Volume 09 Issue 02 Published, May 19, 2005 ISSN 1535-864X DOI: 10.1535/itj.0902

Intel® Technology Journal

Compute-Intensive, Highly Parallel Applications and Uses

Computer Vision Workload Analysis: Case Study of Video Surveillance Systems

Computer Vision Workload Analysis: Case Study of Video Surveillance Systems

Trista P. Chen, Corporate Technology Group, Intel Corporation
Horst Haussecker, Corporate Technology Group, Intel Corporation
Alexander Bovyrin, Corporate Technology Group, Intel Corporation
Roman Belenov, Corporate Technology Group, Intel Corporation
Konstantin Rodyushkin, Corporate Technology Group, Intel Corporation
Alexander Kuranov, Corporate Technology Group, Intel Corporation
Victor Eruhimov, Corporate Technology Group, Intel Corporation

Index words: computer vision, workload analysis, RMS, video surveillance, foreground detection, estimation-maximization, Gaussian mixture, particle filter, condensation filter, Markov chain Monte Carlo, eigen analysis, singular value decomposition, optical flow, motion estimation

ABSTRACT

The vast accumulation of digital data requires new classes of applications that impact a computer user's life. We are investigating computing platforms that can deliver enough performance for these future workloads to enable their use in mass-market applications. Recognition, Mining, and Synthesis (RMS) are three key classes of workloads that distill enormous amounts of data. Among these is Computer Vision (CV), an important workload that will greatly benefit from future architecture and algorithm innovations.

We illustrate these innovations by introducing and characterizing some of the most common CV algorithms and applications. We focus on (1) algorithms for Gaussian mixture models, (2) particle filtering (condensation filtering), and (3) optical flow/motion estimation, which are key ingredients of many modern CV algorithms. We also discuss computer vision applications, such as video surveillance, autonomous (intelligent) vehicles and driver assistance systems, entertainment and augmented reality, and smart health care.

We chose a complete video surveillance application as a representative case study for a complex CV workload. Video surveillance is one of the most resource-demanding CV applications that has wide-spread application. We analyze an entire pipeline of a video surveillance system to obtain computation and bandwidth characteristics.

Our characterization of individual CV algorithms as well as complete CV systems can be used to guide algorithm researchers to develop new algorithms that run faster on existing and future computing platforms. Furthermore, we hope that it will raise the awareness of the application developers to optimize their programs. It will also provide input data for architects to develop future computing platforms that run these workloads more efficiently.

INTRODUCTION

The amount of data in the world is doubling every three years. This includes data found on the Web, in our personal albums, and digital music collections, etc. However, these data are usually not organized efficiently and not used to their full extent. For example, we might spend hours opening images we have on our hard-drives (assuming they are still stored in the hard-drive without losses) to look for one image of ourselves with a particular person. Looking forward, three fundamental processing capabilities: Recognition, Mining, and Synthesis (RMS), will be the key to future data processing, and Computer Vision (CV) is one of its main workloads.

This paper is organized as follows. In "Trends in Computer Vision Workloads," we discuss the key algorithms and applications of CV workloads, as well as current trends. In "Introduction to the Video Surveillance System," we study in more detail one of the most representative CV workloads: video surveillance. We first describe the complete pipeline of a video surveillance system and its individual components, and then we present results from workload analysis of the video surveillance system. We conclude with a summary of our findings and future research opportunities.

TRENDS IN COMPUTER VISION WORKLOADS

Key Algorithms of Computer Vision Workloads

Computer Vision (CV) algorithms can be categorized as low-level image-processing techniques and high-level analysis techniques. Image-processing techniques include filtering, and feature extraction, which have been extensively studied and are thus not elaborated on in this paper. High-level analysis usually involves probability-based approaches. We discuss the Gaussian mixture model, and particle filters, since these are used extensively in tracking and motion analysis.

We believe the trend in CV algorithms is moving towards higher level analyses (such as semantics), which make extensive use of probability-based techniques. Probability-based algorithms are our main focus. We start with discussing probability distribution models. We then discuss multi-model data fusion from color images, the integration of color video and range data, and the integration of face images, fingerprint images, and iris images.

Gaussian Mixture Model

In order to track an object in the scene, knowledge about what the object looks like is needed. Such knowledge is described by the statistical distribution of the region of interest (which could be the foreground object or the background). The Gaussian mixture model is used widely to describe the region of interest [1]-[6], and has been incorporated into a variety of algorithms for tracking and recognition in CV.

A Gaussian mixture model with m components is described as a sum over Gaussian probability distributions:

$$p(x|Obj) = \sum_{m=1}^{M} \pi_m N\left(x, \mu_m, \sigma_m^2 I\right)$$

where u_m is the mean of component m, σ_m is the variance of component m, and π_m is the weight of component m. By applying the Estimation-Maximization (EM) algorithm, the Gaussian mixture model parameters can be trained.

Particle Filter/Condensation Filter

The particle filtering algorithm is a sequential Monte Carlo method. The algorithm is powerful in approximating non-Gaussian probability distributions, and it has a wide range of applications including object tracking in CV [7][8][9]. Particle filtering is based on sequential importance sampling and Bayesian theory. It models the data distribution by random sample measures composed of particles, that are samples from the space of the unknowns, and their associated weights.

There are two main steps in the algorithm: selection/updating and mutation/prediction. The first step selects the particles for reproduction. The particles representing the most likely parameter candidates are the ones most likely to be selected. During this step, heavy particles generate new ones, while light particles are eliminated. The second step allows each particle to evolve according to a given transition probability kernel. The pictorial description of the algorithm is shown in Figure 1.

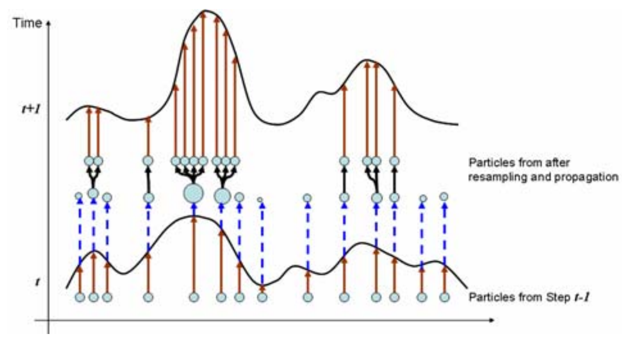

Figure 1: A pictorial description of particle filtering

Similarly, in using random samples to estimate distribution, Markov Chain Monte Carlo (MCMC) is also often used. References can be found in [10][11].

Optical Flow/Motion Estimation

Optical flow estimation is a technique used to compute the apparent image motion field of the scene. Besides tracking objects, the motion field is useful for extracting objects from video sequences. When there is discontinuity in the motion field, it often means there are different depths of pixels and the depth discontinuities may correspond to the outlines of different objects. An introduction to optical flow can be found in [12][13], and its application to road detection can be found in [14].

Key Computer Vision Applications and Opportunities

Digital video recording devices are now ubiquitous in our daily lives. They are mounted indoors in offices, hospitals, and outdoors in parking lots and intersections. Some vehicles even have cameras recording passengers and the surroundings of the car. Massive amounts of video data are collected by these "digital eyes." CV technologies can add intelligence to these digital eyes, adding "brains" to these imaging devices.

With both digital eyes and brains, CV can be a very useful tool: it can be used for video surveillance, entertainment/augmented reality applications, autonomous vehicles and driver assistance systems, robotics, and smart health care. With the rapid growth in CV, we believe more innovation in this field is inevitable. We now discuss these CV applications.

Video Surveillance/Security

Video surveillance addresses real-time observation of humans or vehicles in some environment (indoor, outdoor, or aerial), leading to a description about the activities of the objects with the environment or among the objects. It is used mostly for security monitoring, as well as traffic flow measuring, accident detection on highways, and routine maintenance in nuclear facilities, etc. Interested readers can refer to [17][18].

From a workload analysis perspective, video surveillance is one of the most interesting CV applications. The reason is twofold. First, a complete video surveillance system consists of foreground segmentation, object detection, object tracking, human or object analysis, and activity analysis. It touches many core topics of CV. By understanding and analyzing a complete video surveillance system, we can obtain insights on general CV workloads. Second, video surveillance is currently gaining increasing importance for security applications worldwide. "Traditional" video surveillance systems have been used pervasively in airports, banks, parking lots, military sites, etc. However, to get any useful information from these systems, humans either have to watch a massive amount of video data in real-time with full attention to detect any anomalies, or the video data can only be used as evidence after an abnormal event has occurred, due to the lack of real-time automatic tracking and analysis. Automatic video surveillance, as opposed to traditional video surveillance, adopts CV algorithms such as those mentioned in the last section to alleviate the load on humans and to enable preventative acts when an anomaly is detected. We will use the term video surveillance, instead of automatic video surveillance, hereafter. A comprehensive introduction to video surveillance systems can be found in the next section.

Autonomous Vehicles and Driver Assistance Systems

Driver safety is a very important issue in our lives. CV can serve as a third eye for a driver to enhance the safety of vehicles and of their occupants. Example uses of CV for intelligent vehicles include (1) parking assistance; (2) landmark detection to assist the car in following the road; (3) traffic sign detection and recognition for route planning and alerting the driver; (4) obstacle detection, especially detecting the presence of pedestrians in a driver's blind spot; (5) driver condition monitor for intelligent airbag deployment and to monitor a driver's distraction level [19][20][21].

Figure 2 (from [22]) shows a road detection example. The detection result can be used for the later obstacle detection in knowing the 3D location of the obstacle. Figure 3 from [23] shows an example of how CV can help in locating pedestrians/obstacles. Pedestrian/obstacle detection and tracking is harder than conventional object detection and tracking. Both the camera in the car and the objects are moving. The background changes constantly. Figure 4 (from [24]) shows different poses of a driver for safe airbag deployment as shown in Figure 5 (from [24] as well).

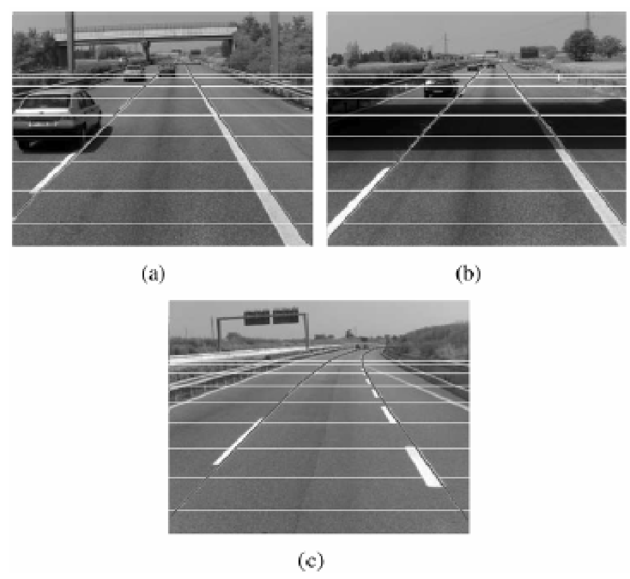

Figure 2: Road-detection result for highways and the 3-D projected model with different conditions. © 2005 IEEE, courtesy of [22]

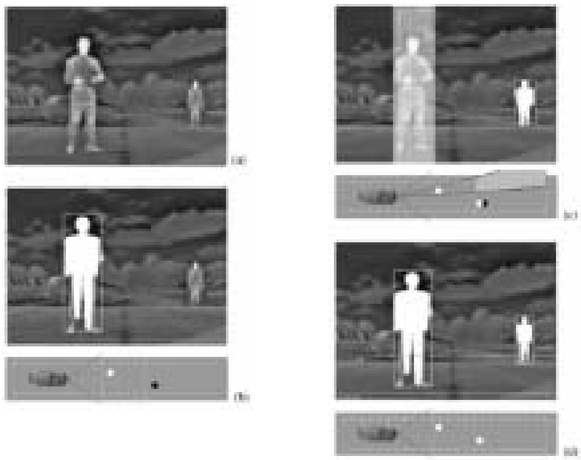

Figure 3: Obstacle detection example: fusion of the results from two resolutions. (a) Input image, (b) results of low-resolution processing, (c) results of original resolution processing, and (d) final fused results. © 2005 IEEE, courtesy of [23]

Figure 4: Example images of occupant script poses. From top left: sitting normally, leaning halfway, leaning completely forward, leaning back, leaning right, leaning left, moving hands about cabin, opening glove box, hands on face, stretching, adjusting radio, hat in lap, putting on hat, moving while wearing hat, removing hat, feet on dashboard. © 2005 IEEE, courtesy of [24]

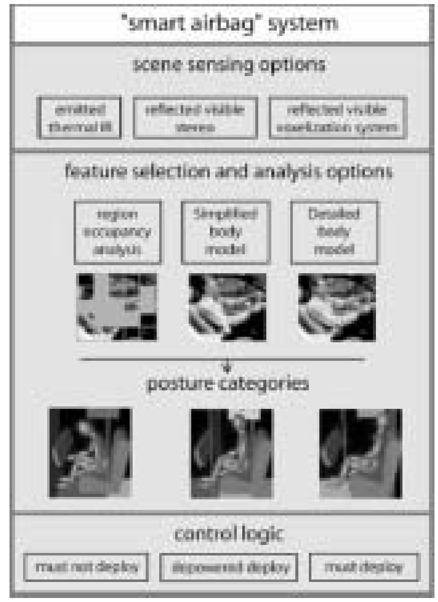

Figure 5: Occupant position and posture-based safe airbag deployment. © 2005 IEEE, courtesy of [24]

Entertainment/Augmented Reality

Another interesting application of CV techniques is augmented reality, which combines real video content, a real object extracted from the video, and rendered graphical models. That is, three-dimensional virtual objects are embedded in the real video scene. Augmented reality applications can be seen in entertainment scenarios such as games and movies. This technique is useful for 3D manipulation and maintenance tasks, and it is helpful during surgical procedures as clinical data can be overlaid on real video content.

CV techniques are needed to extract the 3D information from the environment, so that the virtual objects can be placed in the proper locations (an example is shown in Figure 6 from [25]). Articulated body tracking can be used to animate the character in the gaming environment (an example is shown in Figure 7 from [26]).

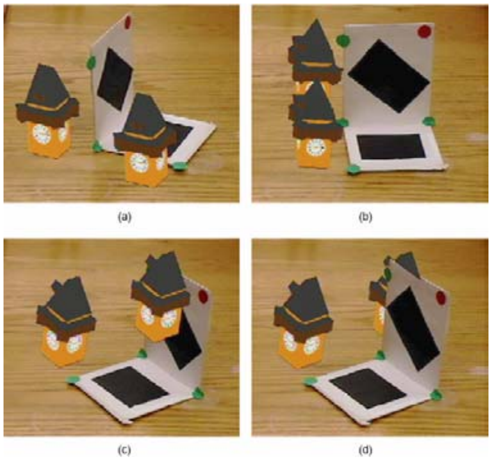

Figure 6: Visible-surface rendering of texture-mapped affine virtual objects. Affine basis points were defined by the centers of the four green dots. The virtual towers were defined with respect to those points: (a) initial augmented view; (b) augmented view after a clockwise rotation of the object containing the affine basis points; (c) hidden-surface elimination occurs only between virtual objects; correct occlusion resolution between physical and virtual objects requires information about the geometric relations between them; (d) real-time visible surface rendering with occlusion resolution between virtual and real objects. © 2005 IEEE, courtesy of [25]

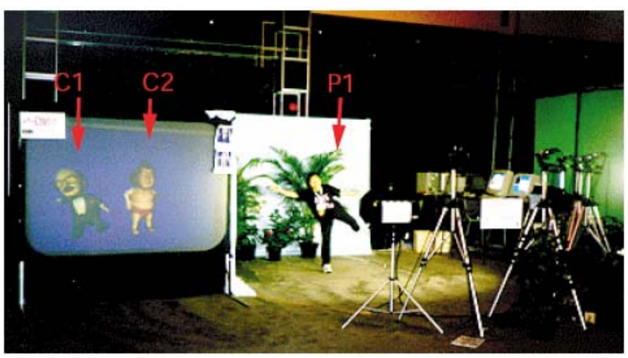

Figure 7: Virtual metamorphosis. The motion of the person (P1) in the dancing area controlled the movement of the tuxedo-wearing cartoon character (C1). A person in another dancing area controlled the movement of the sumo wrestler cartoon (C2). © 2005 IEEE, courtesy of [26]

Smart Health Care

CV can be used in health care for the elderly and the disabled. Human body tracking and activity analysis can help detect anomalies, such as a person falling. Human subjects can also use their hand or body gestures to control the home environment if hand or body tracking is in place [27]. Similar ideas can be applied to smart offices and smart homes.

We do not discuss CV for medical image-related applications in this paper. Although CV has been studied extensively, the techniques used are mostly lower-level image processing. Interested readers can refer to [28].

INTRODUCTION TO VIDEO SURVEILLANCE AND COMPUTER VISION

Traditional video surveillance is labor intensive and usually not very effective. Video surveillance with computer vision techniques, however, saves on labor and provides a consistent monitoring quality. The input to a video surveillance system is video streams from a single or multiple cameras. The system analyzes the video content by separating the foreground from the background, detecting and tracking the objects, and performing a high-level analysis. The high-level analysis provides results such as a scenario being normal or abnormal. The human operator can then focus on the abnormal scenarios and not have to stare at the video trying to find any anomaly.

A general scheme of the video surveillance system is shown in Figure 8, where

- A "Foreground/Background (FG/BG) Detection" module performs FG/BG classification of each image pixel.
- A "Blob Entering Detection" module uses the result (FG mask) of the "FG/BG Detection" module to detect that a new blob object enters the scene.
- A "Blob Tracking" module is initialized by the "Blob Entering Detection" module. This module tracks each blob from the tracked blob list.
- A "Trajectory Generation" module collects all blobs' positions and saves each blob trajectory to hard disk when the motion of the object is no longer presented (for example, when the tracking is lost).
- A "Trajectory Post Processing" module executes a smoothing function on a blob trajectory. This module is optional and need not be included in a specific pipeline.
- A "Trajectory Analysis" module performs a blob trajectory analysis and detection of abnormal trajectories.

We will discuss each module in later sessions.

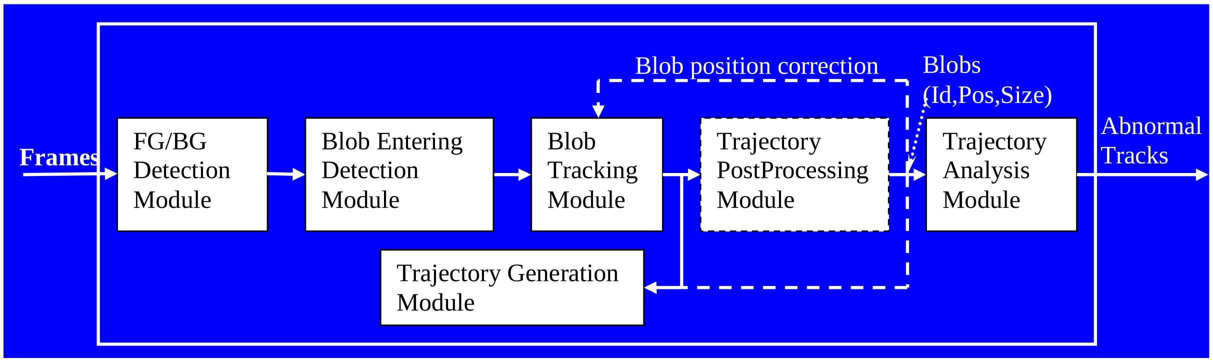

Figure 8: Video surveillance system pipeline

Foreground and Background Estimation

Given a video sequence, extracting the FG from the BG is an important step in the whole video surveillance pipeline. FG estimation is the first stage in the pipeline. Its accuracy affects the accuracy of the later stages. It could affect the performance of the later stages as well. Note that in this paper we use "accuracy" for algorithm accuracy, and "performance" for computation performance such as the speed of computation.

FG detection is generally easier in the indoor environment. Ideally, we want FG/BG estimation to work well in both indoor and outdoor environments. The outdoor environment is more complex, as wavering tree branches, flickering water surfaces, periodic opening and closing of doors, etc. are occurring. We use the approach proposed by Li et al. [29] in our video surveillance system, for its capability in processing complex backgrounds. This method is based on pixel color and color co-occurrence statistics. The pixel color and color co-occurrence distributions are represented by histograms. Bayes decision rule is applied to classify the pixel to BG or FG pixel. The BG is updated after the classification of pixels. The algorithm can successfully handle gradual as well as sudden BG changes, and stationary as well as moving objects.

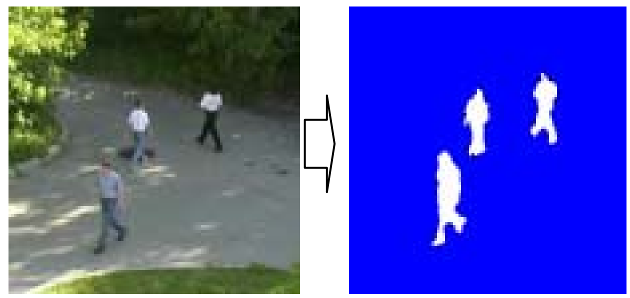

Figure 9: Foreground extraction in outdoor environment

Blob Entering Detection

Our blob detection module is based on a connected component tracker [30]. It does the following:

1. Calculates connected components of the FG mask obtained by the FG/BG estimation module. Each component is considered as a blob.
2. Tracks each blob by trying to find it in the current and the previous frame.
3. Adds a new blob into the tracked blob list if it can be tracked successfully across multiple successive frames.

With the tracked blob list, we can apply object detectors such as those proposed by [31][32] to determine the class of the blob objects: "human," "car," or "unknown."

Blob Tracking

The blob tracking module provides frame-by-frame tracking of the blob position and size. We developed a hybrid object tracker that consists of two components. The first one is a connected-component tracker. It provides reliable and fast tracking results when there is no overlap of two human blobs. The second component is a tracker that is based on mean-shift algorithms and particle filtering [33][34]. A Kalman filter is used to predict the position of the blob in the next frame, thus implying that overlap will occur in the next frame. If overlap is to occur, the second component, the particle filter-based tracker, is used. Otherwise, the fast connected-component-based tracker is used.

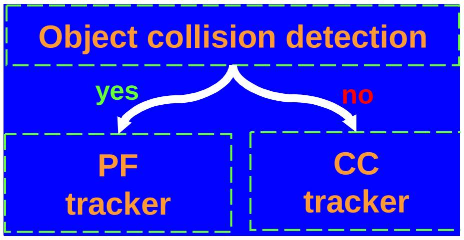

Figure 10: Switching of trackers

Activity Analysis

To detect the anomaly of a scene, we classify the blob trajectories. Trajectory analysis approaches can be found in [35]-[38]. To detect abnormal trajectories, a histogram approach is used. This method treats a trajectory as an independent set of feature vectors. Each feature vector includes such features as blob position, blob velocity, and blob state duration. A 5D histogram of these features is continuously collected and analyzed. Thus, if the current blob has features that were never or rarely observed before, then the blob and its trajectories are classified as abnormal.

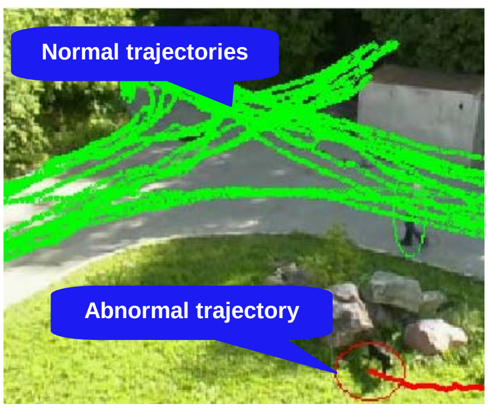

Figure 11: Abnormal trajectory

WORKLOAD ANALYSIS OF THE VIDEO SURVEILLANCE SYSTEM

In this section, we profile the video surveillance system by the Intel® VTune™ Performance Analyzer [39]. We identify hot spots for future performance improvement, either in algorithm modification or in hardware acceleration. Furthermore, we explore the distribution of the operations.

Profiling our system shows that the most computationally expensive modules of the whole video surveillance system are FG detection (up to 95% of execution time) and object tracking (about 5% when only the connected components method is used, and up to 20% when mean shift algorithms with particle filtering is employed). Histogram-based trajectory analysis doesn't take a significant portion of the computational resources; although more sophisticated techniques may potentially contribute more to this workload. Note that the numbers given are not universal to all kinds of scenarios including both indoor and outdoor environments. In a scene where there are more objects presented, object tracking may consume a larger proportion of computational resources, whereas the resources used by FG/BG estimation remain about the same.

The most computationally expensive module of the pipeline, FG/BG estimation, consumes 1 billion microinstructions per frame of size 720x576. On a 3.2 GHz Intel® Pentium® 4 processor, therefore, it takes 0.4 sec.

We further profile the FG/BG module as shown in Figure 12. The most expensive part of the algorithm is the histogram update, which scans all histogram bins. Classification uses only a subset of the histogram. Other parts of the algorithm work only with frame pixels, searching the place for the current pixel value.

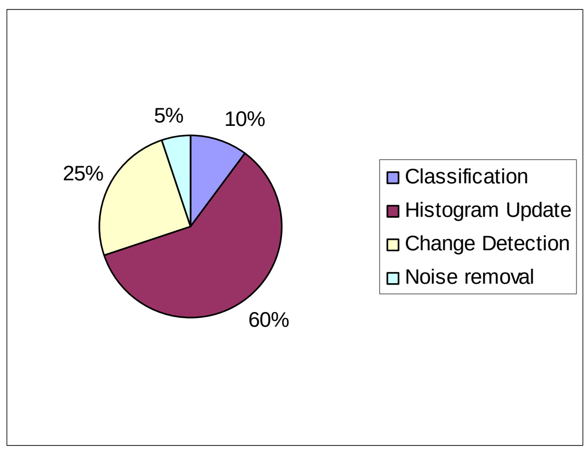

Figure 12: Foreground detection algorithm profile

® Intel and VTune are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

® Intel and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

The algorithm consumes a large amount of memory. Each pixel keeps a set of color and color co-occurrence histograms, which takes up about 1 kb per pixel. The whole frame, of size 720x576, needs about 400 MB of memory. However, according to the memory statistics (see Table 1), accesses to the histogram are mostly cached, so we don't have to read the same histogram values from the memory several times; we can get them from the cache.

Table 1: Memory characteristics of foreground detection algorithm

L1 cache hit rate	90%
L2 cache hit rate	77%
Bus data traffic per frame	134 MB
Bus utilization	6%

We now look at the operation distribution (as shown in Figure 13). Most of the arithmetic operations are from integer operations. Operations done on pixel values and histogram bins are mostly integer operations. We see similar situations in many CV workloads where pixel values, histograms, and array indices calculation, are often involved. Floating-point operations in FG/BG estimation are minimal, since the FG/BG algorithm accesses floating-point data only for the histogram bin that is hit by the pixel value. The branches portion is noticeable. However, the branch prediction is good (about 90% correct). Therefore, mis-predicted branches do not significantly impact performance.

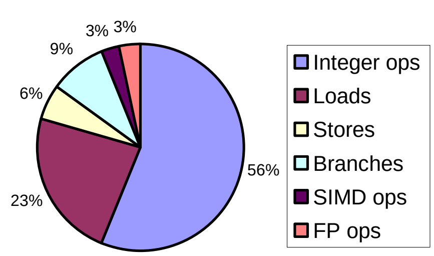

Figure 13: Operations distribution for foreground detection algorithm

SUMMARY

RMS will be the key to future data processing. CV workload is one important RMS workload. In this paper, we talked about trends in CV algorithms and applications. Understanding the trends in CV algorithms and identifying trends in CV applications help Intel in developing future computing platforms.

We then focused on video surveillance systems. A complete video surveillance pipeline captures important aspects of many CV workloads. Video surveillance is one of the most important and resource-demanding CV applications. We identified the hot spots and operation distributions of the system using the Intel VTune Performance Analyzer. Such performance analysis results will be useful for future Intel architecture innovations.

ACKNOWLEDGMENTS

The authors thank Christopher J. Hughes, Yen-Kuang Chen, and Sanjeev Kumar for providing insights into this paper and for editing the technical content. We also thank the paper referees for their valuable comments.

REFERENCES

[1] R. Hammond and R. Mohr, "Mixture densities for video objects recognition," 2000 *IEEE International Conference on Pattern Recognition*, pp. 71-75.

[2] R. Wilson, "MGMM: multiresolution Gaussian mixture models for computer vision," 2000 *IEEE International Conference on Pattern Recognition*, pp. 212-215.

[3] M. Harville, G. Gordon, and J. Woodfill, "Foreground segmentation using adaptive mixture models in color and depth," *2001 IEEE Workshop on Detection and Recognition of Events in Video*, pp. 3-11.

[4] Y. Zhu and K. Fujimura, "Driver face tracking using Gaussian mixture model (GMM)," *IEEE Intelligent Vehicles Symposium*, June 2003, pp. 587-592.

[5] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," ICPR 2004, pp. 28-31.

[6] C. Stauffer C, W.E.L. Grimson, "Adaptive background mixture models for real-time tracking," 1999 *IEEE CVPR*.

[7] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, 50(2), Feb. 2002, pp. 174-188.

[8] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, "Particle filter for positioning, navigation, and

tracking," *IEEE Trans. On Signal Processing*, 50(2), Feb. 2002, pp. 425-437.

[9] P.M. Djuric, J.H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M.F. Bugallo, and J. Miguez, "Particle filtering," *IEEE Signal Processing Magazine*, Sept. 2003, pp. 19-38.

[10] Z. Tu and S.-C. Zhu, "Image Segmentation by Data-Driven Markov Chain Monte," *IEEE Trans. On PAMI*, 24(5), May 2002, pp. 657-673.

[11] J.C. Spall, "Estimation via Markov Chain Monte Carol," *IEEE Control Systems Magazine*, April 2003, pp. 34-45.

[12] J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comput.Vis*., vol. 12, no. 1, 1994, pp. 42-77.

[13] A. Verri and T. Poggio, "Motion field and optical flow: qualitative properties," *IEEE Trans. On PAMI*, 11(5), May 1999, pp. 490-498.

[14] A. Giachetti, M. Campani, and V. Torre, "The use of optical flow for road navigation", *IEEE Trans. On Robotics and Automation*, 14(1), Feb. 1998, pp. 34-48.

[15] H. Sakoe and S. Chiba, "Dynamic programming optimization spoken word recognition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, Feb. 1978, pp. 43-49. (DTW).

[16] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," *AAAI-94 Workshop on Knowledge Discovery in Databases (KDD-94)*, Seattle, WA, 1994. (DTW).

[17] *IEEE Trans. On PAMI* 22(8), August 2000, Special Section on Video Surveillance, R.T. Collins, A.J. Lipton, and T. Kanade, guest editors.

[18] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, A. Senior, C.-F. Shu, and Y.L. Tian, "Smart Video Surveillance: exploring the concept of multiscale spatiotemporal tracking," *IEEE Signal Processing Magazine*, March 2005, pp. 38-51.

[19] J. Manigel and W. Leonard, "Vehicle control by computer vision," *IEEE Trans. On Industrial Electronics*, 39(3), June 1992, pp. 181-188.

[20] *IEEE Trans. On Vehicular Technology*, 53(6), Nov. 2004, Special Section on In-Vehicle Computer Vision Systems, R. Cucchiara, D. Lovell, A. Prati, and M.M. Trivedi, guest editors.

[21] GM Collaborative Research Lab at Carnegie Mellon University, http://gm.web.cmu.edu/*

[22] C. Demonceaux, A. Potelle, and D. Kachi-Akkouche, "Obstacle detection in a road scene based on motion analysis," *IEEE Trans. On Vehicular Technology*, 53(6), Nov. 2004, pp. 1649-1656.

[23] M. Bertozzi, A. Broggi, A. Fascioli, T. Graf, and M.-M. Meinecke, "Pedestrian detection for driver assistance using multiresolution infrared vision," *IEEE Trans. On Vehicular Technology*, 53(6), Nov. 2004, pp. 1666-1678.

[24] Photo courtesy: Professor Mohan Trivedi, Laboratory for Intelligent and Safe Automobiles, UCSD, cvrr.ucsd.edu/LISA Ref: M.M. Trivedi, S.Y. Cheng, E.M.C. Childers, and S.J. Krotosky, "Occupant posture analysis with stereo and thermal infrared video: algorithms and experimental evaluation," *IEEE Trans. On Vehicular Technology*, 53(6), Nov. 2004, pp. 1698-1712.

[25] K.N. Kutulakos and J.R. Vallino, "Calibration-free augmented reality," *IEEE Trans. On Visualization and Computer Graphics*, 4(1), Jan.-March 1998, pp. 1-20.

[26] J. Ohya, J. Kurumisawa, R. Nakatsu, K. Ebihara, S. Iwasawa, D. Iwasawa, and T. Horprasert, "Virtual metamorphosis," *IEEE Multimedia*, 6(2), April-June 1999, pp. 29-39.

[27] A. Mihailidia, B. Carmichael, and J. Boger, "The use of computer vision in an intelligent environment to support aging-in-place, safety, and independence in the home," *IEEE Trans. On Information Technology in Biomedicine*, 8(3), Sept. 2004, pp. 238-247.

[28] J.S. Duncan and N. Ayache, "Medical image analysis: progress over two decades and the challenges ahead," *IEEE Trans. On PAMI*, 22(1), Jan. 2000, pp.85-106.

[29] L. Li, W. Huang, I.Y.H. Gu, Q. Tian, "Foreground object detection from videos containing complex background," *ACM Multimedia**, 2003.

[30] A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, R. Bolle, "Appearance Models for Occlusion Handling," in *proceedings of Second International workshop on Performance Evaluation of Tracking and Surveillance systems in conjunction with CVPR'01*, December 2001.

[31] P. Viola, M.J. Jones, and D. Snow, "Detecting Pedestrians Using Patterns of Motion and Appearance," *IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 734-741, 2003.

[32] H. Schneiderman and T. Kanade, "A statistical model for 3d object detection applied to faces and cars," *IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, June 2000.

[33] D. Comaniciu, V. Ramesh, P. Meer, "Real-time tracking of non-rigid objects using mean shift," *IEEE International Conference on Pattern Recognition*, vol. 2, pp. 142-149, 13-15 June, 2000.

[34] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "A color based particle filter," in *First International Workshop on Generative-Model-Based Vision*, A.E.C. Pece, Ed., 2002.

[35] N. Johnson and D.C. Hogg, "Learning the distribution of object trajectories for event recognition," *Image and Vision Computing*, 14:609-615, 1996.

[36] N. Sumpter., A.J. Bulpitt, "Learning spatio-temporal patterns for predicting object behaviour," *Image and Vision Computing, 18(9),* pages 697-704, 2000.

[37] J. Owens, A. Hunter, "Application of the Self-Organising Map to Trajectory Classification," Proc. Third IEEE Visual Surveillance Workshop, 1 July 2000, Dublin, ISBN 0-7695-0698-4, pp. 77-83.

[38] J. Lou, Q. Liu, and T. Tan, W. Hu, "Semantic interpretation of object activities in a surveillance system," *Pattern Recognition, 2002,* Volume 3, 11-15 Aug. 2002, pp. 777-780.

[39] Intel VTune Performance Analyzers. http://www.intel.com/software/products/vtune/.

AUTHORS' BIOGRAPHIES

Trista P. Chen is a staff research scientist at Intel's Application Research Lab. She received her Ph.D. degree in Electrical and Computer Engineering from Carnegie Mellon University and M.S. and B.S. degrees from National Tsing Hua University, Taiwan. Her research interests include bridging the gap between computer vision and computer graphics, microprocessor architecture, and multimedia communications. Prior to joining Intel, she was with Nvidia, architecting the first video processor for Nvidia graphic processor units. She was with Sony Design Center and HP Cambridge Research Center in the summers of 2000 and 2001, respectively. Her e-mail is trista.p.chen at intel.com.

Alexander Bovyrin is a senior research scientist at Intel's Application Research Lab. He received his M.S. degree in Applied Mathematics and Cybernetics from Nizhny Novgorod State University in 1999 and his Ph.D. degree in the Theory of Dynamic System Control from Nizhny Novgorod State University of Civil Engineering in 2003. Before joining Intel Corp. in 2000, he worked on computer vision industrial applications. He is interested in computer vision and machine learning. His e-mail is alexander.bovyrin at intel.com.

Roman Belenov is a research scientist in Intel's Application Research Lab. His research interests include video compression and processing, multimedia software architecture, microprocessor architecture, and wireless networking. He received a Diploma in Physics from Nizhny Novgorod State University. His e-mail is roman.belenov at intel.com.

Konstantin Rodyushkin is a senior research scientist at Intel's Application Research Lab. He received his Ph.D degree in 2002 from Nizhny Novgorod State University. His research interests are in computer vision and machine learning. His e-mail is Konstantin.Rodyushkin at intel.com.

Alexander Kuranov has been a software engineer at the Intel Russia Research Center since 2001. He graduated from Nizhny Novgorod State University in 2002. His main interests are computer vision and data-mining algorithms. His e-mail is Alexander.Kuranov at intel.com.

Victor Eruhimov is a senior research scientist in the Intel Russia Research Center. He is leading a team of researches investigating the computational properties of algorithms in the area of computer vision and machine learning. He was a senior researcher in the Open Computer Vision project, working on the development of the OpenCV library and conducting research in computer vision. Victor received a Masters degree from the Advanced School of General and Applied Physics in the Institute of Applied Physics, Russian Academy of Sciences, in 1999. His e-mail is victor.eruhimov at intel.com.

Horst Haussecker is a principal engineer in Intel's Corporate Technology Group and manager of the Computational Nano-Vision research project. His research interests include model-based computer vision and application of digital image processing as a quantitative instrument. Horst is co-editor and author of two textbooks in Computer Vision, and he has authored more than 50 peer-reviewed technical articles. He received his M.S. and Ph.D. degrees in Physics from Heidelberg University, and prior to joining Intel in 2001, he was a researcher at the Xerox Palo Alto Research Center (PARC), where he was involved in image sequence analysis and information processing in sensor networks. His e-mail is horst.haussecker at intel.com.

For further information visit:

developer.intel.com/technology/itj/index.htm