

Ocultamiento de información en imágenes

Balocco Hernán, López Emiliano

Captura y Procesamiento Digital de Señales e Imágenes – Trabajo práctico final
Ingeniería en Informática – FICH

Resumen: El objetivo de este trabajo es investigar una determinada técnica de ocultamiento de información en imágenes. Esta consiste en la inserción de texto o archivos en los bits menos significativos de una imagen de Mapa de Bits, dejando la imagen original prácticamente imperceptible al ojo humano. La simpleza del método tienen la desventaja que es susceptible a compresiones con pérdidas, ruido, ecualizaciones, o cualquier posible variación de la imagen. Se presentarán también, posibles mejoras aplicables a este método.

I – INTRODUCCION

La esteganografía es una rama de la criptografía que trata sobre la ocultación de mensajes, para evitar que se perciba la existencia del mismo. El nombre “Steganographia” proviene del griego “escritura oculta”, (steganos = secreto, grafía = escritura). Es el arte y la ciencia de escribir mensajes secretos de tal forma que nadie fuera de quien lo envía y quien lo reciba sabe de su existencia; en contraste con la criptografía en donde la existencia del mensaje es clara pero codificada.

La esteganografía utilizada en imágenes tiene como objetivo la inclusión de datos dentro de estas. La imagen resultante debe cumplir entre otras con las siguientes características:

Invisible al nivel perceptivo: la calidad de la imagen no debe presentar mermas de calidad, teniendo en cuenta las características del sistema visual humano.

Invisible a nivel estadístico: Si todas las imágenes que poseen información oculta presentan una misma característica común, resultará sencillo detectar la información si se dispone de un número considerable de imágenes.

En el presente trabajo nos hemos centrado en la primera de estas dos características. Esto se ha logrado modificando el bit menos significativo (LSB) en imágenes de 24 bits de profundidad.

II – METODOS UTILIZADOS.

Una manera sencilla de enviar un mensaje sin que este sea descubierto consiste en escribir dicho mensaje en una imagen de Mapa de Bits.

Este método sencillo que analizaremos para ocultar información en imágenes, consiste en sustituir el bit menos significativo de una determinada imagen por la información que se desea esconder. Para ilustrar este funcionamiento visualice la figura 1.1. Se trata de una imagen en formato BMP de 24 bits, lo que equivale a 256 tonos para cada componente RGB. La información a ocultar únicamente requiere un bit por píxel.

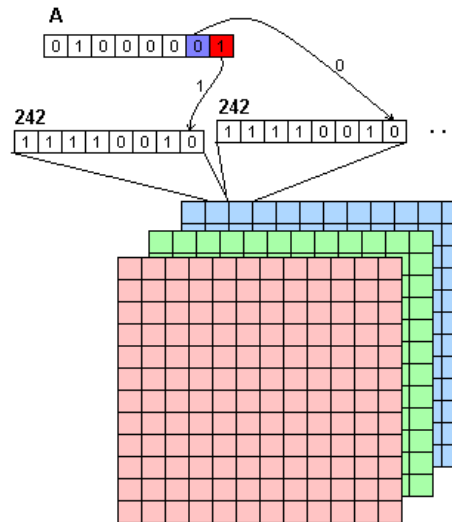


Figura 1.1

Sin embargo, los 24 bits que integran la imagen original no son todos igualmente importantes, sino que el bit menos significativo (LSB) de cada componente RGB aporta menos información a la imagen y, por tanto, su eliminación o sustitución no degradará excesivamente el resultado final. En nuestra implementación hemos modificado solamente el bit menos significativo de la componente Blue.

Los pasos necesarios para implementar esta técnica pueden resumirse de la siguiente manera:

- § Calcular la cantidad de caracteres que posee el texto a ocultar. Este valor es guardado en una posición predeterminada de la imagen, con el fin de saber de manera exacta la cantidad de valores a recuperar en el proceso de recuperación de los datos.
- § Si la información a ocultar es texto, debemos recorrer dicho texto, obteniendo los ocho bits (un byte) de cada carácter. Al mismo tiempo se lee el valor de cada componente RGB del píxel analizado, y se modifica solamente el bit menos significativo de la componente Blue. De esta manera se logra guardar un carácter por cada ocho píxeles de la imagen.
- § Si la información a ocultar es un archivo, se leerá byte a byte, hasta que finalice el mismo, guardando la información de manera similar a la anterior. Una diferencia en este caso es que también debemos guardar el nombre del archivo con su respectiva extensión en algún lugar de la imagen.
- § Para la extracción de los datos el proceso será el inverso, es decir, se leerá la cantidad de datos a recuperar y se irá reconstruyendo un dato (byte) cada ocho píxeles que se lean de la imagen.

III – RESULTADOS OBTENIDOS.

La aplicación ha sido desarrollada en el entorno C++ Builder, la cual permite el ocultamiento y la extracción tanto de texto como de archivos. Un resultado de esta aplicación se puede visualizar en la Figura 1.2 y 1.3, en la cual se oculto la frase: “Captura y Procesamiento Digital de Señales e Imágenes.” Como se observa, la imagen con el texto oculto no presenta ninguna modificación a simple vista.



Figura 1.2- Imagen Original



Figura 1.3- Imagen con Mensaje Oculto

De igual manera, vemos que el histograma de la componente Azul de la imagen no cambia de forma considerable.

Esto es debido a que el cambio de valores en la componente de Azul de la imagen, no es de grandes dimensiones, sino mas bien pequeños. Además, debemos tener en cuenta, que en algunos píxeles, no necesariamente se producirán cambios en el bit menos significativos, esto es, si el bit menos significativo del píxel y el bit procesado del dato (byte) coinciden, no habrá cambio en la componente azul de la imagen.

En la figura 1.4 se observa la poca influencia de los cambios en la componente Blue de la imagen.

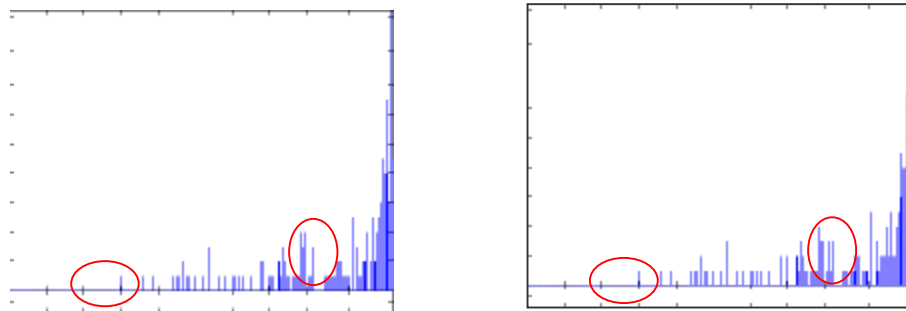


Figura 1.4

La posibilidad de trabajar con un bit de mayor peso en lugar del LSB existe, y es también válida, sin embargo los resultados pueden no ser los óptimos visualmente. Esto se debe a que los tonos que se modifican difieren mucho de los tonos originales. En la figura 1.5 se modifico al algoritmo para guardar la información en el bit mas significativo (MSB) de la componente de Blue.



Figura 1.5

En esta imagen se oculto un archivo de 6 KB, siendo la capacidad total para ocultar información de 12 KB. Al utilizar el MSB se está trabajando con el peor de los casos, tal como se puede observar en la imagen anterior.

El destinatario de esta imagen, al no tener la imagen original, no podría darse cuenta de la existencia de un patrón de información oculta, sino que la interpretaría como algún tipo de degradación en la imagen.

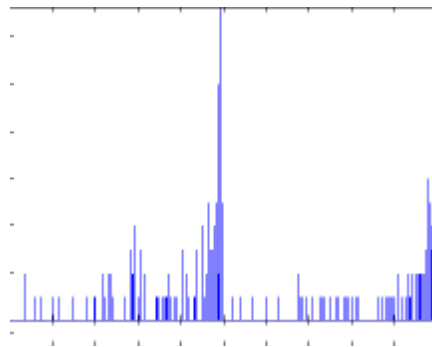


Figura 1.6

La Figura 1.6 corresponde al histograma de la imagen en la Figura 1.5. Como se observa, una gran cantidad de valores de la componente Azul cambiaron su bit más significativo (MSB); algunos habrán incrementado su tono en 128, otros lo habrán decrementado, y otros no se habrá modificado.

IV – MEJORA PROPUESTA.

En la técnica utilizada se pueden mejorar varios aspectos.

Cantidad de información a ocultar: en la implementación anterior, los datos a ocultar se guardaban en un único plano, por lo que por cada byte de información eran necesarios ocho píxeles de imagen. Si en lugar de utilizar una sola componente de color, se utilizarían las tres (RGB), se lograría triplicar la cantidad de espacio disponible para ocultar información; es decir, estaríamos ocultando un byte de información cada tres píxeles de la imagen. Esta mejora hace hincapié en maximizar el tamaño disponible para guardar información, a costa de modificar más los tonos de la imagen original.

Distribución de los datos: una mejora a nivel seguridad, sería implementar una función de distribución de la información a ocultar. Una forma de realizar este cometido es utilizando una clave, con la cual se van calculando las posiciones de los datos. Solamente la fuente y el destino deberían tener conocimiento de dicha clave. Otra posibilidad, sin este último requerimiento, es que la ubicación de la

información depende del contenido de la imagen. De esta manera se tiene un patrón de distribución diferente para cada imagen, logrando así una función de distribución adaptativa.

Encriptación de los datos: para lograr mayor confidencialidad de la información, se puede aplicar una técnica de encriptación de datos, previa a la utilización de cualquiera de los métodos de distribución de datos antes mencionados.

V – CONCLUSION

Entre los aspectos clave a valorar en cualquier algoritmo de ocultación de información cabe destacar:

- Cómo afecta a la calidad de la imagen
- Cuál es el costo computacional del proceso de introducción y extracción
- Robustez frente a manipulaciones posteriores de la imagen

El método descrito en el presente ejemplo no presenta inconvenientes en los dos primeros puntos. En lo que se refiere a la calidad de la imagen con información oculta, los resultados obtenidos han sido ampliamente satisfactorios. Por otro lado, se observa que el costo computacional de estos métodos es bajo, mientras no se procese información extremadamente grande. Sin embargo, resultará vulnerable a un buen número de modificaciones frecuentes en tratamiento de imagen. Se degradará seriamente si se realiza una conversión a un formato que incorpore compresión, como podría ser JPEG, o una simple ecualización de histograma.

VI – REFERENCIAS

- § Marcos Fáundez Zanuy, Tratamiento Digital de Voz e Imagen
- § <http://www.codeproject.com/csharp/steganodotnet.asp>
- § <http://www.wikipedia.com>