Assignment on ME and Jsteg

Due September 28, 2017

A 2.1 Matrix embedding

You have an 8-bit grayscale image consisting of 102,300 pixels. How many bits can you embed using matrix embedding using binary Hamming codes (as in F5) by making *at most* 100 changes? How many changes will you need to embed 8,000 bits? (Compute both the *expected* number of changes and the number of changes in the *worst case*.)

Recall: All you need to know for your answer is that you can embed p bits in $2^p - 1$ pixels by making at most one change (or, $1 - 2^{-p}$ changes on average). Always start by figuring out the block size (determined by p).

A 2.2 Steganalysis of Jsteg

In this assignment, you will implement and test Zhang's quantitative attack on Jsteg as introduced in class (it also appears in Section 5.1.2 in the textbook). Write a routine that can be run in the following manner:

```
alpha_hat = Jsteg_det(image);
```

where "image" is the name of the JPEG image to be steganalyzed. Download the 'test_images.zip' file from Blackboard. It contains ten grayscale JPEG images, some of which are cover and some are stego images. Run your detection routine on all of them, report the estimated relative message length α in a table, and mark those you believe are stego images. Also, submit your Matlab code.

For your convenience, I am supplying you with an embedding routine of Jsteg, 'Jsteg_embed.m', and the extraction routine 'Jsteg_read.m'. Use the embedding routine to test your detection before applying it on those ten test images! Embed some payload, note the relative message length α, and then run your routine on the stego image. If your routine is working properly, you should see an approximate match between the true α and the estimated one.

Things to think about – you do not need to comment on them in your assignment, but think about them: Besides your *subjective judgment* about which images are cover and which are stego, think about a more rigorous procedure to arrive at the decision. What do you think is the importance of the response of your steganalyzer on cover images? Do you think the error distribution of your estimate is Gaussian? How will the shape of the distribution affect your decision making?

Optional problem O.2

Derive the impact of embedding with OutGuess on normalized[1] histograms of individual DCT modes. Assume that a pseudo-random path is used for both embedding the message and for correcting the global histogram. Remember, OutGuess preserves the histogram of all DCT coefficients, $h_0[i] = h_\alpha[i]$, for all i and for all correctable α, but in general it will not preserve the histograms of individual DCT modes because they will have different shapes than the global histogram. Denote $h_0^{kl}[i]$ and $h_\alpha^{kl}[i]$, respectively, the normalized histogram corresponding to DCT mode (k, l), $k, l = 0, \ldots, 7$, before and after embedding with OutGuess, and find the relationship between them as a function of α. This relationship will involve the global histogram as well. As a sanity check, make sure that your formula preserves the kl-th histogram, $h_0^{kl}[i] = h_\alpha^{kl}[i]$, when $h_0^{kl}[i] = h_0[i]$.

Each year, I face new challenges with the JPEG Toolbox as students work on various platforms. If you are experiencing problems with the JPEG Toolbox, here are some tips that will likely help you solve any issues:

1) Run Matlab as administrator (right-click on the Matlab icon on your desktop and choose "Run as administrator" instead of the default "Open")

If this does not help, try:

2) Download Toolbox from our DDE web site and use this code instead. http://dde.binghamton.edu/download/jpeg_toolbox.zip

3) The message you should be embedding using Jsteg-embed has to be a bit-stream, e.g., msg = round(rand(1,10000)); Do not put there a name of a file. The embedding routine does not work this way.

4) If your code does not estimate the message length, check your indices. Most likely you did not implement the Delta variable correctly. Know where your h(0) histogram bin is. Check Delta for bins [-2,-1] and [2 3]. Does it do what it should for these two bins?

5) If you get an error regarding Huffman tables, add this to your code

```
im.ac_huff_tables = [];
im.dc_huff_tables = [];
```

before executing jpeg_write(my_stego_structure, my_stego_file_name);

---

[1] Normalized histogram is scaled so that $\sum_i h[i] = 1$. It is essentially a sample probability mass function (pmf).