

Ocultamiento de Información en Imágenes RGB usando LSB

Dulce Rosario Pérez Hernández¹

Sergio Adán Juárez²

^{1,2} Benemérita Universidad Autónoma de Puebla

^{1,2} Facultad de Ciencias Físico Matemáticas

dulce.perezhe@alumno.buap.mx¹, sergio.adan@correo.buap.mx²

Resumen

Imagina poder esconder mensajes secretos, totalmente invisible en las fotos que vemos todos los días en internet.

Esto es precisamente lo que estudia la esteganografía, una técnica que consiste en ocultar información dentro de otros archivos de manera que su presencia pase desapercibida. A diferencia de la criptografía: Que oculta el contenido pero no su existencia, la esteganografía busca que nadie sospeche siquiera que hay un mensaje secreto [2].

Introducción y objetivo

La esteganografía tiene como objetivo principal ocultar información de manera que su existencia no sea detectada. Se busca modificar el medio digital (imagen, audio, video) de forma que los cambios pasen desapercibidos y sean difíciles de predecir o modelar estadísticamente por un estegoanalista [1].

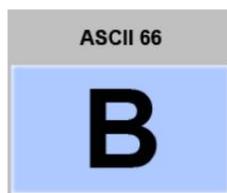
El principio LSB (Bit Menos Significativo) se basa en realizar la mínima modificación posible sobre un byte, que es una alteración en una unidad, implicando cambiar solo el bit menos significativo. Los medios digitales difíciles de modelar estadísticamente, como imágenes, audio y video, son especialmente aptos para esconder información [3].

Información en el Bit Menos Significativo (LSB)

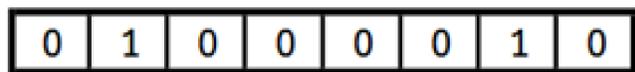
La esteganografía busca ocultar información sin ser detectada, modificando lo mínimo posible el medio. Para evitar ser descubiertos, se eligen datos difíciles de predecir estadísticamente, como imágenes, audio o video. En estos casos, la alteración más sutil y común es cambiar solo el bit menos significativo de un byte [4].

3.1. Ejemplo

Supongamos que queremos ocultar la letra B en una imagen,



su valor en la tabla ASCII es 66, al convertirlo a su representación binaria nos quedaría el siguiente byte (octeto):



Este valor binario sería el que ocultaríamos en la imagen. Por otro lado, sabemos que cada píxel de una imagen RGB a 24 bits está conformado por tres valores numéricos que representan a cada color y que este valor está en el rango de [0, 255]. Por lo tanto se necesitan 3 píxeles por cada letra de un mensaje que se desee ocultar.

Técnicas de Incrustación LSB

Existen dos técnicas habituales para incrustar información en el LSB de un byte:

4.1. LSB Replacement (Sustitución LSB):

Consiste en sustituir directamente el LSB del byte por el bit del mensaje que se quiere ocultar. Ejemplo: Para incrustar un '1' en un byte con valor 160 (10100000), se sustituye el LSB por '1', resultando en 161 (10100001).

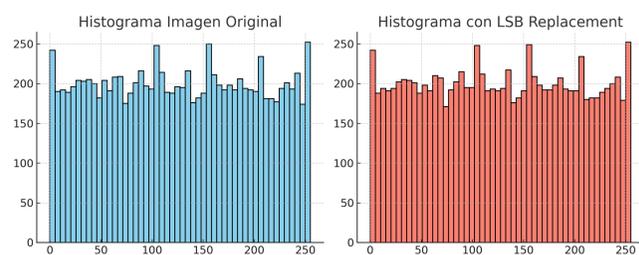


Figura 1: La siguiente gráfica corresponde a una imagen que no ha sido alterada usando LSB replacement.

4.2. LSB Matching (Incrustación ± 1):

Se modifica el LSB sumando 1 o restando 1 aleatoriamente al valor del byte, solo si el LSB actual no coincide con el bit del mensaje deseado. Ejemplo: Si un byte es 160 (10100000) y quieres incrustar un '1', se le suma 1 para que el LSB cambie a '1' (10100001). Es crucial controlar que la suma o resta no genere valores fuera del rango (0-255 para bytes), es decir, no restar de 0 ni sumar a 255.



Figura 2: Imagen a la que se le ha incrustado un mensaje usando LSB replacement



Imagen transmitida Imagen oculta

Figura 3: Imagen a la que se le ha incrustado una imagen usando LSB replacement

Extracción de la Información Oculta

El proceso de extracción es idéntico para ambas técnicas (LSB replacement y LSB matching). Se lee el LSB de cada byte en las posiciones donde se espera el mensaje. Los bits extraídos se agrupan de 8 en 8 para reconstruir los bytes del mensaje original. Si se extraen más bits de los que forman el mensaje, aparecerán caracteres extraños. Una herramienta de esteganografía puede indicar la longitud del mensaje en una cabecera oculta para evitar esto.

Esteganografía LSB en:

Medios Específicos:

- Imágenes Sin Comprimir (Mapas de Bits): Representadas por matrices de píxeles
- Escala de grises: un byte por píxel (0-255)
- Color (RGB/RGBA): tres o cuatro bytes por píxel (Rojo, Verde, Azul, Transparencia)
- Implementación: Se leen los valores de los píxeles (o canales) y se modifican sus LSBs
- Herramientas Python: La librería imageio es útil para leer y guardar imágenes.

Conclusiones

LSB matching es la técnica preferida sobre LSB replacement por su mayor seguridad ante ataques estadísticos.

La capacidad de incrustación debe ser baja para mantener la indetectabilidad del mensaje.

Referencias

- [1] Jessica Fridrich. *Steganography in digital media: principles, algorithms, and applications*. Cambridge university press, 2009.
- [2] Neil F Johnson and Sushil Jajodia. Exploring steganography: Seeing the unseen. *computer*, 31(2):26-34, 2008.
- [3] Daniel Lerch. Esteganografía lsb, 2018. Accedido: 2025-08-18.
- [4] Jarno Mielikainen. Lsb matching revisited. *IEEE signal processing letters*, 13(5):285-287, 2006.