Introduction to Computer Vision

Michael J. Black

Sept 2009

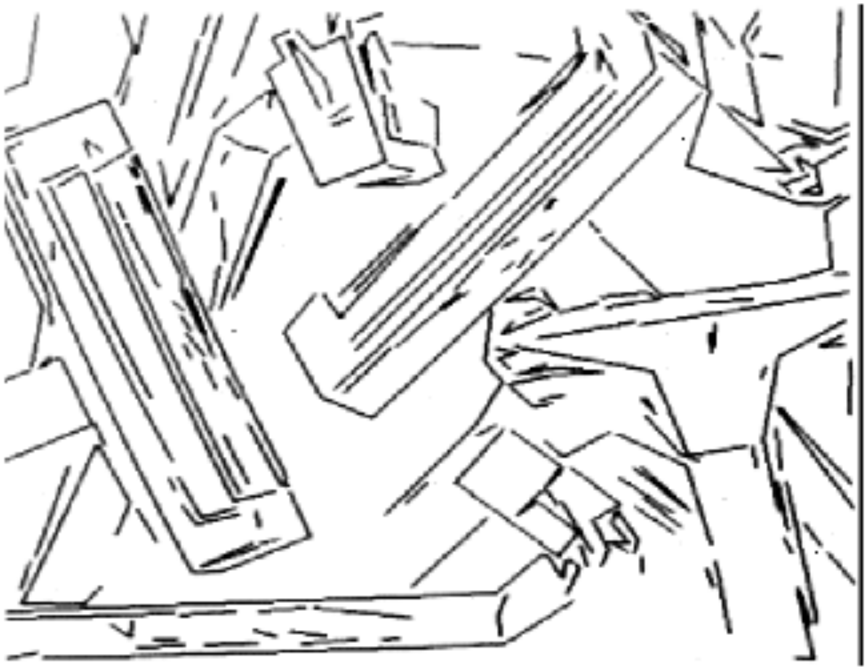
Lecture 6: Introduction (conclusion) and intro to linear filtering

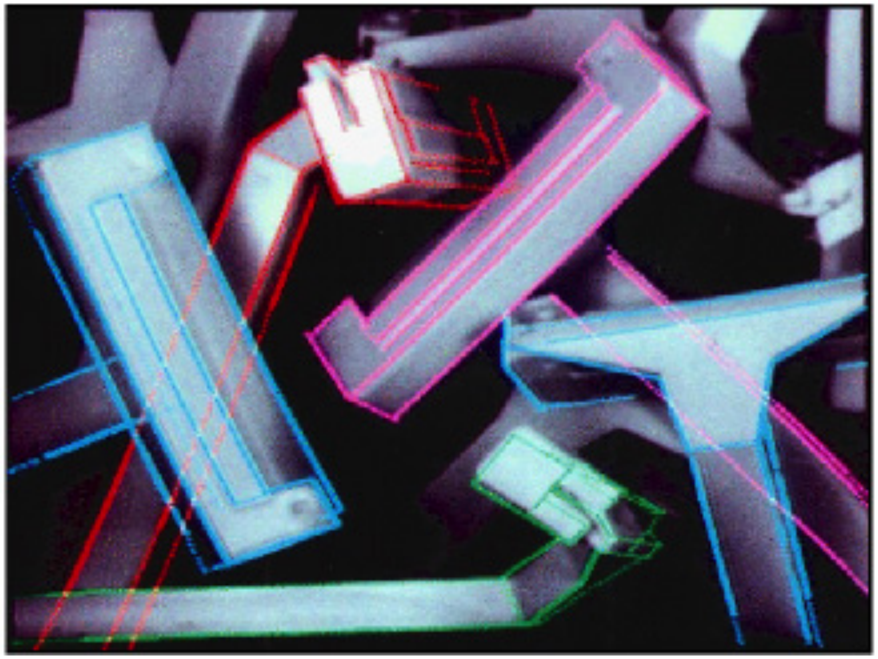
©Michael J. Black

Goals

- Really finish intro.
- Start linear filtering
  - Foundations for asng1.
    - Problem 1

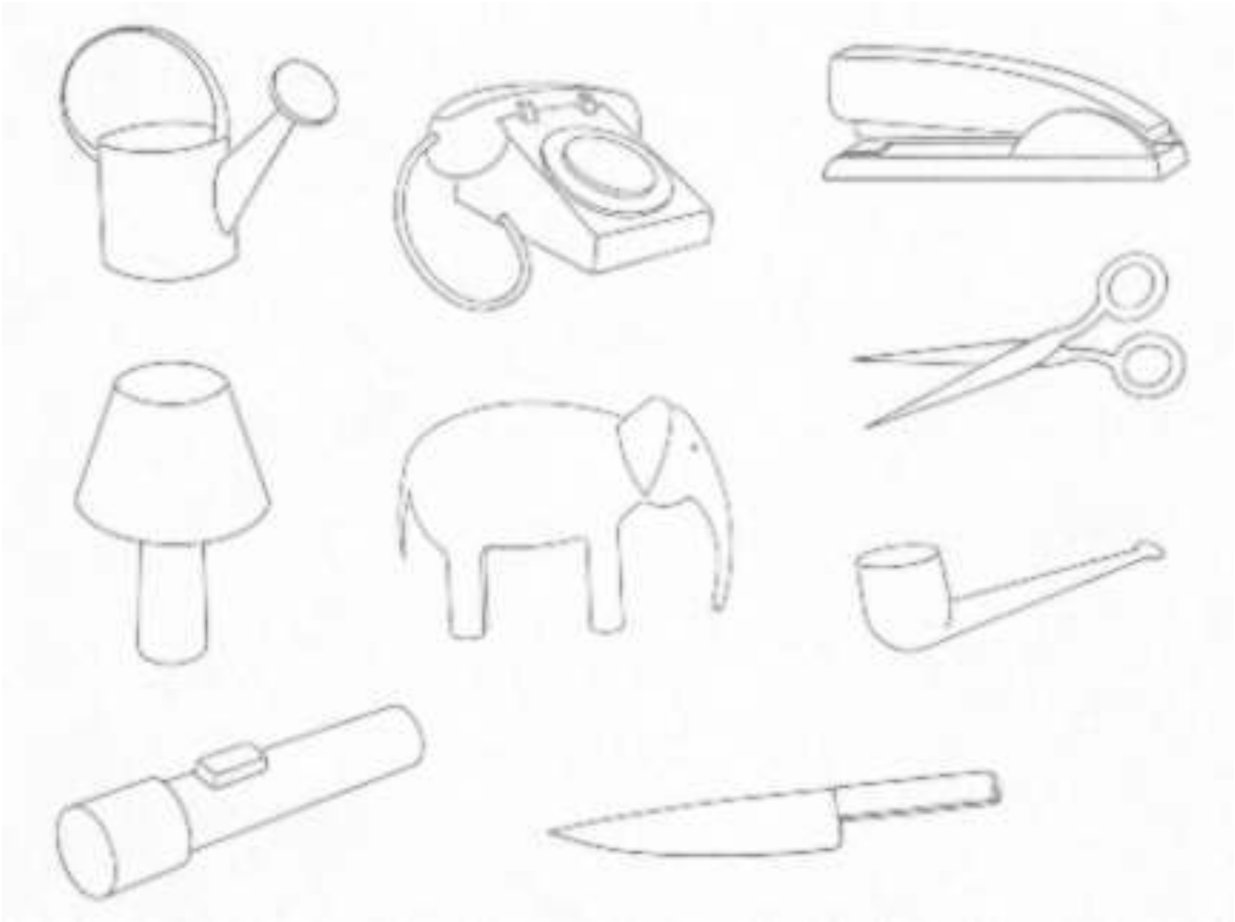
Object detection





David Lowe

"Project" model into image and "match" to lines (solving for 3D pose).



Biederman’s Geons

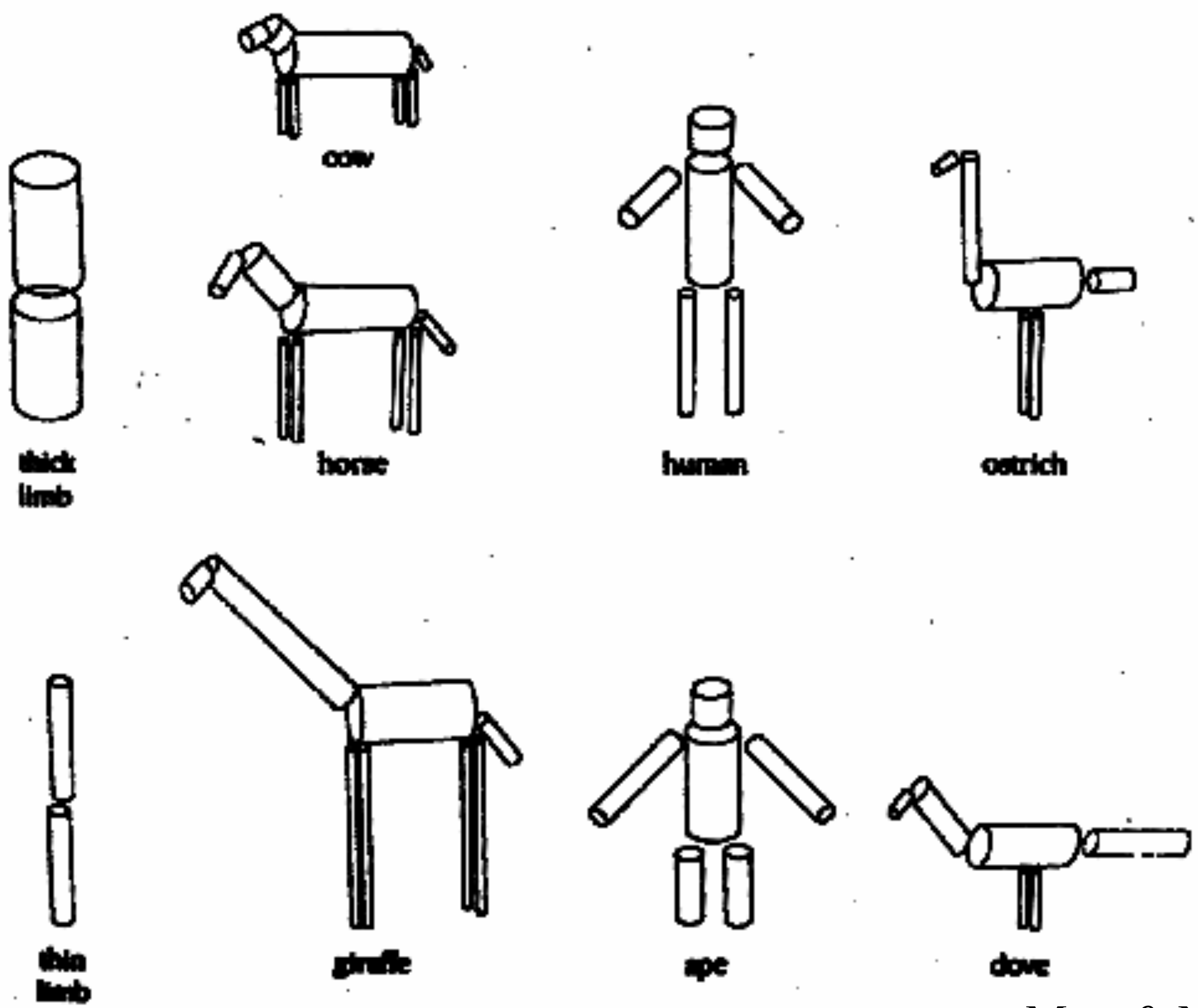
Possible approach: If line drawings are easy to recognize then maybe we should first find lines.

©Michael J. Black

Challenges 7: intra-class variation

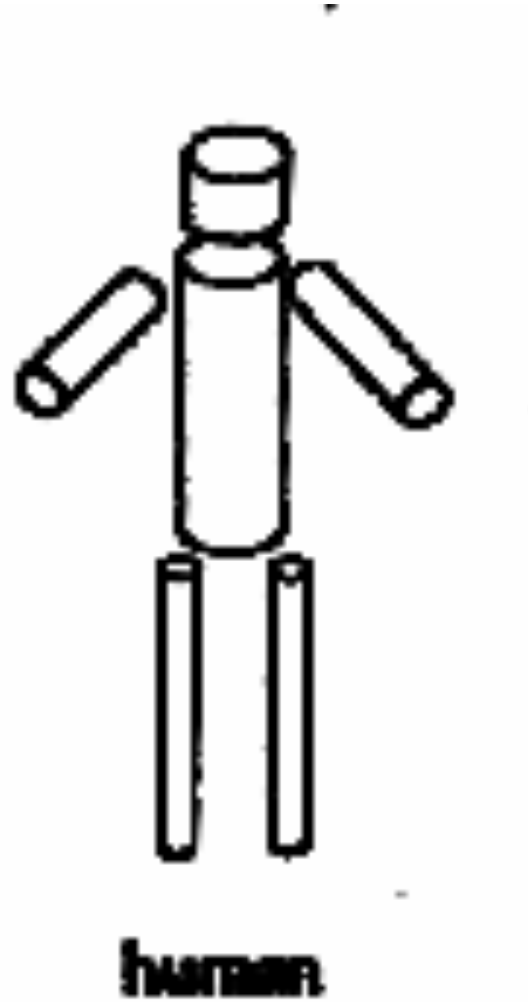
Fei-Fei Li.

©Michael J. Black



Marr & Nishihara

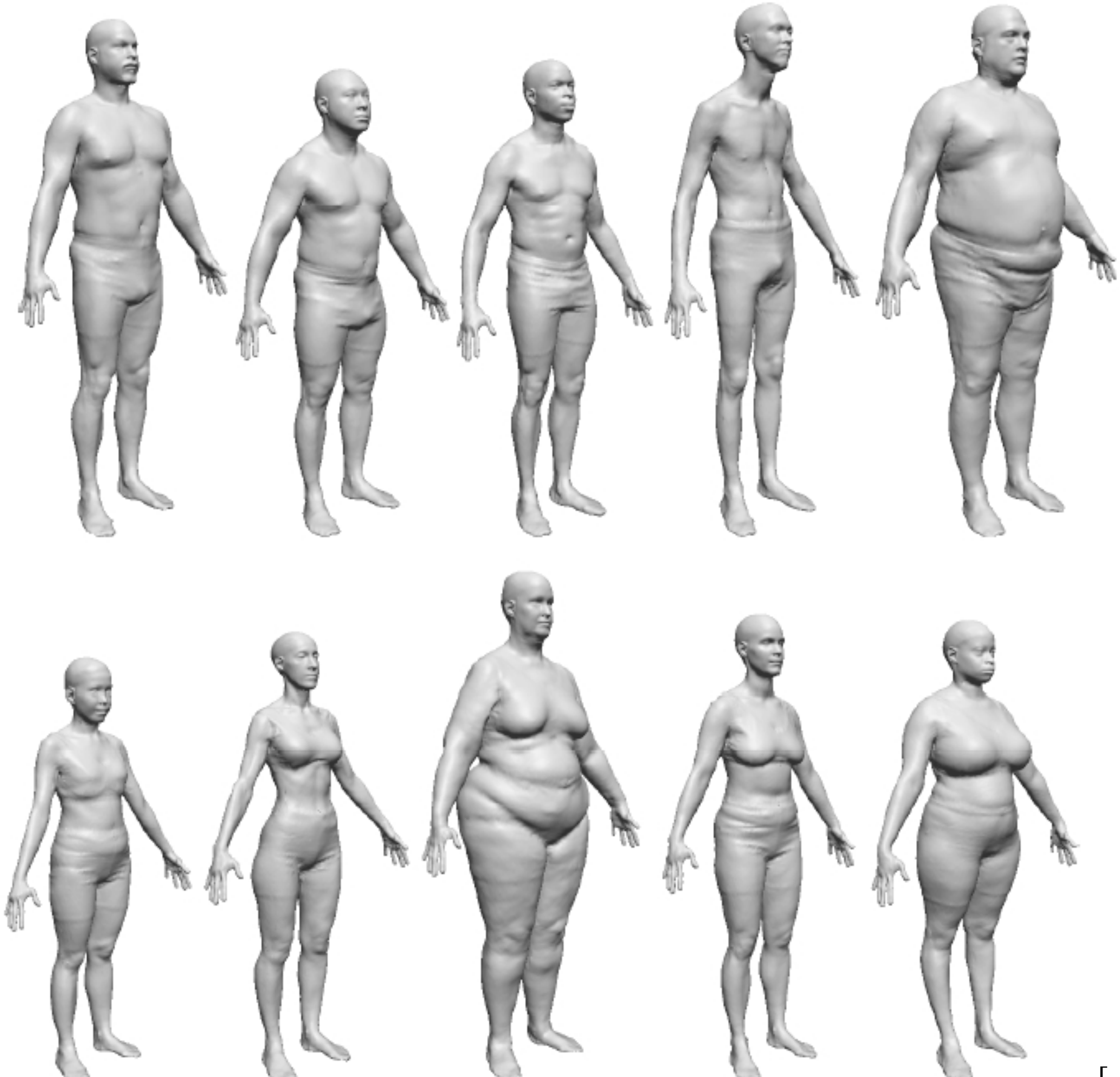
©Michael J. Black



Marr and Nishihara '78

Nevatia & Binford '73

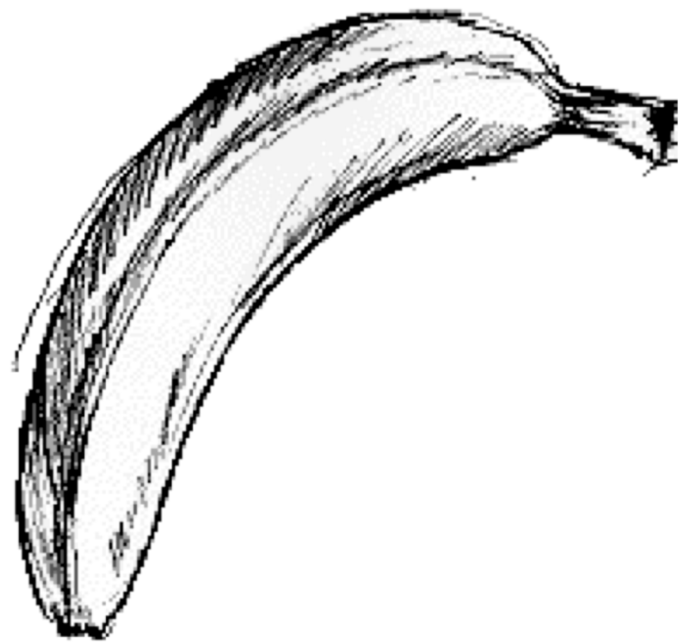
©Michael J. Black



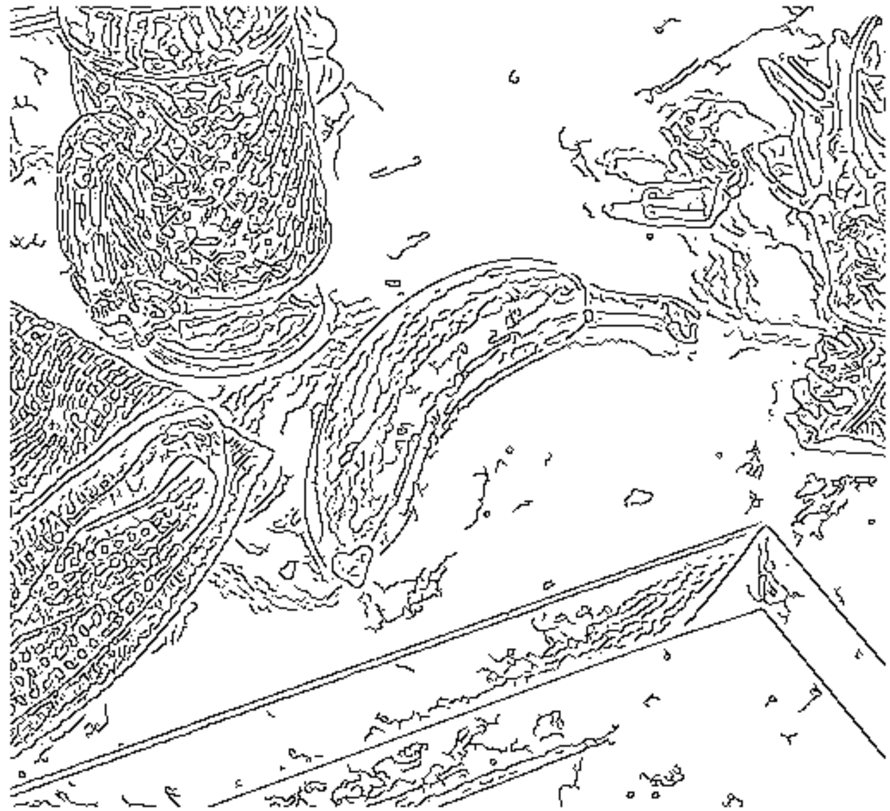
[Allen et al. '03]

Line Drawings

University of South Florida

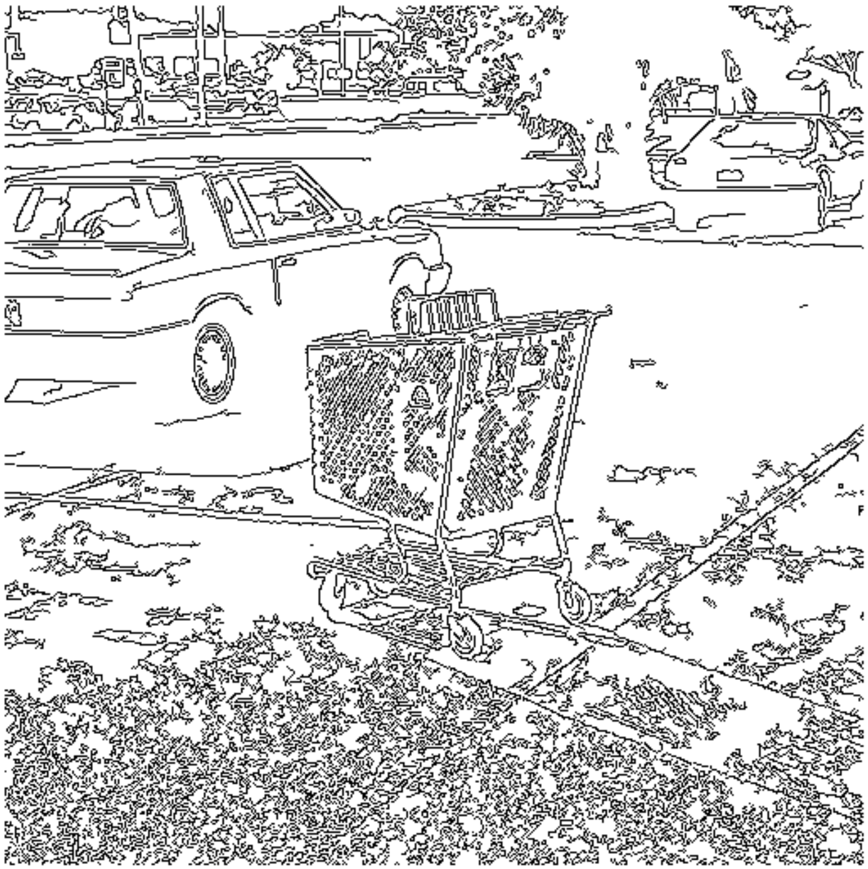
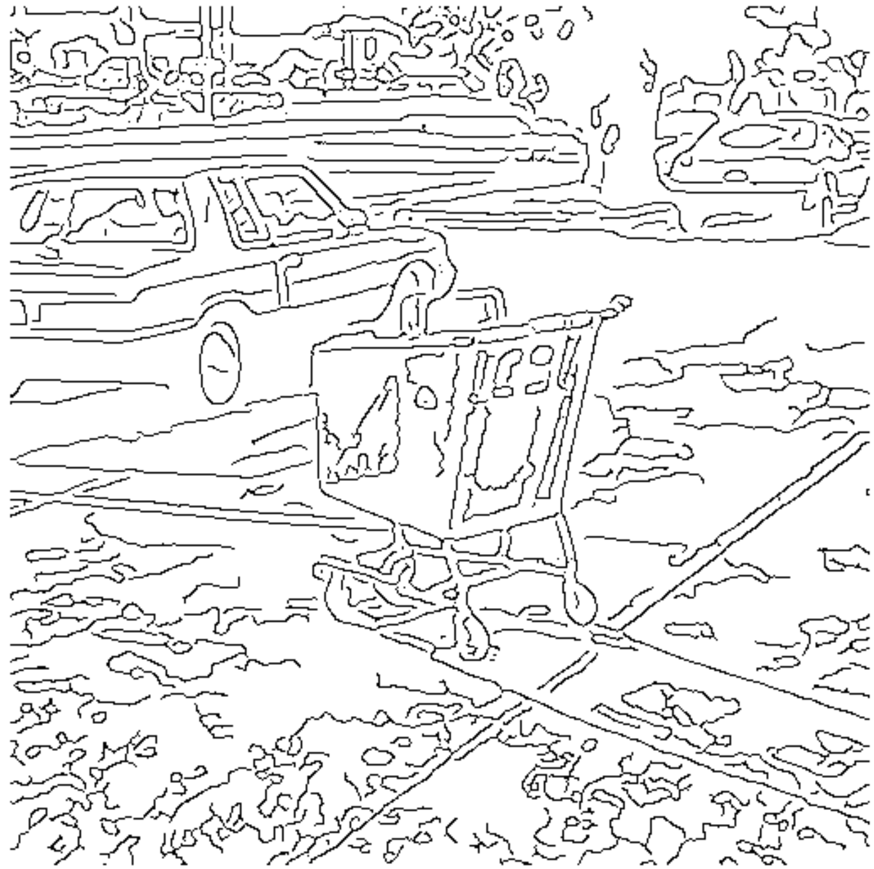


Versus Edge Detection

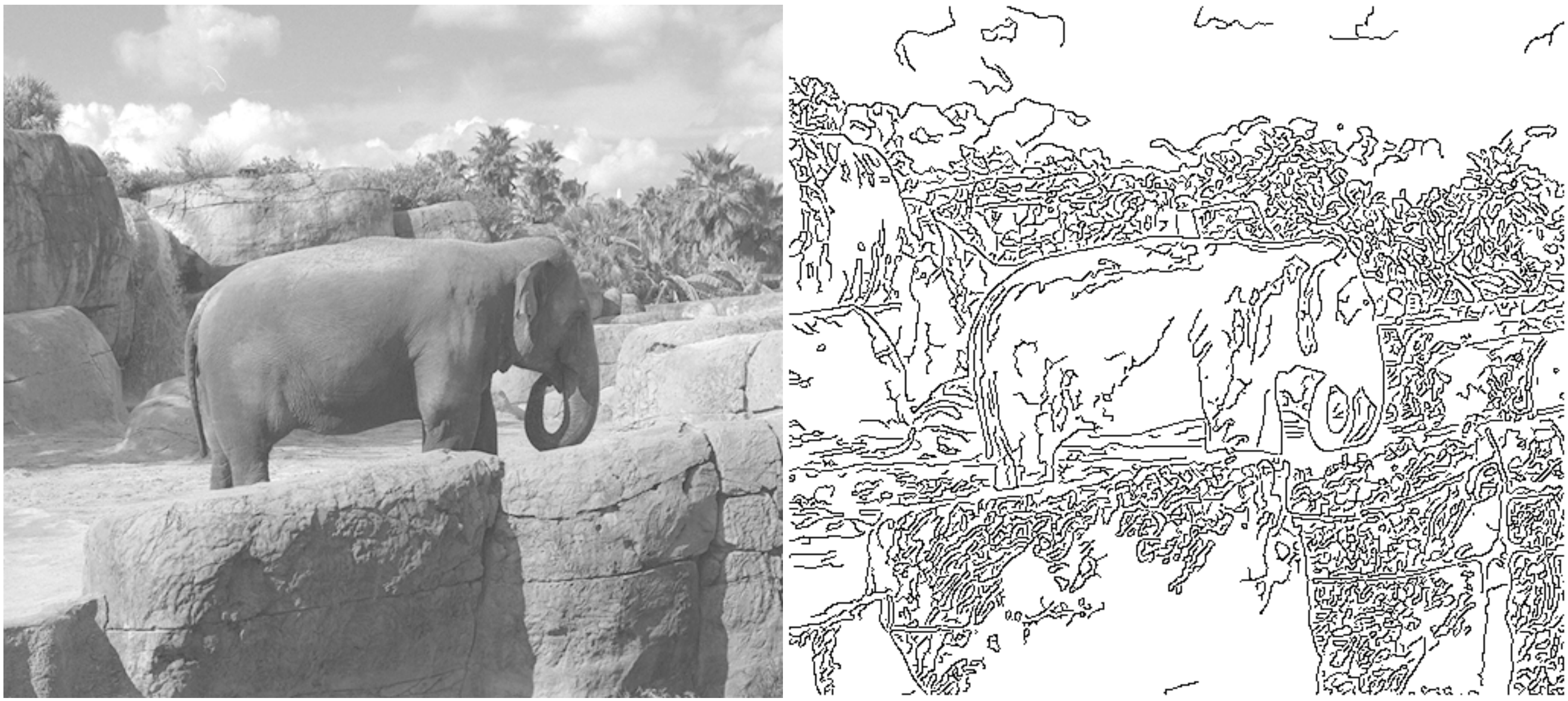
University of South Florida

Edges

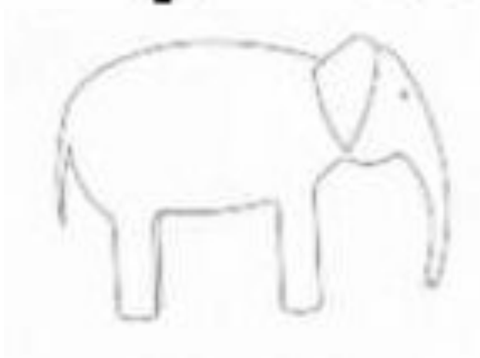
University of South Florida

Object Recognition

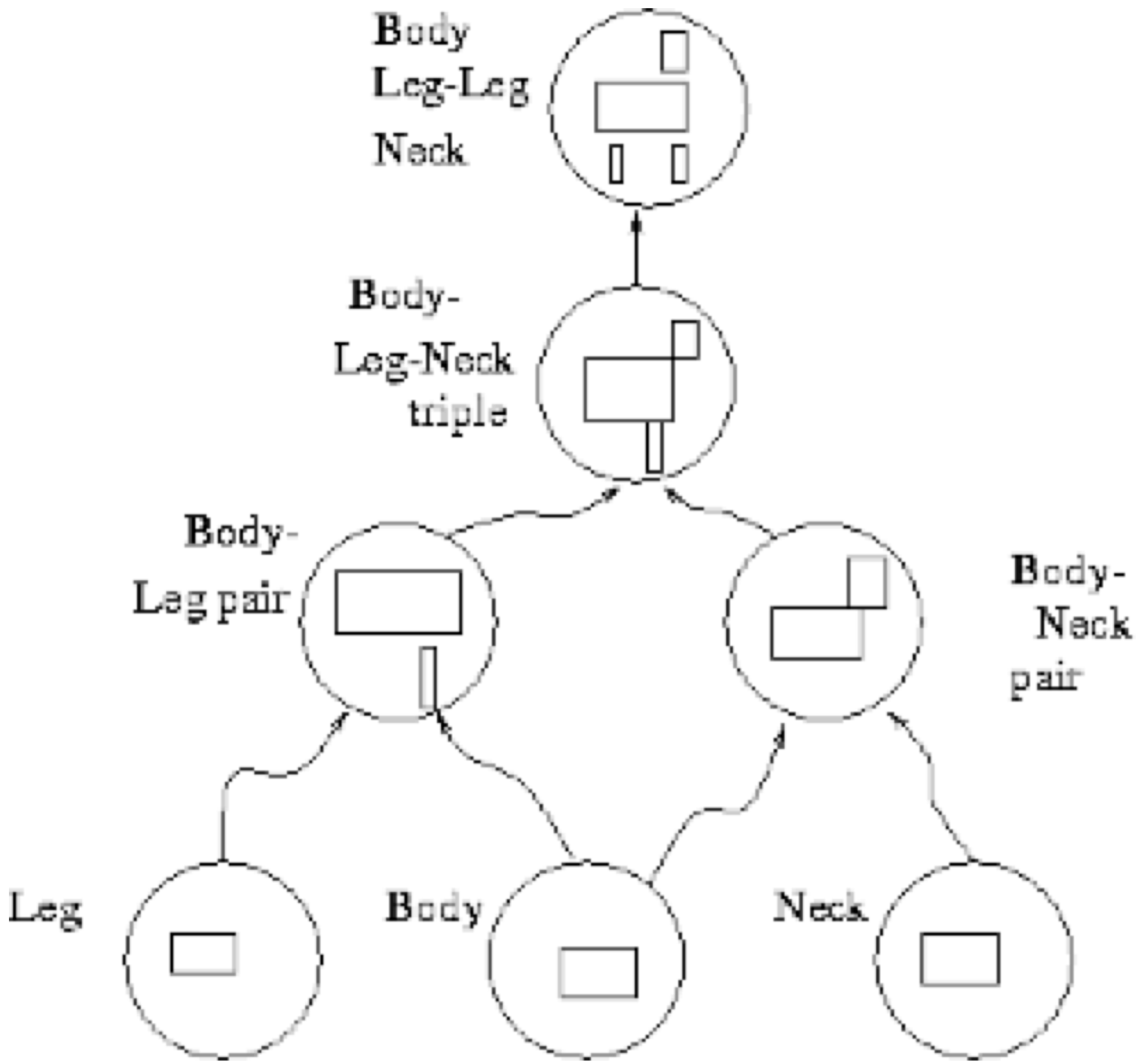


University of South Florida

Object Recognition



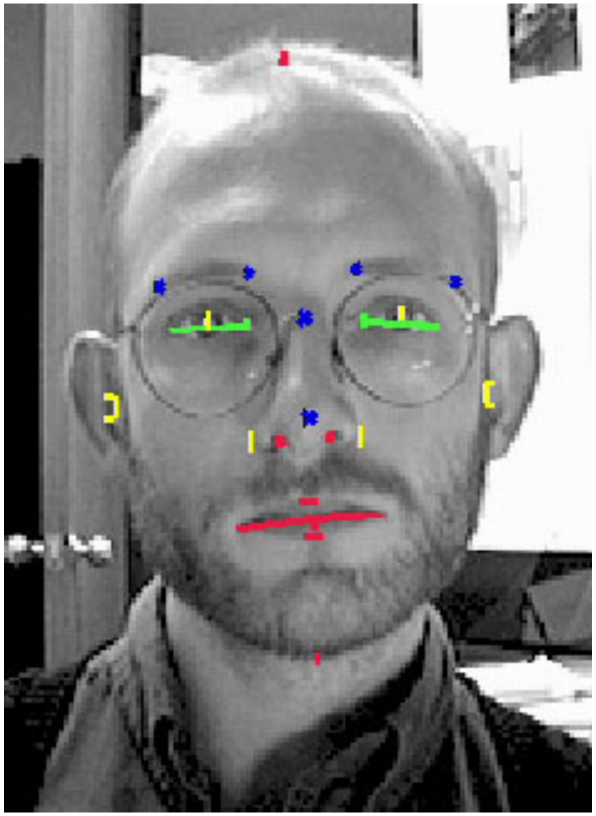
Match “model” to measurements?

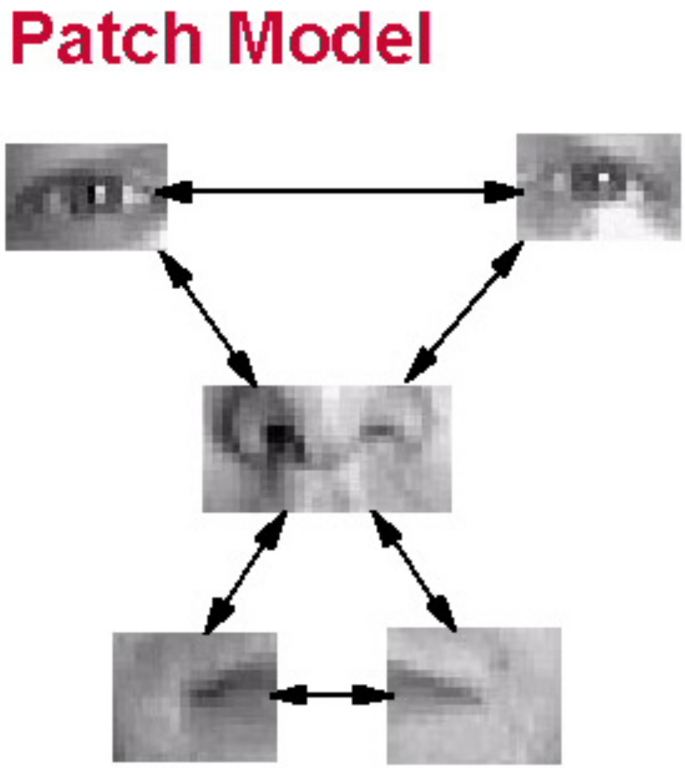


Forsyth

©Michael J. Black

Templates and Relations





http://www.research.ibm.com/ecvg/biom/facereco.html

Parts and Relations

How flexible are the spatial relations of the parts?

How good are our "models"?

Thompson, P. (1980). "Margaret Thatcher: a new illusion." *Perception* **9**:483-484

How good are our "models"?

Thompson, P. (1980). "Margaret Thatcher: a new illusion." *Perception* **9**:483-484

Is it only about matching?

What are our "models"?

How good are they?

Ron Rensick

P Sinha and T Poggio

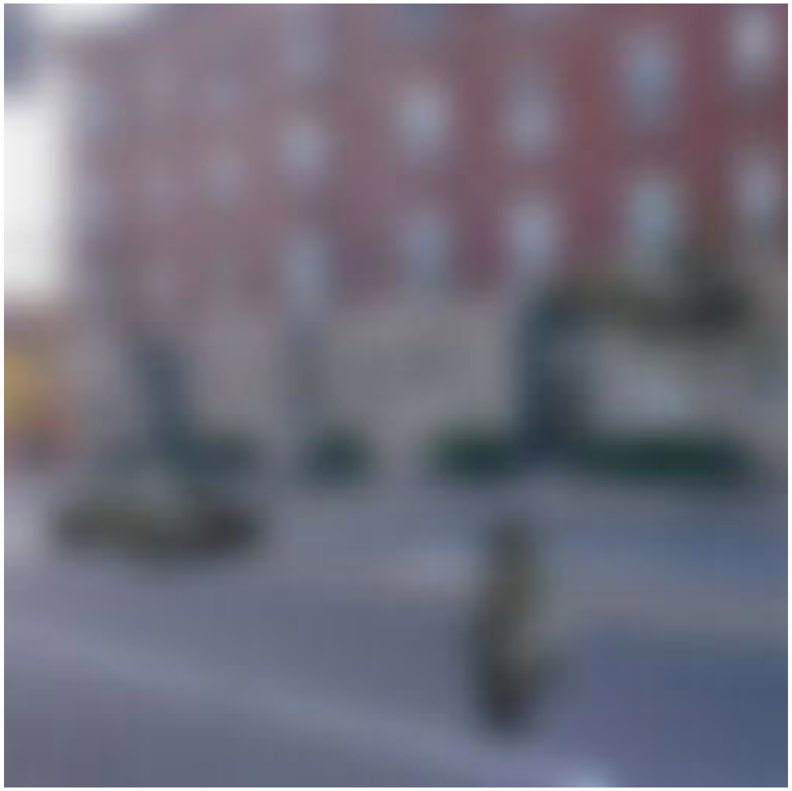
Context

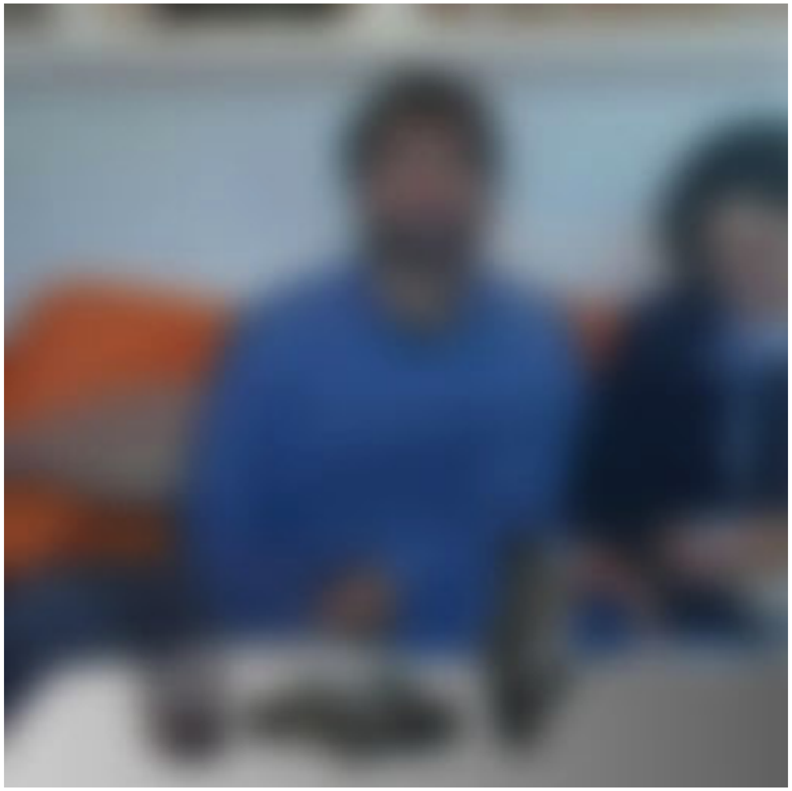
Antonio Torralba

Context

Antonio Torralba

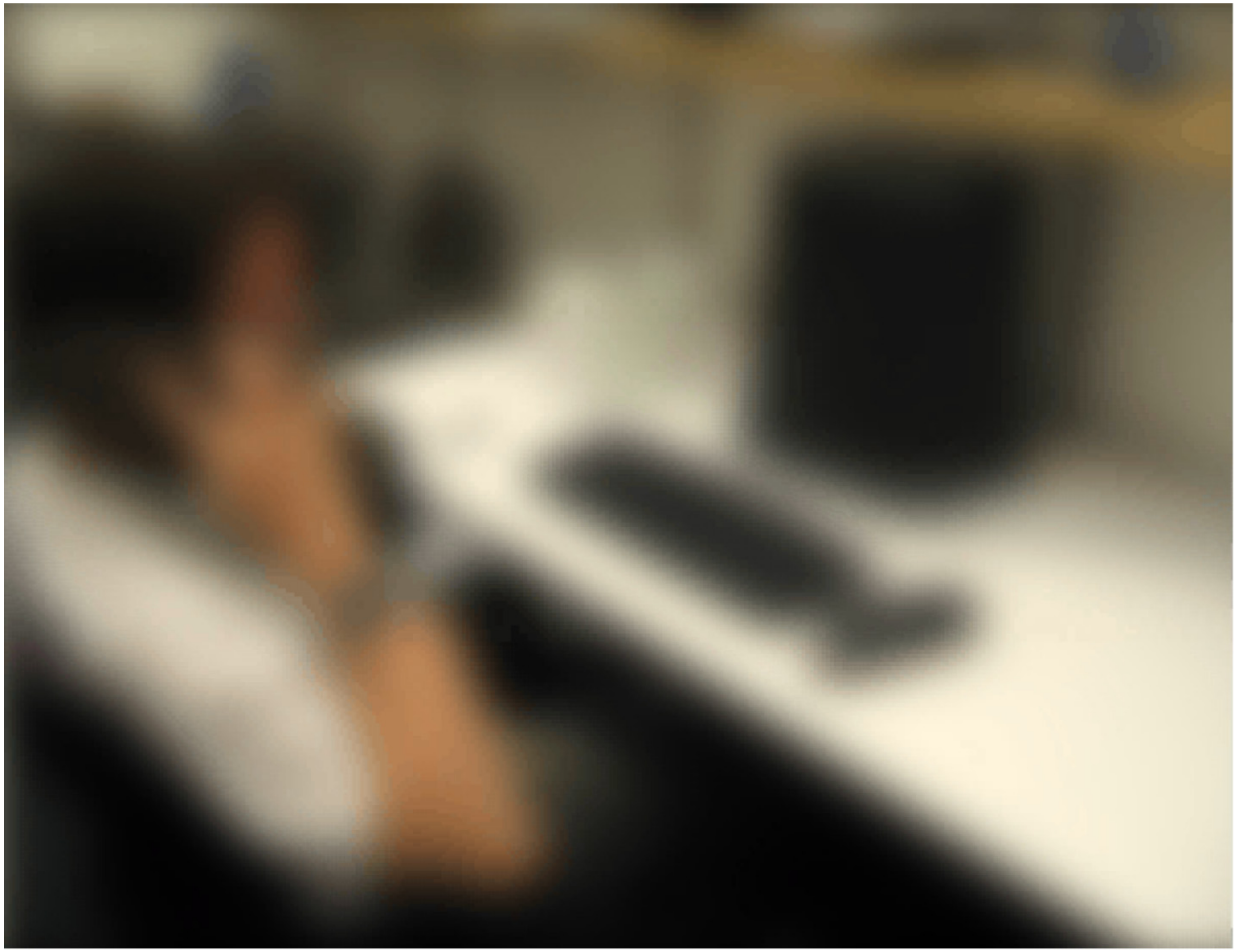
Context





Antonio Torralba

©Michael J. Black



Antonio Torralba

Antonio Torralba

Ambiguity

Inverse problems. Recover information that is lost. Make explicit information that is implicit.

Understand geometry and physics of light and world.

Our measurements are always ambiguous. This means perception involves *inference*. We must use our prior information about the world and the combination of evidence from multiple cues to infer what is in the world.

Understand probabilistic inference.

©Michael J. Black

Computer vision

Formalize

1) approximate physics, geometry and light
2) model the regularity of the world (geometric models, statsitics, learning)
3) create tractable inference/estimation problems
4) use modern optimization techniques to solve

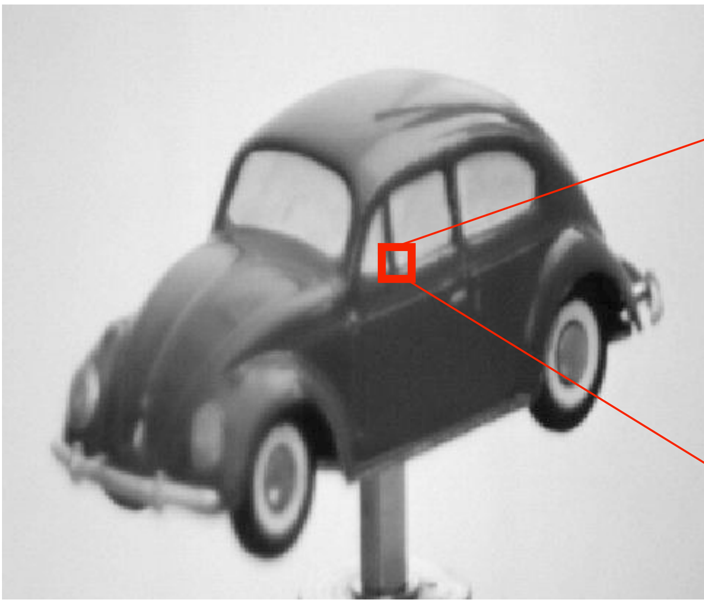
©Michael J. Black

So how do we go from an array of numbers to "perception"?

				49	151	176	182	179
				45	148	175	183	181
				42	146	176	185	184
				35	140	172	184	184
66	64	64	84	129	134	168	181	182
59	63	62	88	130	128	166	185	180
60	62	60	85	127	125	163	183	178
62	62	58	81	122	120	160	181	176
63	64	58	78	118	117	159	180	176

Irani and Basri

From images to understanding?



Huge array of numbers

64	60	69	100	149	151	176	182	179
65	62	68	97	145	148	175	183	181
65	66	70	95	142	146	176	185	184
66	66	68	90	135	140	172	184	184
66	64	64	84	129	134	168	181	182
59	63	62	88	130	128	166	185	180
60	62	60	85	127	125	163	183	178
62	62	58	81	122	120	160	181	176
63	64	58	78	118	117	159	180	176

Classifier?

CAR

Infeasible.
Reduce dimensionality.
Invariance to lighting, rotation, ….
Need to extract some salient structure - ***features***

©Michael J. Black

Image Filtering

3	3	3
3	?	3
3	3	3

3

Image Filtering

3	4	3
2	?	5
5	4	2

3

What assumptions are you making to infer the center value?

Linear functions

- Simplest: linear filtering.
  - Replace each pixel by a linear combination of its neighbors.
- The prescription for the linear combination is called the “convolution kernel”.

10	5	3
4	5	1
1	1	7

Local image data

0	0	0
0	0.5	0
0	1	0.5

kernel

	7	

Modified image data

Freeman

©Michael J. Black

Linear Filtering

- Linear means that the response of the filter at a pixel is a linear combination of other pixels.
  - Typically using a local neighborhood.
  - Linear methods simplest.
- Useful to:
  - Integrate information over constant regions.
  - Modify images (e.g. smooth or enhance)
  - Scale.
  - Detect features.

©Michael J. Black

2-D signals and convolutions

- Continuous $I(x,y)$
- Discrete $I[k,l]$ or $I_{k,l}$

- 2-D convolution (discrete)

$$f[m,n] = I \otimes g = \sum_{k=1}^{K} \sum_{l=1}^{L} I[m-k+\lfloor K/2 \rfloor, n-l+\lfloor L/2 \rfloor] g[k,l]$$

"filtered" image

filter "kernel"

2-D signals and correlation

- Continuous $I(x,y)$
- Discrete $I[k,l]$ or $I_{k,l}$

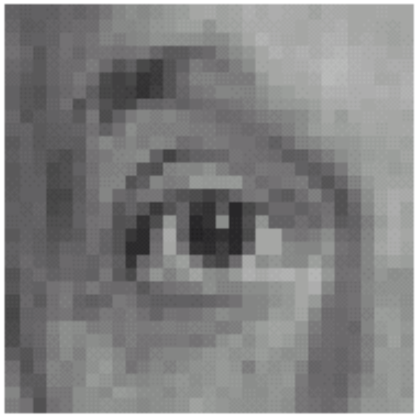
- 2-D correlation (discrete)

$$f[m,n] = I \otimes g = \sum_{k=1}^{K} \sum_{l=1}^{L} I[m+k-\lfloor K/2 \rfloor, n+l-\lfloor L/2 \rfloor] g[k,l]$$

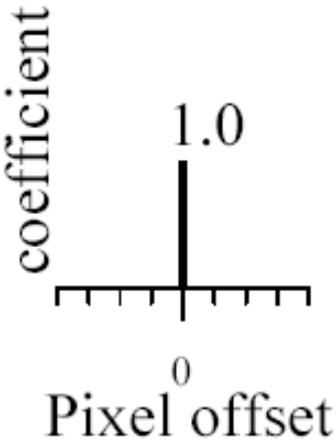
"filtered" image

filter "kernel"

Linear filtering (warm-up slide)



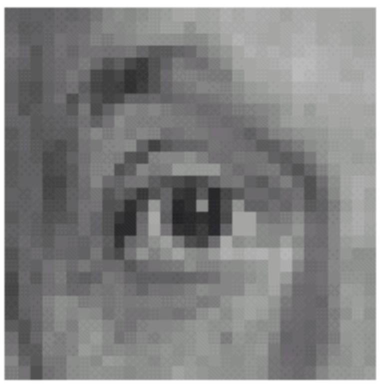
original



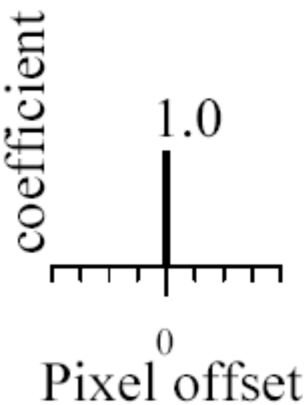
?

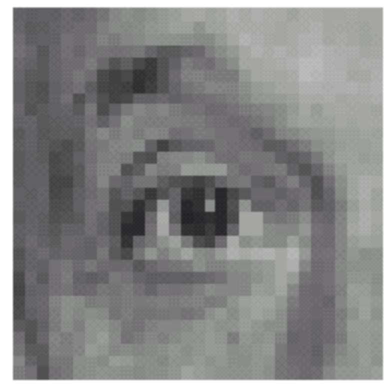
Freeman

Linear filtering (warm-up slide)



original



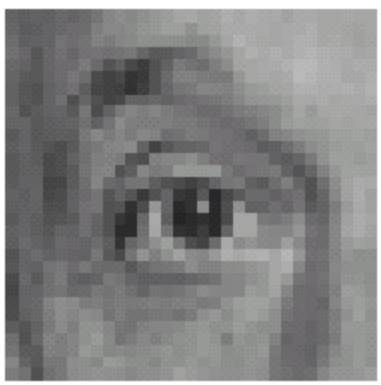


Filtered
(no change)

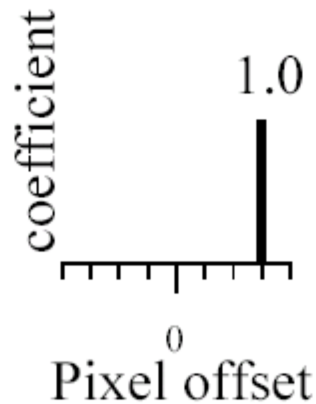
Freeman

©Michael J. Black

Linear filtering



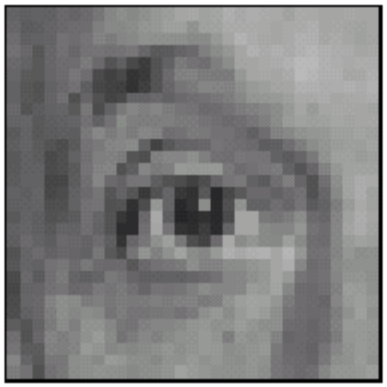
original



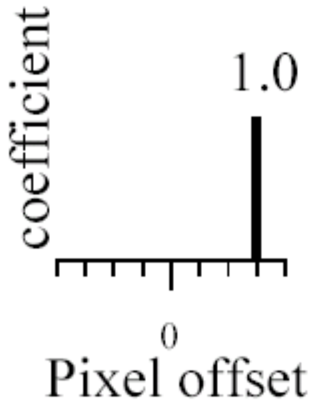
?

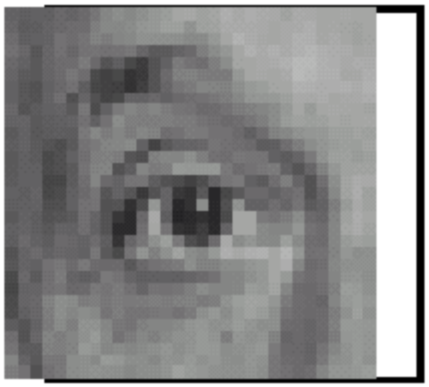
Freeman

shift



original



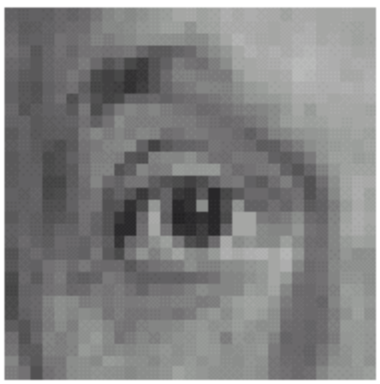


shifted

Freeman

©Michael J. Black

Linear filtering



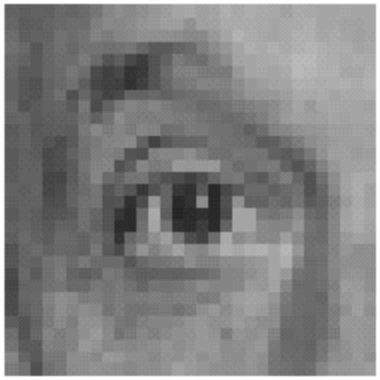
original

?

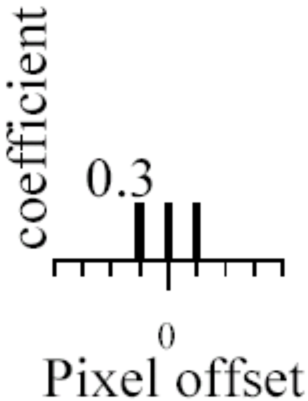
Freeman

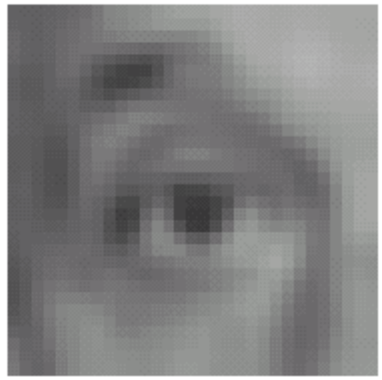
©Michael J. Black

Blurring



original



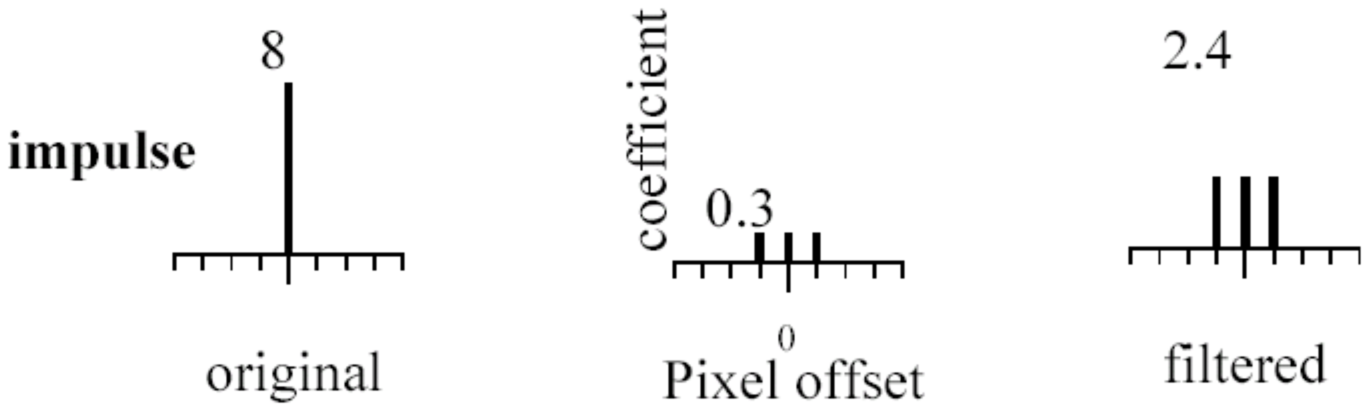


Blurred (filter applied in both dimensions).

Freeman

©Michael J. Black

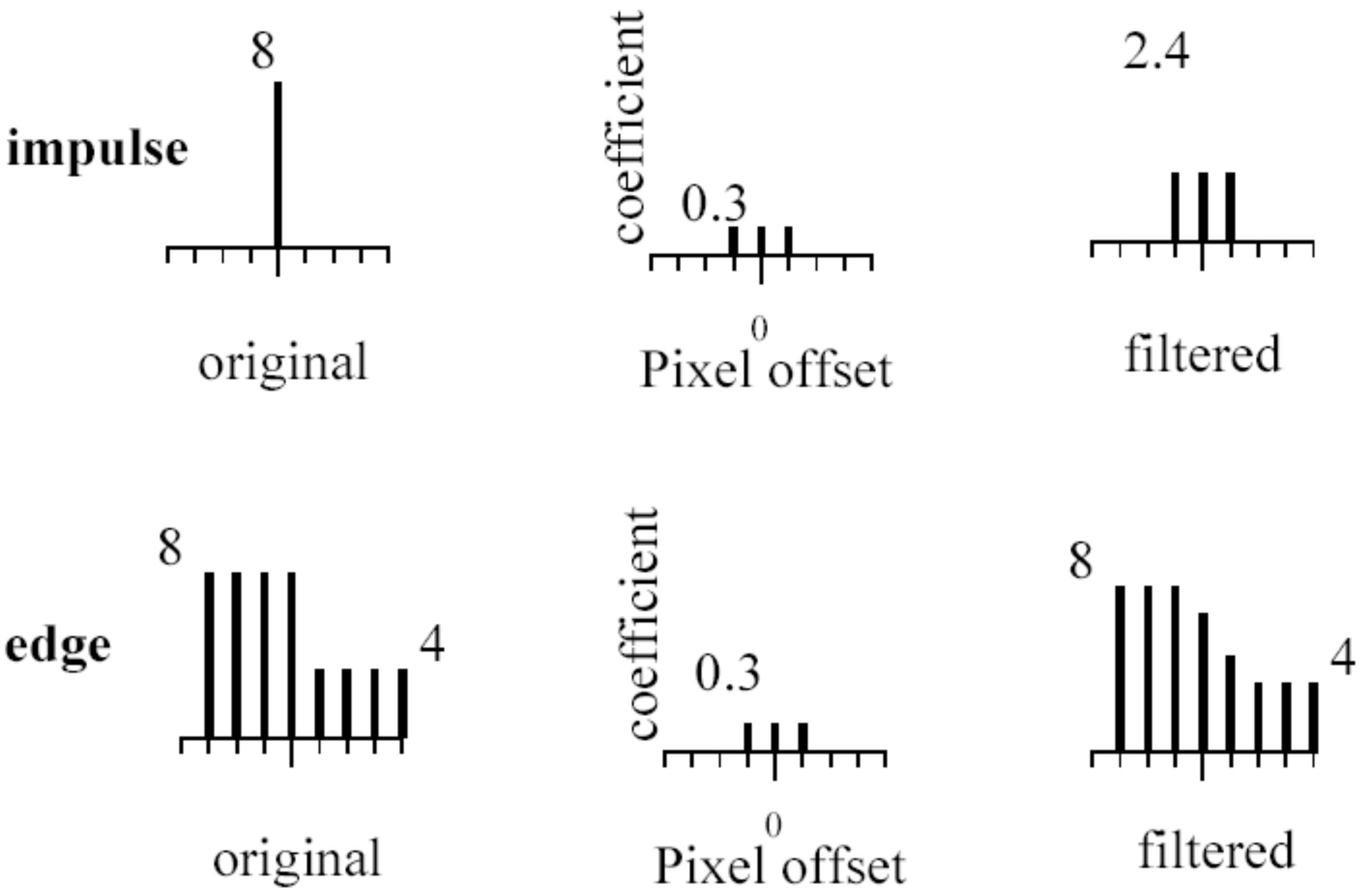
Blur examples



Freeman

©Michael J. Black

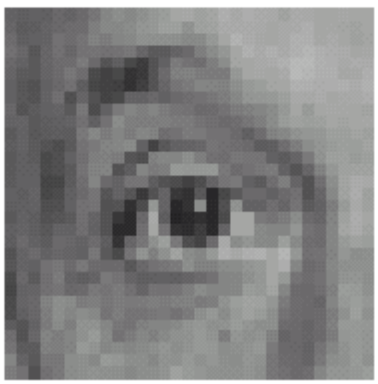
Blur examples



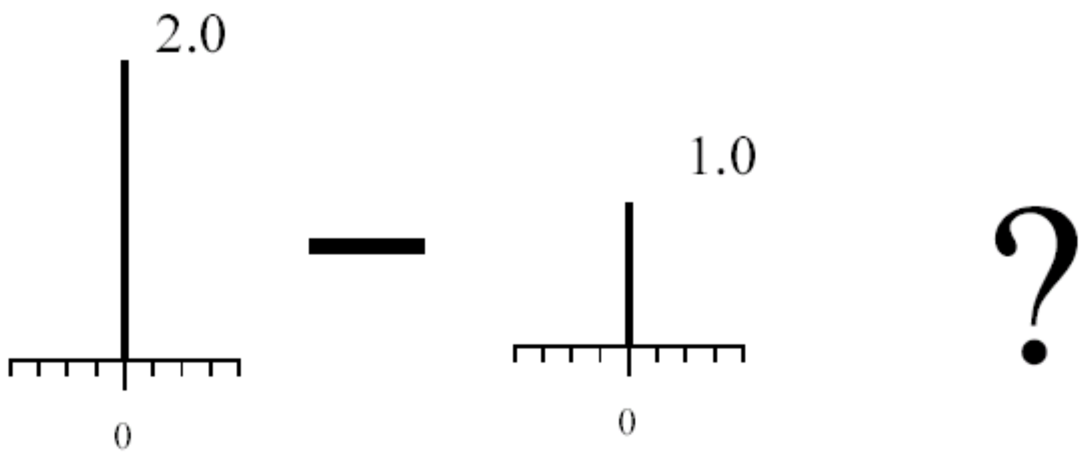
Freeman

©Michael J. Black

Linear filtering (warm-up slide)



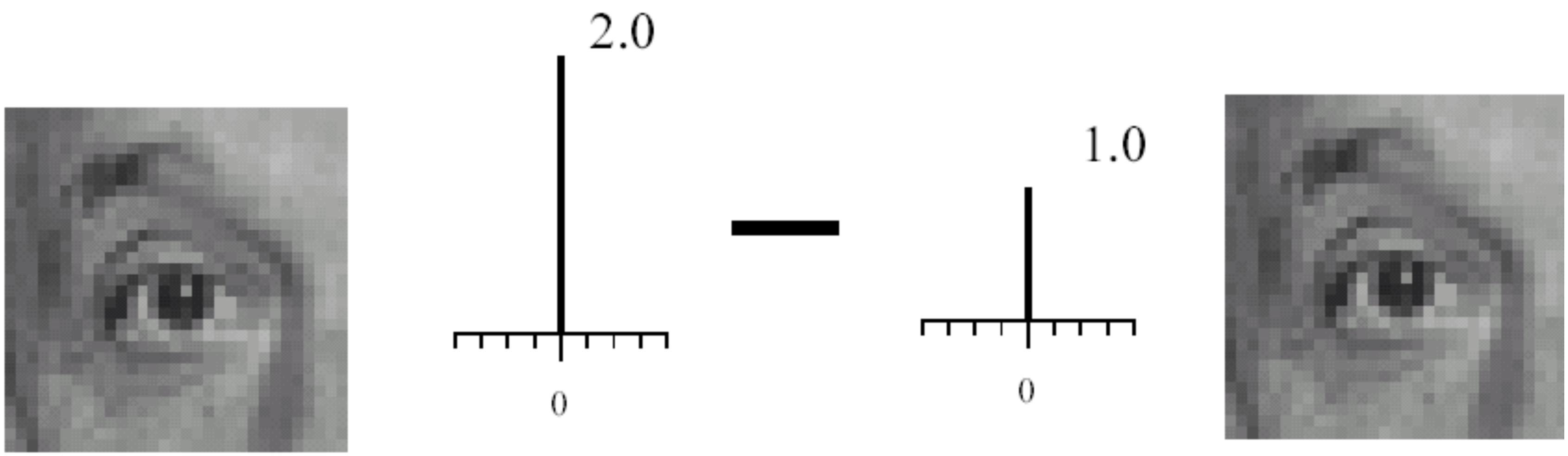
original



Freeman

©Michael J. Black

Linear filtering (no change)

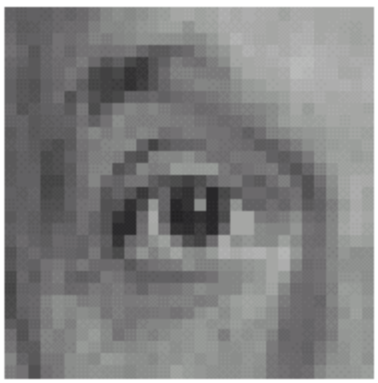


original

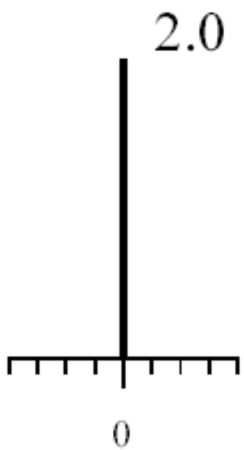
Filtered
(no change)

Freeman

Linear filtering



original



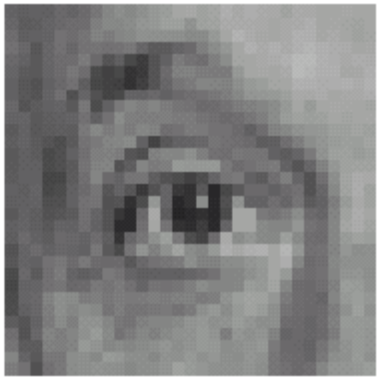
−

?

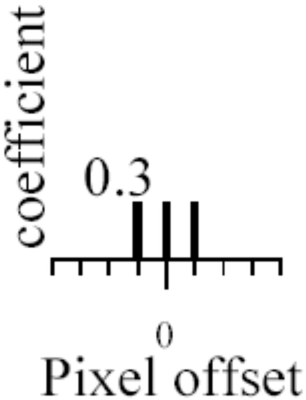
Freeman

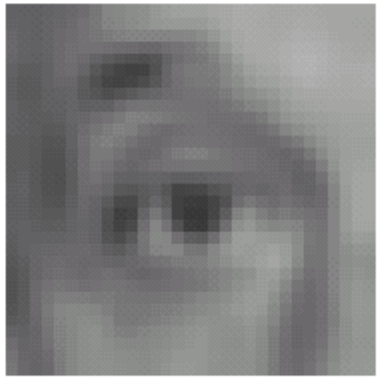
©Michael J. Black

(remember blurring)



original



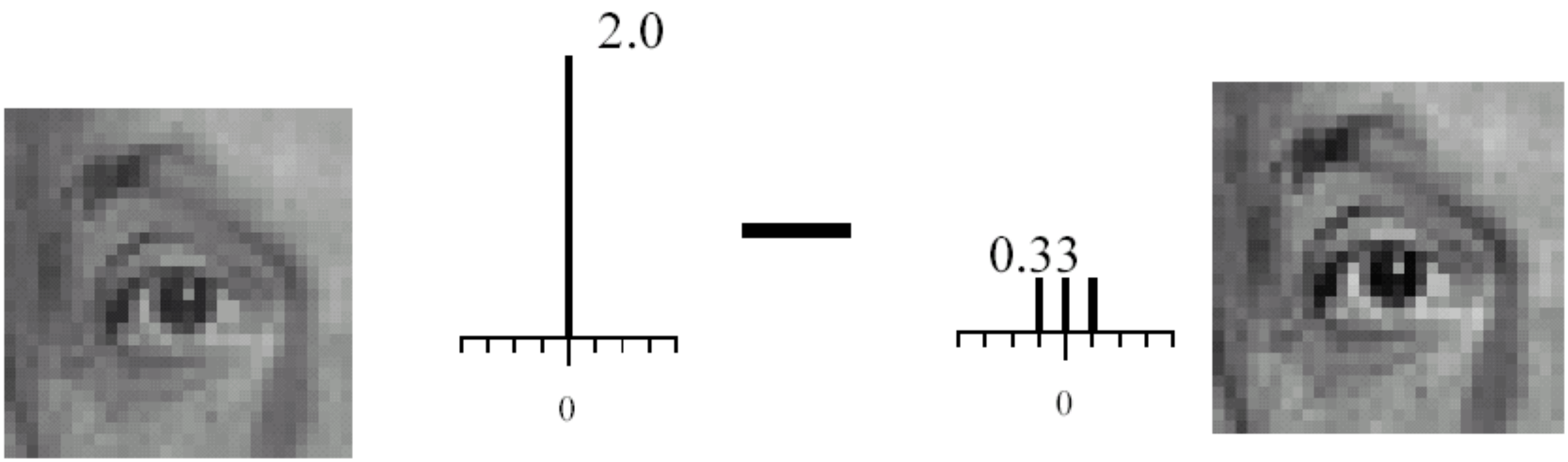


Blurred (filter applied in both dimensions).

Freeman

©Michael J. Black

Sharpening

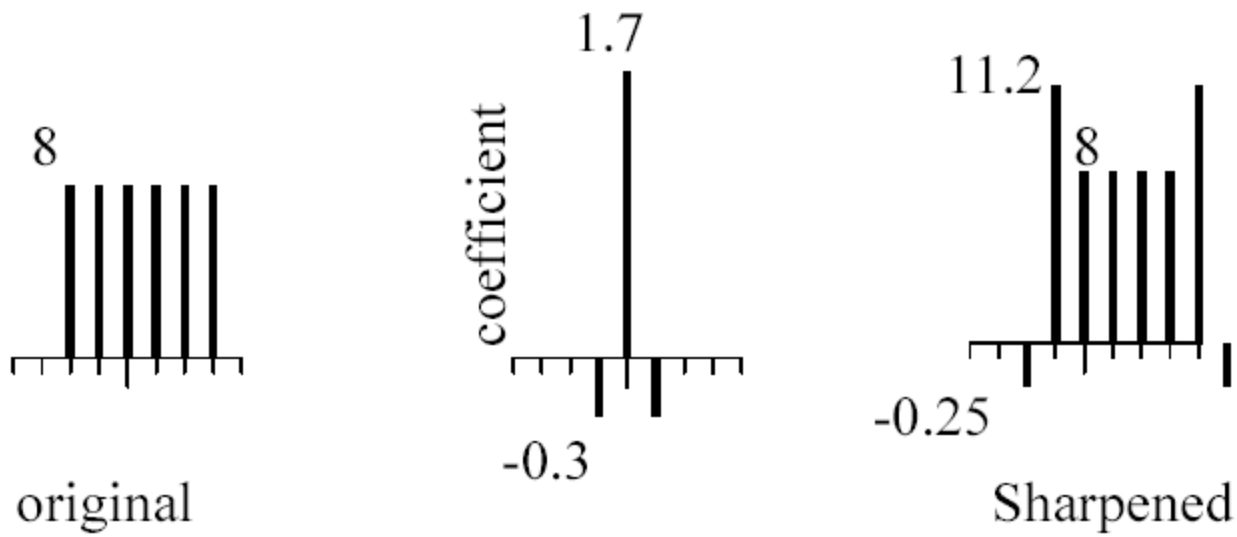


original

Sharpened original

Freeman

Sharpening example

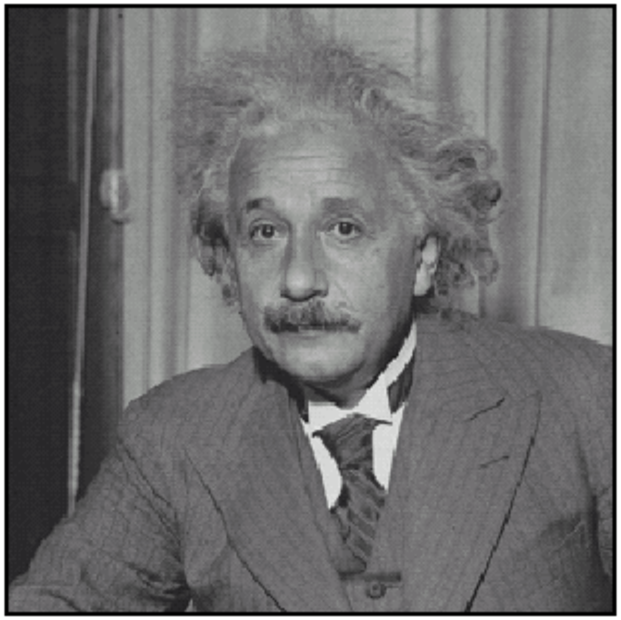


Sharpened (differences are accentuated; constant areas are left untouched).

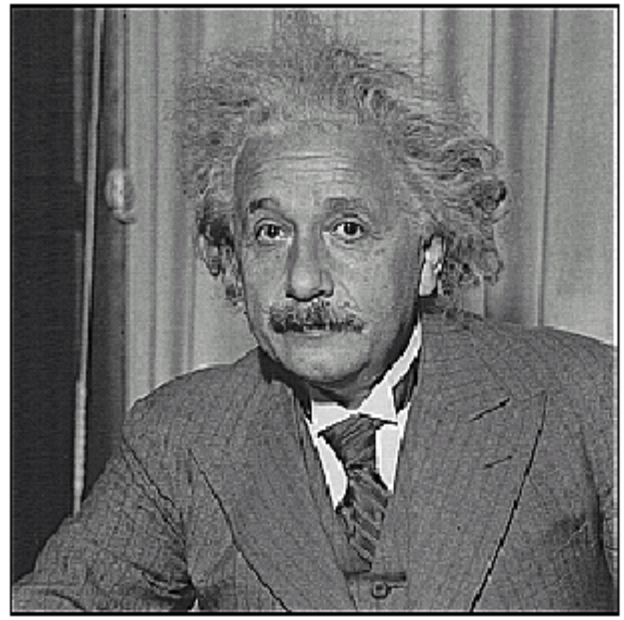
Freeman

©Michael J. Black

Sharpening



before



after

Freeman

Filtering to reduce noise

- “Noise” is what we’re not interested in.
  - We’ll discuss simple, low-level noise today: Light fluctuations; Sensor noise; Quantization effects; Finite precision
  - Not complex: shadows; extraneous objects.
- A pixel’s neighborhood contains information about its intensity.
- Averaging noise reduces its effect.

Average Filter

- Mask with positive entries, that sum 1.
- Replaces each pixel with an average of its neighborhood.
- If all weights are equal, it is called a BOX filter.

F

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

(Camps)

Example: Smoothing by Averaging