Belief propagation and MRF’s

Bill Freeman
6.869
March 7, 2011
Outline of MRF section

• Inference in MRF’s.
  – Gibbs sampling, simulated annealing
  – Iterated conditional modes (ICM)
  – Belief propagation
    • Application example—super-resolution
  – Graph cuts
  – Variational methods

• Learning MRF parameters.
  – Iterative proportional fitting (IPF)
Super-resolution

- Image: low resolution image
- Scene: high resolution image
Polygon-based graphics images are resolution independent
Polygon-based graphics images are resolution independent.

Pixel-based images are not resolution independent.
Polygon-based graphics images are resolution independent

Pixel replication

Polygon-based graphics images are resolution independent
Polygon-based graphics images are resolution independent.

Pixel-based images are not resolution independent.

Pixel replication

Cubic spline
Polygon-based graphics images are resolution independent.

Pixel-based images are not resolution independent.

Pixel replication

Cubic spline, sharpened
Polygon-based graphics images are resolution independent.

Pixel-based images are not resolution independent.

Pixel replication:

Cubic spline, sharpened:

Training-based super-resolution:
3 approaches to perceptual sharpening

(1) Sharpening; boost existing high frequencies.

(2) Use multiple frames to obtain higher sampling rate in a still frame.

(3) Estimate high frequencies not present in image, although implicitly defined.

In this talk, we focus on (3), which we’ll call “super-resolution”.

Wednesday, August 7, 13
Super-resolution: other approaches

- Schultz and Stevenson, 1994
- Pentland and Horowitz, 1993
- fractal image compression (Polvere, 1998; Iterated Systems)
- Follow-on: Jianchao Yang, John Wright, Thomas S. Huang, Yi Ma: Image super-resolution as sparse representation of raw image patches. CVPR 2008
Training images, ~100,000 image/scene patch pairs

Images from two Corel database categories: “giraffes” and “urban skyline”.

Wednesday, August 7, 13
Do a first interpolation

Zoomed low-resolution

Low-resolution
Representation

Zoomed low-freq.  

Full freq. original
(to minimize the complexity of the relationships we have to learn, we remove the lowest frequencies from the input image, and normalize the local contrast level).
Gather ~100,000 patches

Training data samples (magnified)

high freqs.

low freqs.
Nearest neighbor estimate

Input low freqs.

Estimated high freqs.

Training data samples (magnified)
Nearest neighbor estimate

Input low freqs.

Estimated high freqs.

Training data samples (magnified)

high freqs.  low freqs.
Example: input image patch, and closest matches from database

Input patch

Closest image patches from database

Corresponding high-resolution patches from database
Image patch

Underlying candidate scene patches. Each renders to the image patch.
Scene-scene compatibility function,

$$\Psi(x_i, x_j)$$

Assume overlapped regions, d, of hi-res. patches differ by Gaussian observation noise:

$$\Psi(x_i, x_j) = \exp -\frac{|d_i - d_j|^2}{2\sigma^2}$$

Uniqueness constraint, not smoothness.
Image-scene compatibility function, $\Phi(x_i, y_i)$

Assume Gaussian noise takes you from observed image patch to synthetic sample:

$$\Phi(x_i, y_i) = \exp^{-|y_i - y(x_i)|^2 / 2\sigma^2}$$
Markov network

\[ \Phi(x_i, y_i) \]

\[ \Psi(x_i, x_j) \]

image patches

scene patches
Belief Propagation

Input
Belief Propagation

After a few iterations of belief propagation, the algorithm selects spatially consistent high resolution interpretations for each low-resolution patch of the input image.

Iter. 0
After a few iterations of belief propagation, the algorithm selects spatially consistent high resolution interpretations for each low-resolution patch of the input image.
After a few iterations of belief propagation, the algorithm selects spatially consistent high resolution interpretations for each low-resolution patch of the input image.
Zooming 2 octaves

We apply the super-resolution algorithm recursively, zooming up 2 powers of 2, or a factor of 4 in each dimension.

85 x 51 input
Zooming 2 octaves

We apply the super-resolution algorithm recursively, zooming up 2 powers of 2, or a factor of 4 in each dimension.

85 x 51 input

Cubic spline zoom to 340x204
Zooming 2 octaves

We apply the super-resolution algorithm recursively, zooming up 2 powers of 2, or a factor of 4 in each dimension.

85 x 51 input

Cubic spline zoom to 340x204

Max. likelihood zoom to 340x204
Now we examine the effect of the prior assumptions made about images on the high resolution reconstruction. First, cubic spline interpolation.

(cubic spline implies thin plate prior)
(cubic spline implies thin plate prior)
Next, train the Markov network algorithm on a world of random noise images.
The algorithm learns that, in such a world, we add random noise when zoom to a higher resolution.

Original 50x58

Markov network

Training images

True
Next, train on a world of vertically oriented rectangles.

Original
50x58

Training images

True

26
The Markov network algorithm hallucinates those vertical rectangles that it was trained on.

Original
50x58

Markov network

Training images

True
Now train on a generic collection of images.
The algorithm makes a reasonable guess at the high resolution image, based on its training images.

Original 50x58

Training images

Markov network

True
Next, train on a generic set of training images. Using the same camera as for the test image, but a random collection of photographs.
Cubic Spline

Original 70x70

Markov net, training: generic

Cubic Spline

True 280x280

Wednesday, August 7, 13
Kodak Imaging Science Technology Lab test.

3 test images, 640x480, to be zoomed up by 4 in each dimension.

8 judges, making 2-alternative, forced-choice comparisons.
Algorithms compared

• Bicubic Interpolation
• Mitra's Directional Filter
• Fuzzy Logic Filter
• Vector Quantization
• VISTA
Bicubic spline

Altamira

VISTA
User preference test results

“The observer data indicates that six of the observers ranked Freeman’s algorithm as the most preferred of the five tested algorithms. However the other two observers rank Freeman’s algorithm as the least preferred of all the algorithms….

Freeman’s algorithm produces prints which are by far the sharpest out of the five algorithms. However, this sharpness comes at a price of artifacts (spurious detail that is not present in the original scene). Apparently the two observers who did not prefer Freeman’s algorithm had strong objections to the artifacts. The other observers apparently placed high priority on the high level of sharpness in the images created by Freeman’s algorithm.”
Training images
Source image patches

Bandpass filtered and contrast normalized

True high resolution pixels

High resolution pixels chosen by super-resolution

Bandpass filtered and contrast normalized best match patches from training data

Best match patches from training data

Training images
Training images
code available online


Markov Random Fields for Super-Resolution

William T. Freeman        Ce Liu
Massachusetts Institute of Technology  Microsoft Research New England

[Download the package]

This is an implementation of the example-based super-resolution algorithm of [1]. Although the applications of MSFs have now extended beyond example-based super resolution and texture synthesis, it is still of great value to revisit this problem, especially to share the source code and examplar images with the research community. We hope that this software package can help to understand Markov random fields for low-level vision, and to create benchmark for super-resolution algorithms.

When you refer to this code in your paper, please cite the following book chapter:


Algorithm

The core of the algorithm is based on [1]. We collect pairs of low-res and high-res image patches from a set of images as training. An input low-res image is decomposed to overlapping patches on a grid, and the inference problem is to find the high-res patches from the training database for each low-res patch. We use the kd-tree algorithm, which has been used for real-time texture synthesis [2], to retrieve a set of high-res, k-nearest neighbors for each low-res patch. Lastly, we run a max-product belief propagation (BP) algorithm to minimize an objective function that balances both local compatibility and spatial smoothness.

Examples

Several examples of applying the example-based super resolution code in the package are shown below. These examplar images are also included in the package. Once you run the code, it should give you the same result.

We first apply bicubic sampling to enlarge the input image (a) by a factor of 4 (b), where image details are missing. If we use the nearest neighbor for each low-res patch independently, we obtain high-res but noisy results in (c). To address this issue, we incorporating spatial smoothness into a Markov Random Fields formulation by enforcing the synthesized neighboring patches to agree on the overlapped areas. Max-product belief propagation is used to obtain high-res images in (d). The inferred high-frequency images are shown in (e), and the original high-res are shown in (f).
Motion application

image patches

scene patches

image

scene

Wednesday, August 7, 13
What behavior should we see in a motion algorithm?

- Aperture problem
- Resolution through propagation of information
- Figure/ground discrimination
The aperture problem

http://web.mit.edu/persci/demos/Motion&Form/demos/one-square/one-square.html
The aperture problem

http://web.mit.edu/persci/demos/Motion&Form/demos/one-square/one-square.html
The aperture problem
The aperture problem
motion program demo
Inference: Motion estimation results
(maxima of scene probability distributions displayed)

Image data

Iterations 0 and 1

Initial guesses only show motion at edges.
Motion estimation results
(maxima of scene probability distributions displayed)

Figure/ground still unresolved here.

Iterations 2 and 3
Motion estimation results
(maxima of scene probability distributions displayed)

Iterations 4 and 5

Final result compares well with vector quantized true (uniform) velocities.