MMAI_HW2

R02944047　朱雅

1. Color Distance

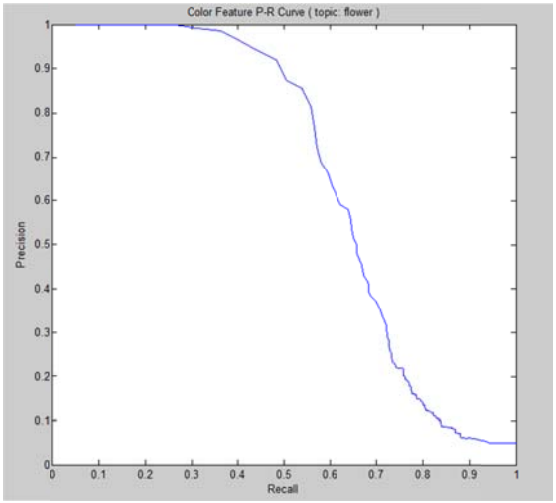
Use region histogram to describe the color feature of the image. The color space is HSV, the histogram has 18H*3S*3V bins for each block, and the image was divided into 4*4 blocks. And each block histogram will be normalized. Then the distance matrix is computed by D1 (i.e. L1 distance) with weighting region histogram:

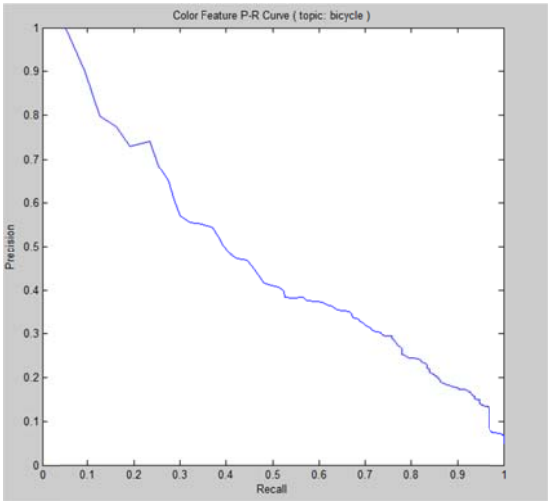
$$d(q,t) = \sum_{m=0}^{M-1}\left|h_q(m) - h_t(m)\right| = \sum_{i=0}^{b-1} w_i \sum_{m=0}^{M-1}\left|h_q^{[i]}(m) - h_{t(m)}^{[i]}\right| ,$$

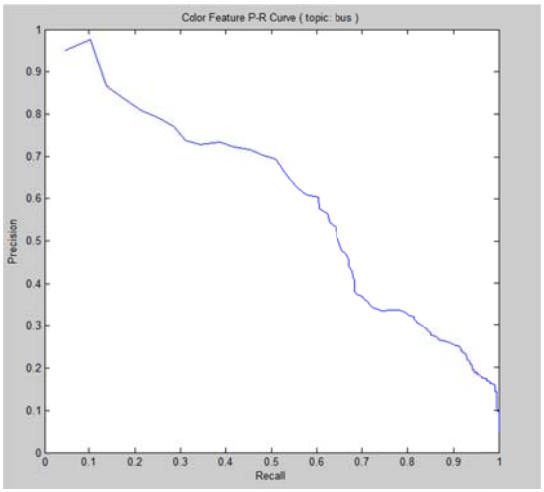
where b is the amount of blocks and w_i is the responding weight of each block. This is the weighting vectors I used for the four topic:

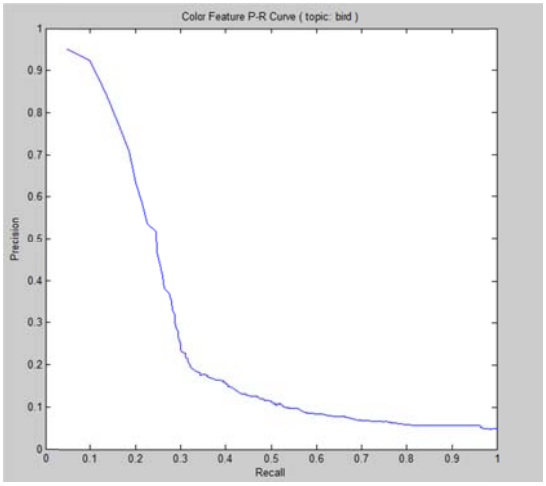
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0.5000	0.5000	0.5000	0.5000	0.5000	1.5000	1.5000	0.5000	0.5000	1.5000	1.5000	0.5000	0.5000	0.5000	0.5000	0.5000
2	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	0.8000	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000	1.5000
3	0.8000	0.8000	0.8000	0.8000	0.8000	1.5000	1.5000	0.8000	0.8000	1.5000	1.5000	0.8000	0.8000	0.8000	0.8000	0.8000
4	0.8000	0.8000	0.8000	0.8000	0.8000	1.3000	1.3000	0.8000	0.8000	1.3000	1.3000	0.8000	0.8000	0.8000	0.8000	0.8000

Then rank the distance matrix to get the most similar images. Here are the P-R curves of the four topics:









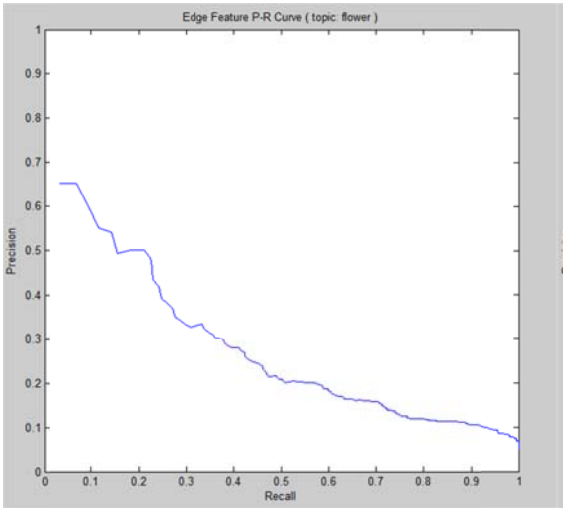
2. Edge Distance

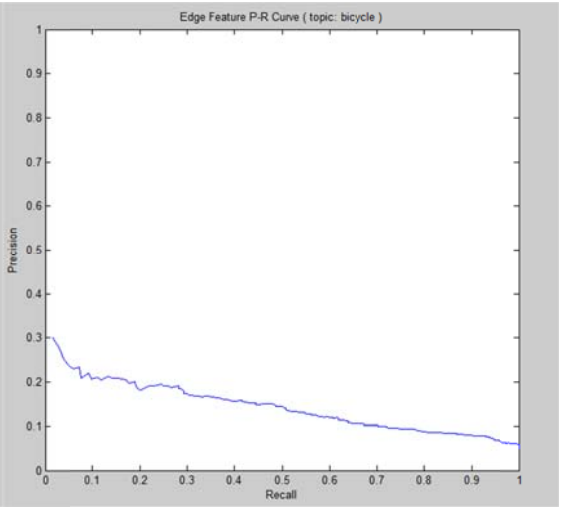
Use PHOG (Pyramid of Histograms of Orientation Gradients) to detect edge feature. The source code is downloaded from:

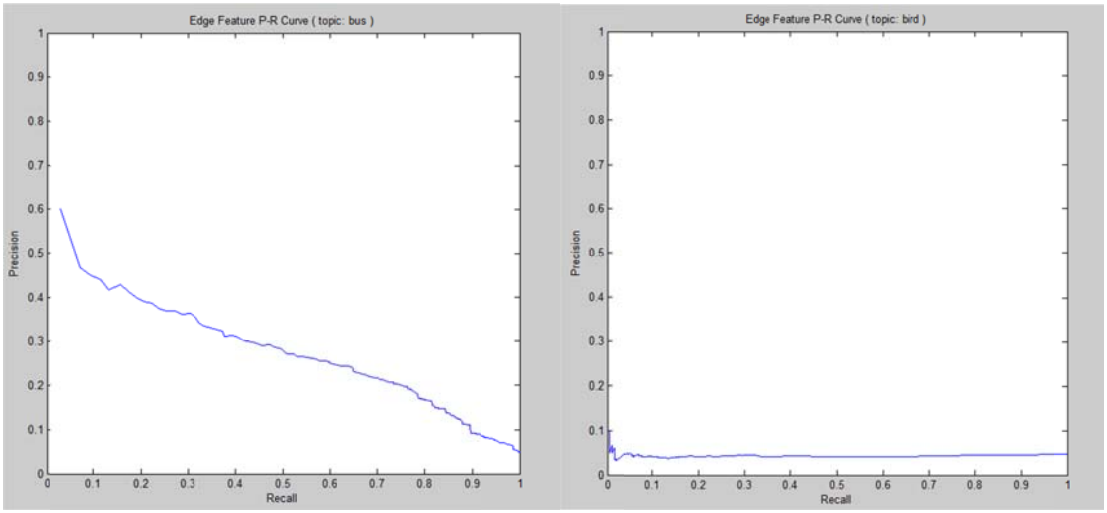
http://www.robots.ox.ac.uk/~vgg/research/caltech/phog.html

This method is little time costing, so I didn't do too much parameter optimization. The following result is experimented by 9 bins, angle=360, L=2.

Still use L1 distance.



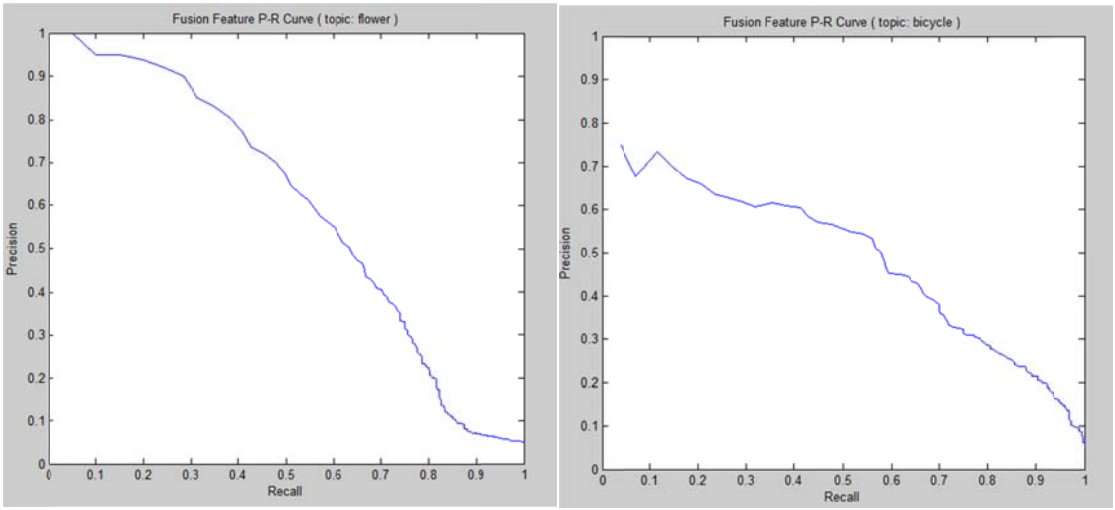


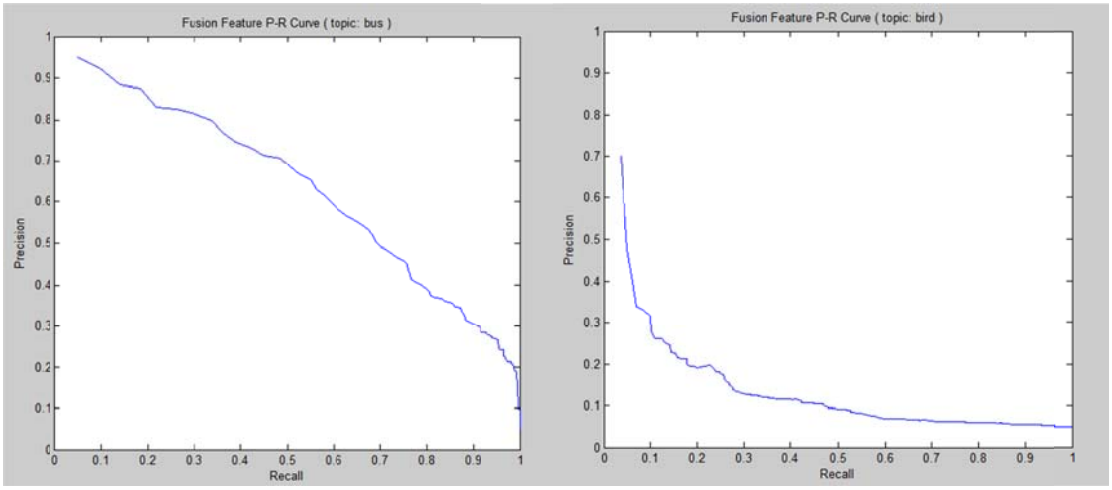


Note that this algorithm fails to do retrieval on topic 2 - 4, because their precisions are lower than 0.5. And it also does poor on topic 1 because its error rate is always greater than 0.3.

3. Fusion

Use ‘Borda Count’ to describe similarity in each feature space, and use weighted sum to calculate the combined similarity. According to the experiments’ results above, color feature describes better than edge feature, so I assigned color feature’s weight=1.5,while edge feature’s weight=0.5. And here are the P-R curves:

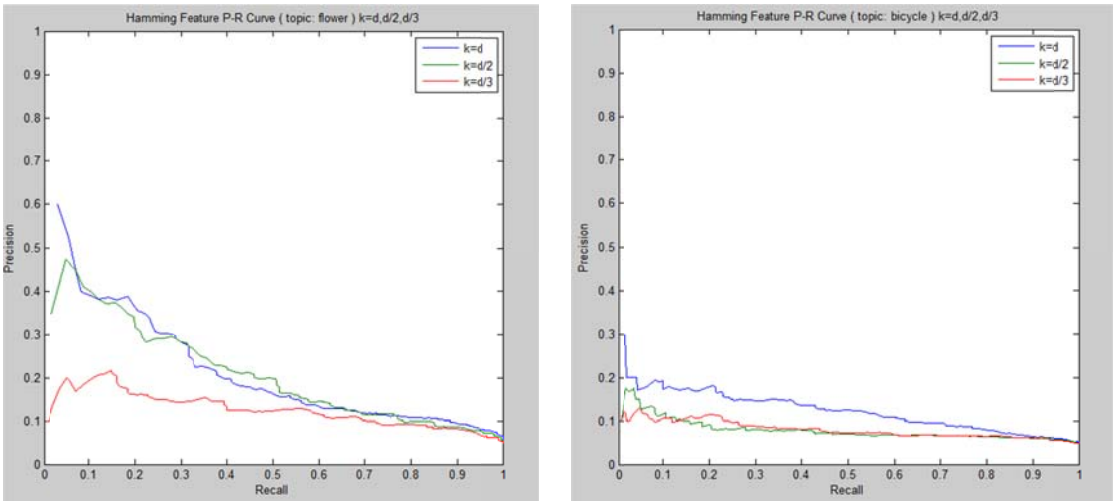


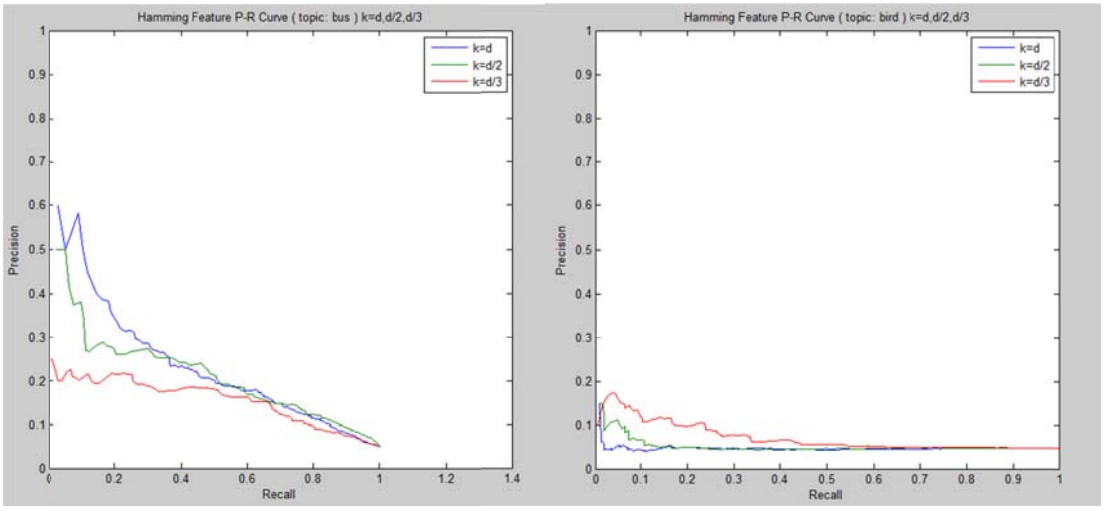


As to 1-3 topics (flower, bicycle, bus), the fusion results are better than both color feature results and edge feature results. But on the last topic, the fusion result is worse than color feature result while better than edge feature result.

4. Hamming distance by random projection

Since region color histogram is not convenient to do binary hashing, so I use edge feature to do binary hashing.





5. Comment

The precision and recall of the result is impacted by the noisy background in the image. The region color histogram can adjust the region weighting to reduce the impact while the PHOG algorithm can't. So the color distance matrix gets better result than the edge distance matrix. But by adjusting the number of bins and the pyramid levels, the result of PHOG algorithm itself can be improved.

And according to the fusion results, the fusion distance of two features can do better than only one feature.

Note that the binary hashing gets worse result than PHOG on edge feature, but it remains similar relative precisions on different topics. Thus proves that this hashing method does make sense on this task, though with lower precision. And according to the result, generally k=d does better than k=d/2, k=d/3, that means the reduction of dimension in this feature matrix probably does no good to the retrieval.

Also note that the 'bird' topic has poor results in both color and edge feature, that is because birds may have quit different colors and their shape vary in different images. So maybe the texture feature is better to describe it since the texture(e.g. feather) of bird will not change too much.