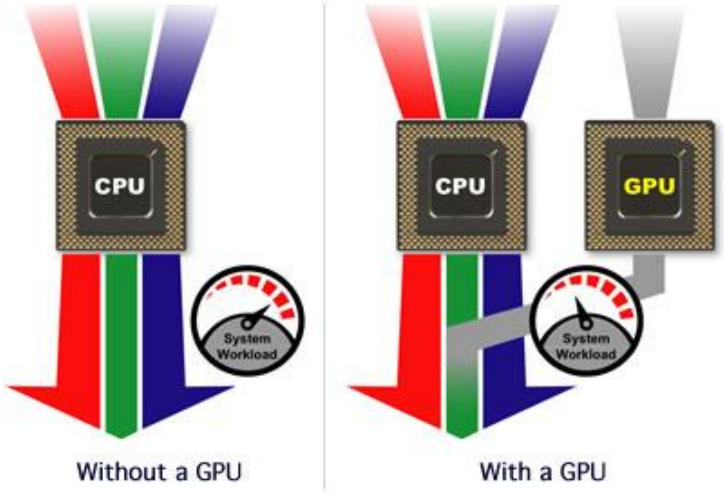
Challenges and Techniques in Using CPUs, GPUs, and FPGAs for Mobile Embedded Vision Applications

Ken Lee

VanGogh Imaging

Overview

- Potential Applications and Challenges
- CPUs (single and multi-core)
  - Advantages and Disadvantages
- GPUs (massively parallel)
  - Advantages and Disadvantages
- FPGAs (hardware acceleration)
  - Advantages and Disadvantages
- CPU vs. GPU vs. FPGA Comparison
- Proposal: Hybrid Processing ➔ CPU + GPU + FPGA(HW)

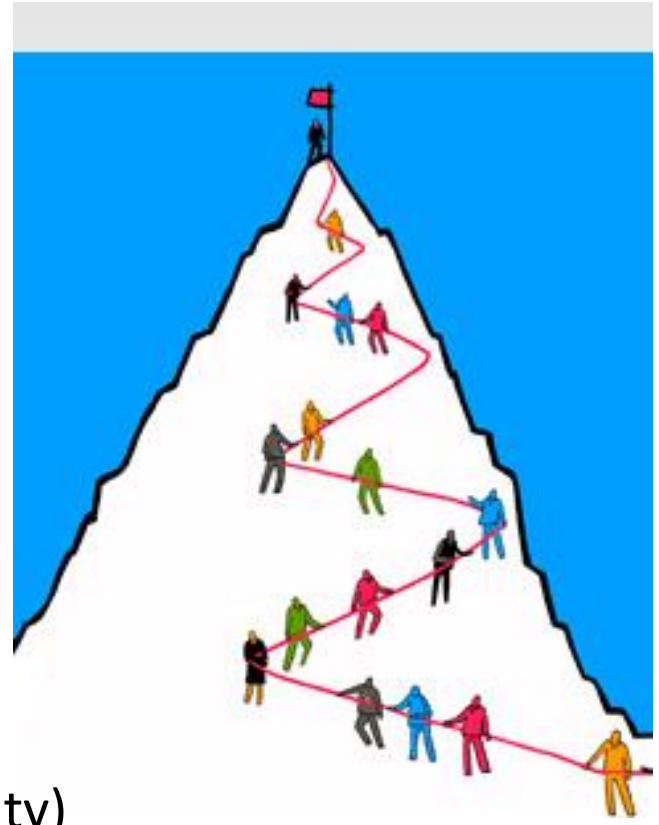


“Leave-on” Vision Applications

- “Leave-on” – Always running
- There are many interesting leave-on applications
  - Gesture recognition
  - Augmented reality
  - Event driven & computational photography
  - Security motion detection
  - Security facial recognition
  - News video analysis
  - Process inspection
  - Industrial control
  - Driver assist, car entertainment

Challenges in Mobile Vision Processing

- Not a Problem
  - Silicon is cheap
  - Number of good available algorithms
- Problems
  - Power is precious
  - Dataset is very large
    - e.g., > 50 Mbps for real-time gesture recognition
  - Existing processor architectures are well suited for legacy applications, but not tailored for vision processing
- Conclusion: These challenges have hindered the adoption of most leave-on vision applications (e.g., augmented reality)



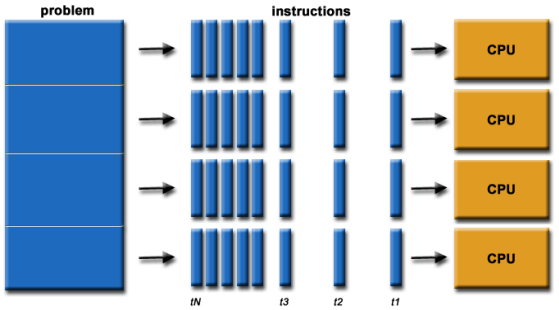
Vision Processing on CPUs

- Advantages
  - Large amount of open-source commercial libraries such as Matlab and OpenCV makes the implementation easy
  - Easy system integration
    - Windows, iOS, Linux platform
    - Readily available drivers for cameras and other external sensors

- Disadvantages
  - CPUs are not well suited for vision algorithms which require a large amount of parallel computing
  - Vision algorithms can monopolize all available processing, memory, and bus bandwidth
  - Compromises are required for design implementation
    - Smaller dataset (e.g., lower frame rate or lower image resolution)
    - Less robust algorithm

Vision Processing on GPUs

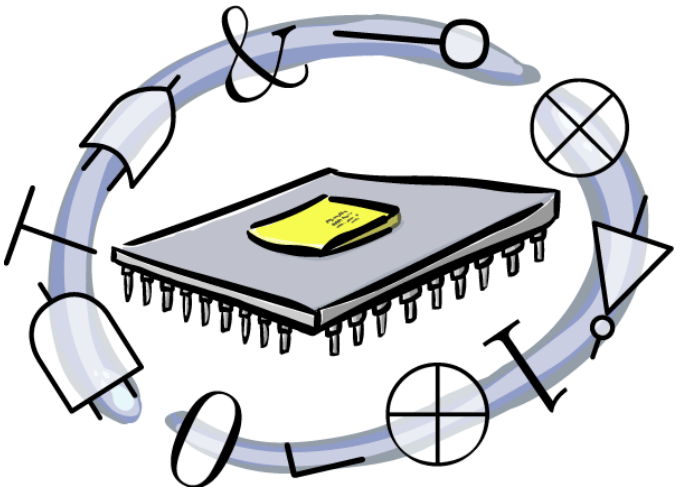
- Advantages
  - Relatively easy to implement
    - Both OpenCV and Matlab now support GPU libraries
    - CUDA & OpenCL high-level languages
  - Parallel computing is a good fit for certain simple vision processing algorithms



- Disadvantages
  - CPU resource is still needed for system level integration
  - GPU architecture is not well suited for certain complex vision algorithms due to slow data transfer between cores
  - Difficult to find GPU programming expertise

Vision Processing on FPGAs

- Advantages
  - Complex algorithms can be custom designed for maximum speed
  - Data transfers between internal circuits are very fast
  - Programmability allows for multiple algorithms to share the same resources as long as they are not running concurrently



- Disadvantages
  - Still need CPU for system processing and GPU for graphic processing
  - More expensive than custom hardware implementation, especially if resources cannot be shared across multiple applications
  - Poor static power consumption due to smaller transistors and related leakage current

Vision Example 1: 2D-Convolution in Gaussian Filter Application: noise removal, feature extraction

$$S(x,y) = \sum_{dx=-w}^{w} \sum_{dy=-w}^{w} I(x+dx, y+dy) \cdot G(dx, dy)$$

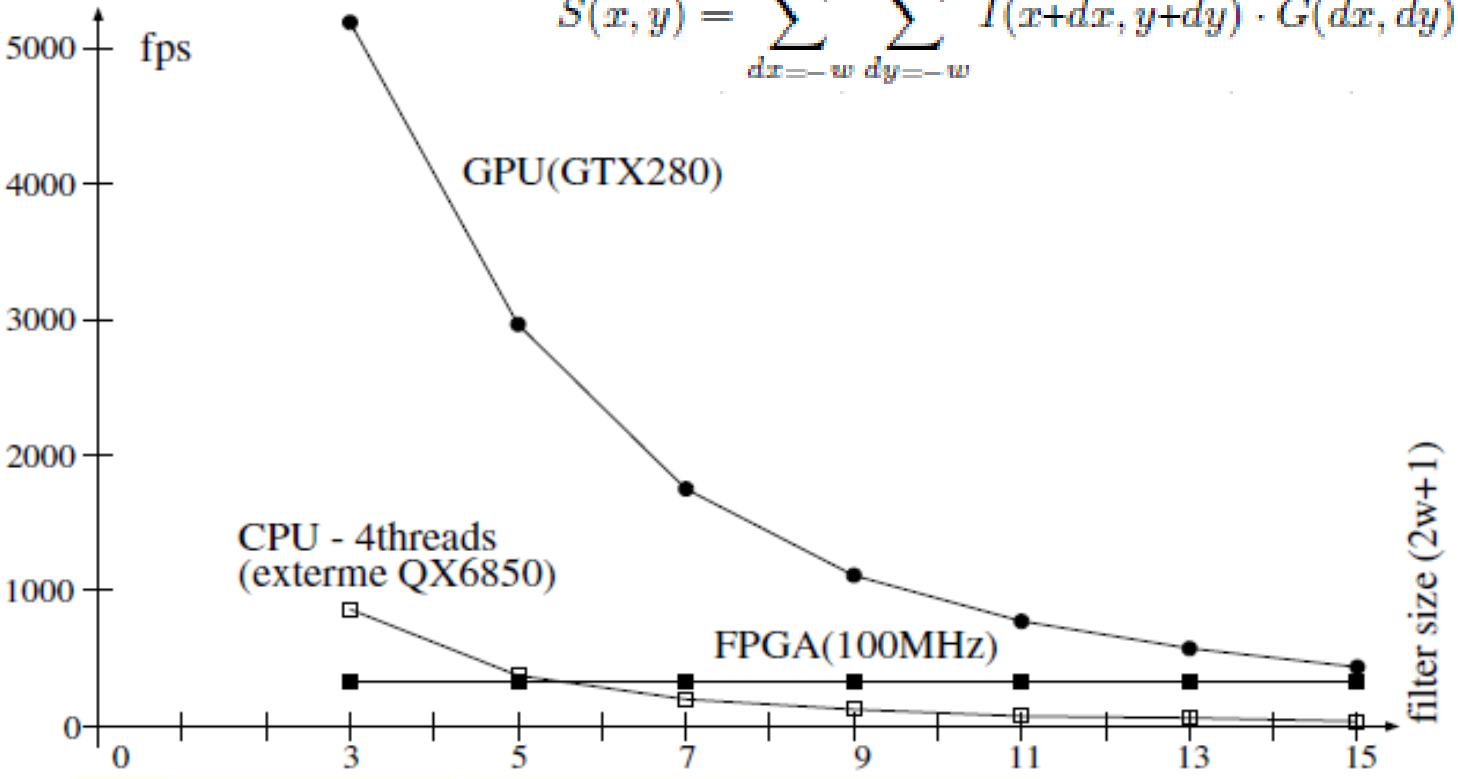


Fig. 8. Performance of two-dimensional filters

Source: "PERFORMANCE COMPARISON OF FPGA, GPU AND CPU IN IMAGE PROCESSING"
by *Shuichi Asano, Tsutomu Maruyama and Yoshiki Yamaguchi 2009 IEEE*

Vision Example 2: Disparity Map for Depth Map Applications: Augmented Reality, Gesture Recognition

FPGA wins because of its on-chip memory, customized circuit

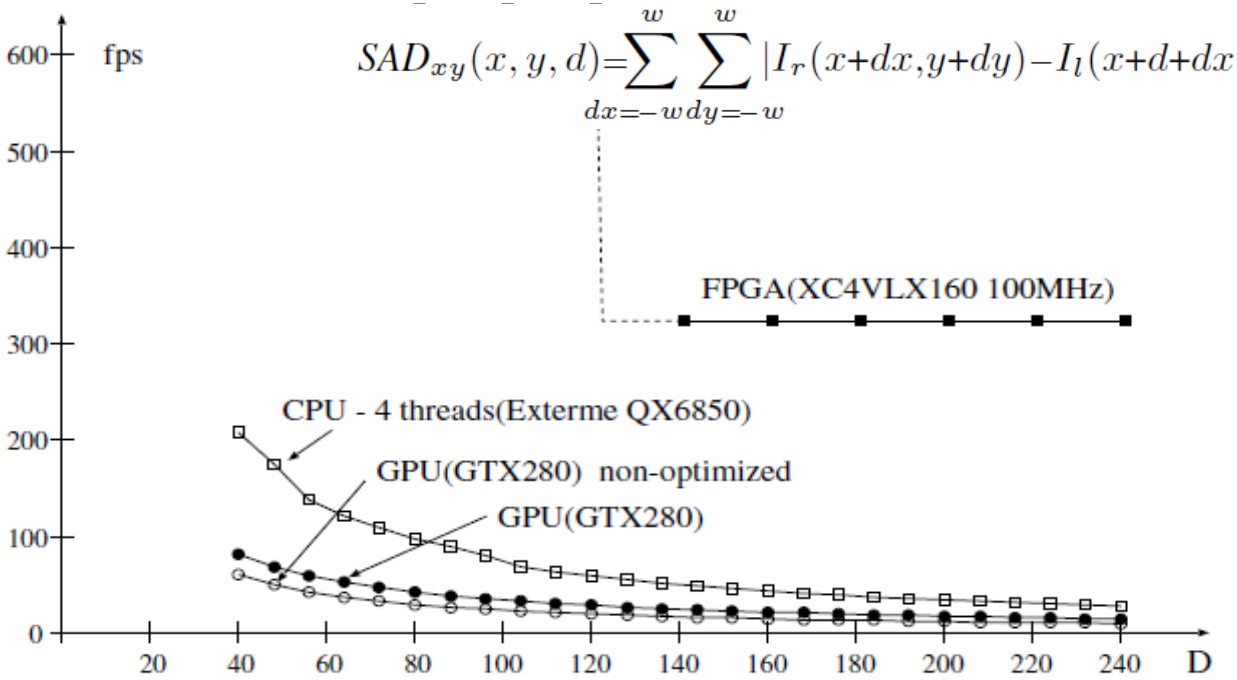


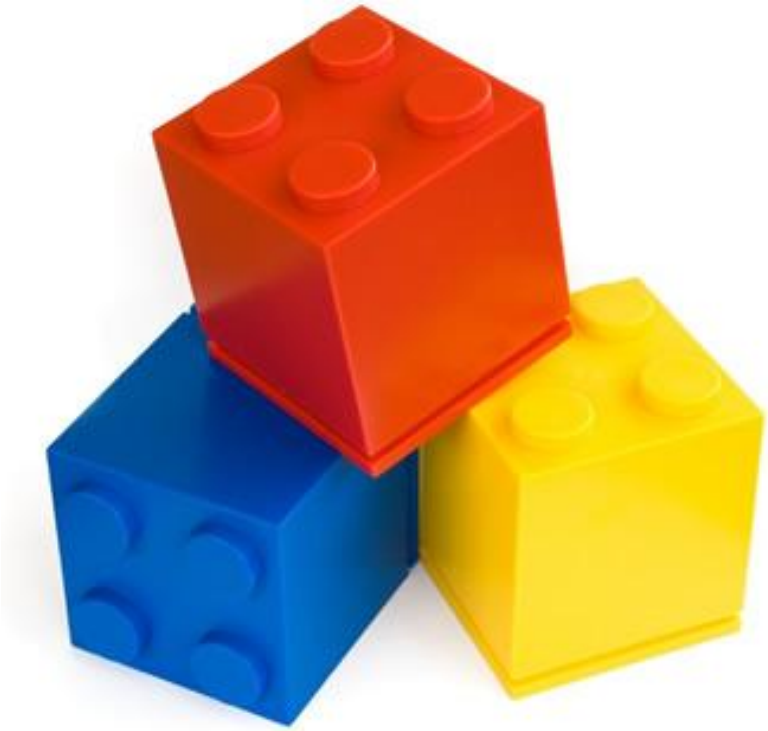
Fig. 9. Performance of the stereo-vision

Source: "PERFORMANCE COMPARISON OF FPGA, GPU AND CPU IN IMAGE PROCESSING" by *Shuichi Asano, Tsutomu Maruyama and Yoshiki Yamaguchi 2009 IEEE*

Summary

CPU+GPU+FPGA/HW is Best for Vision Processing Applications

- GPU: Large number of simple algorithms
- FPGA/HW: Complex algorithms
  - FPGA can be a shared resource across multiple algorithms not running concurrently
  - Custom HW can be used for real-time processing



Appendix: VanGogh Imaging Overview

- Founded in 2008
  - Located in McLean, VA
  - Embedded Vision Technology Provider (Software Only) for Mobile
    - Low-cost, low-power, small form-factor
  - NOT a processor vendor
- Current Customers
  - Industrial
  - Agricultural
  - Mobile and Consumer Electronics
- Special Expertise
  - 3D Computer Vision Processing (clean-up, merging, etc.)
  - 3D Computer Vision Analysis (measurement, recognition, etc.)
  - 2D Computer Vision Analysis

- ***Mission ➔ "Provide 3D Computer Vision for Mobile Applications"***