This document provides additional material for the paper:

Energy consumption models for smart-camera networks
J. SanMiguel (show email) and A. Cavallaro
Submitted to IEEE Transactions on Circuits and Systems for Video Technology, 2015

APPENDIX – DESCRIPTION OF THE EXPERIMENTS WITH REAL MEASUREMENTS

This experiment demonstrates the utility of the proposed power models for smart cameras by studying the battery drain of a laptop (model Portege Toshiba R-700 http://goo.gl/yrzOD8/) running an Ubuntu OS 14.04 64bit (fresh install) where sensing and communication capabilities are provided via USB devices.

Battery readings are obtained from the OS kernel system file */sys/class/power_supply/BAT0/uevent*. where the discharge over time (uW) has been extracted each 5 seconds. Note that reading kernel statistics is widely accepted to estimate energy or latency measurements. Moreover, an USB multimeter was employed to verify the enabling and disabling as well as the consumption of the USB devices (http://goo.gl/FfdEzS)

This experiment has been divided in three phases:

1) **Get the measurements and model fitting using Eq (4)**. Three programs are created to get measurements (8 runs, each run has 200 samples at a rate of 5 seconds/sample):

   (a) CPU modeling for model Intel® Core™ i5-450M (1.20 GHz-2.40GHz)
   We assumed that power in the sleep state ~ 0 mW.
   The objective is to measure the power for CPU running (and idle) at the desired frequency
   Testing conditions:
   - Screen brightness set to minimum
   - Disable all network interfaces, bluetooth, usb ports and CPUs (except CPU0)
   - Sets CPU0 to certain frequency values (user selected, “on-demand” settings)

   Experiments (running and idle) for 1.20GHz, 1.60GHz, 2.00GHz and 2.40GHz
   Unix tools employed: acpitool, libusb, ifconfig, cpufreq-tools

   (b) USB webcam QuickCam® Ultra Vision (http://goo.gl/dbn6YU)
   The objective is to measure:
   - Power for calibration: disable USB, system power
   - Power sleep: USB port powered on
   - Power idle: USB port powered on & camera configured (no streaming)
   - Power active: USB port powered on & camera configured (streaming)

   Testing conditions:
   - Screen brightness set to minimum
   - Disable all network interfaces, bluetooth, usb ports and CPUs (except CPU0, set to 1.20GHz)
   - Enable usb port for the USB webcam

   Experiments (four modes) for 640x480 framesizes using 5, 15 and 25 fps
   Unix tools employed: acpitool, libusb, ifconfig, cpufreq-tools (and opencv library)

   (c) AC600® Wireless Dual Band USB Adapter (http://goo.gl/Mv9COc)
   Created a server & client to perform the measurements. The server only receives and sends data, being located on a network computer. The client is installed in the laptop and connected to a wireless router (< 1m distance), reducing the power to transmit/receive signals.
   The objective is to measure:
   - Power sleep: USB port powered on
   - Power idle: USB port powered on & adapter configured (no rx/tx)
   - Power active RX: USB port powered on & receive data
   - Power active TX: USB port powered on & transmit data

   Testing conditions:
   - Screen brightness set to minimum

- Disable all network interfaces, bluetooth, usb ports and CPUs (except CPU0, set to 1.20GHz)
- Enable usb port for the USB Adapter
- Enable wireless interface “ra0” (the one of the USB Adapter)
- Sets the maximum bitrate to 1Mbps (rx & tx)

Experiments (four modes) for 1 Mbps transmission: sleep, idle, RX & TX
Unix tools employed: acpitool, libusb, ifconfig, cpufreq-tools, wondershaper

2) **Design and development of the video application**.
A video processing application was created to make use of sensing, processing and communication. This application consists on a tracking-by-detection approach based on the upper-body parts of people which transmits visual descriptors that can be used for people re-identification purposes. The OpenCV library was used to implement the visual analysis algorithms.

The obtained measurements consisted in 8 runs (each run has 200 samples at a rate of 5 seconds/sample) obtained from the battery drain:

We assumed that power in the sleep state ~ 0 mW.
The objective is to measure the power for whole application and the activation time of each module (i.e. sensing, processing and communication times).
Testing conditions:

- Screen brightness set to minimum
- Disable all network interfaces, bluetooth, usb ports and CPUs (except CPU0)
- Set CPU0 to 2.40GHz (user selected, “on-demand” settings)
- Enable usb ports for wireless adapter and webcam
- Enable wireless interface “ra0” (the one of the USB Adapter)
- Set the maximum bitrate to 1Mbps (rx & tx)

Experiments for capturing and processing 640x480 frames with 5 fps, 15fps and 25fps.
Unix tools employed: acpitool, libusb, ifconfig, cpufreq-tools, wondershaper