (2001). A survey of computer vision-based human motion capture. *Comput. Vis. Image Understanding*, 81(3):231–268.

Song, B., Ding, C., Kamal, A. T., Farrell, J. A., and Roy-Chowdhury, A. K. (2011). Distributed camera networks. *IEEE Signal Process. Mag.*, 28(3):20–31.

Valera, M. and Velastin, S. A. (2005). Intelligent distributed surveillance systems: a review. *IEE Proc. Vis, Image Signal Process.*, 152(2):192–2004.

Wang, X. (2013). Intelligent multi-camera video surveillance: A review. *Pattern Recognit. Lett.*, 34(1):3–19.

Wang, Y., Velipasalar, S., and Casares, M. (2010). Cooperative object tracking and composite event detection with wireless embedded smart cameras. *IEEE Trans. Image Process.*, 19(10):2614–2633.