MATH 535 Homework 1

Corbin Graham February 10, 2023

1. Problem 1

   (a) Implementation of the algorithm

   The method I created to implement the algorithm uses the LSB replacement technique. It first embeds the size of the message, K, in the first t bits. K is calculated from the length of the binary string and t is calculated from $\lceil \log_2 MN \rceil$. Because t will remain static after changing the image, anyone who uses the same operation to retrieve the message bits from the image. The message bits are simply encoded to the last bit value of each color value ($0 \rightarrow 255$).

   (b) Color image used

   The color image I chose was the first image that was in the list of cover images to be used for the second part. I could not find a better, or easier, image to use. When in Google Colab, the image is reset every time the browser was refreshed so I wanted it to be something I could find easily.

   (c) Solution to the issue $0.1 * M * N$ is not an integer

   The solution, using python, was rather simple. I simply casted the decimal type to integer type using int().

   (d) Problems

   This assignment was plagued with major issues that caused many setbacks. Most were simply due to the language have strange syntactical issues throughout. The issues almost all came from getting and assigning bit values from the image. Since I am new to the library, I was unsure of what functions it had to aid my ability to make those changes. Luckily, I fiddled with it enough until I found working solutions, even if those aren't necessarily the most correct solutions. I had first misunderstood the algorithm and how the bits were embedded in the image, and reviewed. After re-watching all of the lecture videos, I had realized I had completely implemented the algorithm wrong and would have to restart. I think the most important point coming from this would be to test the algorithm before beginning to implement it.

   (e) Errors

   The code was thoroughly tested and debugged for errors. Thus, no errors should arise during review.

   (f) Important Points

   The most important point that I derived from the problem was the flexibility of the algorithm. And the insecurity of basic bit replacement. Even a more advanced algorithm with the 'coin-flip' would do little to hide the message once its size has been realized.

2. Problem 2

   (a) Implementation

   I designed the first part of the assignment in a way that I could simply add a helper-function (embed-multiple()) that would allow me to embed multiple images with the same message in order. There was little that I had to change or add for this to work.

   (b) Change Rates

   It appeared that the change rate was always roughly half that of the embedding rate. The change rate stayed consistent with this measurement for the different embedding rates.

3. Problem 3

   Let $A, B \in F^x$ be F-valued images on X, where F is a value set and X is a point set.
   For purposes here, you may assume that F is the 8-bit integer set $\{0, 1, ..., 255\}$ and X is an MxN array. Thus,

$$A = \{((i,j), a(i,j) : 1 \leq i \leq M, 1 \leq j \leq N, a(i,j) \in F\}$$

   and

$$B = \{((i,j), b(i,j) : 1 \leq i \leq M, 1 \leq j \leq N, b(i,j) \in F\}$$

   The image A is the cover image and the image B is a corresponding stego image created by the embedding algorithm which is equivalent to the following procedure:
   Let a random variable (r.v.) H take on values from the real numbers $\mathbb{R}$, with resulting probability density function $f(x)$. Create a message string by generating MN random values:

$$\{n(i,j) : 1 \leq i \leq M, 1 \leq j \leq N, n(i,j) \in \mathbb{R}\}$$

   where at location $(i,j), n(i,j)$ is a realization of H.

   Thus, for the cover image A, and message value represented by an iid realization $n(i,j)$, the stego image B has value at (i,j) of

$$b(i,j) = round(a(i,j) + n(i,j))$$

   Let

$$p(k) = p(n(i,j) = k) = \int_{k-1/2}^{k+1/2} (f(x)dx)$$

   where $f(x)$ is the probability density function for the r.v H. Derive the expected value of the embedding distortions given in (I) and (II), for $m = 1, m = 2$. Where

$$d_m(A,B) = \sum_{(i,j)} (|a(i,j) - b(i,j)|^m)$$

   (a) d_1

$$E[d_1(A,B)] = \sum_{k=-\infty}^{\infty} kp(k)$$

$$
\begin{aligned}
d_1(A,B) &= 1/MN \times \sum_{(i,j)}^{MN}(|a(i,j)-b(i,j)|) \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(|a(i,j)-b(i,j)|)) \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(|n(i,j)|)) \\
E[d_1(A,B)] &= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(E[|n(i,j)|])) \\
\text{Assume:} f(x) &= 1/255 : 0 \leq x \leq 255 \\
E[n(i,j))] &= \int_{-\infty}^{\infty} |x| f(x) dx \\
&= \int_{0}^{255} x \frac{1}{255} dx \\
&= \frac{1}{255} \int_{0}^{255} x dx \\
&= \frac{1}{255} * \frac{255^2}{2} \\
&= \frac{255}{2} \\
E[d_1(A,B)] &= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(\int_{-\infty}^{\infty} |x| f(x) dx)) \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(255/2))
\end{aligned}
$$

(b) d_2

$$
E[d_2(A,B)] = \sum_{k=-\infty}^{\infty} k^2 p(k)
$$

$$
\begin{aligned}
d_1(A,B) &= 1/MN \times \sum_{(i,j)}^{MN}(a(i,j)-b(i,j))^2 \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(a(i,j)-b(i,j)))^2 \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(n(i,j)^2)) \\
E[d_1(A,B)] &= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(E[n(i,j)^2])) \\
\text{Assume:} f(x) &= 1/255 : 0 \leq x \leq 255 \\
E[n(i,j)] &= \int_{-\infty}^{\infty} |x| f(x) dx \\
&= \int_{0}^{255} x \frac{1}{255} dx \\
&= \frac{1}{255} \int_{0}^{255} x dx \\
&= \frac{1}{255} * \frac{255^2}{2} \\
&= \frac{255}{2} \\
\text{Then:} E[n(i,j)^2] &= \frac{255}{2}^2 \\
E[d_1(A,B)] &= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(\int_{-\infty}^{\infty} |x| f(x) dx)) \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}((255/2)^2)) \\
&= 1/MN \times \sum_{i=1}^{M}(\sum_{j=1}^{N}(16256.25))
\end{aligned}
$$