SS- Spread Spectrum

Recap

Last time we have seen LSB embedding, detection and attack.

LSB is an extremely trivial technique, so this time we will use Spread Spectrum watermarking.

Spread Spectrum watermarking: an independent and identically distributed (i.i.d) Gaussian random vector (the watermark) is **imperceptibly** inserted in a **spread-spectrum-like** fashion into the perceptually most significant spectral components of the data.

SS - Embedding

Modify **CoxEmbedding.m** to embed a watermark using Spread Spectrum

Compute DCT → Coefficients selection → Embed watermark → Compute IDCT

Generate watermark → Embed watermark

Hint: sort the coefficients (sign is irrelevant)

Hint: use multiplicative SS: $v_i' = v_i(1 + \alpha x_i)$

SS - Embedding

Exercises:

- Change the value of alpha
- Change the size of the watermark
- Compute the PSNR and WPSNR of watermarked images for 3 different values of alpha in the multiplicative version of the SS

SS - Detection

Modify **CoxDetection.m** to extract the watermark using Spread Spectrum.
Hint: it is similar to the embedding

Hint: $sim(x, x') = \frac{x' \cdot x}{\sqrt{x' \cdot x'}}$

Compute DCT → Coefficients selection → Extract watermark → SIM ← Extract watermark

SIM → Found/Not found

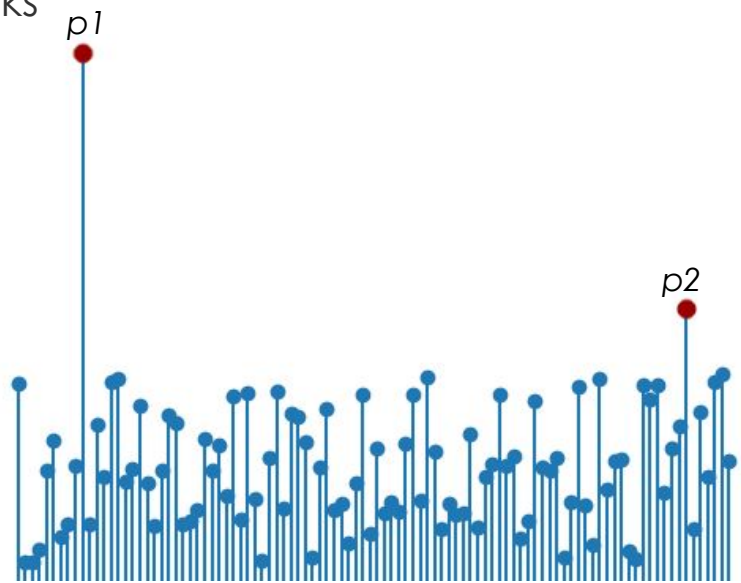
Hint: $x'_i = \frac{v'_i - v_i}{\alpha v_i}$

SS - Similarity threshold

We need to use a threshold to see if the watermark is present.
The threshold can be computed as follows:

- Generate other 999 random watermarks
- Present the 999 + your watermark to your detection system, and consider the 2nd highest correlation peak
- Compute the threshold as

$$T = p2 + 0.10 \cdot p2$$



SS - Detection

Try to modify CoxDetection.m to detect the watermark using Spread Spectrum.

Exercises:

- Can you still detect the watermark if the image is attacked? Use the list of attacks described in LAB2.
- Play with the complete SS code (embedding + detection) and see what happens if you select different coefficients (e.g. low to mid frequencies) or if you embed the watermark only on some areas of the original image (thus, selecting areas in the spatial domain and exploiting block DCT instead of full frame DCT to perform the embedding).