Introduction

Running computation intensive tasks on mobile devices having limited resources leads to longer execution time and more battery power consumption.

Computation Offloading where the tasks are offloaded to a more powerful cloud server could be used to overcome these issues.

Offloading tasks is not always the best solution.

Need for a cost evaluation method to decide whether to offload a mobile task execution to remote cloud server or not.

Objective

To develop a cost evaluation method which considers both time and energy consumption while make the offloading decision.

Method

Cost function = α \left( \frac{T_{\text{cloud}} - T_{\text{local}}}{T_{\text{local}}} \right) + (1-α) \left( \frac{E_{\text{cloud}} - E_{\text{local}}}{E_{\text{local}}} \right)

- \( T_{\text{local}} \) = Time to execute the module locally on the mobile CPU
- \( T_{\text{cloud}} \) = Time to execute the module when it is offloaded to a cloud server
- \( E_{\text{local}} \) = Energy consumption of local execution of the module on mobile CPU
- \( E_{\text{cloud}} \) = Energy consumption when the module is offloaded to a cloud server
- \( \alpha \in [0,1] \) is the weighting factor

Results

![Flow diagram](image)

Android face recognition application has been developed using java openCV library.

Mobile device: LG Nexus 5 (Qualcomm MSM8974 snapdragon 800x 4 2.2656GHz processor , 2 GB of RAM and 2300 mAh LiPo battery)

Cloud Server: Dell Inspiron 15R laptop (Intel i5-3120M @2.50Ghz processor and 8Gb of RAM )

The mobile device and laptop are connected using a Wi-Fi network and LipErMI is used to transfer the input and output data between the two

The execution time and the energy consumption of the face recognition application are measured for images of various sizes and for various set of training images.

The measured values of execution time and the energy consumption of both local mobile CPU execution and the offloaded cloud server execution of a task are used by the cost function to decide whether to offload a task or not

Conclusions

- Reduction in execution time and energy consumption by offloading gets bigger as the size of the training images increases.
- As the size of the training set reaches 100, offloading becomes the default choice for all the values of alpha.

References