Week 13 Exercises (ECE 598 DA)

Exercise (Watermark detection as hypothesis testing): Consider a spread-spectrum watermarking scheme where a known pseudorandom sequence W of length N (with values ± 1) is added to an image (modeled as an N-dimensional vector) to embed a single bit. The detector uses correlation $Z = \frac{1}{N}\langle Y', W\rangle$ as a test statistic on a possibly watermarked image Y'. Model the attack as additive noise: $Y' = Y + N$, where Y is the original watermarked image and N is i.i.d. Gaussian noise $\mathcal{N}(0, \sigma_n^2)$. Assume W is also treated as random with i.i.d. entries ± 1. Formulate the detection problem as distinguishing H_0 (no watermark present) vs H_1 (watermark present). Derive the expected value and variance of the test statistic Z under each hypothesis. What is the form of the optimal decision rule for a given false-alarm rate, and how does the detection threshold relate to σ_n and the watermark strength α?

Exercise (Steganographic capacity estimation): Suppose we have a simplistic model for images where each pixel's noise (or tolerance for change without detection) can be approximated as $\pm\Delta$ (i.e., we can adjust a pixel up or down by at most Δ without noticeable or detectable effect). If an image has $M \times N$ pixels, what is the rough upper bound on the number of bits we could embed (perfectly robustly and imperceptibly) in that image? Now refine the estimate by considering that changes in one pixel can be detected by comparing with neighbors (so not all pixels can be independently at their $\pm\Delta$ extremes). Why is the independent-pixel assumption too optimistic for steganography?

Exercise (QIM vs. Spread Spectrum trade-offs): An image watermarking scheme is implemented in the DCT domain of 8×8 blocks. Scheme A uses spread-spectrum: it adds a small Gaussian noise pattern (with standard deviation σ_w) to all 64 DCT coefficients (except the DC coefficient) of each block to encode one bit per block (presence/absence detected by correlation). Scheme B uses QIM: it takes the largest mid-frequency coefficient in each block and quantizes it to one of two levels depending on the bit to embed (thus also 1 bit per block).

- Which scheme would you expect to be more robust against a mild Gaussian noise attack on the image? Why?
- Which scheme would you expect to have less impact on image quality for the same payload, and why?
- How could scheme B be made more robust without changing the payload, and what is the potential cost of that modification?

Exercise (Deep learning watermarking thought experiment): Imagine a neural network is trained to embed a watermark into images such that it can be decoded after the images are passed through a social media platform that compresses them to JPEG quality 70. Describe how you would set up the training process for such a network (what would be the loss functions, what training data you would need, etc.). What advantages might this learned approach have over a fixed algorithm like spread-spectrum in this specific scenario? What might be some disadvantages or concerns with using a neural network approach here?