MIT CSAIL

6.869 Advances in Computer Vision

Spring 2011

Problem Set 4: Color Constancy and Bayesian Inference

Posted: Wednesday, February 23, 2011	**Due:** Wednesday, March 2, 2011

You should submit a hard copy of your work in class, and upload your code (and all files needed to run it, images, etc) to stellar.
Your report should include images and plots showing your results, as well as pieces of your code that you find relevant.

Problem 4.1 *Color constanacy*

The Optical Society of America is offering a $1000 prize for the contest entrant who best estimates the unknown illumination spectra for each of 10 different images. This week's homework problem is to submit an entry to that contest.

The winner of the Optical Society's contest will be announced this summer, although the current contest rankings will be announced every two weeks. Because the summer is a long time to wait, we'll also be offering our own (smaller) prize for the student who submits the highest ranking entry among all the students enrolled in 6.869: a suitably engraved Ipod Shuffle. You'll need to include your contest entry number with your homework so we can tell from the contest web page who was the best-scoring 6.869 student.

Resources

Contest web site: http://color.psych.upenn.edu/osacontest2011/
Contest data: http://color.psych.upenn.edu/osacontest2011/ContestImageData.zip
Sample matlab code: http://color.psych.upenn.edu/osacontest2011/SimpleIllumEstimation.m

Training data: http://groups.csail.mit.edu/vision/courses/6.869/psets/pset4/TrainData.zip

We will use existing datasets for building priors for illuminants and surface colors. For surfaces, we will use data by [2] and [3], which contains spectral reflectance measurements for two sets of surfaces, courtesy of the Psychtoolbox distribution (http://psychtoolbox.org). For illuminants, we will use measurements of 2600 natural daylights by [1]. All the data is available for you in the training data file. The function `PrepareTrainData.m` performs the necessary conversions to the OSA contest basis functions and wavelength sampling so that you could use the data for building prior models. See the code for more details.

Instructions

(a) Read the contest web page for background and instructions, and download the data files and images.

(b) Run and understand the sample Matlab code `SimpleIllumEstimation.m` and plot the spectra estimated using the Grey World assumption for each of the 10 contest images. It will be necessary for you to understand this code to do the other parts of the problem set. Also make sure you understand how the estimates are evaluated in this contest.

(c) Write this (simple) function:
Input: (i) An illumination spectrum (specified by the value of the 3 illumination spectral basis coefficients), and (ii) eye response (L, M, and S cone) values for each one of some set of surface colors.
Output: The surface spectral basis function coefficients for each of the surfaces that yield those eye response values when the surfaces are illuminated by that given illumination spectrum.

(d) Write prior probability functions for both surface colors and illuminants.
Input: spectral basis function coefficients of either a surface color or of an illuminant (you'll have a different function for each).
Output: The prior probability that that surface color or illuminant exists.

Download the training data, and load into Matlab the file `train_data.mat`. Illuminant and surface spectral coefficients are available in the `illum_coeff` and `sur_coeff` matrices respectively (rows are coefficients for different basis functions; columns are measurements). Assume the coefficients are independent (but not identically distributed) Gaussians. Fit a mean and covariance from the prior observations for each of the spectral coefficients of the surfaces, and of the illuminant. (Different Gaussians for the illuminants and for the surfaces, so a total of 6 Gaussians to fit). Then one final step: some values of the spectral basis function coefficients will lead to negative power in some wavelength bands. You'll need to check that the spectral power is positive everywhere, and otherwise return a prior probability of zero.
Note: We might update the training data file (e.g. for additional measurements from other sources), so design your code so that it would be easy for you to recompute the prior models if needed.

(e) Make a MAP or MMSE estimate (your choice) of the illumination spectra for each of the images of the contest. A MAP estimate is simpler but the MMSE estimate is probably better for this problem.

For both the MAP and MMSE estimates, assume a Gaussian prior for the coefficients of spectral basis functions for both surfaces and the illuminant–use the functions of part (d).

For MAP estimate: search in the 3-d space of illuminant spectra for the most probable configuration of illumination spectrum and surface colors to explain the observed data.

At each candidate illumination spectrum:

1. Use the function of part (c) to find the corresponding surfaces that would give the observed cone responses when viewed under the candidate illuminant. (If the solved-

for surface coefficients (or if the illumination spectrum) involves negative power at any wavelength, then we reject this illuminant as being infeasible).

2. Find the likelihood term for this candidate surface color/ illuminant combination. Apart from small experimental errors, this term will be one for this problem, since for almost any illuminant we can find solve for a set of surfaces that yield the observed L, M, S cone responses.

3. Use your functions of part (d) to find the prior probability of this set of parameters (the illuminant and solved-for set of surface colors). Multiply by the result of (2) to find the posterior probability of this set of parameters.

4. Search over the 3-d space of illumination spectrum coefficient values (using steps 1 - 3) to find the most prior probable illuminant and surface colors explanation of the observed L, M, and S cone responses for this image. That is the MAP estimate. Repeat to estimate the illumination spectrum for all 10 images.

For the MMSE estimate, we will use a Monte Carlo method (averaging many different trials). We will take many random draws of candidate illuminant spectra, find the corresponding surface colors that would explain the observed image data, and then check how probable that set of surface colors would be. We'll use that probability as a weight to form a weighted average of the sampled illumination spectra, which will be the MMSE estimate.

In more detail:

What we really want to do is to draw samples from the posterior distribution and average them together to find the mean, giving the MMSE estimate. We don't know how to draw random samples from the posterior distribution, $p(x|y)$, but we do know how to draw from the prior, $P(x_i)$, a Gaussian. In Monte Carlo methods, this different distribution is called a "*proposal distribution*". To correct for sampling from the proposal distribution rather than from the posterior distribution, we need to weigh the samples in our average by the ratio $\frac{p(x_s, x_i|y)}{P(x_i)}$, in order to estimate the average of the posterior distribution, the MMSE estimate. In that ratio, the prior probability of the illuminant terms cancel, leaving us with

$$\frac{p(x_s, x_i|y)}{P(x_i)} \propto P(y|x_s, x_i)P(x_s) \tag{1}$$

Given that the likelihood term will almost always be 1, the resulting procedure makes sense: we draw random samples of candidate illuminants, calculate what surface colors such an assumed illuminant would require to give us the data we saw, and weigh that candidate spectrum by the prior probability of running across those surface colors. Repeat many times and average the weighted spectra to find the MMSE illuminant estimate. (You would need to normalize the sum of the weighted spectra by the sum of the weights, but in this contest, the spectra are evaluated in a way that is independent of any overall scale factor).

(f) **(optional, and final class project idea)** Make up your own spectral estimation algorithm for this contest.

Anything is allowed (except for hacking into Dave Brainard's computer files to find the true spectra): You can make a human-computer hybrid algorithm, relying on people to identify a color that looks white, or grey. You can crowd source the problem. You can try different priors. If you win, you get $1000, plus glory. (Winners will be announced in July, but you can see where you stand in the current rankings list, updated about every two weeks until then).

Submission

(a) Contest entry: submit your favored solution (from either part (e) or part (f)) to the contest, by emailing to compsubmit@osavisionmeeting.org
Your email should include:

- Your first and last name
- Your address (or departmental address if you prefer)
- A nickname you want to be listed under on the leaderboard (this will default to first-name_lastname)
- Your estimated spectrums in tab delimited text file as described in the contest web page (a 31 by 10 matrix, see the sample program for an example).

(b) Report: after the first email submission you will receive a unique ID number. **Write this ID number on the front page of your report you submit in class**. In your report, describe your work and add the SPD plots and relevant pieces of your code. If you choose to develop your own method, or modify the proposed one, make sure to describe in detail the method and your design decisions. Upload to Stellar all your code and prior models.

Good Luck!

References

[1] J. Hernndez-Andrs, J. Romero, J.L. Nieves, and R.L. Lee Jr. Color and spectral analysis of daylight in southern europe. *Journal of the Optical Society of America*, 18(6):1325–1335, 2001. `http://www.ugr.es/~colorimg/database.html`.

[2] D. Nickerson and D. H. Wilson. Munsell reference colors now specified for nine illuminants. *Illumination Engineering*, 45:507–517, 1950.

[3] M. J. Vrhel, R. Gershon, and L. Iwan. he measurement and analysis of object reflectance spectra. *Color Research and Application*, 19:4–9, 1994.