

Jay Whang

Digital Image

He Zhihai

5/11/20

Digital Image Compression (DCT encoding & IDCT decoding)

I used 20 images to test compression programs that use a DCT-based image compression system (from Python3's cv2 library) along with Huffman coding.

These are 3 original images:

`/images/640-jpeg/empress.jpeg`



`/images/png/couchCat.png`



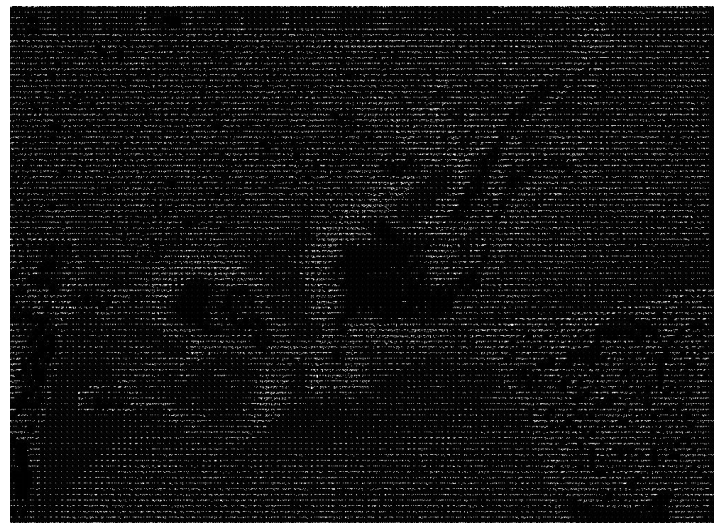
`/images/analog_1.png`



And these are the gray-scaled images in uncompressed bitmap forms (converted to .jpg).



These are each picture's DCT image compression:





And each of them is its decoded form:



PSNR of each images (using cv2's psnr function)

empress.jpeg: 26.93 dB

couchCat.png: 29.1 dB

analog_1.png: 36.3 dB

Compile time

empress.jpeg (640x640): 4.6 seconds for compression, and 2.2 seconds for decoding.

couchCat.png (733x985): 5 seconds for compression, and 2.6 seconds for decoding.

analog_1.jpg (2000x3000): 1 minute and 32.5 seconds for compression, and 16 seconds for decoding.

Conclusion:

As the wavelength of the PSNR is higher, the less the noises the decoded image has. So, the image with higher resolution, even if it took a long time to compress to huffman code and decode, doesn't have much noise than small-size images. There were also some differences when it comes to image noises, when each pixel of grayscale images subtracted with 128 (the half of 255) has a higher wavelength than the one that didn't subtracted with it.