

The impact of errors in Huffman-encoded images

1. Encoded Image

Input: ppm images, raw bitmap of images

Output: encoded ppm images and huffman code

Pipeline: Image can be represented as a matrix of (R, G, B) tuple, and RGB ranges in [0, 255]. Therefore, first get frequency of an array of 256 available, then using huffman coding algorithm, which greedily gets two integers with lowest frequencies and deletes these two corresponding nodes (both integer and frequency representation), afterward inserts an integrated node into tree, and for n-1 (n is 256 here) times, then the huffman tree is available. Traversing the tree to get the code of each leaf node thereafter.

2. Decoded Image

Input: encoded ppm images and huffman code

Output: decoded original image

Using hashmap to store the map of huffman code to integer of pixel of each color, dict in python, it can efficiently decode images. Later show it as a jpeg image.

3. Then change a few values (from 1 byte to 50 bytes) in the encoded files.

After changing a few pixels of ppm image, Matisse-Small.ppm as an example, image looks similar to the original image, but saturation, and some colors are somewhat different. Below are the conducted experiment results,

Change pixel number (R,G,B change together)	Different pixel percent between decoded image and original image
10	0.0167289354821%
20	0.0353166415733%
30	0.0539043476645%
40	0.0724920537556%
50	0.0910797598468%

Table 1 Matisse-Small.ppm changing pixel experiment result

Change pixel number(R,G,B change together)	Different pixel percent between decoded image and original image
10	0.03125%
20	0.065972222222222%
30	0.100694444444444%
40	0.135416666666667%
50	0.17013888888889%

Table 2 bike.ppm changing pixel experiment result

Change pixel number(R,G,B change together)	Different pixel percent between decoded image and original image
10	0.0173183497537%
20	0.0365609605911%
30	0.0558035714286%
40	0.075046182266%
50	0.0942887931034%

Table 3 gogol.ppm changing pixel experiment result

As we can see from the above results, changing some pixels of raw images does not affect the decoded images, and the number of different pixel is comparatively minor. Also, the more pixels have been changed, the percentage difference of pixel between original image and decoded image tends to be higher. We can see that huffman code is a kind of lossless compression, and it can back to original image.