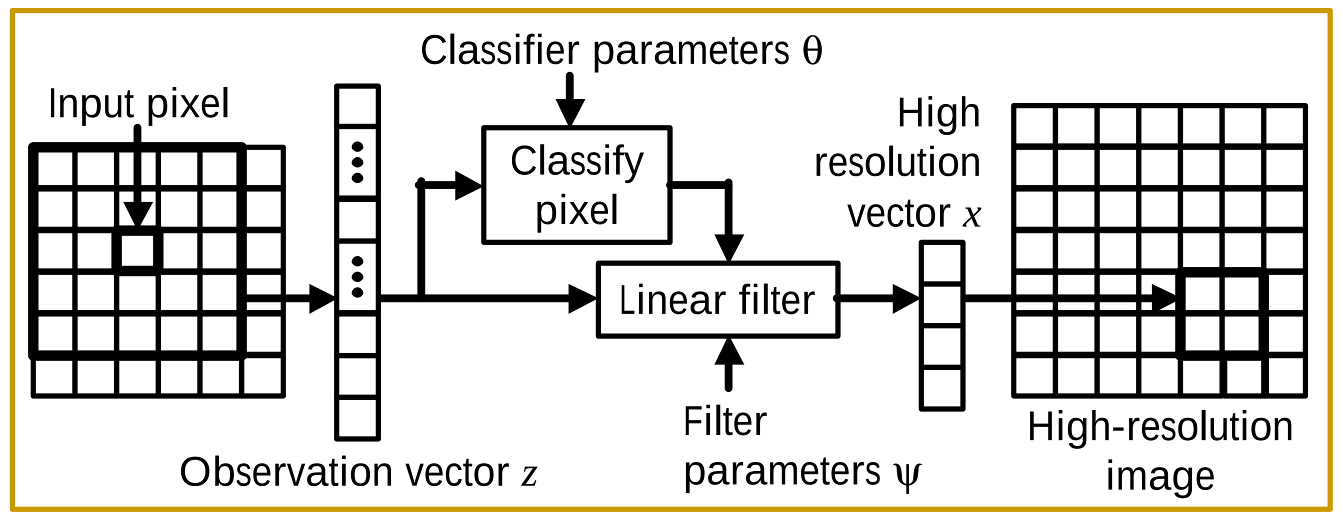
Optimal image scaling using pixel classification

C. Brian Atkins
Hewlett-Packard Laboratories
1501 Page Mill Rd., M/S 4U-6
Palo Alto, CA 94304
cbatkins@hpl.hp.com

Charles A. Bouman and Jan P. Allebach
Purdue University
1285 Electrical Engineering Building
W. Lafayette, IN 47907
{bouman,allebach}@purdue.edu

Overview: Resolution Synthesis

- Goal: MMSE estimate of high-resolution image, given low-resolution image
- Image scaling scheme: for each pixel,



- Stochastic model: assume pixels fall into classes
  - Edges of various orientations
  - Smooth gradients of various orientations
  - Flat regions
- Pixel classification
  - Uses feature vector y extracted from the observation vector z
  - Specified in an unsupervised clustering
- Analysis breaks model into two parts:
  - Classification: class membership in a Gaussian mixture model
  - Optimal prediction filters for each component in the classifier
- Estimate model parameters beforehand by training
  - An instance of the Expectation-Maximization (EM) algorithm
  - High-quality results even with images outside the training set

Prior work

- Regression tree (Atkins, Bouman and Allebach '99)
- Edge-directed methods (Allebach and Wong '96, Jensen and Anastassiou '95)
- B-spline class (Unser, Aldroubi and Eden '91, '95; Hou and Andrews '78)
- Maximum *a posteriori* estimation (Schultz and Stevenson '94)

Optimal image scaling

- Pixel class: an unobservable discrete random variable J taking values in $\{1, \ldots, M\}$
  - Assume any information about pixel class is contained in a feature vector Y extracted from the observation vector Z
    - Formally, Y is a function of Z and $p_{J|Y}(j \mid y) = p_{J|Z,X}(j \mid z, x)$
    - This is a strong assumption, but it simplifies the analysis and enables better results
  - Assume distribution of Y is a Gaussian mixture

$$p_Y(y \mid q) = \sum_{j=1}^{M} \frac{p_j}{(2\pi\sigma^2)^{d/2}} \exp\left(\frac{-1}{2\sigma^2}\|y - m_j\|^2\right)$$

  where d is the dimension of y; we refer to $q = \{m_j, p_j\}_{j=1}^{M}, \sigma$ as the "classifier parameters"
  - Assume distribution of X given Z and J is Gaussian with

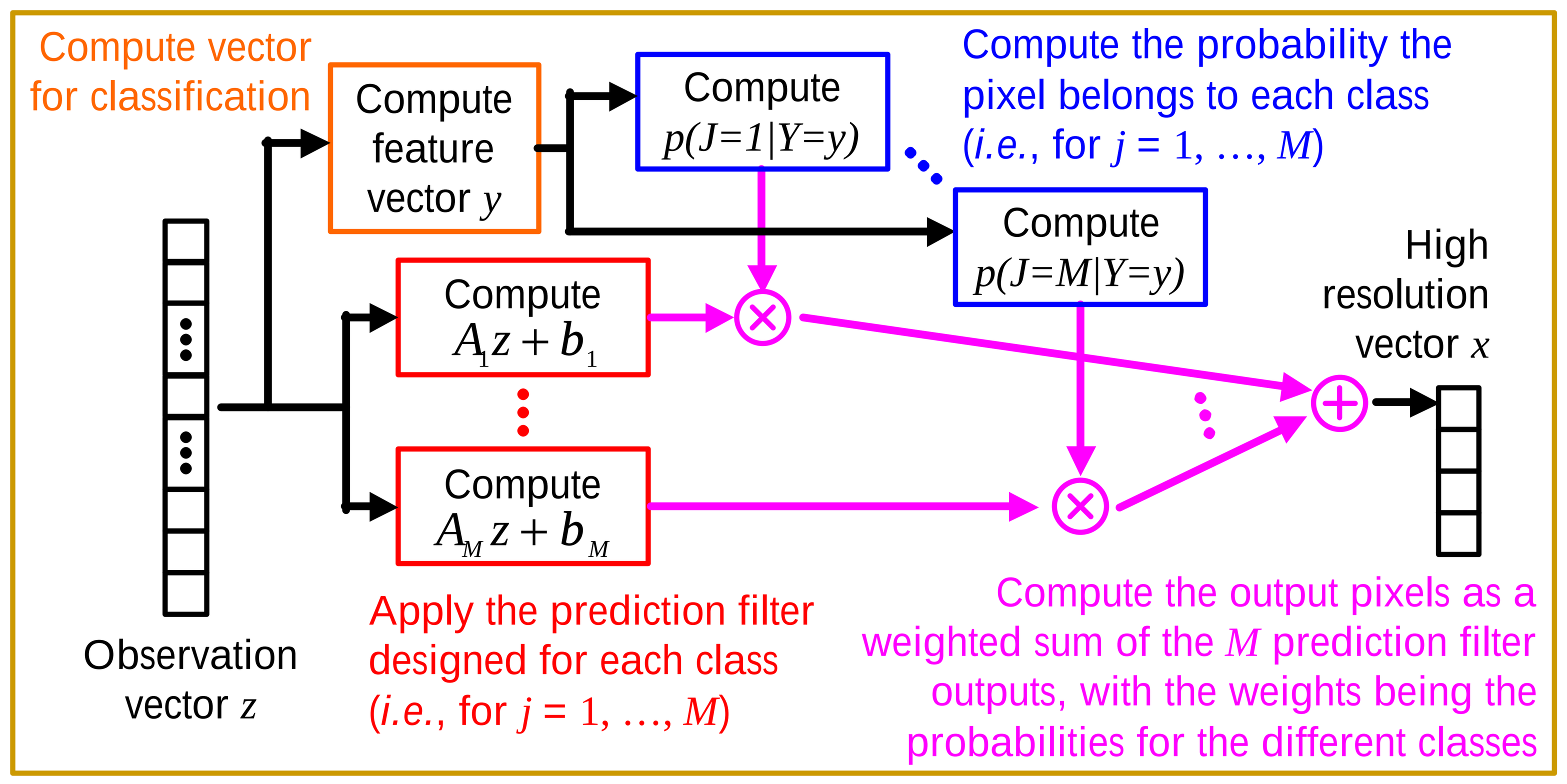
$$E[X \mid J, Z] = A_J Z + b_J$$

  we refer to $\psi = \{A_j, b_j\}_{j=1}^{M}$ as the "filter parameters"
- Using the above assumptions, the MMSE estimate of X given Z is computed as

$$\hat{X} = E[X \mid Z] = \sum_{j=1}^{M} E[X \mid Z, J = j] p_{J|Z}(j \mid Z) = \sum_{j=1}^{M} (A_j Z + b_j) p_{J|Y}(j \mid Y)$$

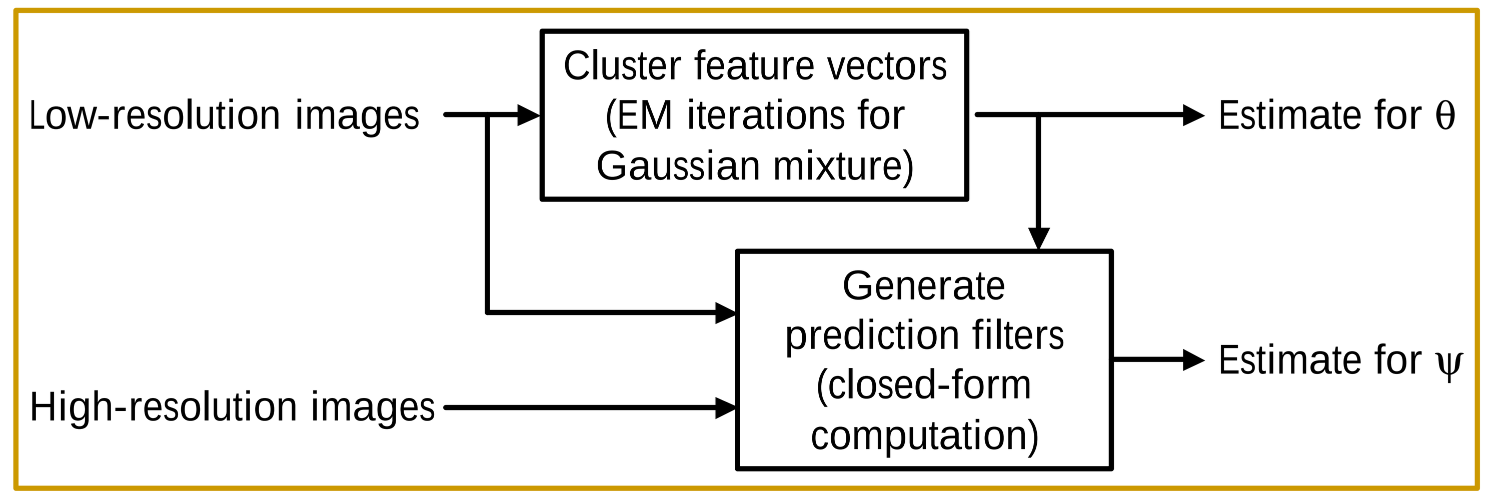
$$= \sum_{j=1}^{M} (A_j Z + b_j) \frac{\exp\left(\frac{-1}{2\sigma^2}\|y - m_j\|^2\right) p_j}{\sum_{l=1}^{M} \exp\left(\frac{-1}{2\sigma^2}\|y - m_l\|^2\right) p_l}$$

- Detailed view of image scaling procedure:



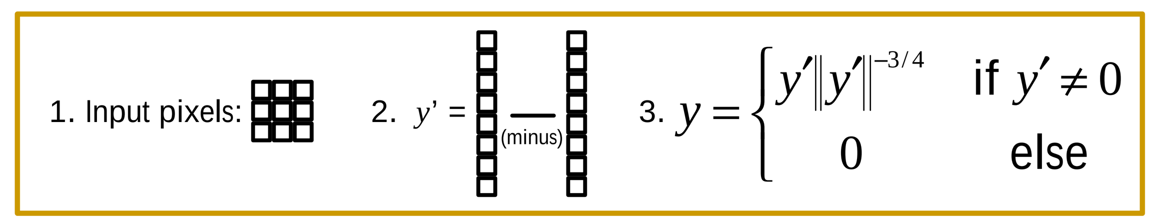
Estimating the predictor parameters

- Goal: estimate parameters θ and ψ from sample image pairs
  - To create a sample image pair, start with a high-resolution image, then block average by the desired scaling factor to create the corresponding low-resolution image
  - For built-in sharpening, can sharpen the high-resolution image
- Approach: maximum likelihood (ML)
  - Direct ML estimation is difficult since data is incomplete
    - Can only observe realizations of (Z,X)
    - The complete data would be (J,Z,X)
  - Solution: Expectation-Maximization (EM) algorithm
- Under our assumptions this can be achieved in a two-stage estimation

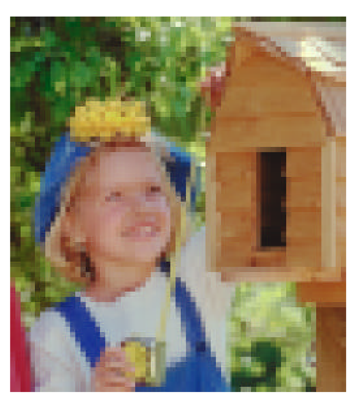


Feature vector used for pixel classification

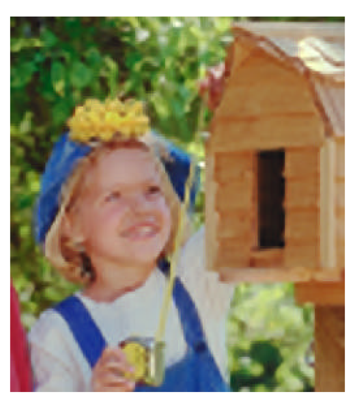
- Formally a function of the observation vector
- Choice of feature vector significantly affects which classes are defined, and ultimately the overall results
- We use an 8-dimensional feature vector:
  - First, define vector y': subtract input pixel from 8 nearest neighbors
  - Feature vector y is computed by modifying the length of y'



Results for 4X image scaling


Pixel replication


Photoshop bicubic interpolation


Resolution Synthesis


Pixel replication


Photoshop bicubic interpolation


Resolution Synthesis

Per-pixel RMSE computed from a random selection of monochrome images, with pixel values in [0,255], with gamma correction removed

image	RS	Photoshop bicubic	bilinear	Pixel replication
0	6.726	7.406	8.005	8.421
1	17.096	18.878	20.382	20.837
2	7.847	8.663	9.542	10.147
3	10.752	11.501	12.099	12.584
4	12.336	12.955	13.574	13.874