Activity Aware Video Collection to Minimize Resource Usage in Smart Camera Nodes (Extended Abstract)

Faisal Z. Qureshi
Faculty of Science, University of Ontario Institute of Technology
Oshawa, ON, Canada
faisal.qureshi@uoit.ca

We envision future video sensor networks comprising tether-less smart camera nodes capable of supporting a variety of applications, ranging from video surveillance to traffic management, smart environments to ecological monitoring, etc. A key difference between video sensor networks and traditional multi-camera systems is that the later typically are not concerned with power, storage, and bandwidth usage. Power requirements, especially, must be considered when designing tether-less smart camera networks, since the operational life of a camera node is closely tied to the available power. Video capture and processing performed on a camera node and the communication between nodes needed to carry out collaborative sensing tasks impact power usage of these camera nodes. Therefore, one must devise strategies to minimize video capture and processing at each node and communication between nodes in order to reduce power consumption at each node, thereby increasing the operational life of a video sensor network.

Pinto *et al.* [3] studied energy consumption in wireless multimedia sensor networks under three scenarios: 1) in the first case camera nodes perform no local processing and transmit video continuously; 2) in the second case the camera nodes perform image analysis to decide when to transmit an image; and 3) in the last case two cameras work together, each performing local image analysis, to decide when to collect or transmit the video to a central receiver (sink). Their results support the opinion held by many in the smart camera networks community that local processing can be effectively exploited to minimize power consumption and bandwidth usage. Earlier work by Kulkarni *et al.* [2] also demonstrated that it is possible to design camera networks that achieves both low latency and energy efficiency. They developed SensEye, a three-tiered camera network comprising Cyclops (resolution 128×128 at 10 frames-per-second), webcams (resolution 640×480 at 30 frames-per-second), and pan/tilt/zoom (resolution 1024×768 at 30 frames-per-second) cameras. Cyclops are always active and are responsible for waking up higher-level sensors upon detecting changes in the scene.

Inspired by these research efforts, we observe that each camera node must do as little as possible while still meeting the sensing requirements set out by the currently active task(s) in order to minimize its energy consumption. In order to design such camera nodes we must first overcome two challenges: 1) spell out the minimum sensing requirements for different observational tasks and 2) develop sensing strategies guaranteed to meet the sensing requirements defined by the currently active tasks.

We are currently developing control strategies that enable a smart camera node to perform activity aware imagery collection. In our work we consider camera nodes with controllable sensing parameters, such as:

- the ability to choose between different image capture resolutions (320×240), (640×480), etc.;
- the ability to pick a regions-of-interest in the captured image for subsequent processing;
- the ability to transmit a sub-region of the captured image, instead of the entire image, to a central receiver or neighboring cameras;
- the ability to store a sub-region of the captured image, instead of the entire image locally; and
- the ability to choose between different image capture rates (1 FPS, 5 FPS, 15 FPS, etc.).

Furthermore, we assume that our camera nodes have local vision processing routines listed below:

- change detection;
- background subtraction;
- blob detection and blob counting;

- pedestrian detection;
- face detection/head detection; and
- appearance-based pedestrian tracking.

These are now standard computer vision routines and are available in many computer vision libraries, including OpenCV [1]. Given such camera nodes we can define high-level procedures that enable our camera nodes to collect the minimal amount of imagery needed to successfully carry out an observation task.

The key idea here is that our camera nodes are able to identify "what *might* be going on" in the scene and use that information to pick an appropriate sensing plan. The implicit assumption being that our camera nodes are able to figure out what is going on in the scene using low resolution, low frame-rate video.

Consider, for example, the situation when there is only a single individual in the scene. Pedestrian recognition typically requires a single high resolution image; where as, pedestrian tracking needs to be performed continuously but it is possible to track pedestrian at much lower image resolutions and frame rates. At the same, we need to set a capture rate so as not to miss important events, such as pedestrian leaving behind an item or interacting with some object in the scene. It is conceivable to set capture rates based upon scene complexity. Scenes with low activity do not need to be monitored at high frame rates. Or we might choose a sensing plan that increases capture rate as the pedestrian approaches "hotspots" in the scene.[1]

It is also possible to extend these ideas to scene with multiple individuals. Consider a situation with two individuals. We capture the scene at low frame rates when pedestrians are far apart, switching to higher capture rates as these individuals move close to each other in order to capture any short-duration interactions between the two individuals. It is also conceivable to use two cameras capturing video out-of-sync at low capture rates when these individuals are close to each other.

It is important to realize that in order for such a scheme to work, the camera nodes must be able to guess not only "what is happening in the scene currently," but also "what will be happening in the scene in the near future." This is due to the fact that it is not possible to execute a new sensing plan instantaneously. There might latencies associated with changing frame rates, capture rates, etc. More importantly, there might power/energy penalties associated with changing the sensing plan.

[1] Borrowing computer games jargon, we refer to regions/items in the scene with which an individual can interact as "hotspots."

Epilogue

We propose to develop smart camera nodes with tunable sensing parameters that are able to loosely infer scene activity from low resolution, low frame-rate video. The scene activity information is then used to pick a sensing plan needed to capture just enough imagery so as to meet the sensing requirements set by the currently active tasks. By necessity the camera nodes must be able to anticipate future activities to successfully handle the latencies associated with executing a new sensing plan. We are currently working on a prototype to explore these ideas and our early results appear promising.

References

[1] G. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, Inc., Sept. 2008.

[2] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu. SensEye: A multi-tier camera sensor network. In *Proc. ACM International Conference on Multimedia*, pages 229–238, Amsterdam, The Netherlands, July 2005. ACM Press.

[3] A. Pinto, Z. Zhang, X. Dong, S. Velipasalar, M. C. Vuran, and M. C. Gursoy. Energy Consumption and Latency Analysis for Wireless Multimedia Sensor Networks. In *Proc. IEEE Wireless Communication and Networking Conference (WCNC)*, pages 1–5, Sydney, Apr. 2010.