Distributed Vision Processing in Smart Camera Networks

CVPR-07

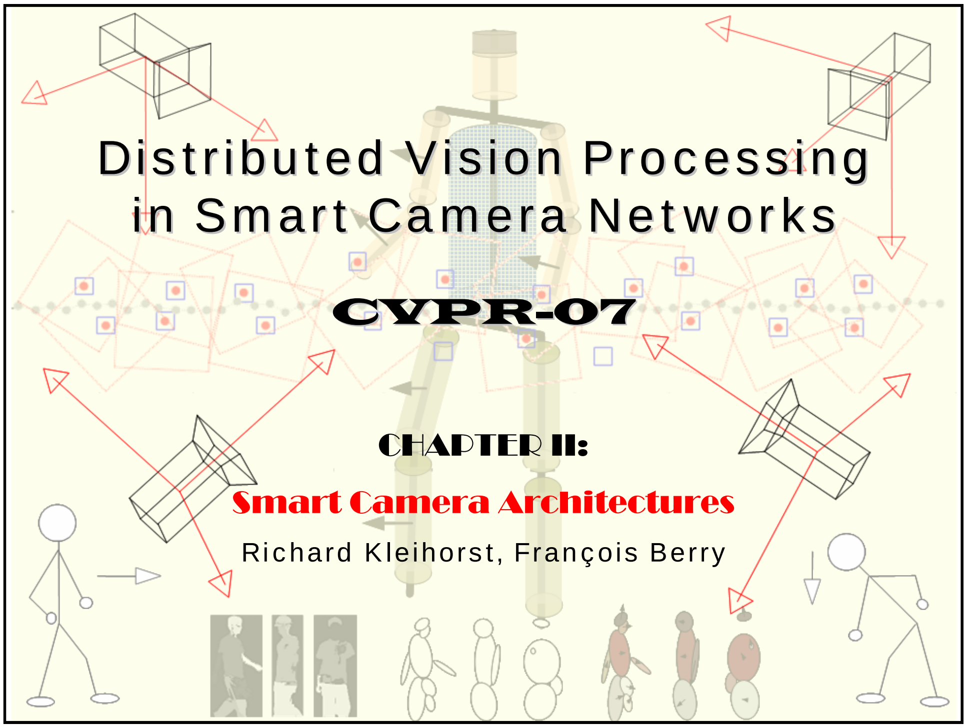
Hamid Aghajan, Stanford University, USA
François Berry, Univ. Blaise Pascal, France
Horst Bischof, TU Graz, Austria
Richard Kleihorst, NXP Research, Netherlands
Bernhard Rinner, Klagenfurt University, Austria
Wayne Wolf, Princeton University, USA

March 18, 2007
Minneapolis, USA

Course Website – http://wsnl.stanford.edu/cvpr07/index.php

Outline

I. Introduction
II. Smart Camera Architectures
    1. Wireless Smart Camera
    2. Smart Camera for Active Vision
III. Distributed Vision Algorithms
    1. Fusion Mechanisms
    2. Vision Network Algorithms
IV. Requirements and Case Studies
V. Outlook

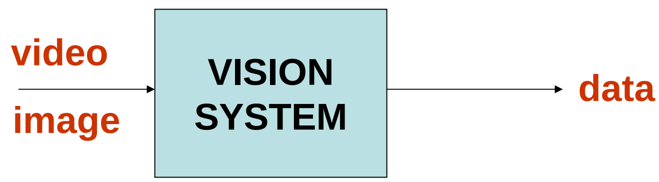


Wireless Camera (WiCa)

Contact: Richard Kleihorst
Richard.Kleihorst <AT> nxp.com
NXP Semiconductor Research

Vision Systems

- Are systems that analyze images and video
- They report in events/objects/properties
- DVD recorders, set-top boxes, smart cameras



"Smart Cameras"

- = Camera + intelligence
- = The basis for new applications
  - ➢Such as: detection, tracking, scene analysis

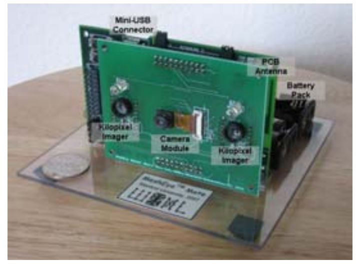

Automotive


Mobile Comm.


Surveillance


Consumer

Some Low-Cost Smart Cameras



CMUcam3 (ARM7)
60 MIPS @ 650mW

Stanford MesyEye
Mote (ARM7)

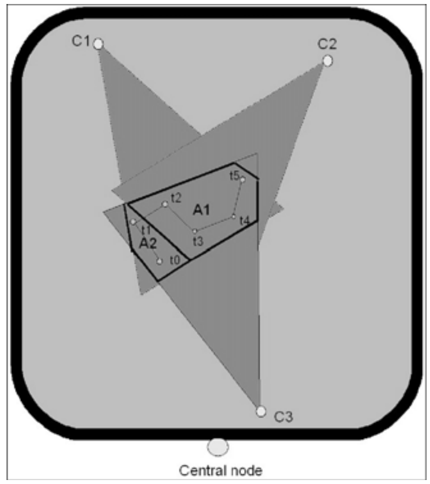
Smart Wireless Camera Platform

WiCa (Xetal SIMD)
50 GOPS @ 600mWatt

Cyclops (AVR RISC)
8 MIPS @ 50mWatt?

Distributed Vision Systems

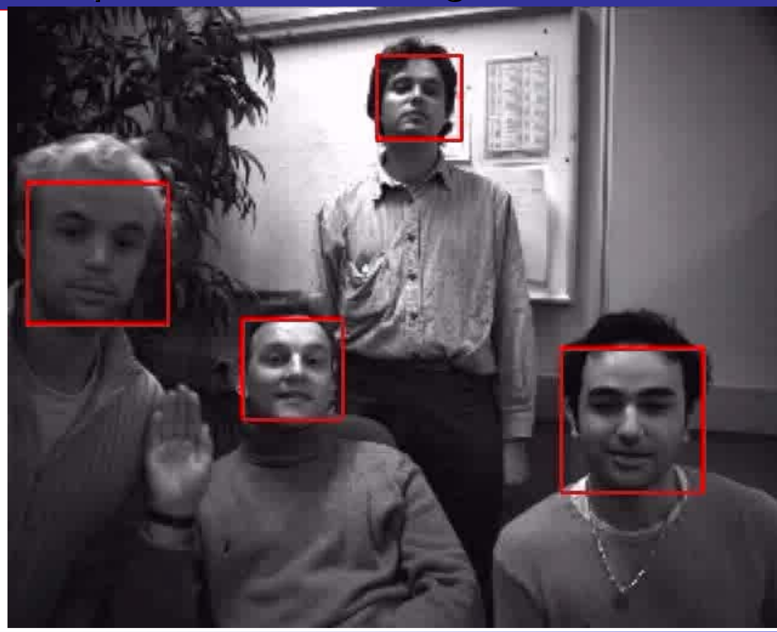
- Use **multiple cameras** to analyze the scene
- Less problems with occlusion
- Camera **networks**
- Distributed processing
- Distributed reasoning



Requirements in System Integration

- Performance
- Energy consumption
- Cost
- Architectural features (for active vision)

Example Event Casting: Face Detection



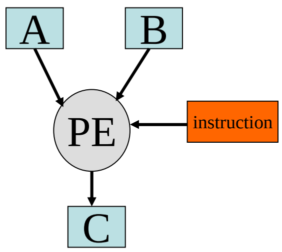
Face Detection Application Mapping

Pixel processing:	Image processing:	Application:
Haar filters	Image pyramid	Draw box, event
for every pixel similar	for every image similar	For every event different
SIMD **10++GOPS**	**FPGA/DSP** **100MOPS**	**CPU** **1MOPS**

SIMD → Single Instruction Multiple Data

Why SIMD for Low-Level?

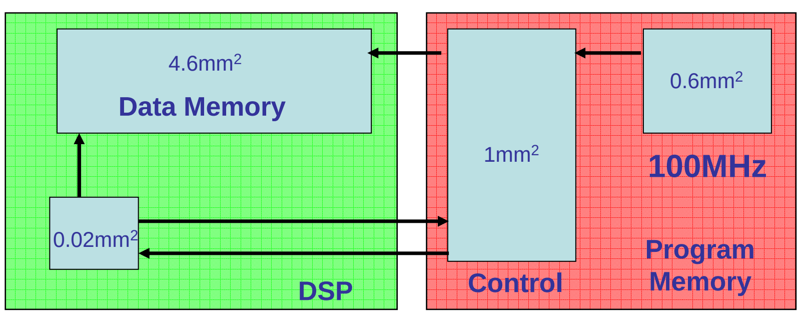
- High-performance (need > 10GOPS)
- High internal- bandwidth (need > 500Gb/s)



Bandwidth =

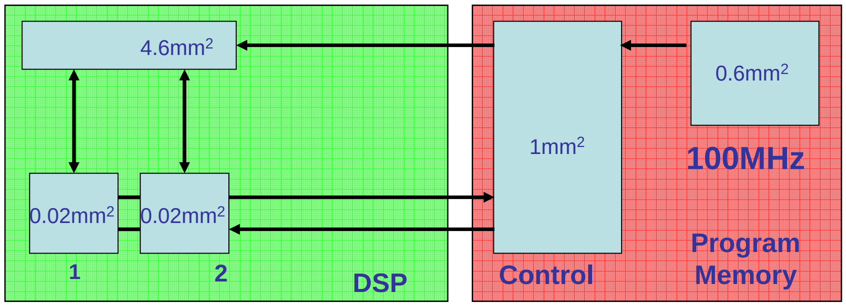
10GOPS * 3 * 16bits

Uniprocessor to SIMD: 1 PE



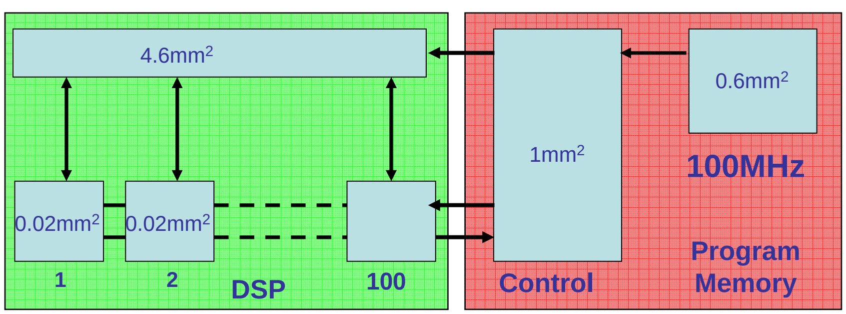
Performance	**100MOPS**
Size	$5.22mm^2$
Performance/area	$19MOPS/mm^2$
Overhead	26%
Bandwidth	**4.8Gb/s**

Uniprocessor to SIMD: 2PEs



Performance	**200MOPS**
Size	5.24mm^2
Performance/area	38MOPS/mm^2
Overhead	25%
Bandwidth	**9.6Gb/s**

Uniprocessor to SIMD: 100PEs



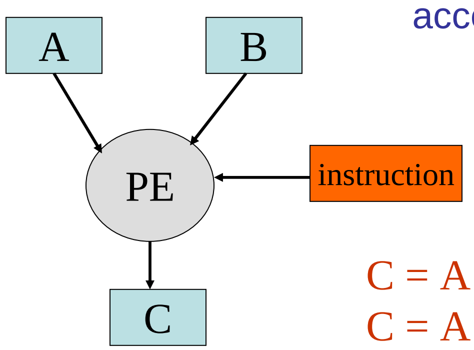
Performance	**10 GOPS**
Size	8.2 mm^2
Performance/area	1.2 GOPS/mm^2
Overhead	20%
Bandwidth	**480 Gb/sec.**

Uniprocessor to SIMD

	RISC : 1PE 50MHz	Xetal-II SIMD : 320PE@150MHz	Pentium4 2.4GHz
Peak Performance	0.05 GOPS	100 GOPS	6 GOPS
Size	6.4 mm^2	44.4 (0.18u) 11.1 (0.09u) mm^2	131 mm^2
Performance /area 0.18u	0.008 GOPS/mm^2	2.25 GOPS/mm^2	0.045 GOPS/mm^2
Overhead	26%	12%	??%
Bandwidth	2 Gb/S	1.5Tb/S	58 Gb/S
Peak Power Consumption		1.0 Watt	59 Watt

Why is SIMD Low-Power?

- Typical DSP instructions need 4 accesses to memory

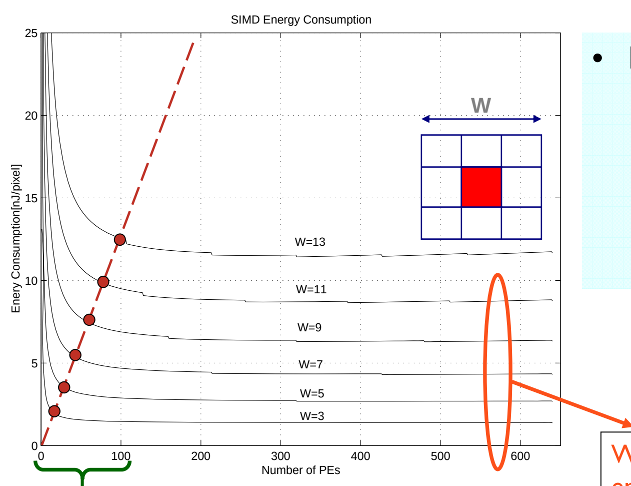


C = A + B;
C = A > B ? A : B;

Why is SIMD Low-Power?

- SIMD has multiple PEs in parallel
- Arithmetic always has to be done
- But: Instruction fetch is shared multiple times
- Data (A,B,C) access is shared in multiple-word-wide memories
- Accessing an 8 times wider memory takes half the amount of energy per data entity

SIMD Energy Consumption

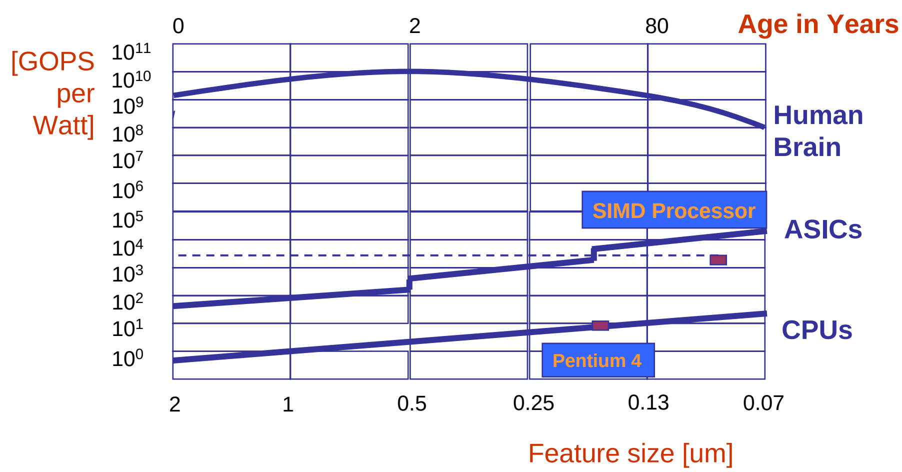


Parallelism → Memory Localization

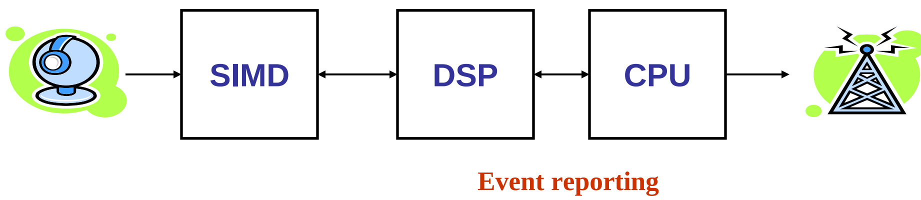
- Basis: Convolution
  - Computation
  - Communication
  - Memory access

Without voltage scaling, energy saving levels off

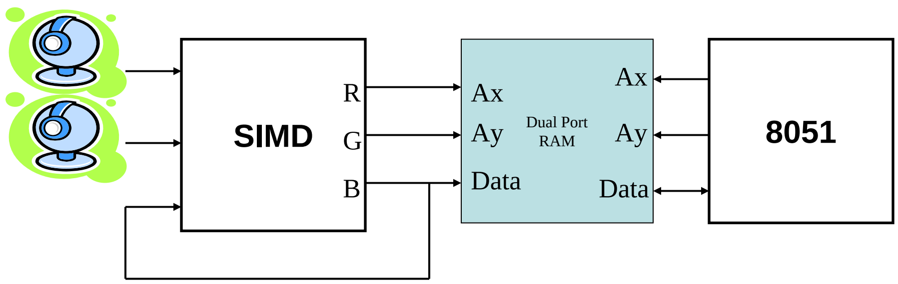
Computational Efficiency Growth (Moore)



Smart Wireless Camera Architecture



Connecting The Processors



Feedback loop:

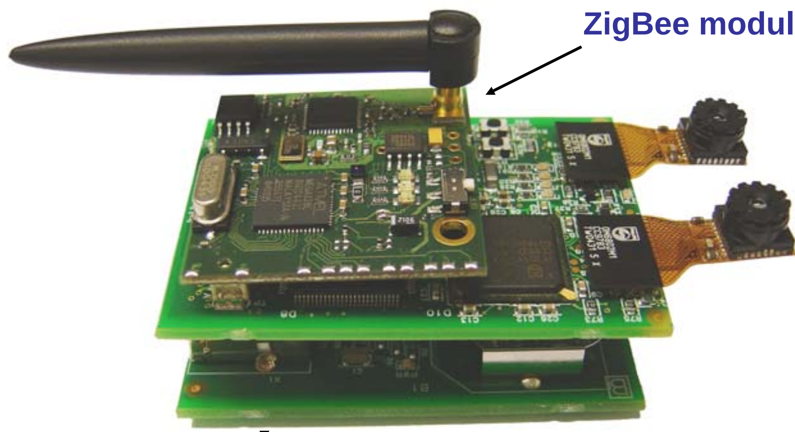
-Frame buffer, working with multiple images
-Look-Up-Table
-Image down-up sizing for pyramid approaches
-Image rescanning for lens-distortion correction

Smart Wireless Camera Platform

"WiCa"

- **IC3D/Xetal3** based
- Stereo sensor input
- 50GOPS performance
- Typical 100milli-Watts
- **ZigBee** node
- **Battery** powered
- C++ programmed

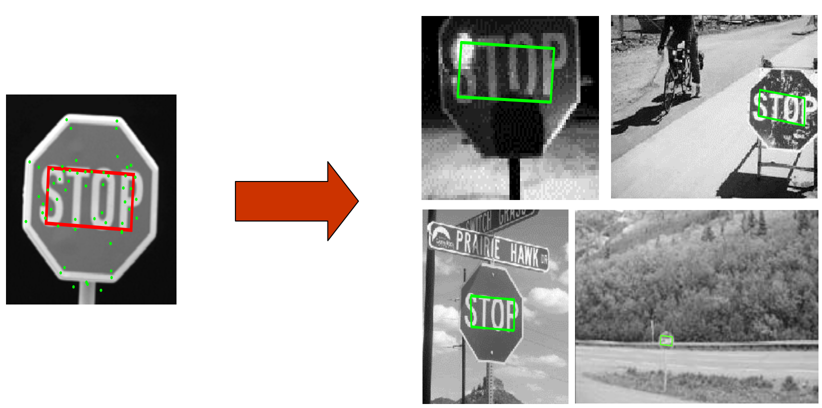
Smart Wireless Camera PCB



Ben Schueler, NXP

What Have We Mapped to WiCa?

Object recognition applications

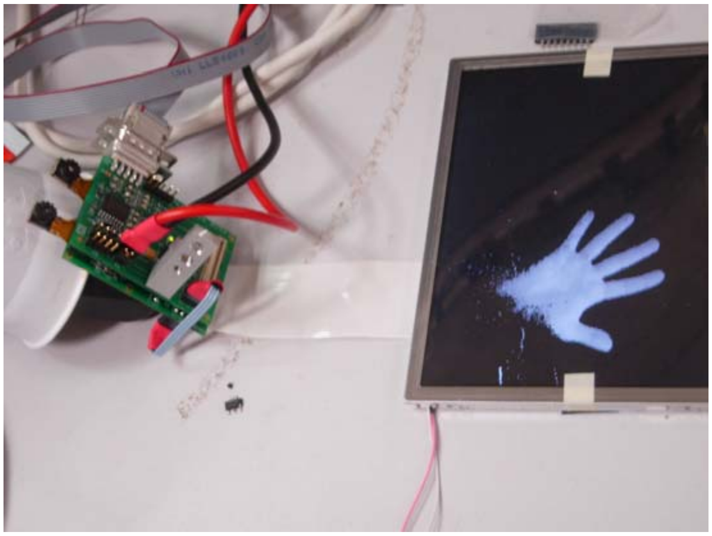


What Have We Mapped to WiCa?

Depth estimation from stereo

What Have We Mapped to WiCa?

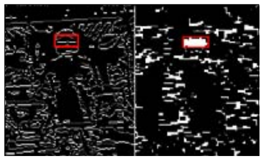
Gesture recognition



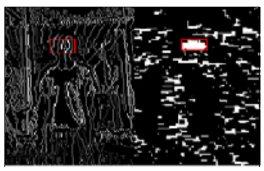
Alexander Danilin, NXP

What Have We Mapped to WiCa?

Face detection: soft edge features



Horizontal soft edges



Vertical soft edges

Which Algorithms Run Easily on WiCa?

- Where much of the application is running on the SIMD
- Where the DSP/CPU is used for limited or occasional tasks only
- Choose appropriate algorithmic basis for scene analysis
  - For example: “feature based”

Some Power Consumption Results

- Object recognition 25mWatt
- Face detection 40mWatt
- Stereo depth estimation 50mWatt
- Gesture recognition 15mWatt

Requirements in System Integration

- Performance
- Energy consumption
- Cost
- Architectural features (for active vision)

Outline

I. Introduction

II. Smart Camera Architectures

1. Wireless Smart Camera
2. Smart Camera for Active Vision

III. Distributed Vision Algorithms

1. Fusion Mechanisms
2. Vision Network Algorithms

IV. Requirements and Case Studies

V. Outlook