

CITY UNIVERSITY OF HONG KONG
香港城市大學

**A Study on the Design of Real-Time
Active Visual Surveillance System for
Wireless Network**

用於無線網絡的實時影像
自動監察系統設計之研究

Submitted to
Department of Computer Science
電腦科學系
in Partial Fulfillment of the Requirements
for the Degree of Master of Philosophy
哲學碩士學位

by

CHIU Kar Ho Calvin
趙嘉豪

July 2005
二零零五年七月

ABSTRACT

In this thesis, we study the important issues in the design of an efficient real-time Wireless vIsual Surveillance systEm (WISE). Due to the limitation of the wireless network and the processing capacity of mobile devices, two important factors are considered (1) to minimize the video workload on the wireless network, and (2) to minimize the processing workload at the front-end video capturing unit.

For a single static camera setting, the first factor is addressed by the proposed *cooperative framework for semantic filtering* of video frames. Instead of forwarding every frame to the backend server for analysis and query evaluation, only those visual objects which are relevant to the monitoring queries are forwarded. The processing workload at the front-end video capturing unit is controlled and reduced by the evaluation frequency of visual objects in such a way that a visual object under monitoring will not be lost tracked and, at the same time, no extra processing workload is consumed.

In addition, an efficient visual object detection and segmentation algorithm – Integer Hybrid Algorithm – is proposed for the front-end video capturing unit which does not have a floating point processing unit (FPU). It combines the advantages as well as compensates the disadvantages from each other from two base algorithms, namely, background subtraction and temporal differencing. The result is a robust algorithm for visual object extraction for processing power limited device.

We further study the job scheduling issues in which a continuously moving panning camera is installed. Due to movement of panning camera, visual object may be out of the camera range at the pre-determined evaluation frequency. In addition, processor overloading may occur due to the dynamic properties of visual

objects together with the movement of camera. Basing on our previous study on static camera setup, a predictive scheme is proposed to control the evaluation workload by rearranging jobs into other execution time window that has less loading and, more importantly, the object is mostly likely being visible by the camera. The loading is further reduced to within the processor capacity by adaptively adjusting the analysis resolution at the cost of scarifying the accuracy of visual object extraction.

TABLE OF CONTENTS

ABSTRACT	I
ACKNOWLEDGEMENTS	III
TABLE OF CONTENTS	IV
1 INTRODUCTION	1
1.1 The Need for Active, Real-Time, and	
1.2 Wireless Visual Surveillance System	1
1.3 Motivation	4
1.4 Visual Surveillance Pipeline	5
1.5 The Proposal	9
2 RELATED WORKS	11
2.1 Introduction	11
2.2 Related Approach	13
2.3 Strategy on Processing Reduction	14
2.4 Methods for Visual Object Detection	15
2.5 Load Management	18
3 COOPERATIVE SURVEILLANCE FRAMEWORK	19
3.1 Design Consideration	19
3.2 System Architecture	21
3.3 Hierarchical Object Model	24
3.4 Monitoring Query	31
3.5 Chapter Summary	34
4 ADAPTIVE OBJECT-BASED EVALUATION	35
4.1 Introduction	35
4.2 Visual Object Detection	37
4.3 Predictive Tracking	44
4.4 Object Confidence	47
4.5 Adaptive Evaluation	48
4.6 Chapter Summary	52

5	LOAD MANAGEMENT WITH CONTINUOUSLY PANNING CAMERA	53
5.1	Introduction	53
5.2	Challenges	54
5.3	Processing Cost Estimation	56
5.4	Time-To-Event and Next-Evaluation-Time	58
5.5	Load Aware Job Rearrangement	61
5.6	Dynamic Multi-Resolution Analysis	63
5.7	Combining LAJA and DURA	67
5.8	Chapter Summary	69
6	IMPLEMENTATION AND PERFORMANCE STUDY.....	70
6.1	Introduction	70
6.2	Implementation Details	72
6.3	Performance Study on APOBE	76
6.4	Performance Study on LAJA and DURA	86
6.5	Chapter Summary	91
7	CONTRIBUTIONS AND CONCLUSION	93
	REFERENCES	96

LIST OF TABLES

Table 6.1: Execution sequence of each algorithm

Table 6.2: Detection results of HYB, HYT and IHA.

Table 6.3: Summary statistics of using different combination of the two approaches.

Table 6.4. Summarized system load statistics.

LIST OF ALGORITHMS

Algorithm 5.1. Determining the correct ROI for TTE calculation.

Algorithm 5.2. Load-Aware Job Arrangement.

LIST OF FIGURES

Figure 1.1: Typical Intelligent Visual Monitoring Pipeline

Figure 3.1: System Components

Figure 3.2: The initial and monitoring phases

Figure 3.3: The interaction between VMA and RSA under the monitoring phase of the cooperative surveillance framework.

Figure 3.4: Example on user-defined static VO.

Figure 3.5: Sample object hierarchy.

Figure 3.6: Park scenario with two VOs, Human and Car. Static VO is denoted by solid line, whereas dotted line denotes Dynamic VO.

Figure 3.7: Master HOM “Home” contains two Local HOMs “Living Room” and “Bedroom”. Whenever there is update on the HOM at VMA, the change is also sent to the RSA keeping the Master HOM up-to-date.

Figure 4.1: The Integer Hybrid Algorithm.

Figure 4.2: Example of Active-Pixel Space.

Figure 4.3: Sample detection result – top-left: actual scene; top-right: estimated background model; bottom-left: motion-history image; bottom-right: detected visual objects.

Figure 4.4: The Dynamic Range (outer box) is extended towards the direction of object motion.

Figure 4.5: Strategy of adaptive object evaluation.

Figure 4.6: Example schedule of each processing cycle (Assume EC_{rate} is 1).

Figure 5.1. A continuously moving panning camera is monitoring the scene.

Figure 5.2. An evaluation schedule consists of detection jobs, and tracking, and evaluation jobs arranged at different execution windows.

Figure 5.3 When calculating the Time-To-Event, only ROIs that the object is approaching are considered. In this example, only ROI2 is considered.

Figure 5.4. After the evaluation of object 5 at execution window t , it is arranged to $t+3\delta t$ according to the NET. With LARA, it is allowed to be re-arranged to another execution window with estimated loading being the minimum and at the same time the object is predicted to be visible. The rectangle filled with cross means that object 5 is predicted invisible at execution window $t+4\delta t$. The final execution window of object 5 being arranged to is $t+2\delta t$.

Figure 5.5: The job sequence where DURA and LARA apply.

Figure 6.1a: The WISE system interface.

Figure 6.1b: Customizable threshold setting in WISE.

Figure 6.2: Internal structure of the VMA.

Figure 6.3: Internal structure of RSA

Figure 6.4: User can define the region of interest (red rectangle in the middle) and monitoring rules associating with. The ROU receives HOM updates from a low bit rate stream coming from the VMA. Visual objects are extracted from the HOM simulating the remote scene.

Figure 6.5: (a) The original frame; foreground pixels extracted by (b) Temporal Differencing; (c) Background Subtraction; and (d) Integer Hybrid Algorithm.

Figure 6.6a: Different values of EC_{rate} affecting the number of evaluations required.

Figure 6.6b: Different values of EC_{rate} affecting the accuracy in trajectory capturing.

Figure 6.6c: The relationships between the EC_{rate} , no. of evaluation, and MSE.

Figure 6.7: Average network workload for communication between RSA and VMA.

Figure 6.8: Job statistics of different schemes.

Figure 6.9: Event detection performance.