MobiTrick – Mobile Traffic Checker

Umair Ali Khan[*1], Martin Godec[2], Markus Quaritsch[1],
Marcus Hennecke[3], Horst Bischof[2] and Bernhard Rinner[1]

1. Institute of Networked and Embedded Systems, Klagenfurt University, Austria
2. Institute for Computer Vision and Graphics, Graz University of Technology, Austria
3. EFKON AG, Graz, Austria

Abstract

MobiTrick is a portable and compact traffic monitoring system that utilizes image processing capabilities to perform typical traffic monitoring tasks. It is based on a heterogeneous-sensors architecture using infrared and visible-light cameras. This setup allows utilizing the advantages of both sensors and additionally enables heterogeneous stereo reconstruction for 3D monitoring of vehicles. Due to the mobility factor, MobiTrick sensing platform is battery operated and hence imposes a strict limitation on power consumption. Therefore, it needs an efficient power management technique that optimizes the overall power consumption of the system. This paper presents MobiTrick's design architecture, novel vision-based techniques to perform the monitoring tasks, and the current work on an online Dynamic Power Management (DPM) strategy to minimize the sensing platform's power consumption.

Keywords: MOBITRICK, TRAFFIC MONITORING, STEREO-VISION, DYNAMIC POWER MANAGEMENT.

Introduction

Most of the current traffic monitoring systems are large, expensive and are based on fixed installations and hence difficult to deploy and maintain. During the deployment or maintenance, not only the road needs to be closed, but a re-calibration of the sensors is also required. In contrast, MobiTrick is designed as a compact, autonomous and portable system and therefore supports a flexible deployment for various monitoring tasks, e.g., law enforcement and construction site monitoring. The system is based on an application-specific, mobile infrastructure and is intended for short-term installations (hours or days). In addition, it uses a few, small form-factor sensors to perform all the required tasks with stereo-vision, hence eliminating the need of expensive sensors (e.g., laser, radar). Apart from this, the sensing platform also has auto-calibration capabilities, so it can adapt the calibration parameters just by the information readily available in the scene. Although it has a low-power design [1], in order to increase the system's lifetime, a power reduction technique is also

[*]Corresponding Author: Umair Ali Khan, MSc., umair.khan@aau.at, +43 / (0)463 / 2700-3872

required that minimizes the system's energy consumption during battery-powered operation. The following sections of the paper include a brief description of MobiTrick's sensing platform, the stereo-vision based building-blocks for capturing, monitoring, and analysis of traffic statistics, the current work on MobiTrick's DPM using a machine learning approach; the Reinforcement Learning (RL), and the conclusion with future work.

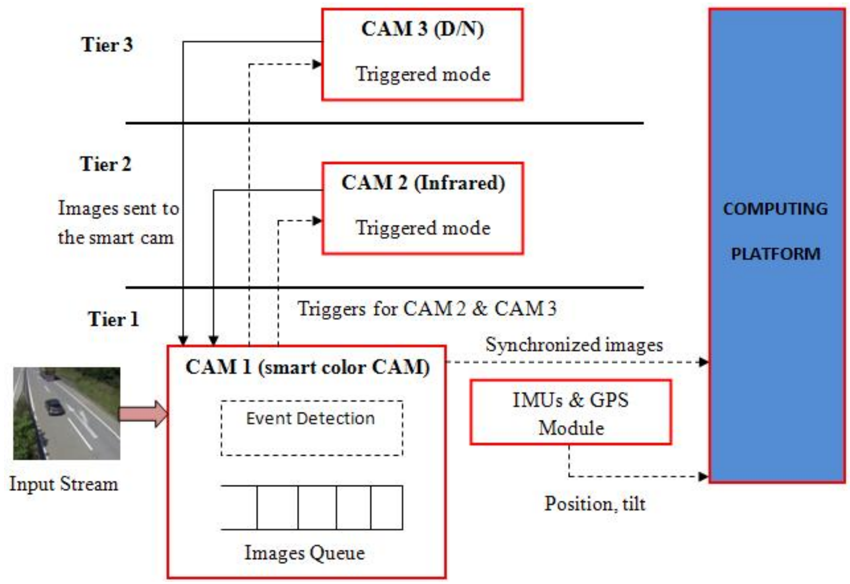

Figure 1: MobiTrick sensing platform

MobiTrick Sensing Platform

The MobiTrick sensing platform is configured with a multi-tier, heterogeneous setup where it uses different types of visual and non-visual sensors. A multi-tier, heterogeneous sensing environment serves many purposes. Not only it distributes tasks among different sensors, but several low-level operations can also be performed with less capable (and more power efficient) sensors at lower tiers where they can trigger the sensors at higher levels at the detection of some events. In addition, the redundant information from multiple sensors can be exploited to increase reliability. Lastly, it helps avoiding the use of additional sensors, such as laser or radar. A low-power Intel Atom based computing board is used in the sensing platform to perform required image processing operations. Figure 1 depicts a high-level overview of MobiTrick sensing platform.

Visual Object Detection & Classification

Traditional surveillance systems use a single camera to observe objects, such as cars or pedestrians. However, this simple setup has some drawbacks: (a) using a single camera restricts to a single light spectrum. If different lighting conditions occur, infra-red cameras may help while colour-cameras usually give better detection results; (b) cast-shadows, which are present in traffic surveillance very often, bring major difficulties to single-camera solutions. Therefore we propose to use a heterogeneous stereo setup using an infrared camera beside a colour camera. While combining the advantages of the different cameras for difficult lighting conditions, this setup enables full 3D reconstruction of images. Thereby we get rid of cast-shadows and can perform measurements on our objects under surveillance. In addition to the different light spectra the cameras also differ in their focal length, which means that one camera gives an overview of the scene while the second delivers an enlarged view of the object.

Scene adaption

Especially for mobile out-door surveillance tasks, adaption to the deployment site and changing conditions (lighting, weather) are very important. We address this issue using several different techniques according the requirements of the different tasks: (a) while the intrinsic parameters of our cameras will not change, the extrinsic parameters (distance and rotation between cameras) may change due to changes in temperature or vibrations. Therefore, we perform recalibration of extrinsic parameters during runtime. We model the parameter set using a Kalman Filter and measure the quality of the current configuration using overall matching costs for stereo reconstruction; (b) we use an adaptive robust block-based background-model [8] do determine regions of interest within each camera. Instead of modelling complex statistics for each pixel, simple intensity averaging within small overlapping blocks is performed. Due to the larger extent of the modelled regions, this background model can cope with waving leaves and camera shake which are often present in outdoor scenarios. To adapt the background-model to changing lighting conditions (moving cast-shadows during day, clouds) we continuously update the statistics. Due to the rectangular shape of the used blocks, the background model can be computed very efficiently using integral images. We use our background model for both visual triggering of the whole camera system and detection of the region of interest for the reconstruction; and (c) we perform online bootstrapping to increase the performance of our classifier, an Online Random Forest (ORF) [9]. Similar to common Random Forests, this classifier uses tree-like structures and is robust to a large amount of noise within training data. The classifier can be trained using streaming data (e.g. training data that arrives sequentially), but the classifier is available for evaluation at any time, even if only a small amount of training data has been processed.

If a certain amount of samples has reached a leaf node of the tree, it chooses a proper test that best separates the processed training data. The node changes its operation mode from “leaf” to “split” and initializes left and right leaf nodes beneath it. This procedure is repeated recursively until a maximum depth has been reached or the leaf node is monolithic (i.e. does only contain samples from a single class).

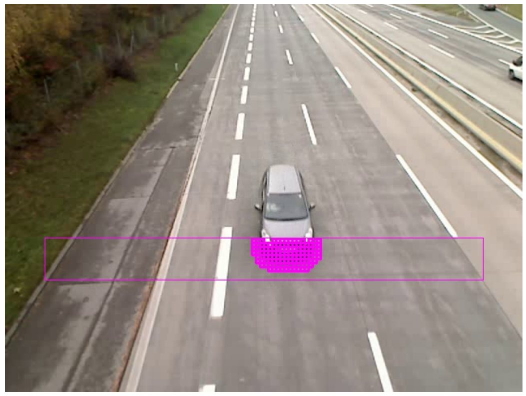

Figure 2: Block-based Background Model used as visual "induction loop"

3D Reconstruction

The proposed stereo setup allows reconstructing the object in 3D. However, heterogeneous stereo matching is not a standard task in computer vision. Therefore we evaluated a vast amount of different feature descriptors and matching techniques to optimize the reconstruction result of our setup [10]. Finally, respecting our limited hardware resources, we use a simplified version of Histograms of Oriented Gradients (HOG) features, which performed best for the heterogeneous data present in our system. Disparity optimization is performed using Semi-Global Matching [11], a recent approach that delivers state-of-the-art results. Figure 3 shows a sample output of the disparity calculation performed on our platform. Beside the full 3D reconstruction, which would exhaust the computing power of our embedded system and increase the power consumption, we are able to only use the "region of interest" depicted by our background model. As an alternative, we can compute a rough height estimation of the passing vehicle by calibrating several "virtual ground-planes" determining different height levels above the ground. However, this requires exactly the same mounting as used for calibration or a more time-consuming calibration at the deployment site has to be done.

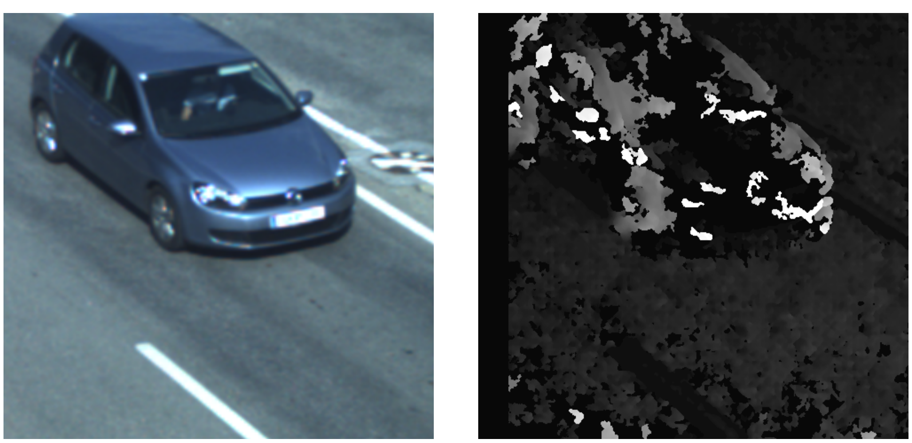

Figure 3: Visual input (visible spectrum camera) and corresponding disparity map. While the disparity map is by far not perfect, it clearly shows a height difference of the vehicle to the common street level.

Reinforcement Learning based DPM for MobiTrick

Among the existing DPM approaches, greedy policy [2], time-out policy [3] and predictive policies [4] perform well only when the system's workload has long intervals between the successive requests or when the requests are correlated. In road traffic situation, this is not always true. On the other hand, stochastic policies [5] require a priori model of the system components and have limited adaptability.

The model-free RL-based DPM approaches [6] [7] have received increasing attention recently. Although, the existing work on these approaches is focused on small devices, its application to more complex systems is also promising. RL allows a learning agent to automatically determine the ideal behaviour within a specific context, in order to maximize its performance. A simple pay-off (or reward) feedback is required for the agent to learn its behaviour. The overall goal of the learning agent is to minimize (or maximize respectively) the long-term cost or reward that it receives from the environment as a consequence of performing some actions. In a DPM problem, the pay-off/cost can be the immediate power consumption and a performance/latency penality resulted after taking an action. While observing the successive cost values, the agent learns to take optimal actions that can minimize the cost function. Since the cost function includes both the immediate power consumption and the performance latency, the agent finds a policy that achieves the best possible trade-off between the power consumption and the performance. We use a type of RL technique, the Q-learning, that works efficiently for such a problem. Q-learning works by learning an action-value function that gives the expected cost of taking a given action in a given state and following a fixed policy thereafter. One of the strengths of Q-learning is that it is able to compare the expected cost of the available actions without requiring a model of the environment.

For MobiTrick DPM, we targeted the computing board which is the major source of power consumption in the sensing platform and works as a Service Provider (SP) in the RL environment. The power model of the SP is shown in Table-1.

P_{sleep}	P_{idle}	P_{busy}	P_{trans}	T_{trans}
4 W	25 W	32 W	15 W	4 Sec

Table 1 – Power model of SP (computing board)

The other components of the RL environment, namely the Service Requestor (SR) that generates requests and the Service Queue (SQ) where the requests are buffered before being processed, reside on the smart camera. The SQ holds the synchronized images from all cameras present in the system. In this way, the smart camera serves as a Power Manager (PM) and issues commands to other components of the system. These commands include the available actions to change the power state of the SP. Based on an action taken, the RL environment assigns a scalar cost to the PM which is a weighted sum of the immediate power consumption and a performance/latency penalty caused by that action. The relative weighting between the power consumption and the performance factor works as a trade-off parameter. The overall aim of the RL algorithm is to minimize the cost function and hence to reach an optimal DPM policy. The relative weight between power and performance can be changed to obtain a Pareto-optimal trade-off curve.

We have tested and compared two models of the system: (i) *Model-1*: a deterministic model with known workload where the images are captured at a constant rate, (ii) *Model-2*: a stochastic model with workload estimation using a multi-layer Artificial Neural Network (ANN) with back-propagation algorithm. In Model-1, the images are captured at a constant rate and buffered in the service queue at the smart camera. Since the request rate is constant, the PM takes decisions only on the current state of the queue and the power mode of the SP. In Model-2, the smart camera runs a vehicle detection algorithm and the other cameras are triggered only at the detection of a vehicle.

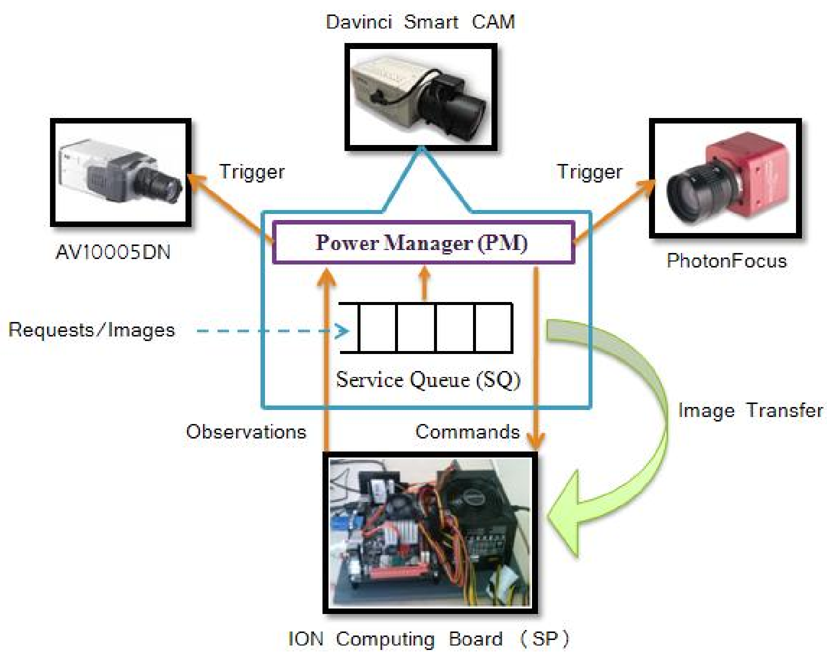

Figure 4 – Depiction of the system under power management

The traffic data used in Model-2 comprises a 24-hours recording of highway traffic where we measured vehicles arrival times with a vehicle detection algorithm. We use a fix-sized window on the history of previous inter-arrival times and input them to the ANN. The ANN estimates the length of the next inter-arrival period and this estimation is incorporated to the RL algorithm in the form of SR state. In sleep state, based on the estimated state of the workload, the PM decides after how many requests buffered in the queue, it would be appropriate to wake-up the board and process the requests. At each selected action, it gets an immediate pay-off in terms of a scalar cost. In the same way, when the SP is in idle mode, the PM executes a selected time-out value as an action, based on the estimated state of the workload and calculates its immediate cost in terms of the immediate power consumption. Figure 4 depicts the system under power management.

Although the two systems have different arrival rates of the requests, the number of requests processed is same and the comparison is made on the basis of both power consumption and system response. In Model-1, the request rate is constant and relatively higher. Therefore, the SP has to wakeup earlier at the accumulation of certain number of requests in the SQ. This does increases the power consumption due to more frequent transitions, but the average latency per request does not exceed a certain level. Whereas in Model-2, the arrival rate varies over time and during the low arrival rate, the SP spends more time in sleep state until certain number of requests is buffered in the SQ. Intuitively, this should decrease the power consumption, but the average latency per request should also increase. However, the workload prediction works fine here. It avoids the SP to spend unnecessary time both in idle state and the sleep state. Not only it decreases the overall power consumption, but also keeps the average latency per request at the same level as in Model-1. Apart from this, it gives a much wider curve of power vs. latency. At the sacrifice of some higher latency, more power savings can be achieved.

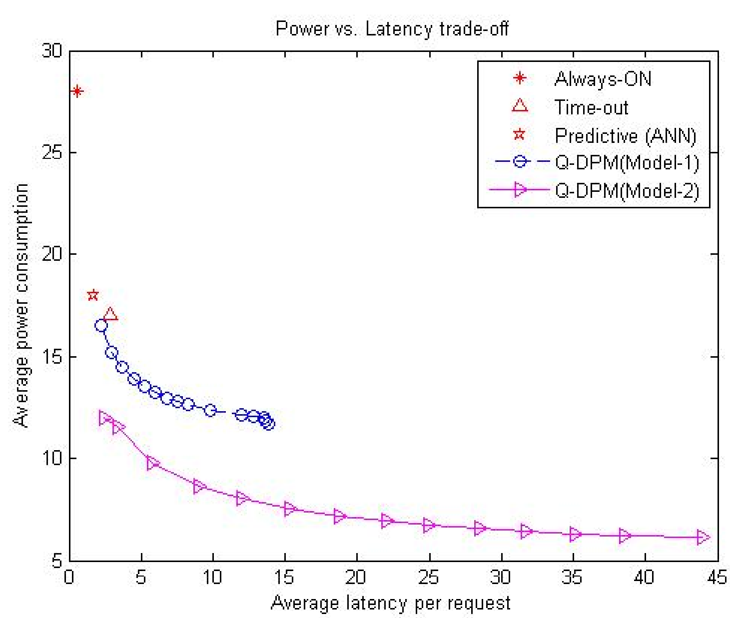

Figure 5 – A comparison of RL-based Model-1, Model-2 and other policies

Figure 5 shows a comparison of power vs. latency trade-off curves of Model-1 and Model-2.

The two models were also compared with some existing policies as shown in Figure 5. RL-based models achieve higher power savings with an acceptable level of system response.

Use-case scenarios

After explanation of the technical details of the proposed platform, we shortly review two possible use-case scenarios and the advantages of our platform in comparison to single-camera solutions.

Vehicle classification scenario

Visual vehicle detection using standard computer vision methods (e.g. [12]) delivers high accuracy predictions. However, for 24/7 operation the detection/classification accuracy has to be nearly perfect due to the high number of vehicles that pass by. Therefore, our system allows using 3D information as additional cue to verify detection hypothesis (i.e. changed blobs detected by the background model). While eliminating hypothesis where no vehicle is present (see Figure 6), the 3D cue can also give additional information used for classification of the passing vehicle, such as rough dimensions (height, width, and length).

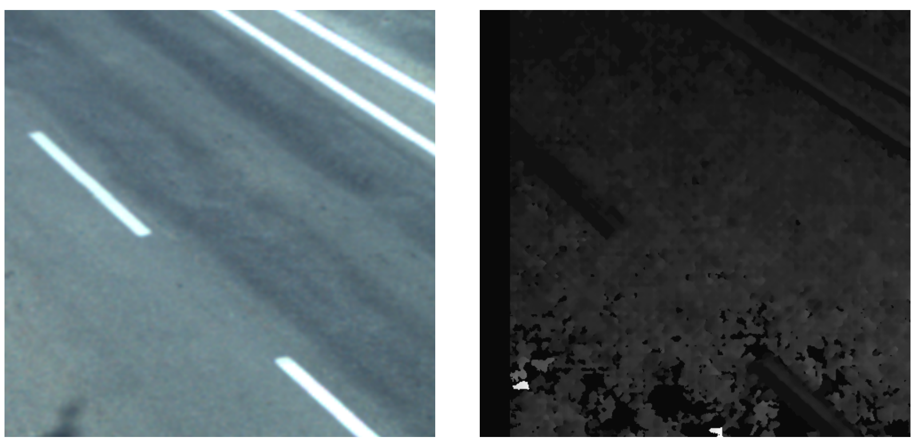

Figure 6: Visual input (visible spectrum camera) and corresponding disparity map. 3D reconstruction of the road surface eliminates false-positive detection. Objects that are present would result in a height elevation above the ground-plane.

Fused sensing

3D reconstruction of vehicles is a useful technology to increase the robustness and accuracy of a monitoring system. However, homogeneous camera setups (e.g. RGB and RGB) will not work for all different external conditions (e.g. night/day, summer/winter, sunshine/rain...) that will occur in traffic surveillance. However, there exist cameras that are suited for such specific conditions. Our heterogeneous setup does not only deliver 3D reconstructions if both cameras are working, but also combines the condition-specific advantages of both cameras. Both cameras can also be used individually, which enables a wider range of applications than a single camera system would allow for.

Conclusion

The paper presents a mobile traffic monitoring system, a brief description of vision-based building-blocks and work on online power management of the sensing platform. Being a full-fledged image processing based traffic monitoring system, our sensing platform is small, compact and flexible and can easily be deployed to any environment without any change in the road infrastructure. In addition, deployment and maintenance requires minimum effort and expertise.

We described all vision-based building blocks necessary to perform different applications, such as vehicle classification or over-width detection. Since our platform features two cameras, it enables 3D reconstruction of the visual input which increases robustness in comparison to 2D processing (e.g. cast shadow removal) and also enables better classification due to additional processing cues.

From the power management perspective, we compared two different models and showed that our approach is applicable for both constant-rate service requests (Model-1) and event-based service requests (Model-2) which are both very relevant for various traffic applications. In comparison with the traditional DPM approaches (time-out, predictive), our RL-based DPM models give much wider power vs. latency trade-off curves. Our future work on power management of the sensing platform includes improving the RL-based algorithm by incorporating time-out values at sleep state also which may result in better system response. Moreover, we are aiming to extend our algorithm to target an embedded computing platform having higher number of sleep and idle states.

Acknowledgement

This work has been supported by the Austrian FFG project MobiTrick (8258408) under the FIT-IT program.

References

1. U. A. Khan, M. Quaritsch, B. Rinner (2011). *Design of a Heterogeneous, Energy-Aware, Stereo-Vision Based Sensing Platform for Traffic Surveillance*, in proceedings of the Ninth Workshop on Intelligent Solutions in Embedded Systems.

2. J. M. Pedram (2002). *Power Aware Design Methodologies*, Kluwer Academic, Norwell, USA.

3. Benini et al (1999). *Policy Optimization for Dynamic Power Management*, IEEE Transactions on Computer Aided Design, vol. 18, pp. 813-833.

4. Srivastava et al (1996). *Predictive System Shutdown and Other Architecture Techniques for Energy Efficient Programmable Computation*, IEEE Transactions on VLSI Systems, vol. 4, pp. 42-55.

5. Y. -H. Lu, G. D. Micheli (2001). *Comparing System Level Power Management Policies*, IEEE Transactions on Design Test of Computers, vol. 18, 10-19.

6. Y. Wang et al (2011). *Deriving a Near-Optimal Power Management Polilcy Using Model-Free Reinforcement Learning and Bayesian Classification*, in 48th ACM Design Automation Conference.

7. G. Dhiman, T. Rosing (2006). *Dynamic Power Management Using Machine Learning*, in IEEE International Conference on Computer Aided Design.

8. P. M. Roth (2008). *On-line Conservative Learning*. PhD thesis, Graz University of Technology, Faculty of Computer Science.

9. Amir Saffari, Christian Leistner, Jakob Santner, Martin Godec, and Horst Bischof (2009). *On-line Random Forests*, in Proc. IEEE On-line Learning for Computer Vision Workshop.

10. Peter Pinggera (2011). *Recovery of Depth Information Using Paired Optical And Thermal Images*, MSc. Thesis, Graz University of Technology, Faculty of Computer Science.

11. Heiko Hirschmüller (2008), *Stereo Processing by Semi-Global Matching and Mutual Information*, IEEE Transactions on Pattern Analysis and Machine Intelligence.

12. Paul A. Viola, Michael J. Jones (2001). *Rapid Object Detection using a Boosted Cascade of Simple Features*. In Proc. IEEE Conference on Computer Vision and Pattern Recognition.