Binning Discontinuities

L32

Goal: To appreciate the difficulties introduced by binning for the purpose of forming a histogram.

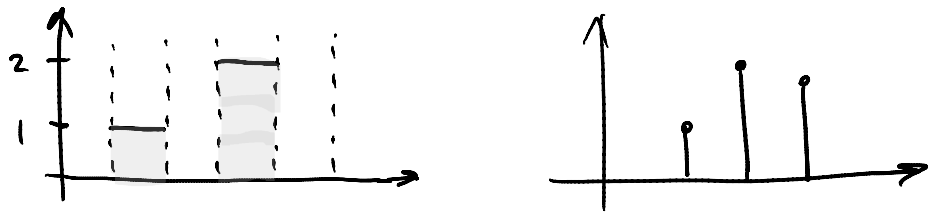
The act of slotting pixels into histograms bins can cause problems.
Example:
Suppose a pixel in f has intensity 99.98. Suppose you rotate f by 0.01° and the intensity of that pixel changes to 100.01. If the histogram bins were set up so that 100 was a bin boundary, then this tiny transformation will cause a pixel to jump bins, and change the cost function in a discontinuous manner. A rotation of -0.01° might change the intensity to 99.95, and the pixel would stay in the same bin and the cost would NOT change at all.
In this sense, histogram binning cuases quantization; the cost function changes by discrete jumps.

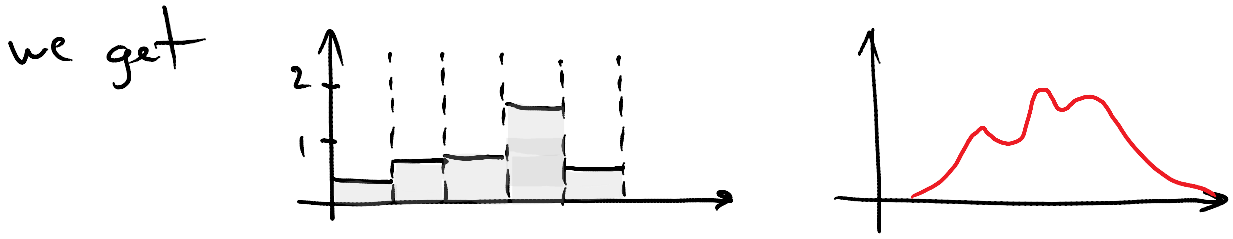
(divinely-inspired Matlab demo)

Parzen Windowing

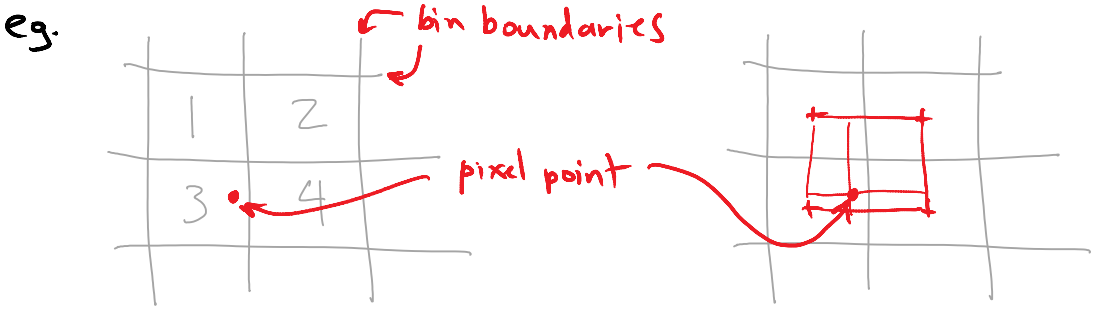
One method to address this issue is to spread, or blur, the contribution of a pixel over a number of histogram bins.

i.e. instead of



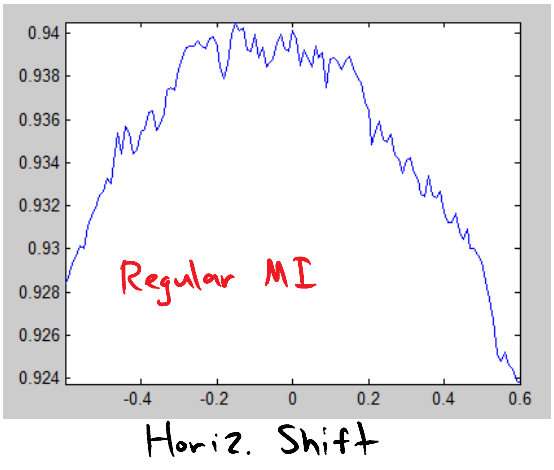


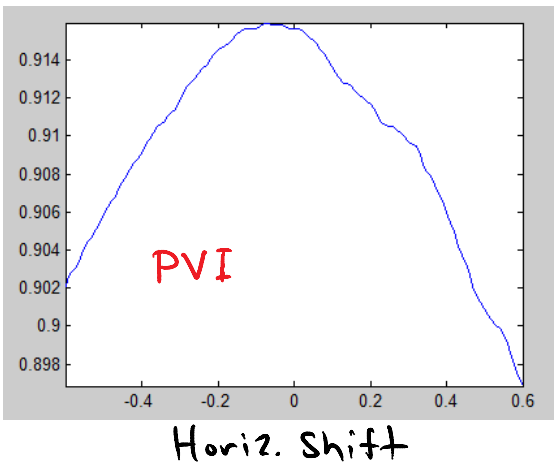
This method is called **Parzen Windowing.**
One particular form is **Partial Volume Interpolation (PVI).**
This method essentially divides the contribution of a pixel among the **nearest bins**, in the same sense that interpolation uses the values from the nearest samples.



Without Parzen techniques, the pixel would only contribute to bin 3.

However, using PVI, it contributes to bins 1, 2, 3 and 4 according to how close the pixel is to each bin centre.





The bumpiness becomes a big problem when you try to optimize this cost function. Imagine trying to estimate the slope. The smoothness of the PVI version helps methods like gradient descent to converge properly.

(exquisite Matlab demo)