CS9840
Machine Learning in Computer Vision
Olga Veksler

Lecture 4

Image Representation

Outline

- How to represent an image as a feature vector?
- Histogram based representation
  - Based on intensity, edge, texture
- Global vs. Local histograms
- Spatial pyramids

Pixel Representations

- Intensity image
  - One value per pixel
  - pile all values into one vector, say in row order

10	12	30
20	17	21
33	91	24

$$\rightarrow \begin{bmatrix} 10 \\ 12 \\ 30 \\ 20 \\ 17 \\ 21 \\ 33 \\ 91 \\ 24 \end{bmatrix}$$

Pixel Representations

- Small change in image appearance

Slide by Erik Learned-Miller

Pixel Representations

- Leads to a large change in feature vector

difference image

10	12	30
20	17	21
33	91	24

9	10	12
19	20	17
32	33	91

[10 12 30 20 17 21 33 91 24]

[9 10 12 19 20 17 32 33 91]

Slide modified from Erik Learned-Miller

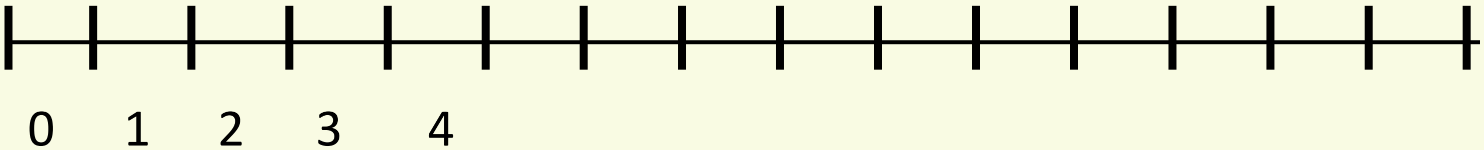
Pixel Representations

- Pixelwise representations:
  overly sensitive to position
- Nevertheless it has been successfully used in applications
  - eigenfaces, the first successful face detection system

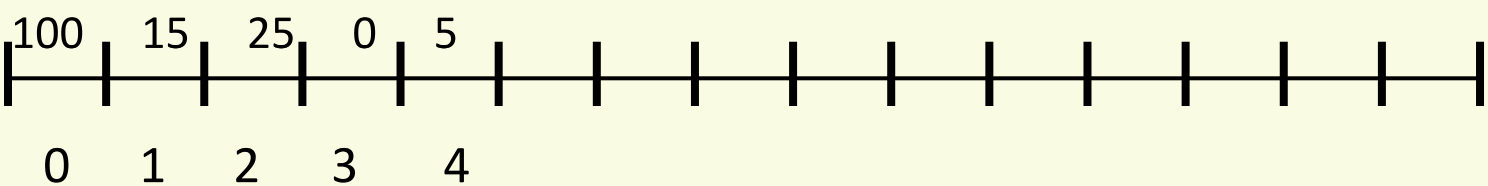
Slide by Erik Learned-Miller

Global Intensity Histogram

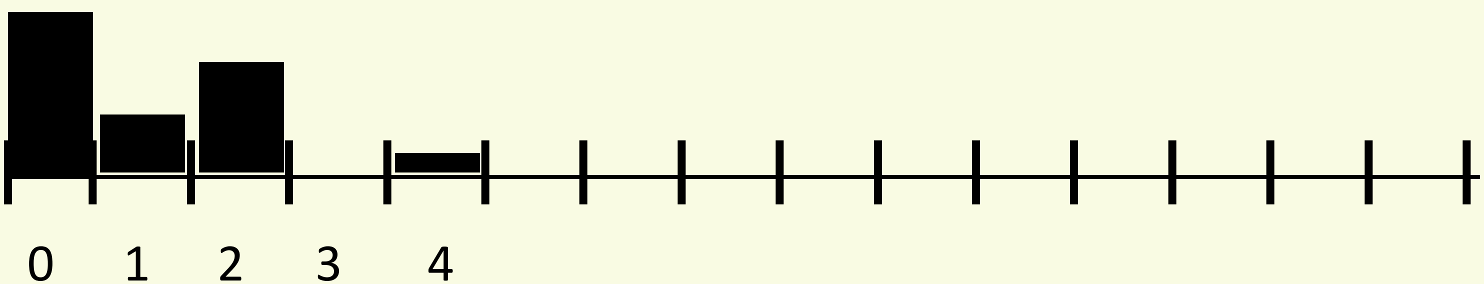
- Think of each intensity value as a "bin"



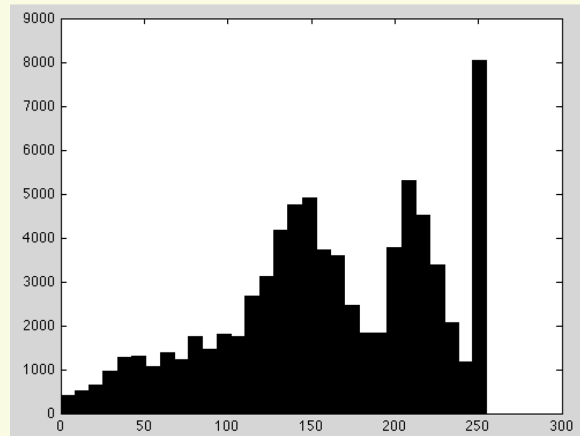
- Histogram counts the number of values that fall in each bin



- Visual plot:



Global Intensity Histogram



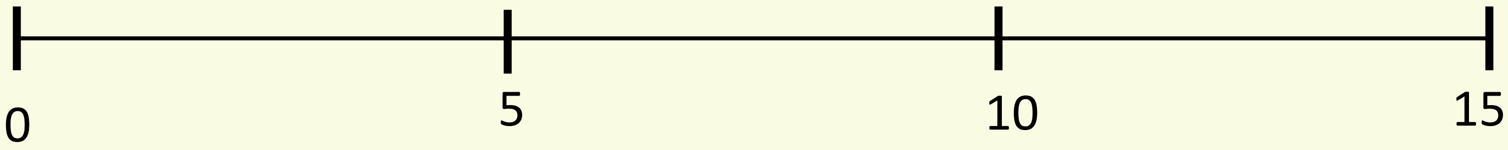
$$\begin{bmatrix} 5 \\ 10 \\ 4 \\ \ldots \\ 100 \end{bmatrix}$$

$$\begin{bmatrix} 0.04 \\ 0.08 \\ 0.03 \\ \ldots \\ 0.8 \end{bmatrix}$$

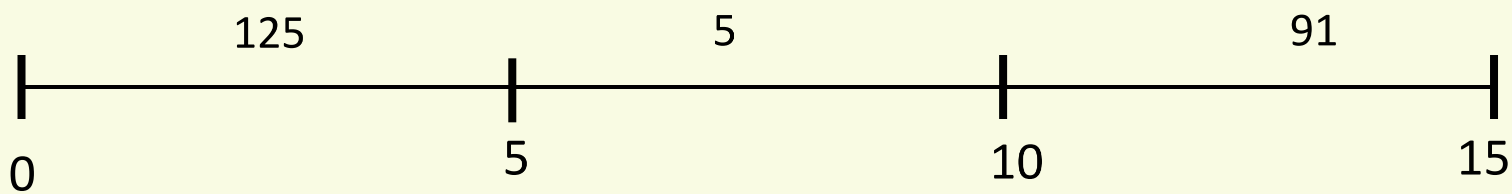
- Insensitive to changes in pixel location
- Often use normalized histogram
  - sums up to 1

Global Intensity Histogram Quantization

- Can quantize intensities (larger bins)



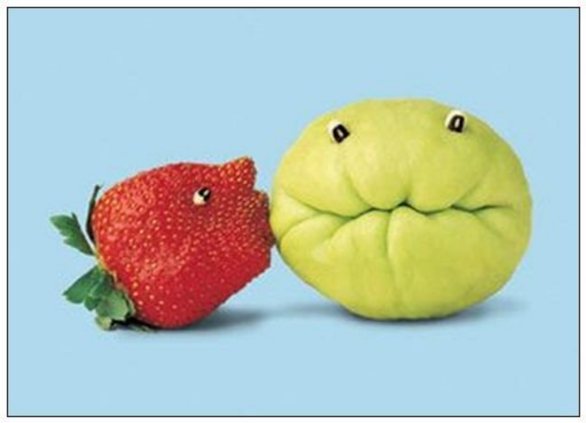
- Histogram: count number of values that fall in each bin



- Quantization
  - helps to improve efficiency
  - groups similar values together (i.e. removes fine distinction) which may help for recognition

Other Image Features

- Intensity is not enough for most applications
- Other often used features:



Color:
3 values per pixel

Edge:
1 value per pixel

Texture:
$\approx$ 48 values per pixel

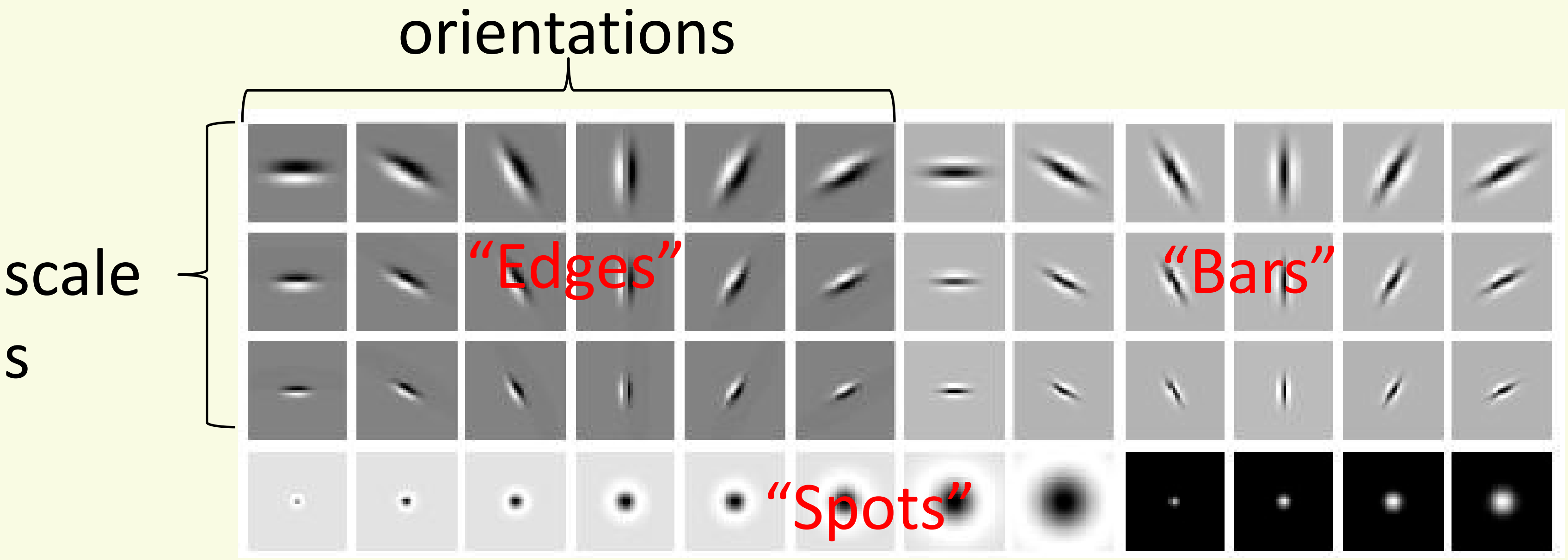
Right features depend on what you want to know

- Object: 2D shape
  - Local shape info, shading, shadows, texture
- Scene : overall layout
  - linear perspective, gradients
- Material properties: albedo, feel, hardness, …
  - Color, texture
- Motion
  - Optical flow, tracked points

Slide by Derek Hoiem

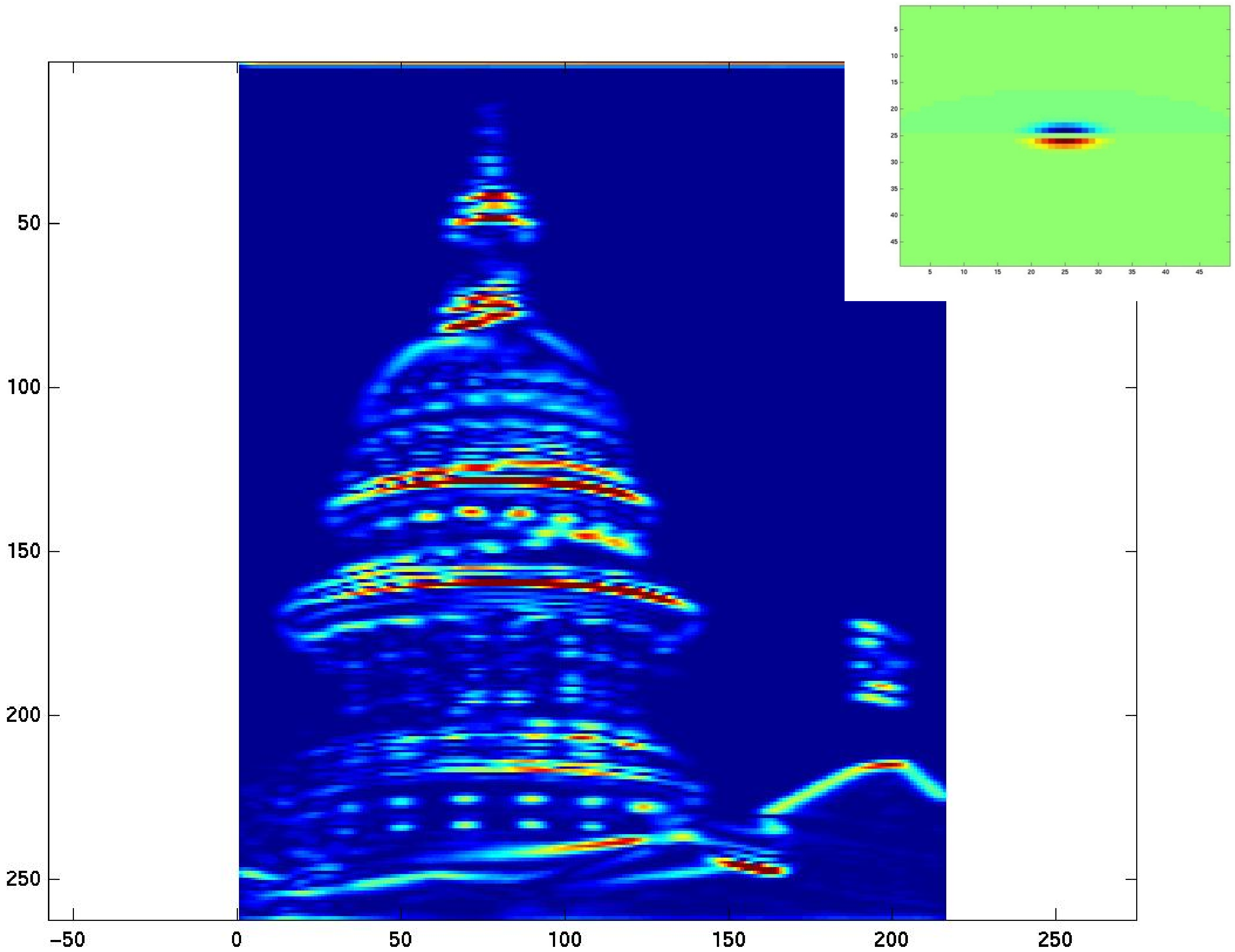
Extracting Texture

- Texture filter bank:

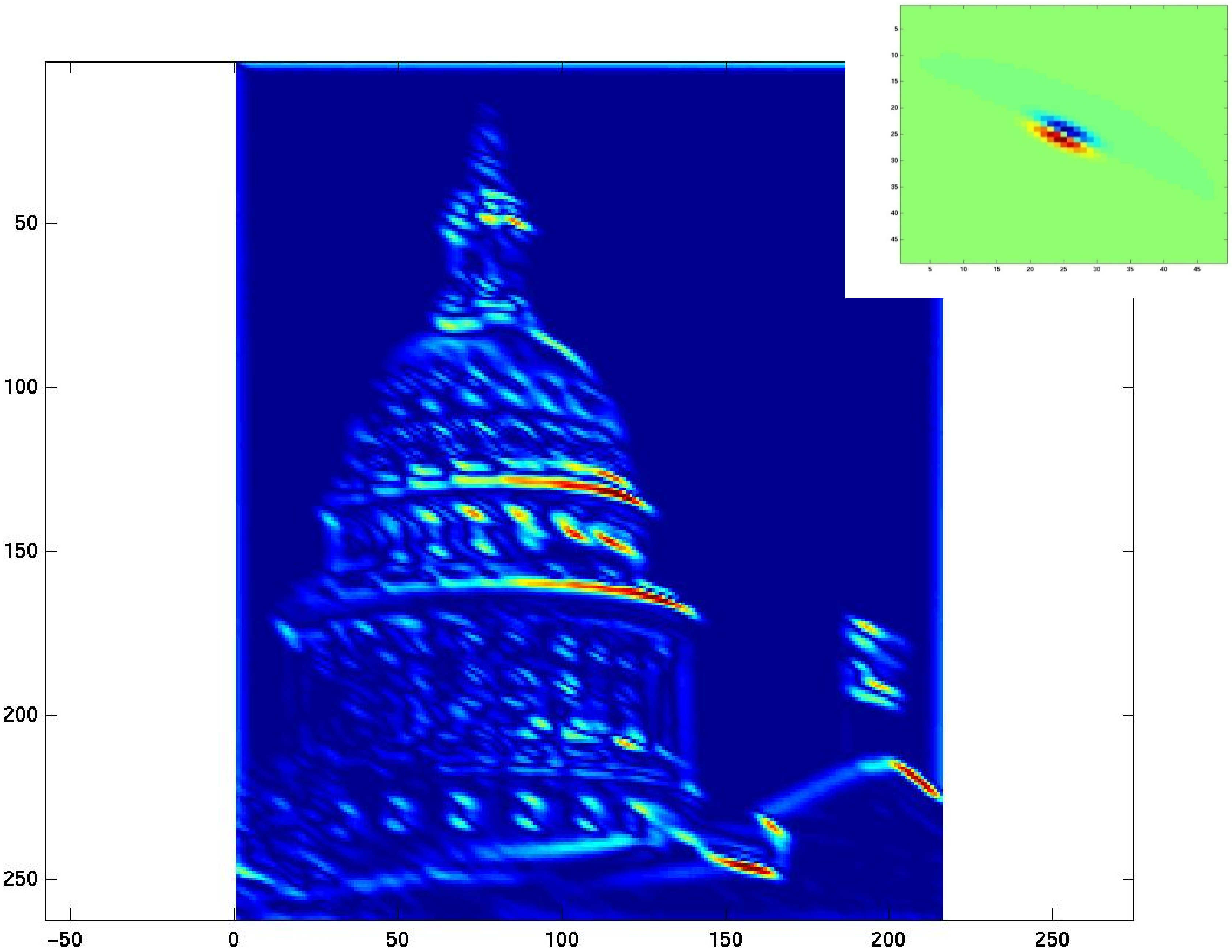


- Convolve image with each filter
  - 48 responses per pixel

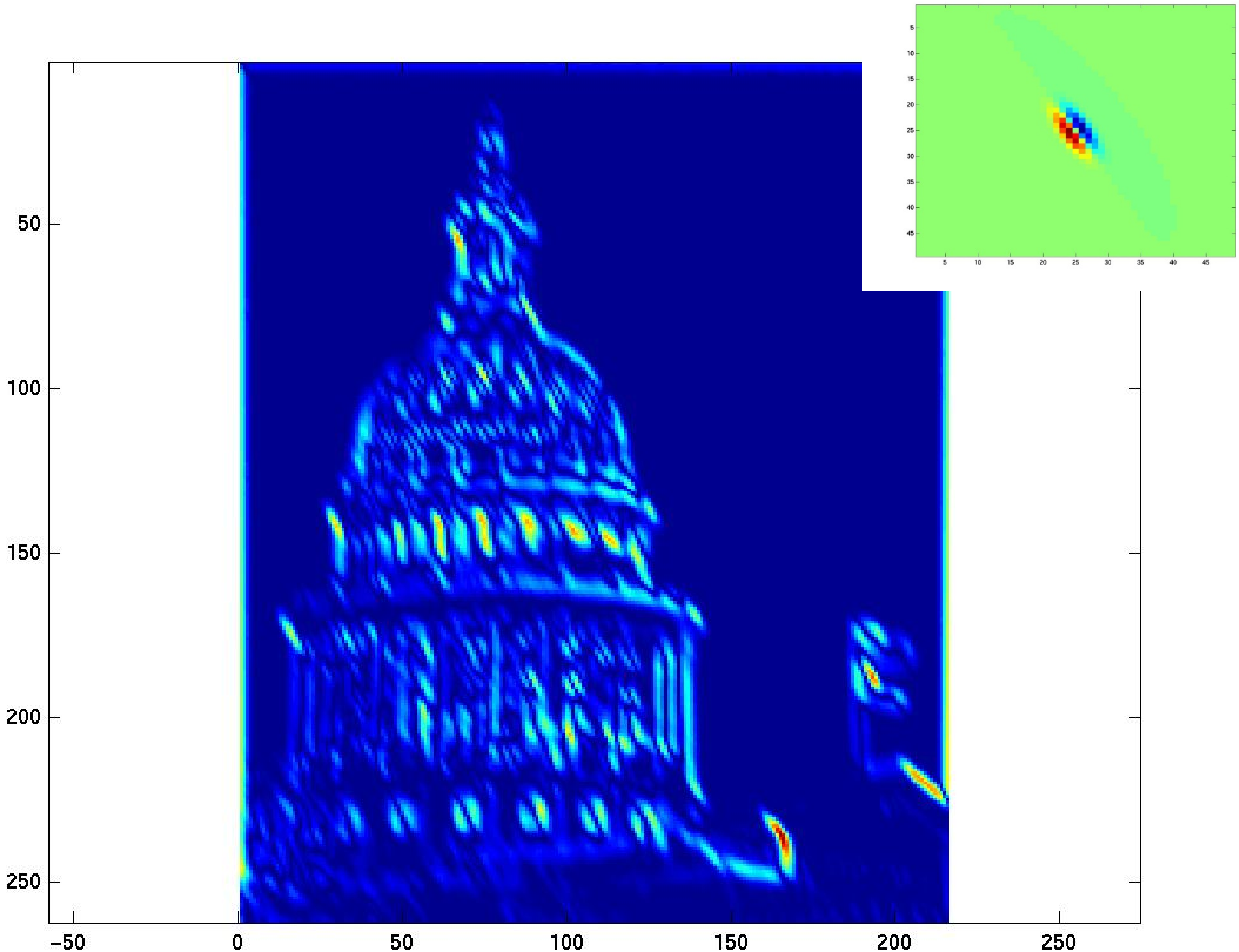
Kristen Grauman



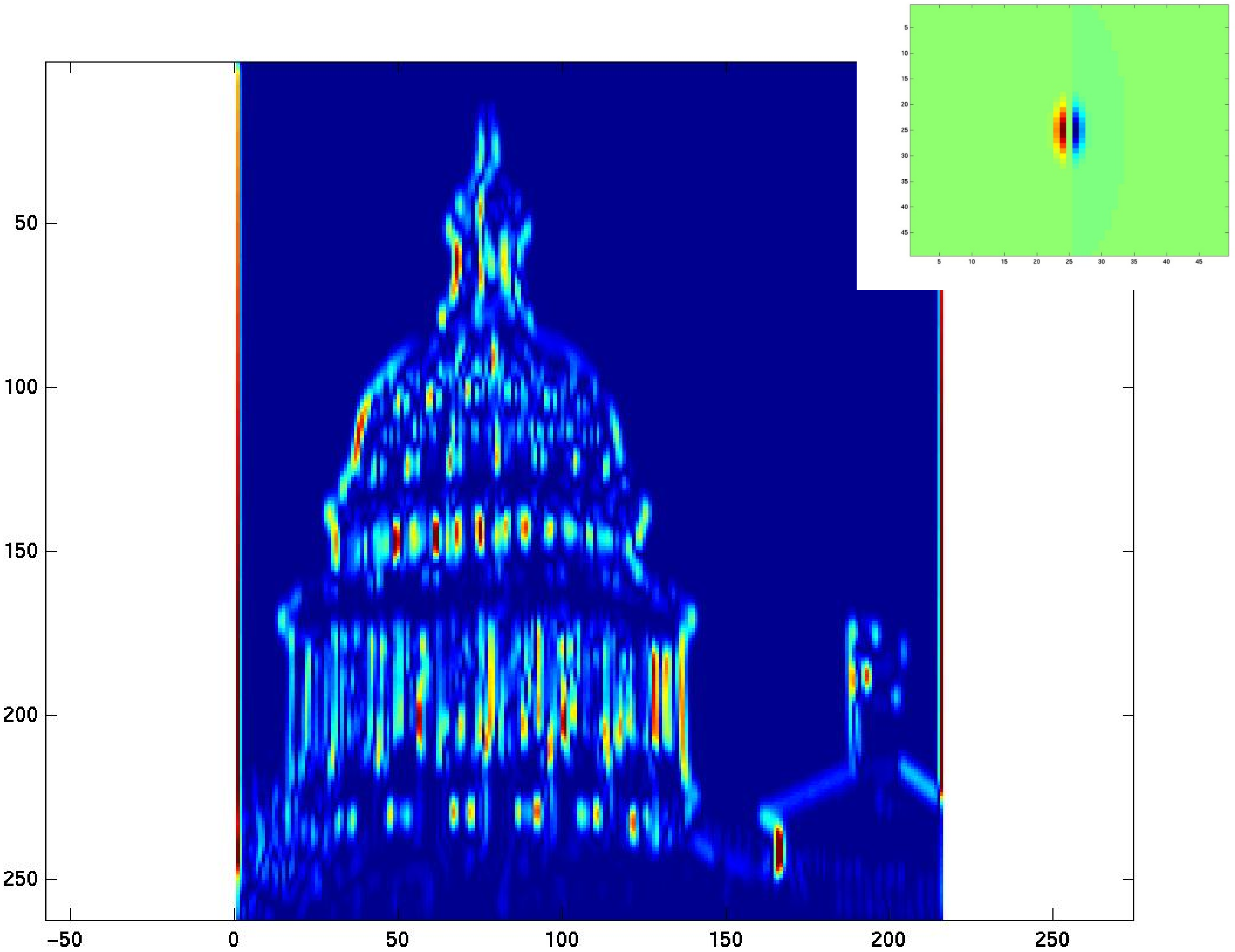
Kristen Grauman



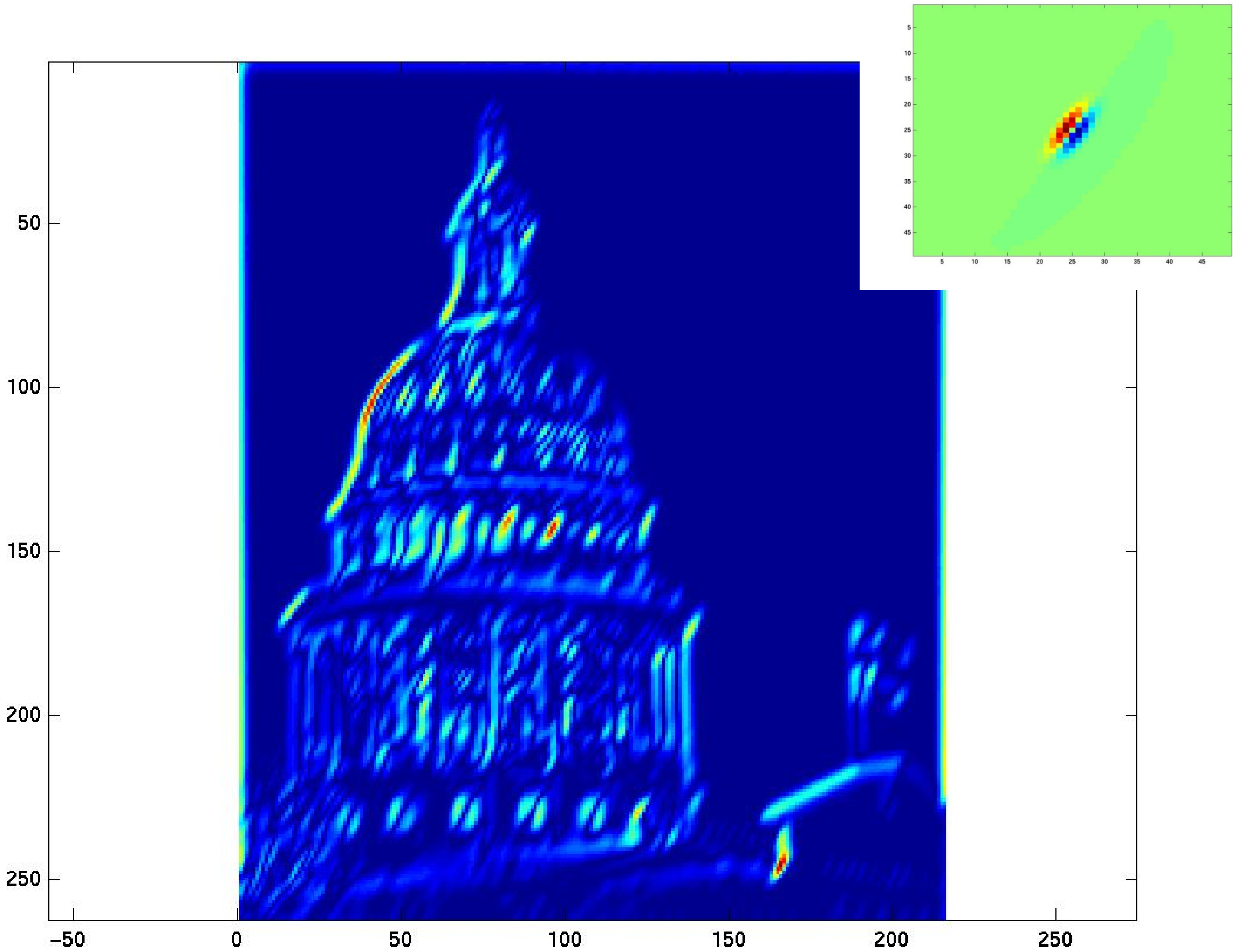
Kristen Grauman



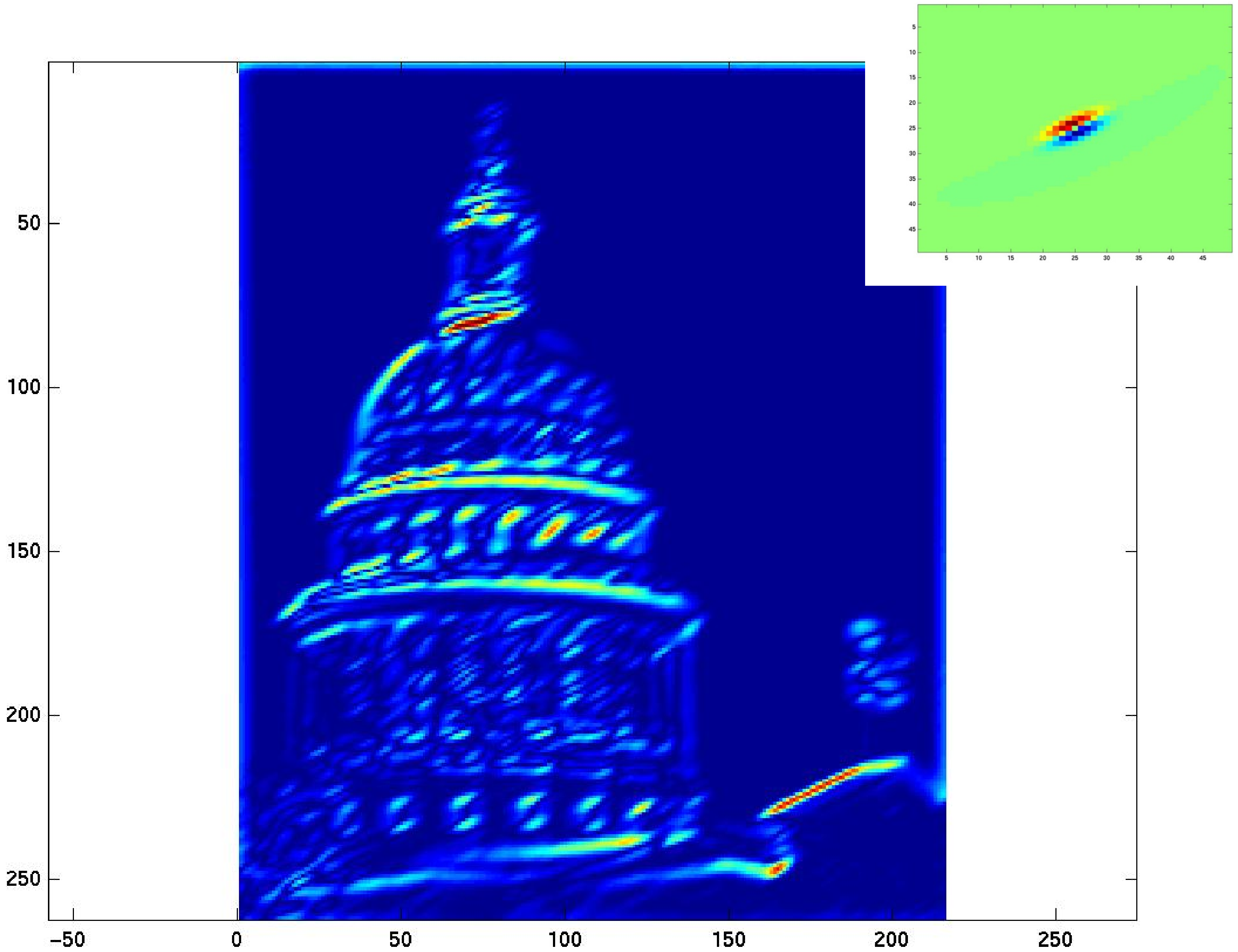
Kristen Grauman



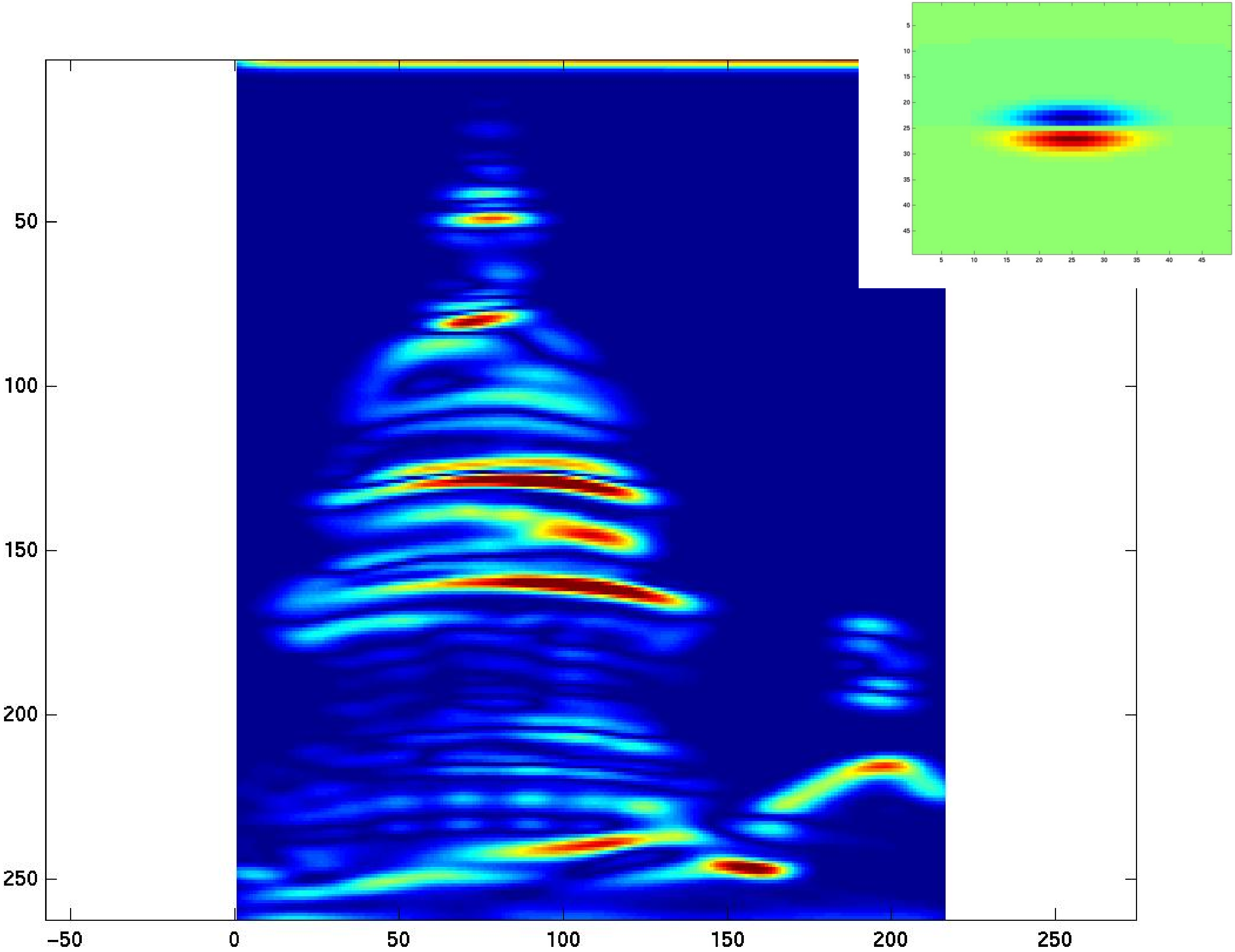
Kristen Grauman



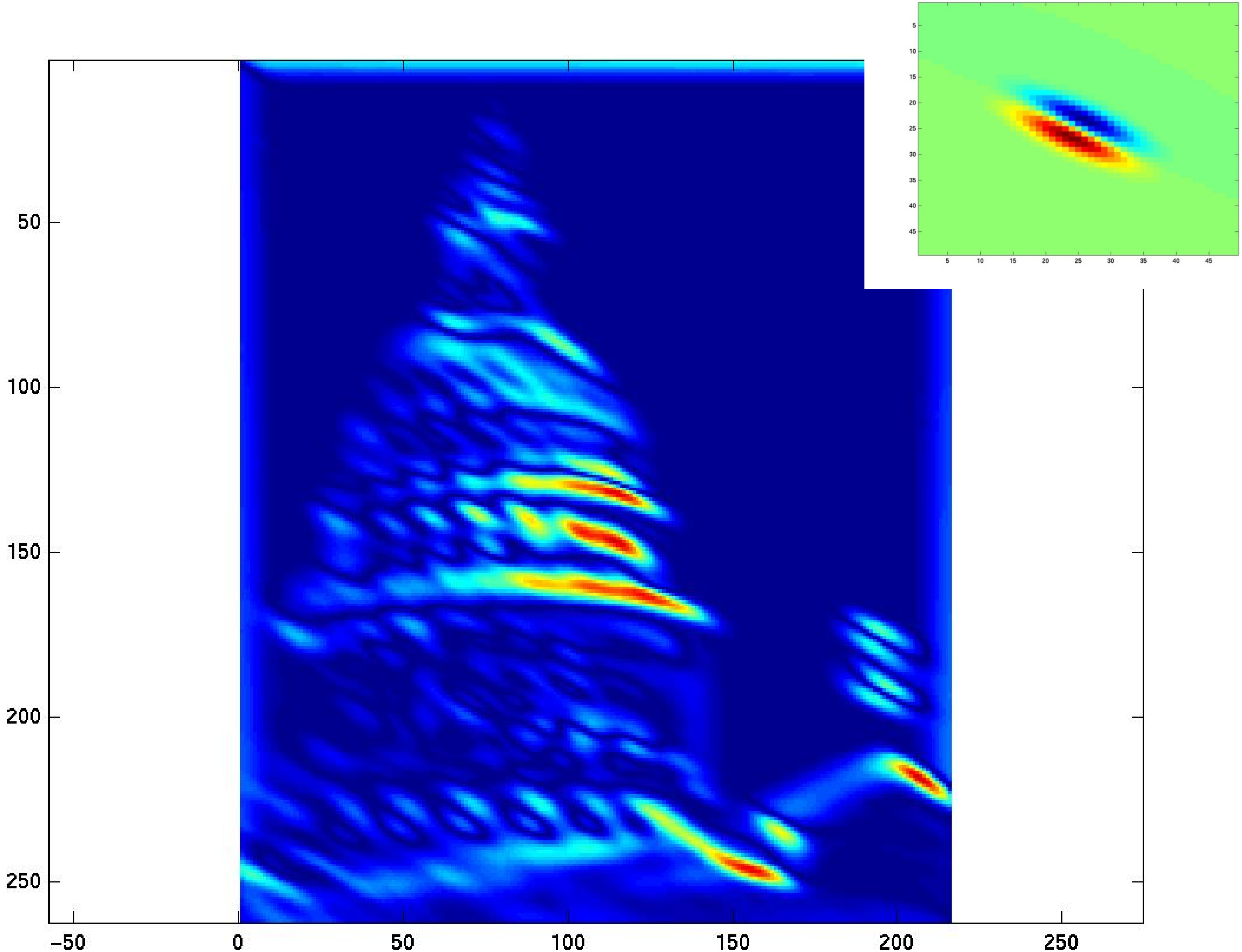
Kristen Grauman



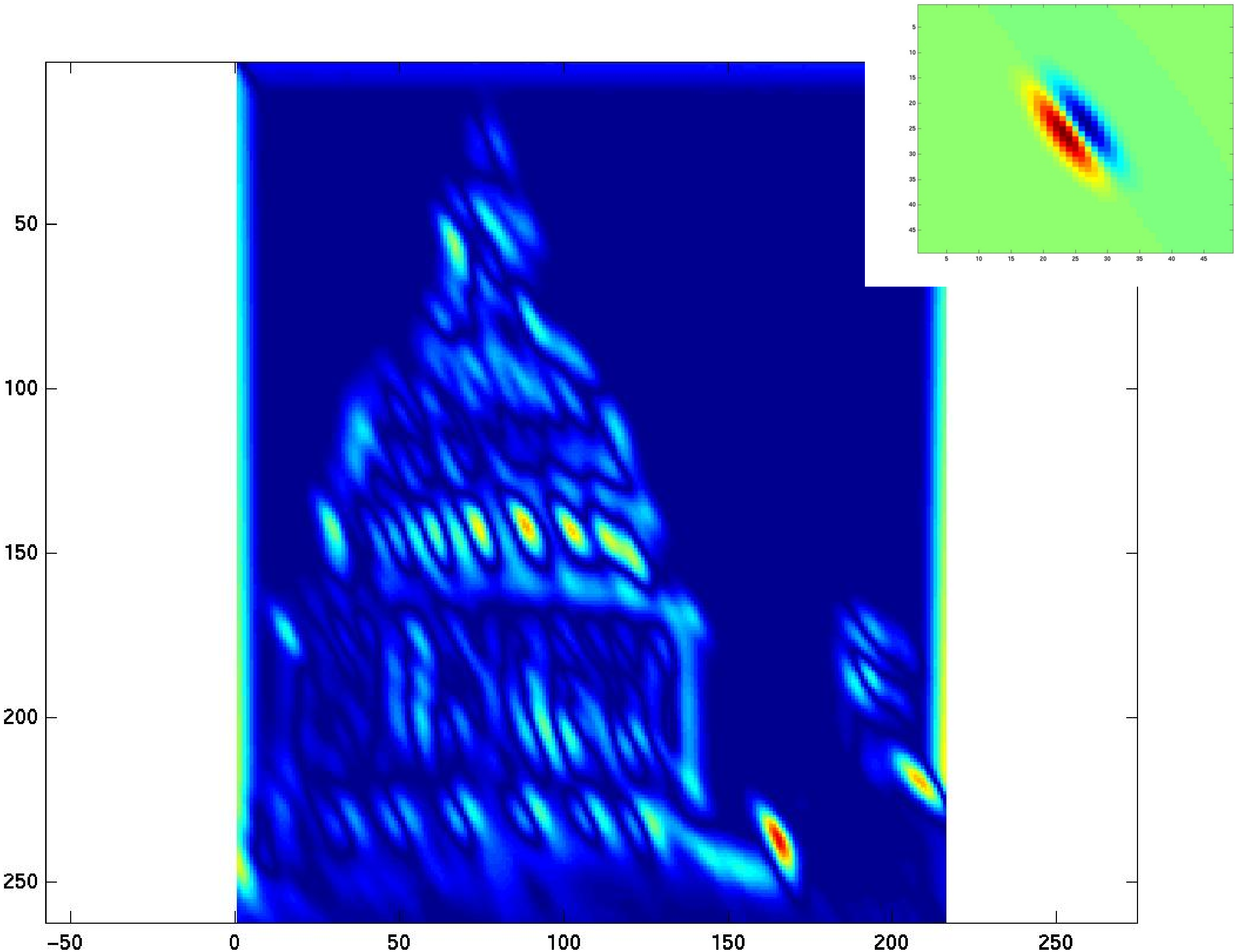
Kristen Grauman



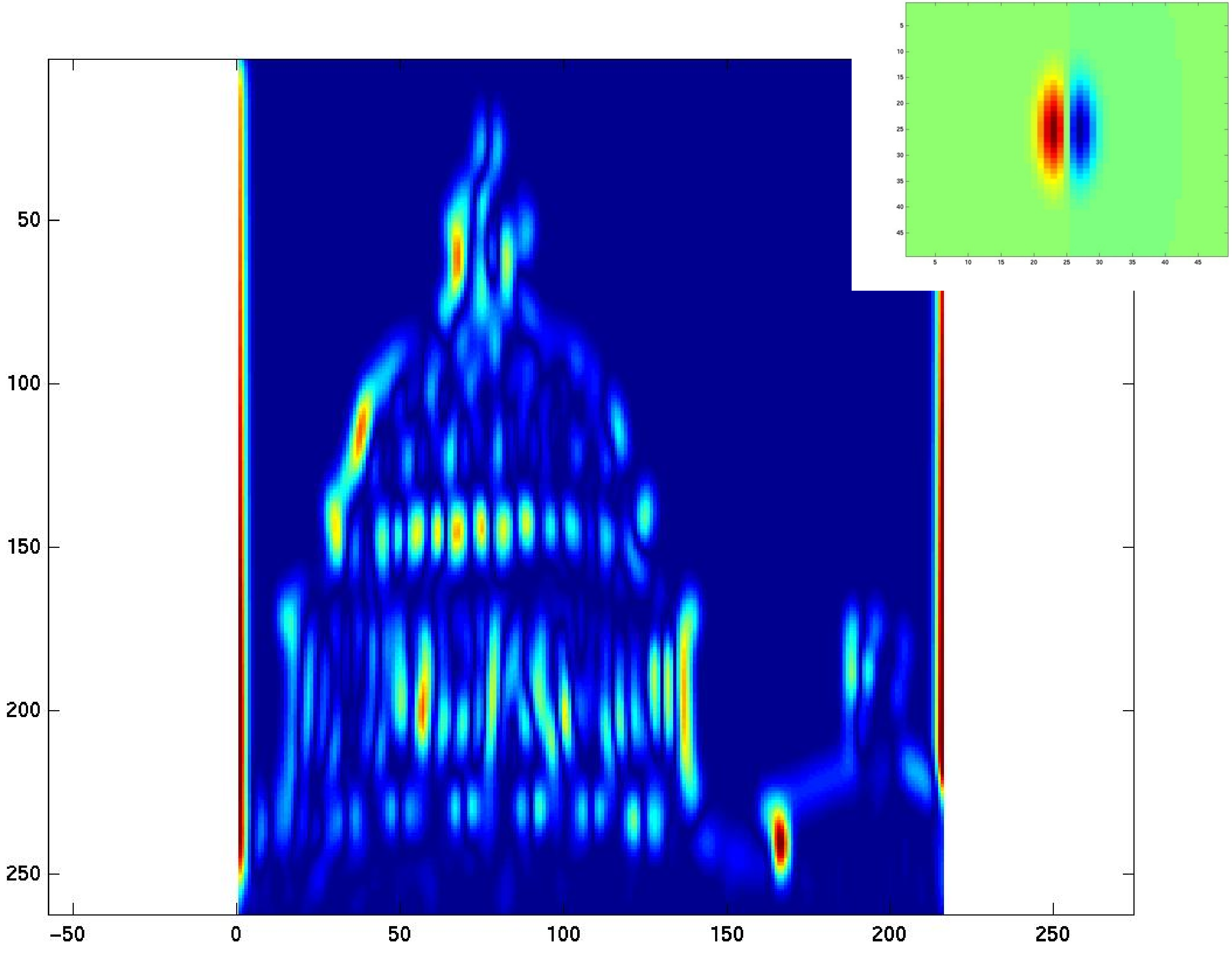
Kristen Grauman



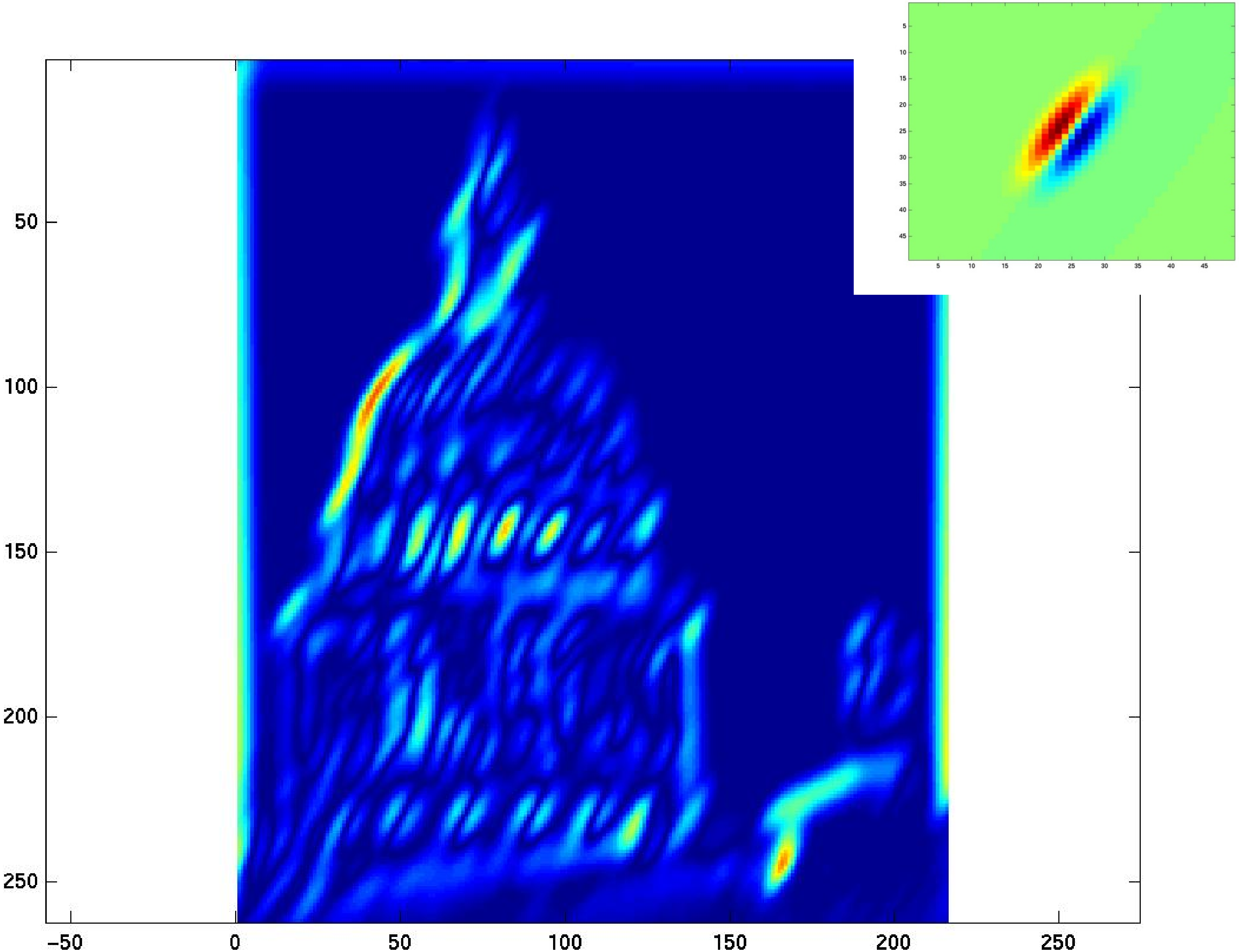
Kristen Grauman



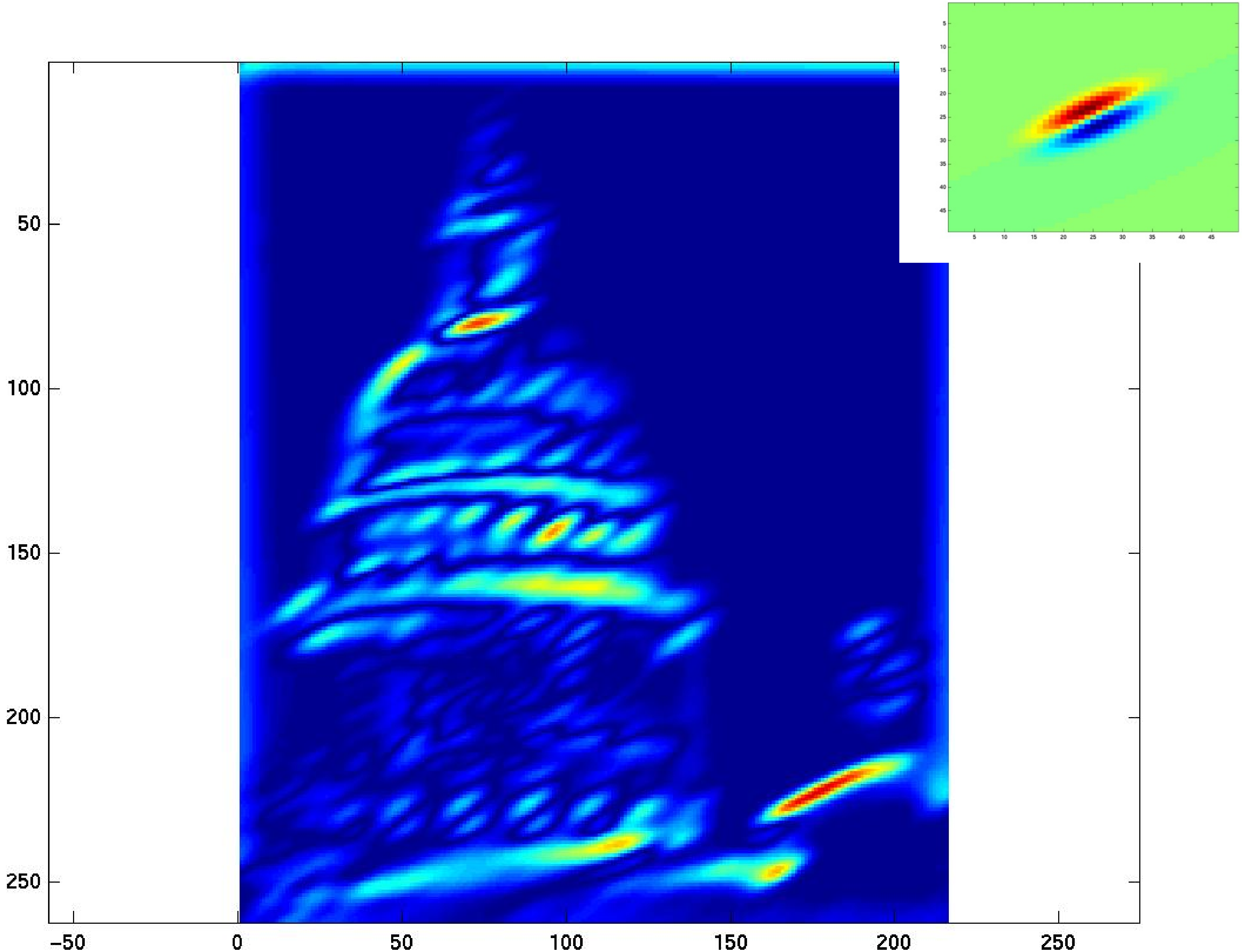
Kristen Grauman



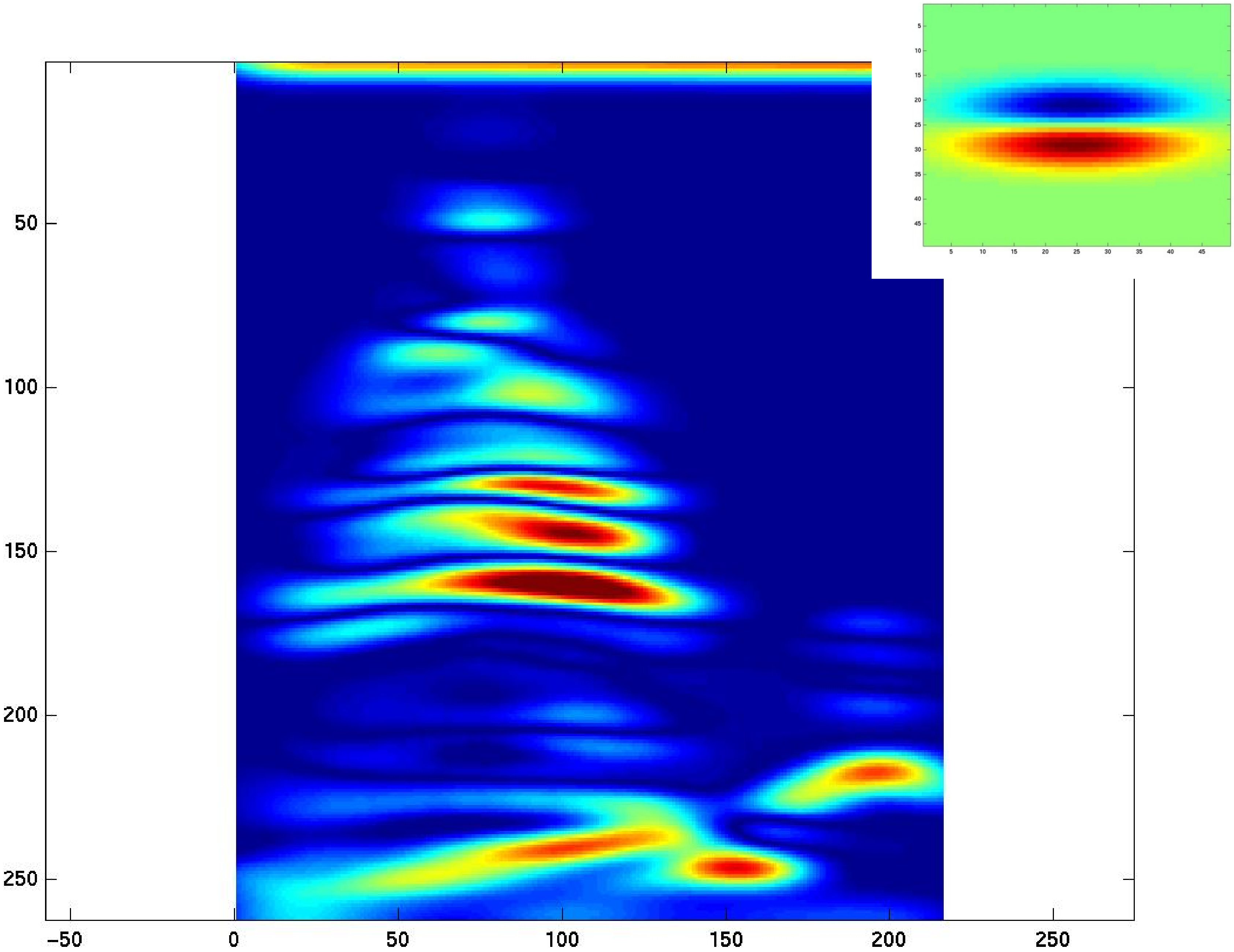
Kristen Grauman



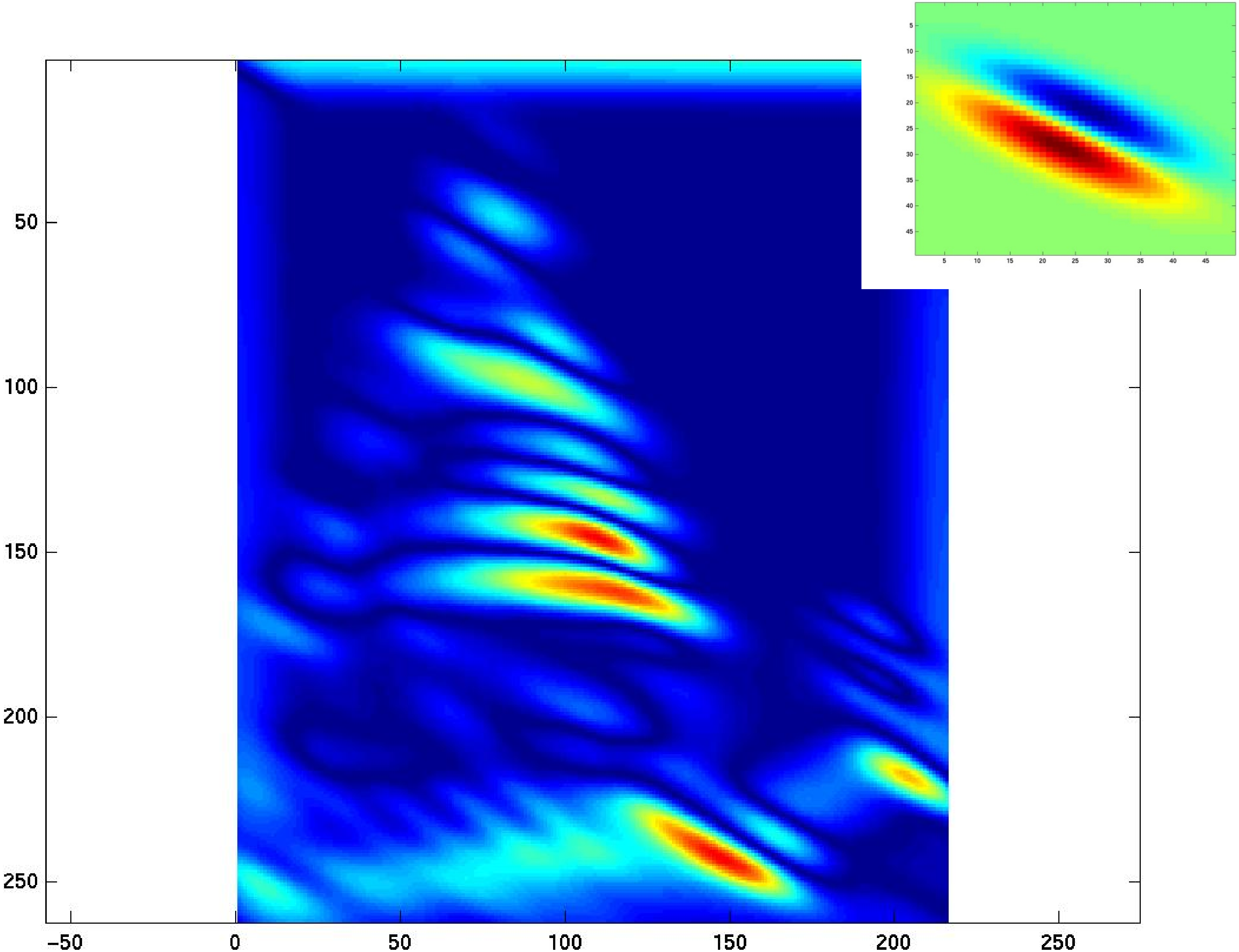
Kristen Grauman



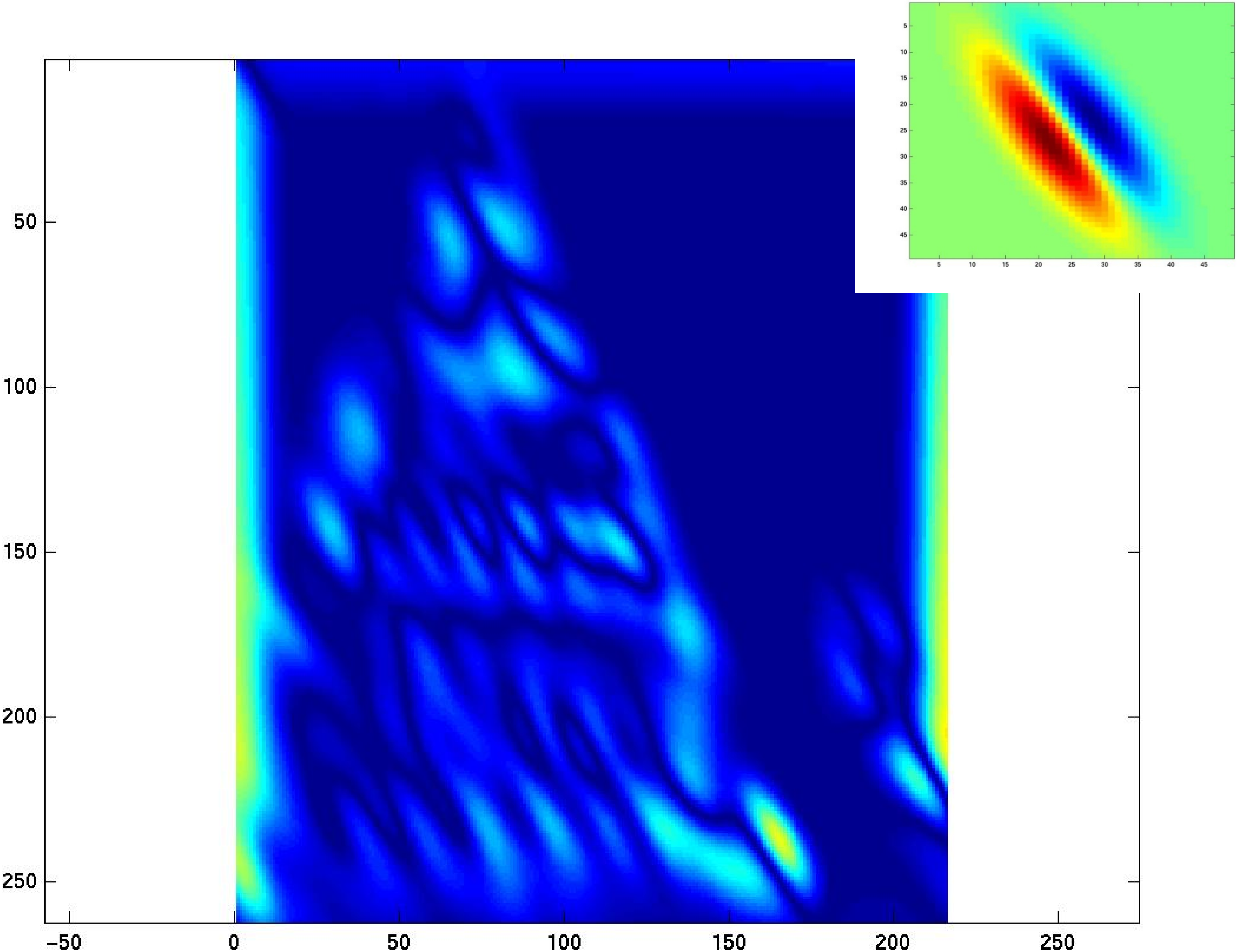
Kristen Grauman



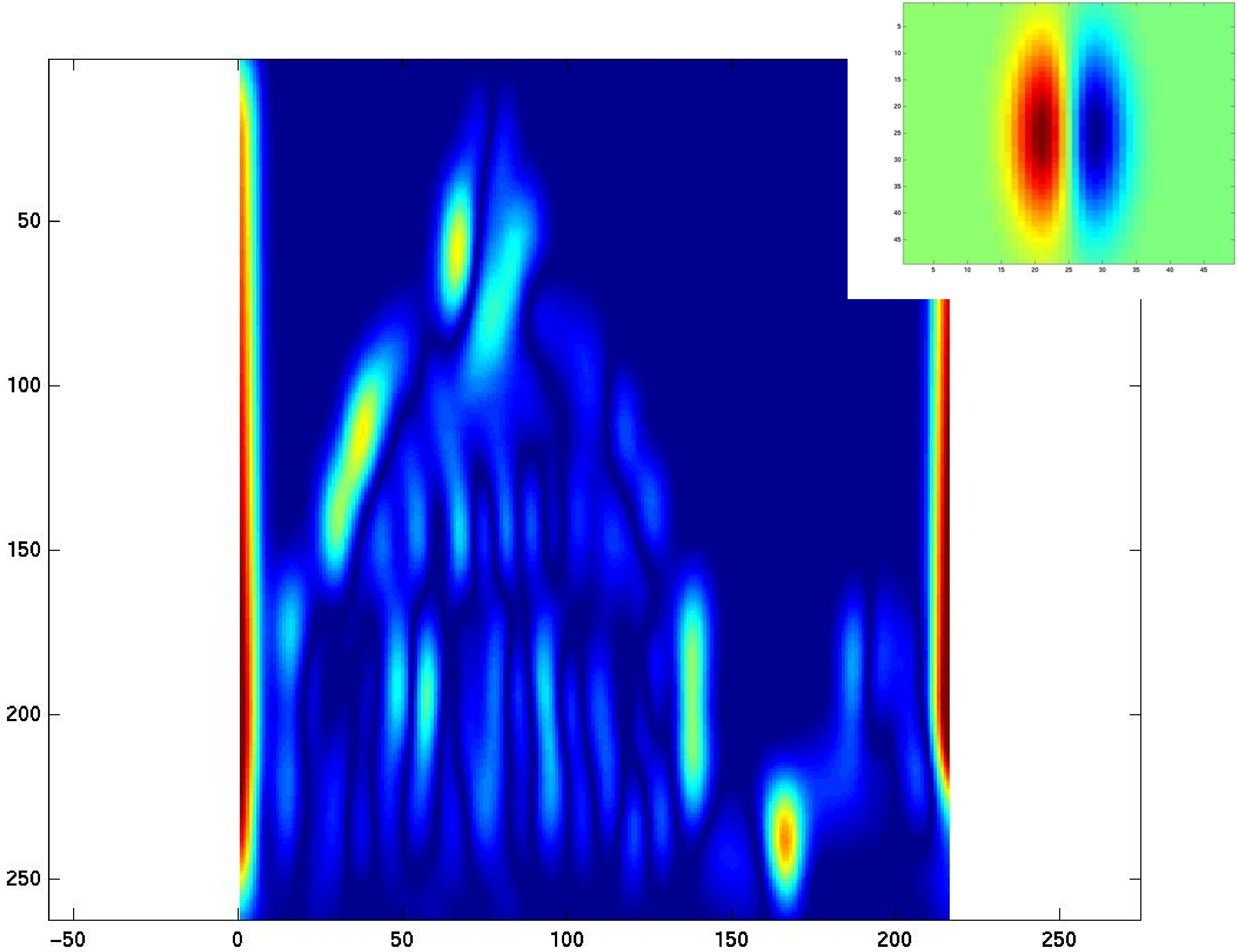
Kristen Grauman



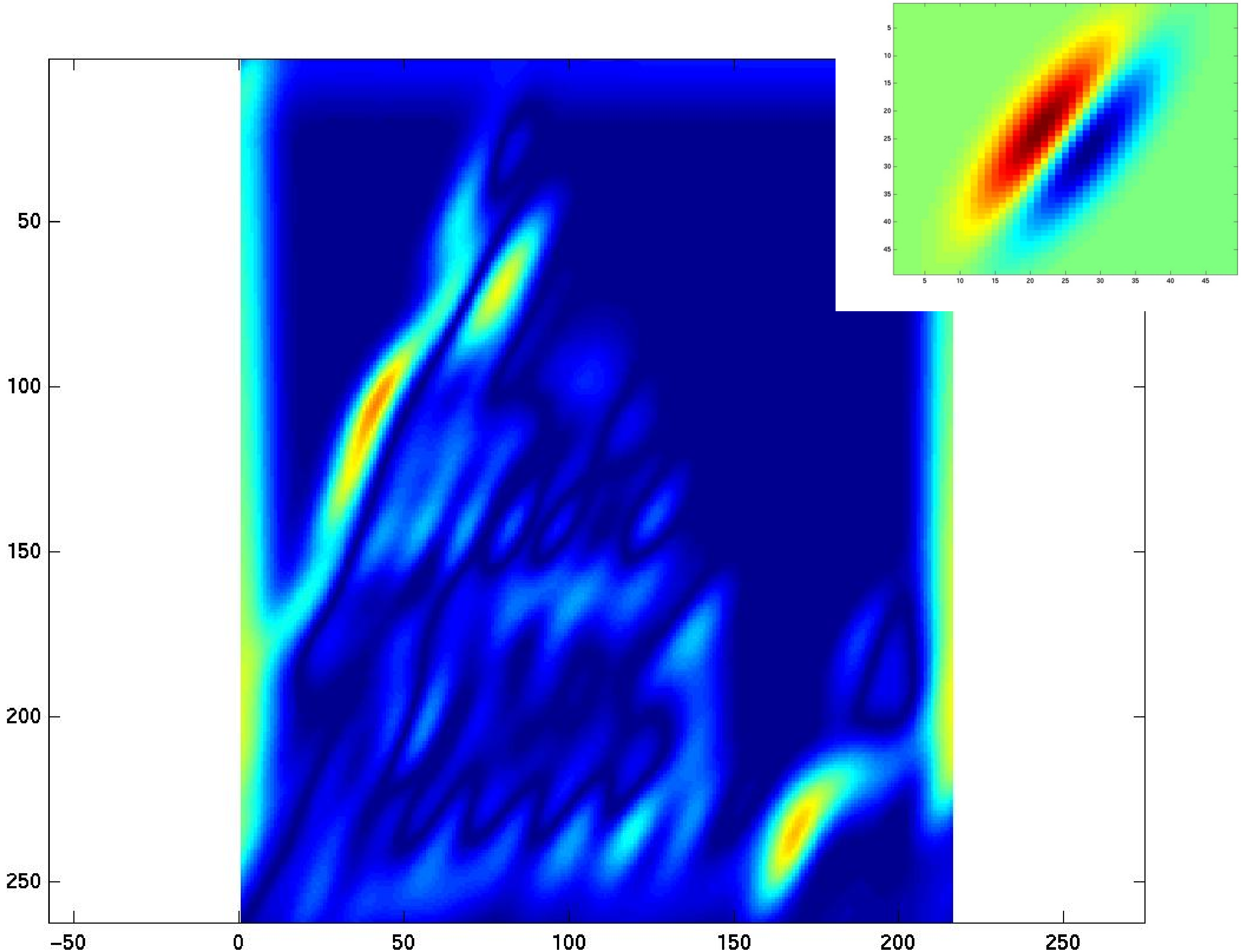
Kristen Grauman



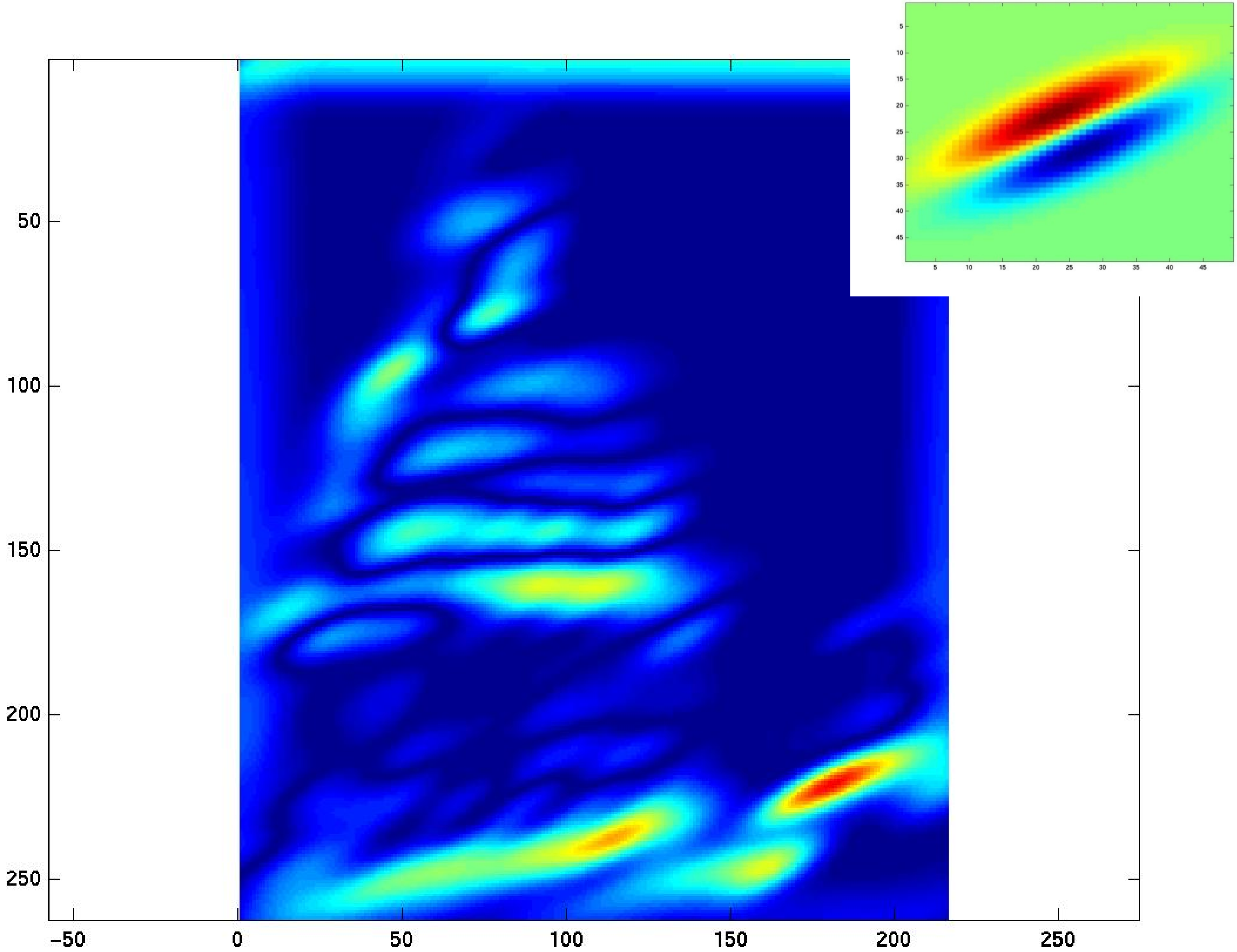
Kristen Grauman



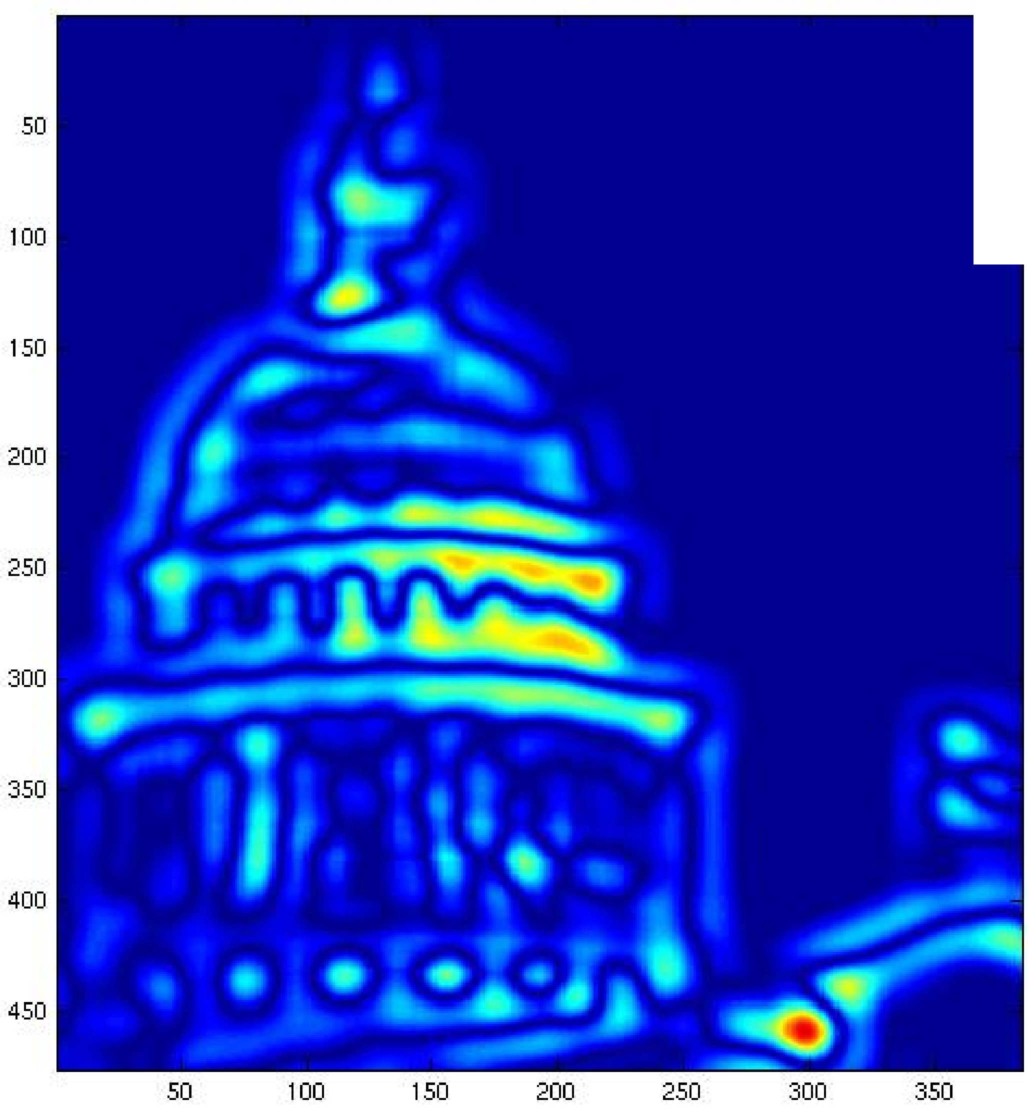
Kristen Grauman

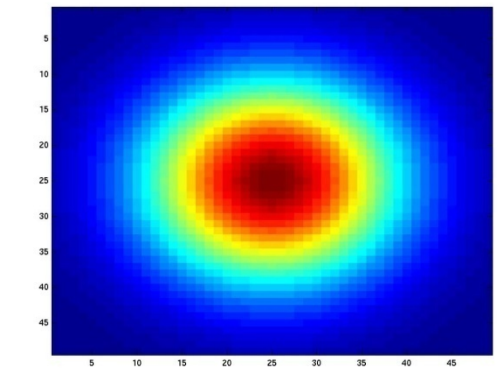


Kristen Grauman



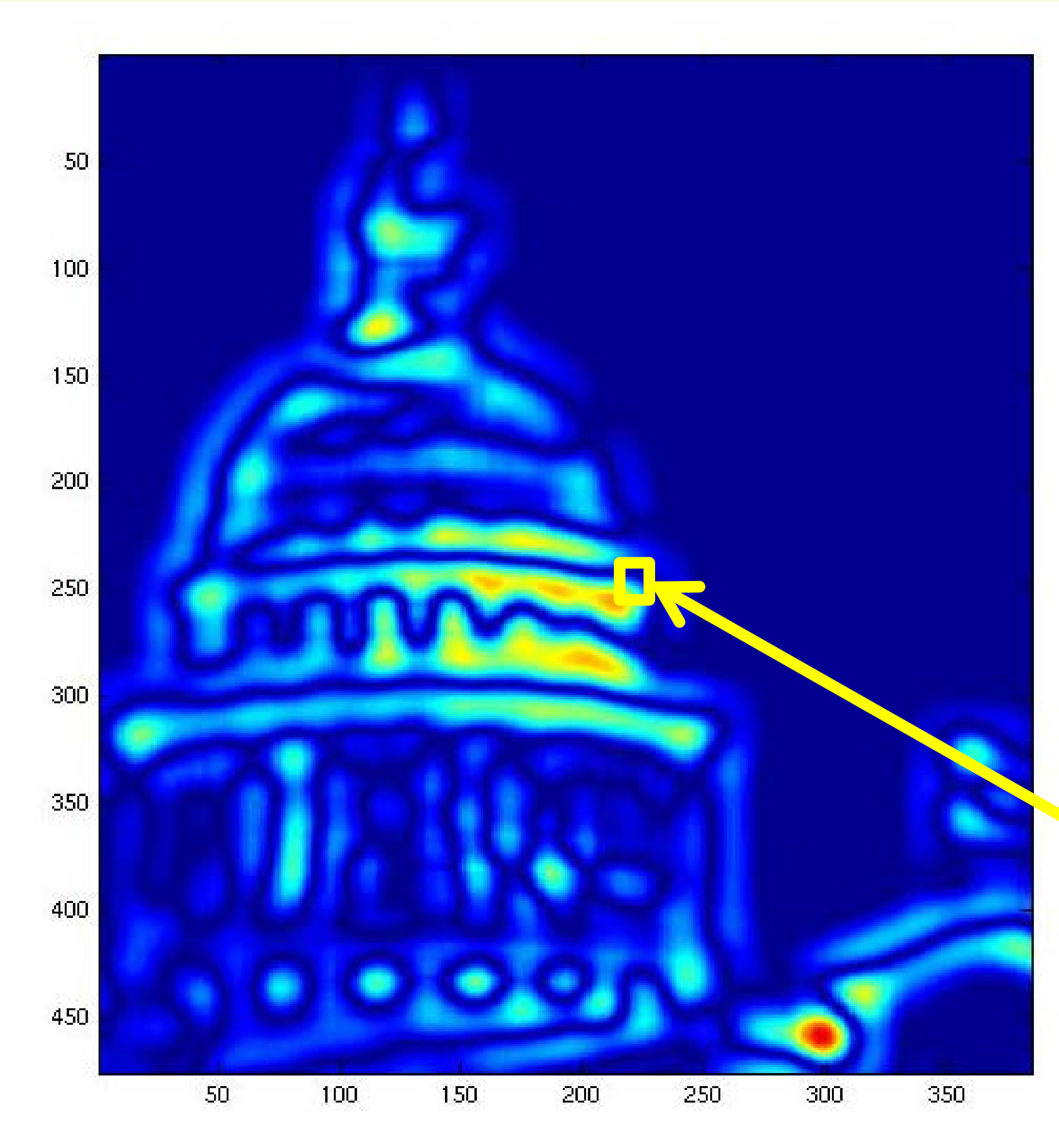
Kristen Grauman

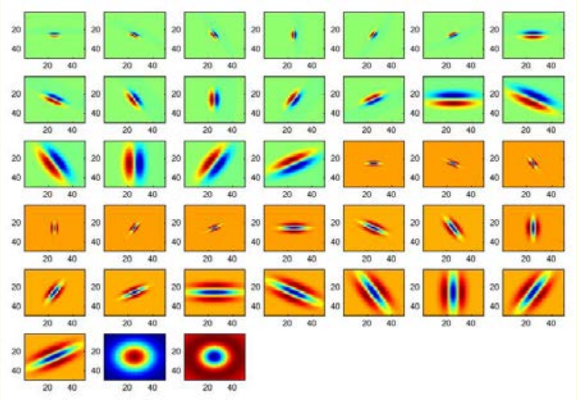




Kristen Grauman

Extracting Texture





Form a feature vector from the list of responses at each pixel

$[r_1, r_2, \ldots, r_{38}]$

Kristen Grauman

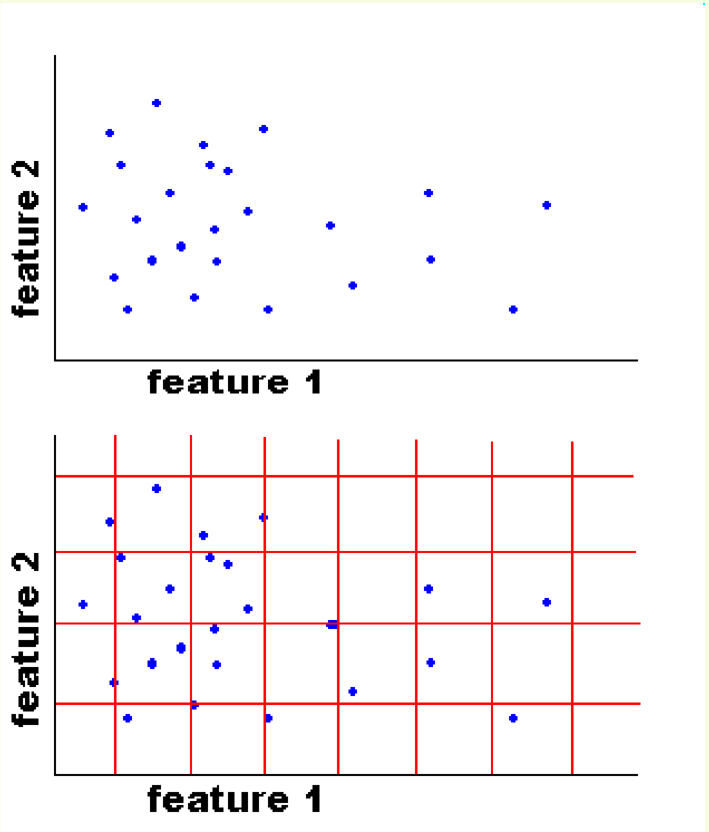
Multi-Dimensional Histograms

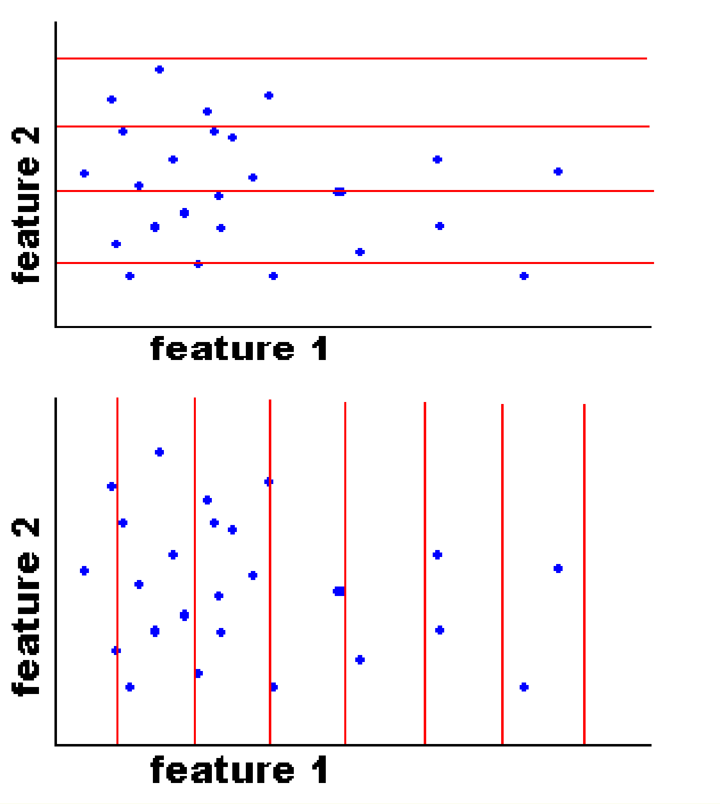
- Thus at each pixel we may extract many values
    - color, texture, optical flow, etc.
- How to build histogram?
- Have to quantize, too sparse without quantization

How to Quantize Multi-Dimensional Data?

1. Joint histogram
   - Need lots of data to avoid empty bins
   - Make bins coarse to simulate lots of data $\rightarrow$ loose resolution

2. Marginal histogram
   - more data per bin than joint histogram
   - works best for independent features
     - loose correlation information

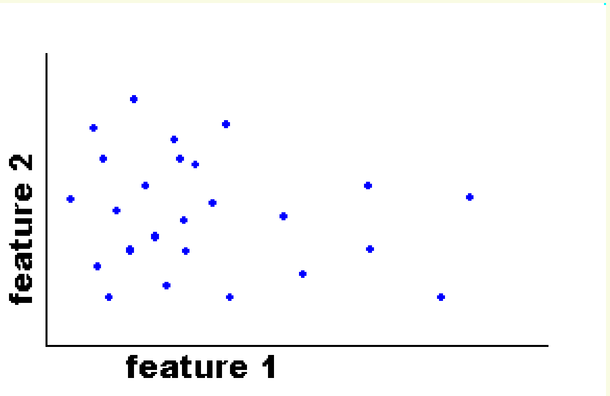


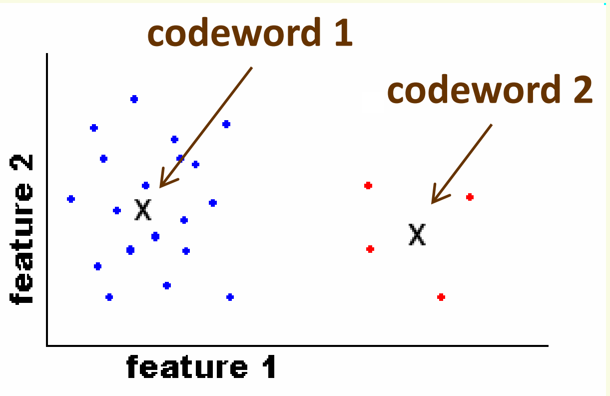


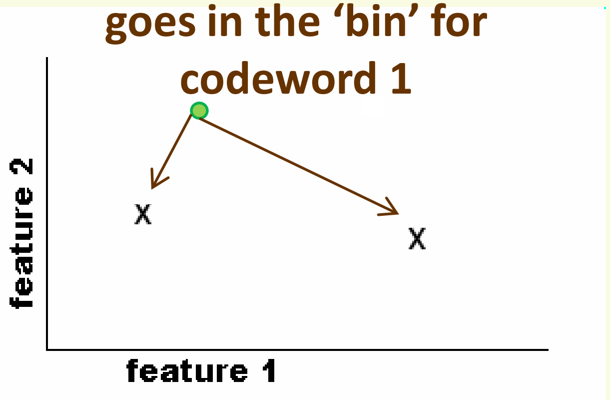
slide credit: Dave Kauchak

Histograms based on Irregular Partitioning

- Idea: use irregular partitioning (quantization)
    - often based on clustering (k-means is often used)





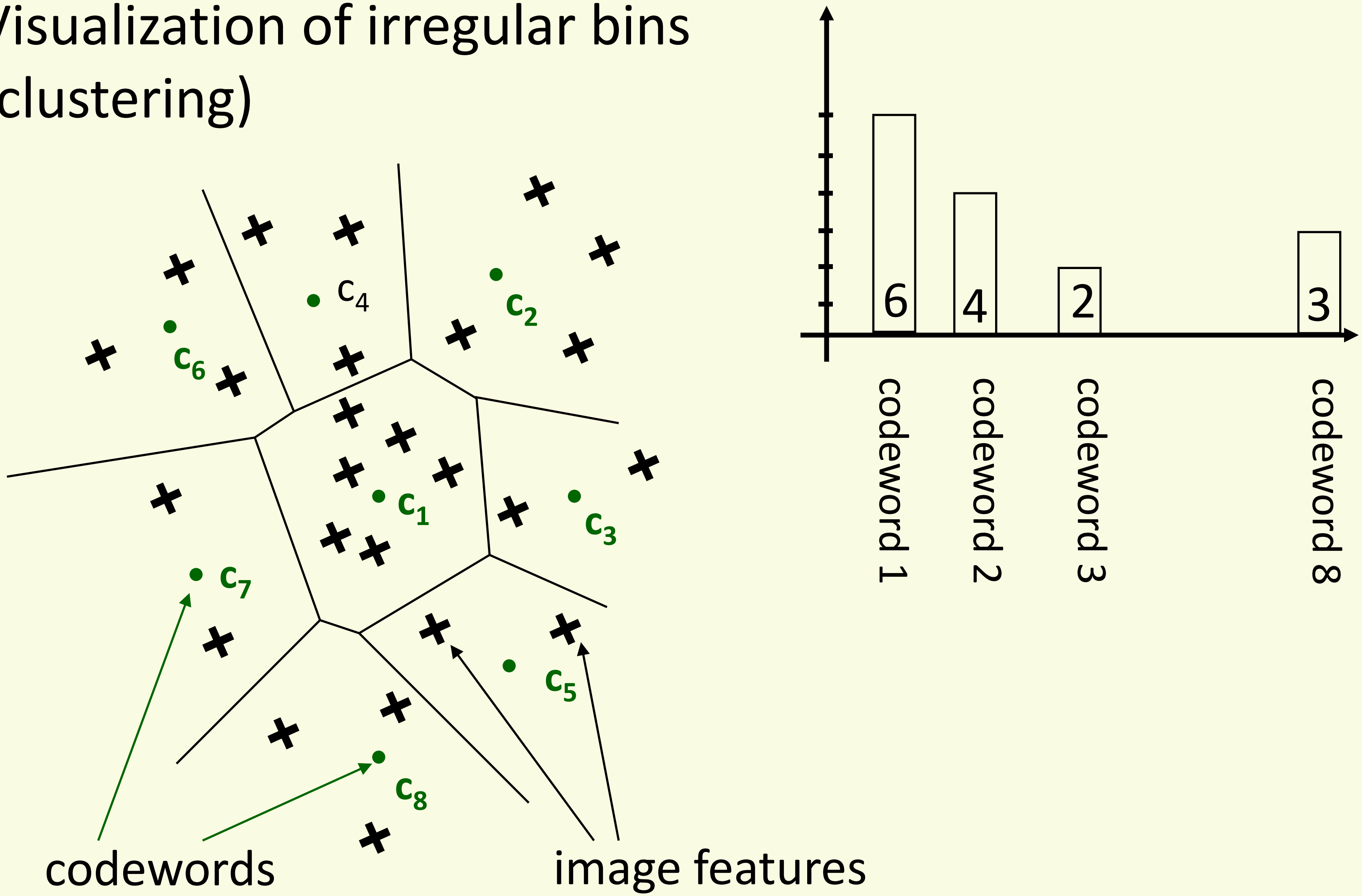


- After clustering, cluster centers (or codewords) stay fixed, these give us "bins" of irregular size
- A sample is identified with the closest codeword
- Build histogram over the codewords
    - that is count how many samples are closest to codeword_1, to codeword_2, etc

Slide Credit: Dave Kauchak

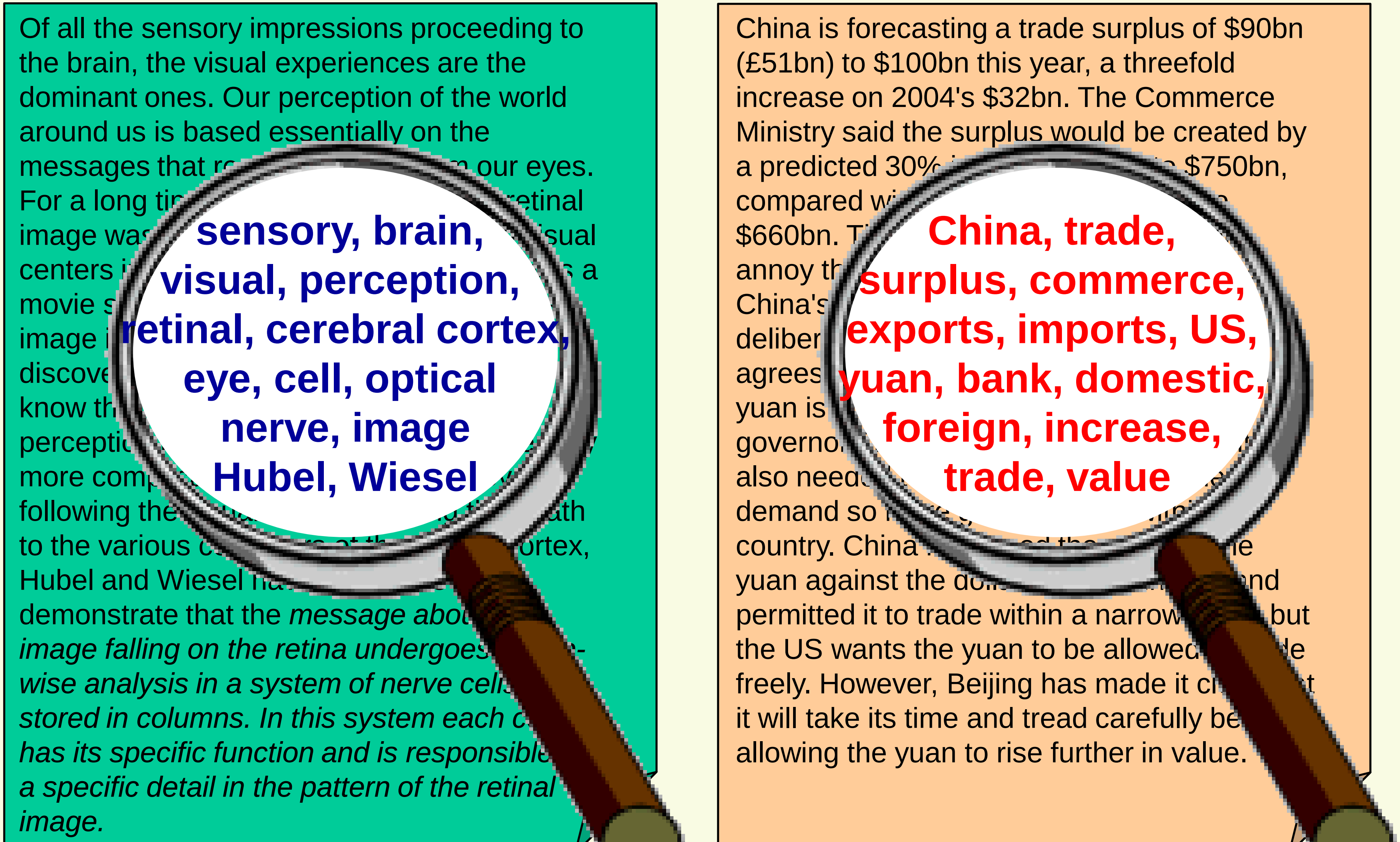
Voronoi Diagram visualization

- Visualization of irregular bins (clustering)



Analogy to documents: Bag of Words

- Inspiration comes from text classification



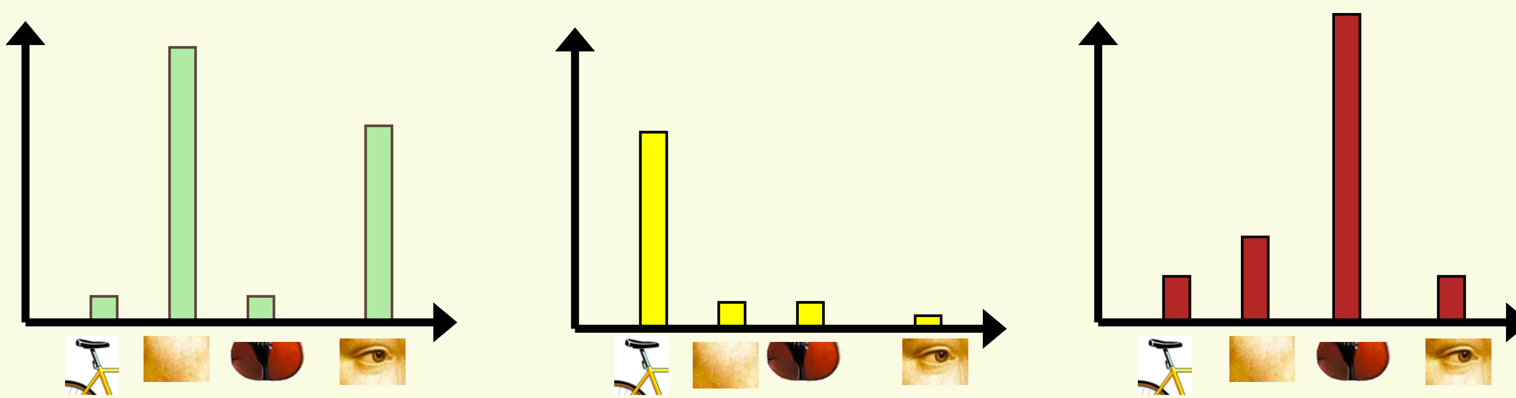
ICCV 2005 short course, L. Fei-Fei

Bag of visual words

- Training images

- visual words or codewords

- Bow histogram

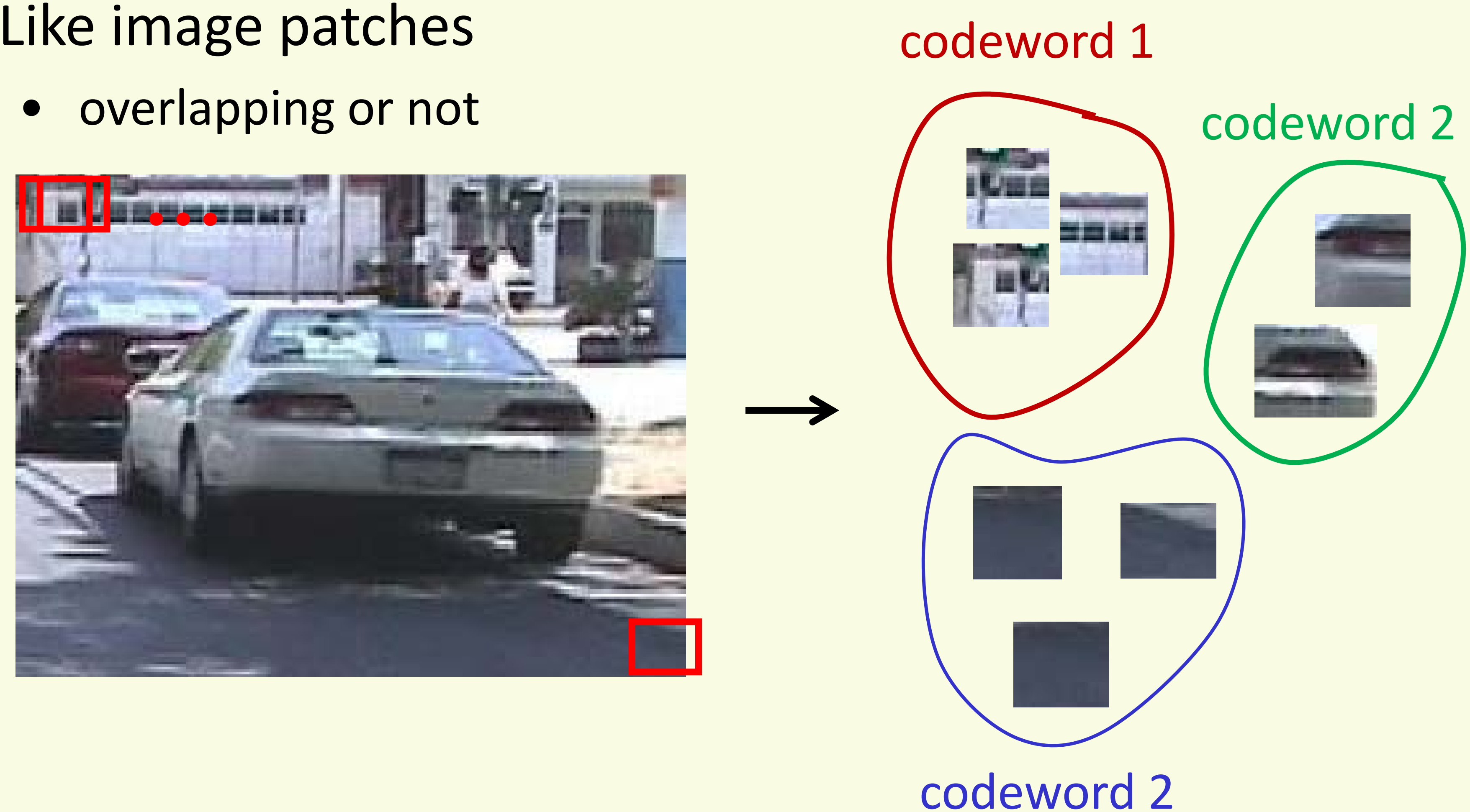


codewords

Slide by Derek Hoiem

Clustered Patches

- So far we clustered feature responses at each pixel
- Can cluster other things
- Like image patches
  - overlapping or not

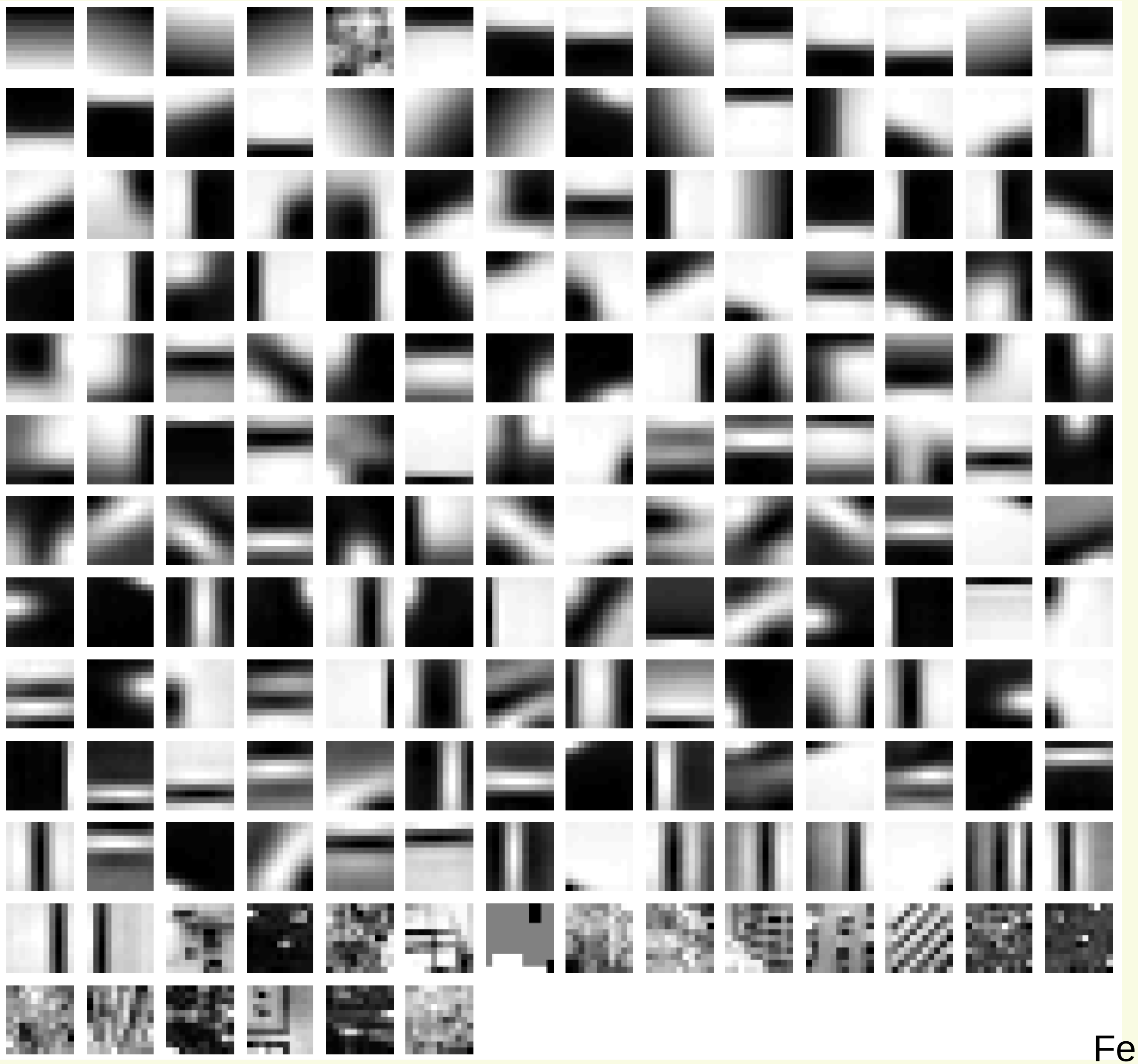


Codewords

- We find codewords on training data, not just one image

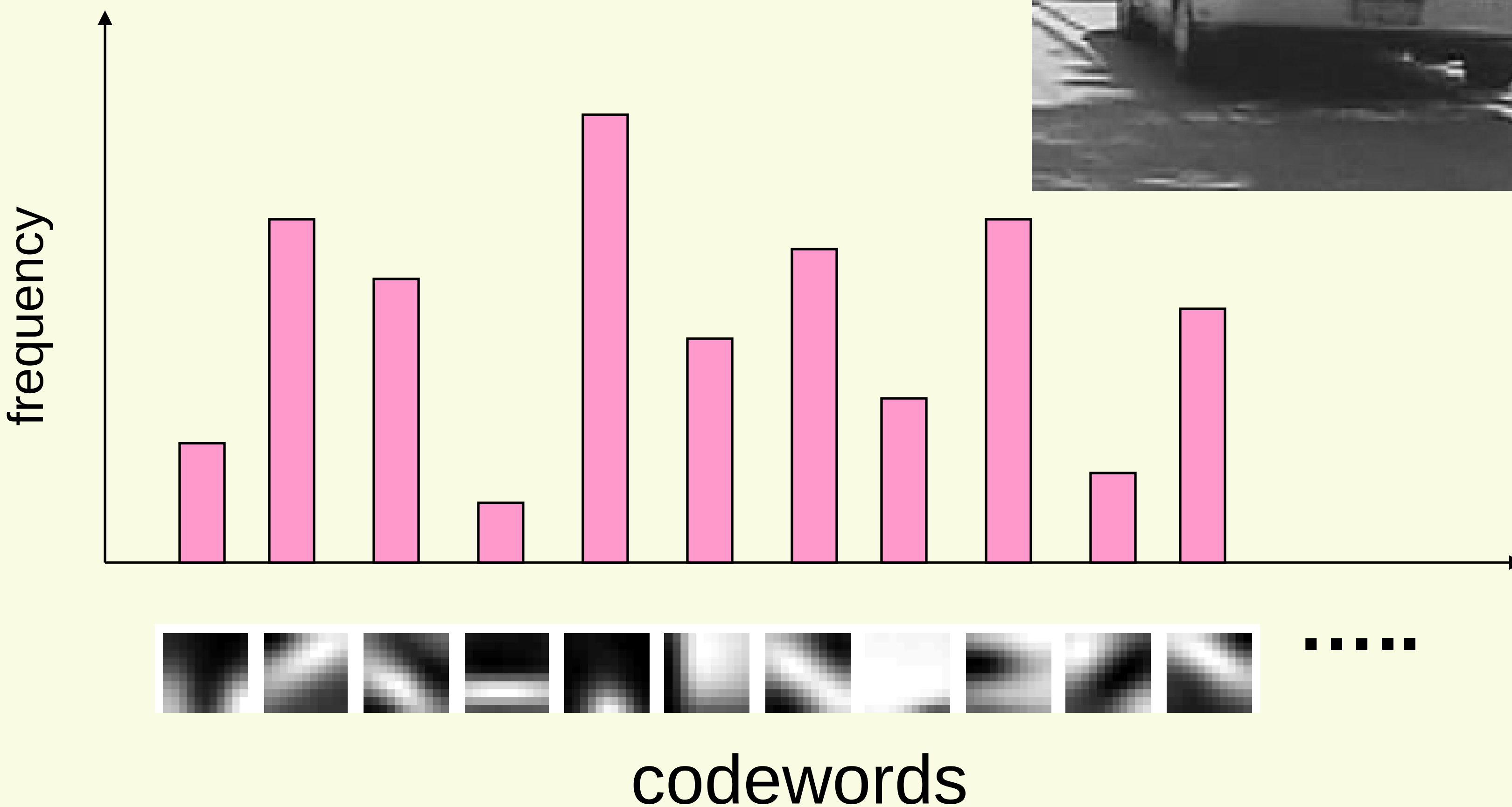
- But not on test data

Clustered Image Patches

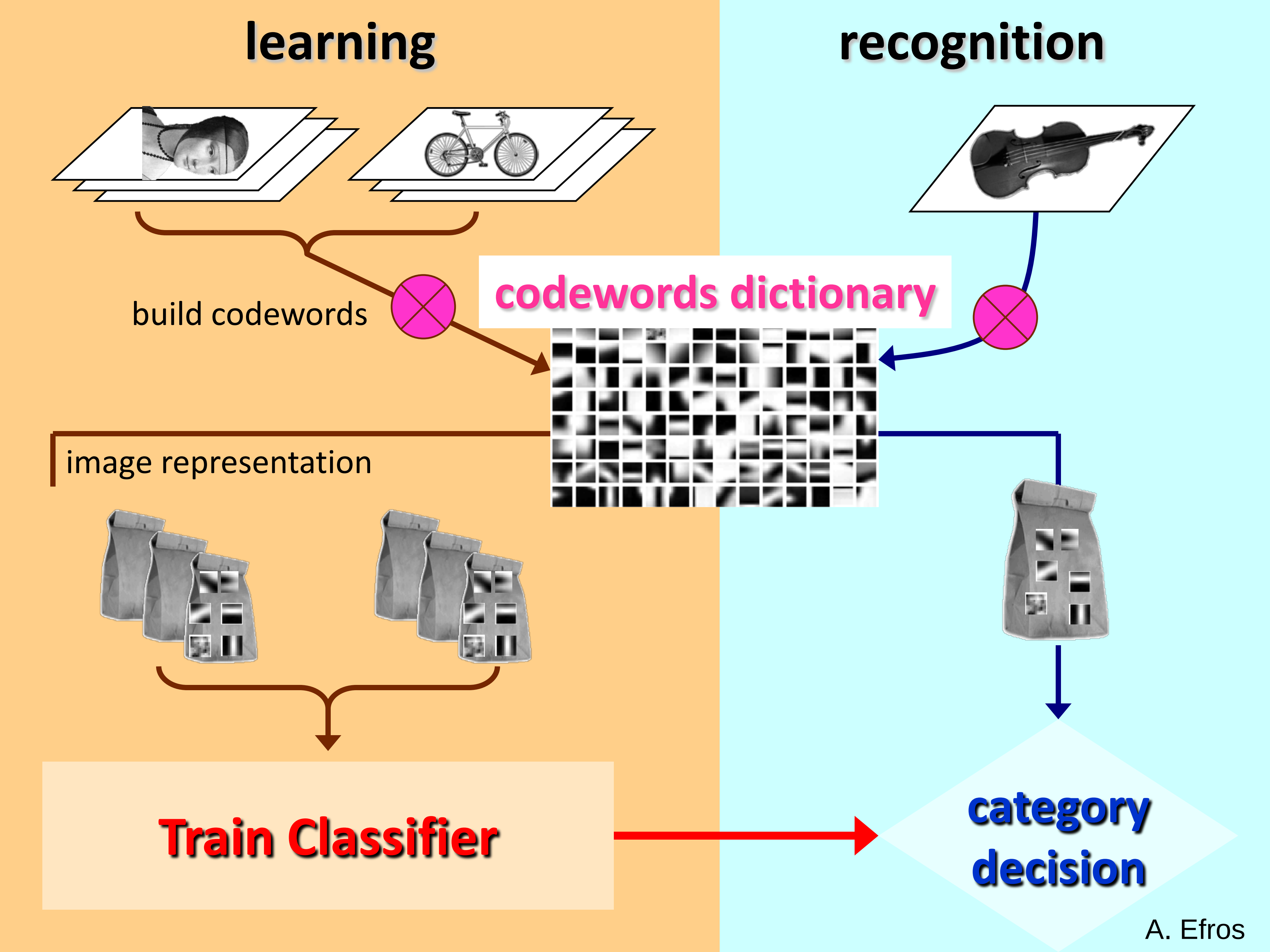


Fei-Fei et al. 2005

Image Representation

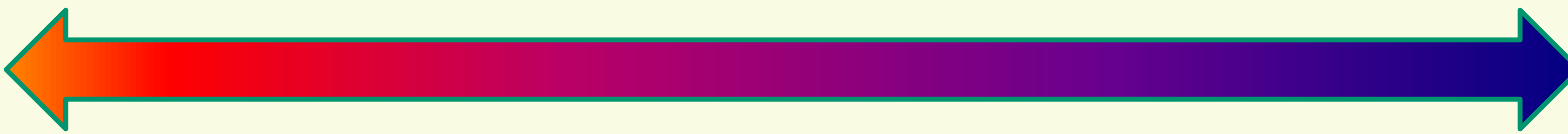


A. Efros


learning
recognition
build codewords
codewords dictionary
image representation
Train Classifier
category
decision
A. Efros

Histograms: Implementation issues

- Quantization
  - Grids: fast but applicable only with few dimensions
  - Clustering: slower but can quantize data in higher dimensions
- How many bins (clusters)?

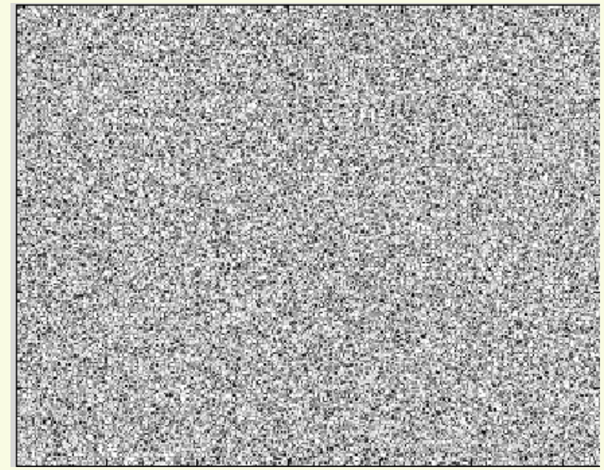


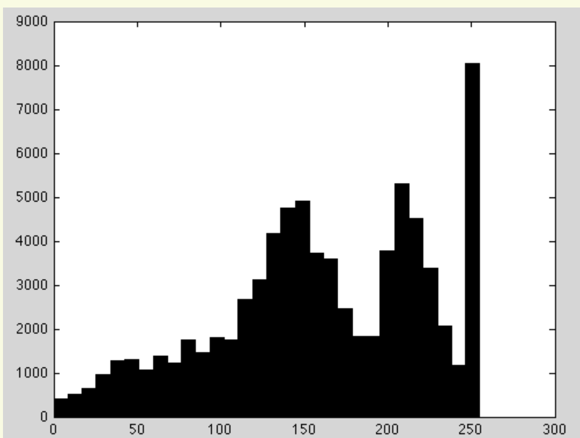
Few Bins
Need less data
Coarser representation
If too coarse, distinction is lost

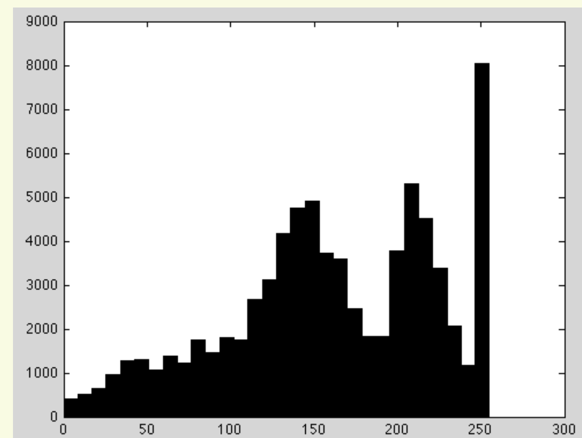
Many Bins
Need more data
Finer representation
If too fine, more distinction than necessary

Slide Credit: Derek Hoiem

Problem with Global Histogram



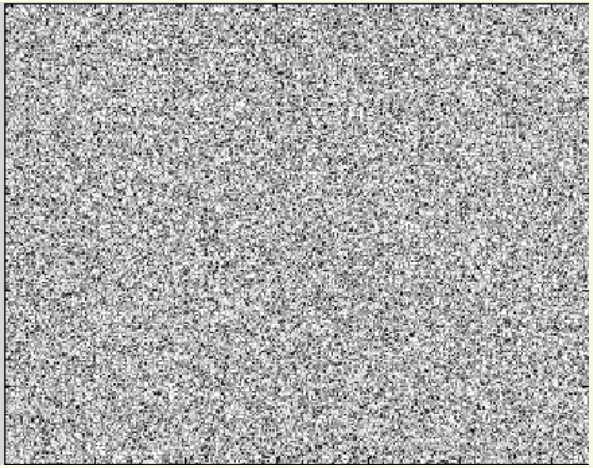




- Identical feature vectors!

Slide by Erik Learned-Miller

Problem with Global Histogram



Have equal histograms!

Slide by Erik Learned-Miller

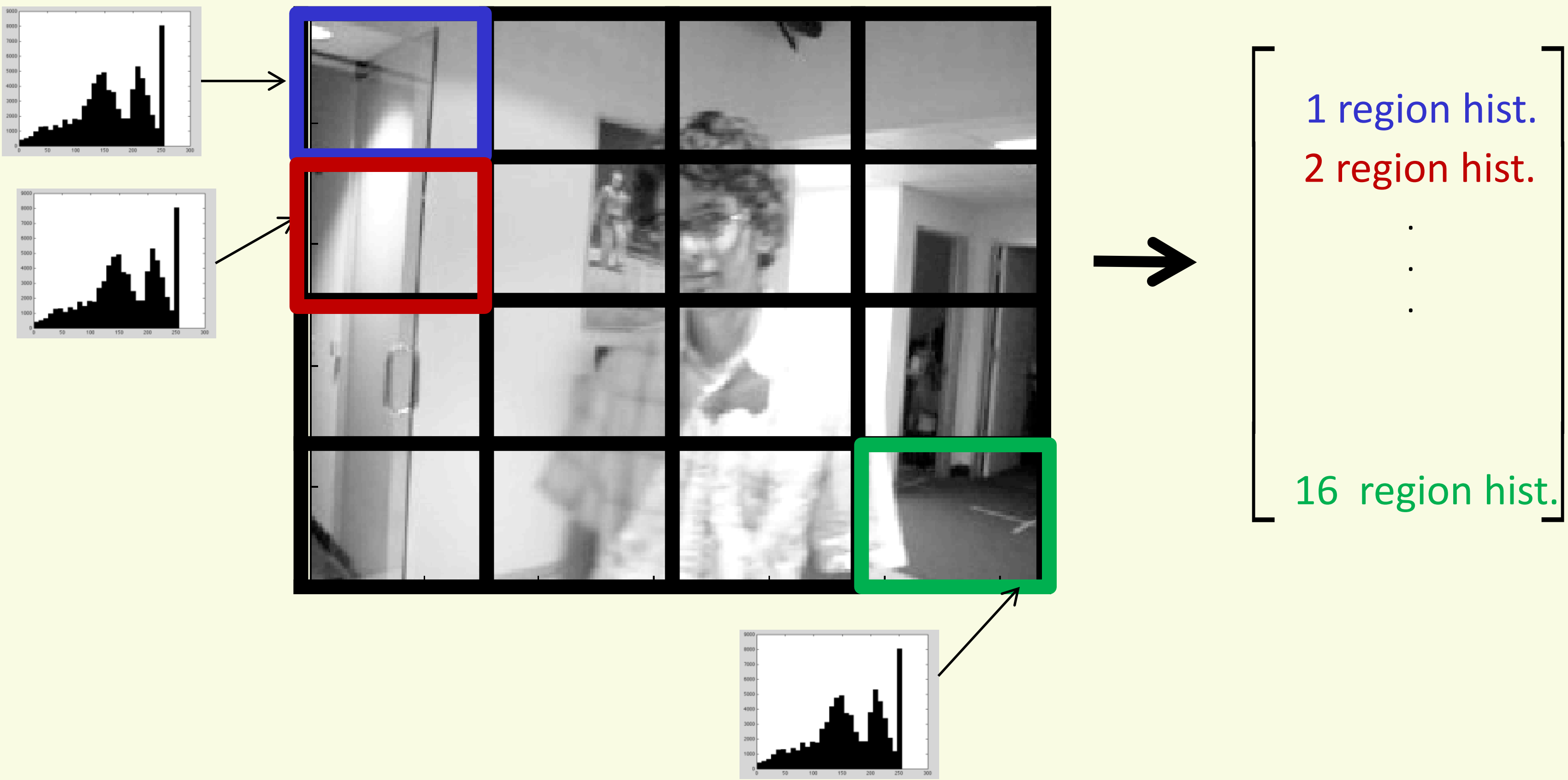
Conclusions

1. Pixel representations:
   overly sensitive to position
2. Global histogram representations:
   under-sensitive to position

Slide by Erik Learned-Miller

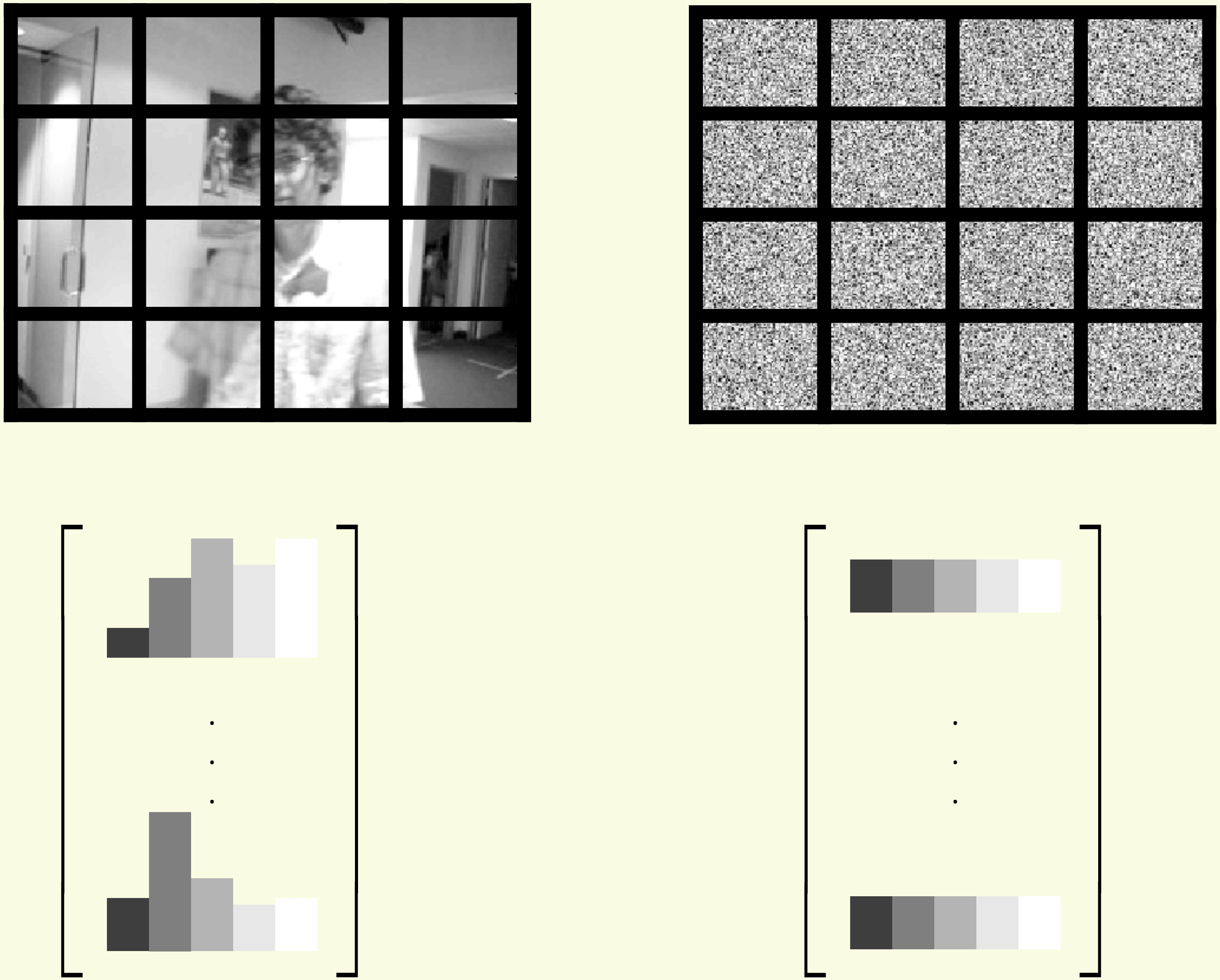
A Compromise: A local histogram

A separate (normalized) histogram for each region

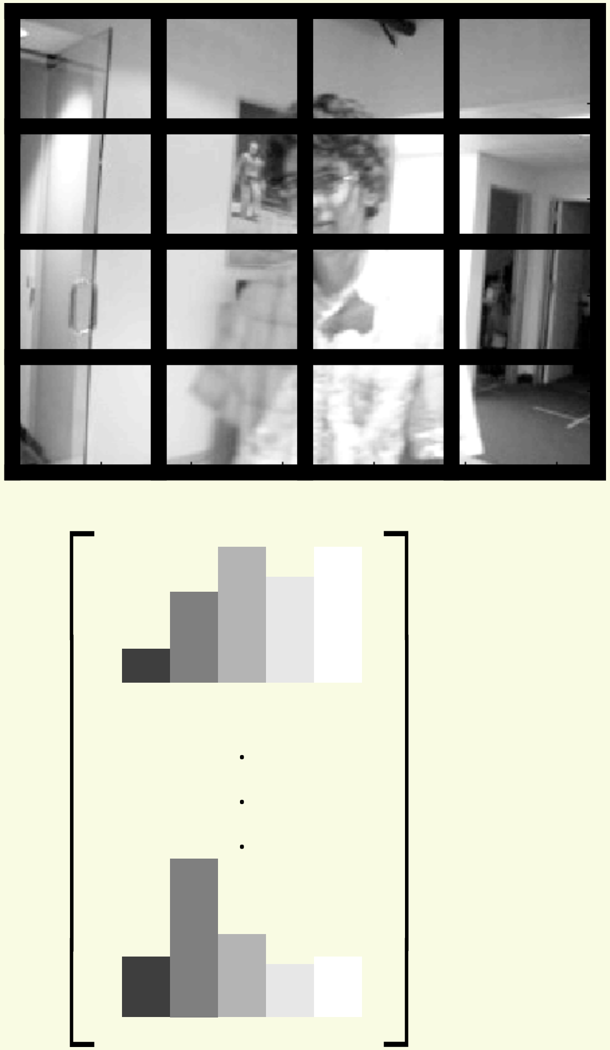


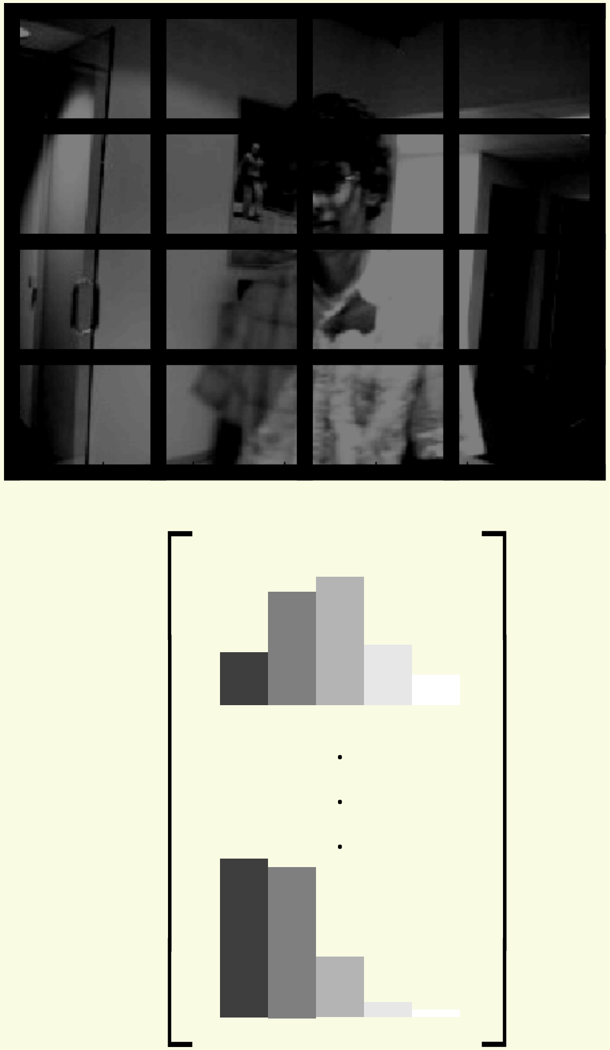
Slide by Erik Learned-Miller

Local Intensity Histogram



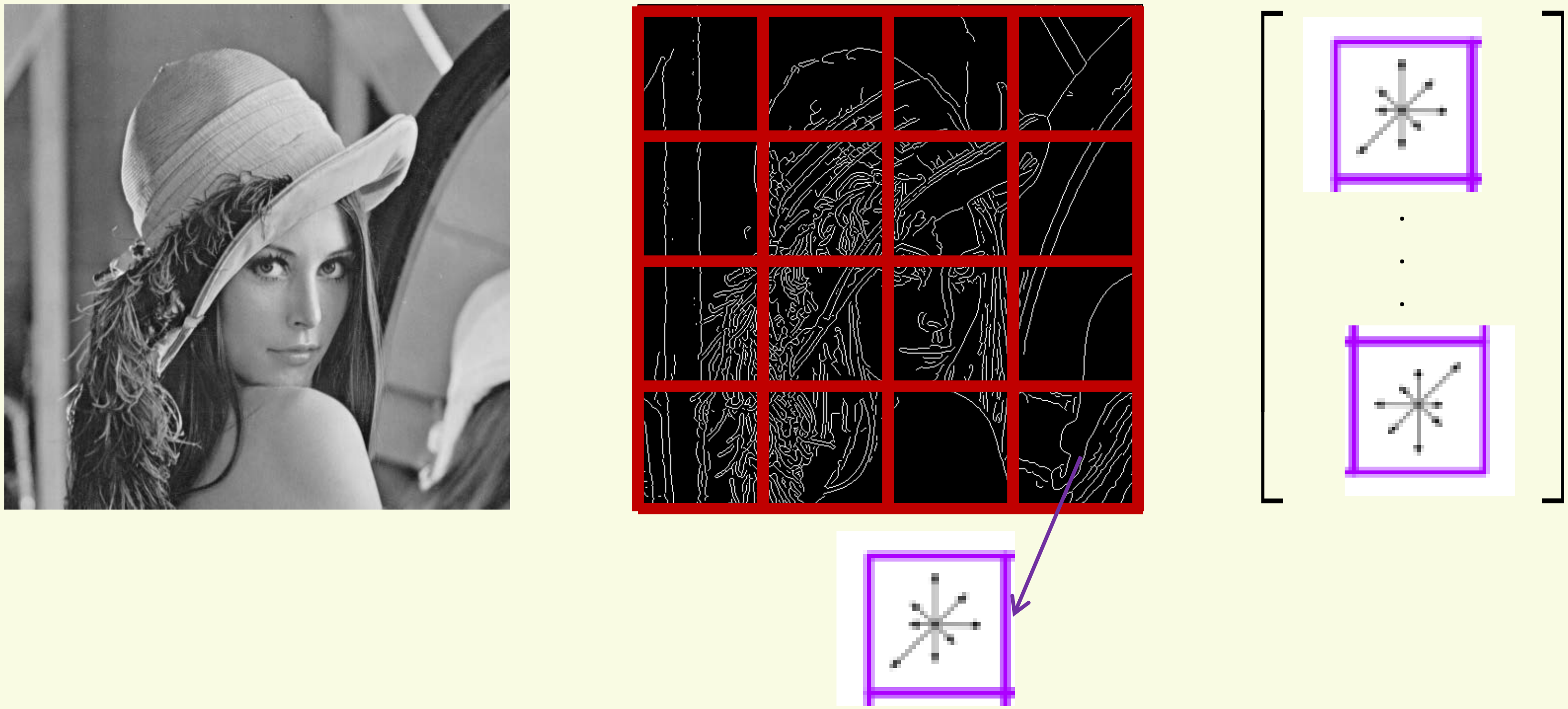
Local Intensity Histogram





- Intensity histogram is sensitive to lighting changes

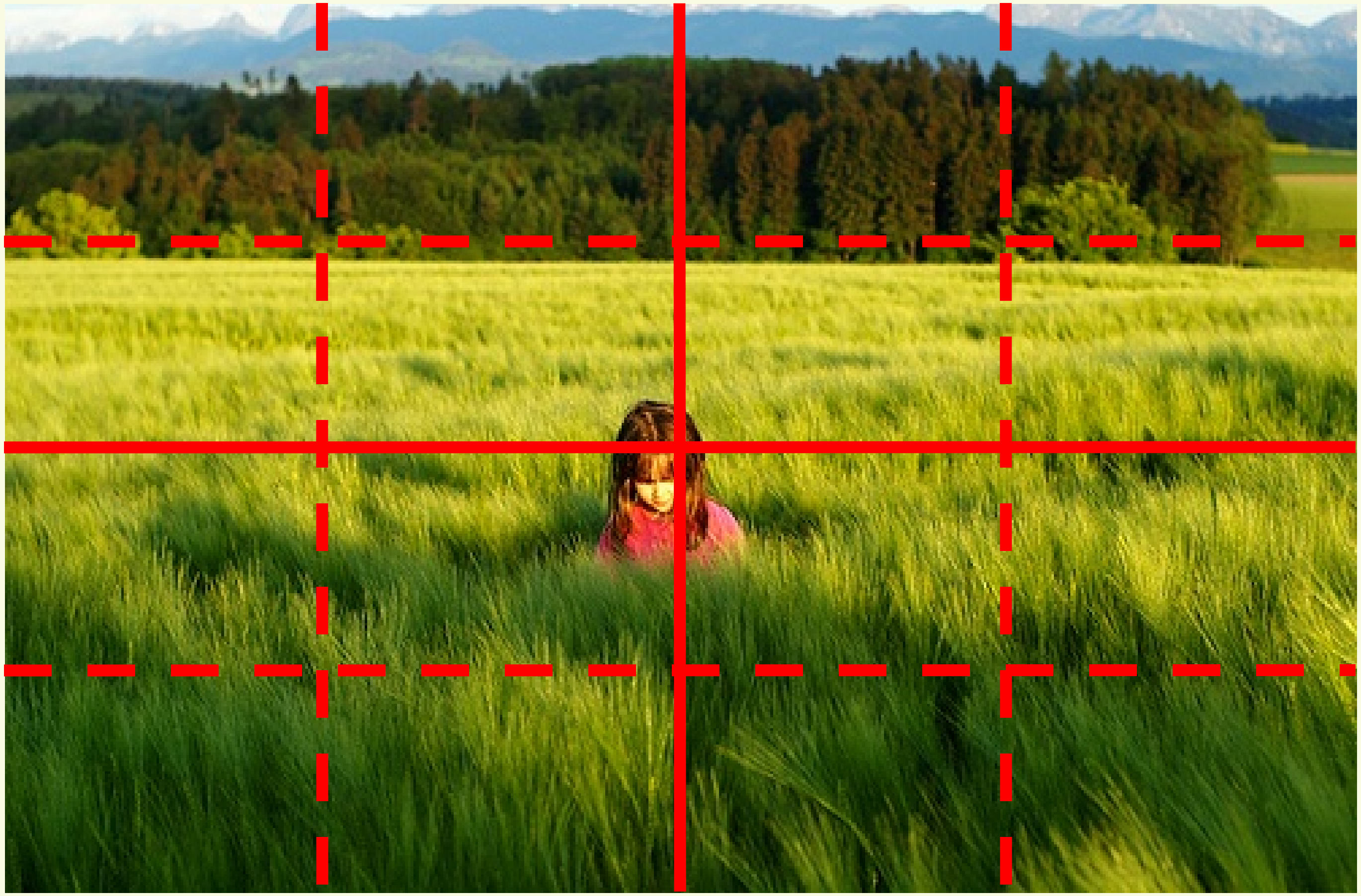
Local Edge Orientation Histogram



- Edges are not as sensitive to lighting changes
- Compute histogram of edges
  - typically consider only edge orientation
- How do we choose the right box size?

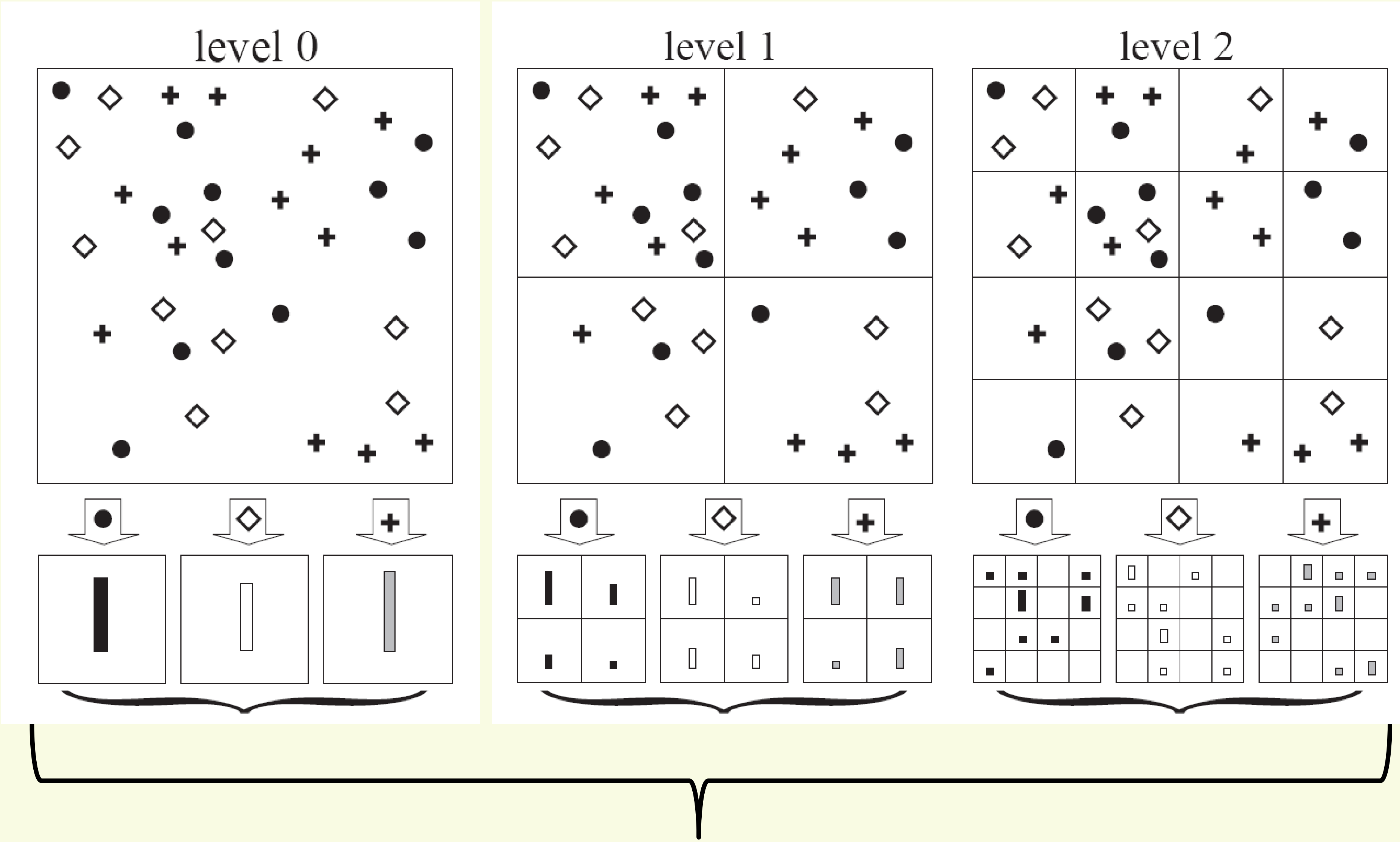
Spatial pyramid

- Use boxes of different sizes!



Slide Credit: Derek Hoiem

Spatial Pyramid



These get piled up into one feature vector

Slide Credit: Derek Hoiem

Other Representations

- Many image representation schemes are based on histogram of
  - texture
  - corner features
  - SIFT features
  - etc.
- There are other ways to represent an image as a feature vector