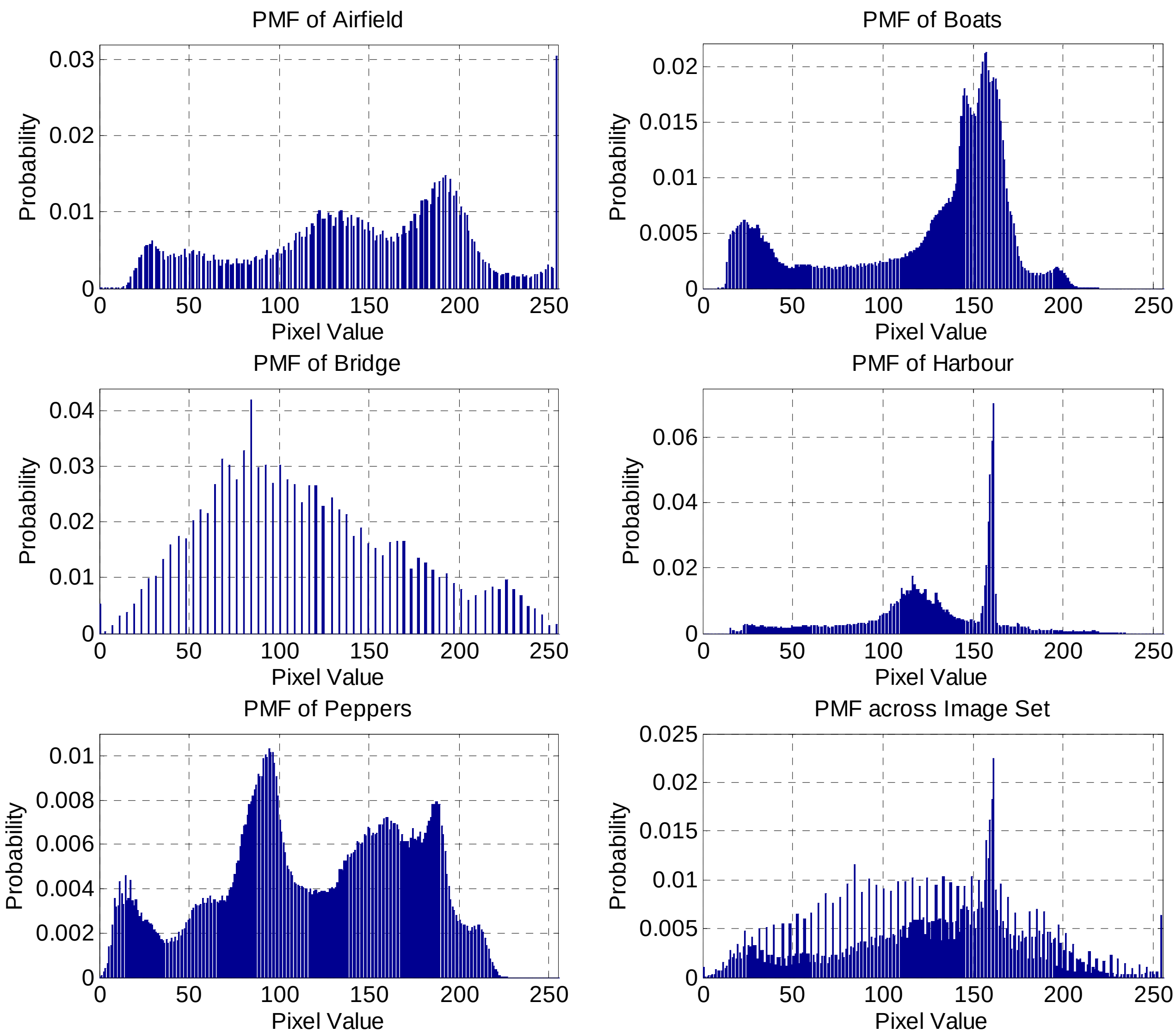
Christopher Tsai
February 11, 2008
EE 398 – Image Compression

Problem Set IV – Predictive Coding

Problem #2 – Lossless Encoding of Gray-Level Images

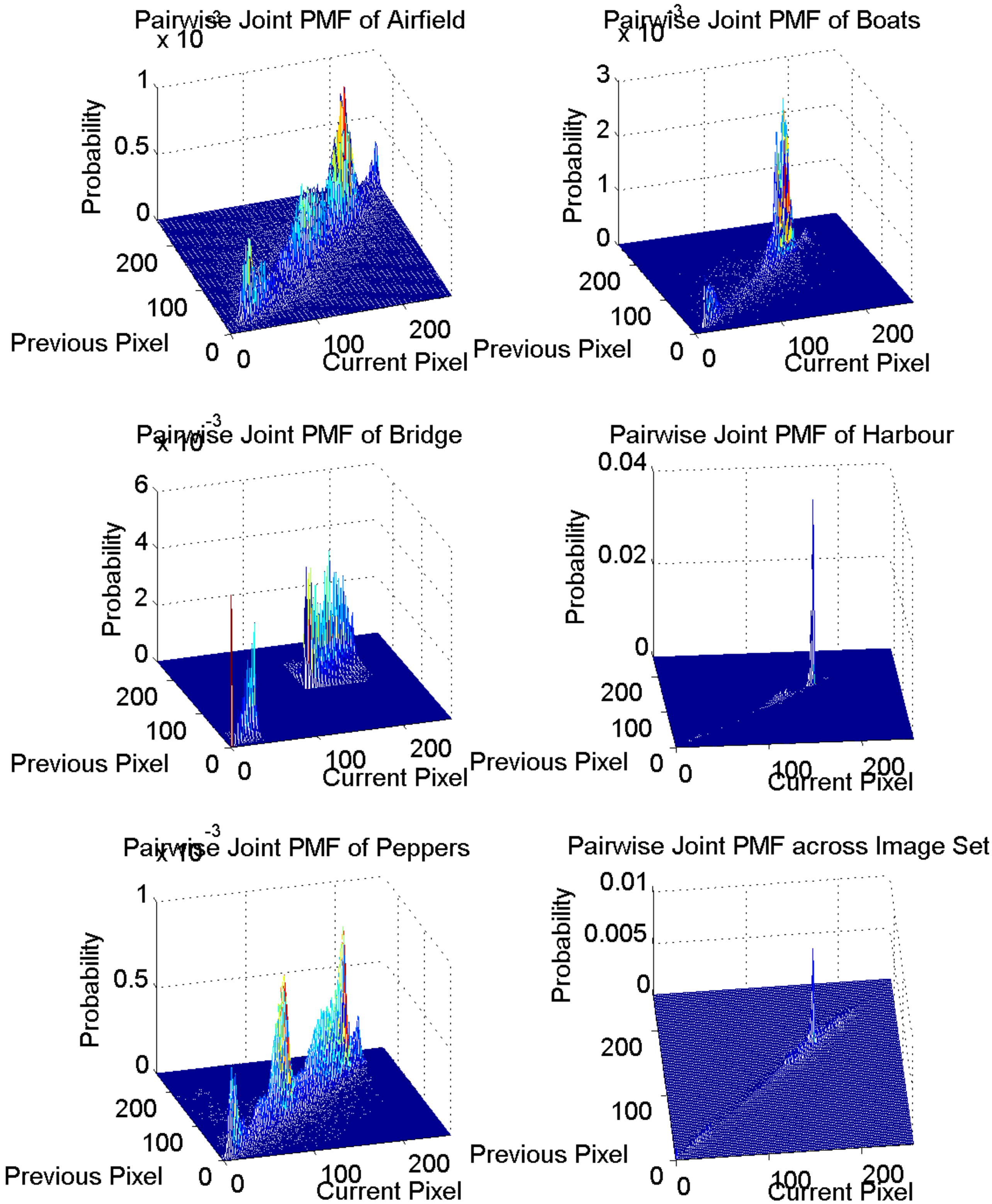
To build our codebook, we first construct the normalized histogram as a probability mass function (PMF) for each one of our five images, as well as their composite sum (added and normalized):



We exploit each statistical distribution by using a variable-length code, with shorter codes representing more probable outcomes, as dictated by the PMF. If we then encode each image using a variable length code table, then we can attain a minimum codeword length. Customizing a set of codewords to fit each separate image's PMF, we obtain a lower entropy than we would achieve by using one single codebook created from the composite image; applying the composite codebook for all images might be computationally convenient, but its distance to each individual image is no longer optimal, therefore increasing the average code length:

[bits/pixel]	Airfield	Boats	Bridge	Harbour	Peppers	IMAGE SUM
Memoryless Custom Code	7.1205	7.0333	5.7056	6.7575	7.5924	7.5129
Memoryless Single Code	7.8362	7.4766	7.1167	7.2392	7.8957	7.5129
Pairwise Custom Code	6.1776	5.8996	4.8098	5.6997	6.2574	6.3179
Pairwise Single Code	6.7996	6.2643	5.8209	6.1100	6.5946	6.3179
Pairwise Custom Code Gain	0.9429	1.1337	0.8957	1.0578	1.3350	1.1950
Pairwise Single Code Gain	1.0365	1.2123	1.2958	1.1292	1.3011	1.1950

Thus, encoding pixels in consideration of their joint statistics with an adjacent pixel improves the minimum length code by approximately 1 bit per pixel for all images. The improvement burgeons when we apply this joint encoding technique to the single code result, which indicates that we gain even more from knowledge of neighboring pixels when we begin with a more crude solution. The only surprise arises from the fact that this joint encoding on a single code solution works so well despite our use of the composite image as the basis! In other words, all images have strong similarities in their joint statistics, allowing us to exploit the neighborhood knowledge even in a composite sum of several images; the strong diagonality in the joint histograms manifest this fact. As a result, the codebook we generalize across several images works even better because these joint statistics are more similar from image to image than their individual pixel-by-pixel counterparts.


Pairwise Joint PMF of Airfield
x 10^-3
Probability
Previous Pixel
Current Pixel
Pairwise Joint PMF of Boats
x 10^-3
Probability
Previous Pixel
Current Pixel
Pairwise Joint PMF of Bridge
x 10^-3
Probability
Previous Pixel
Current Pixel
Pairwise Joint PMF of Harbour
Probability
Previous Pixel
Current Pixel
Pairwise Joint PMF of Peppers
x 10^-3
Probability
Previous Pixel
Current Pixel
Pairwise Joint PMF across Image Set
Probability
Previous Pixel
Current Pixel

Problem #3 – Lossless Predictive Coding of Images

Encoding the prediction *errors* of each of our five images with each of the four prediction schemes, we obtain the following minimum code lengths:

[bits/pixel]	Airfield	Boats	Bridge	Harbour	Peppers	IMAGE SUM	Predictive Gain
Left Neighbor	6.1896	5.1594	5.5403	5.2626	5.2323	5.4769	0.8410
Minimum Variance	6.0145	4.6936	5.9173	5.2795	5.1909	5.4192	0.8987
Minimum Entropy	5.9605	4.7056	5.8777	5.2357	5.1339	5.3827	0.9352
JPEG-LS	5.9905	4.6341	5.3198	5.1271	5.0845	5.2312	1.0867

Thus, predictive coding of the *errors* rather than the image pixels themselves offers us an improvement of approximately 1 bit per pixel no matter the method. In particular, JPEG-LS excels in extracting the most out of prediction, and, to no surprise, since its nonlinear predictor is the most complex of all four methods. The bridge image, with its plethora of edges, responds most favorably to JPEG-LS predictive coding, most likely because other forms of predictive coding – especially the left-neighbor technique – fail for some orientation of bridge edge.