CS 3220 Homework 8

out: Tuesday 7 April 2009
due: Monday 13 April 2009

Problem 1: Approximate de-projection.

Given a 31-vector $\mathbf{s}$, consider its products with two 3×31 matrices, $\mathbf{C}$ and $\mathbf{E}$. In a particular system I get to observe the product $\mathbf{Cs}$ (which is a 3-vector), but what I actually need is the product $\mathbf{Es}$.

I would like to ask the question "Given $\mathbf{Cs}$ for some $\mathbf{s}$, find the vector $\mathbf{Es}$." Sadly, this cannot be done, in general.

1. Given the matrices $\mathbf{E}$ and $\mathbf{C}$, explain how to construct two vectors $\mathbf{s}_a$ and $\mathbf{s}_b$ so that $\mathbf{Cs}_a = \mathbf{Cs}_b$ but $\mathbf{Es}_a \neq \mathbf{Es}_b$.

Given the failure to solve the problem exactly, we can look for another problem, which we *can* solve, that approximates the original problem.

2. Suppose $\mathbf{s}$ is guaranteed to be a linear combinations of three fixed vectors $\mathbf{s}_1$, $\mathbf{s}_2$, and $\mathbf{s}_3$. Explain how to compute a 3×3 matrix $\mathbf{M}$ that answers the original question. That is, given $\mathbf{Cs}$, multiplying it by $\mathbf{M}$ should compute $\mathbf{Es}$.

We have removed the ambiguity that made the original problem impossible by introducing an assumption about the input $\mathbf{s}$—we could say we are "imposing a low-dimensional linear model on $\mathbf{s}$."

Problem 2: Camera color correction.

The application that underlies the previous problem is color correction for digital cameras. Color cameras measure the distribution of light energy across different wavelengths; for this purpose it is sufficiently accurate to think of light as existing at a number of distinct wavelengths, 400, 410, 420, $\ldots$, 700 nanometers, so that a spectrum is a 31-vector giving the amount of energy at each wavelength.

Each pixel of a digital camera makes three measurements of the wavelength spectrum coming in through the lens—these are the red, green, and blue color signals. Each of the three color signals is a sum of energy at all wavelengths, waited by the *spectral sensitivity* of the sensor for that color. For example, if the red signal is R,

$$R = \sum_{i=1}^{31} r_i s_i$$

where r_i is the sensitivity of the red sensor to light at the i^{th} wavelength. This is a dot product of 31-vectors: $\mathbf{r}^T\mathbf{s}$. If we stack up the three sensitivity vectors as the rows of a matrix $\mathbf{C}$, then the three color signals, as a 3-vector $[R\ G\ B]^T$, are $\mathbf{Cs}$. So multiplication by $\mathbf{C}$ models the color detection process in the camera.

As it turns out, the human eye senses colors in very much the same way as a camera. So there is another matrix $\mathbf{E}$ that describes how the eye makes three linear measurements of a spectrum. Color scientists use the letters X, Y, and Z for the eye's three color signals, so $[X\ \ Y\ \ Z]^T = \mathbf{Es}$. Multiplication by $\mathbf{E}$ models color detection in the eye.

The math of the previous problem tells us how to take the colors from the camera and compute colors as seen by the eye, *if* the spectra all come from a 3D subspace. So we can use that math if we can come up with a reasonable linear model for the spectra we'll find in the world; the accuracy will depend on how well we really can describe all spectra with three basis spectra.

1. Use the SVD to find a 3D subspace that best approximates the set of "representative" spectra provided with this homework. Plot your three "principal spectra."

2. Use the resulting three principal spectra to compute a color correction matrix M that maps camera RGB colors to XYZ colors, for each of the two cameras "Camera 1" and "Camera 2."

3. Which of these matrices will magnify noise in the input image more, in the sense of relative error? How can you tell?

4. Compute the camera RGB colors for the 24 provided test spectra (they are the spectra of the squares of the color test chart that's shown in some of the images), the true XYZ colors of those spectra, and the XYZ colors computed by your matrix from the RGB colors. Report the error (as 2-norm in XYZ space) in each of the 24 colors for each camera.

Congratulations—you've now implemented a state-of-the-art color calibration method similar to what's used by camera manufacturers. For fun, apply this matrix to the provided raw-color camera images to see how the color comes out.

For further fun, do the same process using the provided sensitivity matrix for Camera 3, a hypothetical 5-color camera, and see how the error changes.