CÓDIGOS Y CRIPTOGRAFÍA

Esteganografía

INDEX

Método LSB

- Objetivo: Ocultar un mensaje en una imagen.
- Elementos:
  - Mensaje a ocultar, el cual escribiremos en bits.
  - Imagen, que servirá de cobertura.
- Idea: Utilizar el bit menos significativo para ocultar el mensaje en la imagen.

Método LSB

- Mensaje: "Hello world".
- Pasamos el mensaje a ASCII:
  72 101 108 108 111 32 119 111 114 108 100
- Y después a bits:
  01001000 01100101 01101100 01101100 01101111 00100000 01110111 01101111 01110010 01101100 01100100

Método LSB

- Por otro lado, necesitamos una imagen. En este caso, en escala de grises:

- Cada pixel debe corresponderse con un valor entre 0 (completamente negro) y 255 (completamente blanco).

Método LSB

- ¿Cómo ocultamos el mensaje? ¡De la forma más obvia!
- Supongamos que los primeros píxeles tienen la forma:
  00010001 01001101 01110111 10101101 ...
- Sustituimos el bit menos significativo por el correspondiente bit del mensaje. Como $H = 01001000$:
  00010000 01001101 01110110 10101100 ...
- La imagen queda prácticamente inalterada.

Método LSB

Imagen original

Imagen con mensaje

- El receptor, que sabe que hay un mensaje oculto, debe pasar la imagen a bits y tomar los dígitos correspondientes para recuperar el mensaje.
- Es posible aumentar la complejidad (salto de bits “acordados”, usar imágenes con color, etc).

INDEX

Desordenando una imagen

- Objetivo: Modificar una imagen para hacerla irreconocible.
- Elementos:
  - Una imagen cuadrada.
  - Un método para desordenar los píxeles que pueda deshacerse.
- Idea: Utilizar una matriz $2x2$ para mover la posición de los píxeles.

Desordenando una imagen

- Volvemos a tomar nuestra imagen anterior (291×291 píxeles):

- Como sabemos, cada píxel de la imagen se determina por tres parámetros: dos coordenadas (posición) y un número adicional que indica la "intensidad" de gris. Pixel=(100,120,200).

Desordenando una imagen

- Consideramos una matriz $A \in M_2(?)$ que nos permita desordenar los píxeles: Por ejemplo: $A = \begin{pmatrix} 1 & 5 \\ 2 & 3 \end{pmatrix}$
- Cada posición del píxel la modificamos de acuerdo a la matriz considerada:

$$A \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 1 & 5 \\ 2 & 3 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 17 \\ 13 \end{pmatrix}$$

  Es decir, si tenemos $(2, 3, 120) \to (17, 13, 120)$.

- Cuidado: Si intentamos hacer lo mismo con $(90, 40, 20) \to (290, 300, 20)$, que no estaría en la imagen.
- Nos tocará trabajar en $\mathbb{Z}_{291}$.

Imagen original

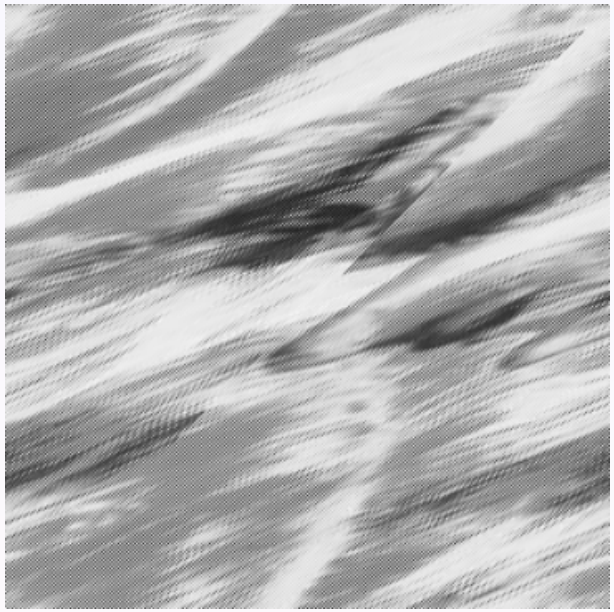

Imagen desordenada

Desordenando una imagen

- El proceso anterior nos devolverá una imagen con los píxeles desordenados, e inicialmente irreconocible.
- Para recuperar la imagen anterior, nos hará falta la matriz A. Además, dicha matriz deberá ser **invertible** en $\mathbb{Z}_{291}$ para poder asegurar que el proceso es revertible.
- Observamos que en $M_2(\mathbb{Z}_{291})$ solo hay 291^4 elementos, por lo que necesariamente si A es invertible, $A^n = I$ para un cierto n.
- Podemos aumentar complejidad incluyendo imágenes a color, matrices adicionales por cada canal de color, etc.

INDEX

Reparto de secretos

- Objetivo: Repartir una imagen entre un número n de miembros, de modo que ninguno tenga la imagen original, y solo puedan reconstruirla si se alcanza el quorum necesario.

- Elementos:
  - Una imagen a repartir.
  - Un número de individuos n.
  - Un mínimo quorum $m \leq n$.

- Idea: Utilizar polinomios interpoladores para obtener los datos de la imagen.

Reparto de secretos: Código (Shamir)

- Primer caso: Supongamos que queremos repartir un código numérico S.
- Construimos un polinomio $p(x)$ de grado $m-1$ de modo que $p(0) = S$.
- Repartimos $c_i = (x_i, p(x_i))$ con $i = 1, \ldots, n$ entre los usuarios, donde cada $x_i \in \mathbb{R} \setminus \{0\}, i = 1, \ldots, n$.

  Cualquier subconjunto de m elementos de $\{c_i\}_{i=1}^{n}$ permite obtener el polinomio $p(x)$ y recuperar la clave.
- Usamos el polinomio interpolador de Lagrange (o el de Newton).

Reparto de secretos: Imágenes (Shamir)

- Consideremos una imagen en blanco y negro (si bien el procedimiento puede darse con imágenes a color igualmente).
- De nuevo el procedimiento se basa en utilizar la reconstrucción de polinomios para el reparto del secreto.
- Sin embargo, la idea varía en varios puntos a lo anterior:
  - Trabajaremos en $\mathbb{Z}_{250}$, ya que 251 es el primo más cercano a 255 (truncamiento).
  - En lugar de trabajar pixel a pixel, se trabaja en bloques de píxeles, los cuales se utilizan para construir la matriz.
  - Para cada píxel podremos considerar un polinomio distinto.

Reparto de secretos: Imágenes (Shamir)

Creación de las "Sombras":

- Hacemos un búcle que pase por cada uno de los píxeles, y para el píxel (i, j) tomamos el valor $s^{i,j}$ (truncamos a 250 si es necesario).
- Construimos un polinomio aleatorio:

$$p^{i,j}(x) \equiv s^{i,j} + a_1^{i,j}x + \cdots + a_{m-1}^{i,j}x^{m-1} \pmod{250}.$$

- Tomamos $p^{i,j}(1), \ldots, p^{i,j}(n)$.
- Construimos la imagen $Sombra_k$ donde el píxel (i, j) es $p^{i,j}(k)$, con $k = 1, \ldots, n$.

Recuperando la imagen:

- Dadas k sombras, mediante Lagrange recuperamos $p^{i,j}(x)$ y podemos calcular $p^{i,j}(0)$.

Reparto de secretos: Thien and Lin

El método anterior puede mejorarse de diversas formas, por ejemplo la siguiente propuesta por Thien and Lin:

- Primero, la imagen se desordena (podemos usar lo visto en la sección anterior).
- Segundo, en lugar de trabajar píxel a píxel, se trabaja en bloques de píxeles. Concretamente, supongamos que los primeros m píxeles son $(a_0, a_1, \ldots, a_{m-1})$.
- Construimos el polinomio:

$$q(x) \equiv a_0 + a_1 x + \cdots + a_{m-1} x^{m-1} \pmod{250}$$

- Se calcula $q(1), \ldots, q(n)$ y el primer píxel de $Sombra_k$ es $q(k)$.
- Se repite el proceso con el siguiente bloque.

Reparto de secretos: Thien and Lin

Varias cuestiones a tener en cuenta:

- Ahora, al recuperar el polinomio, recuperamos un bloque de píxeles.
- Esto hace que las "Sombras" sean de menor tamaño que la imagen original.
- Se "desordena" la imagen original para asegurar que la "sombra" no sea reconocible. Esto puede hacerse también en el esquema de Shamir.
- Podemos aumentar la complejidad viendo cómo repartir la forma de desordenar la foto.