Visual Computing

Prof. Marc Pollefeys

Prof. Stelian Coros

ETH

inf | Informatik
Computer Science

Two parts

- Part I - Image processing

  Prof. Marc Pollefeys

  Main assistant: Remi Pautrat

- Part II - Computer graphics

  Prof. Stelian Coros

  Main assistant: James Bern

ETH

Topics – Image Processing

- Digital images and sensors
- Image segmentation and morphology
- Convolution and image features
- Fourier transform and filtering
- Unitary transformations and image compression
- Warping, Optical flow and video compression
- Texture analysis and Radon transform

ETH

Topics – Computer Graphics

- Drawing triangles, rasterizing
- Transformations, Geometry and textures
- Rendering pipeline
- Lighting, shading, visibility and shadows
- Animation, rigging
- Physically-based simulation, PDEs

Course Logistics

https://graphics.ethz.ch/teaching/viscomp18/

- Lectures:
  - Tue: 10-12, HG G-3
  - Thu: 13-15 HG G-3

- Excercises (from next week):
  - Tue: 13-16, CHN G 42
  - Thu: 09-12, IFW A36

Exercises

- Grade determined by final exam
- Attend one of the two sessions each week
- Registration for the exercise groups

- Exercises complement lectures to provide insight and hands-on experience
- Goals is to finish most exercises during session with assistants (no homework)

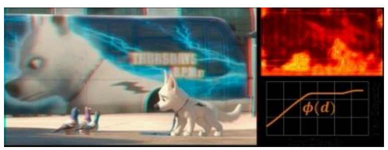
Visual Computing @ ETHZ

- Institute of Visual Computing
  - Computer Vision and Geometry Lab – Prof. Marc Pollefeys
  - Computer Graphics Lab – Prof. Markus Gross
  - Interactive Geometry Lab – Prof. Olga Sorkine
  - Prof. Siyu Tang (Spring 2020)
- Advanced Interactive Technologies – Prof. Otmar Hilliges
- Computational Robotics Lab – Stelian Coros
- Computer Vision Laboratory
  - Prof. Luc Van Gool, Prof. Ender Konukoglu, Fisher Yu (Fall 2020)
- Institute for Machine Learning – Prof. Buhmann, Prof. Hoffman, Prof. Krause, …
- Photogrammetry and remote sensing – Prof. Konrad Schindler
- Autonomous system laboratory – Prof. Roland Siegwart, Prof. Margarita Chli
- Institute of Neuro-Informatics – Prof. Davide Scaramuzza, Prof. Tobi Delbrueck, …

Computer Vision in Zurich

... and other labs and companies based in Zurich

Disney Research SCIENCE AT PLAY



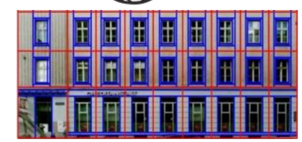
kooaba

ptc

procedural

esri

SCANDIT



Google

Leica Geosystems

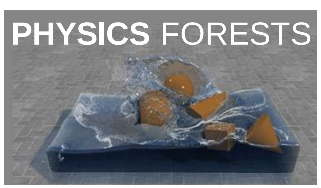
magic leap

Dacuda

logitech

LiberoVision

vizrt



FASHWELL

NOMOKO

faceshift

Super computing systems

Microsoft

astrivis

ETH

Computer Vision and Image Analysis

Examples and Applications

3D + Interactive Maps

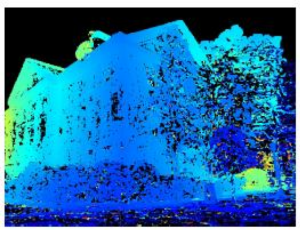
ETH

Automated 3D modeling

Gallup et al. 2010

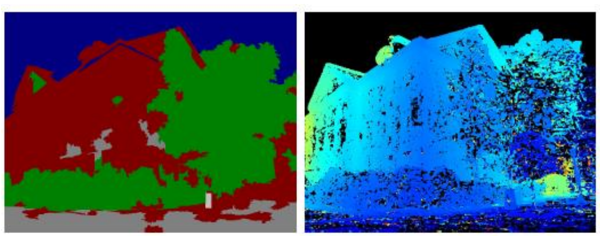
Joint 3D reconstruction and class segmentation

(Haene et al CVPR13)



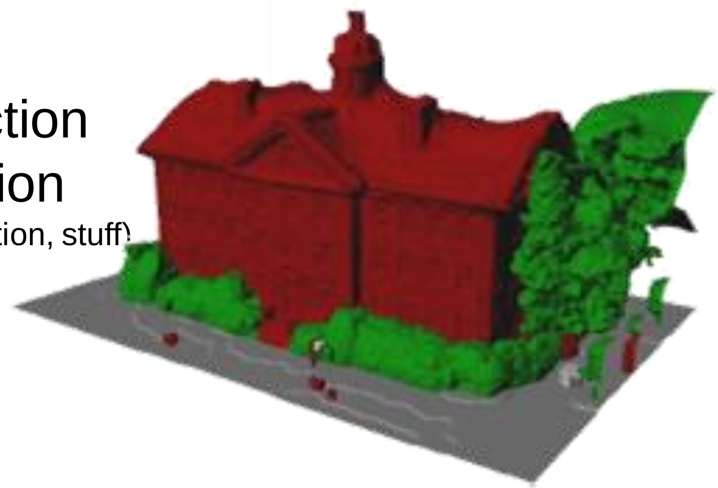
reconstruction only

(uniform smoothness prior)



joint reconstruction
and segmentation

(ground, building, vegetation, stuff)



Robots Learning from Observation

D. Bentivegna & C. Atkeson

Willow Garage's PR2 plays pool

poolshark

ETH

Driver Assistance

[mobileye]()

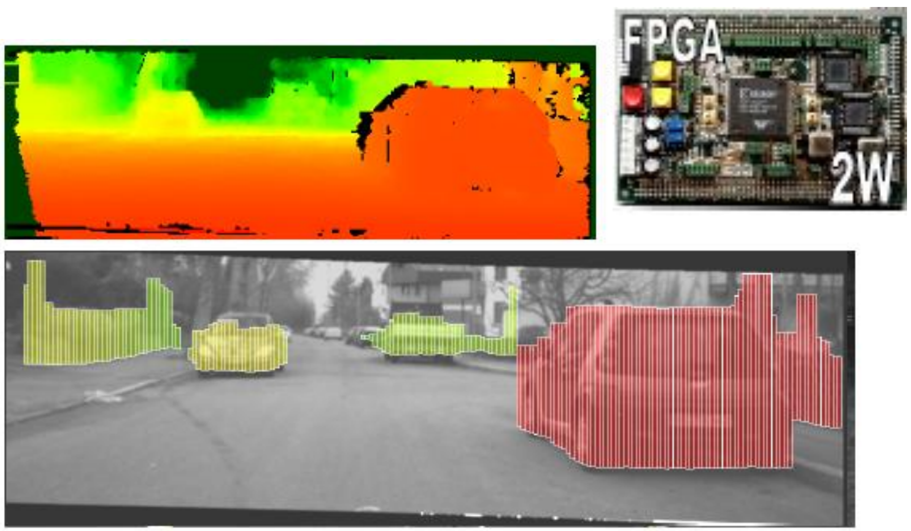
ETH

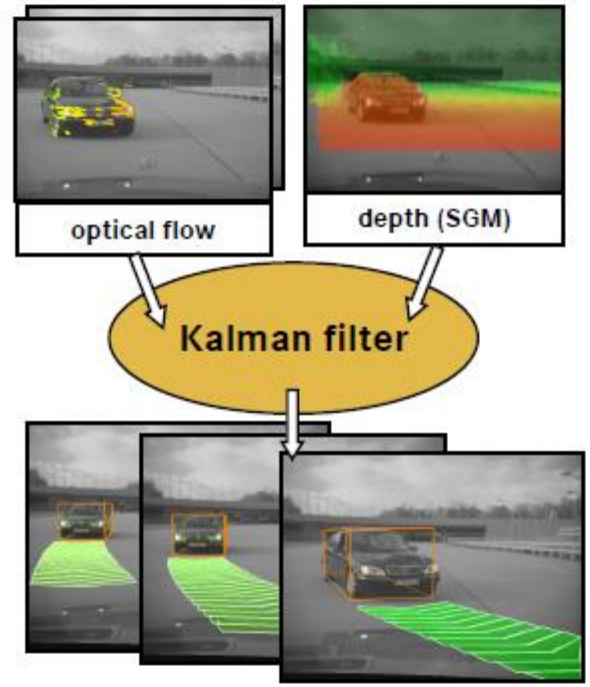

MOBILEYE®
MOBILEYE
Mobileye PCW
Pedestrian Collision Warning

ETH

Driver Assistance

Daimler stereo system





[6D vision]()

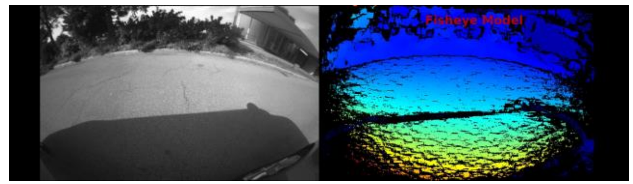
Daimler 6D vision

ETH

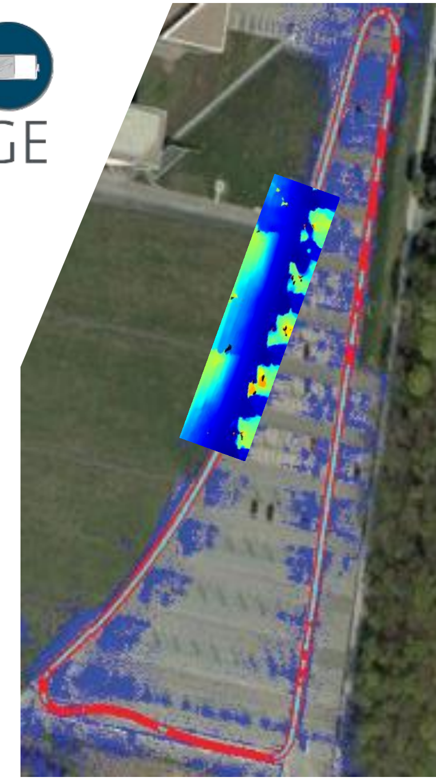
3D mapping for autonomous driving

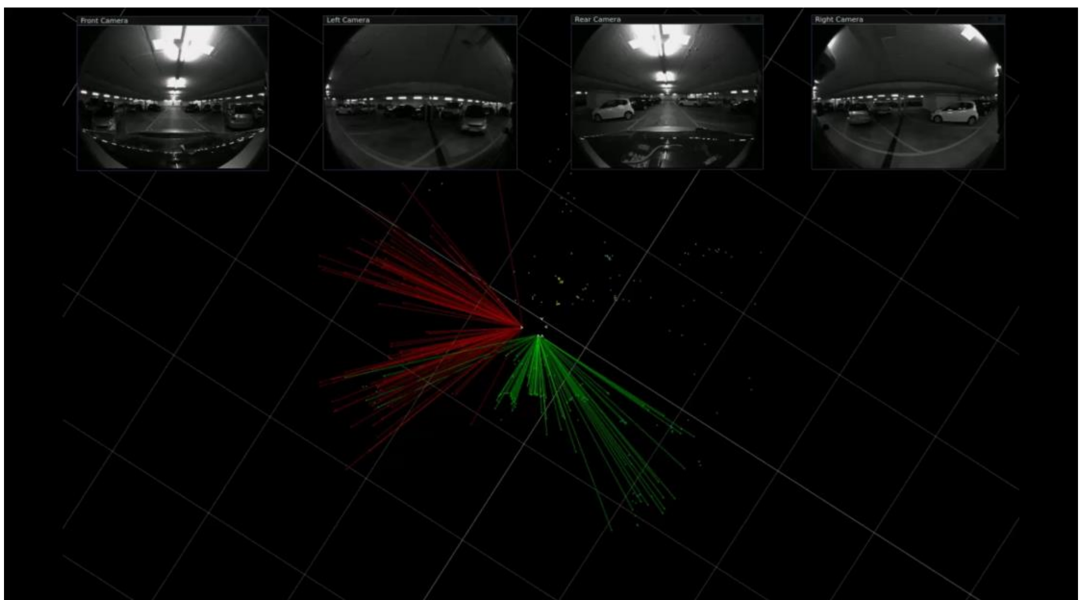
(Lee et al CVPR13;
Heng et al. ICRA14;
Haene et al…)

V-CHARGE

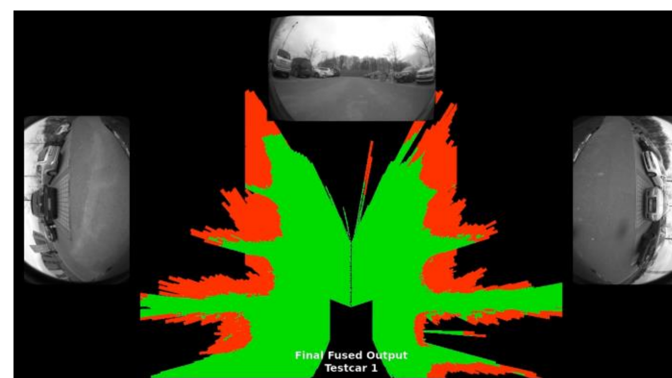


Dense real-time temporal fisheye stereo (GPU)





omnidirectional visual simultaneous localization and mapping



ETH

cameras as main sensors: *no lasers, no GPS*

Robot navigation

Online Environment Mapping

Supplementary Video

Paper ID: #828

collaboration with Honda

Micro Aerial Vehicles

pixhawk

http://pixhawk.ethz.ch/

ETH

auterion

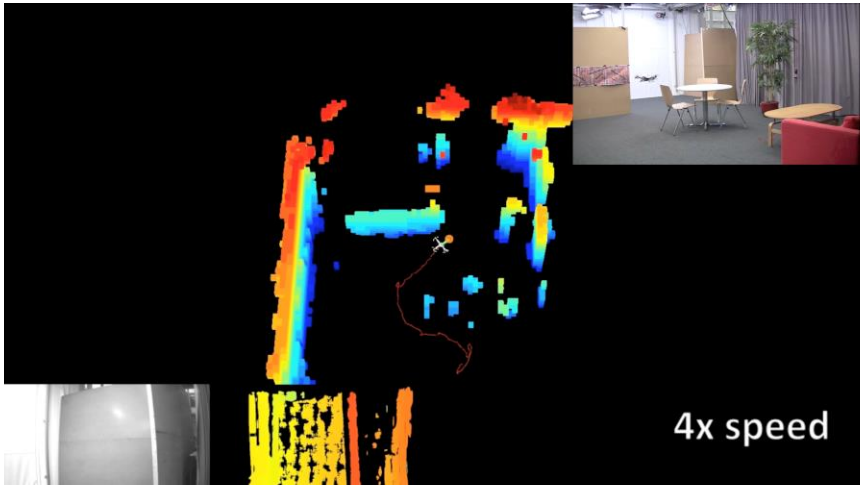
Full autonomous vision-based navigation and mapping

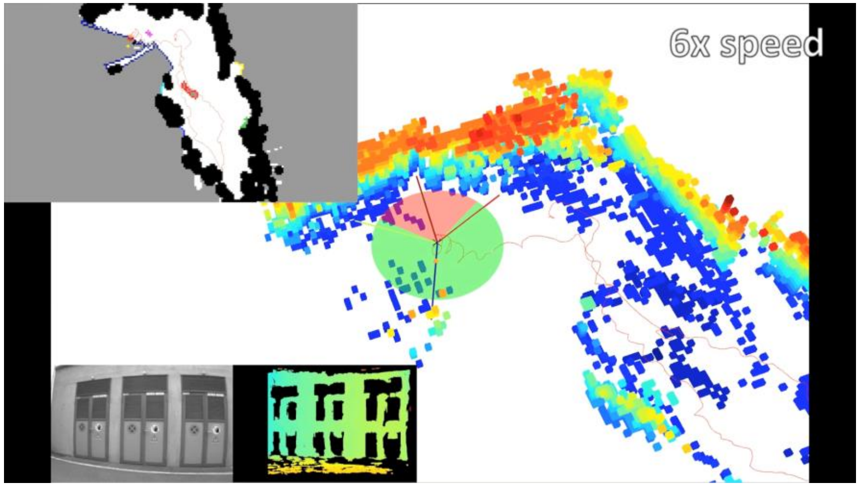
(Fraundorfer et al. IROS12 best paper finalist)

full on-board processing
2+1 cameras + IMU

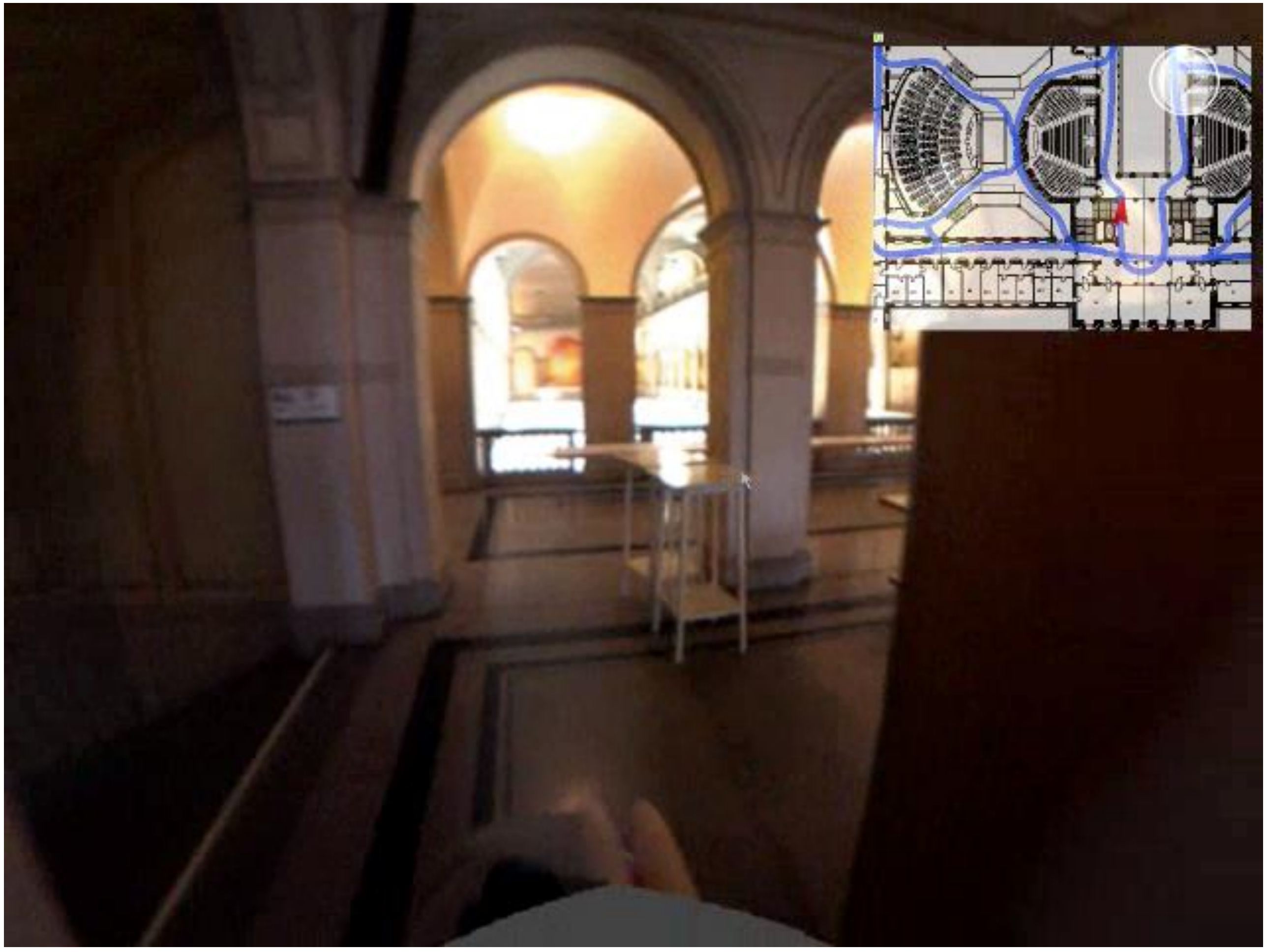
indoor and outdoor operation
obstacle avoidance, mapping
and exploration

no laser, no GPS, no network

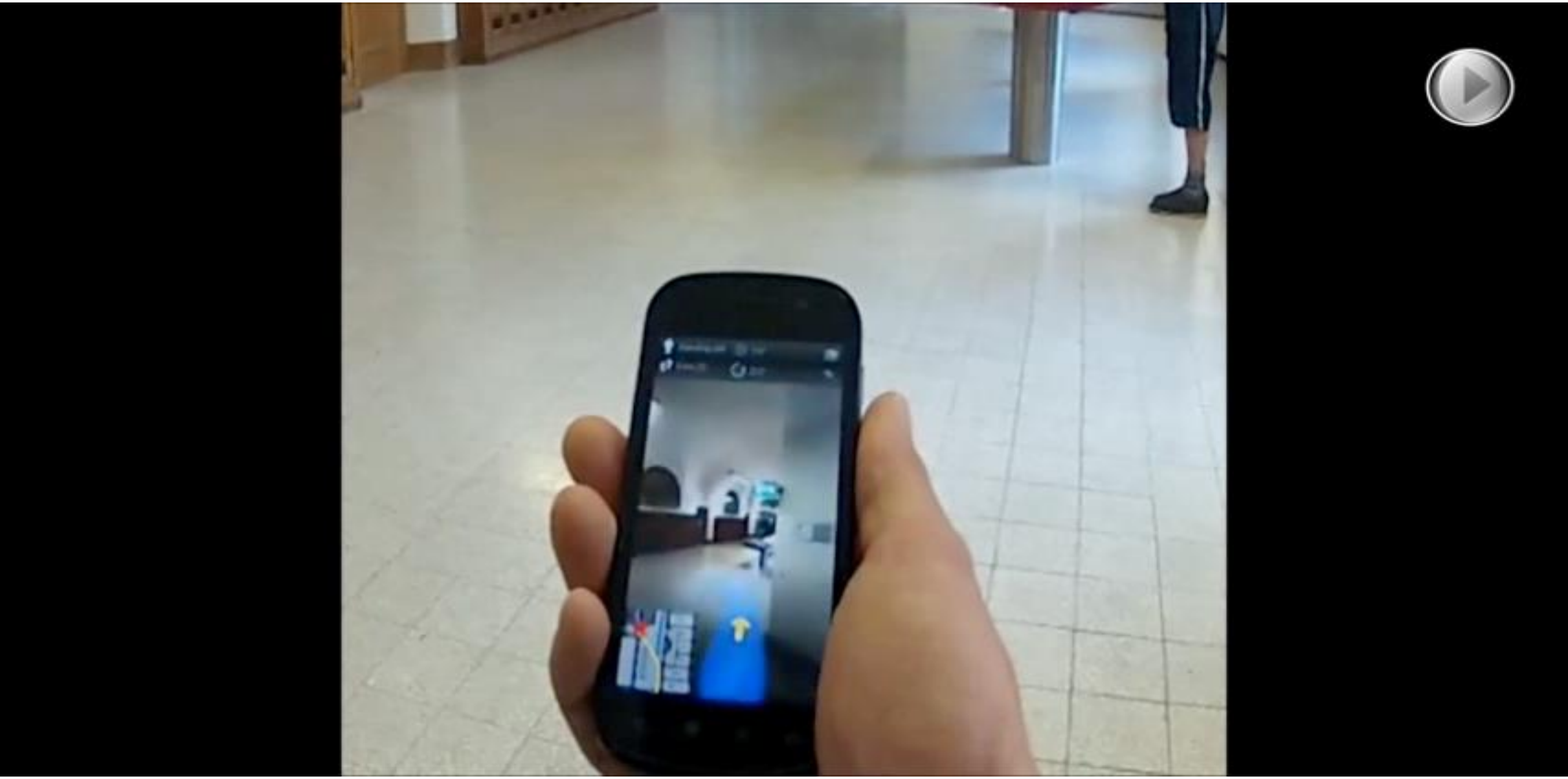




Mapping and localization



MobileTour



Uses accelerometers (as pedometer) and gyroscope to track user location

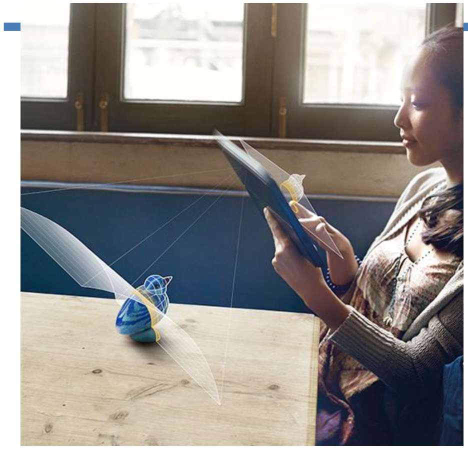
Image-based retrieval

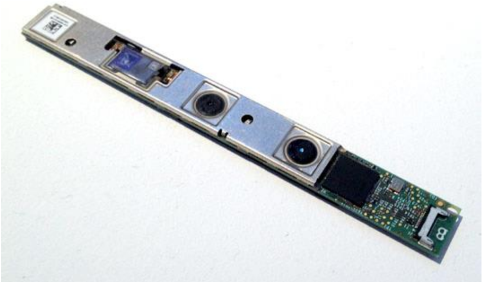
- Google Image Search

- Kooaba (now PTC) (http://www.kooaba.com/)
- Nokia Point & Click
- Seeing AI Microsoft

Intel RealSense technology

Low power (<1W) models for tablet and for PC



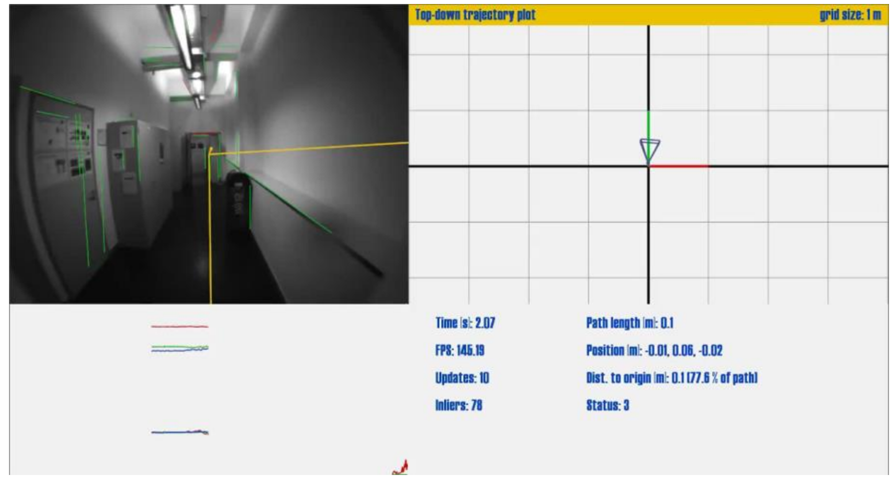


depth from 2 (or 1) cameras + (un)structured laser pattern

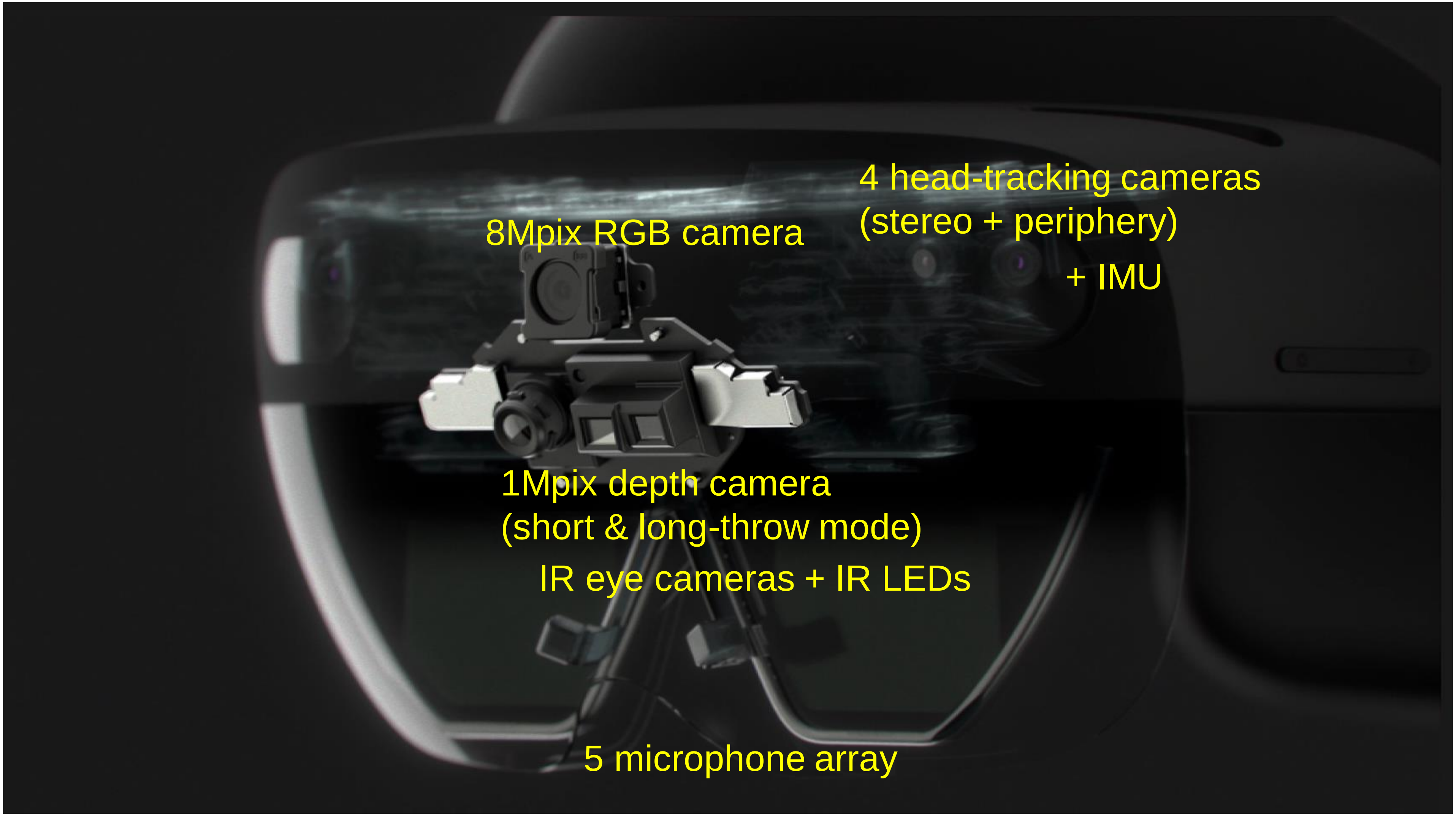

Project Tango

Research partner of Google's Project Tango

Project Tango



ETH


4 head-tracking cameras
(stereo + periphery)
+ IMU
8Mpix RGB camera
1Mpix depth camera
(short & long-throw mode)
IR eye cameras + IR LEDs
5 microphone array

HoloLens2 unpacked

By Alex Kipman, Microsoft Technical Fellow

ETH Global Lecture, October 3, 10am, AudiMax

Visual Modeling for Archaeology

Visual modeling with a hand-held camera, Pollefeys et al. 2004

3D on mobile phone

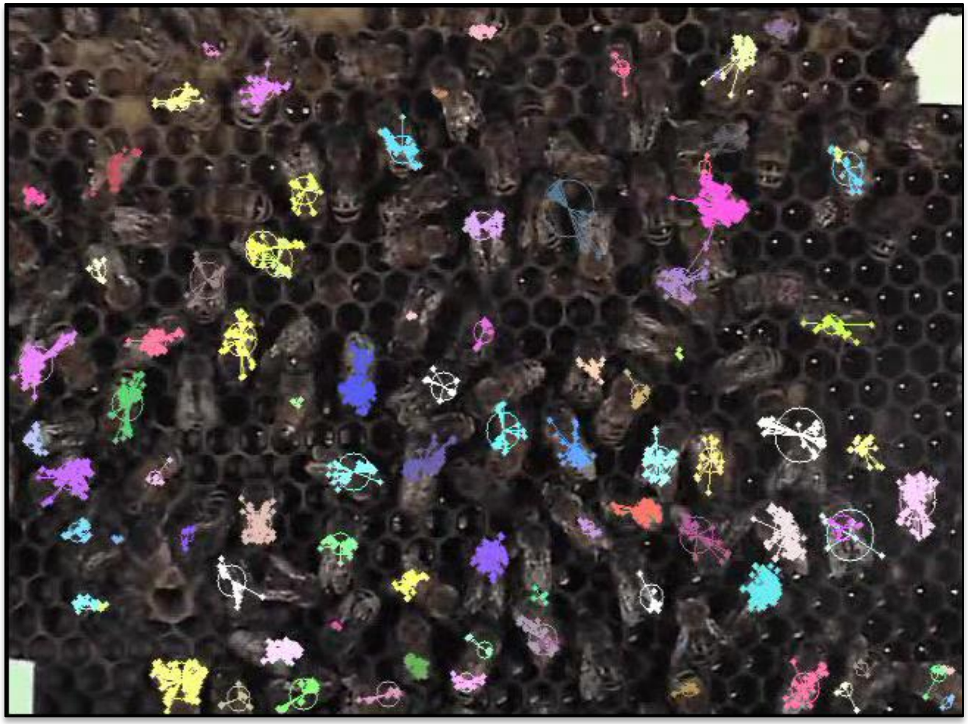
(Tanskanen et al. ICCV13)

(Delaunoy et al. CVPR14)

3D selfies

erc

Surveillance

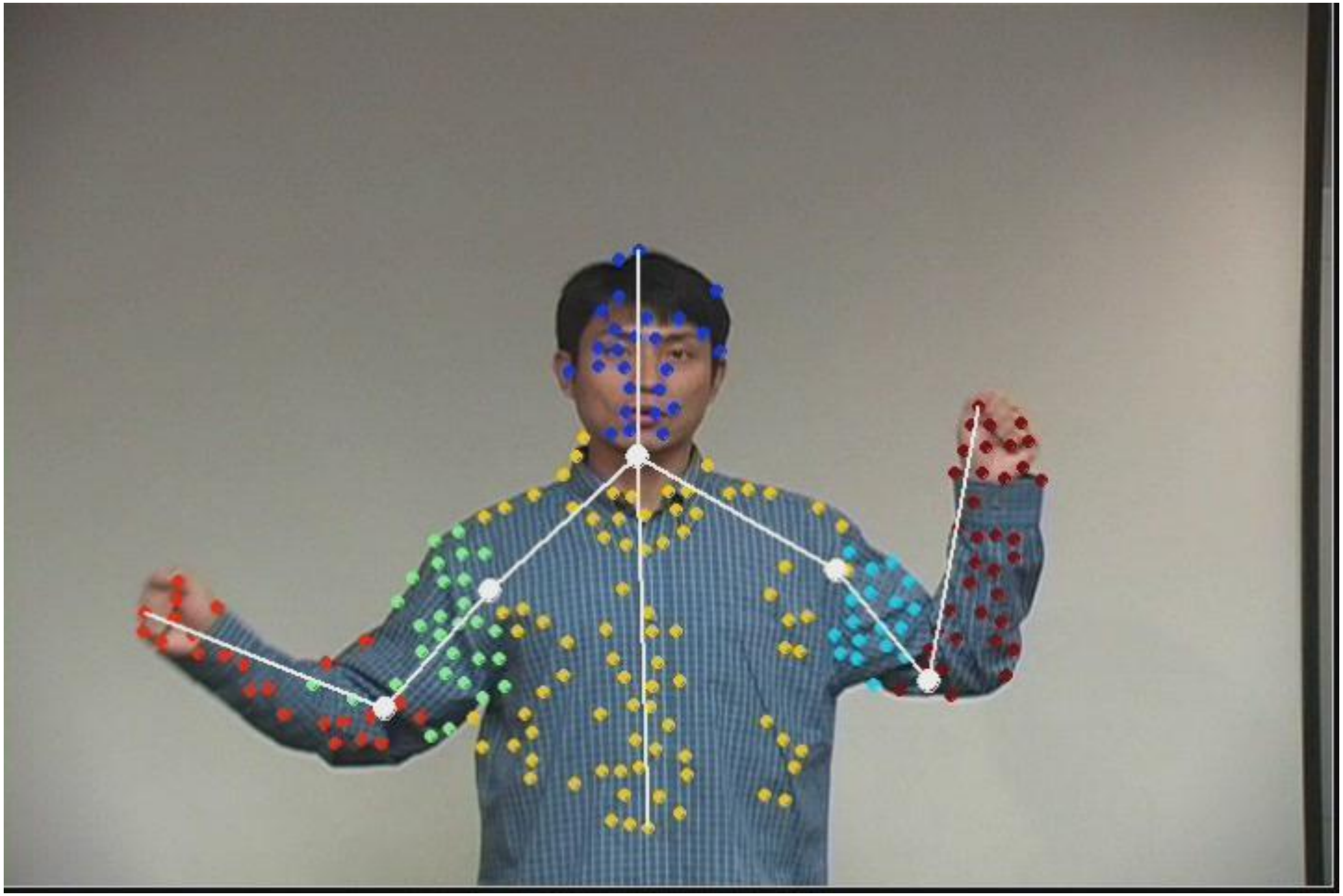


Brostow & Cipolla 2006

Character Animation: Motion Capture

Michael Jackson - Ghost

Articulate Motion Analysis



Yan and Pollefeys 2006

Markerless motion capture

Ballan and Cortelazzo 3DIMPVT2008

MS Kinect

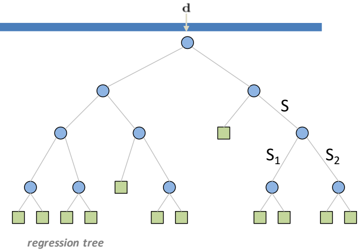
http://www.xbox.com/en-US/kinect

kinect2

ETH

Bodytracking

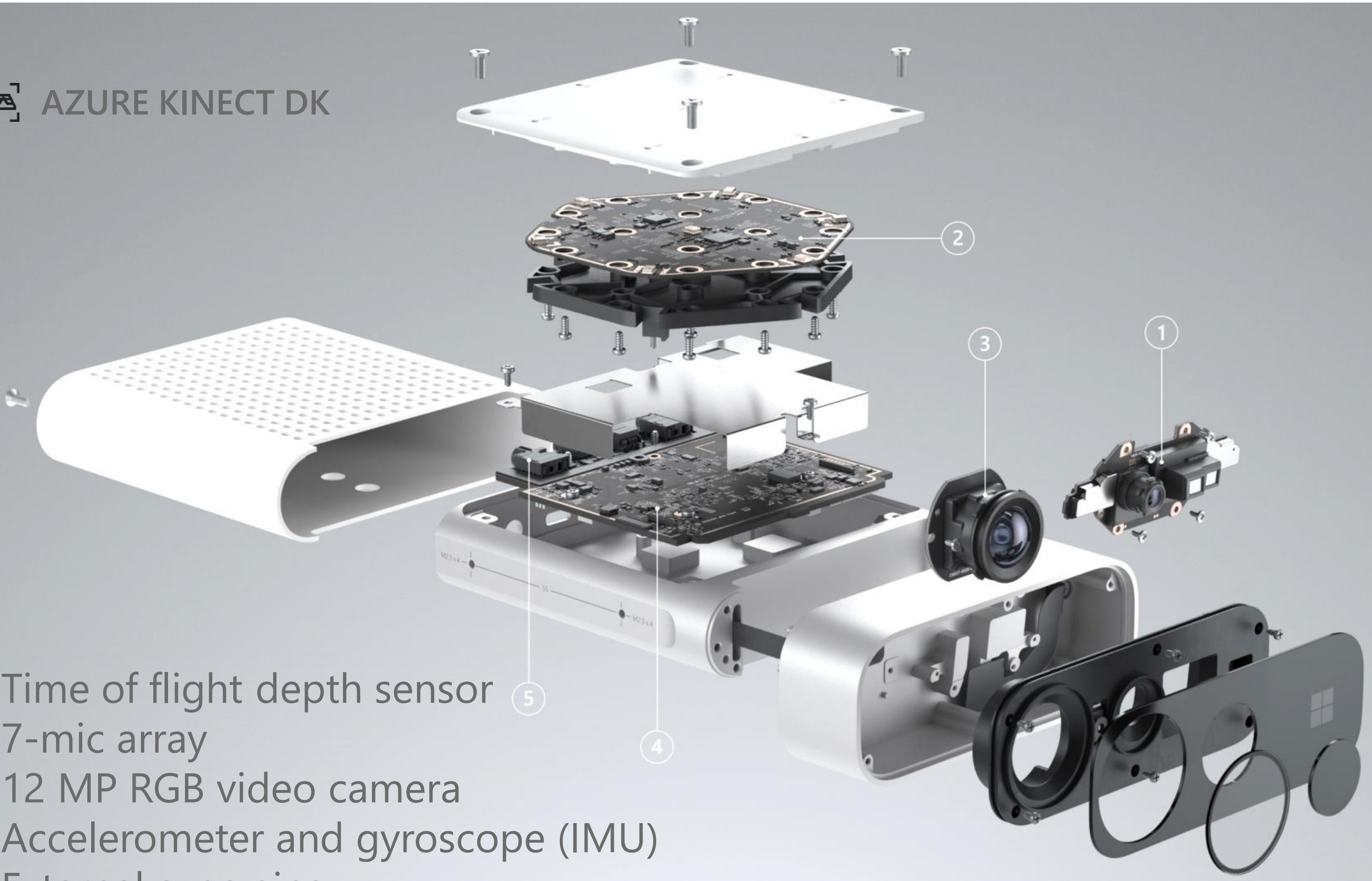
Randomized forest (Shotton et al.)



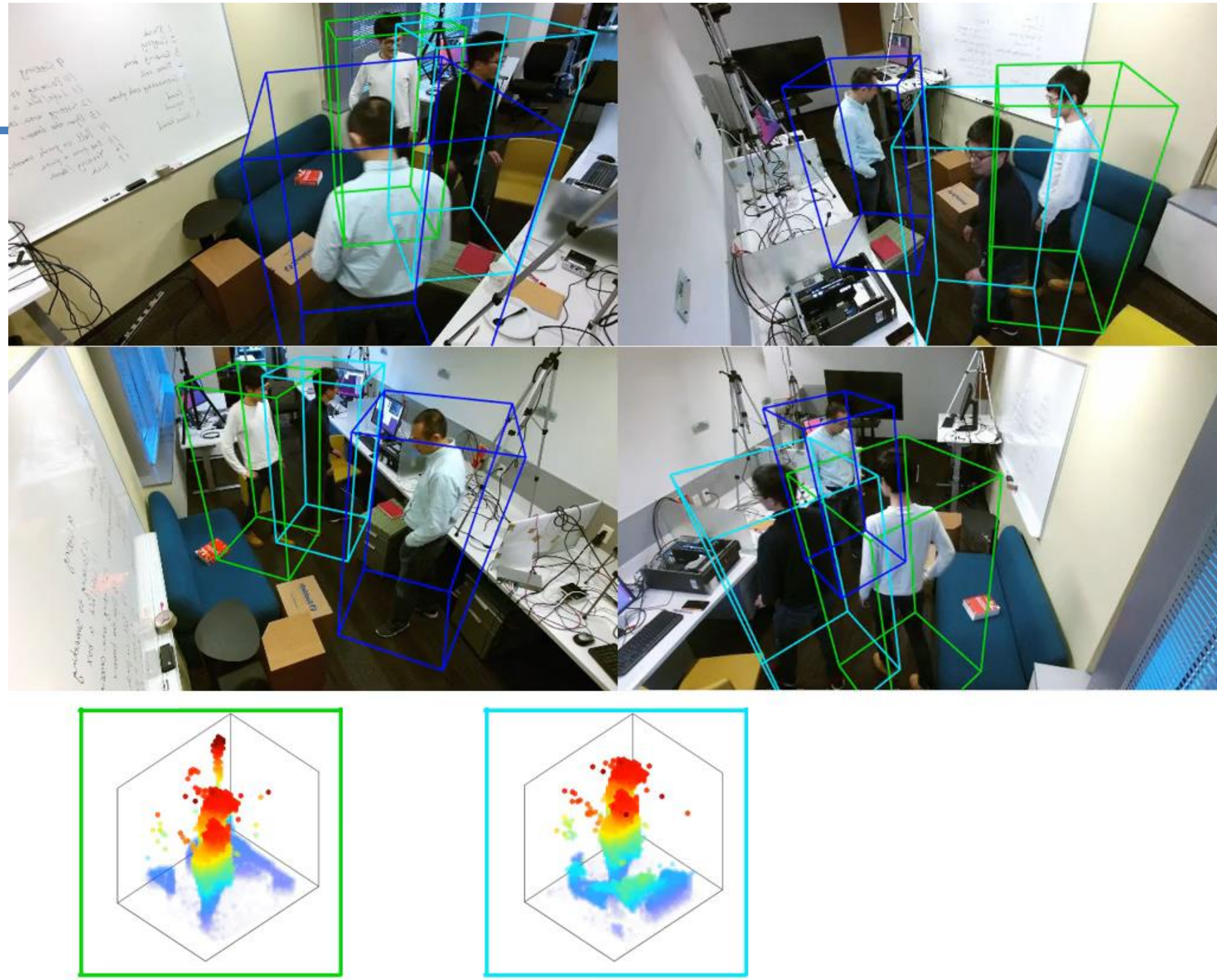
based on simple depth difference tests at each node

ETH

AZURE KINECT DK



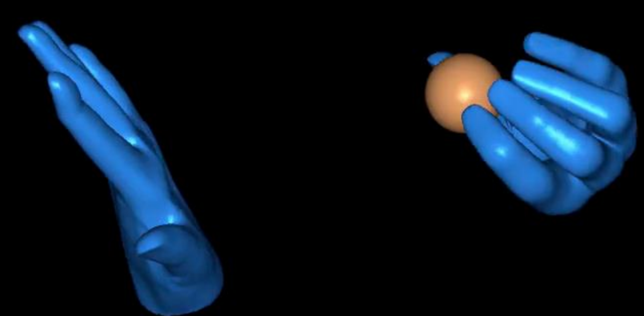
(1) Time of flight depth sensor
(2) 7-mic array
(3) 12 MP RGB video camera
(4) Accelerometer and gyroscope (IMU)
(5) External sync pins



Holding and Passing a Ball

Cam #3
(Result overlaid on Input Video)

Cam #5
(Input video)



Cam #5
(Result)

Facial animation capture

ETH

Match Moving: Joining CG + Real

ETH

Visual Computing:
The Digital Image

Prof. Marc Pollefeys
Prof. Markus Gross

ETH

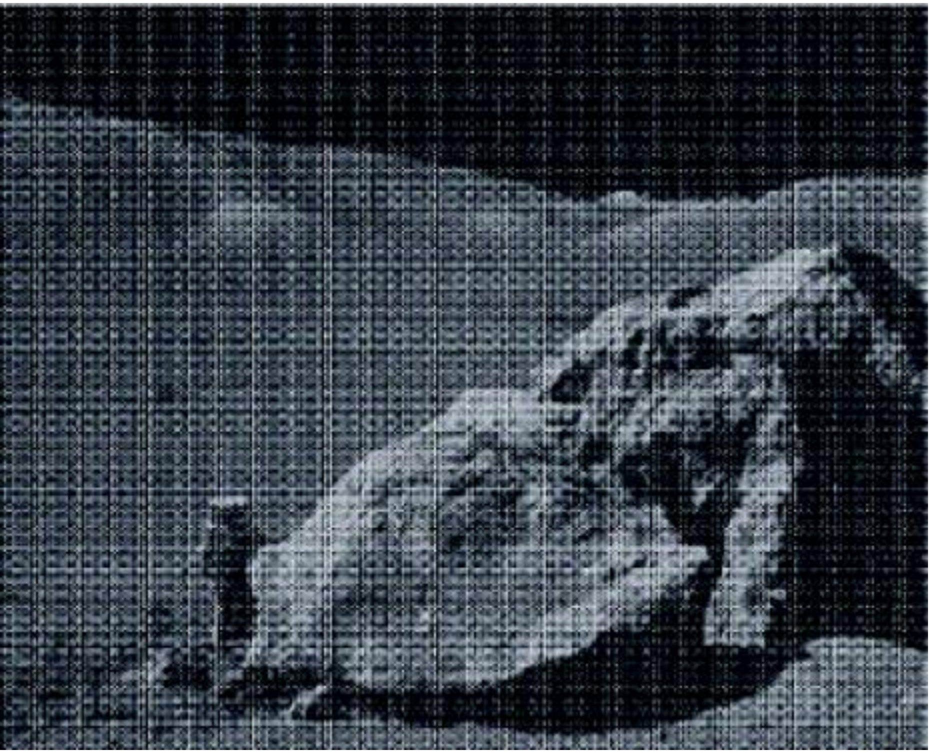
inf | Informatik
Computer Science

Digital cameras are the best sensors ever!

(Example video)

With a few problems...

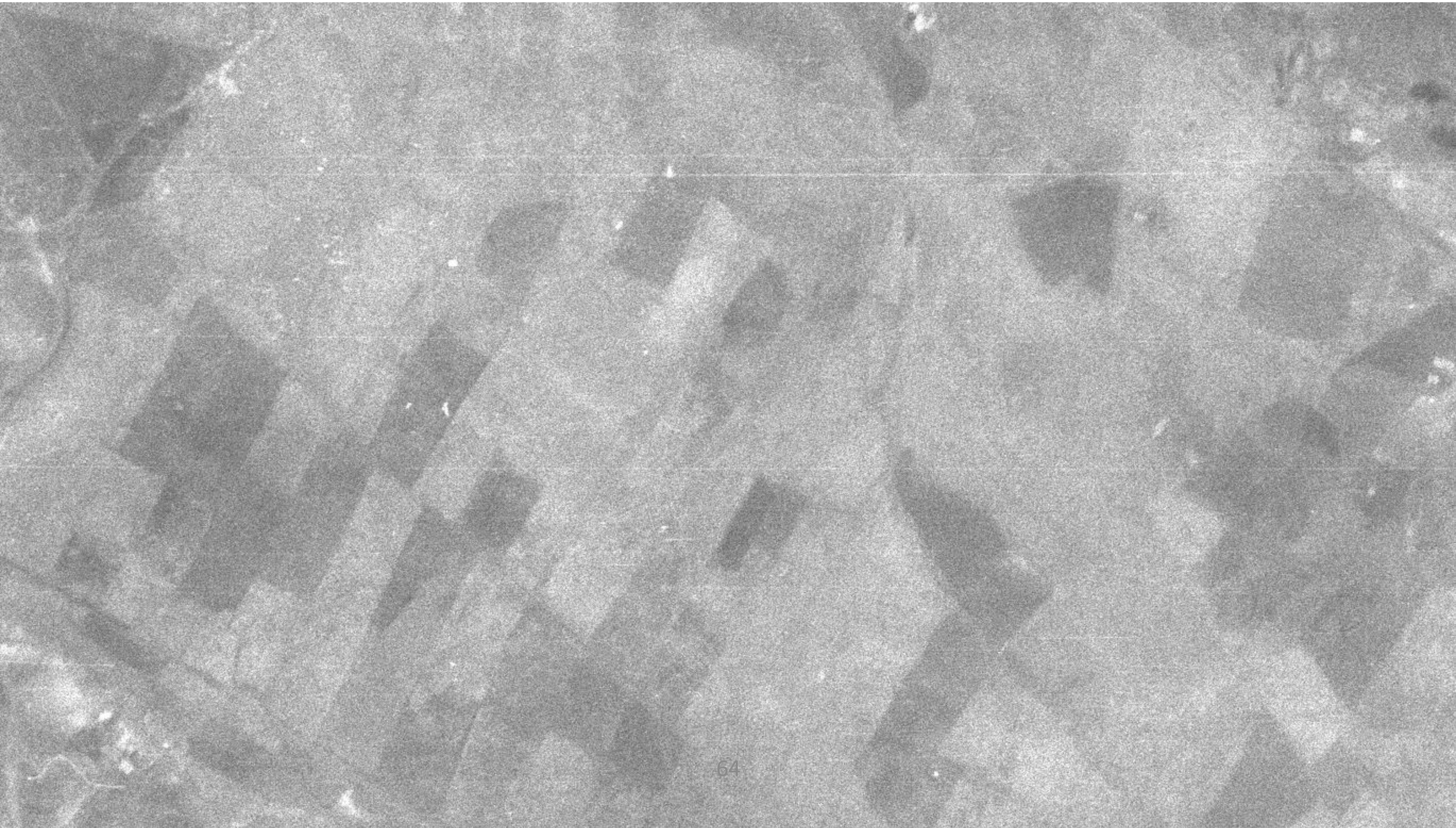
Transmission interference



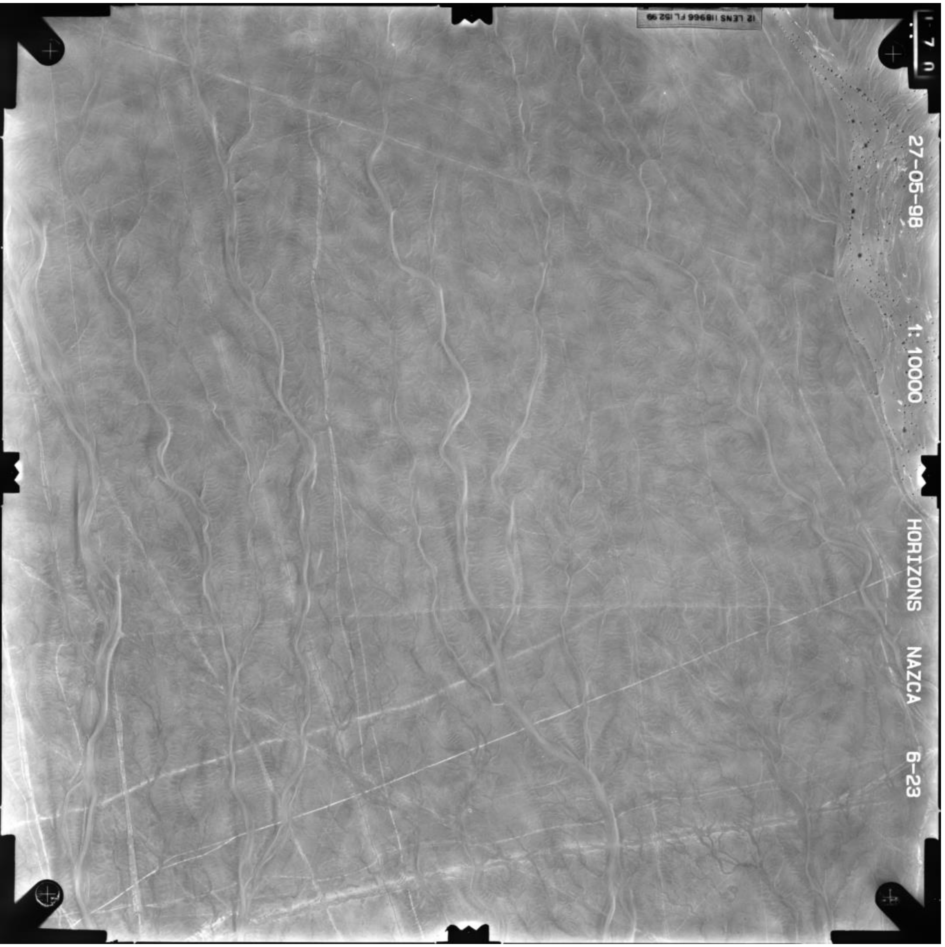
Compression artefacts

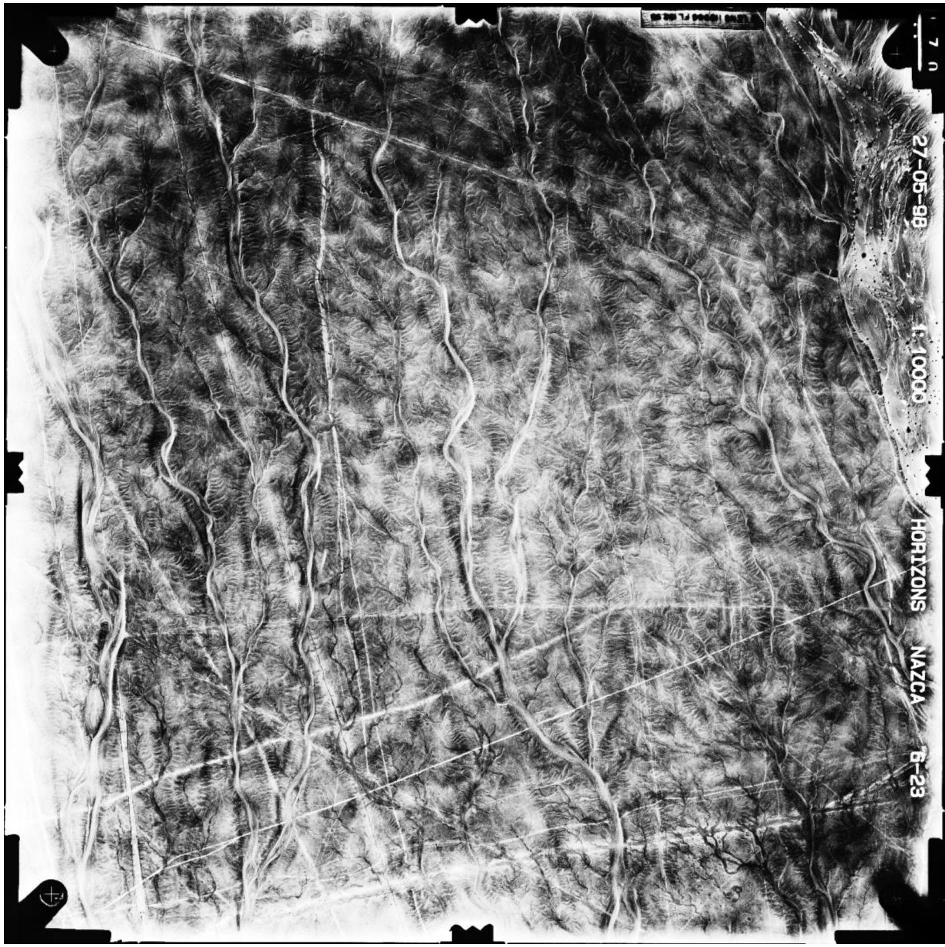
Spilling

Scratches, Sensor noise

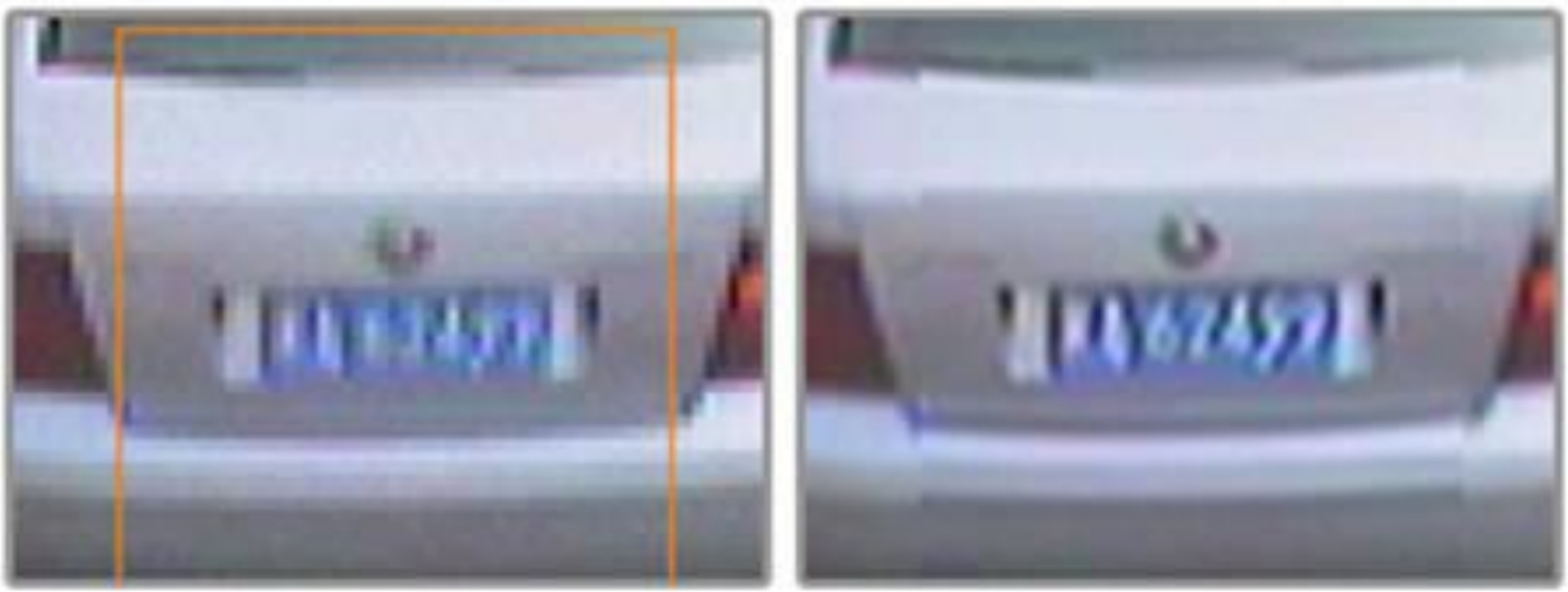


Bad contrast





Resolution → Super resolution?



Super resolution

Video

Removing motion blur

Original image

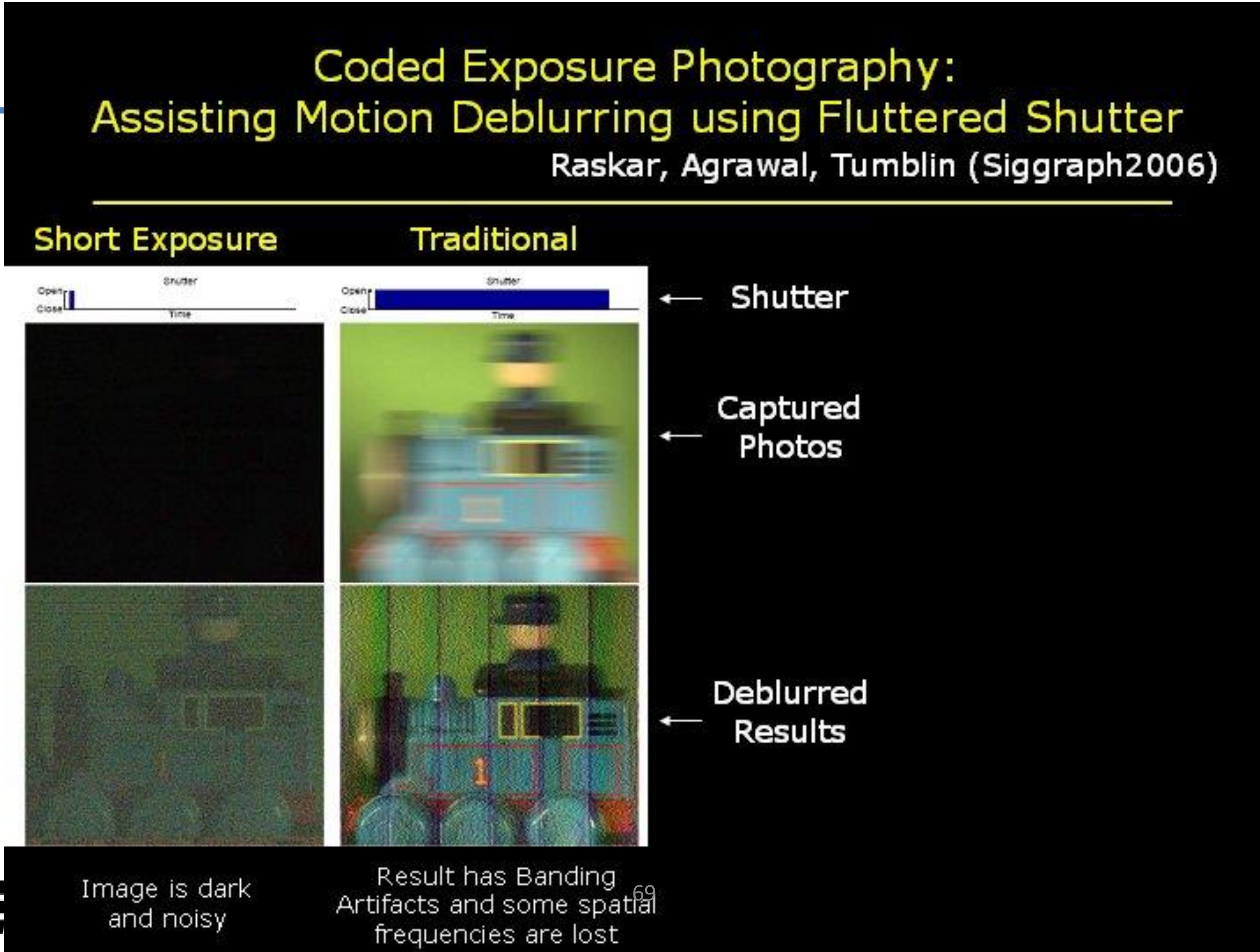
Cropped subwindow

After motion blur removal

[Images from Amit Agrawal]

ETH

Removing motion blur

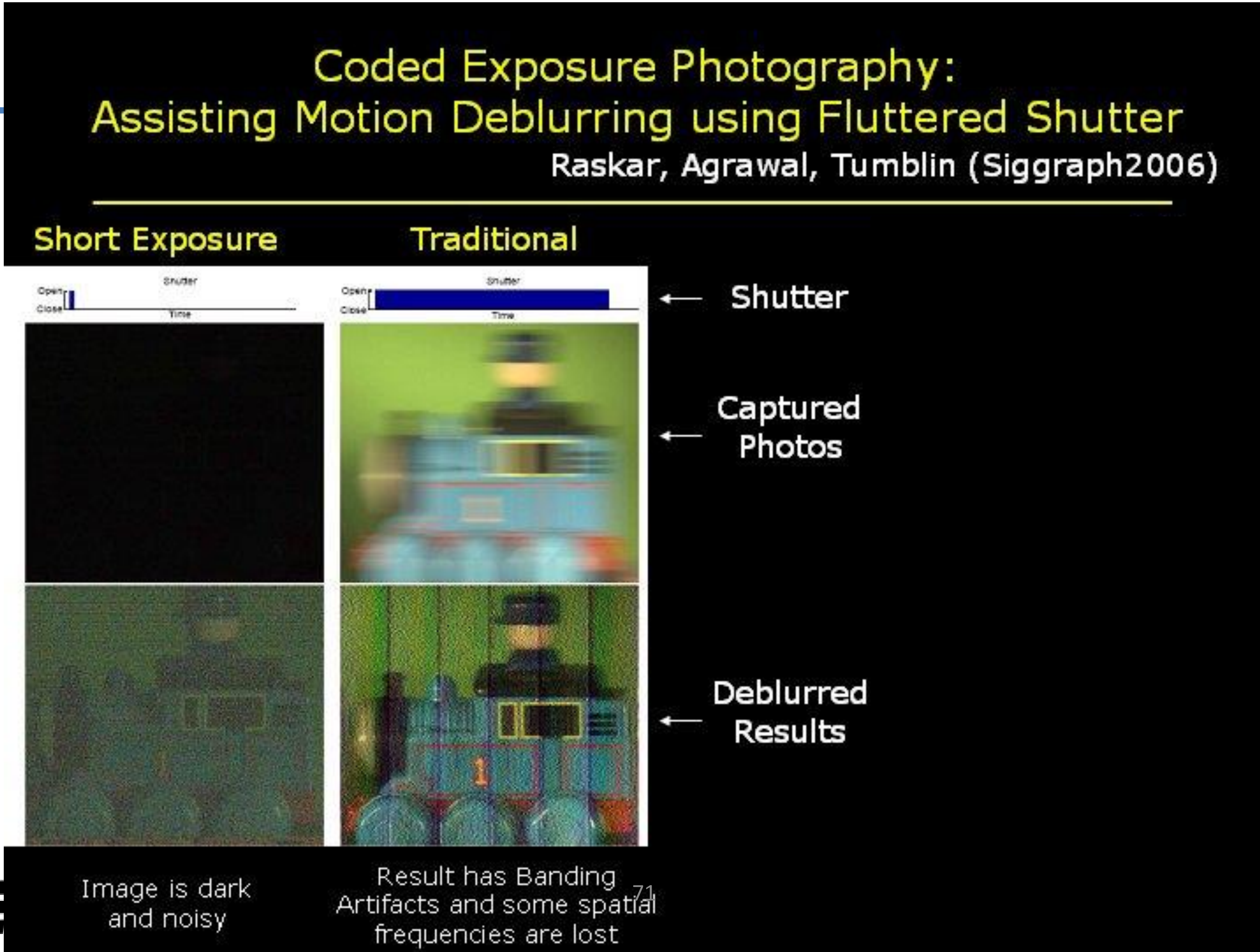


Fluttered Shutter Camera

Raskar, Agrawal, Tumblin Siggraph2006

Ferroelectric shutter in front of the lens is turned opaque or transparent in a rapid binary sequence

Removing motion blur



Matlab is Your Friend

- Start matlab:

`% matlab` or

`% /opt/matlab7/bin/matlab`

- Download any simple image

- Load it into matlab:

```
>>  I = imread('foo.jpg');
```

ETH

Unassessed Assignment

- Display the image in Matlab:

```
>> imshow(I);
```

- Print the image data array:

```
>>  I
```

- Print the size of the image array and create a subimage:

```
>> size( I )
>> Isubwindow = I(72:92, 62:82);
>> imshow(Isubwindow);
```

ETH

Unassessed Assignment

- Start the Matlab help tool (Help menu).
- In the “Contents” pane to the left of the window. Click on MATLAB.
- Go through the “Getting Started” section.
- Continue to the “Using MATLAB” section when you have time.

What is an image?

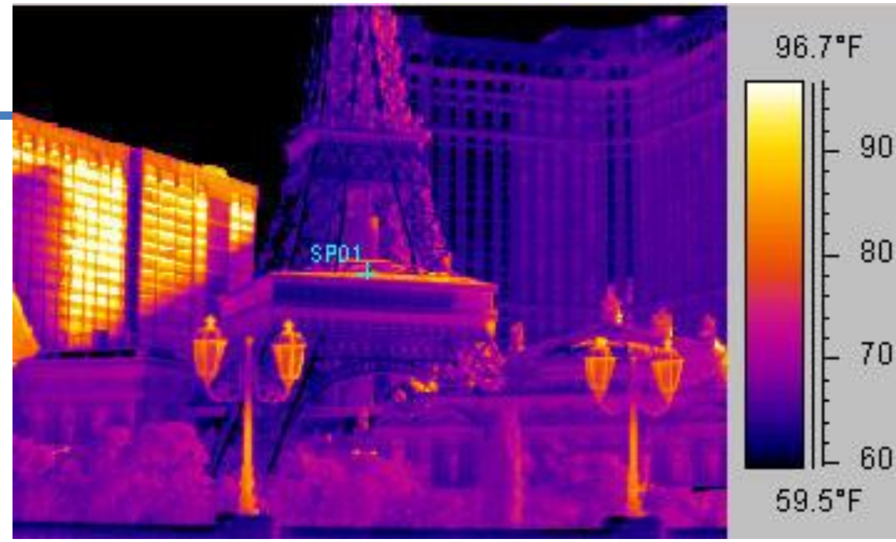
Image as 2D signal

- Signal: function depending on some variable with physical meaning
- Image: continuous function
  2 variables: xy - coordinates
  3 variables: xy + time (video)
- Brightness is usually the value of the function

- But can be other physical values too: temperature, pressure, depth …

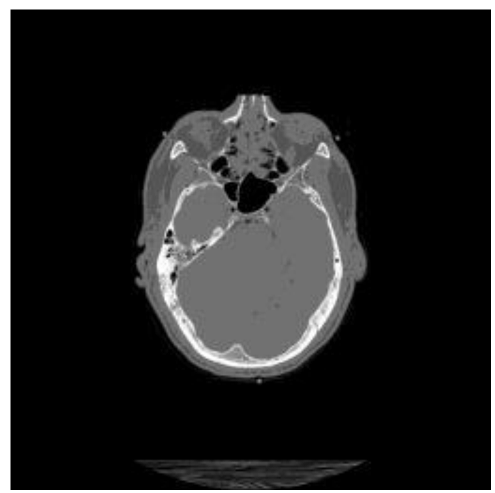
Example 2D Images

ultrasound



temperature (far IR)

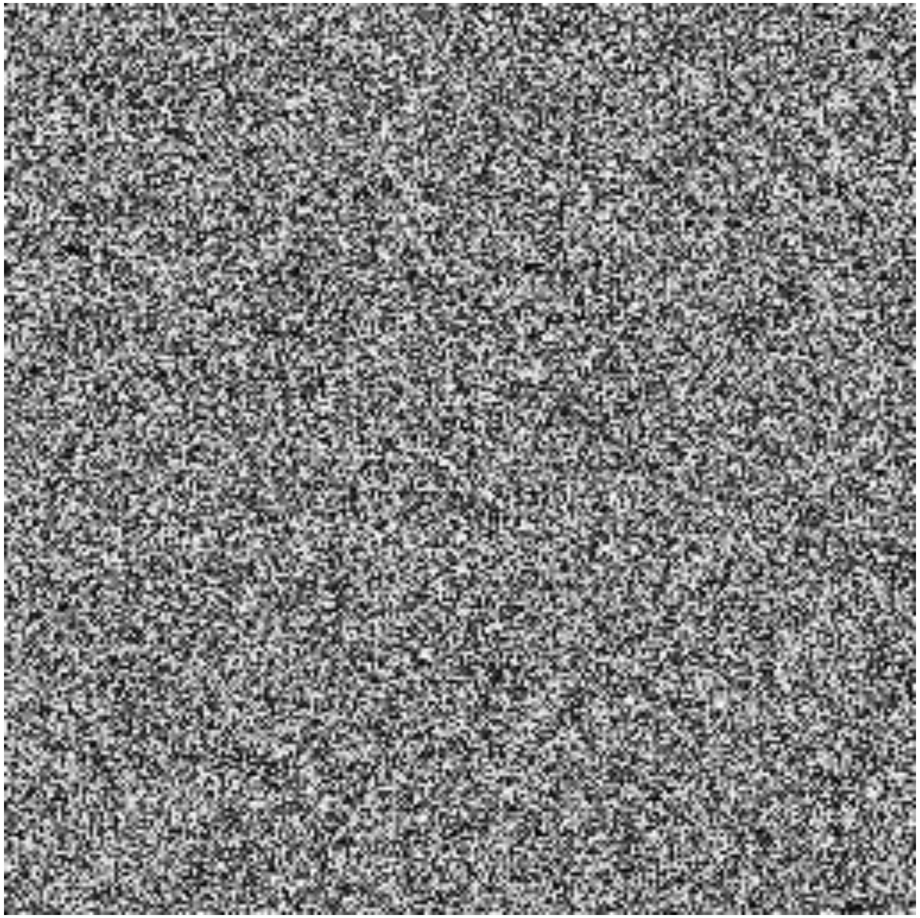
camera image



CT

Random Image

```
>> t=rand(256,256);
>> imshow(t)
```

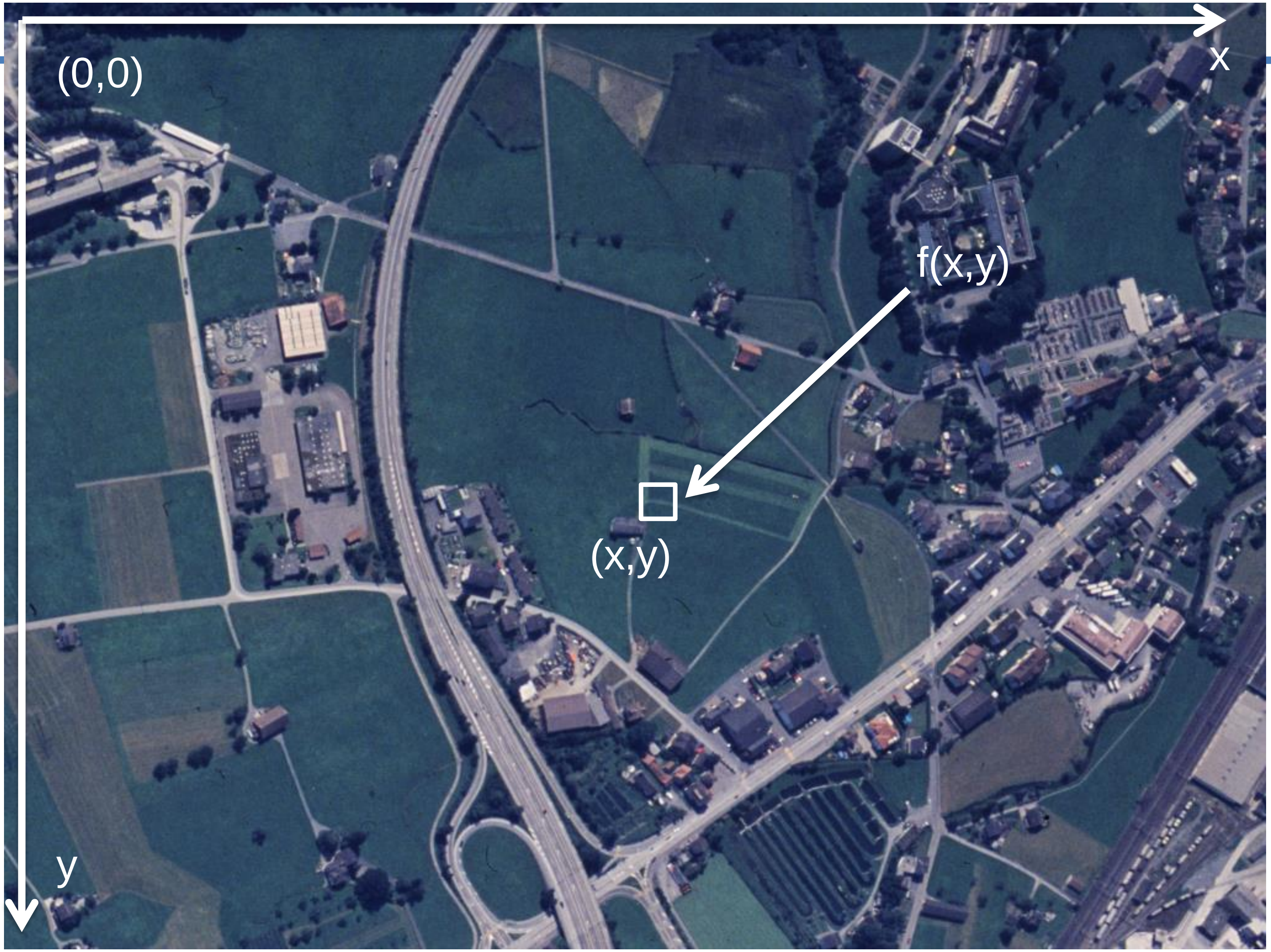


What is an image?

- A picture or pattern of a value varying in space and/or time.
- Representation of a function
- $f : \Re^n \rightarrow S$
- In digital form, eg:
- I:{1, …, X}×{1, …, Y}→S.
- For greyscale CCD images, n = 2, $S = \Re^+$.

ETH

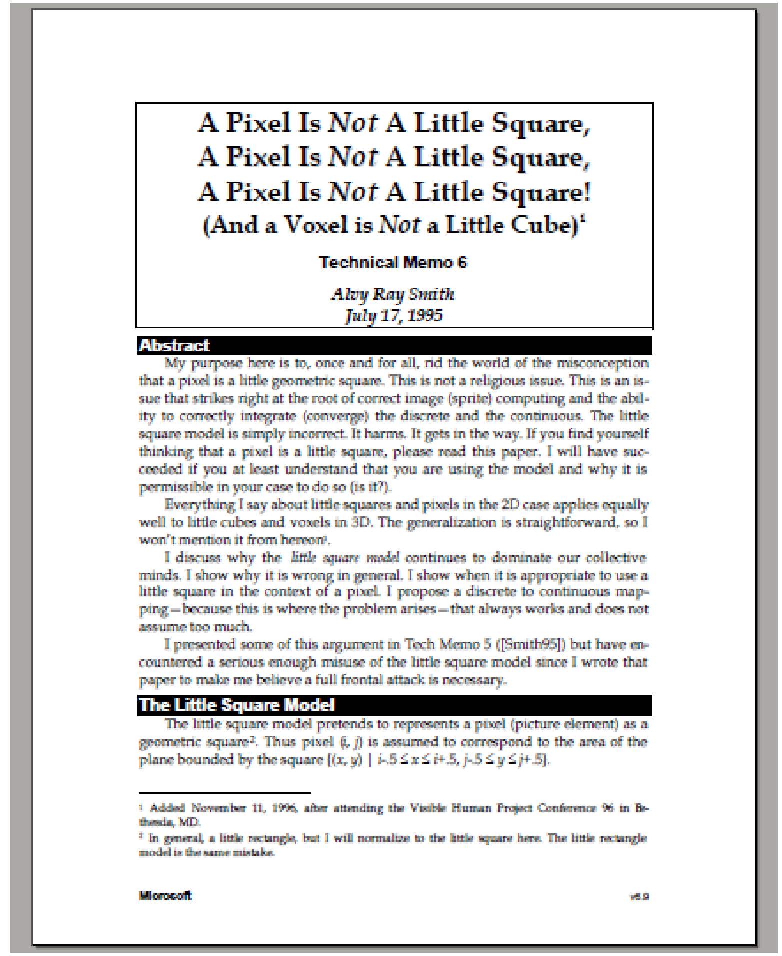
What is a pix-el?



Not a little square!

- ***A Pixel Is* Not *A Little Square, A Pixel Is* Not *A Little Square, A Pixel Is* Not *A Little Square! (And a Voxel is* Not *a Little Cube),***
  - **Alvy Ray Smith,**

  MS Tech Memo 6, Jul 17, 1995



A Pixel Is *Not* A Little Square, A Pixel Is *Not* A Little Square, A Pixel Is *Not* A Little Square! (And a Voxel is *Not* a Little Cube)[1]

Technical Memo 6

Alvy Ray Smith
July 17, 1995

Abstract

My purpose here is to, once and for all, rid the world of the misconception that a pixel is a little geometric square. This is not a religious issue. This is an issue that strikes right at the root of correct image (sprite) computing and the ability to correctly integrate (converge) the discrete and the continuous. The little square model is simply incorrect. It harms. It gets in the way. If you find yourself thinking that a pixel is a little square, please read this paper. I will have succeeded if you at least understand that you are using the model and why it is permissible in your case to do so (is it?).

Everything I say about little squares and pixels in the 2D case applies equally well to little cubes and voxels in 3D. The generalization is straightforward, so I won't mention it from hereon[1].

I discuss why the *little square model* continues to dominate our collective minds. I show why it is wrong in general. I show when it is appropriate to use a little square in the context of a pixel. I propose a discrete to continuous mapping—because this is where the problem arises—that always works and does not assume too much.

I presented some of this argument in Tech Memo 5 ([Smith95]) but have encountered a serious enough misuse of the little square model since I wrote that paper to make me believe a full frontal attack is necessary.

The Little Square Model

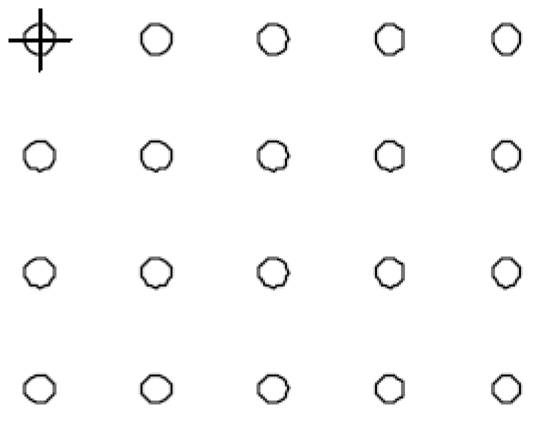
The little square model pretends to represents a pixel (picture element) as a geometric square[2]. Thus pixel (i, j) is assumed to correspond to the area of the plane bounded by the square $\{(x, y) \mid i\text{-}.5 \le x \le i\text{+}.5, j\text{-}.5 \le y \le j\text{+}.5\}$.

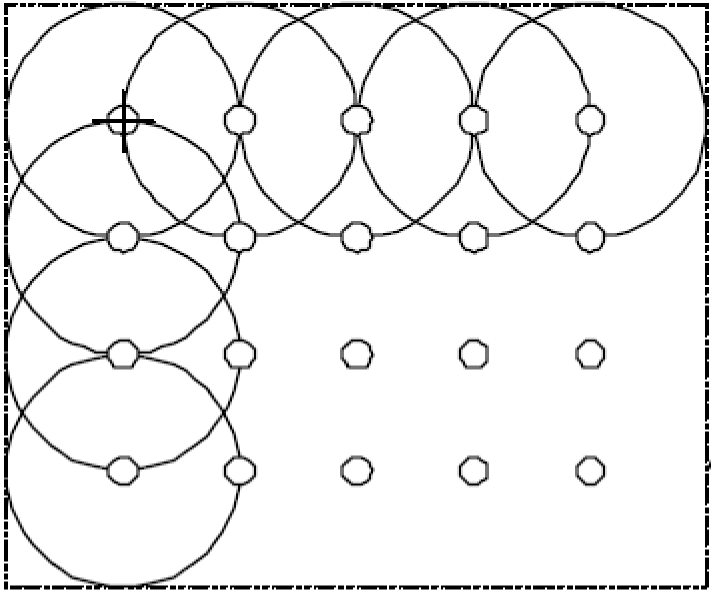
[1] Added November 11, 1996, after attending the Visible Human Project Conference 96 in Bethesda, MD.

[2] In general, a little rectangle, but I will normalize to the little square here. The little rectangle model is the same mistake.

Microsoft v5.9

Not a little square!

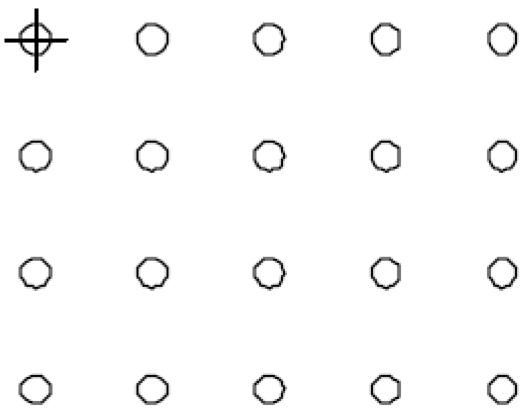


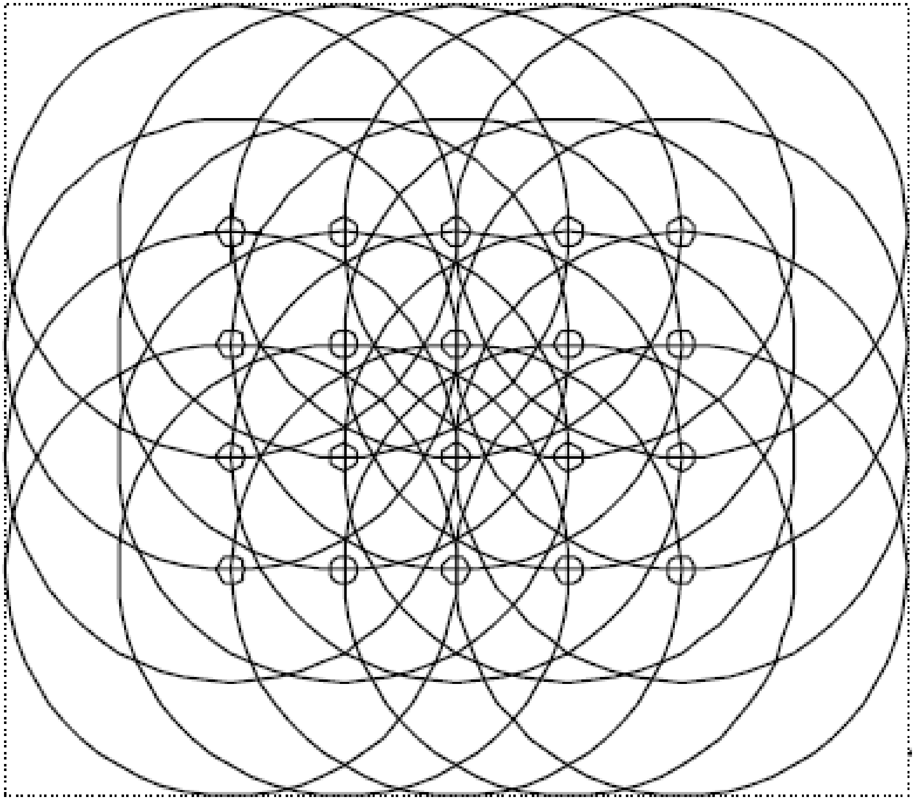


Gaussian reconstruction filter

Illustrations: Smith, MS Tech Memo 6, Jul 17, 1995

Not a little square!





Cubic reconstruction filter

Illustrations: Smith, MS Tech Memo 6, Jul 17, 1995

Not a little square!

Graphics: Dick Lyon, 2006

Where do images come from?

- Digital cameras
- MRI scanners
- Computer graphics packages
- Body scanners
- Laser range finders

- Many more…

ETH

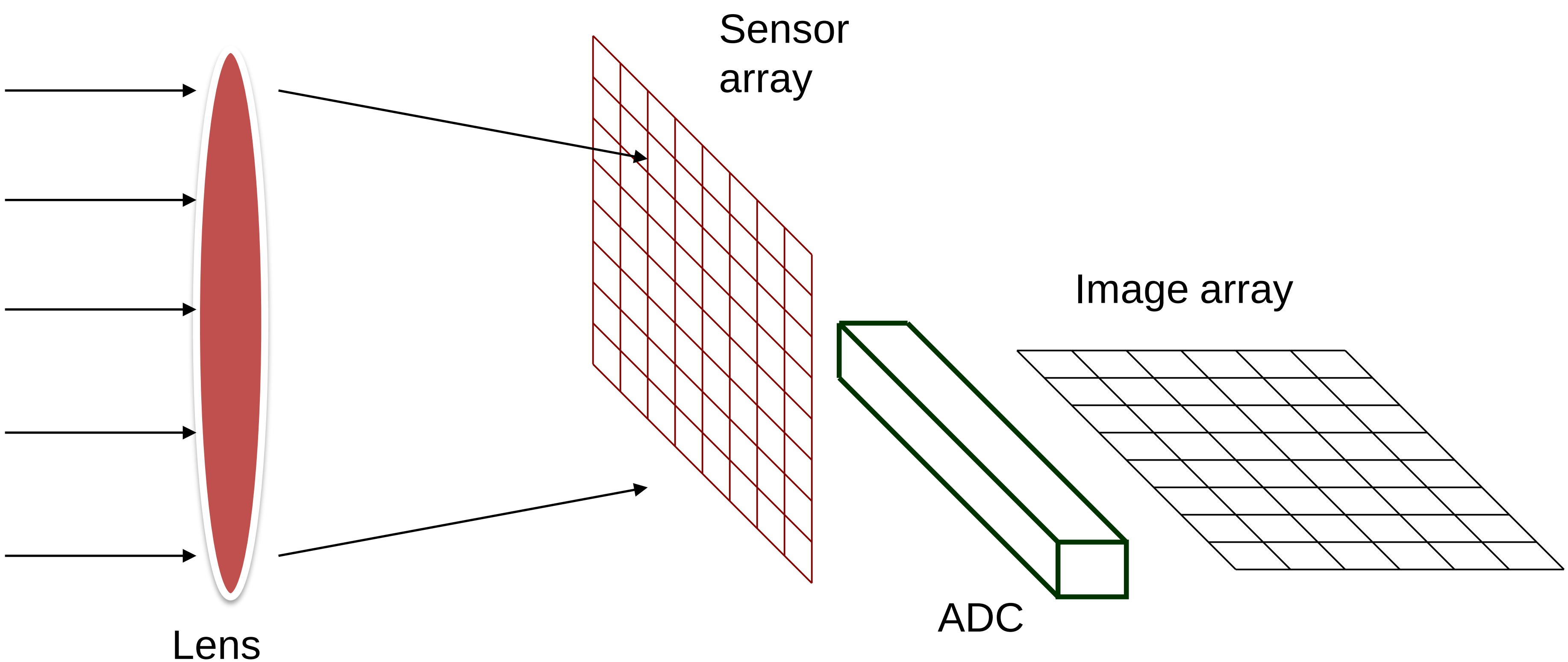
Where do images come from?

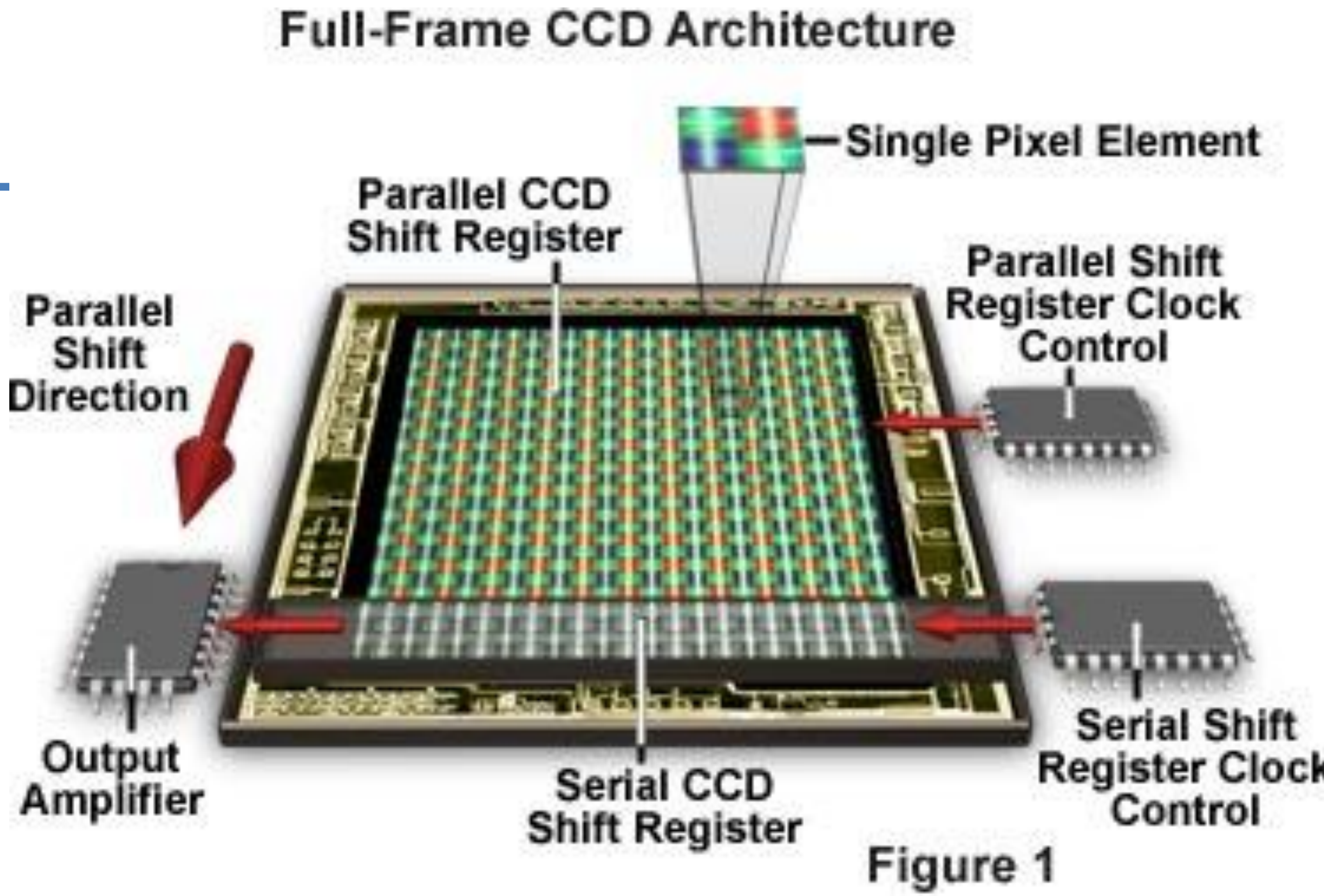
- **Digital cameras**
- MRI scanners
- Computer graphics packages
- Body scanners
- Laser range finders

- Many more…

The digital camera

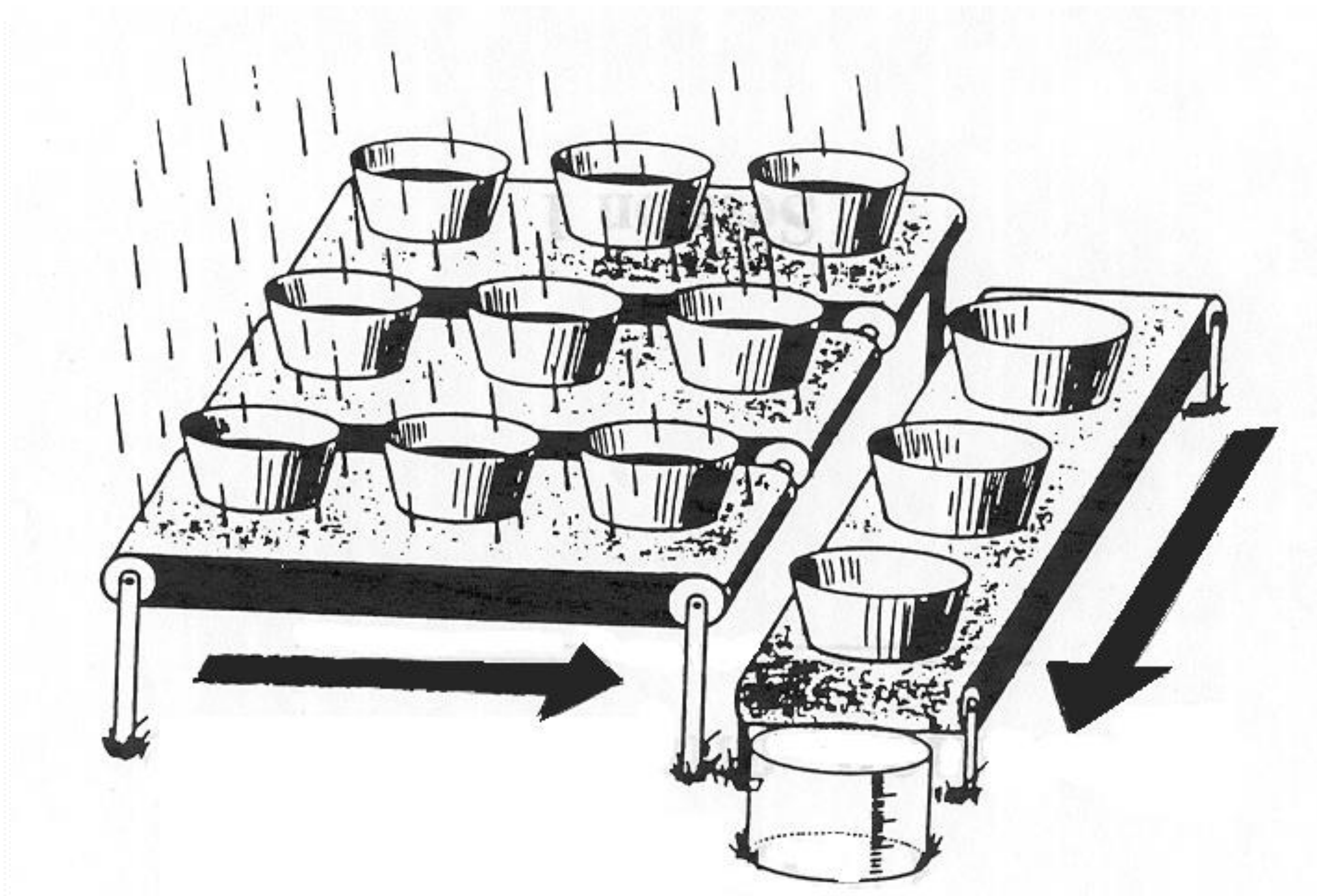
- A *Charge Coupled Device* (CCD).





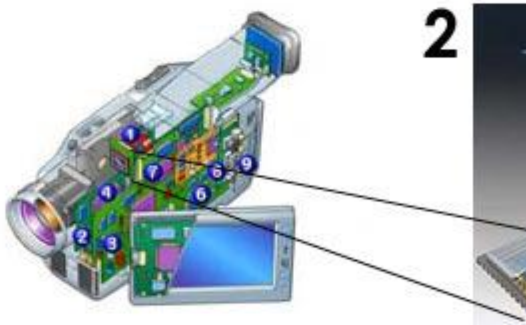
http://www.astro.virginia.edu/class/oconnell/astr121/im/CCD-fullframearc-FSU.jpg

Capturing photons

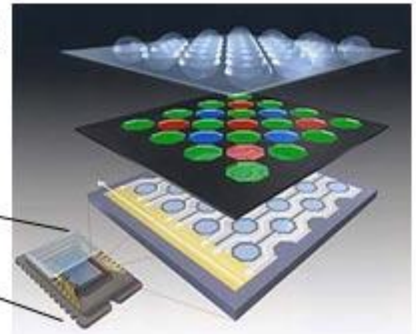


[From: Lecture Notes – EAAE and/or Science "Nuggets" 2000]

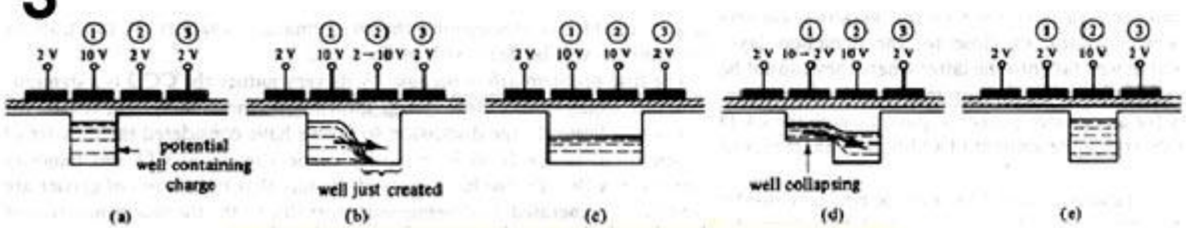
1



2



3

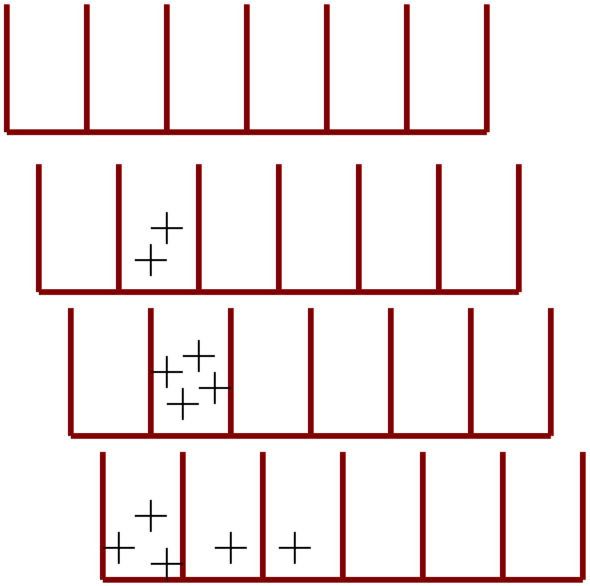


4

http://www.science.ca/images/scientists/s-boyle-infographic.jpg

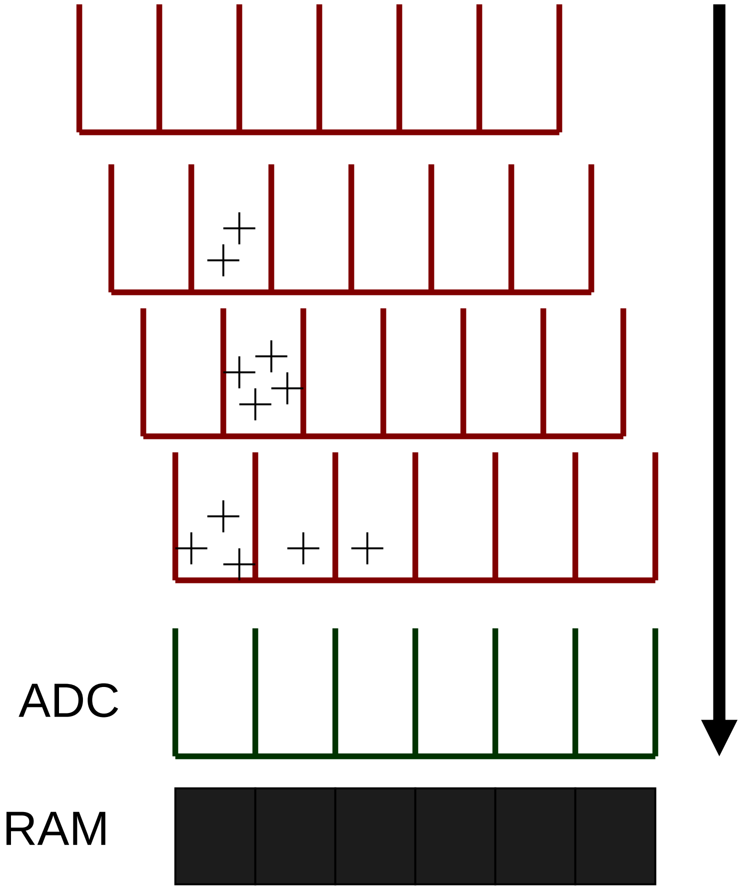
The sensor array

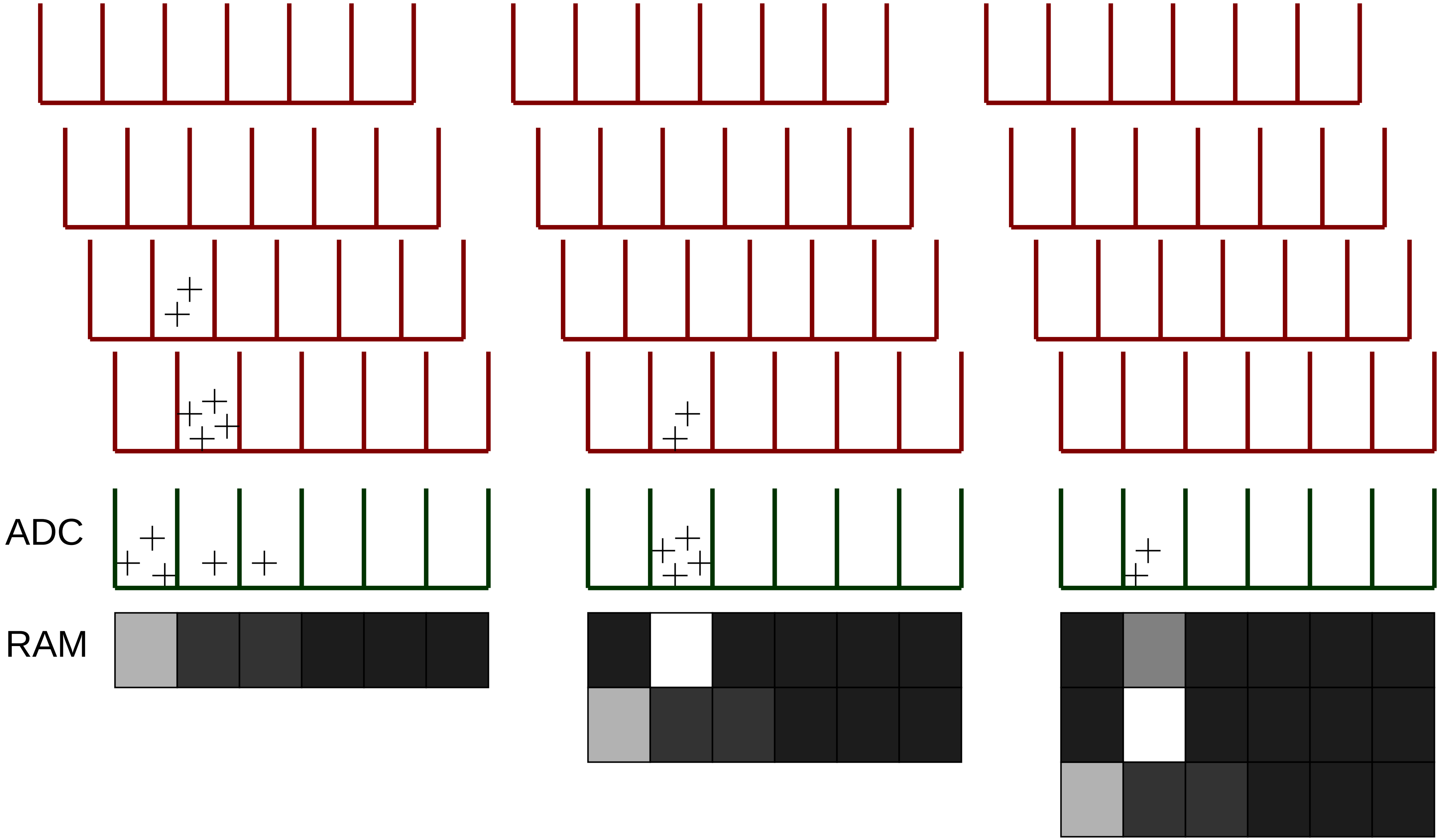
- Can be $< 1cm^2$.
- An array of *photosites*.
- Each photosite is a bucket of electrical charge.
- They contain charge proportional to the incident light intensity during exposure.



Analog to Digital Conversion

- The ADC measures the charge and digitizes the result.
- Conversion happens line by line.
- The charges in each photosite move down through the sensor array.



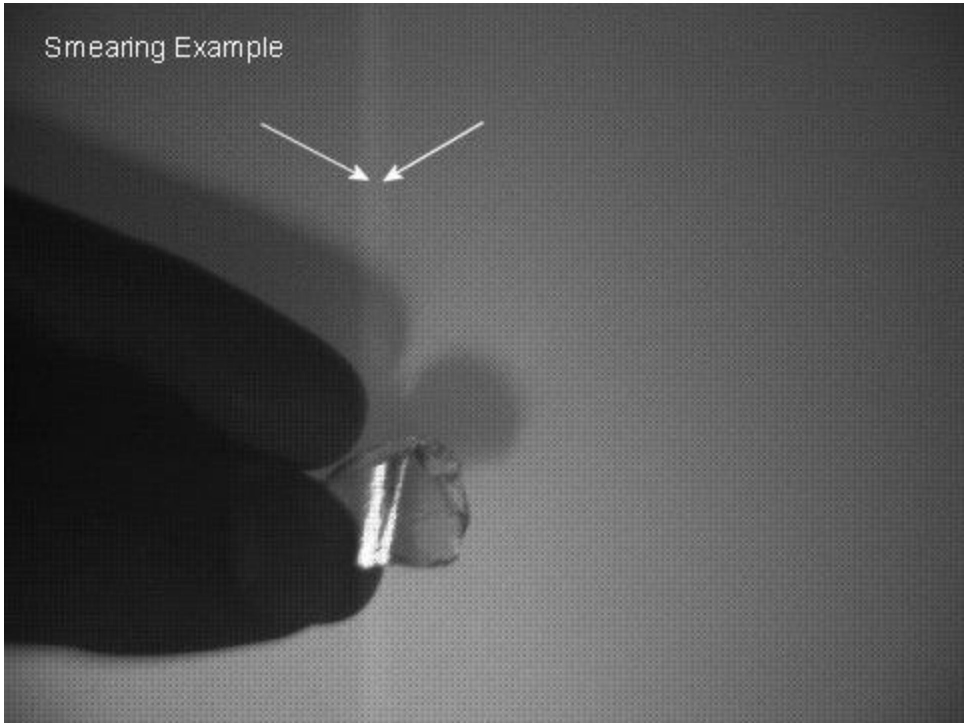

ADC
RAM

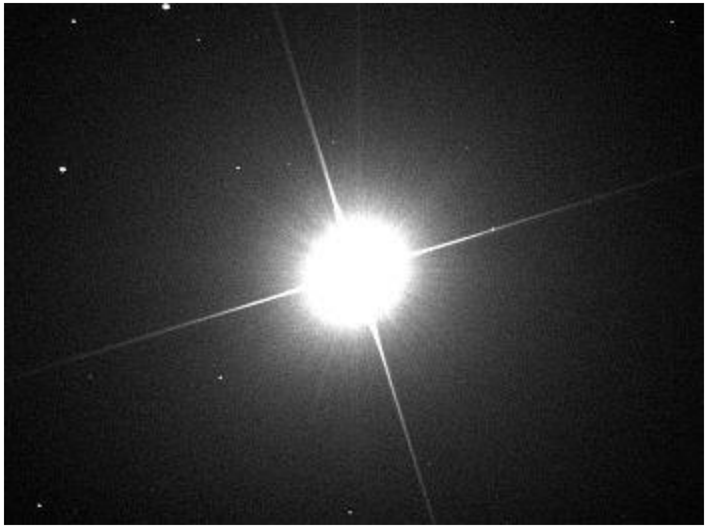
Blooming

- The buckets have finite capacity
- Photosite saturation causes blooming

ETH

Bleeding or smearing

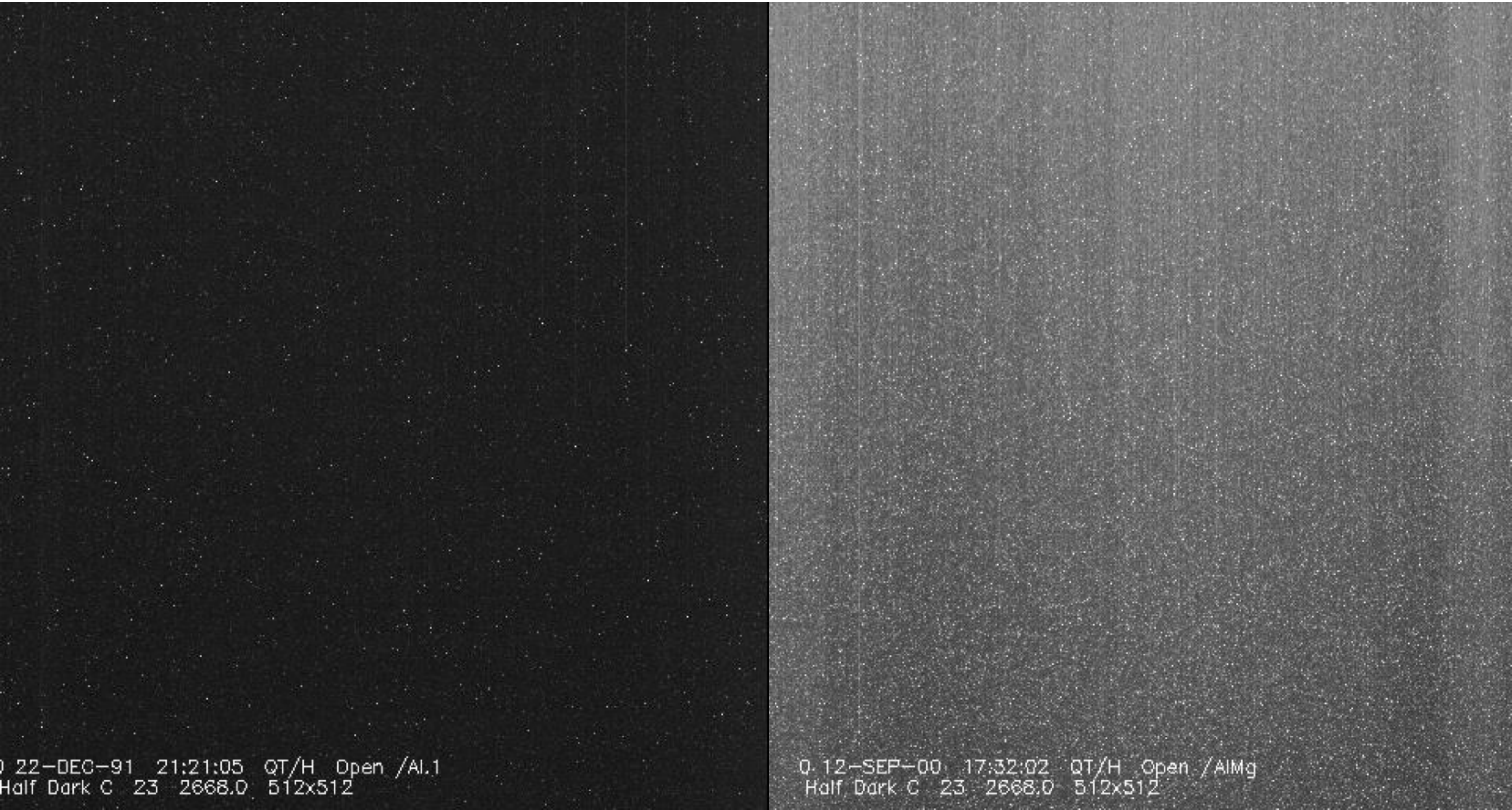




During transit buckets still accumulate some charges
Influenced by time ‘in transit’ versus integration time
Effect is worse for short shutter times (only problem with electronic shutter)

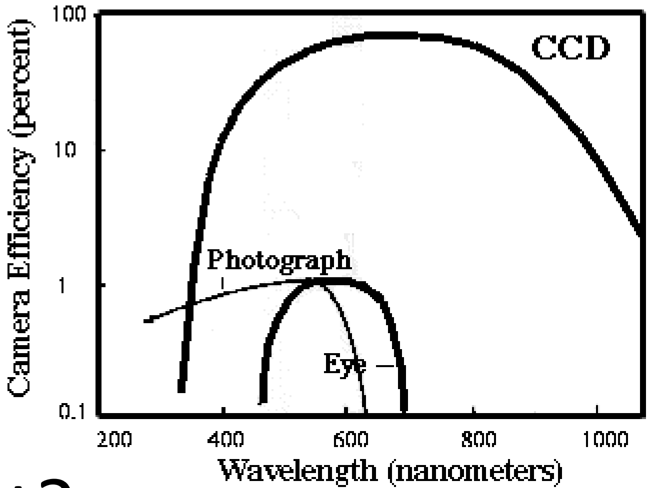
Dark Current

Yohkoh satellite, 9 years apart ..



Dark Current

- CCDs produce thermally-generated charge.
- They give non-zero output even in darkness.
- Partly, this is the *dark current*.
- Fluctuates randomly.



From: Lecture Notes - EAAE

- How can we reduce dark current?

CMOS

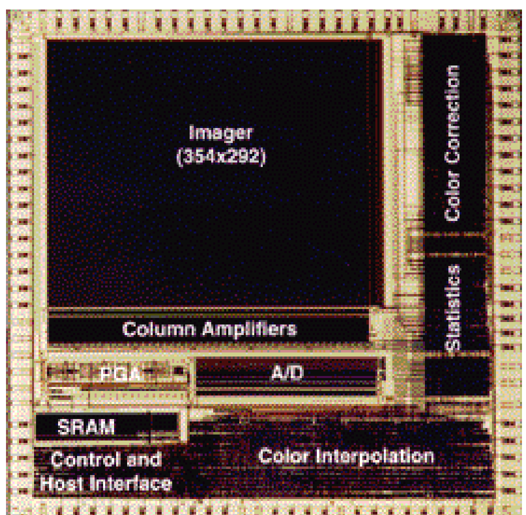
Same sensor elements as CCD

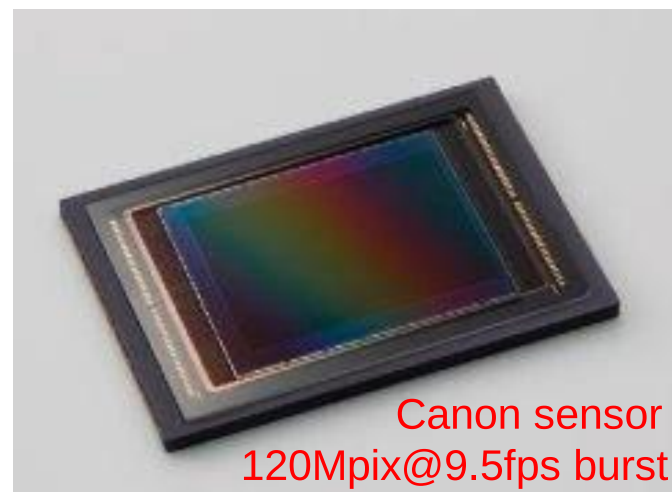
Each photo sensor has its own amplifier

- More noise (reduced by subtracting 'black' image)
- Lower sensitivity (lower fill rate)

Uses standard CMOS technology

- Allows to put other components on chip
- 'Smart' pixels





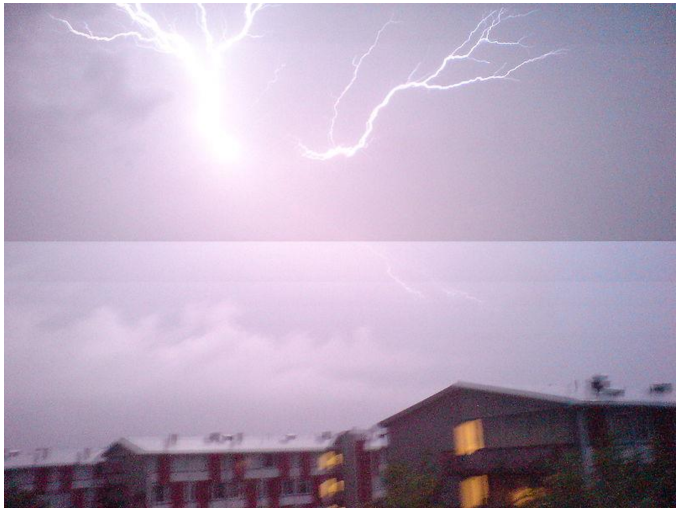
CCD vs. CMOS

- Mature technology
- Specific technology
- High production cost
- High power consumption
- Higher fill rate
- Blooming
- Sequential readout

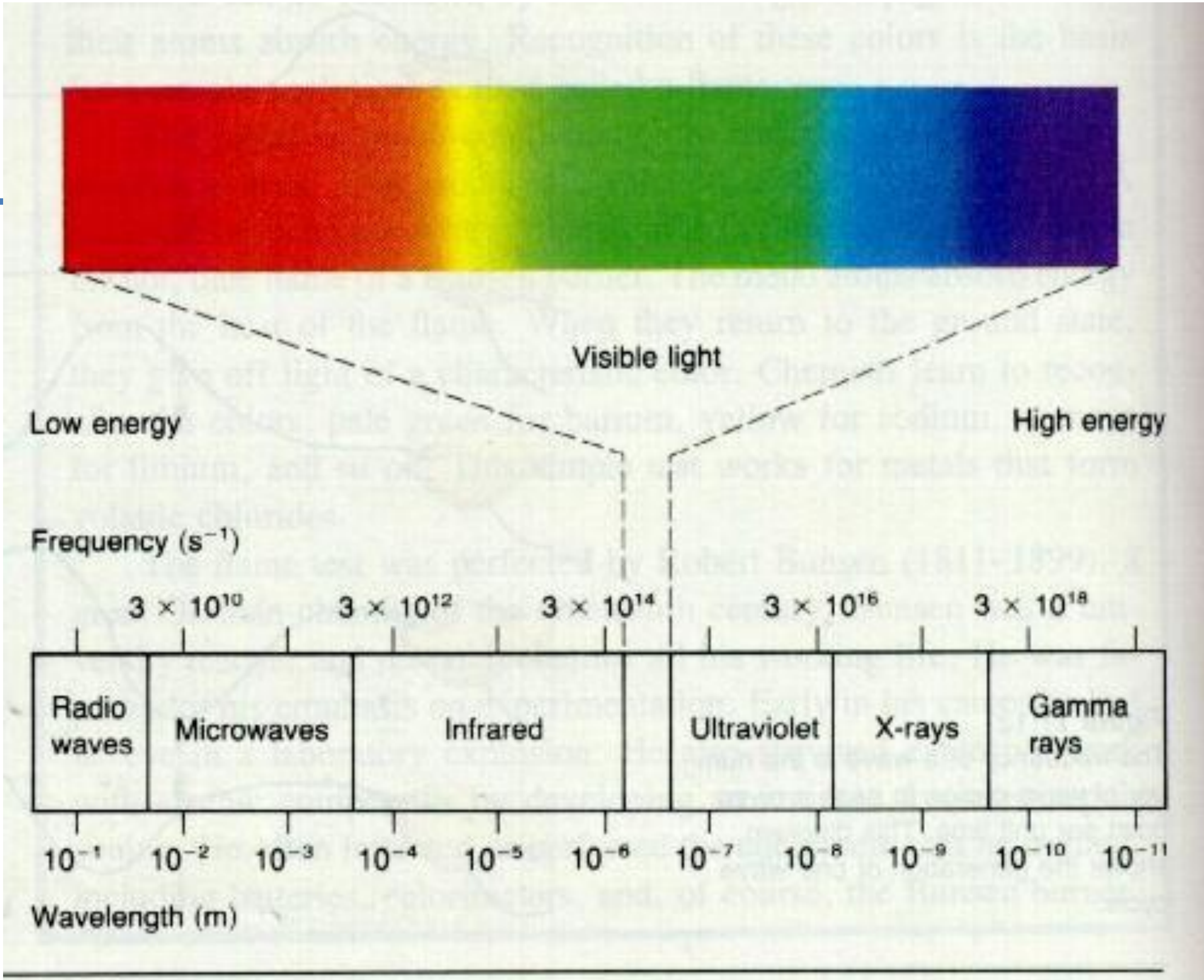
- Recent technology
- Standard IC technology
- Cheap
- Low power
- Less sensitive
- Per pixel amplification
- Random pixel access
- Smart pixels
- On chip integration with other components

CMOS video sensor issues

- Rolling shutter
  - Sequential read-out of lines



video


Visible light
Low energy
High energy
Frequency (s⁻¹)
3 × 10¹⁰
3 × 10¹²
3 × 10¹⁴
3 × 10¹⁶
3 × 10¹⁸
Radio waves
Microwaves
Infrared
Ultraviolet
X-rays
Gamma rays
10⁻¹
10⁻²
10⁻³
10⁻⁴
10⁻⁵
10⁻⁶
10⁻⁷
10⁻⁸
10⁻⁹
10⁻¹⁰
10⁻¹¹
Wavelength (m)

- Color sensing from Computational Photography?

… and much more
see you Thursday!