Mobile Visual Computing

Kari Pulli

Senior Director

NVIDIA Research

From Super Phones to Supercomputers

And then some...

Evolution of GPU Vision Computing

GPU Technology

Fixed Function Pipeline

Texture Combiners

Programmable Shaders

Single Precision Floating Point

Framebuffer Objects,

Render to Texture

CUDA Architecture

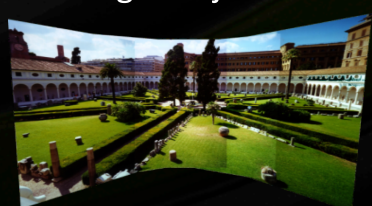
C- for CUDA

Double Precision

Vision Mappings

Image Derivatives, simple edge detection

Image Projections

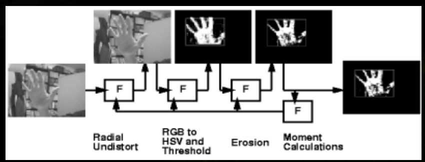


On-GPU Math

Feature Detection

Image Filtering,

Bayer Demosaicing



Imaging Pipelines

Fast "Gather" and (Global) Reduction Operations

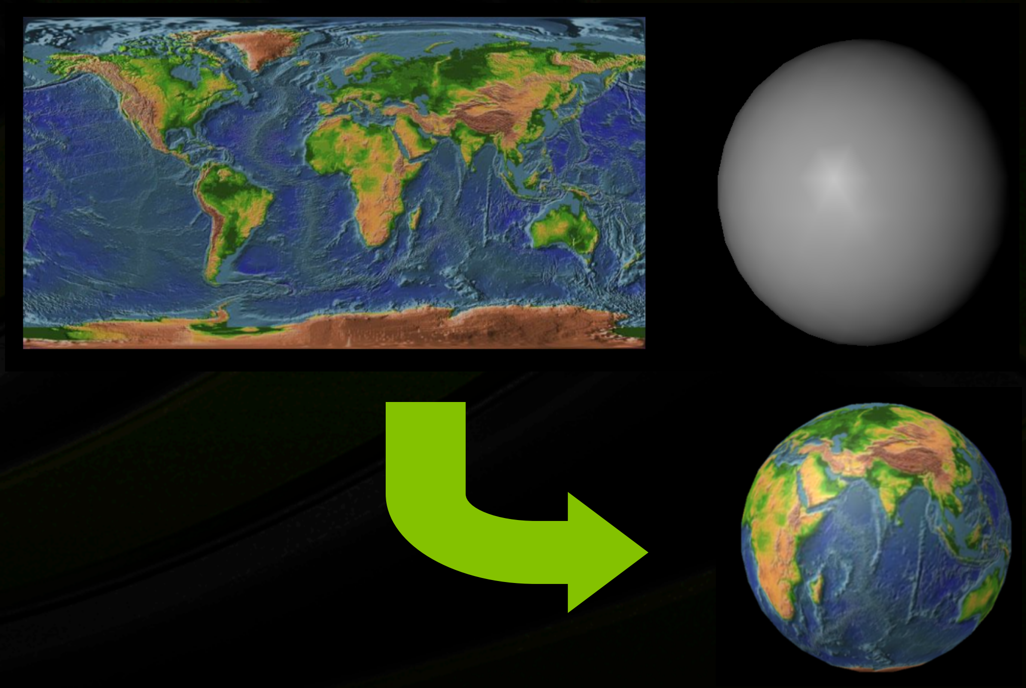
General Numerical Computing

Full "On-GPU" algorithms

CUDA for Image and Video Processing

Texture Hardware

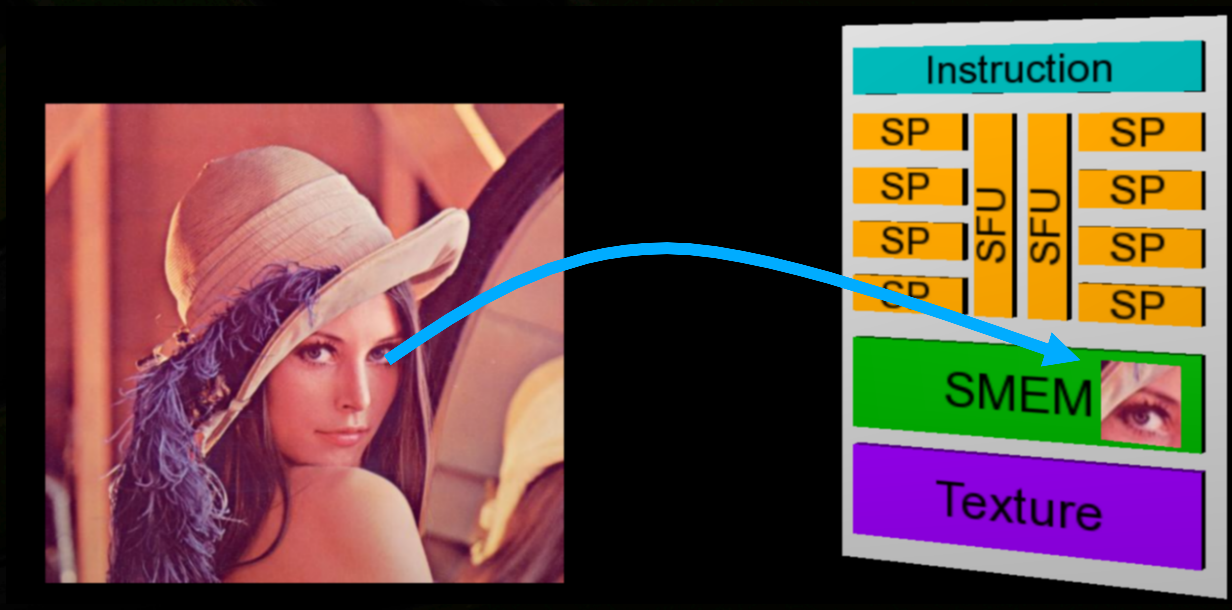
- A custom hardware block for image data access



CUDA for Image and Video Processing

Shared Memory (SMEM)

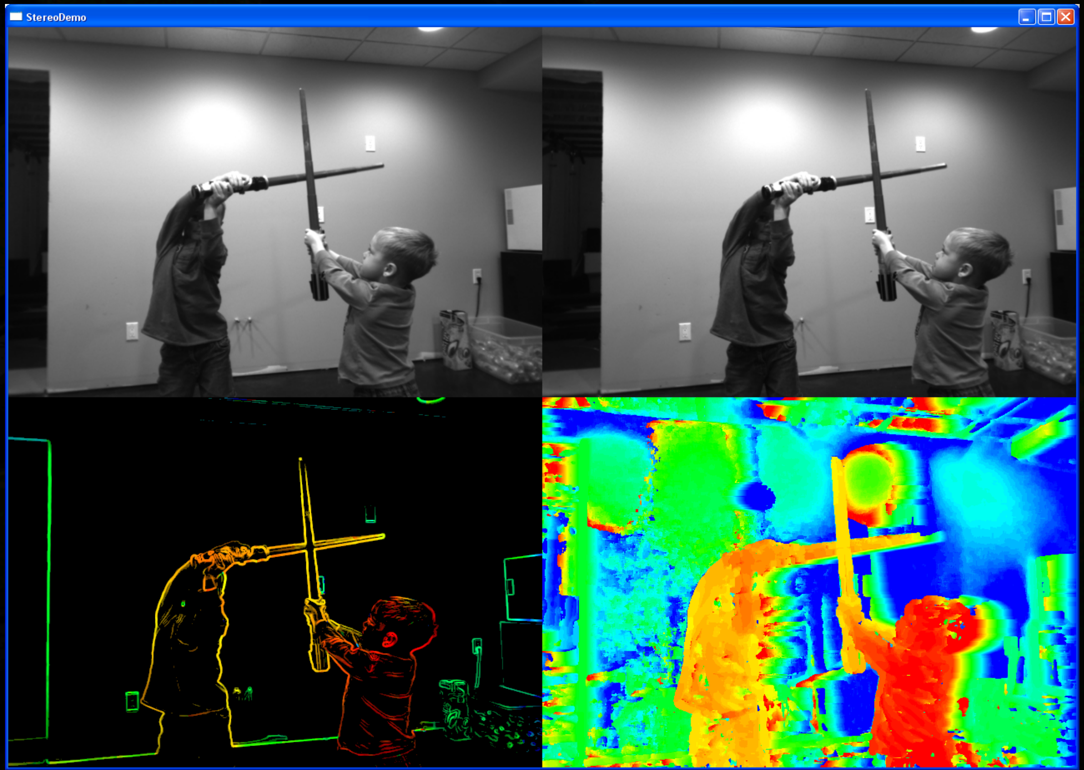
- Ultra-high speed "on-chip" memory
- Load an image tile into SMEM and share pixels between threads



CUDA for Image and Video Processing

Visualization: Graphics Interoperation

- CV frequently needs interactive display
- CUDA Interop with graphics allows compute and display in the same place, without data transfer overhead

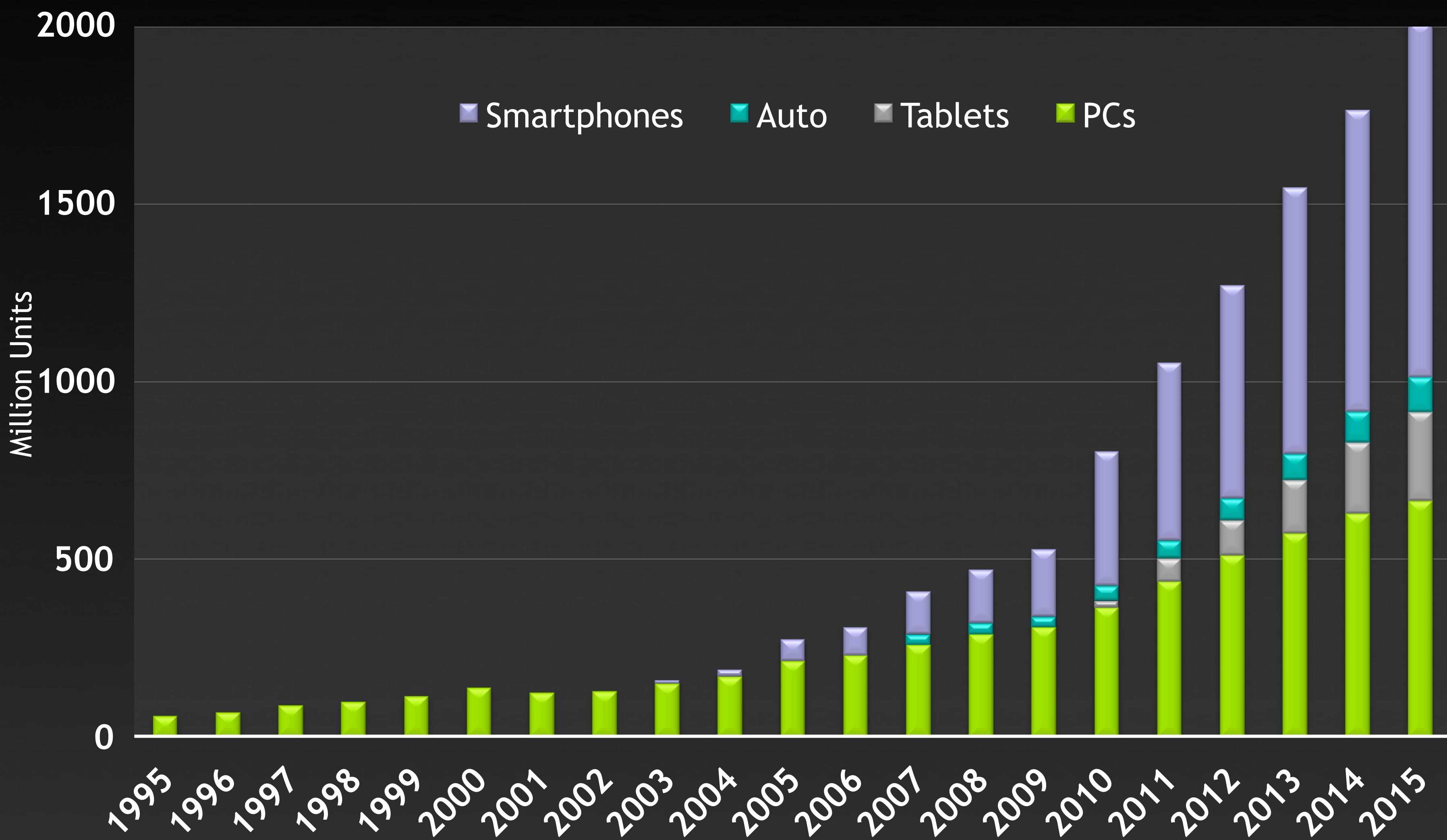


NPP: Image Processing Primitives

- **Library of high performance image processing primitives**
- **http://www.nvidia.com/nvpp**

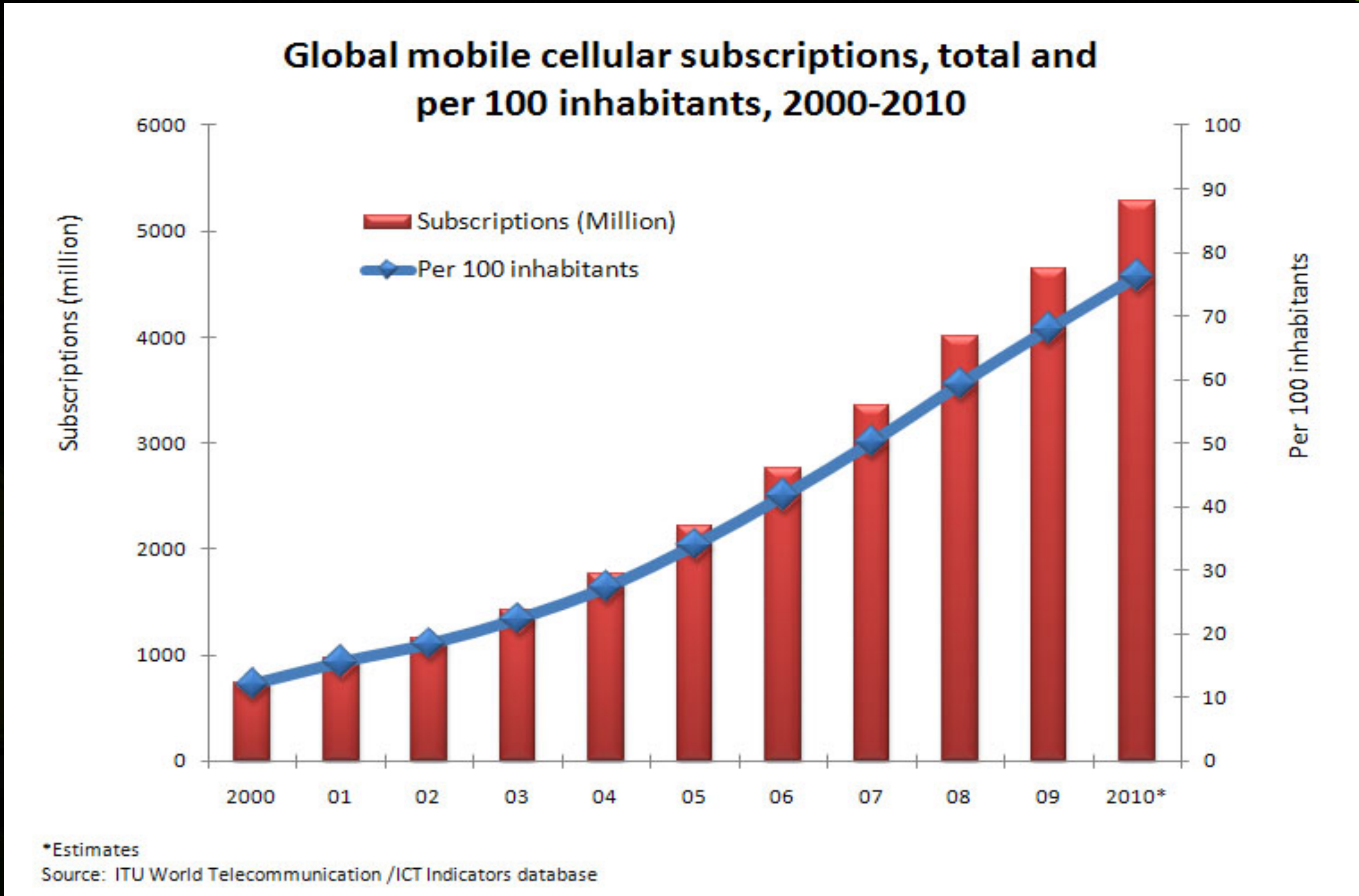
- **Data exchange & initialization**
  - Set, Convert, CopyConstBorder, Copy, Transpose, SwapChannels
- **Arithmetic & Logical Ops**
  - Add, Sub, Mul, Div, AbsDiff
- **Threshold & Compare Ops**
  - Threshold, Compare
- **Color Conversion**
  - RGB To YCbCr (& vice versa), ColorTwist, LUT_Linear
- **JPEG**
  - DCTQuantInv/Fwd, QuantizationTable
- **Filter Functions**
  - FilterBox, Row, Column, Max, Min, Median, Dilate, Erode, SumWindowColumn/Row
- **Geometry Transforms**
  - Mirror, WarpAffine / Back/ Quad, **WarpPerspective** / Back / Quad, Resize
- **Statistics**
  - **Mean, StdDev, NormDiff, MinMax,** Histogram, SqrIntegral, RectStdDev
- **Computer Vision**
  - **ApplyHaarClassifier, Canny**

Rising demand for GPUs



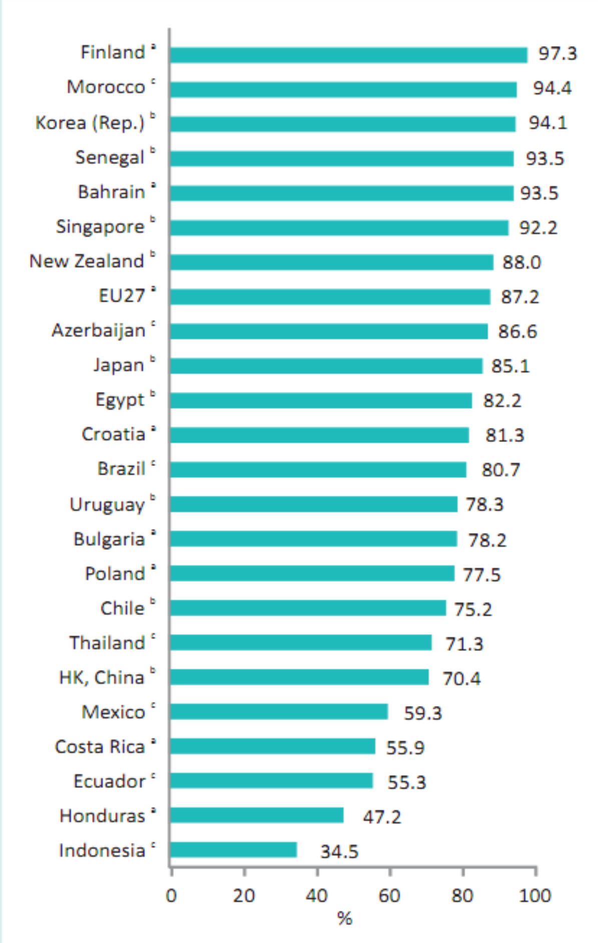
Source: IDC, Gartner, Morgan Stanley

TEGRA™
The Processor for Your Most Personal Computer


Global mobile cellular subscriptions, total and per 100 inhabitants, 2000-2010
Subscriptions (Million)
Per 100 inhabitants
Subscriptions (million)
Per 100 inhabitants
6000
5000
4000
3000
2000
1000
0
100
90
80
70
60
50
40
30
20
10
0
2000
01
02
03
04
05
06
07
08
09
2010*
*Estimates
Source: ITU World Telecommunication /ICT Indicators database

NVIDIA

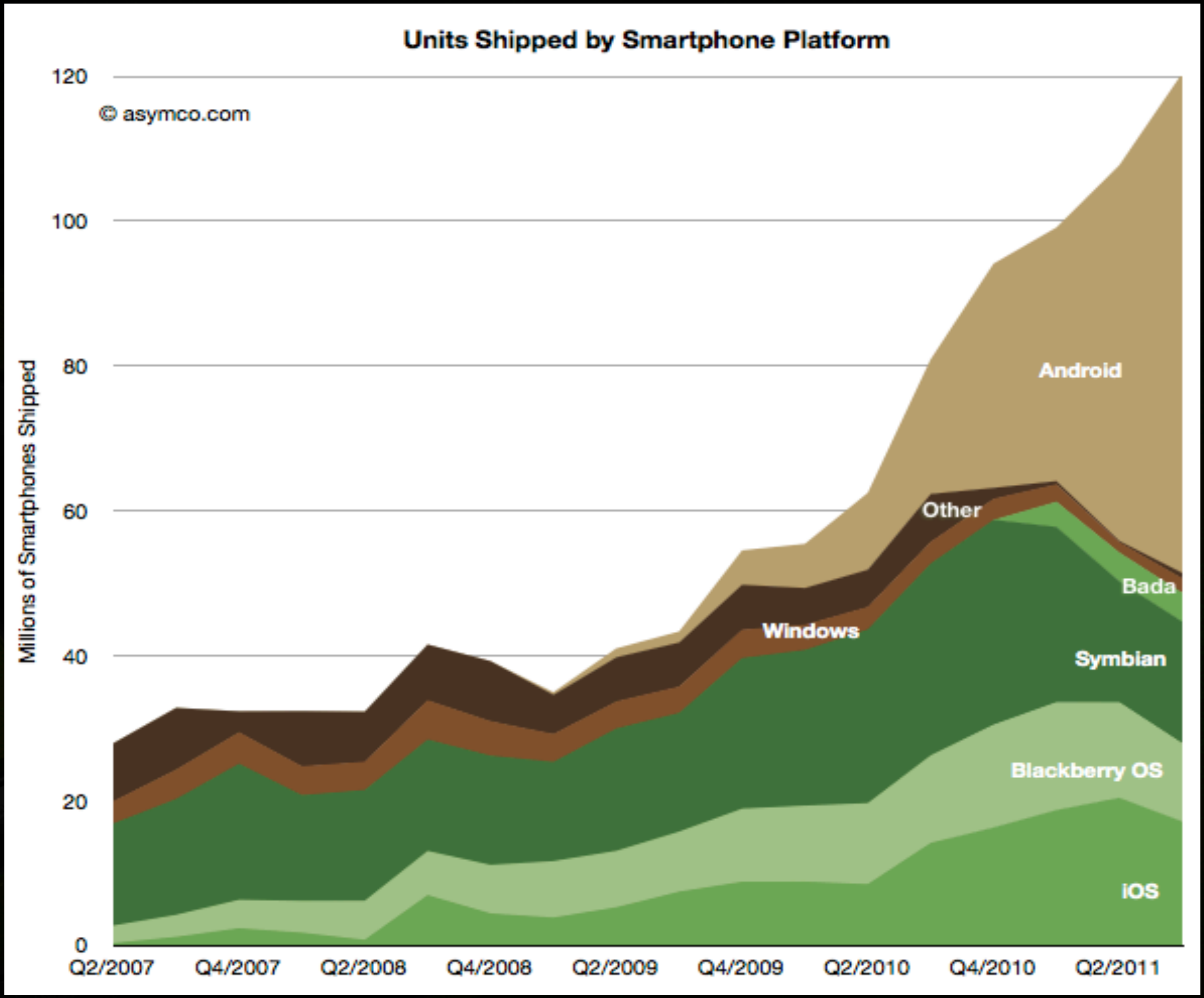
Chart 5.3: Percentage of individuals aged 15 to 74* using a mobile phone, latest available year



USA: 83% of adults

http://pewinternet.org/Reports/2011/Cell-Phones.aspx

source: ITU



Units Shipped by Smartphone Platform
© asymco.com
Millions of Smartphones Shipped
120
100
80
60
40
20
0
Q2/2007
Q4/2007
Q2/2008
Q4/2008
Q2/2009
Q4/2009
Q2/2010
Q4/2010
Q2/2011
Android
Other
Windows
Bada
Symbian
Blackberry OS
iOS

Trends in camera phone sales

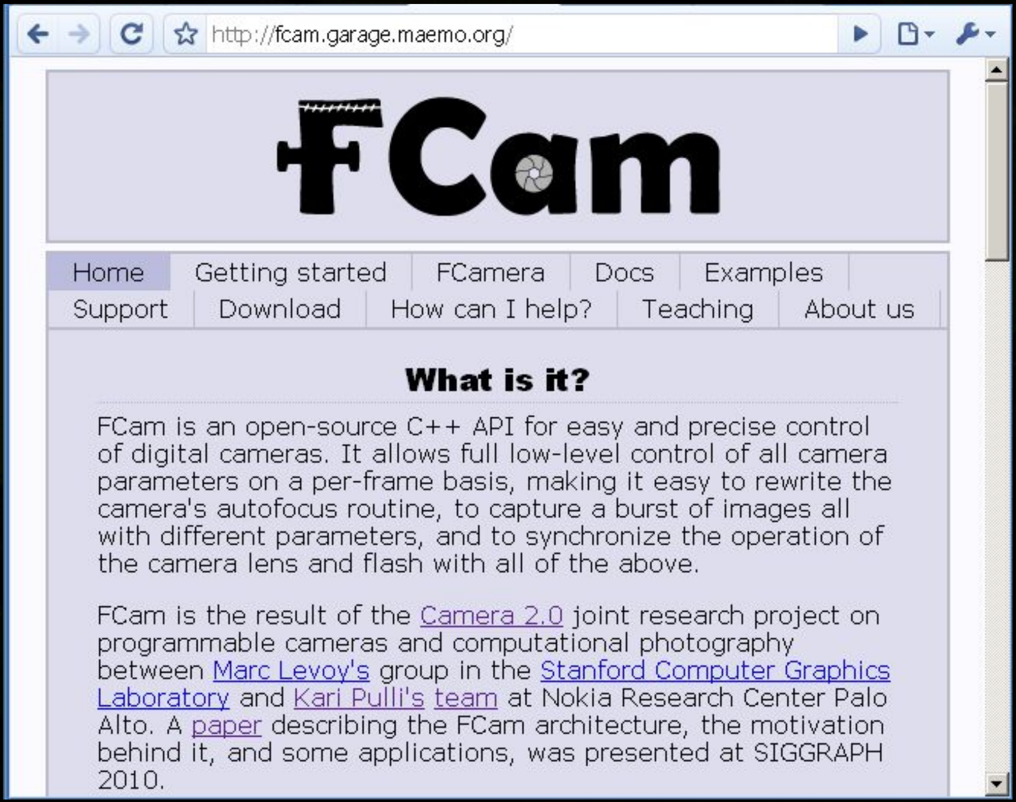
- Growing sales

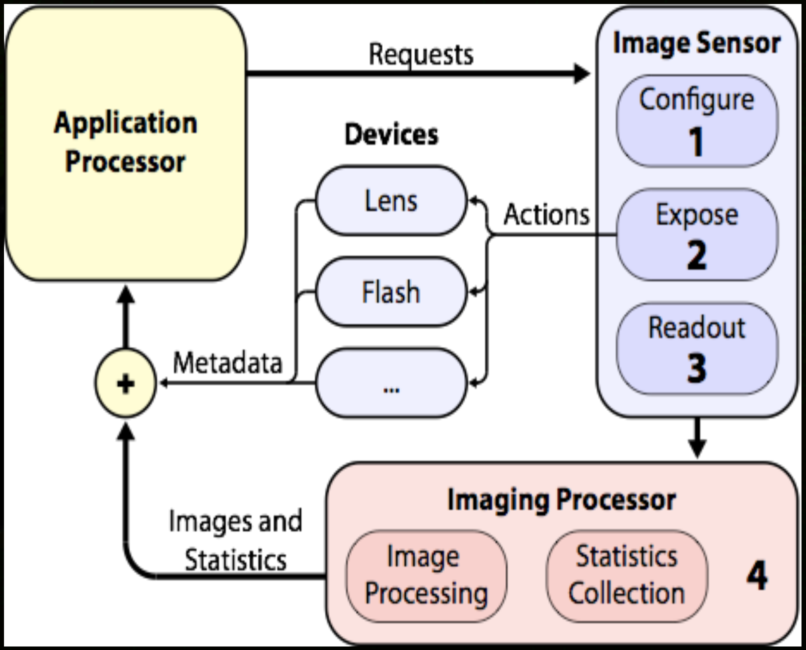
2002	16 million	3% of the phones
2003	85 million	16% of the phones
2010	805 million	65% of the phones
2011	1.1 billion	71% of the phones
2014	1.3 billion	85% of the phones

- Average resolution grows too
  - 2008 1 MP
  - 2009 2 MP
    - 2010 5+MP: > 100 million
    - 2011 5+MP: ~ 360 million
  - 2014 5 MP
    - 2014 5+MP: > 550 million

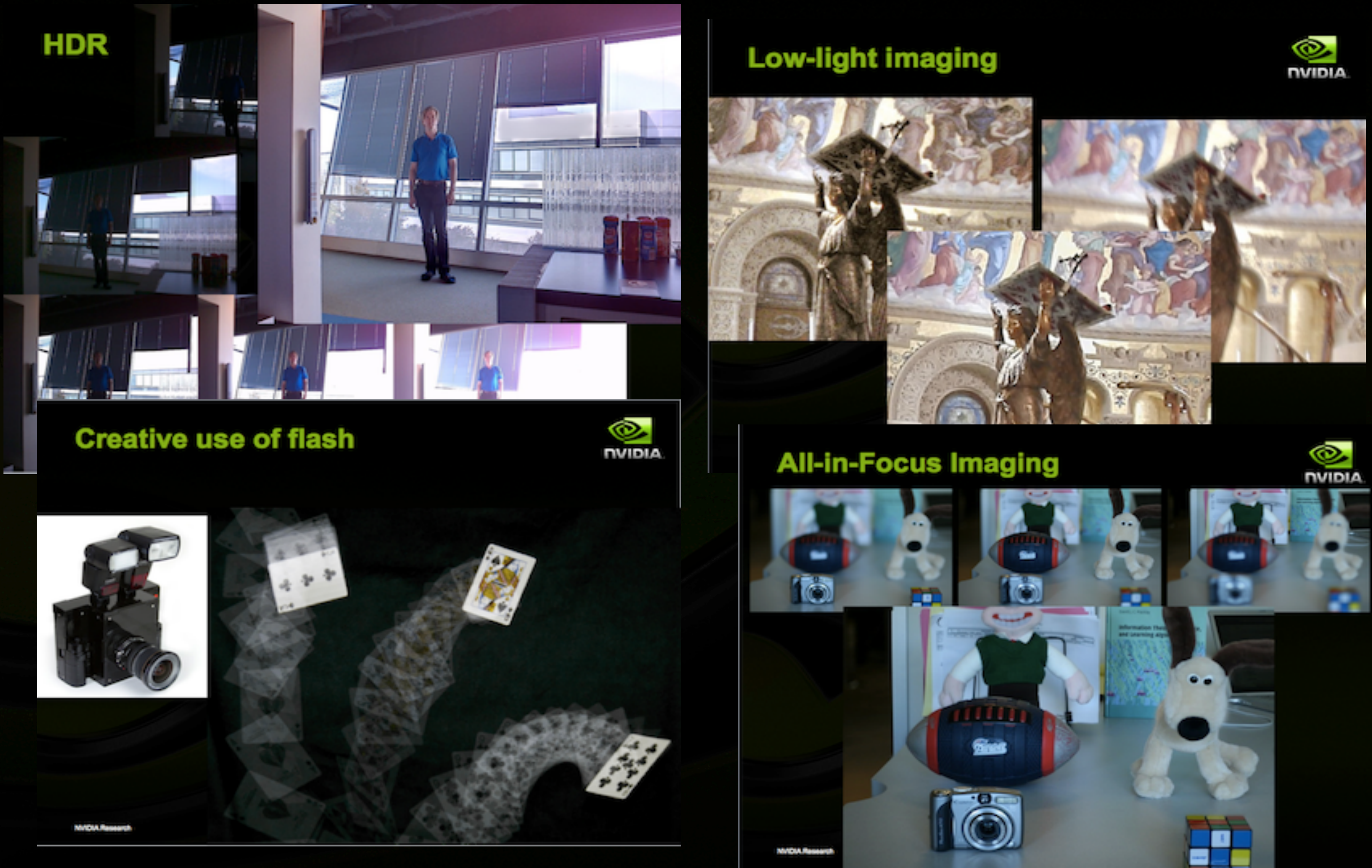
APIs: FCam

- **Open Source API for camera control for computational photography**
  - **Nokia Linux phones**
  - **NVIDIA Tegra 3 dev boards**



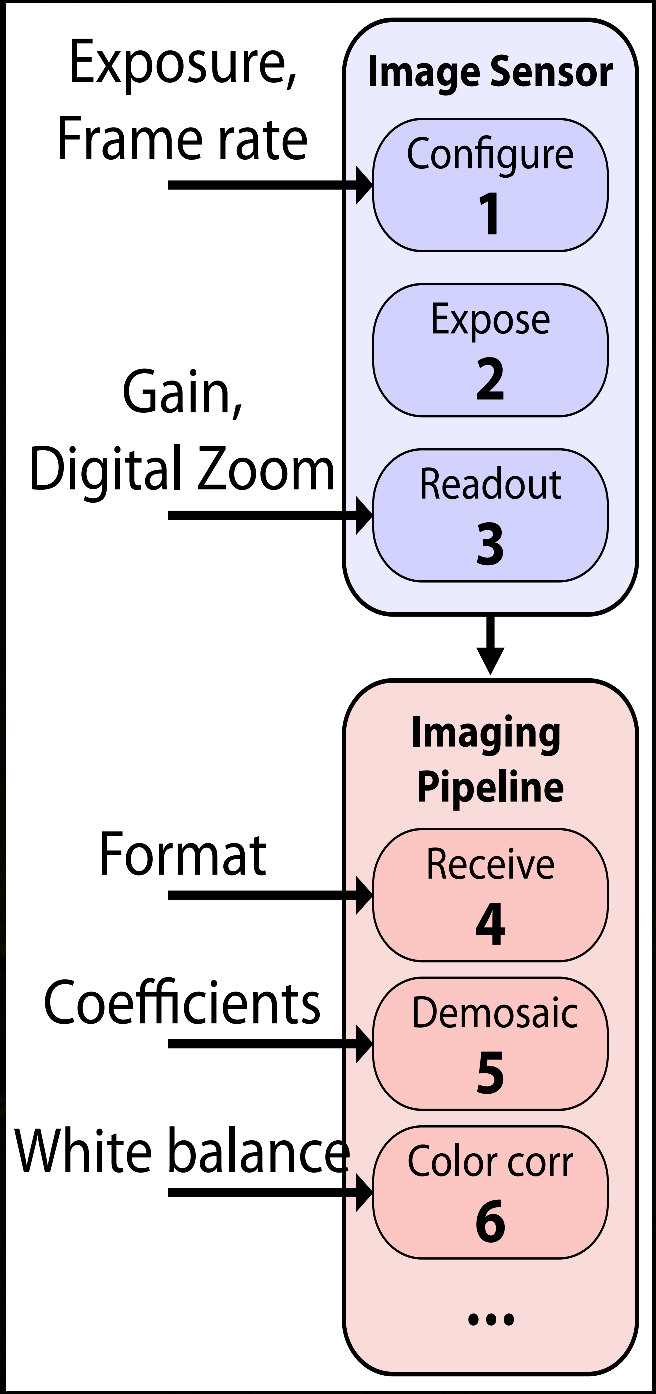


Applications



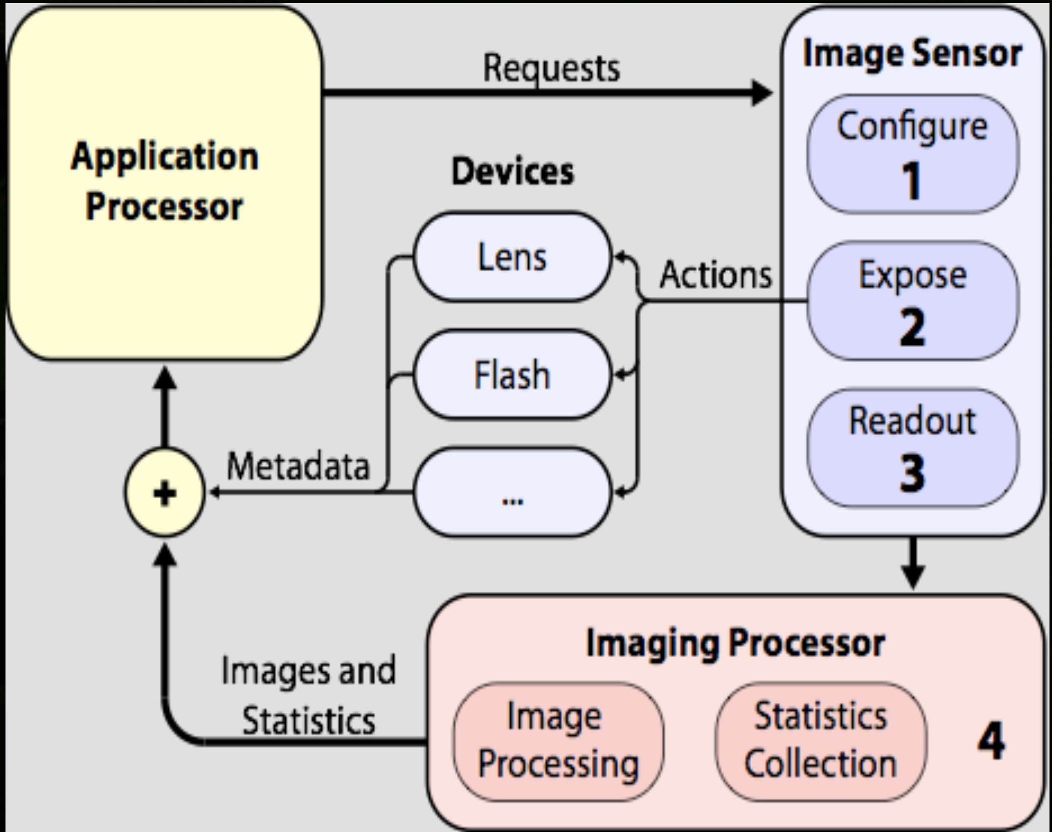
Traditional sensor model does not work for Comp. Photogr.

- **Real image sensors are pipelined**
  - while one frame exposing
  - next one is being prepared
  - previous one is being read out
- **Viewfinding / video mode:**
  - pipelined, high frame rate
  - settings changes take effect sometime later
- **Still capture mode:**
  - need to know which parameters were used
  - → reset pipeline between shots → slow



The FCam Architecture

- No global state
  - instead, state travels with image requests
  - every stage in pipeline may have different state
  - → allows deterministic, fast state changes
- Synchronize devices
  - flash
  - lens
  - capture sound
  - gyro
  - …



FCam / Tegra Course at Stanford

graphics.stanford.edu/courses/cs478/

CS 478 - Computational photography

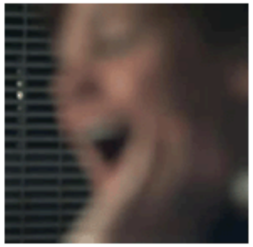
Winter, 2012

A cutaway view showing some of the optical and electronic components in the Canon 5D, a modern single lens reflex (SLR) camera. In the first part of this course, we'll take a trip down the capture and image processing pipelines of a typical digital camera.

This is the Stanford Frankencamera, an experimental open-source camera we are building in our laboratory. It's bigger, heavier, and uglier than the Canon camera, but it runs Linux, and its metering, focusing, demosaicing, denoising, white balancing, and other post-processing algorithms are programmable. We'll eventually be distributing this camera to researchers worldwide.

This is the Nokia N900, the first in a new generation of Linux-based cell phones. It has a 5-megapixel camera and a focusable Carl Zeiss lens. More importantly, it runs the same software as our Frankencamera, so it's programmable right down to its autofocusing algorithm.

This is a prototype Nvidia tablet featuring the Tegra 3 processor. It has stereo back-facing cameras, Android OS, and a ported implementation of our FCam API. Each student will receive a tablet for the duration of the course, to try his hands at mobile computational photography.



In the second part of the course, we'll consider problems in photography and how they can be solved computationally. One such problem is misfocus. By inserting a microlens array into a camera, one can record light fields. This permits a snapshot to be refocused after capture.

Most digital cameras capture movies as well as stills, but handshake is a big problem, as exemplified by the home video above. Fortunately, stabilization algorithms are getting very good; look at this experimental result. We'll survey the state-of-the-art in this evolving area.

Quarter
Winter, 2012
Units
3-4 (same workload) (+/NC or letter grade)
Time
Mon/Wed 2:30 - 3:45
Place
392 Gates Hall (graphics lab conference room)
Course URL
cs478.stanford.edu
Discussion
CS478 @Piazza

Instructors
Jongmin Baek, Dave Jacobs, Kari Pulli (Guest Lecturer)
Office hours
Wed 3:45 - 5:00, Thurs 2:30 - 3:45, Gates 360
Prerequisite
An introductory course in graphics or vision, or CS 178; good programming skills
Televised?
No

Dark shot

Bright shot

Naively fused image

Initial warp of bright over dark

Filling some holes…

Final warped

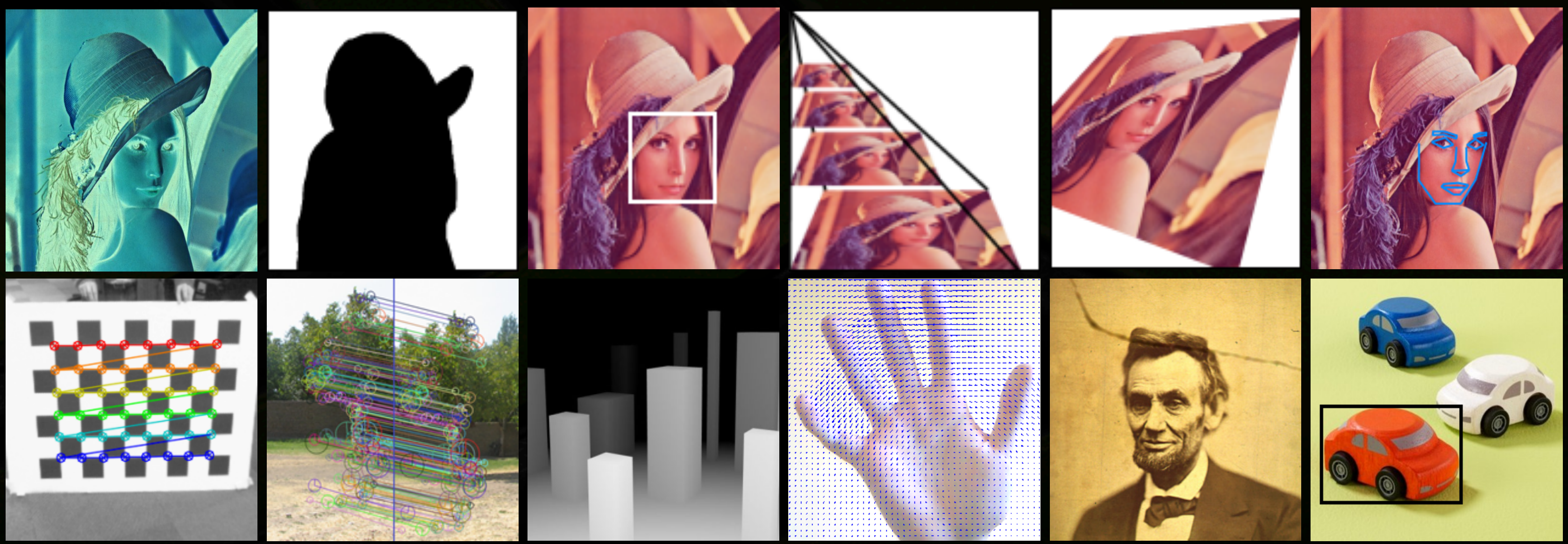
Dark shot

Fused image

Naively fused image

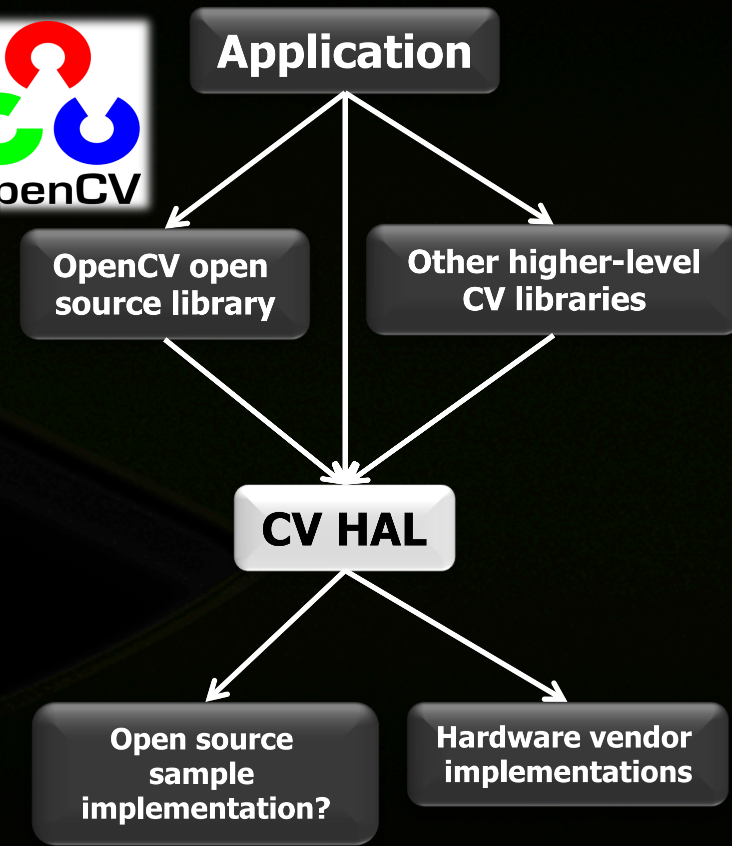
APIs: OpenCV

- For the compute part of computer vision and computational photography
  - a de-facto standard
  - optimized for both CUDA and Tegra



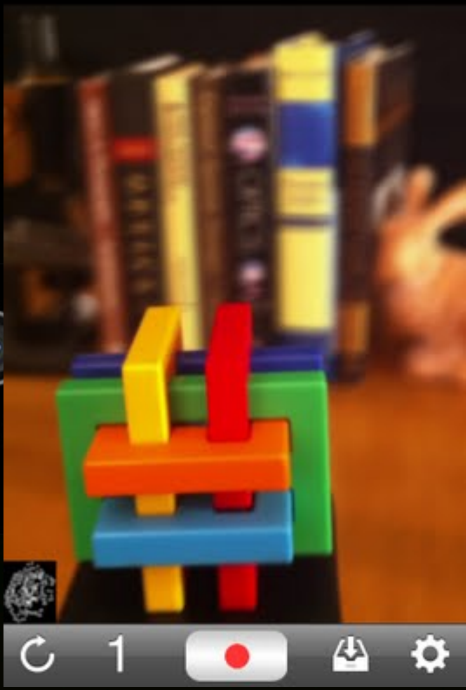
APIs: Khronos Vision Working Group

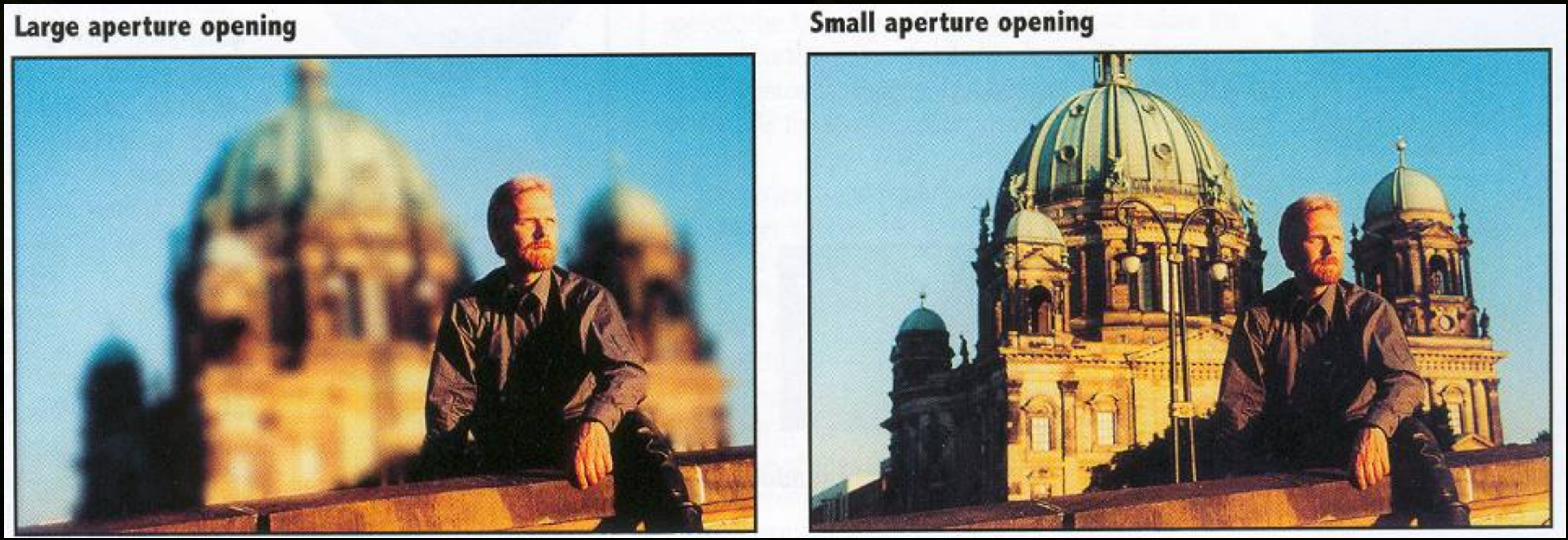
- Vision HW Acceleration Layer
  - enable hardware vendors to implement accelerated imaging and vision algorithms
- Can be used by high-level libraries or applications directly
  - primary focus on enabling mobile and embedded systems



Camera apps on smartphones

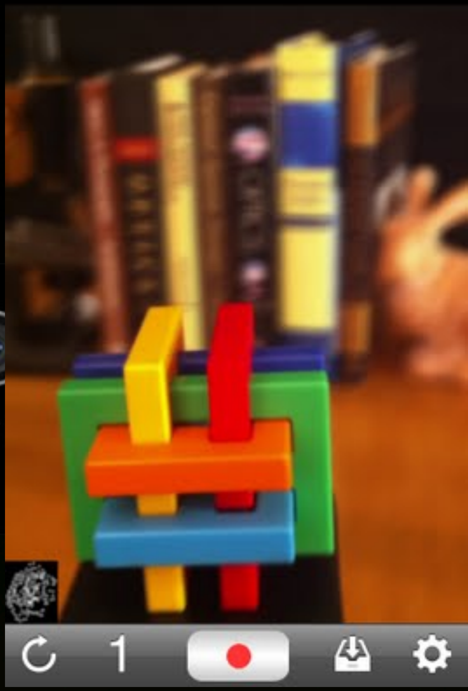
- **Synthcam**





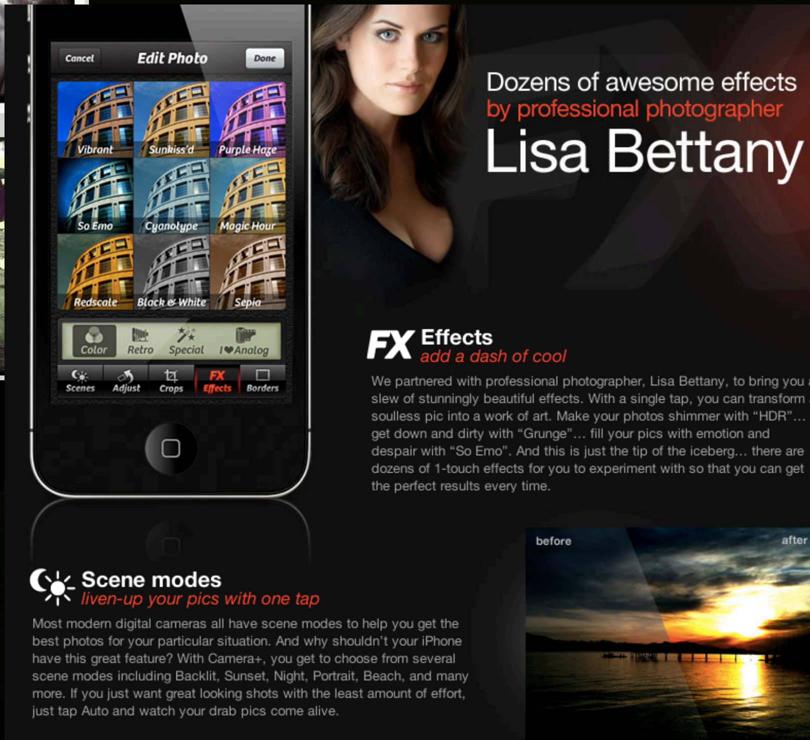
Camera apps on smartphones

- Synthcam



- Hipstamatic

- Camera+



http://taptaptap.com/blog/cameraplus-reaches-6-million-sales-milestone/

… but to make some real money …

Instagram Quickly Passes 1 Million Users

By NICK BILTON | December 21, 2010, 9:0

Facebook Buys Instagram For $1 Billion, Turns Budding Rival Into Its Standalone Photo App

JOSH CONSTINE AND KIM-MAI CUTLER

Monday, April 9th, 2012

Photos collected from Instagram's Popular page

Insta

facebook

Scalado Rewind

Polaroid on Android

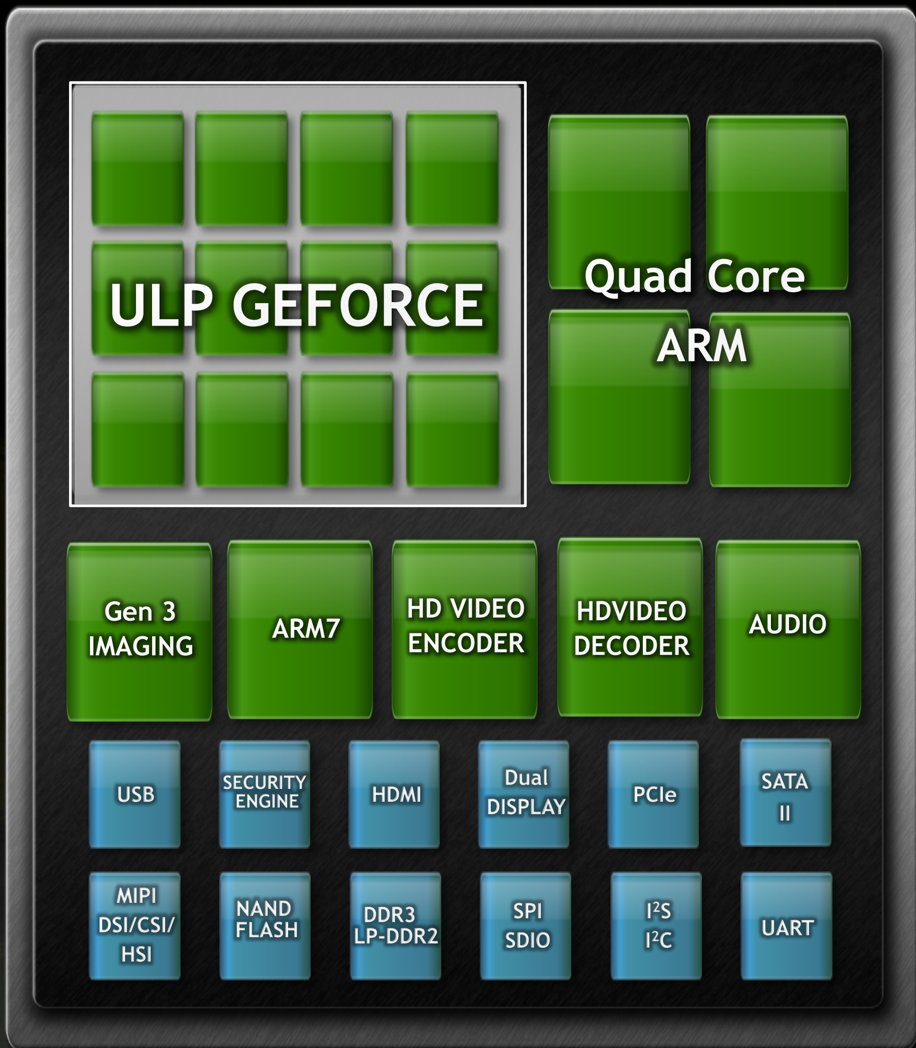
Performance on a budget

- Need to save power
  - you want the batteries to last
- But occasionally you want a lot of performance
- What do you do?

SOC = System-On-Chip

- Put a full computer on the same chip
- Saves power
  - off-chip communication is expensive

Processing: Tegra 3 SOC

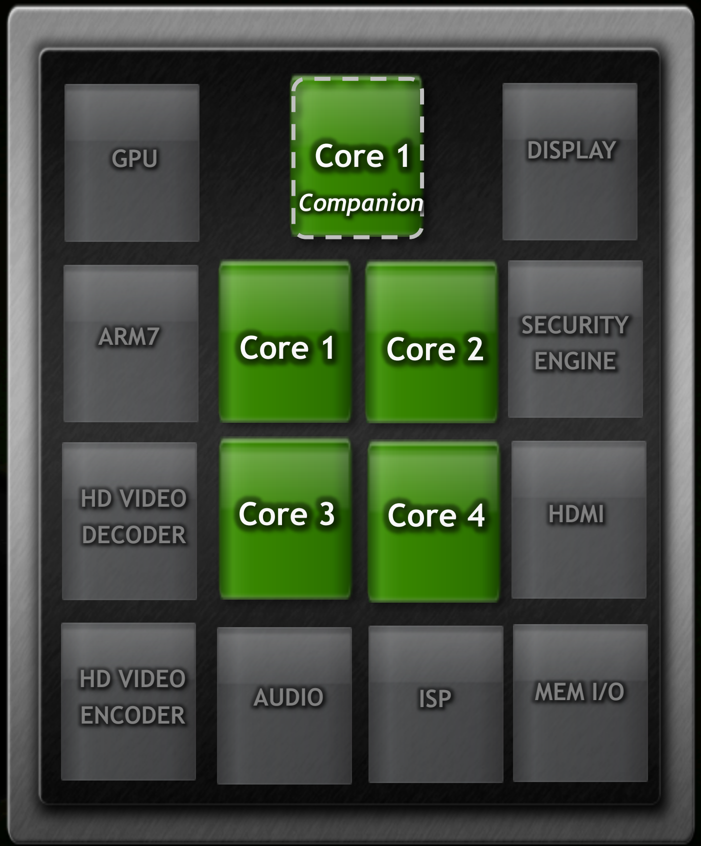


CPU	3X Performance *Quad Core NEON*
POWER	20x Lower Power *Due to ULP mode*
VIDEO	4X Complexity *1080i/p High Profile*
GRAPHICS	3X Performance *12 Core, Dual Pixel Pipe*
MEMORY	3X bandwidth *DDR3L up to 1600 data rate*
IMAGING	Better noise reduction & color rendition *Two simultaneous streams*
AUDIO	HD Audio *7.1 channel surround*
STORAGE	2 - 6X faster *e.MMC 4.4 and SATA-II*

4 + 1

- An extra core
  - “companion core”
  - “shadow core”
  - built with low-leakage process
    - lower peak performance
- All cores identical
  - the same code runs on all cores
  - application programmer doesn’t have to worry
    - power is adapted automagically

5 CPU Cores



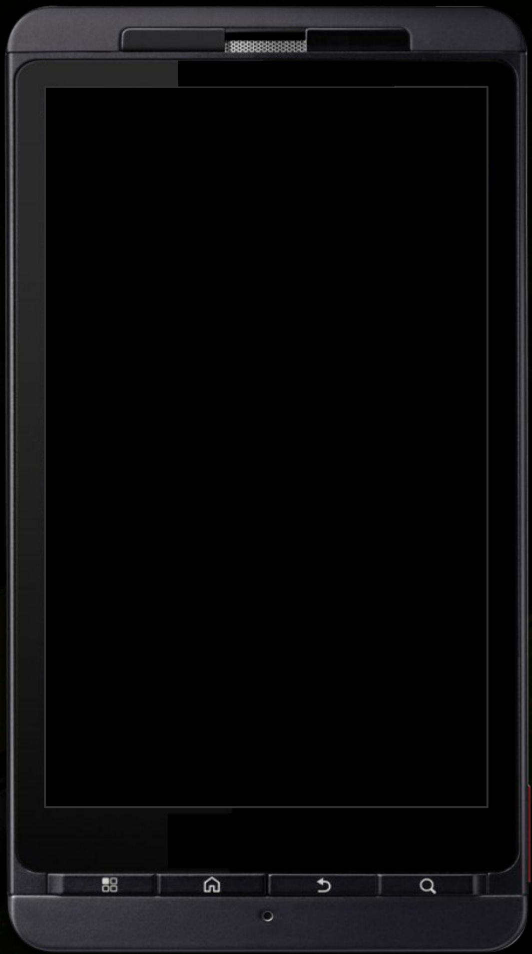
Most Common Use Cases

~80%

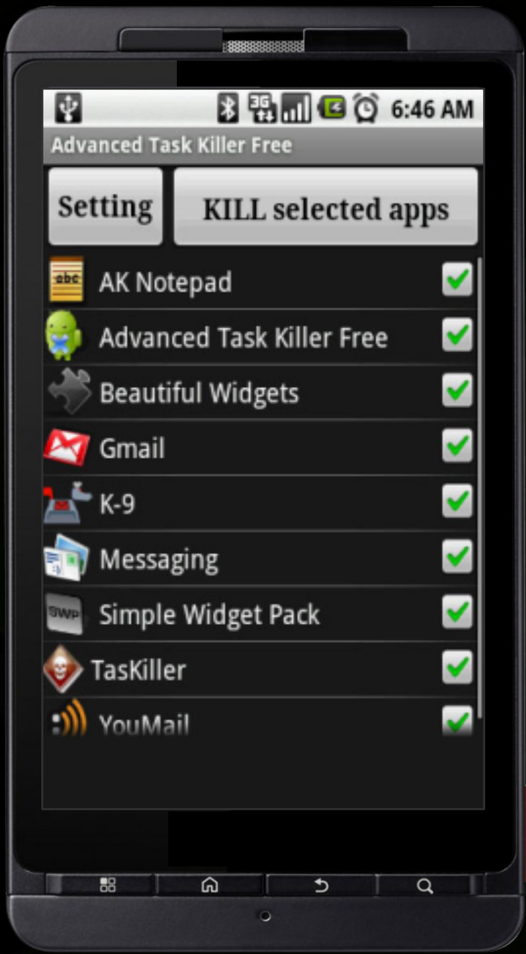
Gaming, Web, Email, News

~20%

Active Standby



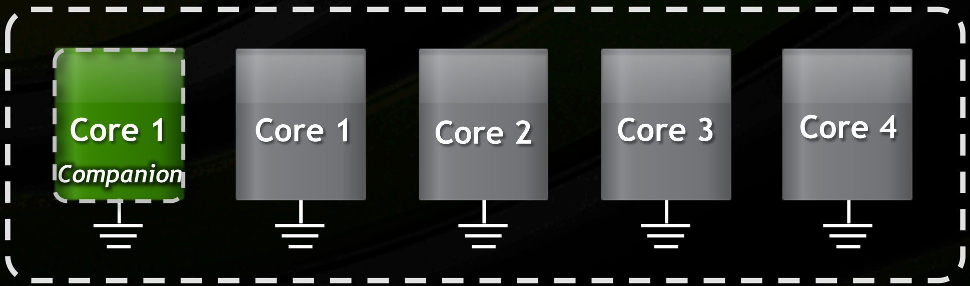
What's really running

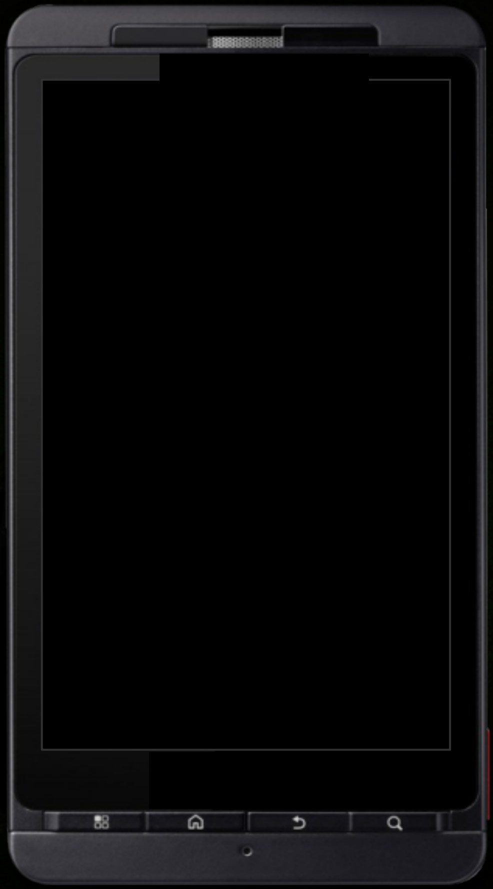


Variable SMP

- ✓ "Companion" Core Active
- ✓ 0 to 500 MHz
- ✓ 0 to 200 mW

ARM CPU Cores



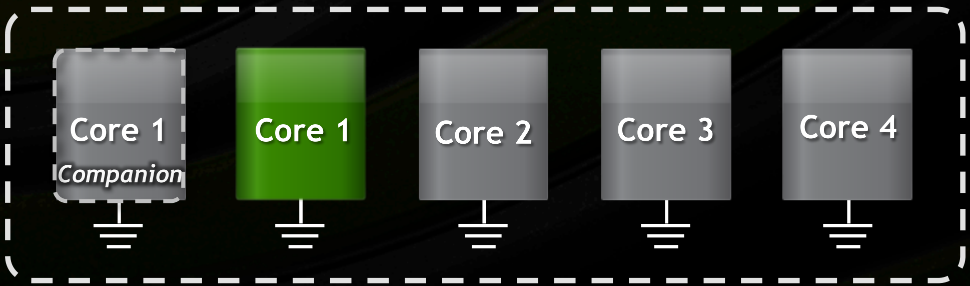


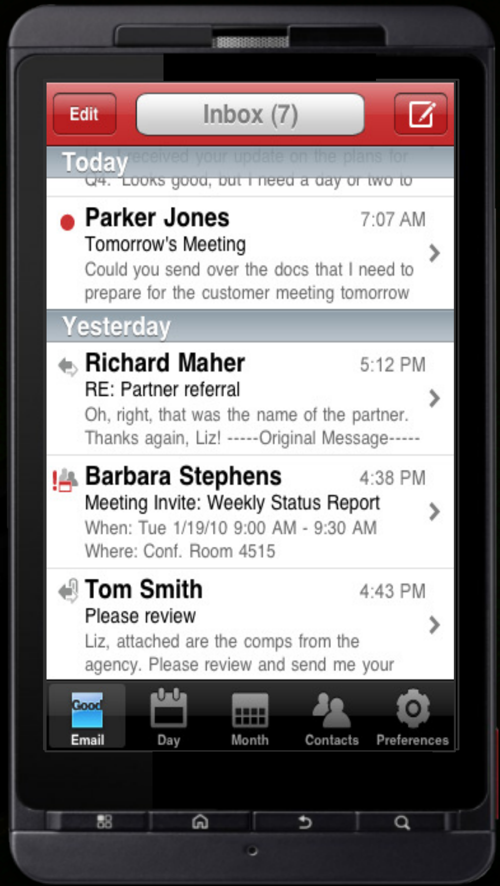
Active Standby, Music, Video

Variable SMP

- ✓ 1 CPU Core Active
- ✓ 0 to 1.7 GHz
- ✓ 30 to 400 mW

ARM CPU Cores



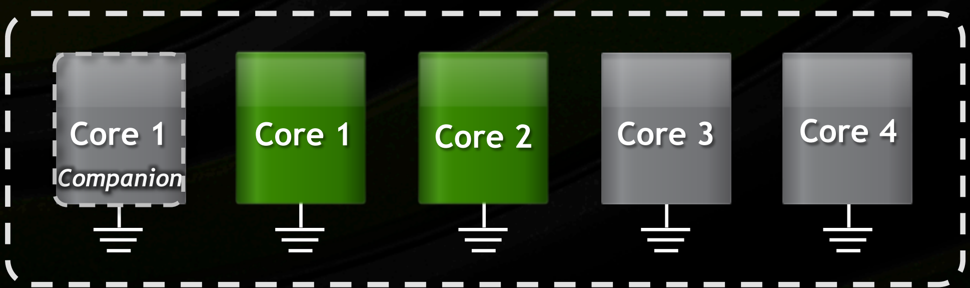


Email

Variable SMP

- ✓ 2 CPU Cores Active
- ✓ 0 to 1.6 GHz
- ✓ 50 to 800 mW

ARM CPU Cores



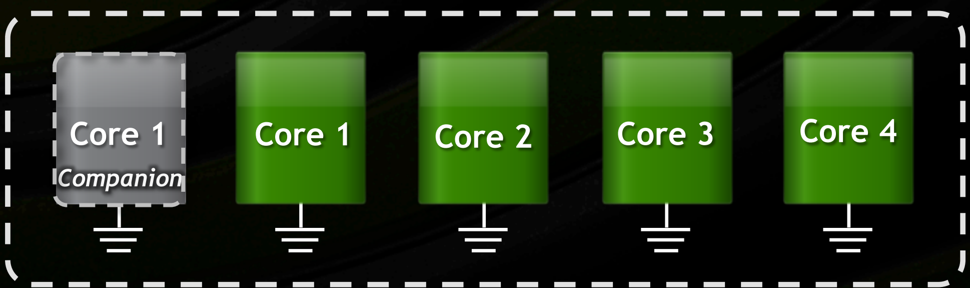
Web

Variable SMP

- ✓ 4 CPU Cores Active
- ✓ 0 to 1.6 GHz
- ✓ 100 to 1600 mW

Gaming, Multimedia Apps

ARM CPU Cores



What is the power budget?

- Thermal design point
  - the maximum amount of power the system can use and not break down
- Smartphone
  - 2-3 W
    - the battery on my Motorola Atrix
      - 1.88 Ah * 3.7 V = 7Wh
    - but if you really would use that, the would battery only last ~2.5 h
- Tablet
  - up to ~10 W

An Analysis of Power Consumption in a Smartphone

Aaron Carroll
NICTA and University of New South Wales
Aaron.Carroll@nicta.com.au

Gernot Heiser
NICTA, University of New South Wales and Open Kernel Labs
gernot@nicta.com.au

Originally published in Proceedings of the 2010 USENIX Annual Technical Conference

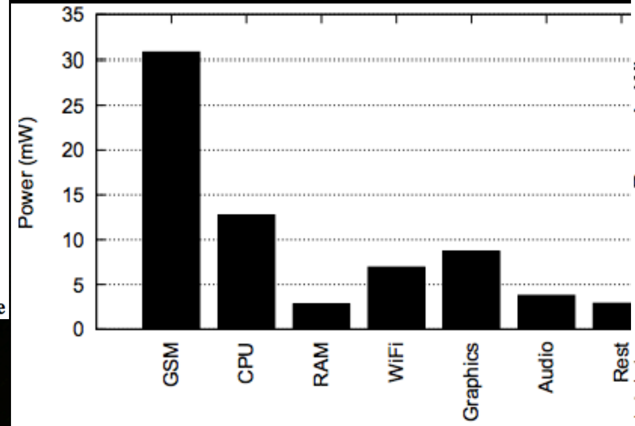


Figure 2: Power breakdown in the suspended state. The aggregate power consumed is 68.6 mW.

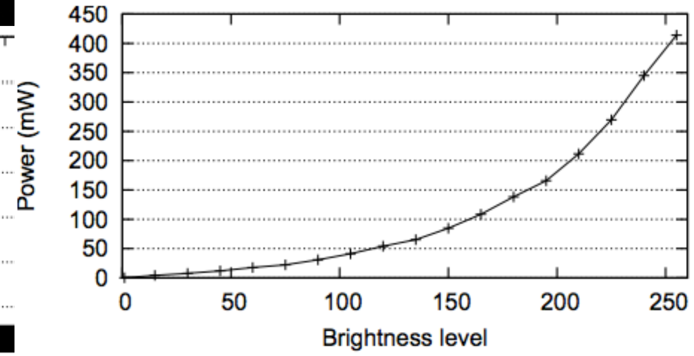


Figure 4: Display backlight power for varying brightness levels.

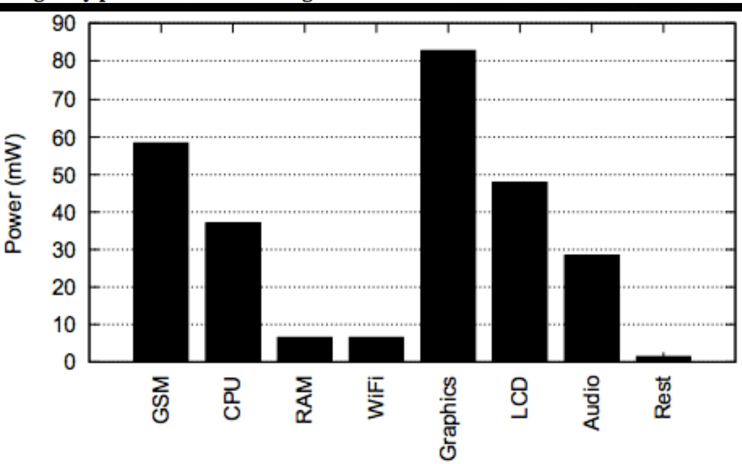


Figure 3: Average power consumption while in the idle state with backlight off. Aggregate power is 268.8 mW.

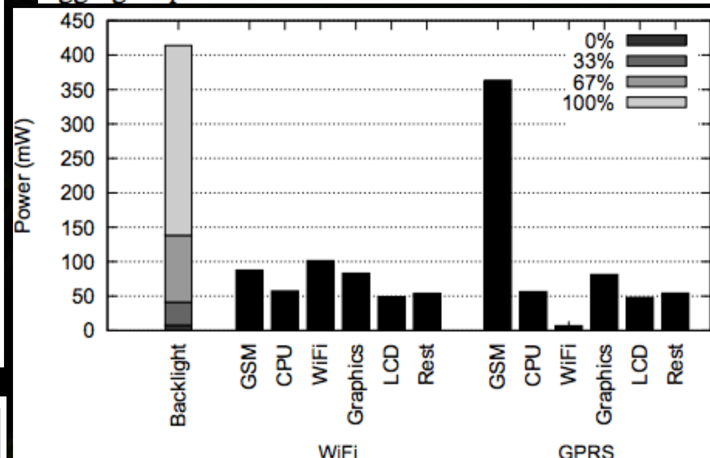


Figure 12: Power consumption for the email macro-benchmark. Aggregate power consumption (excluding backlight) is 610.0 mW over GPRS, and 432.4 mW for WiFi.

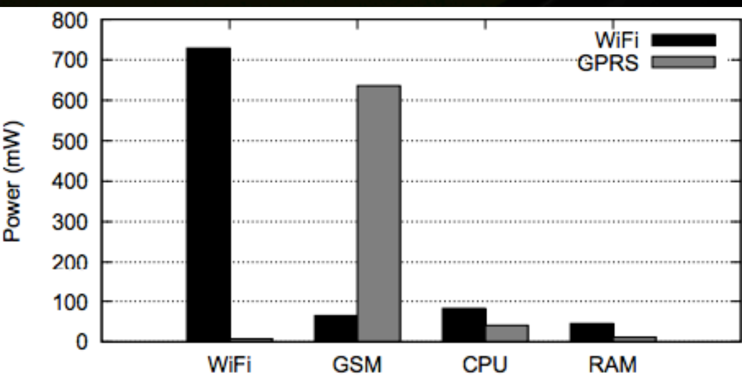


Figure 7: Power consumption of WiFi and GSM modems, CPU, and RAM for the network micro-benchmark.

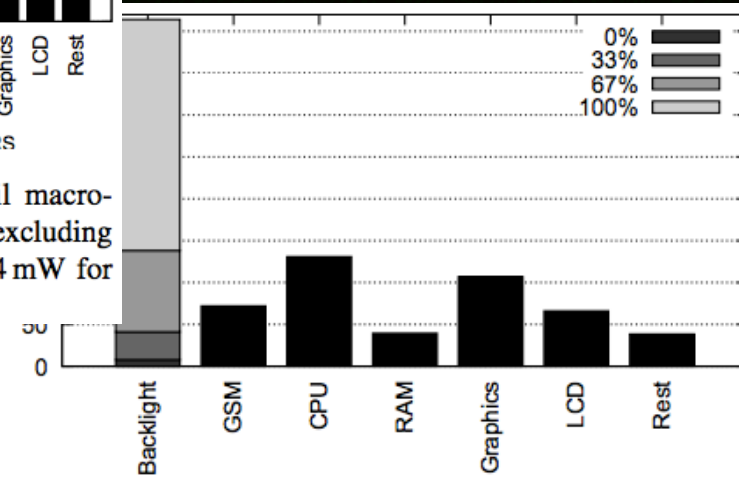


Figure 9: Video playback power breakdown. Aggregate power excluding backlight is 453.5 mW.

Which takes more energy?

Performing a 64-bit floating-point FMA:

```
      893,500.288914668
×          43.90230564772498
--------------------------------
= 39,226,722.78026233027699
+           2.02789331400154
--------------------------------
= 39,226,724.80815564
```

Or moving the three 64-bit operands 20 mm across the die:

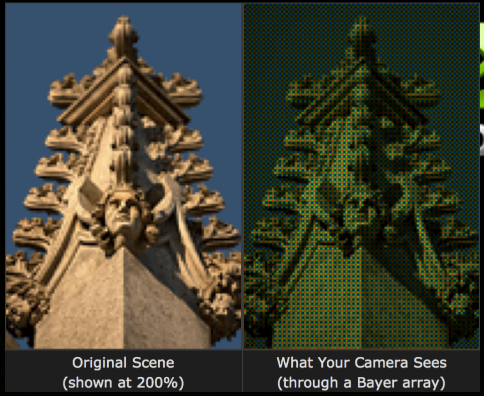
This one takes over 4.7x the energy today (40nm)!

It's getting worse: in10nm, relative cost will be 17x!

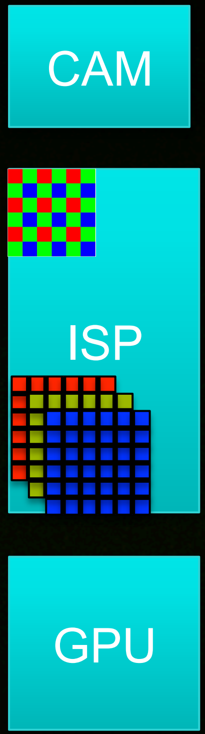
Loading the data from off chip takes >> 100x the energy

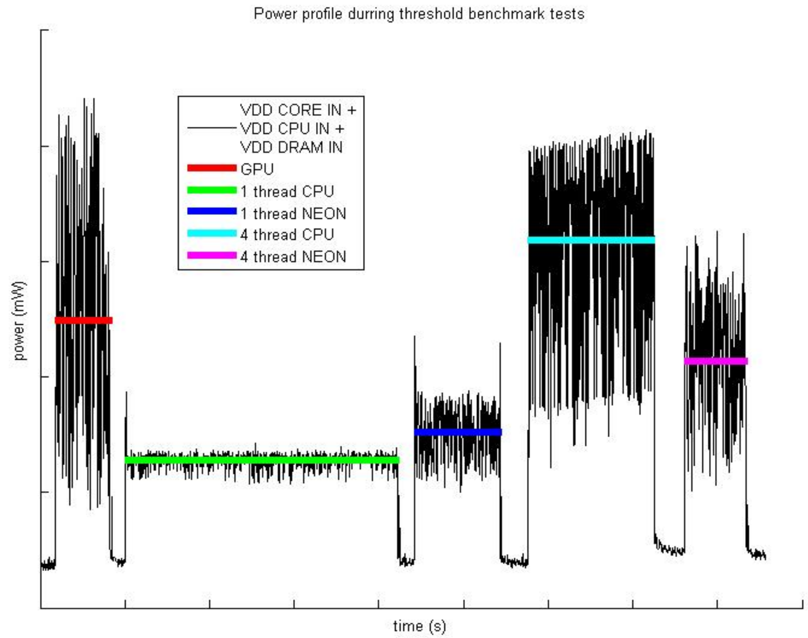
And wire *delay* (ps/mm) is not improving

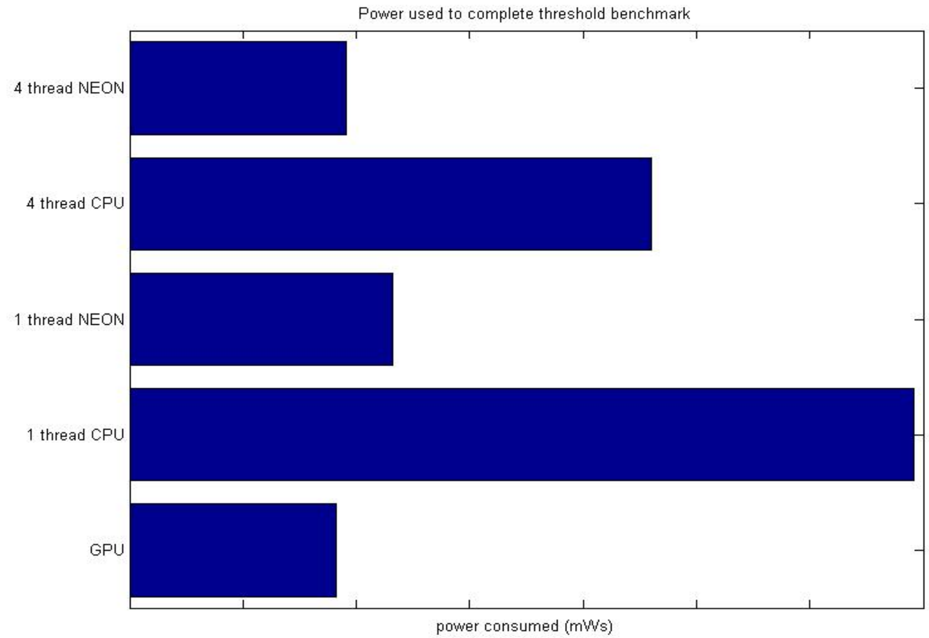
ISP vs. GPU

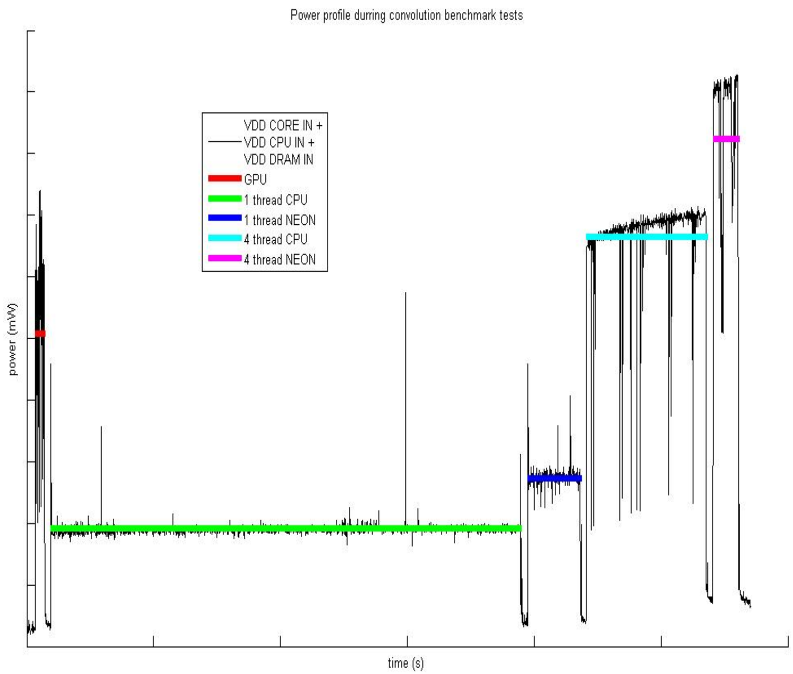


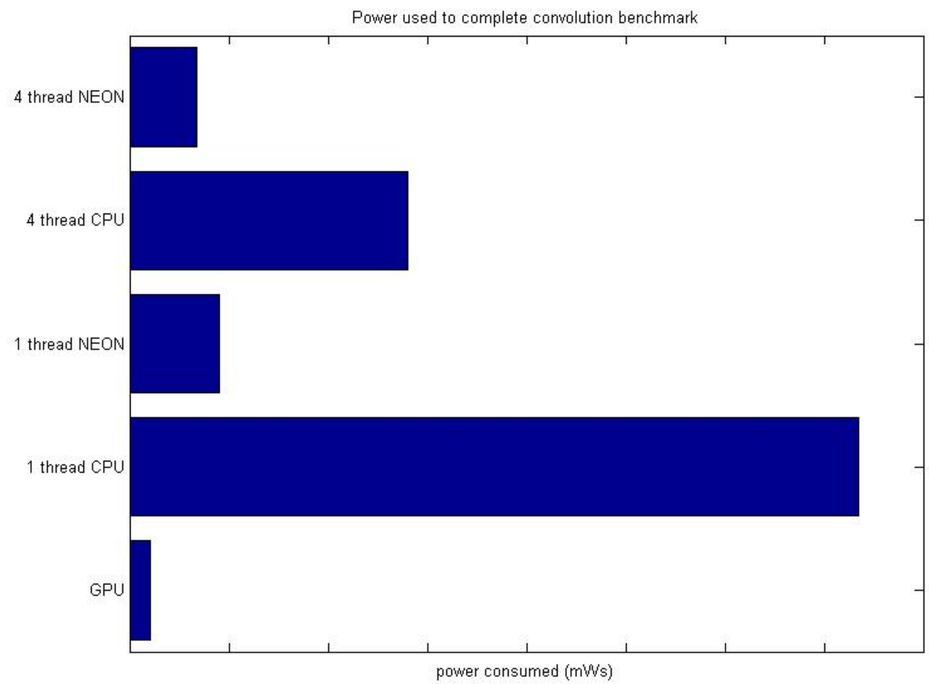
- ISP = Image Signal Processor
  - processes the image right after the camera
  - line-based: most of the processing elements just see a (piece of) single scanline
    - demosaicking needs to see several
  - not very programmable
  - low-power
- GPU
  - can access all pixels (or at least tiles)
  - programmable
  - uses less power than CPU, but more than ISP
- Future?
  - what are the roles, how should they collaborate




Power profile durring threshold benchmark tests
VDD CORE IN +
VDD CPU IN +
VDD DRAM IN
GPU
1 thread CPU
1 thread NEON
4 thread CPU
4 thread NEON
power (mW)
time (s)


Power used to complete threshold benchmark
4 thread NEON
4 thread CPU
1 thread NEON
1 thread CPU
GPU
power consumed (mWs)

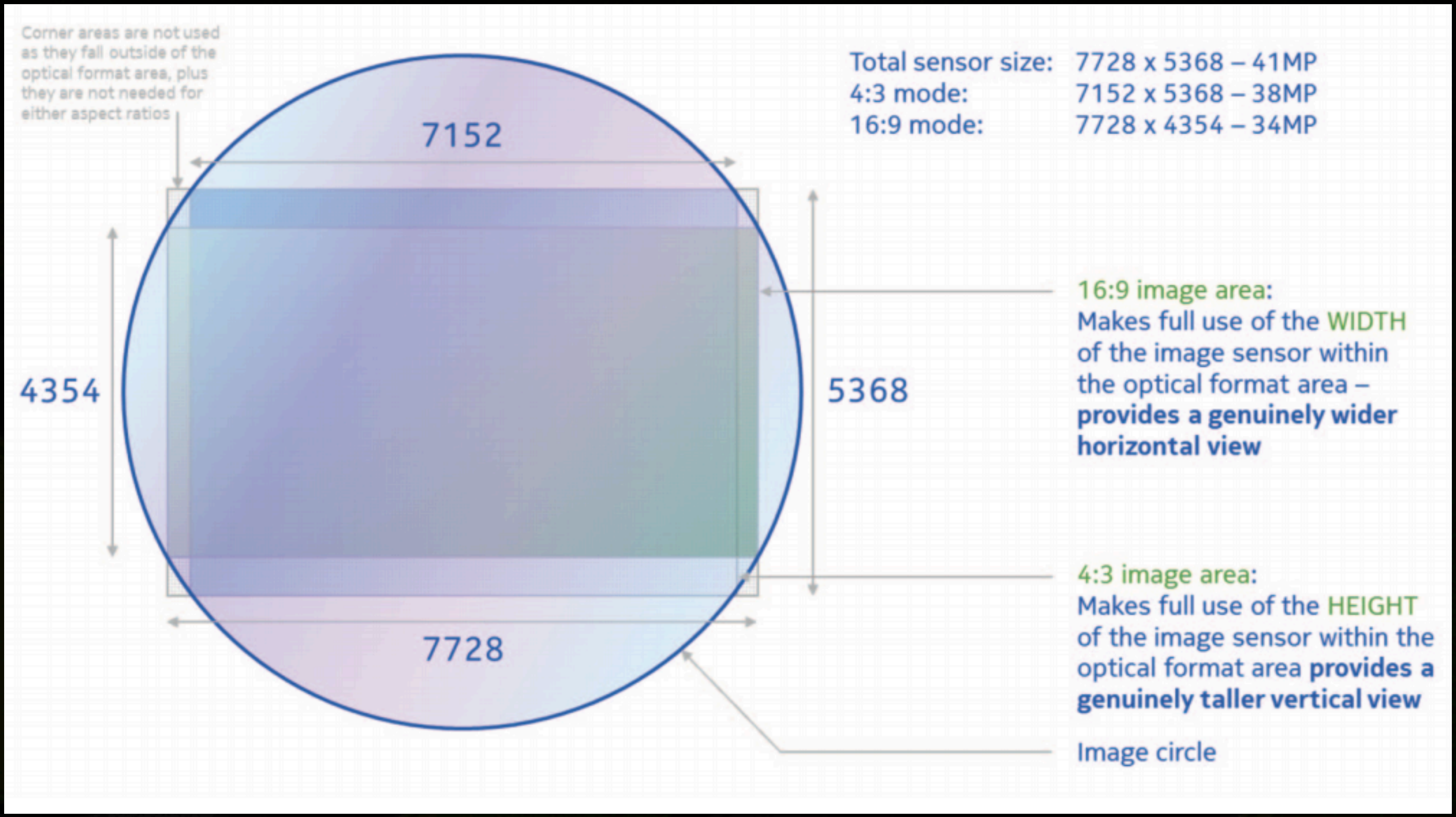

Power profile during convolution benchmark tests
VDD CORE IN +
VDD CPU IN +
VDD DRAM IN
GPU
1 thread CPU
1 thread NEON
4 thread CPU
4 thread NEON
power (mW)
time (s)


Power used to complete convolution benchmark
4 thread NEON
4 thread CPU
1 thread NEON
1 thread CPU
GPU
power consumed (mWs)

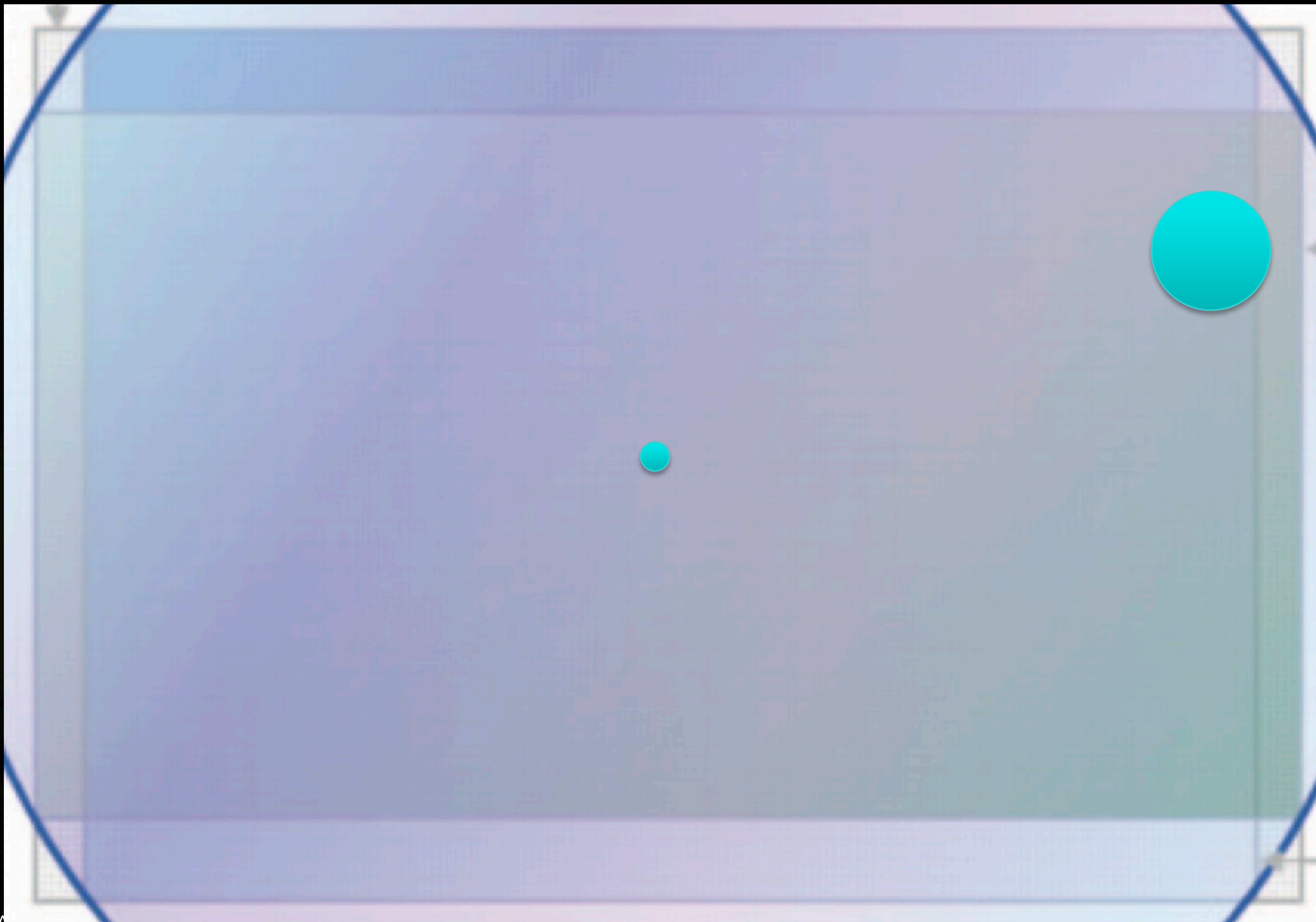
Megapixel race

- Sony Xperia: 12 MP
- HTC Titan II: 16 MP
- More pixels
  - smaller
  - lower quality
- Good for digital zoom
  - and marketing!

Nokia Pureview – Huge 41MP sensor

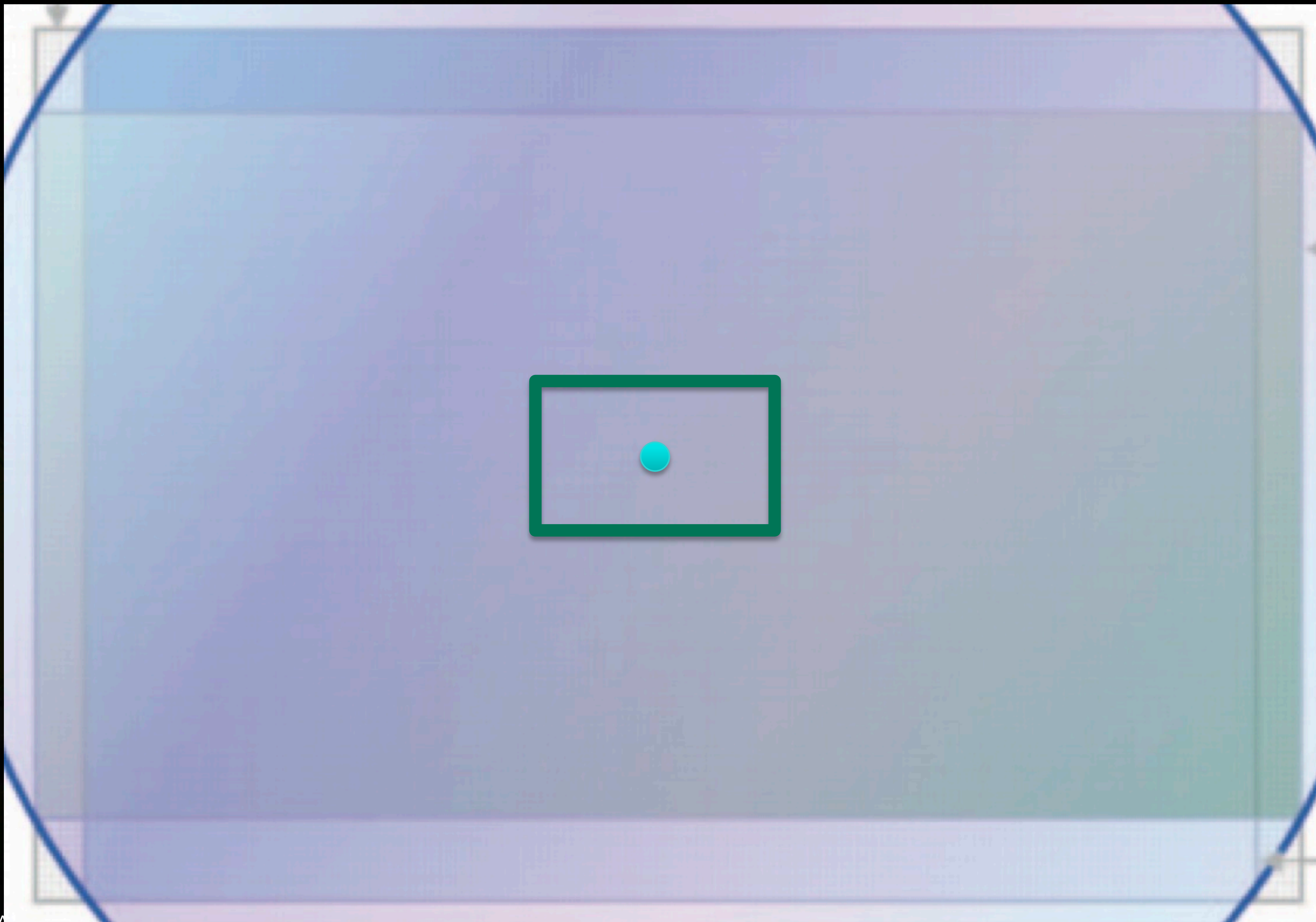


Lens focus varies

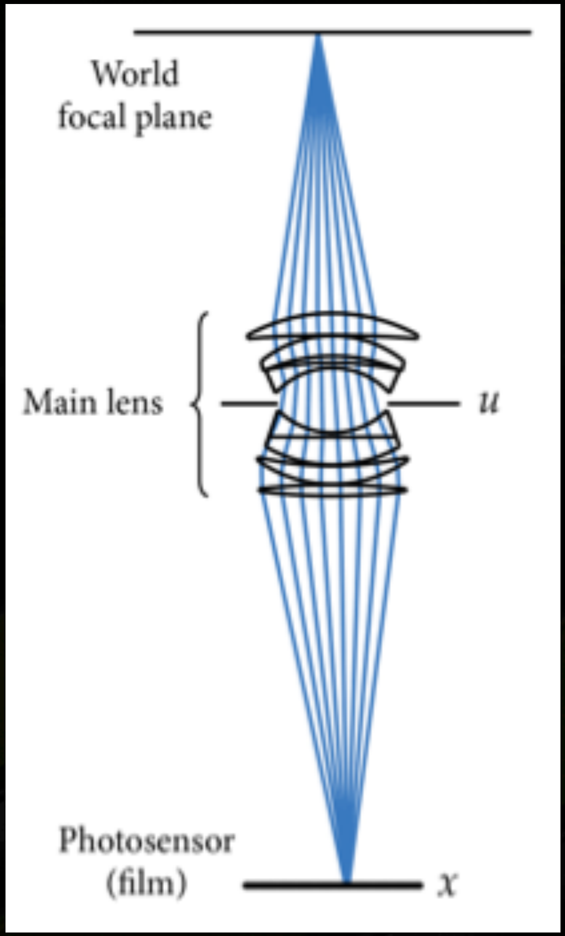


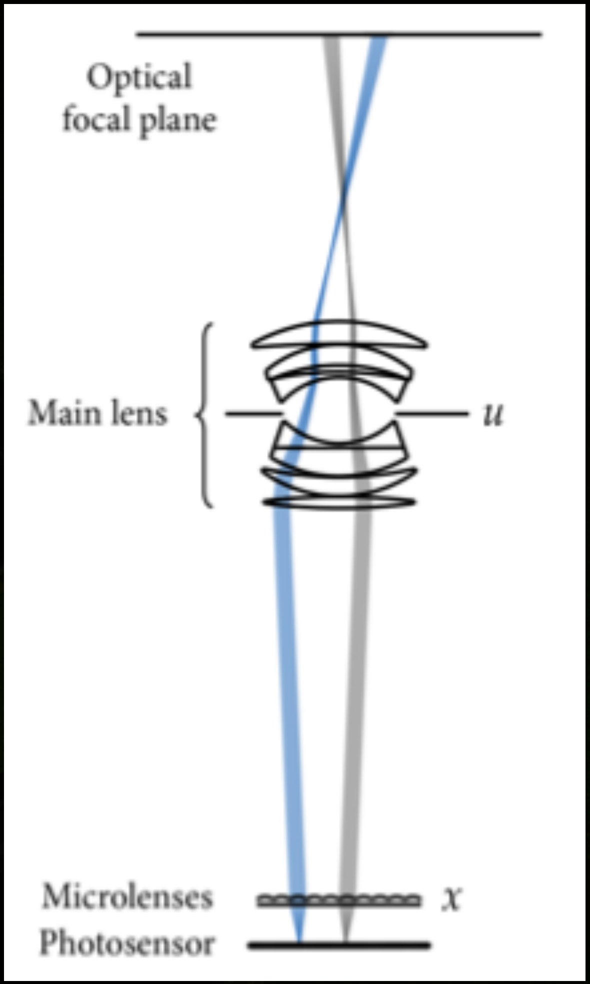
Full view: 3/5/8 MP really low-noise pixels

Zoom in: 3/5/8 MP as good as "normal" camera



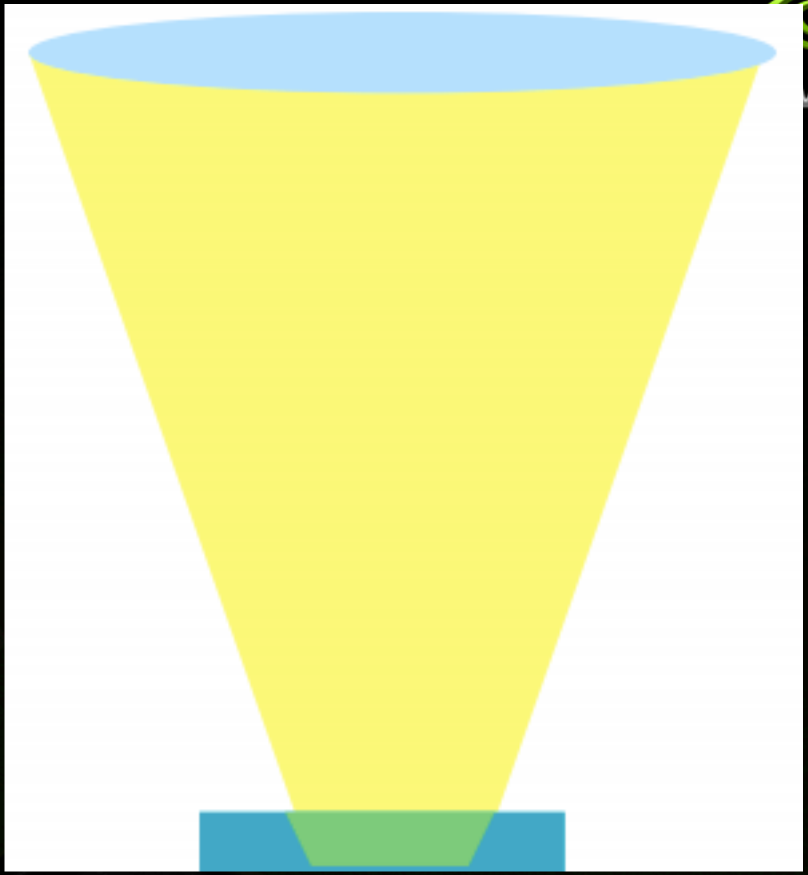
Lots of pixels? Use them differently

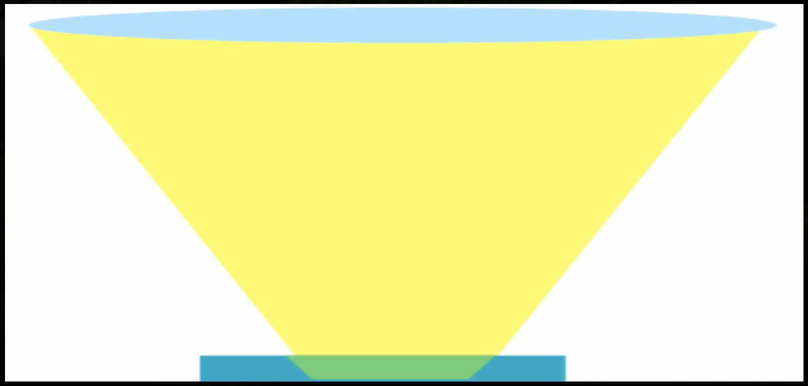


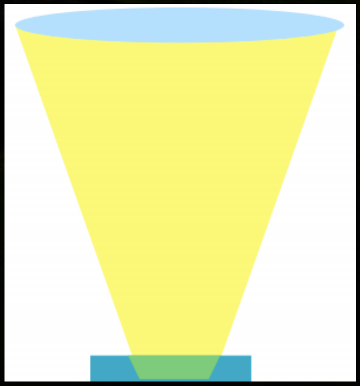


Lots of pixels? Use them differently

Thickness problem







Want thinner optics? Spread it around.

Big optics?

i9 Concept

http://www.blackda.com/black-cs-leicai9.html

WVIL Concept (Wireless Viewfinder Interchangeable Lens)

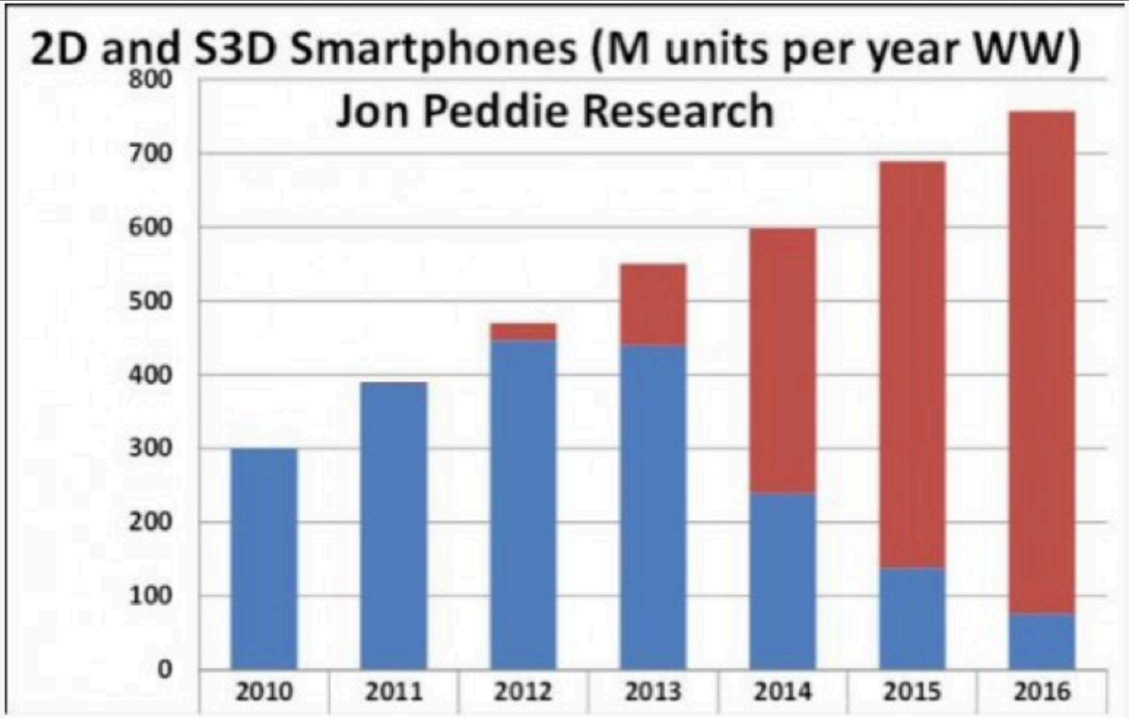
Ricoh GRX

The GXR Body Itself Without a Lens/Sensor Module

Stereo

At least Jon Peddie believes in 3D

- **It's not just about games; the addition of two cameras will enable gesture-based controls, advanced AR applications, visualization with depth, and real-world data capture**



80% of smartphones will have S3D capture and display capability by 2015 (Source: Jon Peddie Research)

Online comment by "ralphg":
3D tvs failed. 3D cinema is limping. 3D digital cameras have not taken off. Popup books are not more popular than flat paper ones.

…

As for smartphones, in the real world, people use them simply as portable computers: check email, send texts, make phone calls, and a few other tasks. No 3D in that.

Stereo can help with "traditional" photos

Real-Time Disparity Map-Based Pictorial Depth Cue Enhancement

Eurographics 2012

Christoph Rößing[1] Johannes Hanika[2] Hendrik Lensch[3]

[1]Daimler AG, Ulm, Germany [2]Weta Digital, New Zealand [3]University of Tübingen, Germany

Figure 1: *Monocular depth cue enhancement by adding depth of field and selective local contrast boosting*

Mobile Panorama

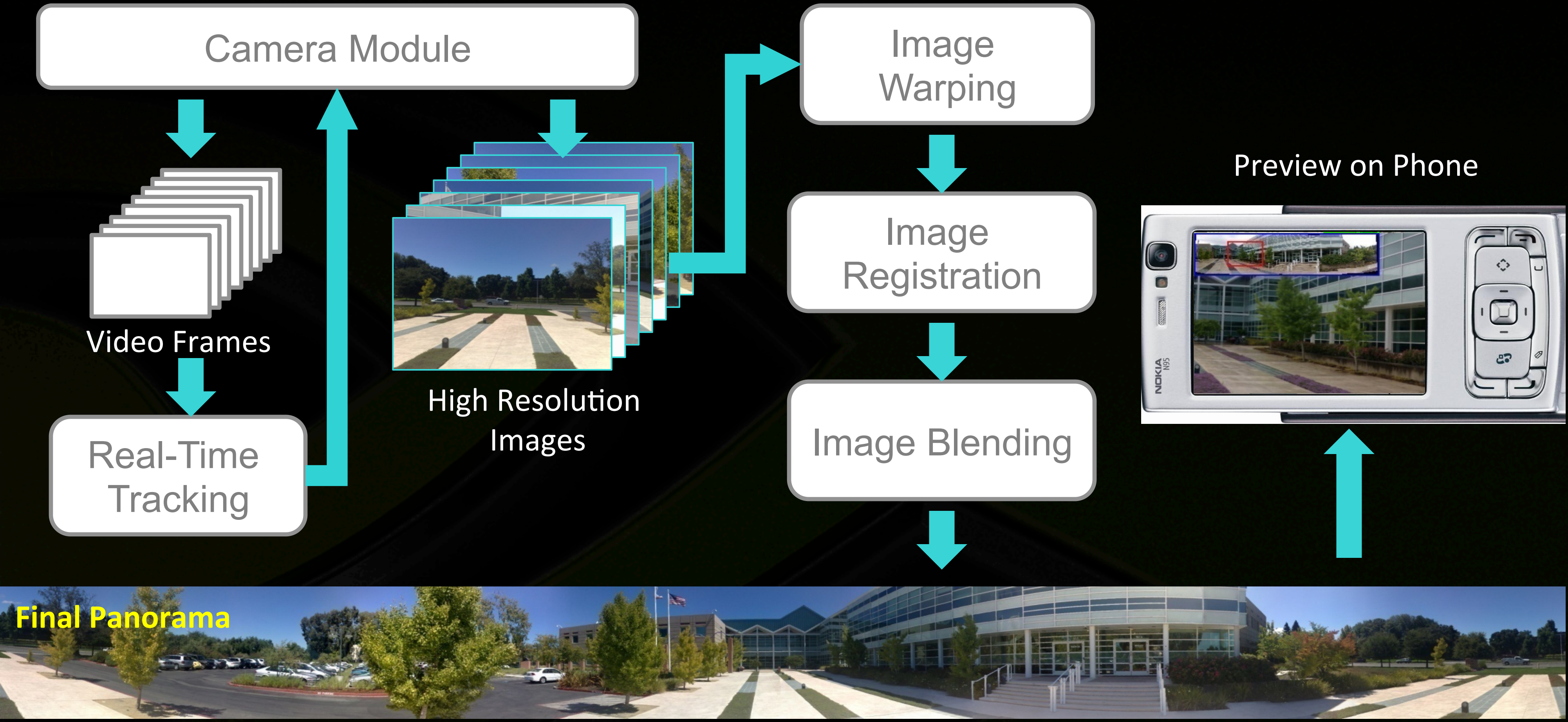


Photo by Marius Tico

SIGGRAPH 2007

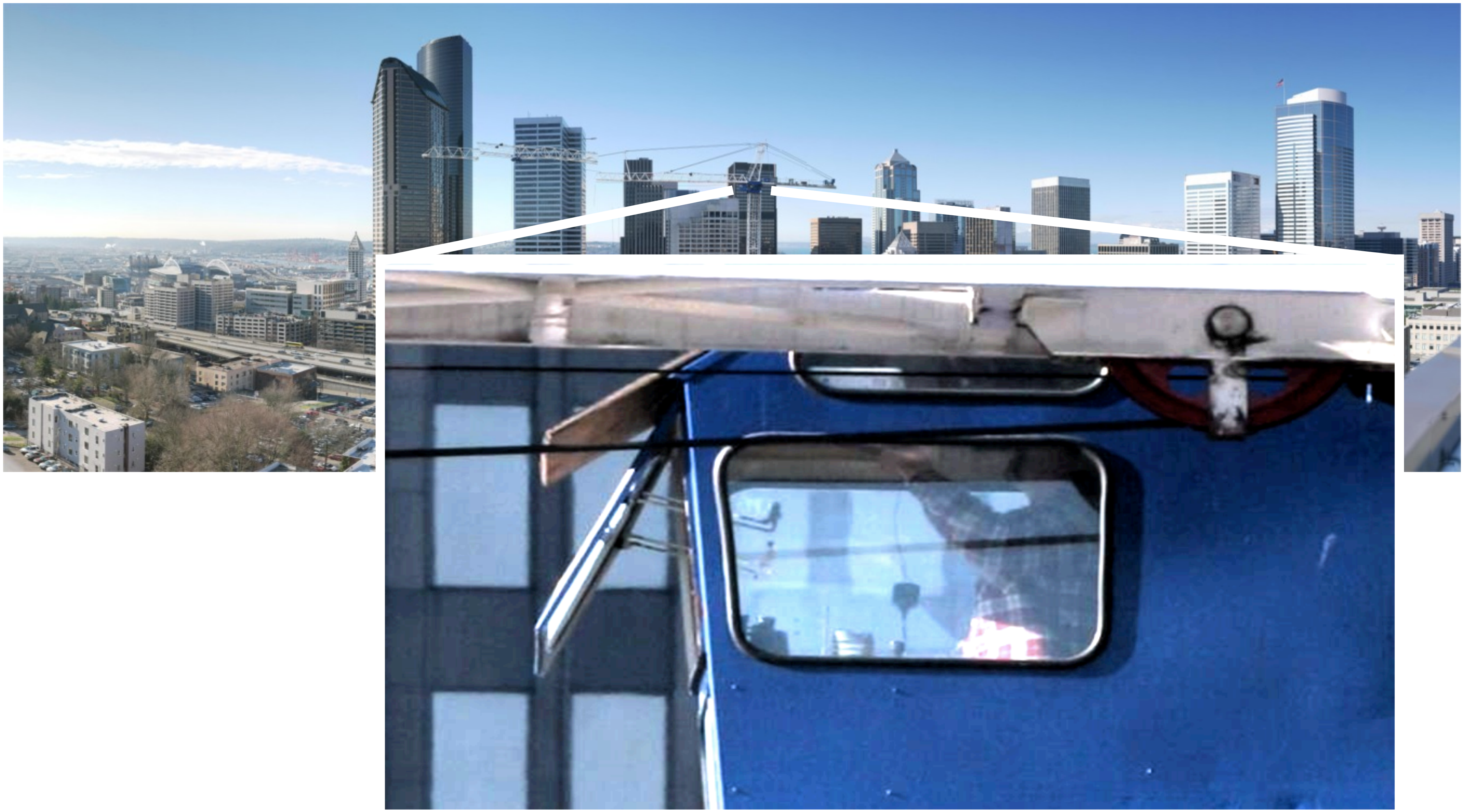
Capturing and Viewing Gigapixel Images

Johannes Kopf — University of Konstanz
Matt Uyttendaele — Microsoft Research
Oliver Deussen — University of Konstanz
Michael F. Cohen — Microsoft Research

3,600,000,000 Pixels

Created from about 800 8 MegaPixel Images

BIG



272 Gigapixel Image Made Using 12,000 Photos from a Canon 7D

by ERIC REAGAN on JANUARY 14, 2011

in PHOTOGRAPHERS

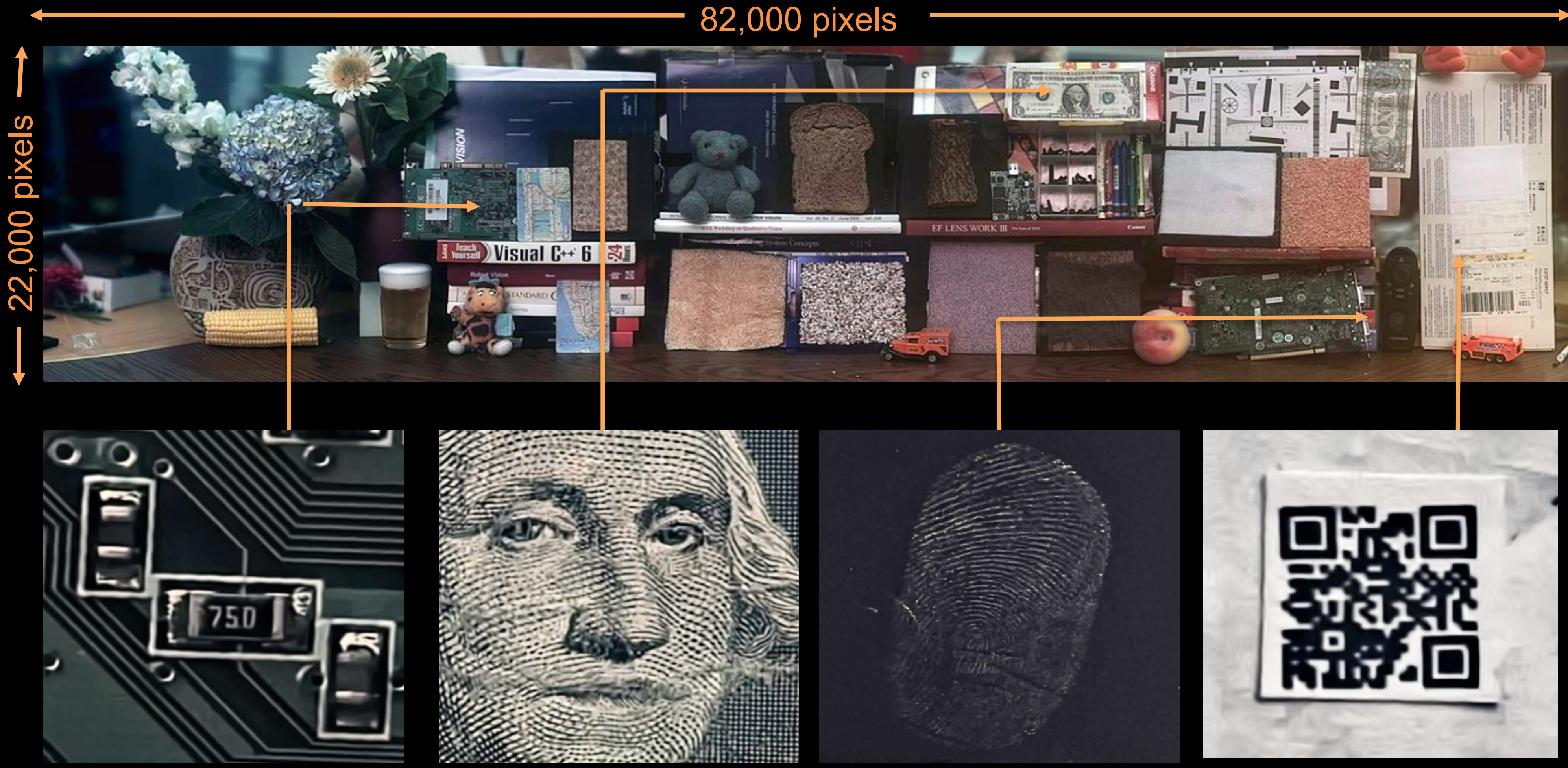
http://gigapan.org/gigapans/66626/

Photographer Alfred Zhao captured this 272 gigapixel image of the Shanghai skyline using the GigaPan EPIC Pro and a Canon 7D with a 400mm f/5.6 lens and 2x teleconverter attached. He was setup and started shooting at around 8:30am and after 12,000 images were in the bag, it was just before dusk. It took months to complete image and get the final 1.09TB file uploaded.

Just how big is a 272 gigapixel image? 1 gigapixel = 1000 megapixels = 1 billion pixels. That's 272 BILLION pixels. Printed at standard resolution, this image would cover over 7000 billboards.

But now it's done and Zhao holds a world record for the largest digital photo. There's no time to rest though, as Zhao says, "This is not the end of my panorama journey, it is a new start, challenging the limit is an infinite process. New records will appear in the future, it is only a matter of time."

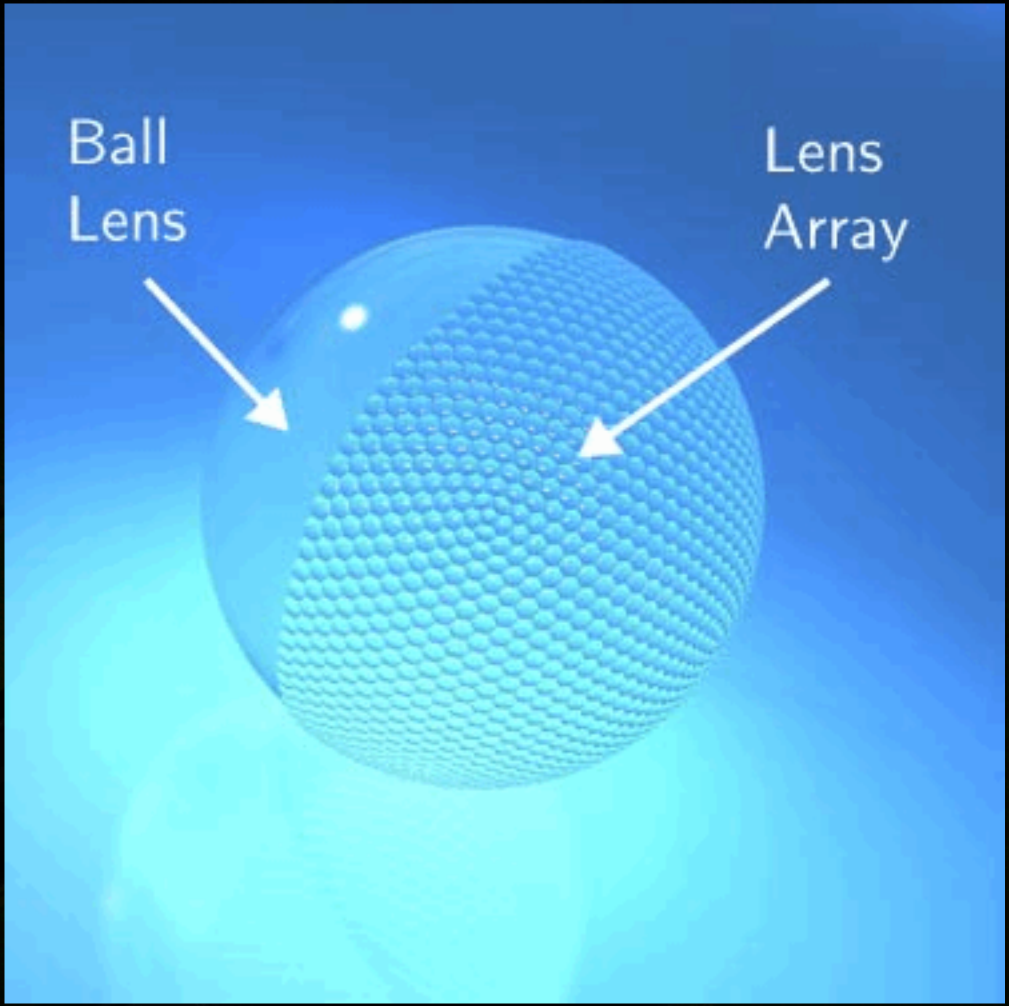
Still Life (1.7 Gigapixels)



Resistor | Dollar bill | Fingerprint | 2D barcode

URL: http://gigapan.org/gigapans/0dca576c3a040561b4371cf1d92c93fe/

Gigapixel Camera



http://www.cs.columbia.edu/CAVE/projects/gigapixel/

Harry Potter Pictures

Video Textures

Schödl, Szeliski, Salesin, Essa. Video textures. SIGGRAPH 2000

Cinegraphs / Cinemagraphs

http://cinemagraphs.com/

http://cinemagram.tumblr.com/

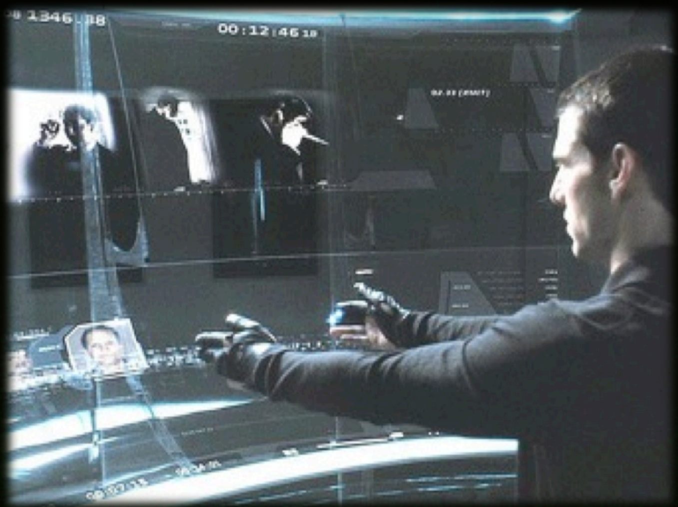
Computer Vision = New Applications

Augmented Reality

Augmented Reality Ghost Hunger (Argh)

Wordlens

Gesture interfaces



Google Goggles

Vision-based AR

Camera video stream sent to the compositor

Camera images used to track the camera's location and orientation

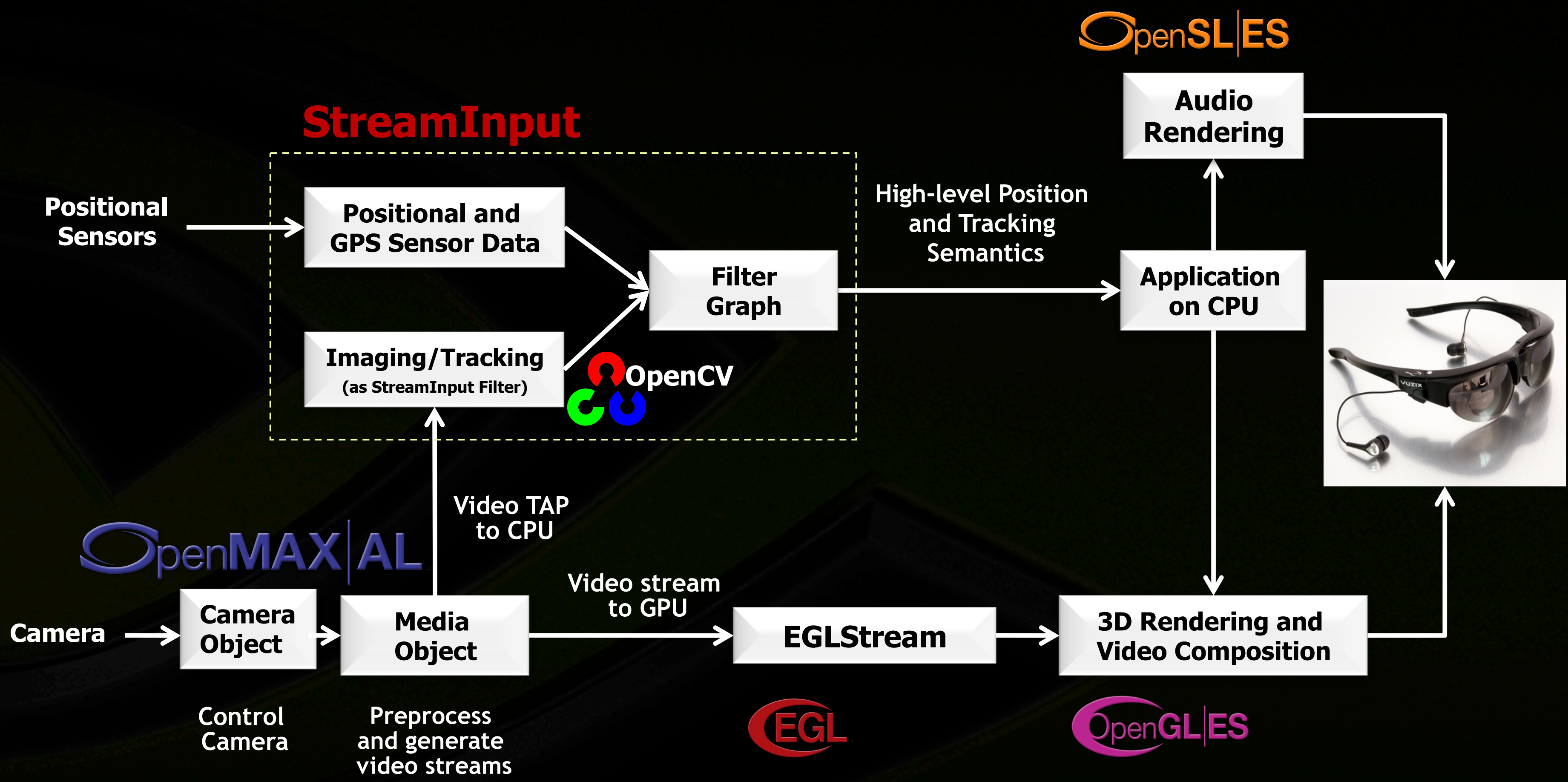
3D augmentations composited with video stream

Camera Tracking

Camera-to-scene transform locks the 3D rendering to the real world

3D Augmentation Rendering

Tegra AR Flow – Maximizing Acceleration



Key AR enabler: Tracking

Using active light

Your face is your key

Gesture UIs

Kinect makes it easier

Summary

- Camera is a key driver for compute on mobile devices
- Cameras provide tools for story telling
  - and allow interaction with real world
- Small size is at odds with high-quality cameras
  - compensate with computation
- Small size and batteries → limited power
  - need to optimize the whole system, both SW and HW
- The relative roles and programming of ISP and GPU
  - an interesting research topic