# Course Calendar

| Lecture | Date | Description | Readings | Assignments | Materials |
|---|---|---|---|---|---|
| 1 | 2/1 | Course Introduction Cameras and Lenses | Req: FP 1.1, 2.1, 2.2, 2.3, 3.1, 3.2 | PS0 out | |
| 2 | 2/3 | Image Filtering | Req: FP 7.1 - 7.6 | | |
| 3 | 2/8 | Image Representations: Pyramids | Req: FP 7.7, 9.2 | | |
| 4 | 2/10 | Image Statistics | | PS0 due | |
| 5 | 2/15 | Texture | Req: FP 9.1, 9.3, 9.4 | PS1 out | |
| 6 | 2/17 | Color | Req: FP 6.1-6.4 | | |
| 7 | 2/22 | Guest Lecture: Context in vision | | | |
| 8 | 2/24 | Guest Lecture: Medical Imaging | | PS1 due | |
| 9 | 3/1 | Multiview Geometry | Req: Mikolajczyk and Schmid; FP 10 | PS2 out | |
| 10 | 3/3 | Local Features | Req: Shi and Tomasi; Lowe | | |

# Course Calendar

Today

| Lecture | Date | Description | Readings | Assignments | Materials |
|---|---|---|---|---|---|
| 3 | 2/8 | Image Representations: Pyramids | Req: FP 7.7, 9.2 | | |
| 4 | 2/10 | Image Statistics | | PS0 due | |

# Reading

- Related to today’s lecture:
  - Adelson article on pyramid representations, posted on web site.
  - Farid paper posted on web site.

# Image pyramids

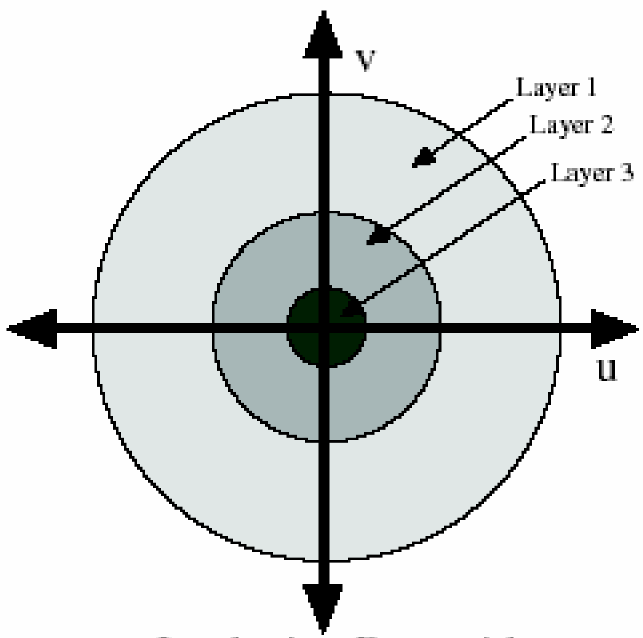
- Gaussian
- Laplacian
- Wavelet/QMF
- Steerable pyramid

# Steerable pyramids

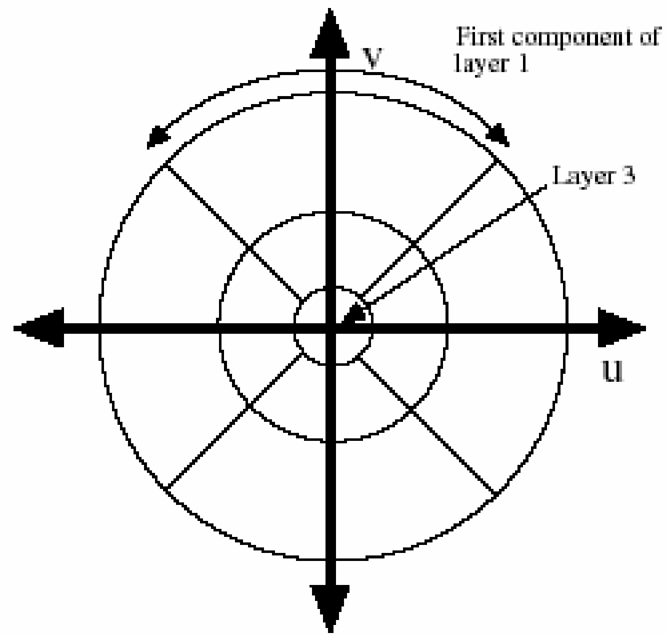
- Good:
  - Oriented subbands
  - Non-aliased subbands
  - Steerable filters
- Bad:
  - Overcomplete
  - Have one high frequency residual subband, required in order to form a circular region of analysis in frequency from a square region of support in frequency.

# Oriented pyramids

- Laplacian pyramid is orientation independent
- Apply an oriented filter to determine orientations at each layer
  - by clever filter design, we can simplify synthesis
  - this represents image information at a particular scale and orientation




Laplacian Pyramid




Oriented Pyramid

| | Laplacian Pyramid | Dyadic QMF/Wavelet | Steerable Pyramid |
|---|---|---|---|
| self-inverting (tight frame) | no | yes | yes |
| overcompleteness | $4/3$ | 1 | $4k/3$ |
| aliasing in subbands | perhaps | yes | no |
| rotated orientation bands | no | only on hex lattice [9] | yes |

**Table 1:** Properties of the Steerable Pyramid relative to two other well-known multi-scale representations.

http://www.cns.nyu.edu/ftp/eero/simoncelli95b.pdf

Simoncelli and Freeman, ICIP 1995

But we need to get rid of the corner regions before starting the recursive circular filtering

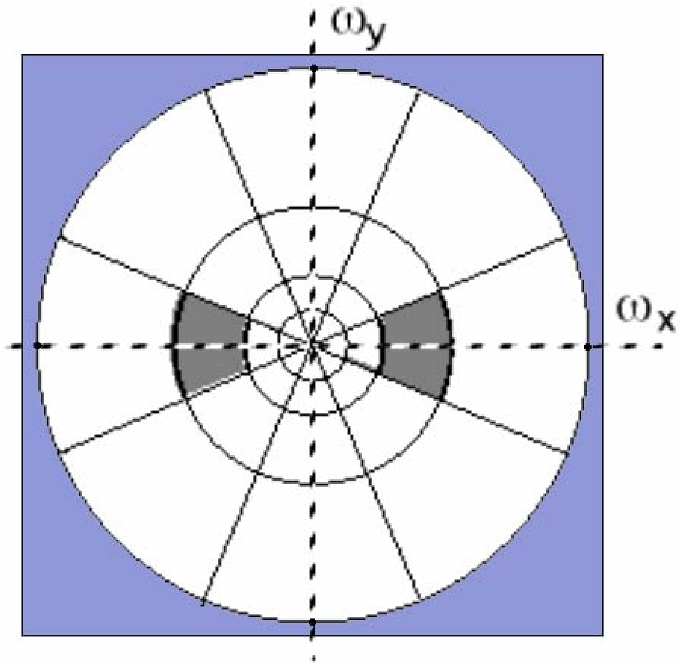
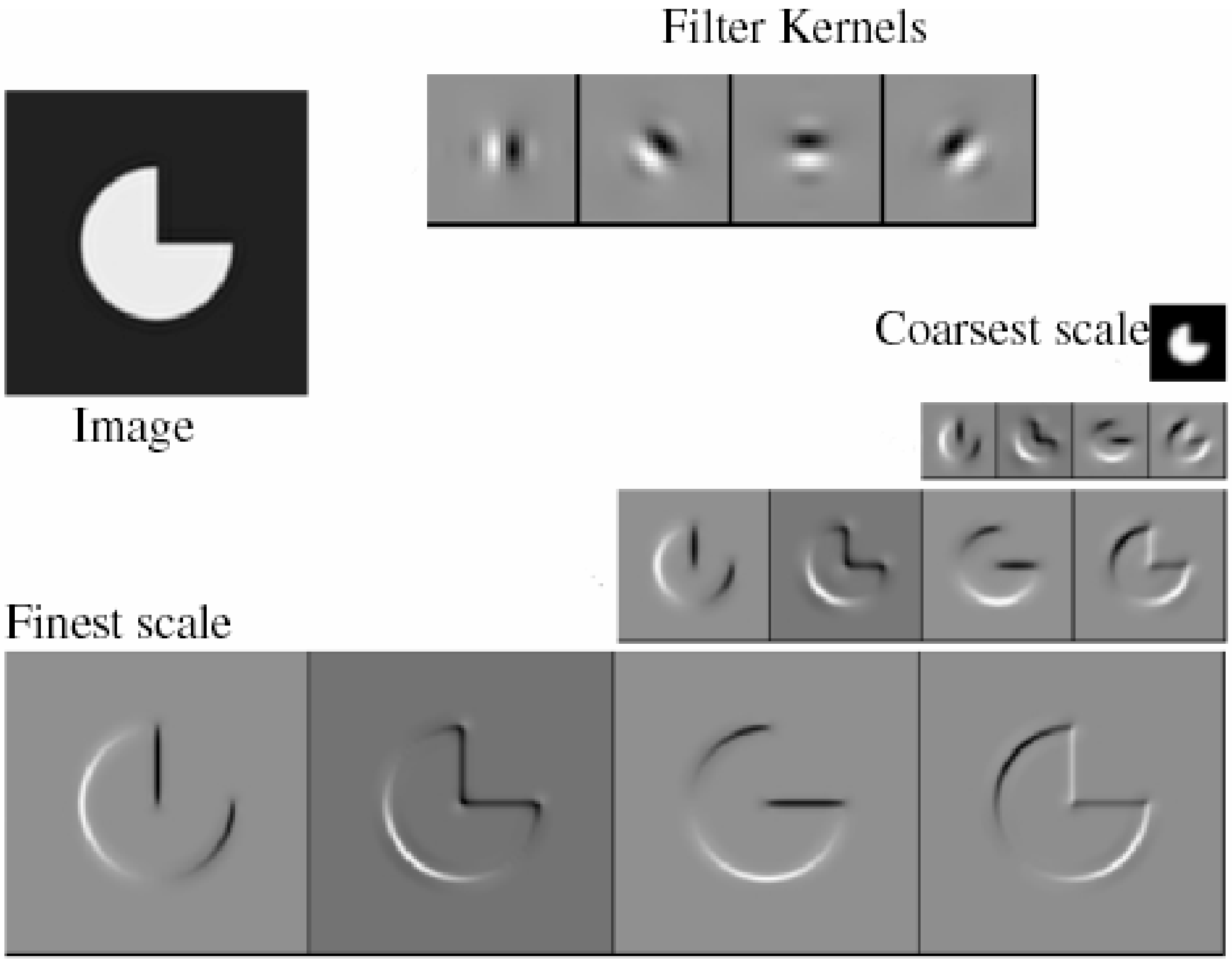



**Figure 1.** Idealized illustration of the spectral decomposition performed by a steerable pyramid with $k = 4$. Frequency axes range from $-\pi$ to $\pi$. The basis functions are related by translations, dilations and *rotations* (except for the initial highpass subband and the final lowpass subband). For example, the shaded region corresponds to the spectral support of a single (vertically-oriented) subband.

http://www.cns.nyu.edu/ftp/eero/simoncelli95b.pdf

Simoncelli and Freeman, ICIP 1995



Filter Kernels
Image
Coarsest scale
Finest scale

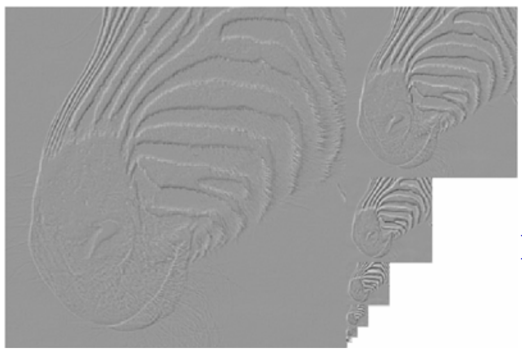
- Summary of pyramid representations

# Image pyramids

- Gaussian

Progressively blurred and subsampled versions of the image. Adds scale invariance to fixed-size algorithms.

- Laplacian

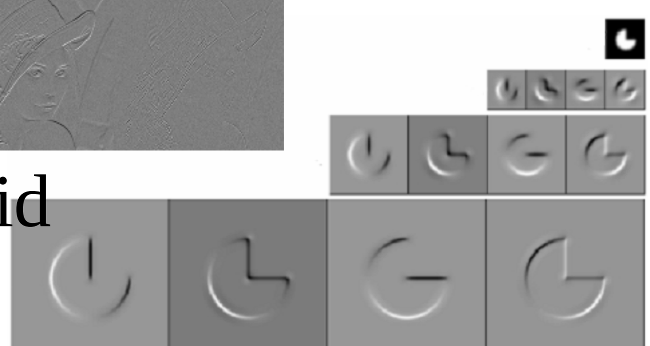


Shows the information added in Gaussian pyramid at each spatial scale. Useful for noise reduction & coding.

- Wavelet/QMF

Bandpassed representation, complete, but with aliasing and some non-oriented subbands.

- Steerable pyramid



Shows components at each scale and orientation separately. Non-aliased subbands. Good for texture and feature analysis.

# Linear image transformations

- In analyzing images, it's often useful to make a change of basis.

transformed image

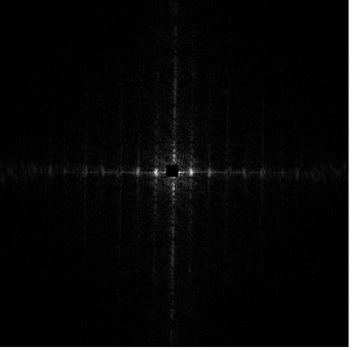
$$\vec{F} = U\vec{f}$$

Vectorized image

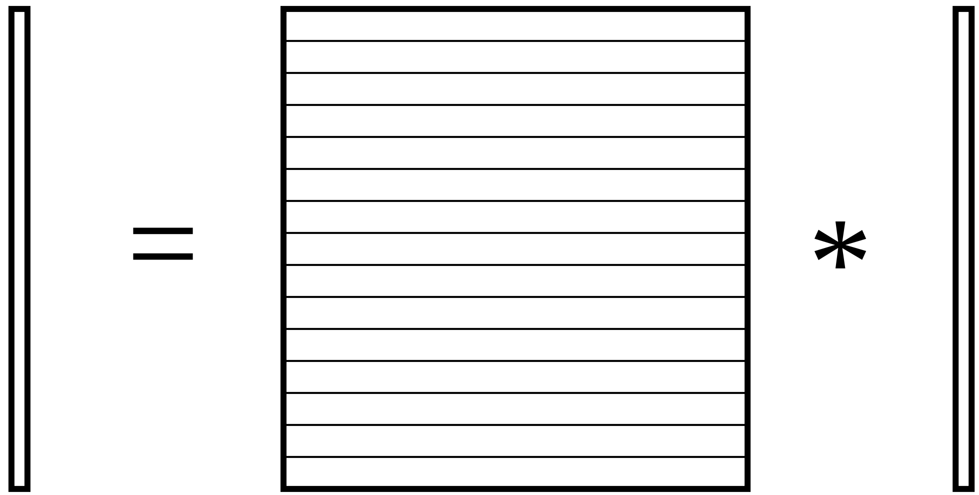
Fourier transform, or
Wavelet transform, or
Steerable pyramid transform

# Schematic pictures of each matrix transform

- Shown for 1-d images
- The matrices for 2-d images are the same idea, but more complicated, to account for vertical, as well as horizontal, neighbor relationships.



# Fourier transform

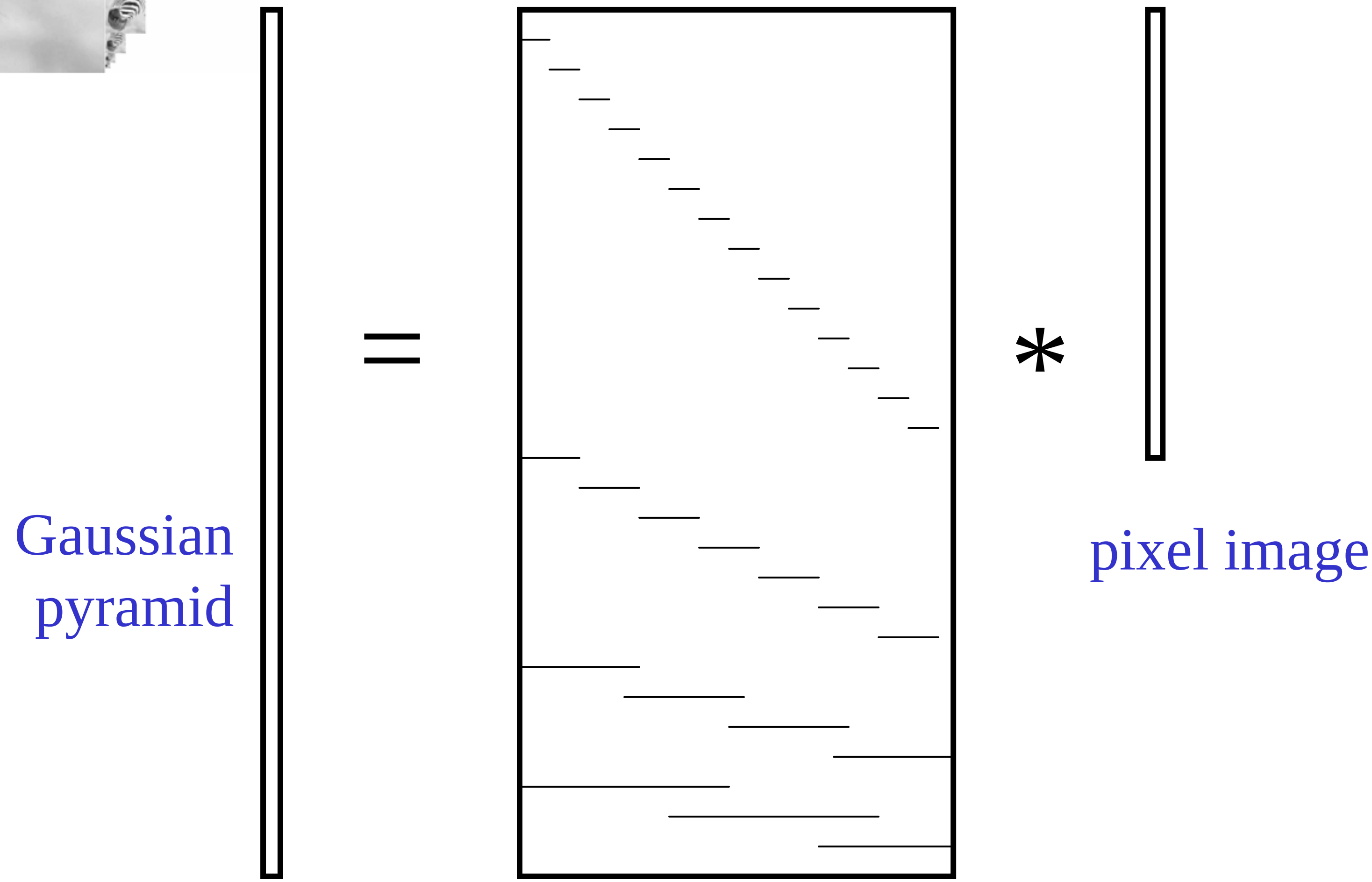



Fourier transform

Fourier bases are global: each transform coefficient depends on all pixel locations.

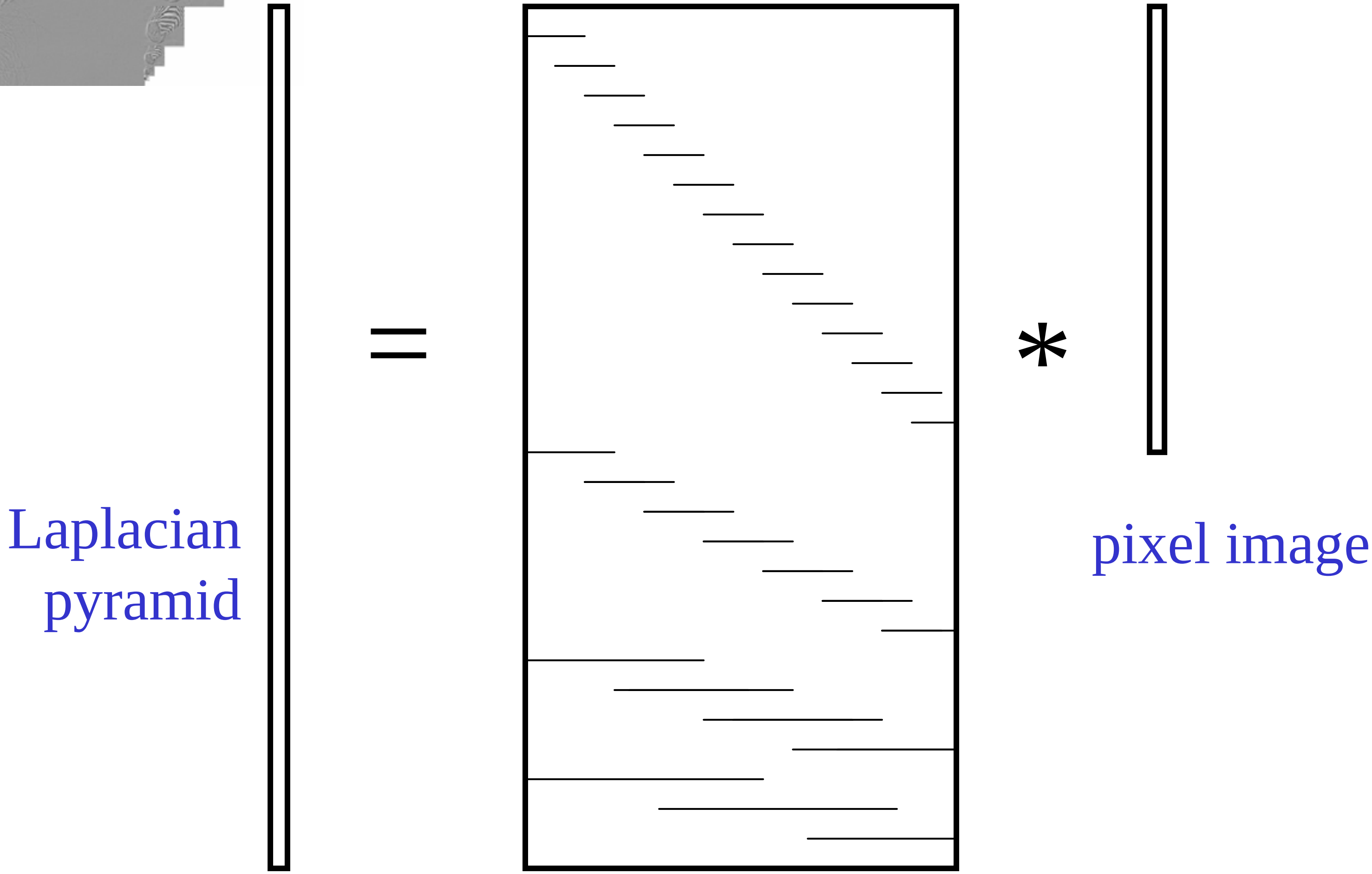
pixel domain image

# Gaussian pyramid

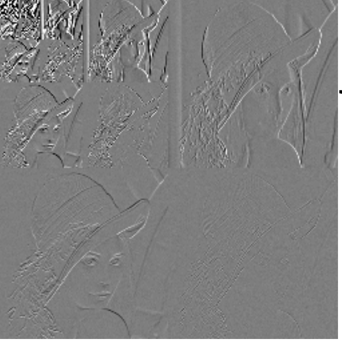



Overcomplete representation.
Low-pass filters, sampled
appropriately for their blur.

# Laplacian pyramid

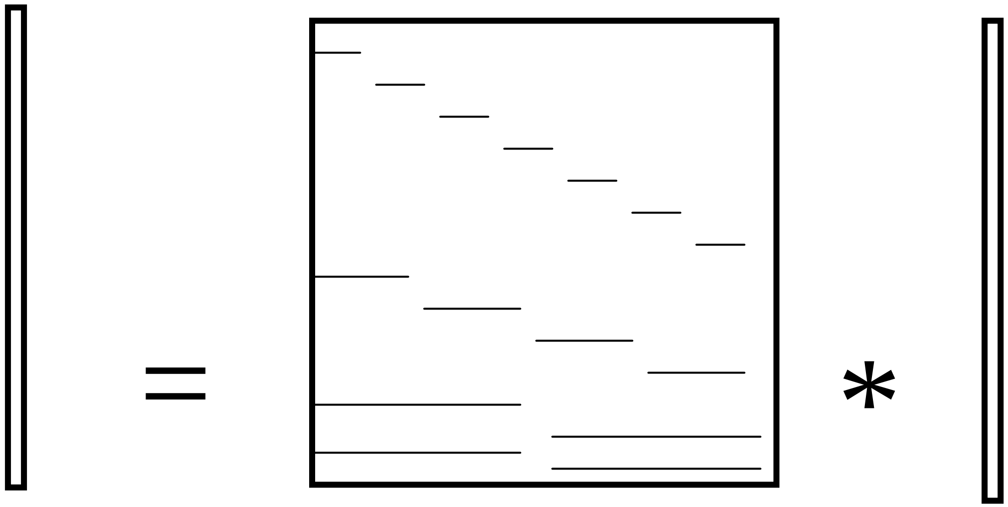



Overcomplete representation.
Transformed pixels represent
bandpassed image information.



# Wavelet (QMF) transform

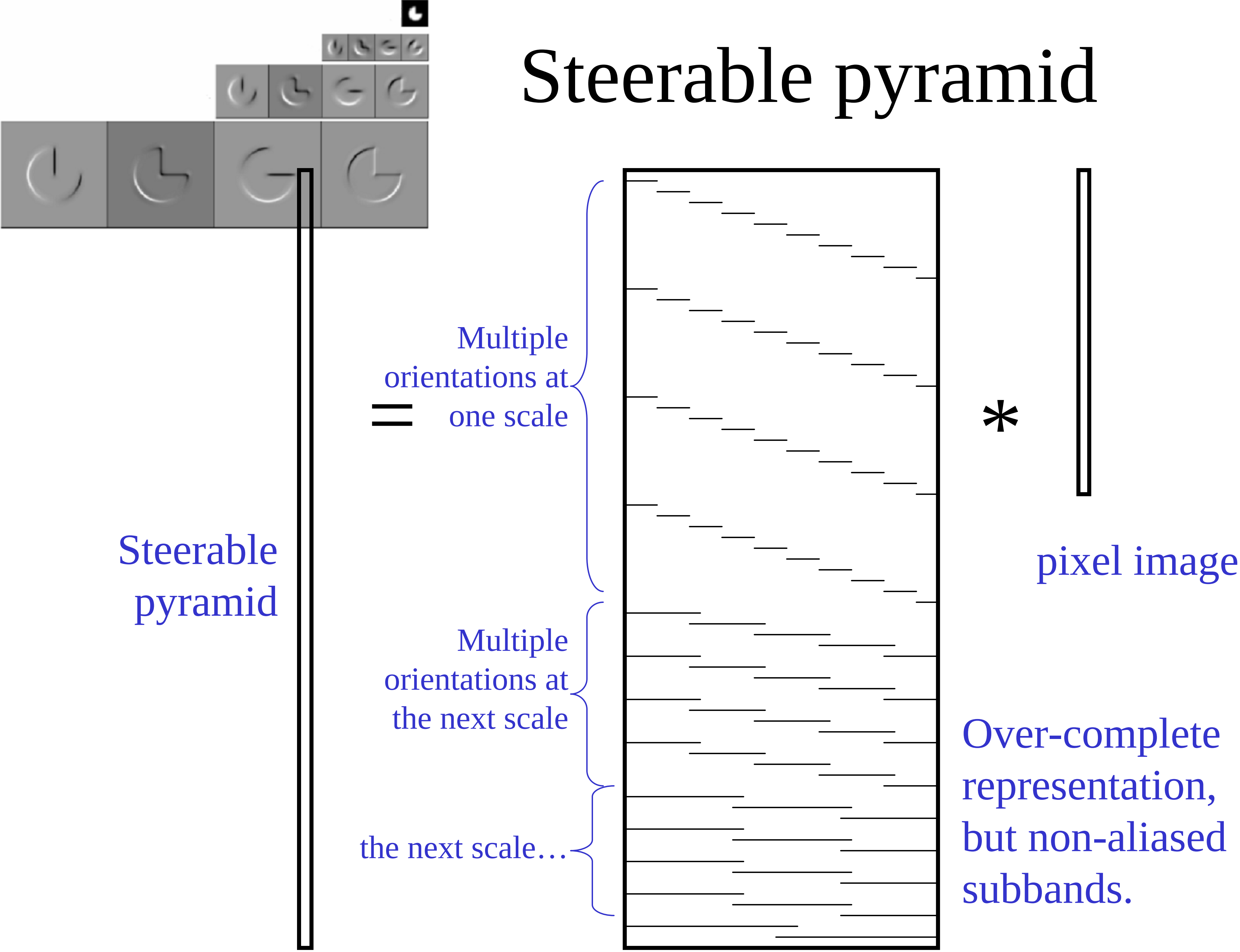
Wavelet pyramid




Ortho-normal transform (like Fourier transform), but with localized basis functions.

pixel image

# Steerable pyramid



# Matlab resources for pyramids (with tutorial)

http://www.cns.nyu.edu/~eero/software.html

lcv

Laboratory for Computational Vision

Home | People | Research | Publications | Software

## Publicly Available Software Packages

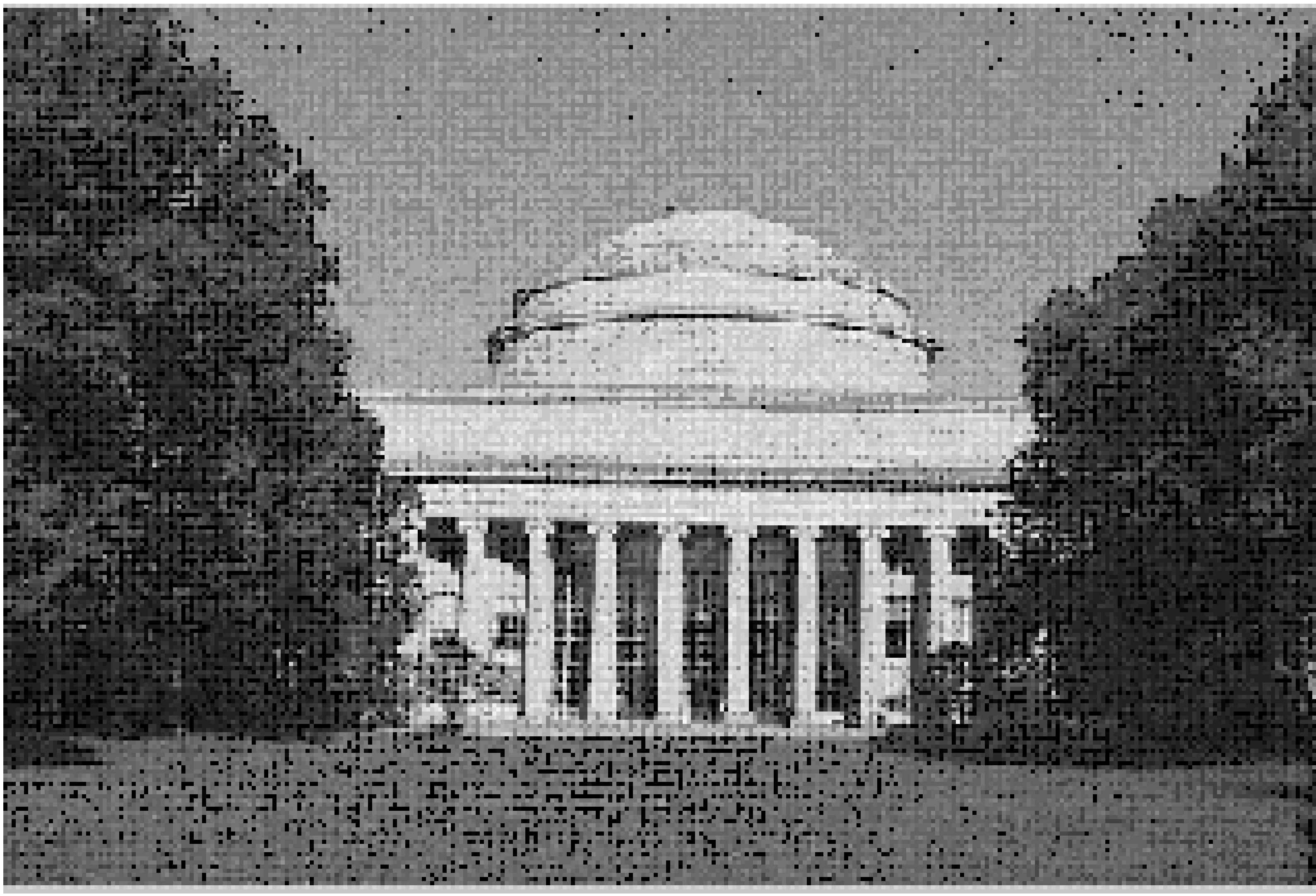
- Texture Analysis/Synthesis - Matlab code is available for analyzing and synthesizing visual textures. README | Contents | ChangeLog | Source code (UNIX/PC, gzip'ed tar file)
- EPWIC - Embedded Progressive Wavelet Image Coder. C source code available.
- **matlabPyrTools** - Matlab source code for multi-scale image processing. Includes tools for building and manipulating Laplacian pyramids, QMF/Wavelets, and steerable pyramids. Data structures are compatible with the Matlab wavelet toolbox, but the convolution code (in C) is faster and has many boundary-handling options. README, Contents, Modification list, UNIX/PC source or Macintosh source.
- The Steerable Pyramid, an (approximately) translation- and rotation-invariant multi-scale image decomposition. MatLab (see above) and C implementations are available.
- Computational Models of cortical neurons. Macintosh program available.
- EPIC - Efficient Pyramid (Wavelet) Image Coder. C source code available.
- OBVIUS [Object-Based Vision & Image Understanding System]: README / ChangeLog / Doc (225k) / Source Code (2.25M).
- CL-SHELL [Gnu Emacs <-> Common Lisp Interface]: README / Change Log / Source Code (119k).

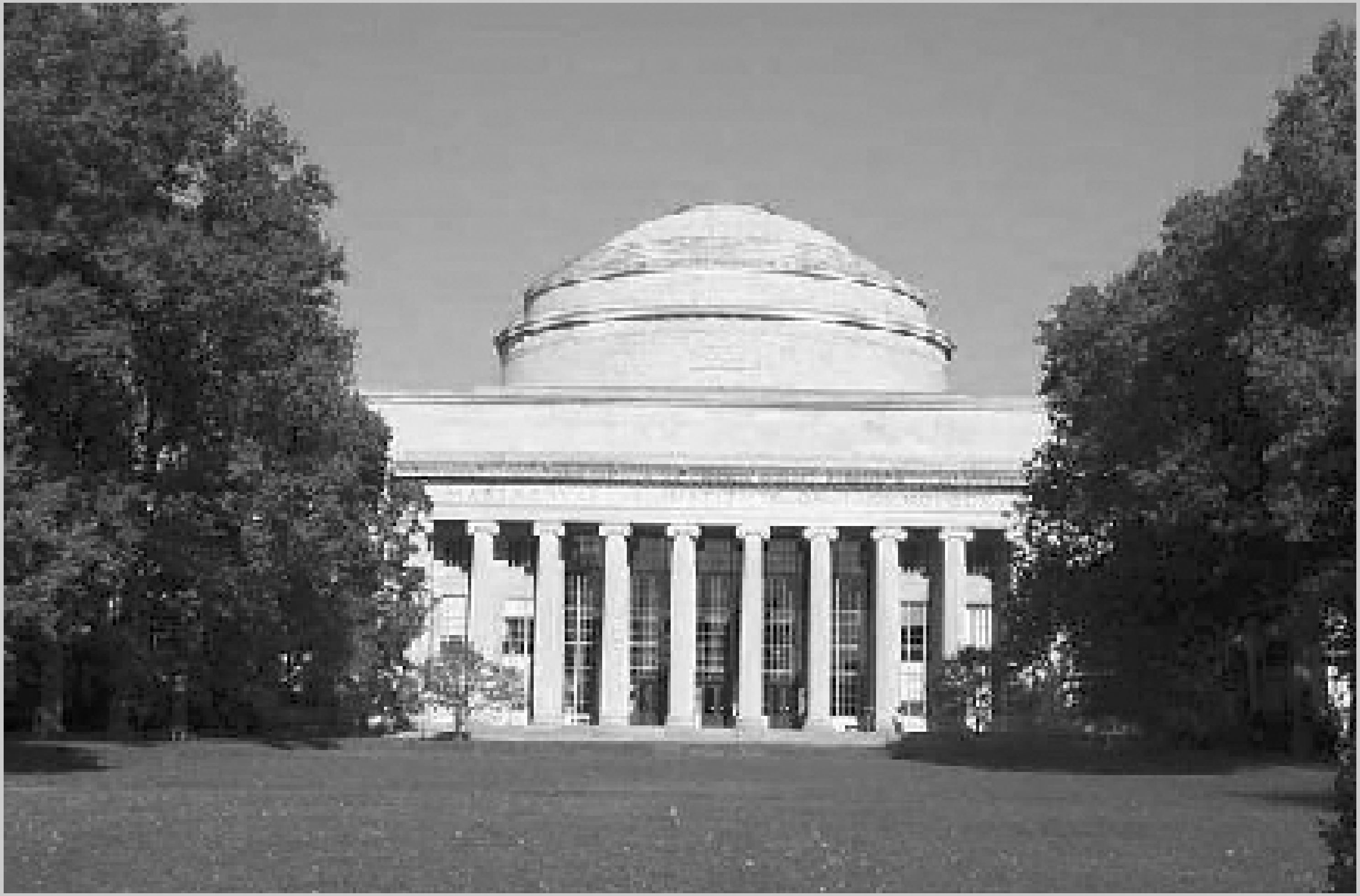
# Why use these representations?

- Handle real-world size variations with a constant-size vision algorithm.
- Remove noise
- Analyze texture
- Recognize objects
- Label image features

# An application of image pyramids: noise removal

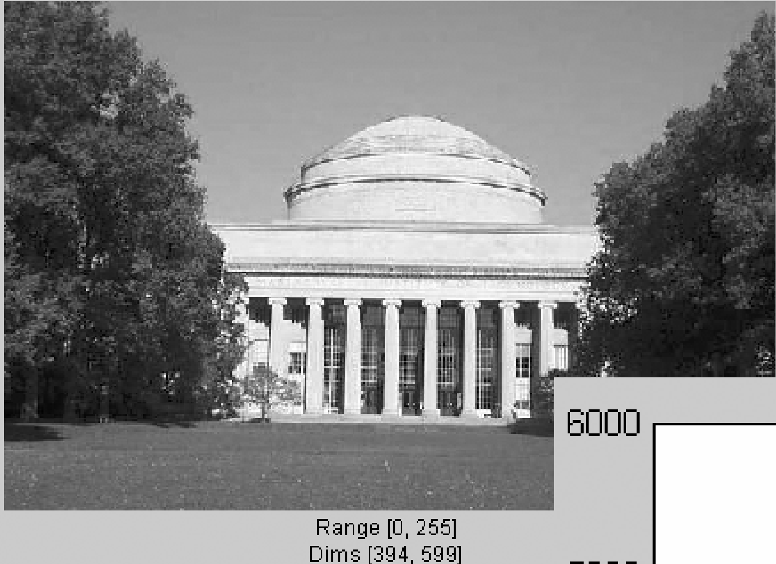
# Image statistics (or, mathematically, how can you tell image from noise?)

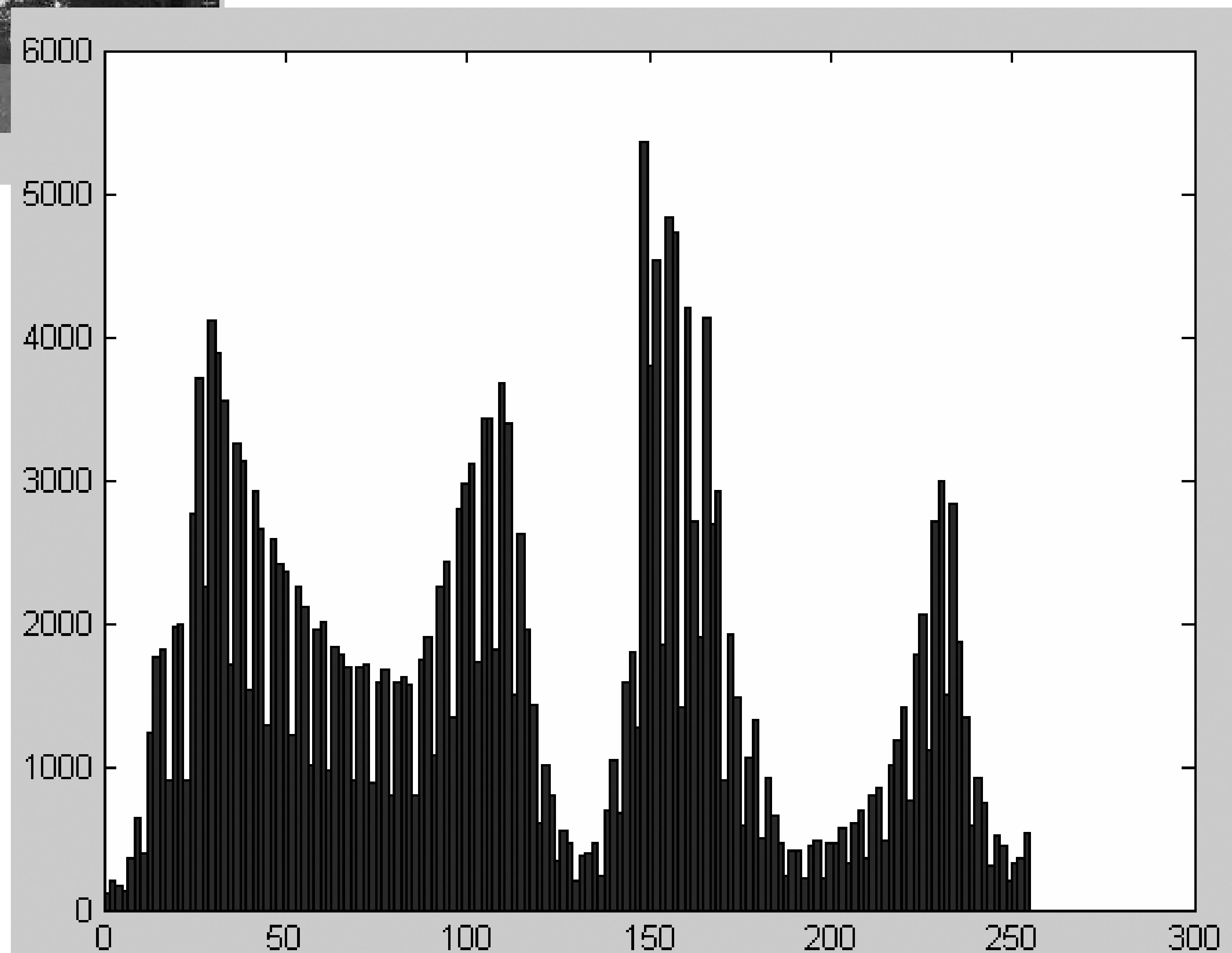




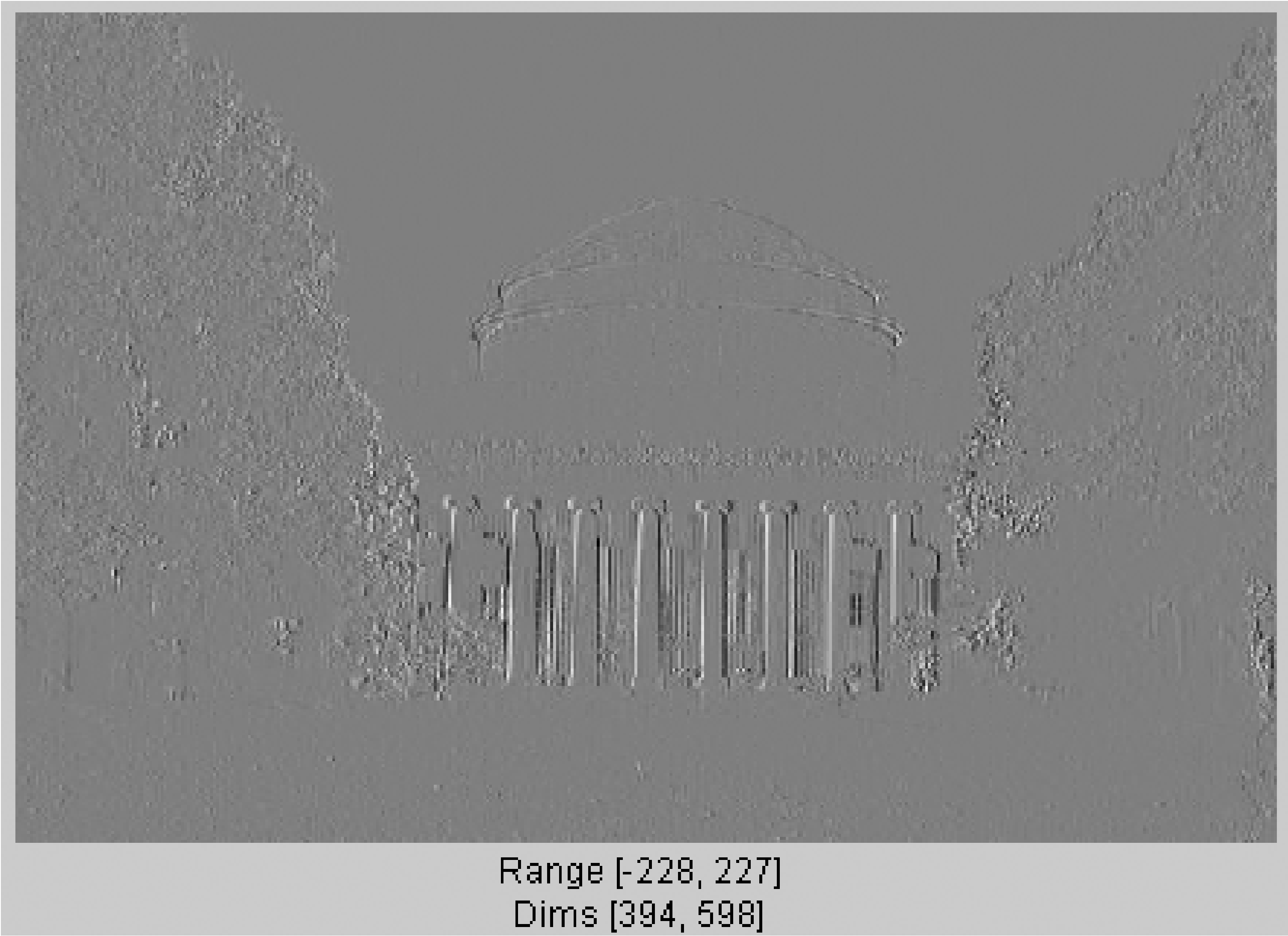
Range [0, 255]
Dims [394, 599]

# Pixel representation
# image histogram

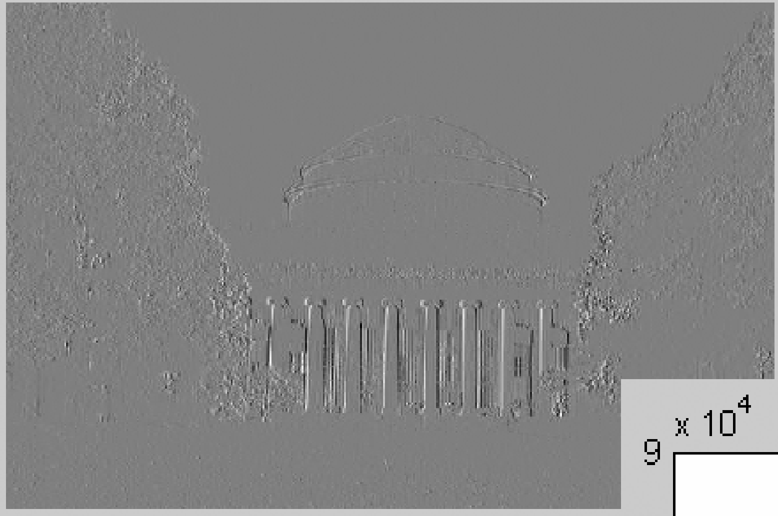


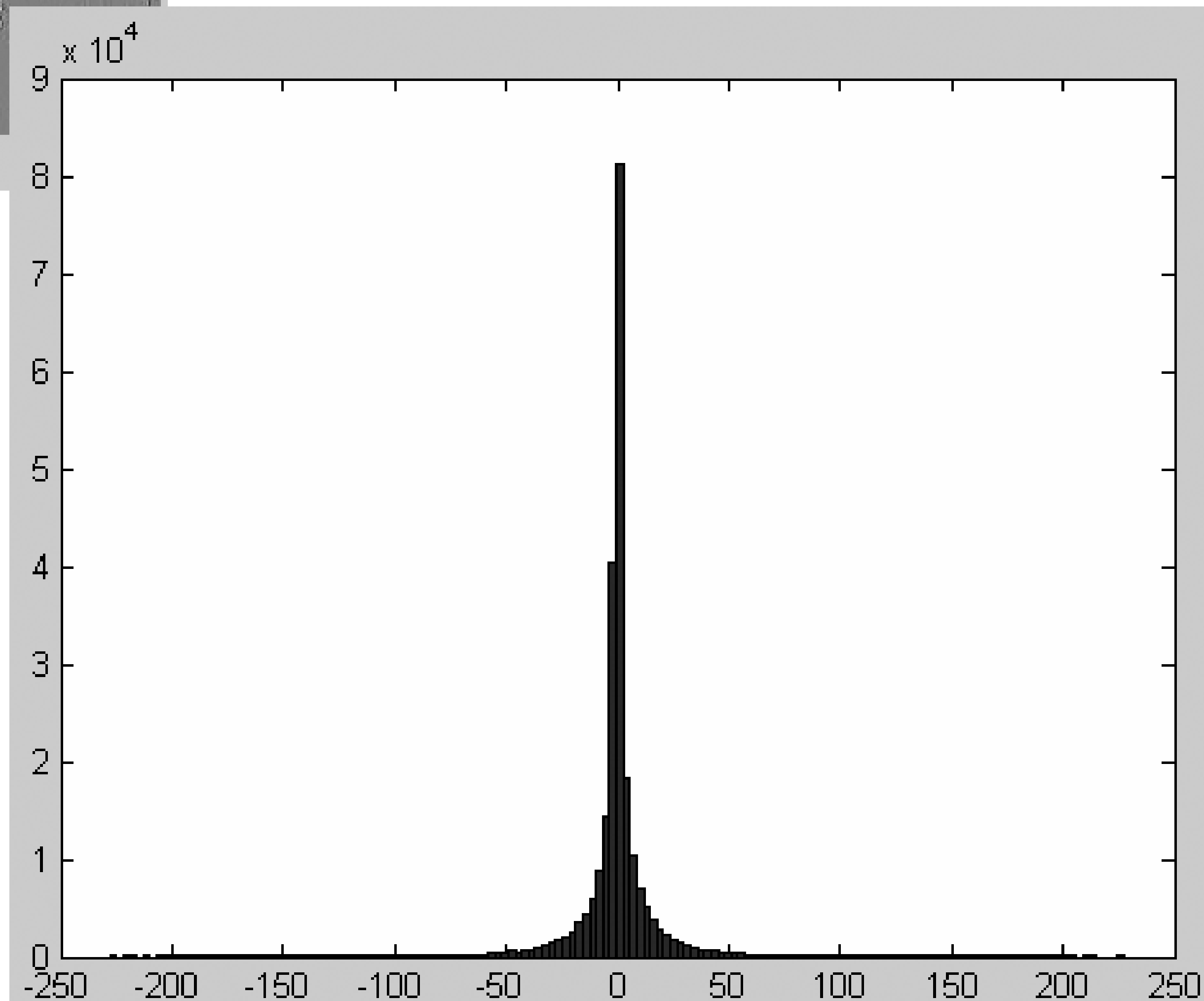



# bandpass filtered image

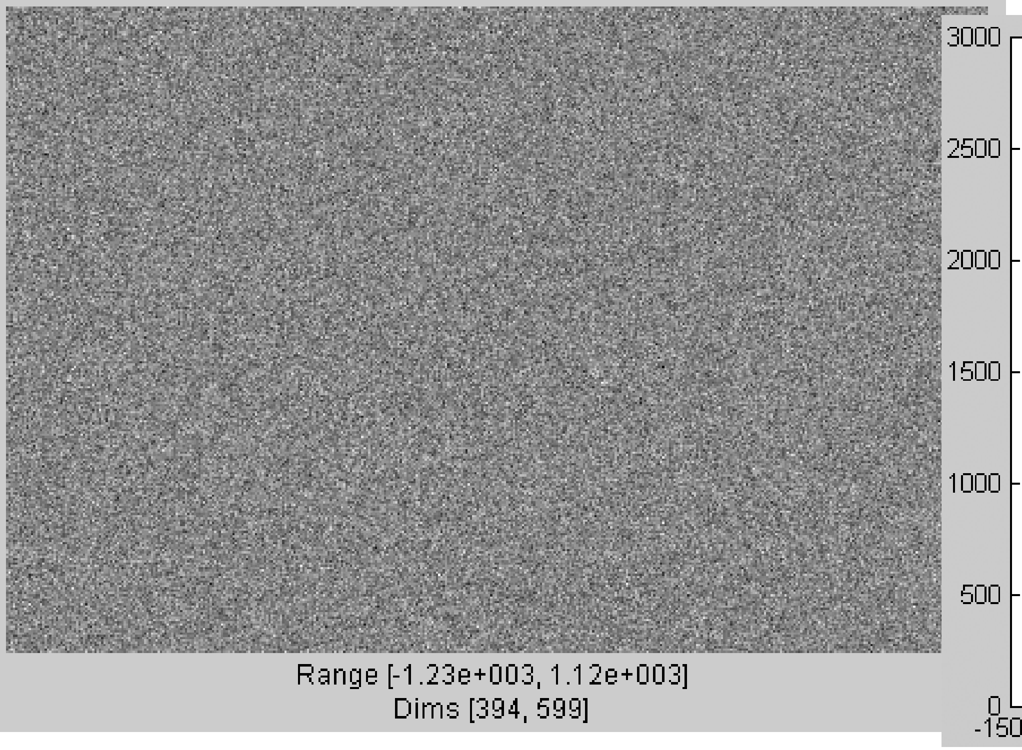


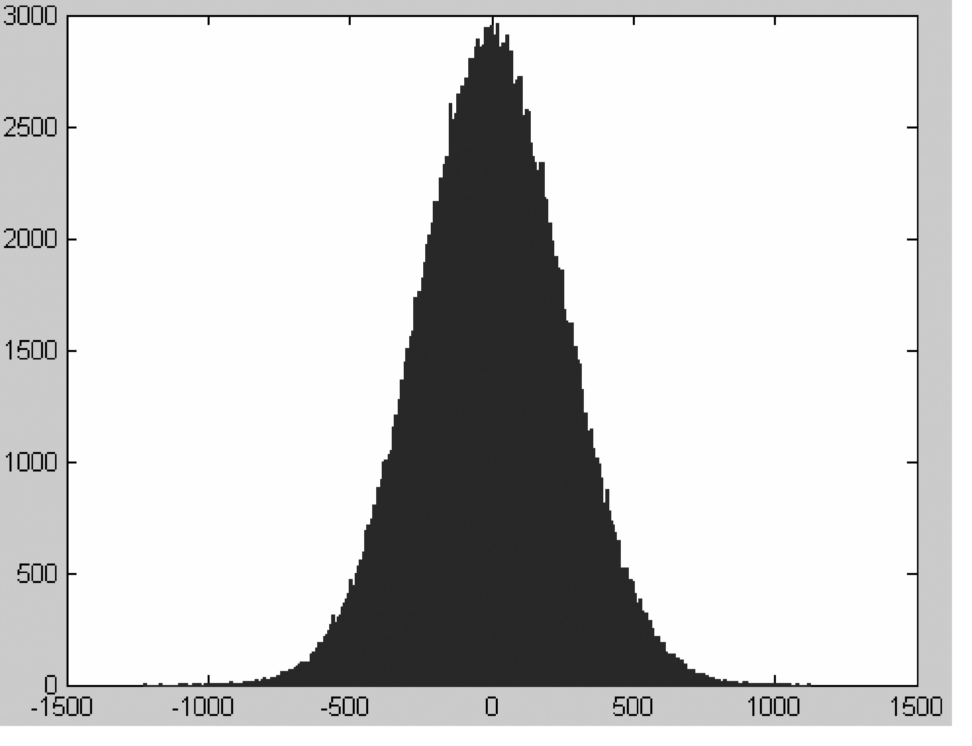
# bandpassed representation image histogram



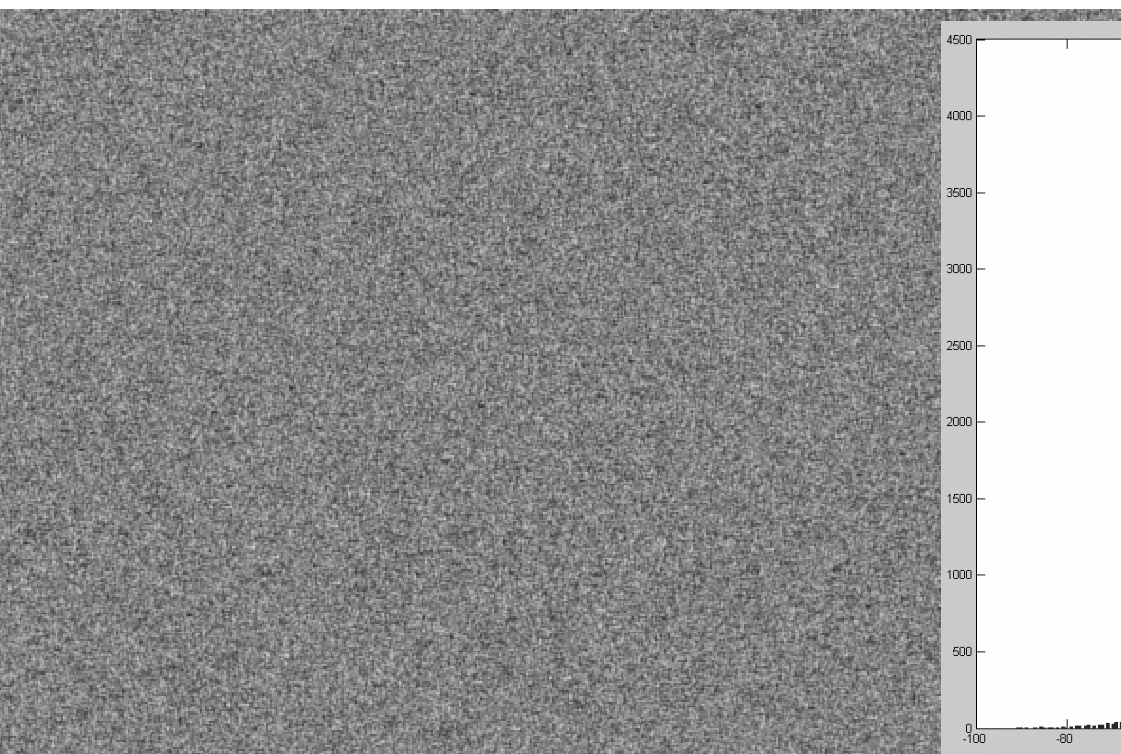



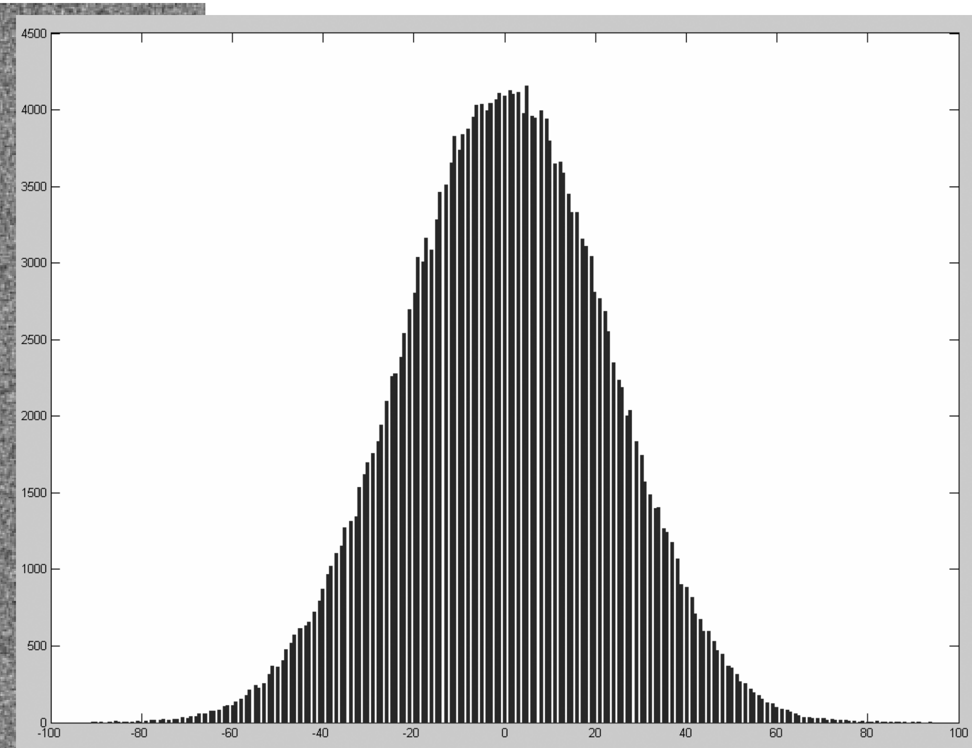
# Pixel domain noise image and histogram



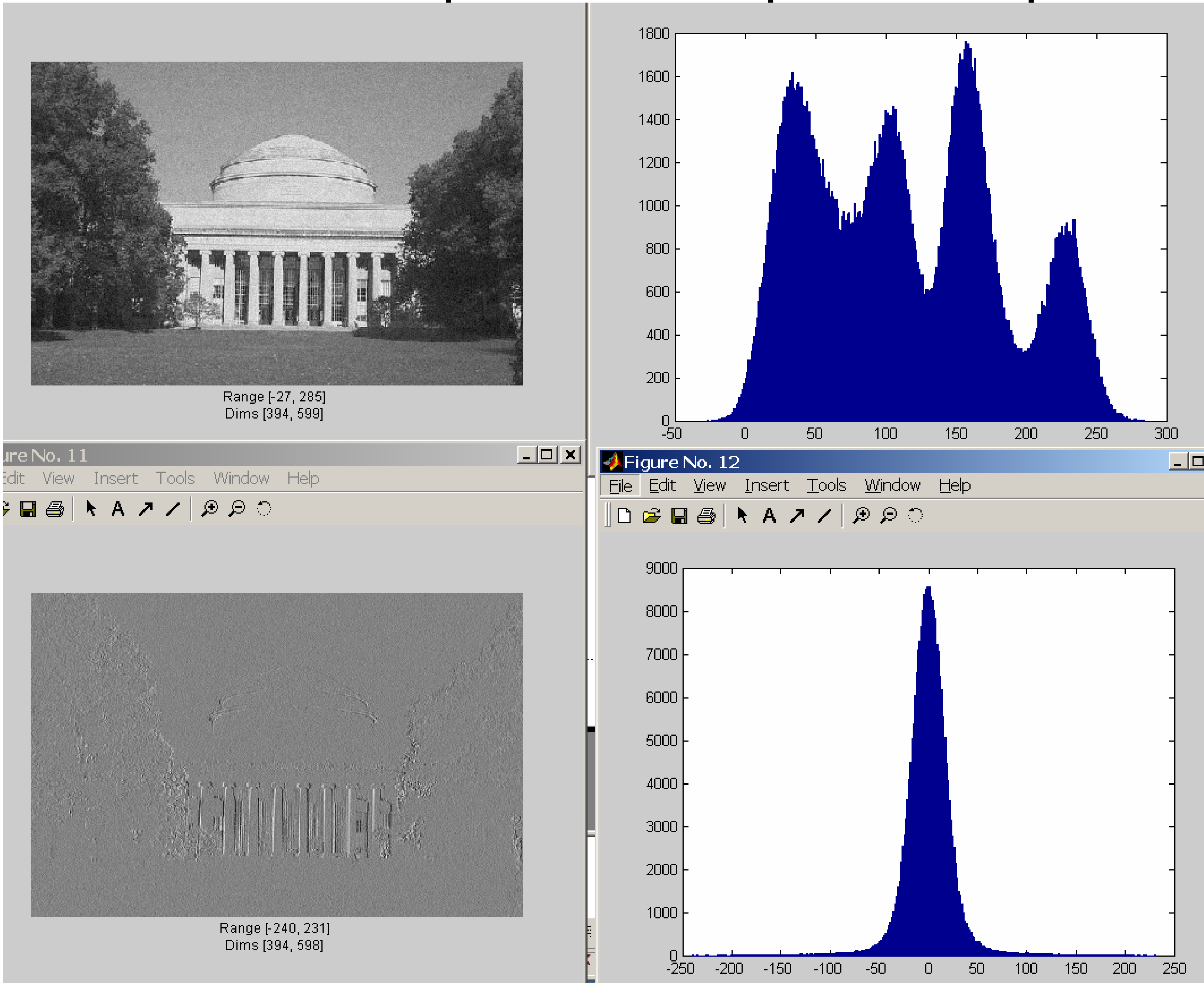



# Bandpass domain noise image and histogram





# Noise-corrupted full-freq and bandpass images



# Bayes theorem

$$P(x, y) = P(x|y)\, P(y)$$

so

$$P(x|y)\, P(y) = P(y|x)\, P(x)$$

and

$$P(x|y) = P(y|x)\, P(x) \,/\, P(y)$$

- $x$: The parameters you want to estimate
- $y$: What you observe
- $P(y|x)$: Likelihood function
- $P(x)$: Prior probability
- $P(y)$: Constant w.r.t. parameters x.

# Bayesian MAP estimator for clean bandpass coefficient values

Let x = bandpassed image value before adding noise.

Let y = noise-corrupted observation.

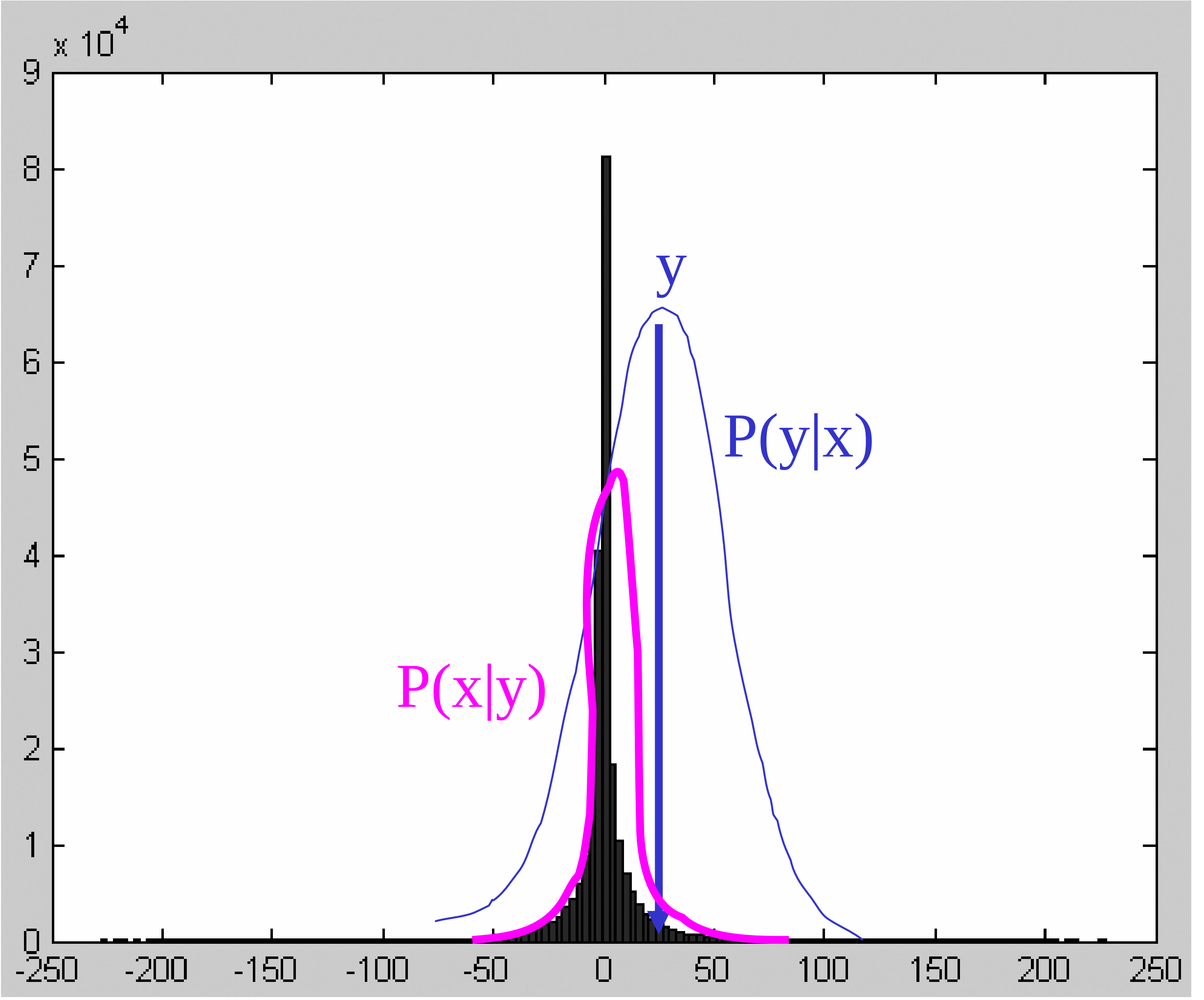
By Bayes theorem

P(x|y) = k P(y|x) P(x)

P(x)

P(y|x)

P(x|y)



# Bayesian MAP estimator

Let x = bandpassed image value before adding noise.

Let y = noise-corrupted observation.

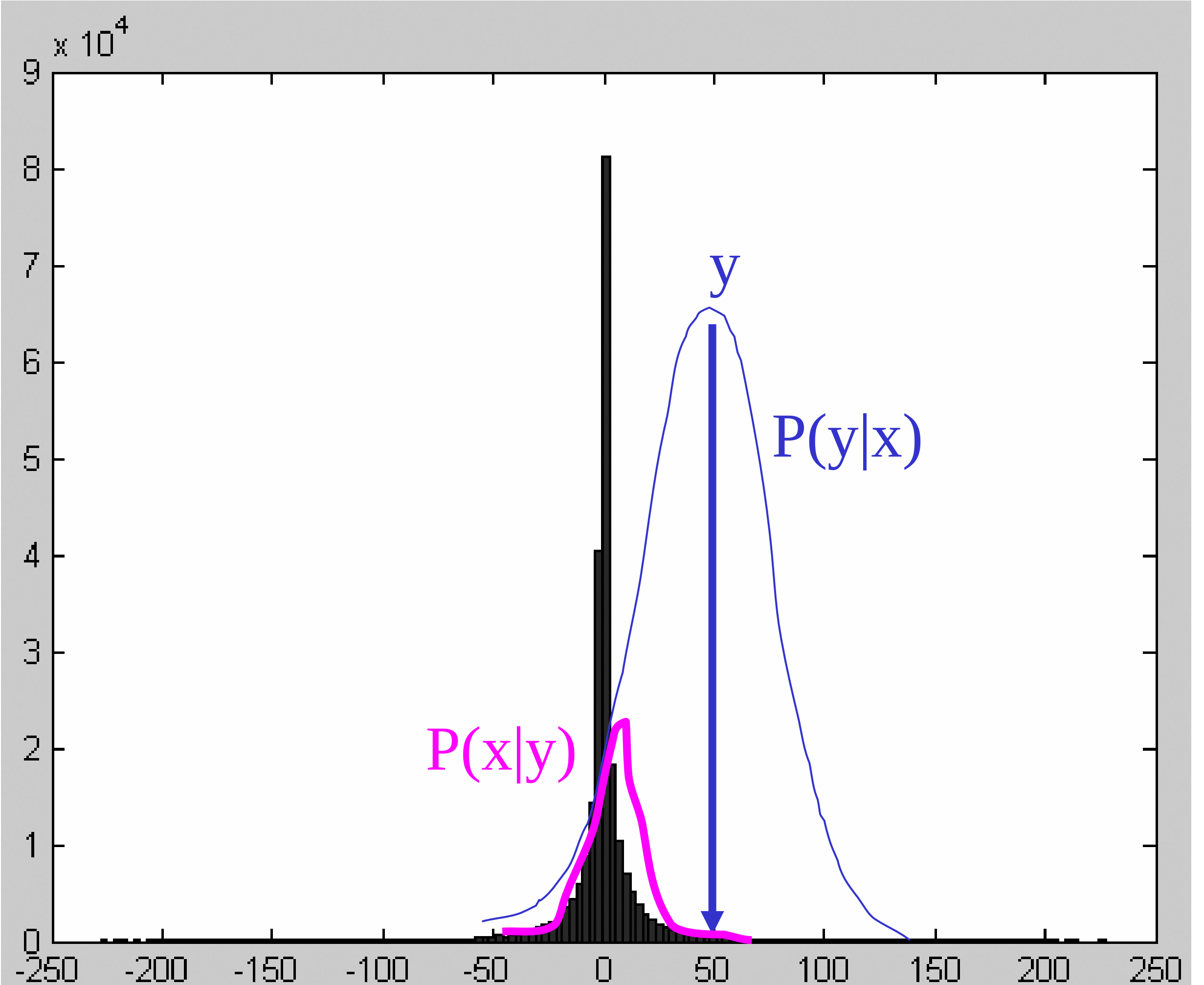
By Bayes theorem

P(x|y) = k P(y|x) P(x)

P(x)

P(y|x)

P(x|y)



# Bayesian MAP estimator

Let x = bandpassed image value before adding noise.

Let y = noise-corrupted observation.

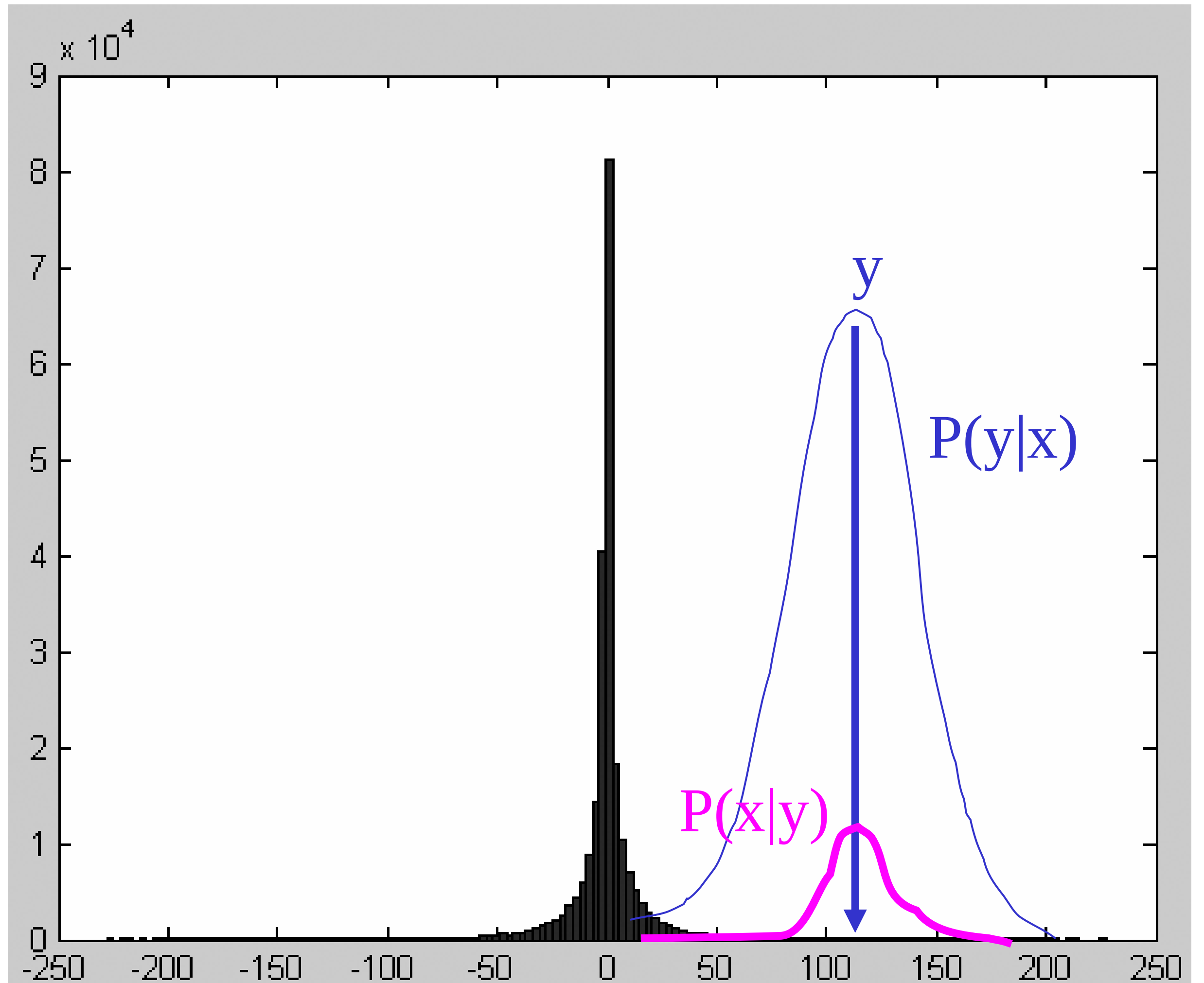
By Bayes theorem

P(x|y) = k P(y|x) P(x)

P(x)

P(y|x)

P(x|y)



# MAP estimate, $\hat{x}$, as function of observed coefficient value, y

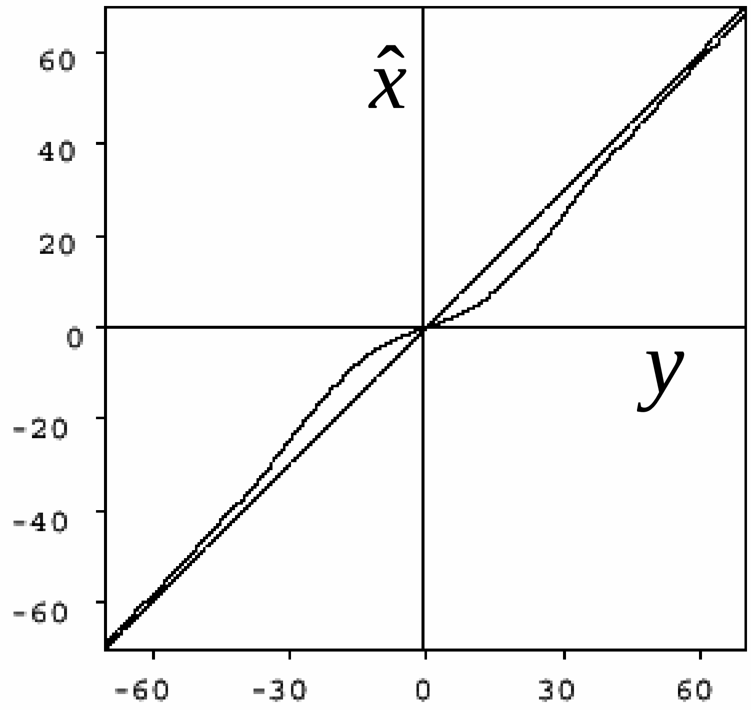



**Figure 2:** Bayesian estimator (symmetrized) for the signal and noise histograms shown in figure 1. Superimposed on the plot is a straight line indicating the identity function.

http://www-bcs.mit.edu/people/adelson/pub_pdfs/simoncelli_noise.pdf

**Simoncelli and Adelson, Noise Removal via Bayesian Wavelet Coring**

# Noise removal results

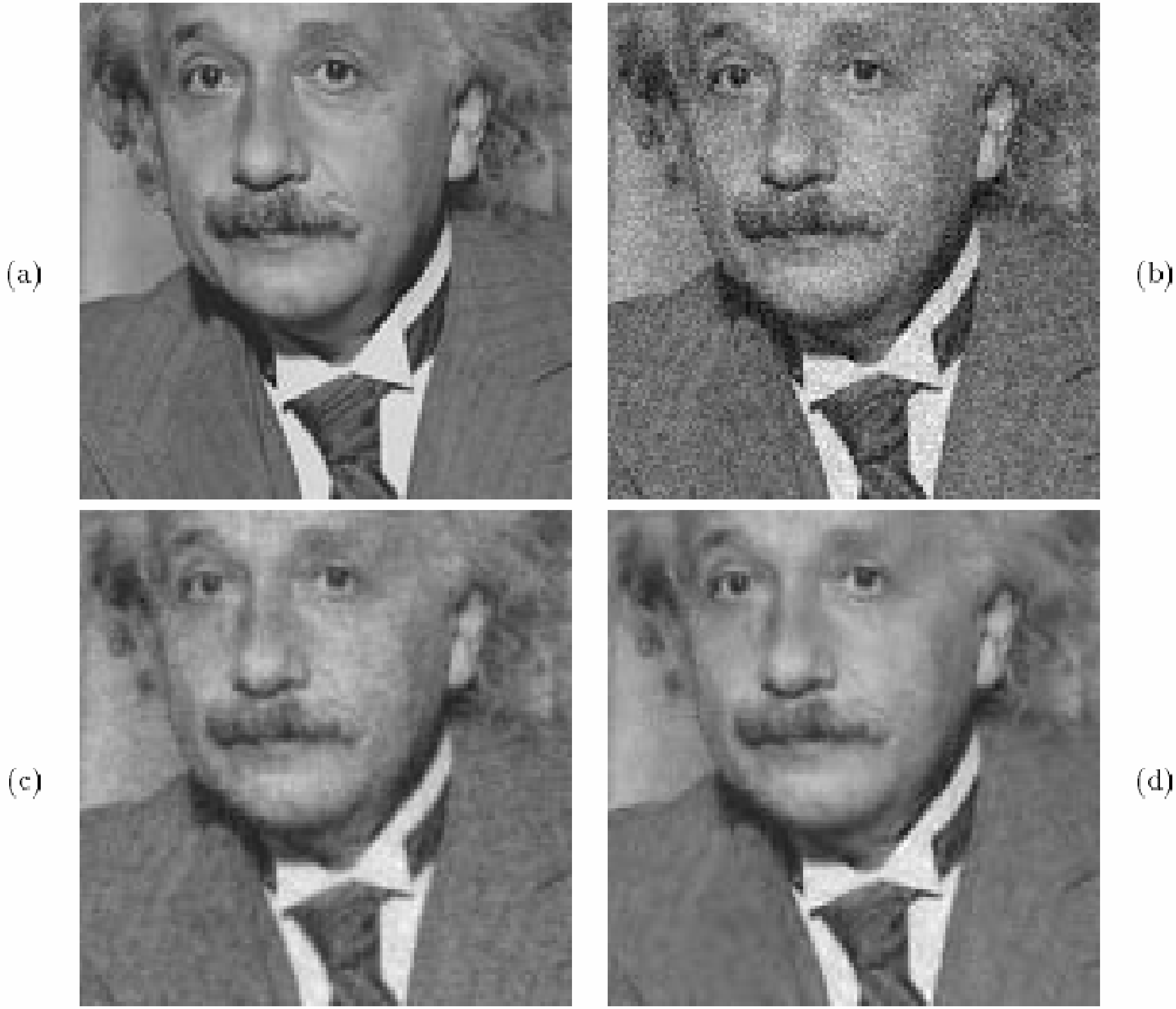



**Figure 4:** Noise reduction example. (a) Original image (cropped). (b) Image contaminated with additive Gaussian white noise (SNR = 9.00dB). (c) Image restored using (semi-blind) Wiener filter (SNR = 11.88dB). (d) Image restored using (semi-blind) Bayesian estimator (SNR = 13.82dB).

http://www-bcs.mit.edu/people/adelson/pub_pdfs/simoncelli_noise.pdf

**Simoncelli and Adelson, Noise Removal via Bayesian Wavelet Coring**

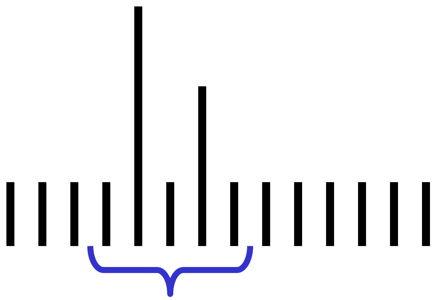
# Insert hany farid slides

# Non-linear filtering example

# Median filter

Replace each pixel by the median over N pixels (5 pixels, for these examples). Generalizes to “rank order” filters.

In:

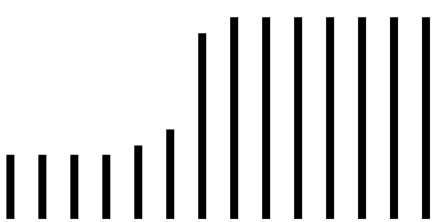


5-pixel neighborhood

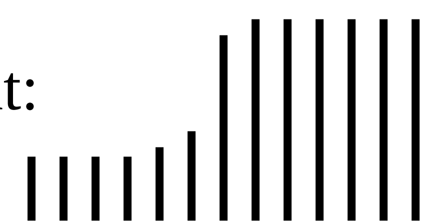
Out:

Spike noise is removed

In:

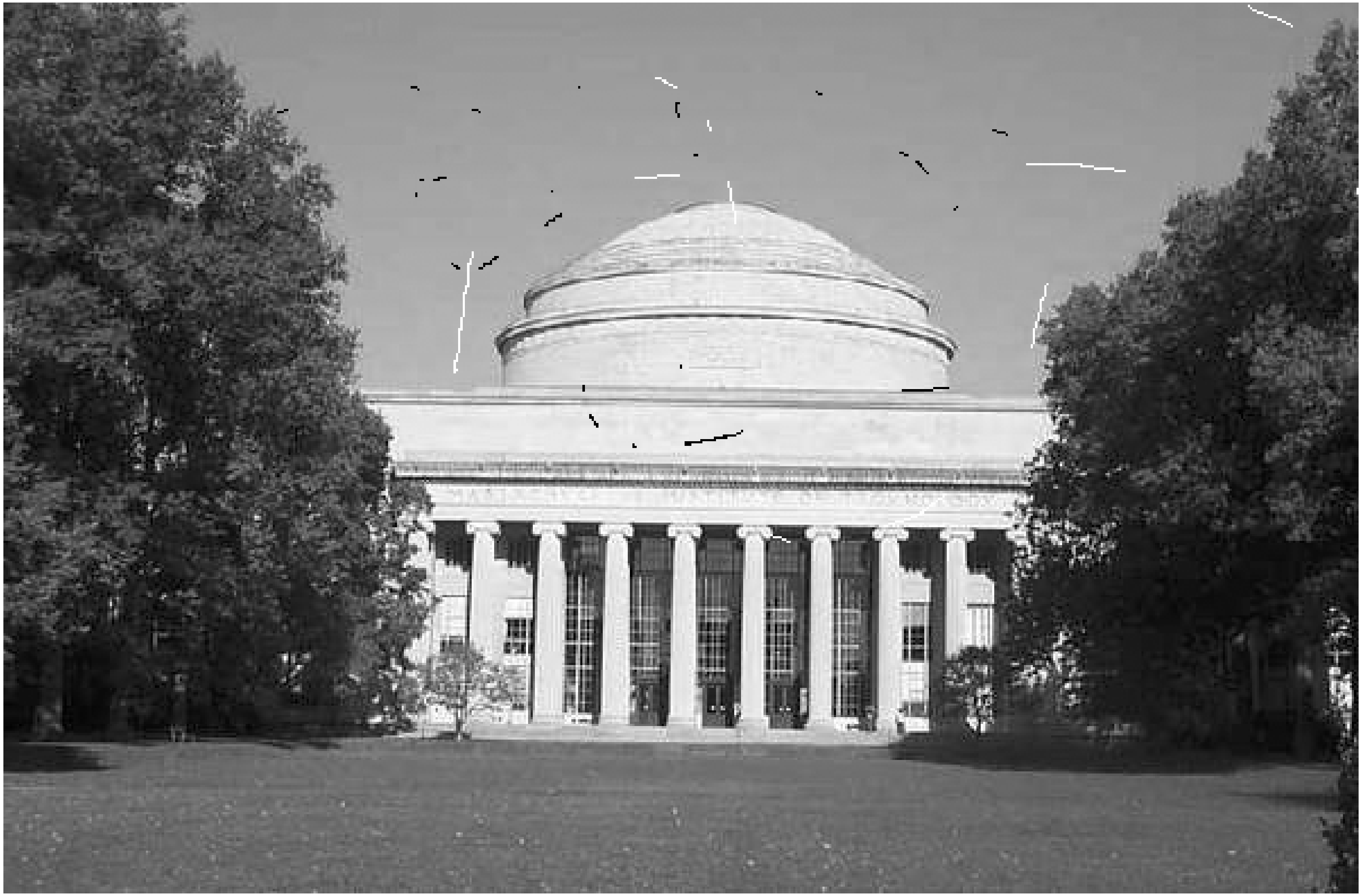


Out:



Monotonic edges remain unchanged

# Degraded image



# Radius 1 median filter

# Radius 2 median filter

# CCD color sampling

# Color sensing, 3 approaches

- Scan 3 times (temporal multiplexing)
- Use 3 detectors (3-ccd camera, and color film)
- Use offset color samples (spatial multiplexing)

# Typical errors in temporal multiplexing approach

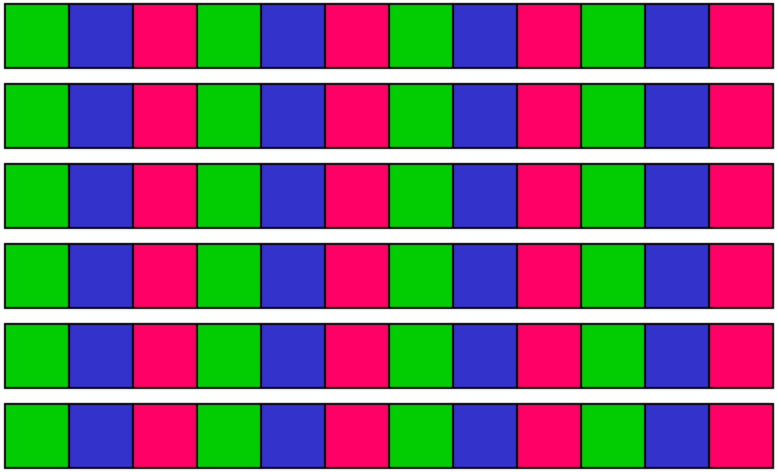
- Color offset fringes

# Typical errors in spatial multiplexing approach.

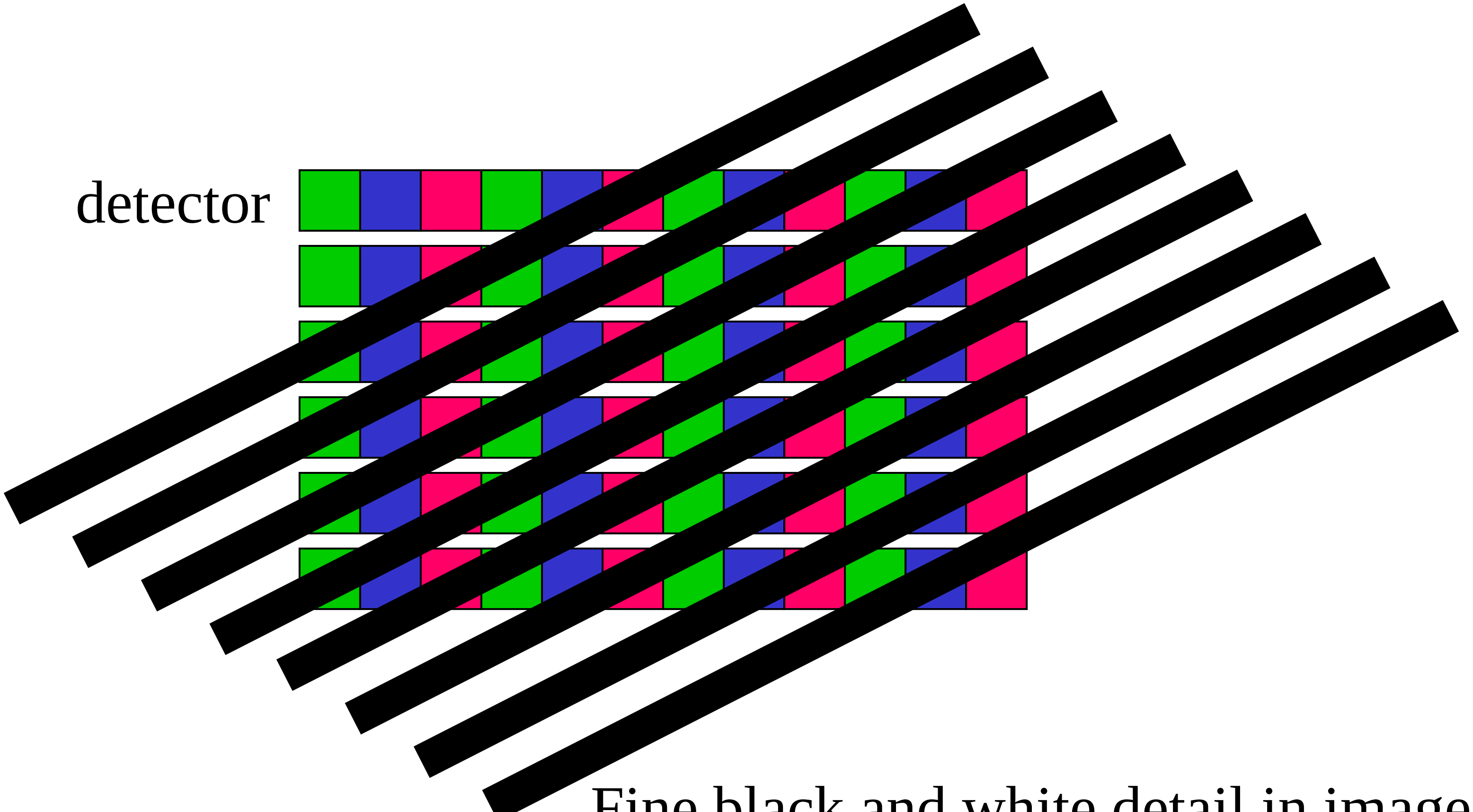
- Color fringes.

# CCD color filter pattern

detector



# The cause of color moire

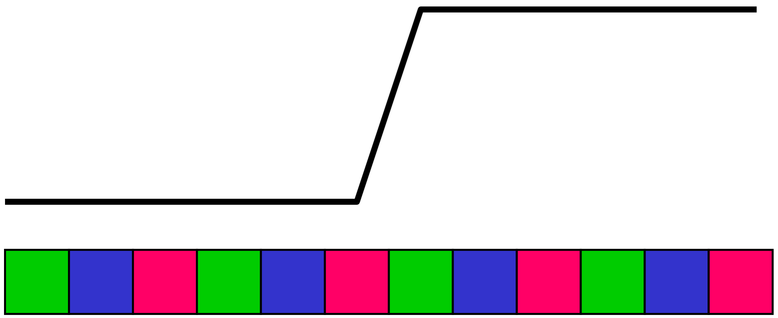



Fine black and white detail in image
mis-interpreted as color information.

# Black and white edge falling on color CCD detector

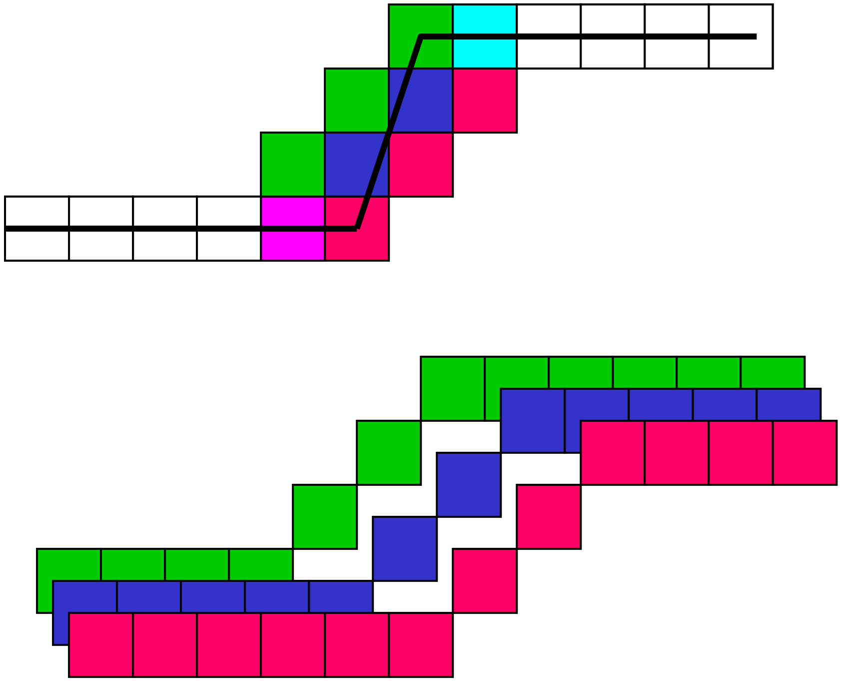
Black and white image (edge)

Detector pixel colors



# Color sampling artifact

Interpolated pixel colors,
for grey edge falling on colored
detectors (linear interpolation).

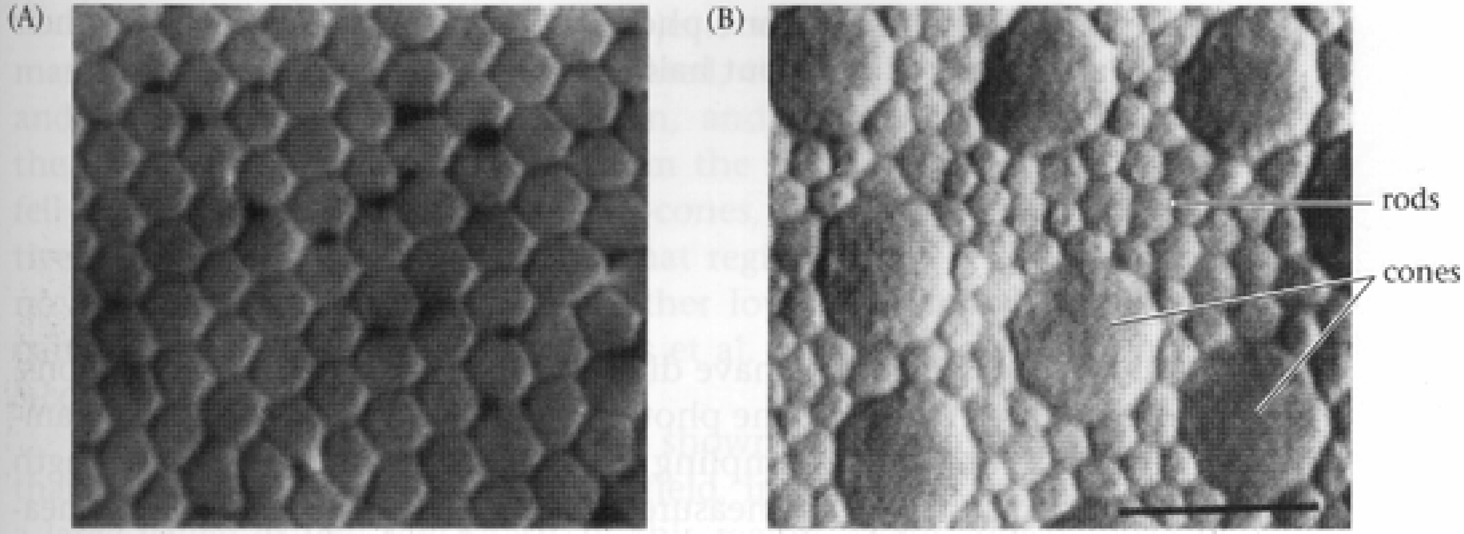


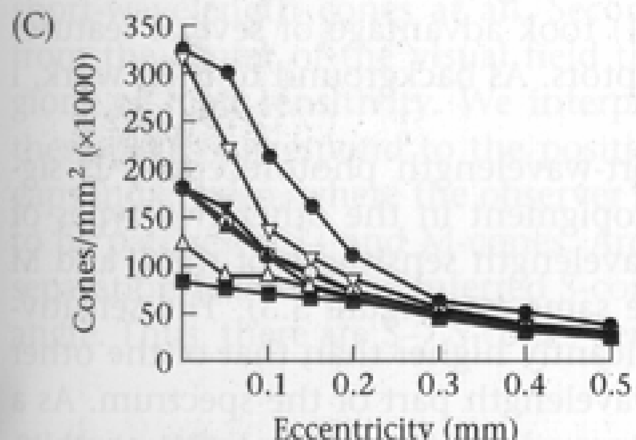
# Typical color moire patterns

Blow-up of electronic camera image. Notice spurious colors in the regions of fine detail in the plants.

# Color sampling artifacts

# Human Photoreceptors



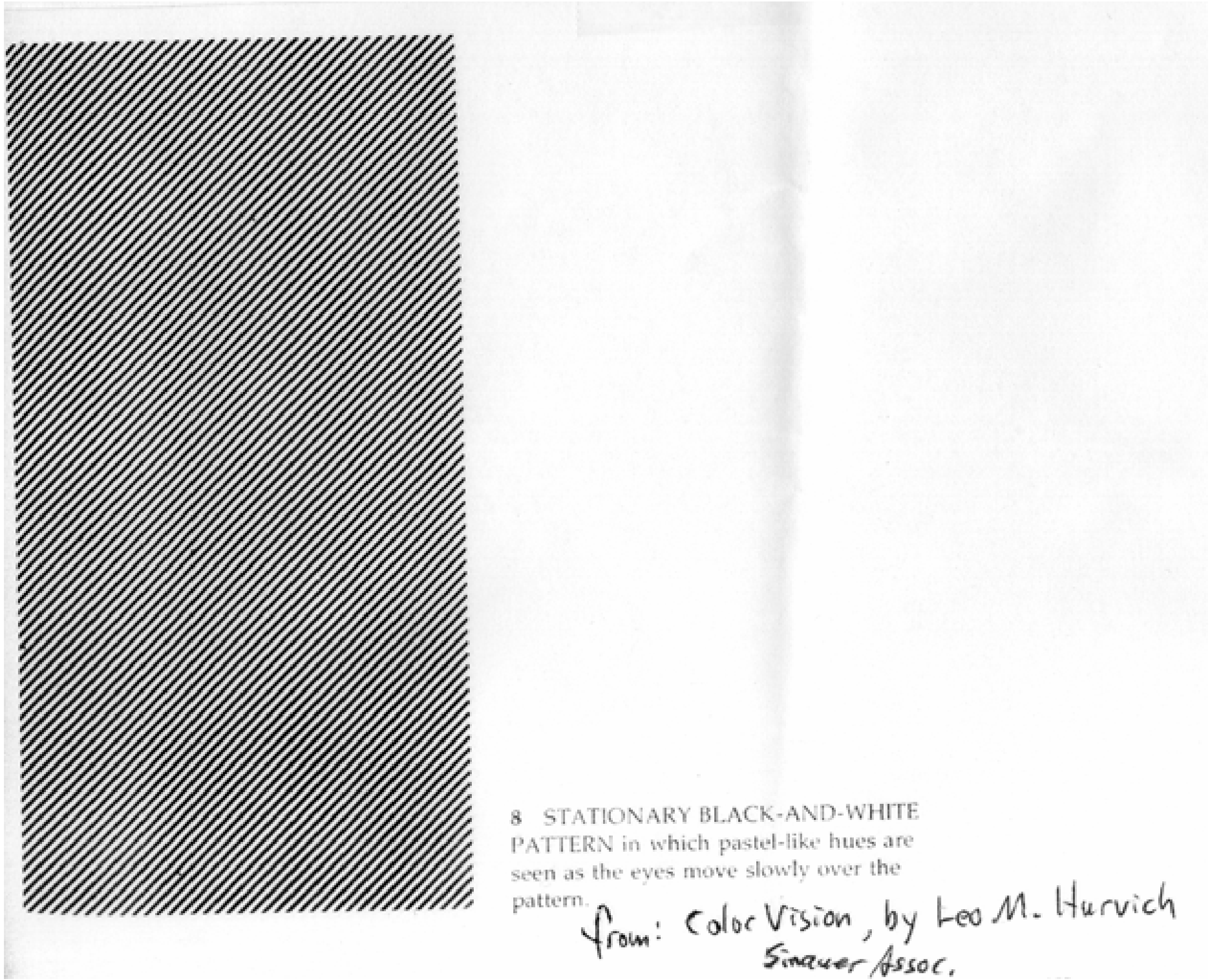




**3.4 THE SPATIAL MOSAIC OF THE HUMAN CONES.** Cross sections of the human retina at the level of the inner segments showing (A) cones in the fovea, and (B) cones in the periphery. Note the size difference (scale bar = 10 $\mu$m), and that, as the separation between cones grows, the rod receptors fill in the spaces. (C) Cone density plotted as a function of distance from the center of the fovea for seven human retinas; cone density decreases with distance from the fovea. Source: Curcio et al., 1990.

(From Foundations of Vision, by Brian Wandell, Sinauer Assoc.)

# Brewster's colors example (subtle).

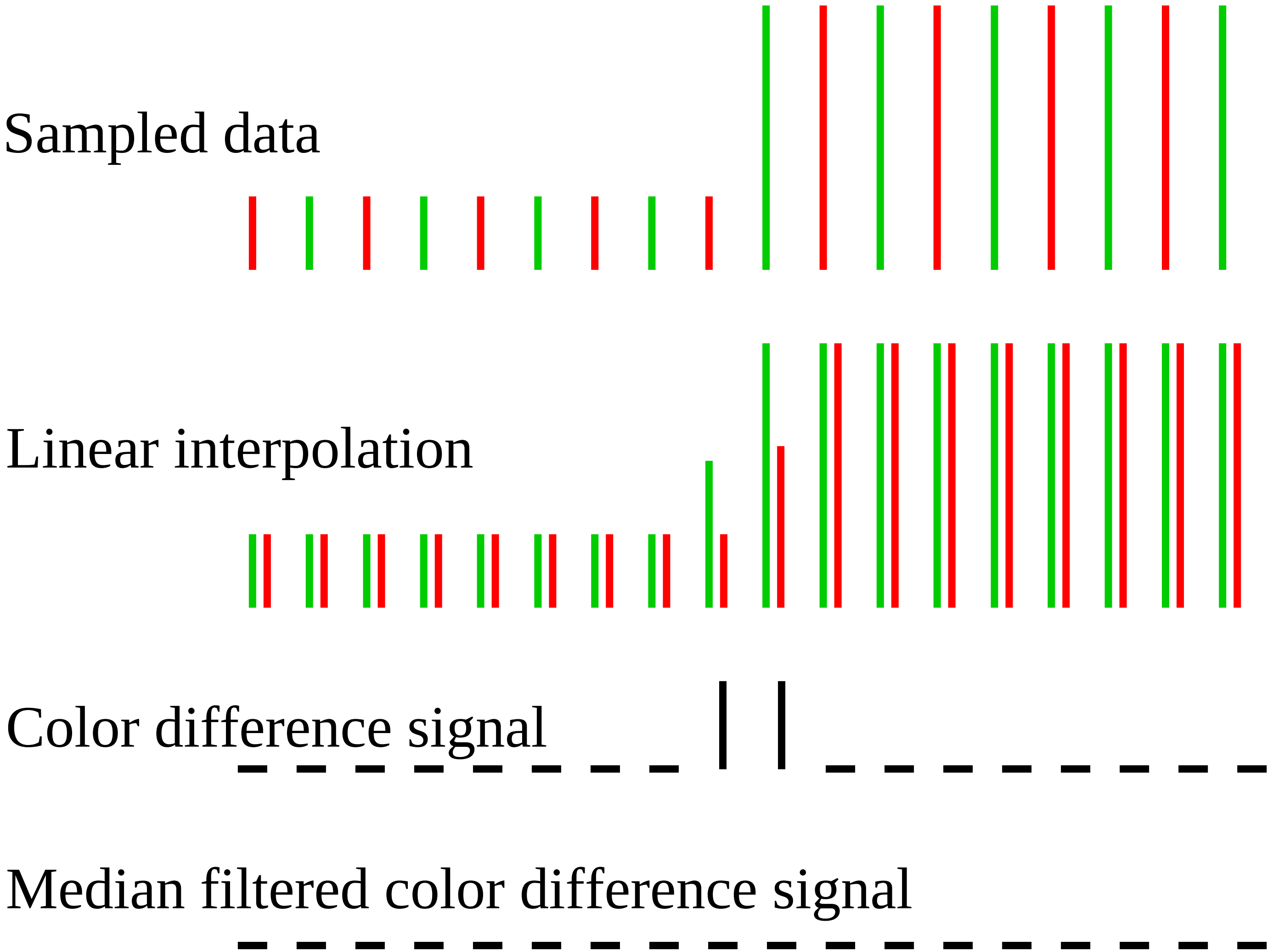
Scale relative to human photoreceptor size: each line covers about 7 photoreceptors.



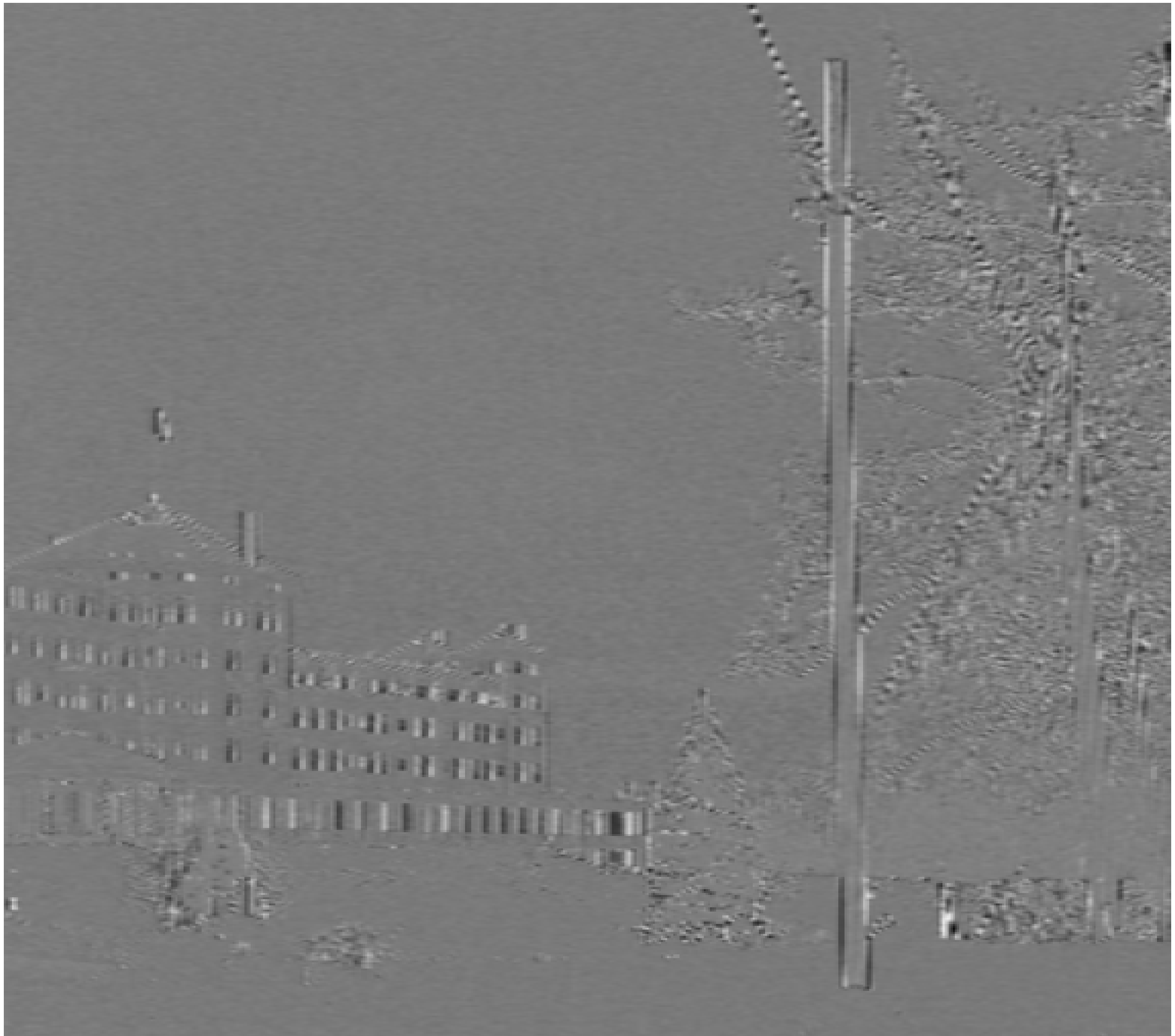
# Median Filter Interpolation

- Perform first interpolation on isolated color channels.
- Compute color difference signals.
- Median filter the color difference signal.
- Reconstruct the 3-color image.

# Two-color sampling of BW edge



# R-G, after linear interpolation



# R – G, median filtered (5x5)

# Recombining the median filtered colors

Linear interpolation

Median filter interpolation