Advanced Data Analysis: Report 1

David GAVILAN (05D38070)
Computer Science Dept.
e-mail: david@img.cs.titech.ac.jp

April 19th, 2005

1 Data set

I work with digital photographs in my research. I focus on color pictures taken with mobile phones. Although the cameras attached to these phones have high resolution, the screen does not exceed a size of 36×48 mm, at a resolution of 120×160 pixels, or 240×320 pixels. For that reason, I assume a default size of 120×160 pixels.

One of the goals of my research is properly segment images to extract objects in a wide variety of conditions. Therefore, there will be unknown objects of different sizes. To capture these objects, it is important to use the *scale-space*.

The *scale* parameter is defined as a scalar σ representing the scale we are using to observe the image, and the *scale-space* is the set of images $\{L_s\}$ generated continuoslly from an image L, representing the different scales.

Within the class of linear transformations the Gaussian kernel is the only kernel for generating a scale-space.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

In a discrete way, you can imagine applying the Gaussian kernel over an image under different scales, as the example in Figure 1.

Regarding color, images are stored using the $sRGB$ color model, but they are usually converted to a more intuitive space, that is, a space where chromaticity and brightness are separated, such as the HSB color space.

Summarizing, a 2D image can be seen as a three-dimensional function

$$f(x, y, \sigma) = (h, s, b); \quad (2)$$

where (x, y) is the position of a pixel in the space, σ is its position in the scale, and (h, s, b) are the hue, saturation and brightness of that pixel, respectively. Therefore, if we consider at least 6 different scales, as in figure 1, for our images of size 120×160 pixels, our data will be vectors of size

$$160 \times 120 \times 6 \times 3 = 345600 \text{ dimensions.} \quad (3)$$

2 Goal

The goal of the dimensionality reduction in my research is to achieve good image segmentation results. Figure 2 shows some examples of different segmentation algorithms.

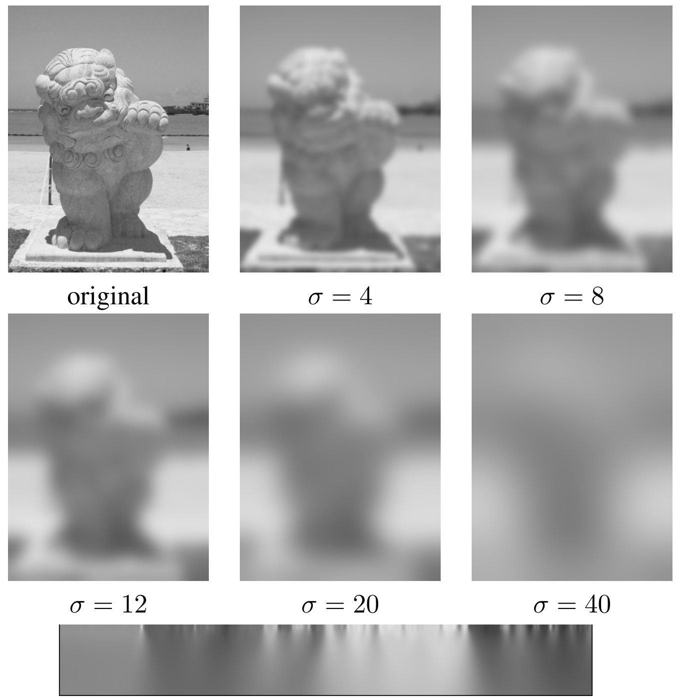

Figure 1: Scale Space of an image. *Bottom:* Continuous scale-space function over an horizontal line of the image, where σ varies from 0.1 to 8.

The first one is the state-of-the-art in color segmentation, and is the mean-shift based color segmentation [2]. This is a kernel-regression technique. The difficulty in this method is setting an appropiate bandwith.

The second is the Blobworld system [1]. With the idea of capturing the natural number of objects present in the image, they predefine the number of clusters in which to divide the space. The space contains the color of each pixel, a scale parameter, and a texture property. The space is clustered using a mixture of gaussians model and the E-M procedure.

In our Color Blobs approach [3], the scale is pre-selected and the color space is clustered a priory using a trained neural network that divides the color space in a few color categories, based on human perception. Since this is a priori knowledge, the method is fast -and it has been implemented in 3G mobile phones. However, the method is not robust to high variations in

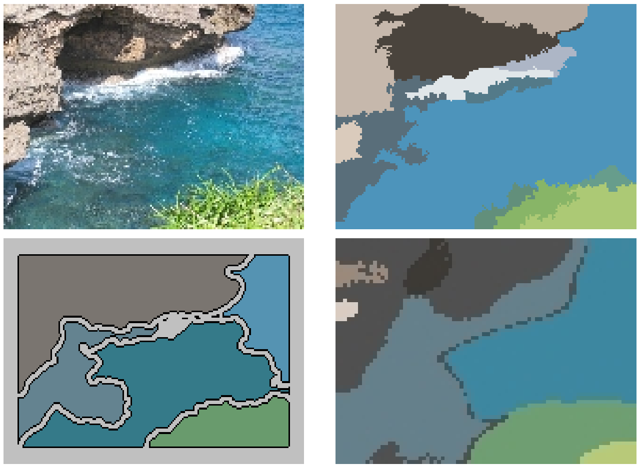

Figure 2: From left to right, up down: original image; image segmented using the mean-shift procedure; image segmented with the Blobworld system; image segmented using Color Blobs.

the color space, due to the fact that the image itself is not used to modify the algorithm.

3 Computing Environment

There are 5 floating licenses in the Grid system in our department. I can use Matlab there when available. Also, I can use Octave in my desktop computer, although it has very restricted support for images.

4 Additional Comments

During the introduction of the first class, we discussed about how to clarify learning mathematically, in part because of the inability to make a model of the brain, better to say, a model of intelligence. However, last year I read a book called *On Intelligence* [4], that, in my honest opinion, models very well the human intelligence and the neocortex. The book is also recommended as a must-read by two nobel laureates.

Hawkins defines Intelligence using what he calls the *Memory-Prediction framework*. While traditionally, intelligence has been defined in terms of intelligent behaviour, he assumes that this is wrong, and that the Turing test of intelligence is pointless. Even if you are fooled by a machine, it doesn't mean that the machine is intelligent. The brain, or precisely, the neocortex, is not a CPU that gets some inputs, process them, and generates some outputs. The neocortex, where intelligence resides, is a memory system, but a very special one.

The neocortex has more connections going backwards than those going forward. These connections have been usually ignored, but Hawkins gives a plausible meaning to them: the ability to make *predictions*. Our brain is just a memory that makes predictions and tries to fit the input with the predicted output. So when the input comes, we can almost immediately know if that is what we expected or not. Faster silicon chips performs worse than neurons because they require millions of instructions to do the same. Of course, the input modifies the neurons configurations, since our memory works with *invariant representations*.

Basically, he wants to show that there is a universal algorithm for all the neurons. And that it is simple and efficient.

References

[1] Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein, and Jitendra Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(8):1026–1038, August 2002.

[2] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[3] David Gavilan, Hiroki Takahashi, and Masayuki Nakajima. Image categorization using color blobs in a mobile environment. *Computer Graphics Forum (EG 2003)*, 22(3):427–432, September 2003.

[4] Jeff Hawkins. *On Intelligence*. Times Books, 1st edition, 2004.