Proceedings of the 2014 Winter Simulation Conference
A. Tolk, S. Y. Diallo, I. O. Ryzhov, L. Yilmaz, S. Buckley, and J. A. Miller, eds.

FRUGAL SIGNAL CONTROL USING LOW RESOLUTION WEB-CAMERA AND TRAFFIC FLOW ESTIMATION

Kumiko Maeda
Tetsuro Morimura
Takayuki Katsuki
Masayoshi Teraguchi

IBM Research – Tokyo
NBF Toyosu Canal Front
6-52, Toyosu 5-chome
Koto-ku, Tokyo 135-8511, JAPAN

ABSTRACT

Due to rapid urbanization, large cities in developing countries have problems with heavy traffic congestion. International aid is being provided to construct modern traffic signal infrastructure. But often such an infrastructure does not work well due to the high operating and maintenance costs and the limited knowledge of the local engineers. In this paper, we propose a frugal signal control framework that uses image analysis to estimate traffic flows. It requires only low-cost Web cameras to support a signal control strategy based on the current traffic volume. We can estimate the traffic volumes of the roads near the traffic signals from a few observed points and then adjust the signal control. Through numerical experiments, we confirmed that the proposed framework can reduce an average travel time 20.6% compared to a fixed-time signal control even though the Web cameras are located at 500 m away from intersections.

1 INTRODUCTION

Traffic jams are often caused by traffic signals and various efforts have been paid to the problem over recent decades. Many adaptive control strategies have been proposed (Robertson and Bretherton 1991, Lowrie 1982, Mirchandani and Head 2001). However, most of them strongly depend on a detection system with high accuracy and suitably configured by a skilled engineer. Developing countries cannot apply the latest signal control technologies to reduce the traffic jams because of budget limitations, the skill levels limits of the local staff, and so on. Sometimes, they use fixed-time signal control with limited maintenance. People often ignore traffic signals with such poor control, so policemen must control the traffic manually at intersections during rush hours.

In this paper, we propose a frugal signal control framework which uses image analysis with low-cost Web cameras and traffic flow estimates from a limited number of observation points. We have already deployed a traffic awareness pilot system in Nairobi, Kenya, based on our prior work. Our pilot system broadcasts the traffic status information using Twitter or SMS and offers route recommendations. This paper proposes a design for a frugal signal control framework that will extend the pilot system. The system uses a signal control strategy based on observations obtained from a small number of low-cost Web cameras.

The spread of the Internet allows us to see current video images of vehicle traffic from Web cameras in city in developing countries (AccessKenya.com 2014, MMDA 2014). However, Web camera images have low resolutions and low frame-rates so it is hard to estimate the number of vehicles and their velocity, so we are proposing a method to use such low-quality images (Idé et al. 2013). The method uses a simple regression model combined with an optimal threshold for binarization instead of the standard template-matching approach. This allows us to estimate the vehicle flow at the locations monitored by the Web cameras sufficiently accurately even if the quality of the images is too low for template matching.

We also note that the number of Web cameras is insufficient for traffic signal control and the locations of the Web cameras are not determined by considering traffic control but are, for example, simply attached to convenient buildings near the congested roads. Therefore we use traffic estimation techniques (Morimura, Osogami, and Idé 2013) to assess the traffic volumes to control the signals. The goal is to estimate the traffic flow in arbitrary links of network, based on the observed traffic flows from a limited number of the links. The problem is similar to network tomography and link-cost prediction. However, unlike the network tomography (Santini 2000, Zhang et al. 2003), we need to infer all of the link traffic instead of source-destination demands, and, unlike link-cost prediction (Ziebart et al. 2008, Idé and Sugiyama 2011), our inputs are stationary observations instead of trajectories. Our approach is to approximate each driver's movement with a Markov-chain process for road network. The Markov-chain model is created by solving the inverse Markov chain problem with partial observations. By using the estimated model, we infer the traffic volumes for the roads near the traffic signals.

We extended our traffic awareness system with these techniques to produce a signal control strategy based on the estimated traffic volumes. In this paper, our basic heuristic is to make the green lights last longer on the busier streets to reduce the length of the traffic jam and even the delay of vehicles on each side of signals.

We describe simulations using the IBM Mega Traffic Simulator (Osogami et al. 2012) to see how well our techniques work for traffic signal control. The simulation results showed that our signal control framework reduces the average travel time by 20.6% compared to a fixed-time signal control and reduces the average delay 22.1%, even though the Web cameras were located 500 m from intersections. The observations of roads with lower and steadier varying traffic volumes were also important to derive good green lengths for the signals with our traffic flow estimation module.

The rest of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the structure of our framework and overviews of the image analysis system, the flow estimation, and the traffic signal control. Section 4 is about our simulations to assess the performance of signal control with the estimated traffic volumes and we conclude in Section 5.

2 RELATED WORK

There are many adaptive control research reports (Robertson and Bretherton 1991, Lowrie 1982, Mirchandani and Head 2001), but they have not been deployed widely due to the costs of implementation and maintenance. Most of the adaptive control systems rely on high-accuracy sensors. Some research focuses on robustness and fault-tolerance. Chiu and Chand (1993) propose a distributed adaptive control algorithm that is robust against faults at some of the intersections and Mirchandani and Head (2001) manage errors from the sensors by using a Bayesian model but neither considers sensors under such bad conditions as Web cameras. Yin (2008) proposes a robust signal control algorithm for pre-timed signals considering demand fluctuations. Our approach does not use pre-timed controls but the robustness may be useful for errors in the image analysis and the traffic estimates, so this approach could be combined with our system.

There has been some research into signal control using machine learning (Mikami and Kakazu 1994; Spall and Chin 1997; Srinivasan, Choy, and Cheu 2006). Mikami and Kakazu (1994) used genetic reinforcement learning to optimize large numbers of signals without extensive communications. A neural network approach proposed by Spall and Chin (1997) eliminated the complex traffic modeling that made the system unstable when the traffic changed due to weather influence, seasonal influence or other causes. Srinivasan, Choy, and Cheu (2006) focused on online learning for neural network-based traffic signal control. Our approach uses Markov-chain processes in estimating traffic flows to manage poor observation conditions. The flow estimation phase is independent of the signal control algorithm, so these machine learning-based signal control algorithms can be combined with our frugal control system and some of them might fit with our conditions.

3 OVERVIEW OF OUR FRAMEWORK

In this section, we present an overview of our proposed frugal signal control framework and its main components. Figure 1 shows the overview.

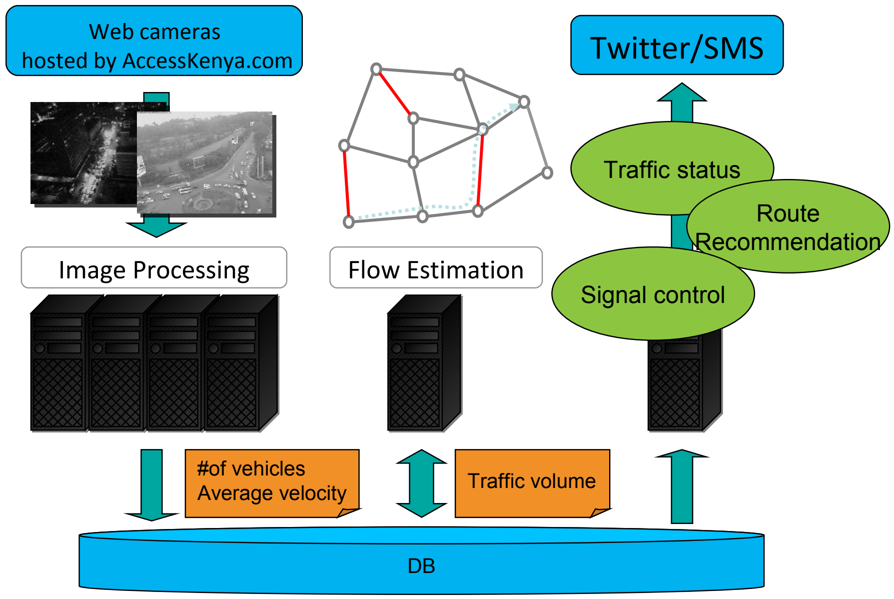

Figure 1: Frugal traffic framework overview.

For the Kenya pilot system that offers current traffic status reports and route recommendations through Twitter, SMS or other means, we used five Windows™machines and one Linux™machine, each of which has an Intel Geon™4 core processor and 4 GB of memory. The image processing module runs on four of the Windows™machines, the flow estimation module runs on one Windows™machine, and a DB is running on the Linux machine. First we receive jpeg images using HTTP from about 20 Web cameras provided by AccessKenya, with about 6-second intervals between the images and analyze them to estimate vehicle densities and speeds about 50 roads. The estimated densities and speeds are stored in the DB. The flow estimation module runs at 5-minute intervals using the last 10 minutes of densities and speeds as stored in the DB. This module estimates the traffic flows and average vehicle speeds for about 5,000 roads in an 8 sq. km. area around the center of the Nairobi. Our current pilot system offers current traffic status and route recommendation and we extend the system to support a signal control.

3.1 Traffic Camera Image Processing Module

It is desirable to utilize general-purpose Web cameras typically mounted on buildings due to cost and security concerns when deploying ITS systems in developing countries. Movies and images that can be obtained from Web cameras typically have very low resolutions (Figure 2 for examples) and very low frame-rates (1 frame each 6 seconds in our Kenya pilot system). Standard object recognition and template matching technologies such as those used in number plate recognition (Buch, Velastin, and Orwell 2011) are inapplicable.

Our image processing module uses sophisticated machine learning techniques (Idé et al. 2013) to tackle the problem instead of the standard template-matching approach. First, our module binarizes a image into a black-and-white image to highlight the vehicle areas with an optimizing threshold. Then we find a regression function relating the number of vehicles and the number of white pixels that represent the vehicle areas. Our algorithm works well for low-quality images and avoids the need for the individual

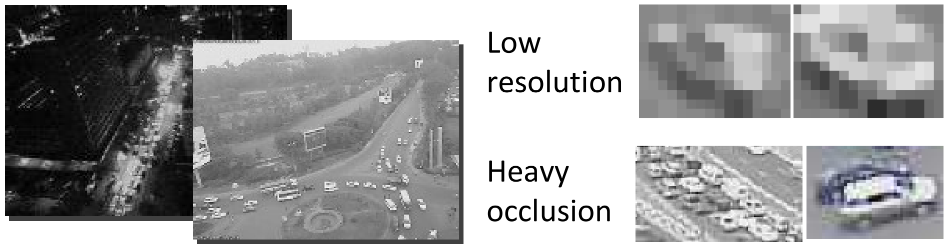

Figure 2: Low quality images from Web cameras.

vehicle recognition, and is also robust against the low frame-rates because our method does not use time dependencies between the frames such as is used with background subtraction.

Our module also offers a good traffic velocity estimation for low frame-rate observations from Web cameras. We use two approaches for velocity estimation, one approach based on the movement of a plane through the road and the other is a Bayesian approach using the time-sequential images.

3.2 Flow Estimation

We have a very limited number of observation points where the Web cameras are installed. It is useful for various applications if we can estimate the traffic conditions of the unobserved roads. Our flow estimation module solves the inverse problem of the Markov chain (Morimura, Osogami, and Idé 2013). From the partial observations at a limited number of states and links, it finds the Markov model with these parameters: (1) the initial-state probabilities that is the source-link preferences, (2) the state-transition probability matrix that is the inter-link transition probabilities for all of the intersections, and (3) the stopping probability of the trips. The observations include the number of vehicles that went through each observation point.

This module was developed using MATLAB$^{\text{TM}}$. We have two phases to run each estimation module, a training phase and an estimation phase. In the Kenya pilot system, the training phase used observation data from 20 points for 7 days and created training data separately for three time slots of weekdays and weekends. The training phase takes a few hours for all time slots but we do not need to run the training phase unless the traffic pattern changes, for example, when a new shopping mall opens. The estimation phase uses the latest 10 minutes of observed data for each point and takes about 10 seconds. The output of the module includes the number of vehicles passing through each road segment for a time period as well as an average vehicle velocity for each road in the given road network.

3.3 Signal Control

We simply used the estimated amount of traffic on the adjacent roads at each signal to define the control of the signals. We assume that we can change the split of the cycle of the signals periodically through a network connection. We determine the split of each signal by considering the estimated traffic volume using the formula

$$G_i = A + (C - nA)\frac{V_i}{\sum_{j=1..n} V_j}$$

where n is the number of phases of the signal, $G_i(i = 1..n)$ is a time period for phase i, C is the cycle length of the signal, A is the minimum length of each phase and V_i is the maximum traffic volume for the lanes that turn green on phase i. We use the traffic volume derived by the traffic estimation module as V_i. We simply define the length of each phase as G_i but that actually includes not only the green light time but the yellow light time, the clearance time, and so on. The cycle length C is defined by various factors such as the structure of the intersection, the presence of one or more protected turns, the total traffic volume, and so on (Koonce et al. 2008). We use a fixed cycle length and change the ratio of the time for each phase. The minimum length of each phase A is usually determined by the clearance time of each road as derived from the length of each side of the intersection, and so on (Koonce et al. 2008).

4 SIMULATIONS

In this section, we describe our simulations to assess the dynamic signal control system with limited observations. First, we investigated how well the signals are controlled and how much the traffic jams are reduced by varying the distances between the Web cameras and the intersections. As mentioned before, the locations of the Web cameras are not determined freely but are, for example, simply attached to convenient buildings near the congested roads. We used estimated traffic volumes to define the traffic signal parameters, but the accuracy of the traffic estimation decreases as the distance from the observation point increases. This simulation verifies whether the estimated traffic volumes as derived by the traffic estimation module are sufficiently accurate and whether our signal control system is sufficiently practical. Second, we investigated how each observation point affected the performance of the traffic signal control. We used the IBM Mega Traffic Simulator (Osogami et al. 2012) instead of a field experiment. Figure 3 shows the process flow of the simulations.

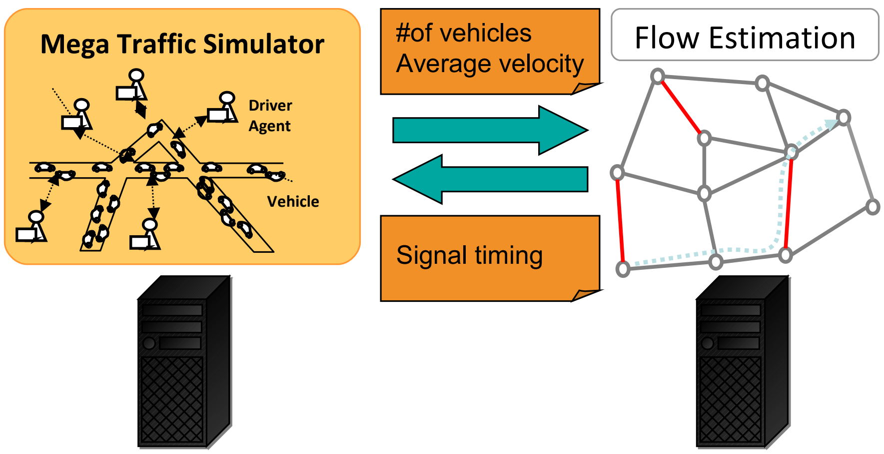

Figure 3: Simulation process flow.

Each simulation period was 600 seconds long (6 cycles of the traffic signals), and the green lights of the traffic signals were adjusted based on the results of the last simulation period to take effect in the next period. We paused the simulation after each 600 seconds, collected the statistics that corresponded to the image analysis results from the Web cameras (the densities and the average speeds of the vehicles on the observed roads) and ran a traffic flow estimation module to obtain the traffic volumes on all of the roads. The green light time was calculated based on the equation from Section 3.3 using the estimated values.

4.1 Traffic Simulator

The IBM Mega Traffic Simulator can simulate millions of vehicles in an entire city to evaluate the lengths of traffic congestion, the travel time of vehicles, the CO2 emissions, etc. The IBM Mega Traffic Simulator was developed on the X10-based multi-agent simulation platform XAXIS (Suzumura and Kanezashi 2013), which offers scalable multi-node simulations on the X86 and Power architectures. The IBM Mega Traffic Simulator includes tools that extract a road network structure from OpenStreepMap (OpenStreetMap 2014), generate vehicle trips from simple settings, and generate KML (Open Geospatial Consortium 2014) to visualize the simulation results so we can quickly create a simple simulation for any location in OpenStreetMap.

4.2 Simulation Settings

We simulated vehicles in Nairobi, Kenya, where our frugal ITS pilot system was deployed. The road structure graph of Nairobi has 3,434 nodes and 5,308 arcs. Some traffic signals have already been installed in Nairobi, but they are not yet in use. We assumed there are six traffic signals to be controlled, as shown in Figure 4 and that there are from 3 to 5 observation points for each signal. The total cycle length of each signal is 100 seconds, the minimum green light time was 10 seconds and the green light time for each phase is calculated by our signal control module as described in Section 3.3. In this experiment, the north-bound and south-bound lights turn green at the same time for all six signals. The timing of the east-bound and west-bound lights differ because the green light time of each signal is determined based on the traffic volume coming into the signal.

The dynamic signal control is useful when the balance of traffic volumes between each side changes. We changed the traffic balance during the simulations by adding vehicles from the north to the south and from the south to the north to verify whether our proposed method can respond effectively to manage the changing traffic flows. The total length of each simulation was 3 hours. A total of 9,636 vehicles travelled in the 8 km square region for 3 hours and an additional 2,000 vehicles that travelled from north to south or from south to north departed from their origins at times from 30 to 60 minutes after the start of the simulation. The traffic from 0 to 30 minutes represents the time period when vehicles travel smoothly and the additional 2,000 vehicles represent commuters who could cause traffic jams.

Figure 4: Simulation map of Nairobi. The streets on the map are based upon OpenStreetMap data (©OpenStreetMap contributors) and licensed under ODbL. ©2014 Google Image ©2014 DigitalGlobe http://www.google.com/permissions/geoguidelines/attr-guide.html

4.3 Simulation Results

Figure 5 shows the average travel times of the vehicles that pass through each side of the signals while varying the distance of the camera from the intersection. We tested 0 m, 500 m, 750 m and 1,000 m. We investigated whether or not the traffic jams are reduced for each distance of the cameras. The case of 0 m represents the case when we can get completely accurate observations of the traffic volumes. Most existing systems propose comprehensive signal control algorithms assuming accurate observations. The columns "fix" in Figure 5 are the results where the green length times of the signals are determined based on the total traffic volume over the 3 hours and do not change during the simulation. The labels on the x-axis

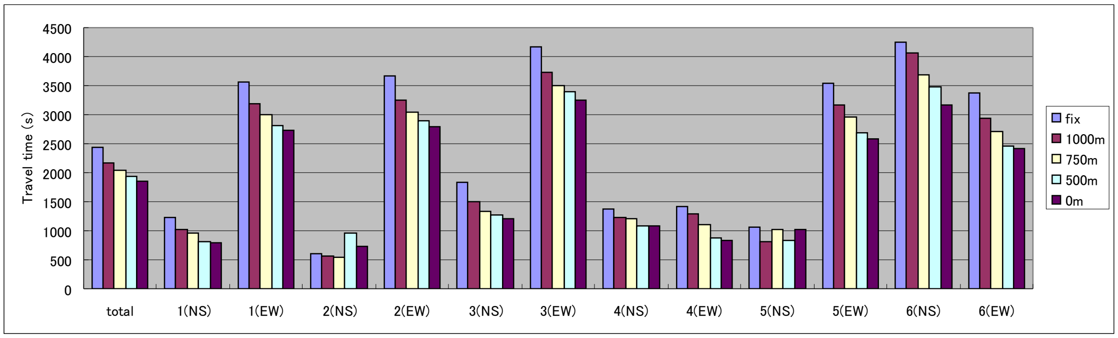

Figure 5: Travel time of vehicles.

represents the IDs of the traffic signals in Figure 4 and the directions of the traffic signals, where "NS" means north or south side and the "EW" means east or west side. The column "total" is the average travel time of the vehicles that pass through more than one of the signals.

The differences between the estimated and the actual traffic volumes increase as the camera position becomes father from the intersection. The total travel time decreases by 20.6% for 500 m and by 11.1% for 1,000 m compared to the case of "fix". The travel time for 500 m increases by 3.8% compared to that for 0 m, when we had precisely accurate traffic volumes. The estimated traffic volumes are sufficiently accurate for the dynamic control of the traffic signals.

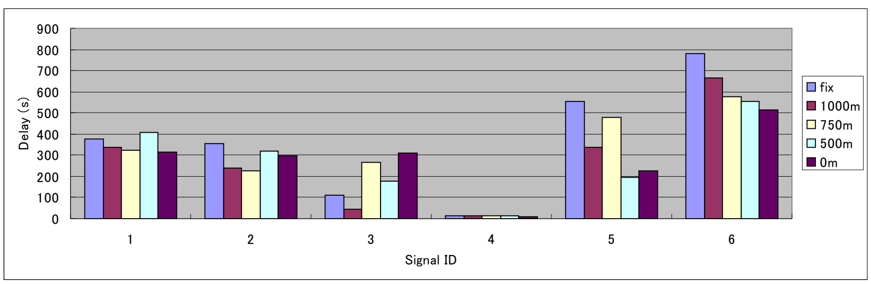

Figure 6: Delay at each intersection.

Figure 6 shows the delay time, which is the elapsed time spent driving at a speed of less than 10 km/h at each signal. The delay decreases at the two traffic signals in the south (The numbers 5 and 6 in Figure 6). Much more traffic from the south is handled at the southern signals with the dynamic signal control, so the traffic jams were moved to the north side. The average delay for all of the intersections in the 500-m case decreased by 82.7 seconds (22.1%) compared to the fixed control approach. The delay at the signal No. 5 in the 750-m case is long. The observation point in that case was just at the sharp curve and the lower observed speed led to the difference between the estimated and the actual traffic volume. If we add some calibration techniques for the reported speeds from the Web cameras, we may be able to achieve better performance.

Figure 7 shows the results of tests of removing an observation point from one of the four directions at signal No. 6. This tested whether our system could maintain good performance with a smaller number of Web cameras. The case "w/o E" removes the east-side camera, "w/o W" removes the west-side camera, "w/o S" removes the south-side camera and "w/o N" removes the north-side camera. The performance was worse without some of the observations on the east or west sides while it was actually improved without some data from the south or north. Figure 8 shows the estimated and actual traffic volumes after 90 minutes of simulation for each side of the signal with and without observations. The estimated traffic volume is around 0.2 vehicles/s without a nearby observation point. The volume is influenced by other

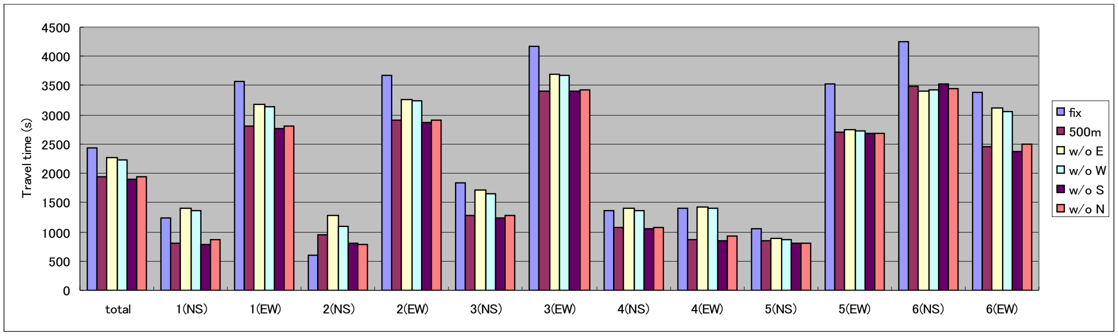

Figure 7: Travel time of vehicles.

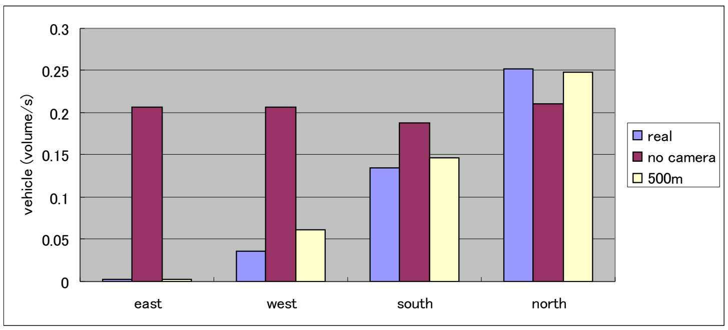

Figure 8: Traffic volume of signal No. 6 with and without observations.

nearby observations. The actual traffic of the north side approaches the correct value so the performance without the north observations is good. The timing of the signal split is determined by the maximum traffic volume of the roads, so the observations of the south side do not affect the travel times. The observations of the roads on which the traffic volumes are low and do not change much are also required to obtain good traffic signal control using the flow estimation (or some other reference value of traffic volumes may also be acceptable instead of observations).

We confirmed that our dynamic signal control with distant Web cameras was able to respond to the changes of traffic volume with good performance through the simulations. Using the existing rich signal control algorithms and field studies to verify the errors of image processing still remains as our future work. However, the proposed method using Web cameras can support an inexpensive dynamic signal control system.

5 CONCLUSION

We are working on a frugal traffic signal control framework which uses image analysis and traffic flow estimation. Our image processing module utilizing binarization works sufficiently well with low-quality images obtained from low-cost Web cameras and our traffic flow estimation module offers enough high precision for a traffic signal control with rather limited observations. In simulations, our frugal traffic signal control reduced the average travel time by 20.6% with 26 Web cameras located at 500 m distance from the intersections. Of course the performance depends on the scenario, so it is important to find suitable numbers and locations of Web cameras required for the traffic signal control. However, the result shows the possibility deploying a frugal and dynamic traffic signal control infrastructure supported by image analysis and flow estimation. Our future work will include an evaluation with more practical traffic signal control strategies, considering errors in the image analysis and evaluating robustness against the various errors for the signal control.

6 ACKNOWLEDGEMENTS

This research was supported by JST, CREST.

REFERENCES

AccessKenya.com 2014. "AccessKenya.com". http://traffic.accesskenya.com/.

Buch, N., S. A. Velastin, and J. Orwell. 2011. "A review of computer vision techniques for the analysis of urban traffic". *IEEE Transactions on Intelligent Transportation Systems* 12 (3): 920–939.

Chiu, S., and S. Chand. 1993. "Adaptive traffic signal control using fuzzy logic". In *Proceedings of the 2nd IEEE International Conference on Fuzzy Systems*, 1371–1376. IEEE.

Idé, T., T. Katsuki, T. Morimura, and R. Morris. 2013. "Monitoring Entire-City Traffic using Low-Resolution Web Cameras". In *Proceedings of the 20th ITS World Congress, Tokyo*.

Idé, T., and M. Sugiyama. 2011. "Trajectory Regression on Road Networks". In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, 203–208.

Koonce, P., L. Rodegerdts, K. Lee, S. Quayle, S. Beaird, C. Braud, J. Bonneson, P. Tarnoff, and T. Urbanik. 2008. "Traffic signal timing manual". Technical report.

Lowrie, P. 1982. "The Sydney coordinated adaptive traffic system-principles, methodology, algorithms". In *Proceedings of the 1982 International Conference on Road Traffic Signalling*, 67–70.

Mikami, S., and Y. Kakazu. 1994. "Genetic reinforcement learning for cooperative traffic signal control". In *Proceedings of the 1st IEEE Conference on Evolutionary Computation*, 223–228. IEEE.

Mirchandani, P., and L. Head. 2001. "A real-time traffic signal control system: architecture, algorithms, and analysis". *Transportation Research Part C: Emerging Technologies* 9 (6): 415–432.

MMDA 2014. "MMDA Traffic Mirror". http://mmda.nowplanet.tv/.

Morimura, T., T. Osogami, and T. Idé. 2013. "Solving inverse problem of Markov chain with partial observations". In *Advances in Neural Information Processing Systems*, edited by C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Volume 26, 1655–1663: Cambridge, MA: MIT Press.

Open Geospatial Consortium 2014. "KML". http://www.opengeospatial.org/standards/kml/.

OpenStreetMap 2014. "OpenStreetMap". http://www.openstreetmap.org/.

Osogami, T., T. Imamichi, H. Mizuta, T. Morimura, R. Raymond, T. Suzumura, R. Takahashi, and T. Ide. 2012. "IBM Mega Traffic Simulator". Technical report, IBM Research - Tokyo.

Robertson, D. I., and R. D. Bretherton. 1991. "Optimizing networks of traffic signals in real time: The SCOOT method". *IEEE Transactions on Vehicular Technology* 40 (1): 11–15.

Santini, S. 2000. "Analysis of traffic flow in urban areas using web cameras". In *Proceedings of the 2000 IEEE Workshop on Applications of Computer Vision*, 140–145.

Spall, J. C., and D. C. Chin. 1997. "Traffic-responsive signal timing for system-wide traffic control". *Transportation Research Part C: Emerging Technologies* 5 (3-4): 153–163.

Srinivasan, D., M. C. Choy, and R. L. Cheu. 2006. "Neural networks for real-time traffic signal control". *IEEE Transactions on Intelligent Transportation Systems* 7 (3): 261–272.

Suzumura, T., and H. Kanezashi. 2013. "A holistic architecture for super real-time multiagent simulation platforms". In *Proceedings of the 2013 Winter Simulation Conference*, edited by R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, 1604–1612. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Yin, Y. 2008. "Robust optimal traffic signal timing". *Transportation Research Part B: Methodological* 42 (10): 911–924.

Zhang, Y., M. Roughan, C. Lund, and D. Donoho. 2003. "An Information-Theoretic Approach to Traffic Matrix Estimation". In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 301–312: ACM New York.

Ziebart, B. D., A. L. Maas, J. A. Bagnell, and A. K. Dey. 2008. "Maximum Entropy Inverse Reinforcement Learning". In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence*, 1433–1438.

AUTHOR BIOGRAPHIES

KUMIKO MAEDA is a researcher at IBM Research – Tokyo. She belongs to the Smarter Cities Solution group. She received M.E. and Ph.D. degrees from the Institute of Computer Science and Technology, Osaka University in 2006 and 2009. Her research interests include vehicle traffic simulations, scheduling for heterogeneous clusters and mobility models in wireless network simulations. She is a member of the Information Processing Society of Japan. Her email address is kumaeda@jp.ibm.com.

TETSURO MORIMURA is a researcher at IBM Research – Tokyo. He received M.E. and Ph.D. degrees in engineering from the Nara Institute of Science and Technology, Japan, in 2005 and 2008. His research interests include reinforcement learning, machine learning, and their applications to behavior analytics, business intelligence, and industries. His email address is tetsuro@jp.ibm.com.

TAKAYUKI KATSUKI is a researcher at IBM Research – Tokyo. He received an M.E. degree in electrical engineering and bioscience from Waseda University, Tokyo, Japan, in 2012. His research interests are machine learning, data mining, and their applications to image processing. His email address is kats@jp.ibm.com.

MASAYOSHI TERAGUCHI is a researcher at IBM Research – Tokyo. He received an M.E. degree in computer science and technology from Osaka University, Japan, in 2000. His research interests are social analytics, crowdsourcing, and Web application security. His email address is teraguti@jp.ibm.com.