CS 664 Lecture 11: A Visionary Cornocopia.

Scribed by Milo Polte

Feb. 29 2000

1 Picking Good Energy Functions For Stereo

Approaching vision problems in terms of energy minimization, requires good energy functions and an efficient way to minimize them. We have already discussed the using graph cuts to minimize our energy functions and the excellent results they achieve. However, by using energy functions which more accurately model our assumptions and knowledge of the real world, we may get even better behavior. Consider the CS 664 mantra once again:

$$E(f) = \sum_{p \in \mathcal{P}} D_p(f_p) + \lambda \sum_{\{p,q\} \in \mathcal{N}} V_{\{p,q\}}(f_p, f_q)$$

We may improve behavior by manipulating $V_{\{p,q\}}(f_p, f_q)$ to take into account static clues in the image. For example, consider two sets of adjacent pixels:

$\begin{array}{cc} p & q \\ \boxed{0} & \boxed{255} \end{array}$ - Large Difference between adjacent pixels.

$\begin{array}{cc} p' & q' \\ \boxed{124} & \boxed{124} \end{array}$ - Small difference between adjacent pixels.

Under our assumption that most objects in the world are smooth (i.e. is adjacent pixels of the same object should have similar intensities), it is likely that p and q represent different objects, possibly with different depths. We should make the cost of a discontinuity in the disparities we'll assign these pixels very small (i.e. $V_{\{p,q\}}(f_p, f_q)$ should be small) because such objects with different depths will move differently between images.

On the other hand p' and q' have very similar intensities and are probably part of the same object. We should make the cost of a discontinuity large here.

We may revise our function V then to something that looks more like

$$\mathcal{W}_{p,q} \times V_{\{p,q\}}(f_p, f_q)$$

where $\mathcal{W}_{p,q}$ is a function on the intensities of p and q in our image.

This is just one example of how our assumptions and prior knowledge may go into our choice of an effective energy function.

2 Multi-way Cuts for Slanted Surface

So far we've been focusing on using Multi-way Cuts to restore images which are piecewise constant. However, we can also use them for piecewise smooth surfaces.

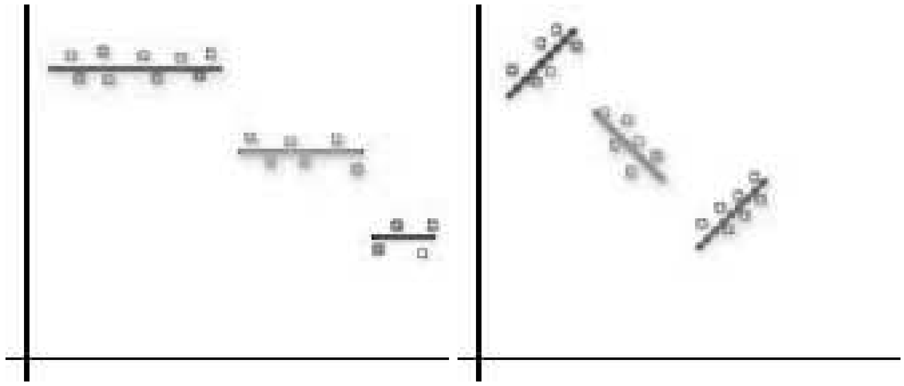

Figure 1: Piecewise Constant vs. Piecewise Smooth

If we try to use the Potts model for a piecewise smooth graph, we'll get output much like the following:

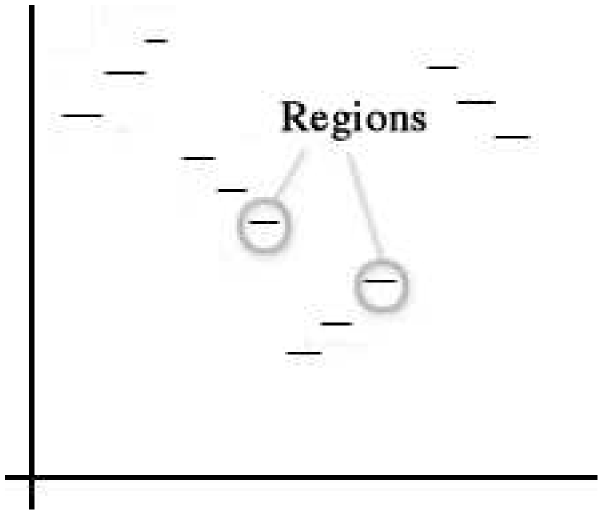

Figure 2: Potts model output for piecewise smooth image

The problem is that the Potts model will make every discontinuity equally bad, labelling the image with step-like regions.

The solution is a cute algorithm from Stanford:

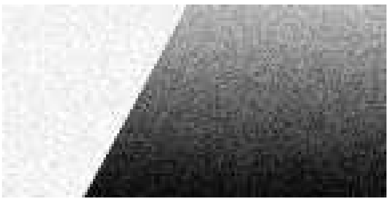

Figure 3: Noisy Smooth Image

1. Start by running the Potts model normally with a set of labelings that are integer labelings.

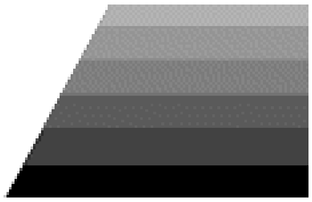

Figure 4: First Potts model output with connected components

2. Take each region–that is a connected component–outputed by the Potts model and find a plane equation that best fits that region in the *original image*. One may use Least Squares to do this.

3. Take all these planar equations as a new set of labels.

4. Compute the MWC with the Potts model again, but this time use the set of equations as labels.

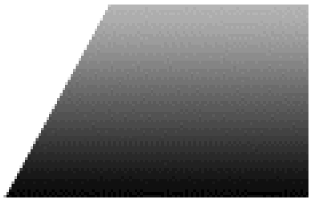

Figure 5: Final Output

Because the Potts model hates discontinuities, it will choose the single best plane equation for the entire piece.

Thus the first few steps of the algorithm gives us a set of planar equations which fit our data nicely, and the last steps choose the best of these equations.

3 More Energy Minimization

Returning to the subject of energy minimization in stereo, let us now consider the data cost part of our energy function.

$$\sum_{p \in \mathcal{P}} D_p(f_p)$$

3.1 The Mixed Pixel Problem

The world is complicated: Pixels are solid angles, and the places where we want the best behavior, the borders of objects, are where we get the worst behavior.

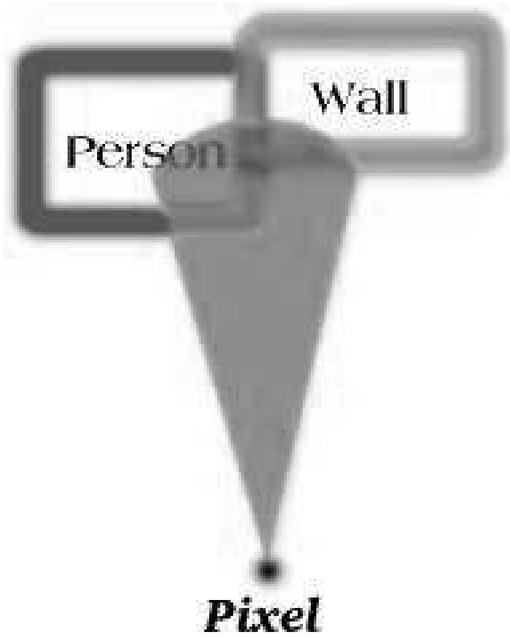

Figure 6: What's My Value?

The intensity of pixels near borders comes from two objects, which is going to make our lives difficult for finding disparities. This is called the *Mixed Pixel Problem*.

3.2 Bad Solutions

Several bad solutions have been put forward. Higher resolution would give us more precision near the borders, but it's really just avoiding the problem: If we could have that much control over our images, vision wouldn't be so hard.

A more intelligent idea is to try to interpolate intensity values so that we can get sub-pixel resolution when we calculate disparities. The idea is that instead of a disparity of 2 we would get something like 2.0132, takinging into account the mixed pixel. The problem is that we tend to get the wrong answer to the **left** of the decimal point, which is the wrong answer with more precision.

3.3 A Better Solution

A better solution is to use the Birchfield-Tomasi modification to our energy function. The idea is to compute an intensity interval for each pixel, based on its neighbors.

Compute the averages between the center pixel and each of its neighbors. The minimum and maximum averages will form an intensity interval of values we'd be happy giving to this pixel. When we're trying to find the corresponding pixel, we'll let the data cost equal zero if the pixel's intensity is within the interval and the distance outside the interval otherwise.

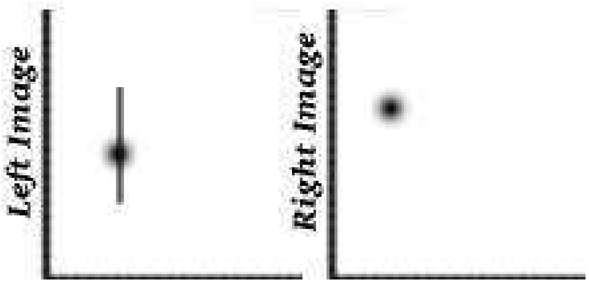

Figure 7: A no cost situation.

We've made mixed pixels easier to satisfy which will improve the behvaior of our stereo energy minimization functions.

4 Gain and Bias

Departing from the subject of energy minimization for stereo, we have yet another stereo problem to consider.

So far we've assumed in all our methods that corresponding pixels have identical intensities modulated by noise or mixed pixels. This is called the *Constant Brightness Assumption*.

Unfortunately, shiney objects and multipal light sources mean that the world isn't nearly that simple. To make matters even worse, we also have to deal with some ugly engineering.

Cameras modulate their observed intensity with two variables: **gain** and **bias**, such that the relation between the true intensity and the camera's observed intensity is something like

$$I = \alpha I_{\mathcal{T}} + \beta$$

where I is the outputed intensity, α is the gain, $I_{\mathcal{T}}$ is the true intensity, and β is the bias.

Why do cameras do this? We simply don't have enough bits to avoid it. Most cameras have an auto-gain feature: They try to keep all intensities within the range of 0-255. The result is that pictures with a great range of intensities (such as a room half in sunlight, half in dimlight) can appear overly dark.

A problem that is more germane to stereo is that two different cameras will have different gains and biases. This will result in a linear change in all our pixel values between images! We shall discuss methods to account for gain and bias in future lectures.