Beyond stream processing??? A distributed vision architecture for the Internet of Things

Beyond Stream Processing - a Distributed Vision Architecture for the Internet of Things

Peter Corcoran, *IEEE Fellow*
Center for Cognitive, Connected & Computational Imaging (C3I), National University of Ireland Galway.
dr.peter.corcoran@ieee.org

***Abstract*-- An 'edge-based' image processing architecture for the internet of things (IoT) is devised, drawing on the existing image processing pipelines that are implemented in today's imaging modules in digital cameras and smartphones. A key element of this IoT framework is that image or video data does not have to be streamed across the network, but can be largely processed on the sensing nodes with metadata transmitted to intermediate control nodes that can detect certain events/conditions and trigger corresponding actions.**

I. Introduction

Consumer imaging systems have evolved rapidly and the computational capabilities of today's smartphones are quite astonishing. In additon to advanced image signal processors there are dedicated hardware subsystems that can provide advanced face detection & analysis capabilities. These technical capabilities are being driven strongly by market demands. This leads to the key focus of this paper – can we leverage the rapidly evolving smartphone imaging ecosystem to develop insights into how a potential imaging and computer vision architecture might look for a distributed Internet of Things computer vision architecture.

A. Computational Imaging in Smartphones

In the smartphone much of the raw image processing has been devolved into an image prodcessing pipeline (IPP) that extracts and pre-processes image frames prior to compression. The initial pre-processing mirrored that of digital cameras to include de-mosaicking, white & color balance and multi-frame operations such as auto-focus. More recently additional functionality, in particular detection and analysis of face regions and facial features has also appeared in the hardware pipeline. Often the hardware outputs feed into software post-processing on a dedicated image signal processor (ISP). The ultimate outputs are locations and states of face regions and face features but similar object matching techniques can be used to detect and analyse other scene elements – vehicles, animals, body extremities (arms & legs), buildings and so on. Multi-frame processing enables more sophisticated techniques such as auto-focus, face tracking, and HDR image enhancements.

A key point is that much of this pre-processing is performed prior to the image/video data reaching the main application processor (AP) of the smartphone. Now consider it the imaging front-end were to be separated from the AP and connected instead to a network to provide a distributed processing model. The market forces of consumer sector are already driving this front end to become more energy efficient, with more advanced capabilities including hardware for dynamic distortion correction and intra and inter-frame motion analysis. With the addition of some additional hardware elements this can enable a wide range of 'smart imaging' IoT systems and applications.

B. Internal Imaging Architecture of Smartphone Systems

Figure 1 below shows the architecture of a state-of-art smartphone imaging system including the main data stores, and delineates which elements take part in *computational imaging* and a more specific variant employing *hybrid computational imaging* (HyCI). The HyCI takes advantage of the most computationally intensive parts of an image processing algorithm by implementing these in hardware. Computation is effected as rows of data are clocked from the image sensor. The outputs are made available in a temporary data store and are available for additional software processing until the following image frame is offloaded.

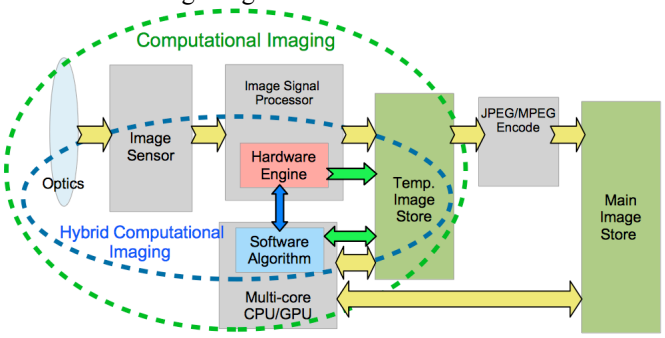

Fig. 1. Architecture of Smartphone Imaging System; yellow arrows show main image data flow; green arrows show image primitives & metadata flows; blue arrows show control/configuration messages.

II. Architectural Elements

A. Vision Sensor Nodes (VSNs)

The camera modules available in today's smartphones already provide many of the capabilities required in tomorrow's smart imaging sensors for the IoT. Most imaging sensors are not just simple cameras, but specialized image processing engines that can analyze image frames and extract *metadata* that can be useful for additional processing. They can process an imaged scene sufficiently to determine a particular event has occurred, or as aspect of the scene has exceeded a threshold metric, or a particular object or person is identified or authenticated. Put simply, these nodes can act as independent processing elements, detecting that complex criteria have been met within the imaged scene.

B. Back-End Data Nodes (BEDs)

A second key architectural element of our architecture is that of a data collection and analysis node. This node serves as an aglommerator paired with multiple VSNs, its purpose to collect metadata, raw data portions, and other image primitives from multiple VSNs. It then performs specific processing on these data elements and may reach an action threshold, triggering additional system events. In some cases the data collection end-point is a secure cloud-based service; other use cases provide for a home-terminal or gateway as a central node for actuating and/or processing system-level events. A more detailed outline of the system architecture will be given at ICCE 2016.

III. Practical Use Cases

To better understand how a distributed VSN/BED network might be used to solve practical problms lets consider some of the functionality that is available on today's devices and how it might map into our distributed vision processing network.

A. VSN Capabilities

1) Face Detection & Tracking

Face detection has largely settled on the techniques of Viola & Jones [1] relying on boosted rectangular haar classifiers. A hardware engine to implement the classifier chain provides a vision sensing node that can effectively detect faces as they are directed at the imaging camera. Extending the classifier sets and implementing a frame-to-frame algorithm enables more sophisticated processing and analysis that could provide information not only on the number of faces, but also tracking faces to measure the engagement times of individual facial regions. Knowing the size of face regions tells you how near/far they are and a basic recognition algorithm provides information on re-engagement by the same person [2], [3].

Note that this *metadata* does not requires any video or image frames to be transmitted over the network link – only the *metadata* about the image scene is sent to a BED node.

2) Generic Objects – Vehicles, Animals & People

An example of a hardware implementation for advanced face detection & tracking is given in [4]. In subsequent improvements this now incorporates updatable templates that can by dynamically programmed with a broader range of classifier chains. As a consequence it is possible to adapt VSNs to different object classes or multi-component objects such as the human body. Human motion tracking involves tracking a multi-component-object sequence. While this is a more complex task than simple face-tracking the proposed architecture enables some of the computational effort to be offloaded to a BED.

B. System Level Capabilities

1) Object Recognition

The detection & tracking capabilities outlined in the previous sections do not directly allow recognition/authentication of a particular object – this requires a larger dictionary than can be stored locally. However a sequence of raw object views can be stored on the BED and a comparison effected via a secure cloud service, or a local gateway system with more storage and processing capability.

2) Triggering on Detections

So far the sensing capbilities of this distributed vision architecture have been highlighted. But to effect real change in how vision sensing is implemented it is also important that system events can be actuated. The simplest criterion for triggering an event is the detection of a pre-programmed object. For a practical use case, consider a consumer device that turns on when someone looks at the device.

3) Track and Trigger

A more sophisticated capability it to track an object, determining its dynamic state and triggering when changes are detected or some threshold is exceeded. A simple use case is to add a 'like' to an image or video clip if the viewer is determined to be smiling which the image/clip is displayed.

4) Track, Analyze and Trigger

A more complex use case is that of an elderly person living alone who is monitored to determine their *activities for daily living* (ADL) capabilities – these are used as a metric to judge a person's ability for independent living. Different VSNs can be placed to monitor activity in the kitchen, living and bedroom environments. As no video is recorded there is not a privacy issue. Each VSN tracks certain activities by monitoring body movements; these data are analyzed and statistical metrics compiled by one or more BEDs that trigger appropriate events when certain ADL metrics are not achieved (e.g. no preparation of meals) or on detection of specific events (e.g. fall detection). The most common action is a simple notification to the social worker responsible to indicate that a personal visit might be necessary; on a more extreme case a priority notification to emergency services and/or a neighbouring keyholder.

More technical and use case details will be presented at ICCE 2016, including an outline of a programming model for such a distributed vision system.

Acknowledgment

This research is funded under the *SFI Strategic Partnership Program* by Science Foundation Ireland (SFI) and FotoNation Ltd. Project ID: 13/SPP/I2868 on *Next Generation Imaging for Smartphone and Embedded Platforms.*

References

[1] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proc. 2001 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition. CVPR 2001*, vol. 1, 2001.

[2] P. Corcoran and G. Costache, "Automated sorting of consumer image collections using face and peripheral region image classifiers," *Consum. Electron. IEEE Trans.*, vol. 51, no. 3, pp. 747–754, Aug. 2005.

[3] P. Corcoran, C. Iancu, F. Callaly, and A. Cucos, "Biometric Access Control for Digital Media Streams in Home Networks," *Consum. Electron. IEEE Trans.*, vol. 53, no. 3, pp. 917–925, Aug. 2007.

[4] P. Bigioi, C. Zaharia, and P. Corcoran, "Advanced hardware real time face detector," *IEEE Int. Conf. Consum. Electron. Electron. (ICCE),* 2012.