Benemérita
Universidad Autónoma de Puebla

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

Licenciatura en Ciencias de la Computación

Esteganografía con Imágenes Digitales y una Aplicación

TESIS

Que para obtener el grado de

Licenciado en Ciencias de la Computación

PRESENTA

Eleazar Nieves Zuñiga

Asesor

M.C. Alba Sánchez Gálvez

Coasesor

Dr. Manuel Martín Ortiz

Resumen

En este trabajo se presenta y se implementan, técnicas de esteganografía en imágenes digitales, utilizando el método de la Transformada Discreta de Coseno (DCT), el cual trabaja en el domino de las frecuencias y, el método del Bit Menos Significativo (LSB), que trabaja en el domino espacial. Los cuales consisten en insertar información en la imagen digital mediante la realización de modificaciones sobre la misma. La ejecución de este proceso sobre la imagen deberá ser imperceptible para el ojo humano, no afectando a su calidad. La información que se insertara tendrá un proceso previo, de encriptación utilizando RSA si el usuario a si lo desea, con ello proporcionándole una mayor seguridad a la información. Las imágenes portadoras de la información tendrán el formato JPEG, el sistema se programa en el lenguaje JAVA.

Índice

Introducción

Introducción

En el transcurso del tiempo la seguridad en las computadoras va tomado una gran importancia, en nuestra sociedad, globalizada, la cual se rige por la comunicación y el intercambio de información mediante el Internet, ya que este medio proporciona varias ventajas por mencionar algunas rapidez, y bajos costos, siendo comunicación e información partes fundamentales para su desarrollo, con ello se han adoptado diferentes formas para la protección de la información, ya que los riesgos de ser interceptada es muy alta, pudiendo a si plagiarla o modificarla. Por tal motivo el uso del la Criptografía, Esteganografía y Marcas de Agua, han tomado un papel muy importante para la seguridad de la información, dichas áreas tiene su campo de trabajo muy definido, dando resultados buenos, pero se pueden complementar la una con la otra y con ello darán resultados mejores. Aclarando que el concepto de seguridad en la información es mucho más amplio que la simple protección de los datos a nivel lógico. Para proporcionar una seguridad real hemos de tener en cuenta múltiples factores, tanto internos como externos.

De modo que la esteganografía, es un conjunto de técnicas que nos permiten ocultar o camuflar cualquier tipo de datos dentro de información considerada como válida, en nuestro caso imágenes digitales. En este trabajo se utilizaran técnicas de esteganografía para imágenes digitales, utilizando el método de la Transformada Discreta de Coseno (DCT), el cual trabaja en el domino de las frecuencias y, el método del Bit Menos Significativo (LSB), que trabaja en el domino espacial. Los cuales consisten en insertar información en la imagen digital mediante la realización de modificaciones sobre la misma. La ejecución de este proceso sobre la imagen deberá ser imperceptible para el ojo humano, no afectando a su calidad.

Los motivos por los cuales se desarrollo este trabajo, son diversos pero uno de los principales es el enfoque que le da la esteganografá, a los datos los cual van ocultos en diversos medios, con ello se les proporciona una forma de seguridad diferente, con respecto a la encriptación, su forma de actuar de la esteganografía permite burlar la vigilancia electrónica en el Internet, o simplemente que terceras personas no tengan acceso a dicha información. El tema de la información escondida, es relativamente nuevo exclusivamente esteganografía en imágenes, inicia con la publicación de un artículo en el año de 1992, el cual describe un método, con ello sentó las bases para la creación de nuevos métodos mas sofisticados. Con esto fueron interviniendo diferentes áreas por mencionar algunas, compresión de imágenes, redes, sistemas operativos, etc. La esteganografía se ha ido consolidando como otra opción para la seguridad de la información, los diferentes medios donde se puede insertar información son variados van desde, imágenes digitales, audio, código ejecutable, protocolo de redes, discos duros, texto. Existen otros métodos los cuales no son computables.

En este trabajo se implementaron los métodos: El LSB es uno de los mas fácil de implementar, en el cual no intervienen procesos ajenos al algoritmo, simplemente se aplica el algoritmo de codificación para insertar información en la imagen, de la misma forma el proceso inverso de, decodificación. Con el método del DCT no sucede lo mismo, la imagen es tratada con diferentes procesos por mencionar alguno, compresión de datos para poder realizar la inserción de la información.

Cabe mencionar que la relación que guardar Esteganografía y Marcar de Agua es muy estrecha, pero existe una diferencia muy notable, las Marcas de agua ("Watermarks"), éstas se utilizan para poner información visible que se perciba a simple vista, como podría ser algún fichero, texto plano, imagen, etc. Bien conocidas son las marcas de agua de los billetes, visibles a trasluz. A pesar de que las marcas de agua son un derivado de la información escondida, la gran diferencia que existe entre ellas es la intención con que se realizan cada una. La esteganografía oculta mensajes, porque el mensaje oculto es un medio de comunicación. Y las marcas de agua introducen información considerada atributos de la cubierta, es decir, es una forma de hacer un copyright o establecer una información de licencia.

Formato JPEG

El formato *Joint Photographic Experts Group* (JPEG) es un estándar de compresión de imágenes con pérdida. Dicho estándar fue creado en el año 1992 por un grupo independiente, llamado JPEG (Joint Photographic Experts Group), fundado en el año 1986. El estándar fue aprobado finalmente en 1994 convirtiéndose en un estándar ISO.

El documento se organiza de la manera siguiente:

En el **Capítulo 1,** Se presenta una introducción a la información escondida de manera general, enfocándonos en la esteganografía, denotando su historia, los tipos de esteganografía y sus aplicaciones.

En el **Capítulo 2,** Se describe el método de sustitución del bit menos significativo, en el dominio espacial mostrando su codificación y decodificación, en la inserción y extracción del mensaje.

En el **Capítulo 3,** Contiene, la implementación del método de esteganografía en el domino de las frecuencias, utilizando la transformada de coseno y compresión Huffman.

En el **Capítulo 4,** Se trata el proceso de encriptación del mensaje, utilizando el algoritmo RSA.

En el **Capítulo 5,** Describimos las etapas de ingeniería de software, que son análisis, diseño e implementación para el desarrollo del sistema.

En el **Capítulo 6,** Se presentan los resultados obtenidos, después de haber implementado los dos métodos de esteganografía.

Por ultimo las **Conclusiones** y la **Bibliografía**

Capítulo 1

Introducción a la Información Escondida

Esteganografia

La palabra estaganografia proviene de la palabra griega.

Stegos: Oculto, Secreto.
Graphy: Texto, o Dibujo.

La Esteganografía de la imagen define, como ocultar un mensaje secreto dentro de una imagen de una manera tal que otros no puedan discernir la presencia o el contenido del mensaje ocultado. El concepto de esteganografía se usa de manera muy frecuente, sin embargo el término más amplio es "Información Oculta". Como se muestra en la figura 1.1. La cual describe la jerarquía de la información escondida.

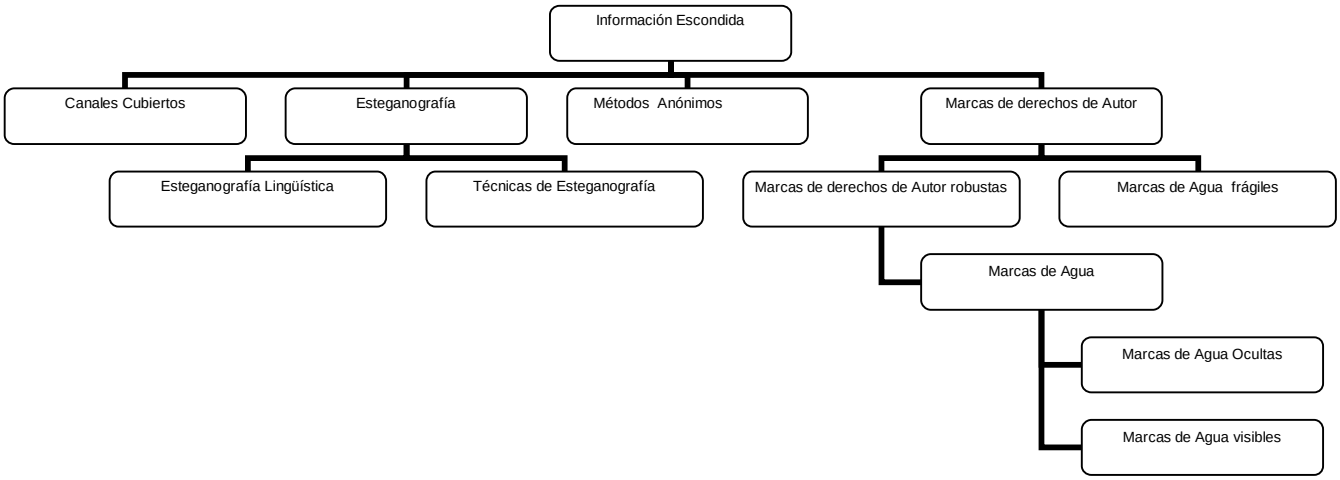


Fig. 1.1) Jerarquía de la información escondida

1.1.1 Historia de la Esteganografía.

Dentro de la historia la Esteganografía, ha tenido un papel interesante, data sus inicios en el año 486 – 425 A.C. Cuando el Historiador Herodoto, detalló como sus coetáneos, intercambiaban tablas cubiertas de cera lo cual daba una apariencia inofensiva, pero, por debajo de la cera en la base de la madera, fueron raspadas con el mensaje oculto.

Otro ejemplo es del griego Histiaeus, que deseaba informar a sus aliados cuándo atacar a sus enemigos. Para conseguir esto, él afeitó la cabeza de un criado en quien confiaba y después tatuaba un mensaje en su cabeza afeitada. Después de espera una temporada para el cabello del esclavo creciera de nuevo, le enviaron a través del territorio enemigo a los aliados.

Otros métodos esteganográficos más comunes y accesibles, a los que todos hemos tenido acceso sin tener ni idea de que era eso la esteganografía, son la escritura con tintas invisibles (leche,jugo de frutas,vinagre) que al ser expuestas al calor se oscurecen dejando entrever el mensaje oculto. Igualmente sencillo e inocente resulta enviar un mensaje escribiendo un texto, y luego, tomando la primera letra de cada palabra, de esta forma, se puede leer el mensaje que en realidad se quiere transmitir. Durante la II Guerra Mundial, comenzó a utilizarse el micropunto: un mensaje secreto, era fotográficamente reducido a la medida de un punto y pegado como el de la letra i en un papel que contuviese un mensaje cualquiera escrito.

1.1.2 Esteganografía y Criptografía

La diferencia entre Esteganografía y Criptografía, consiste en lo siguiente, cuando se usa únicamente Criptografía los datos pueden ser ilegibles, dicho mensaje puede contener números o caracteres, en un orden sin relación alguna aparentemente, pero el mensaje puede despertar sospechas, deduciendo que existe información escondida. A diferencia de la esteganografía que oculta los datos en un medio portador como pude ser, imágenes, audio, video, texto. De esta forma camuflageando la información no despertará sospechas el medio portador.

CRIPTOGRAFIA ⇒ **INFORMACION ILEGIBLE**

ESTEGANOGRAFíA ⇒ **INFORAMCION OCULTA**

La forma en que actúan juntas la criptografía y esteganografía es que la criptografía hace el dato ilegible a quien no conozca la clave y la esteganografia oculta la existencia de esos datos. Así los archivos siendo ocultados, hacen que no sea ni leído ni detectado fácilmente. Hoy en día, con la ayuda de las computadoras ambas técnicas son perfectamente combinables, complementándose la una a la otra y consiguiéndose una seguridad aún mayor.

Con respecto a la esteganografía no hay leyes asociadas y la criptografía tiene algunas leyes asociadas. Por su naturaleza la esteganografía despierta menos sospechas, por los medios que utiliza, imágenes, video audio. A diferencia de la criptografía que utiliza caracteres en orden aleatorio.

1.1.3 Principios de la Esteganografía

Simmons describe un excelente ejemplo de esteganografía, común en lo que se llama problema de "Prisioneros". Alice y Bob son los dos personajes de ficción, en este ejemplo, que han sido detenidos y se colocan en diferentes celdas. Su objetivo es desarrollar un plan de escape, que los seque fuera de la cárcel (Fig. 1.2), la única forma de comunicarse es a través de la guardia, Wendy. Al ser un guardia capaz, Wendy no permitirá que Alice y Bob se comunique en código (encriptación). Y si ella ve algo sospechoso, uno de ellos o ambos será inmediatamente puesto en régimen de aislamiento. Así que Alice y Bob deben comunicarse de una manera que no sucite la sospecha, deben comunicarse invisiblemente utilizando esteganografía. [2]

Una manera inteligente de hacerlo es a ocultar la información en un mensaje de aspecto inofensivo o con imágenes. Bob podría hacer un dibujo de una vaca de color azul en una zona verde de pasto, y pedir una Wendy entregar a Alice. Wendy, por supuesto, mira la imagen que pasa y pensando que es sólo una pieza de arte abstracto, y no le da importancia, sin saber que los colores en la imagen que transmitió es el mensaje.

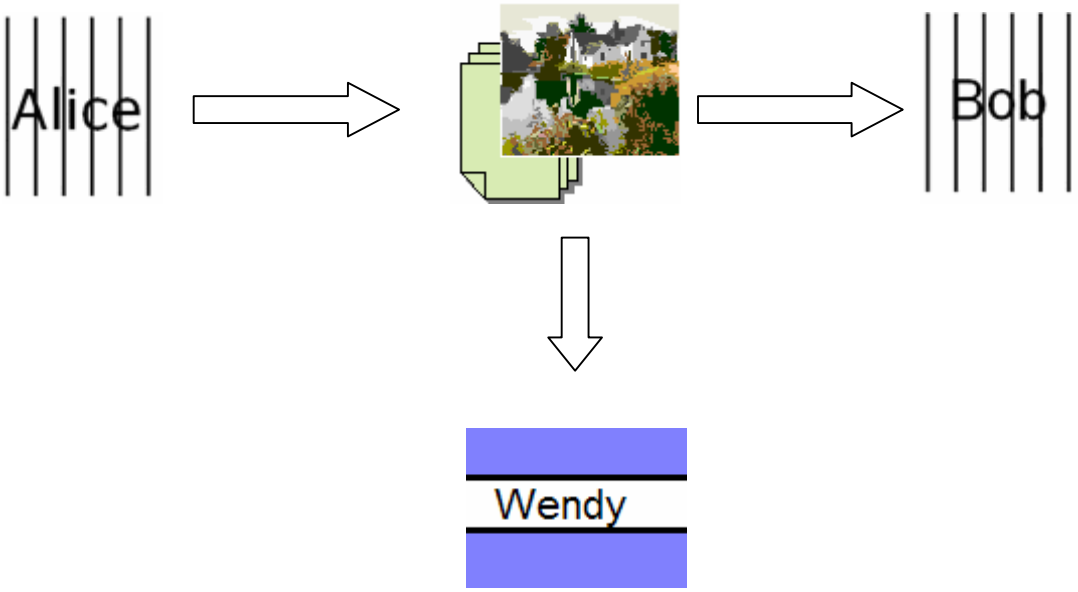


Fig. 1.2 Problema de Prisionero

El Problema de los “Prisioneros” es un modelo que puede aplicarse a una gran cantidad de situaciones en la esteganografia que puede ser utilizado para la comunicación. Alice y Bob son las dos partes que quieren comunicarse y Wendy es la «escucha», y mientras que este modelo puede ser un medio eficaz de comunicación, el potencial de los pasivos, activos, o ataques maliciosos siempre debe ser considerados.

De esta forma se planteo los elementos básicos de la esteganografia, y como se utiliza para comunicarse, vamos a mostrar algunas técnicas reales, que se están utilizando, para ilustrar algunas de las formas que pueden tomar la esteganografia y la eficiencia que puede tener.

1.2 Técnicas de Esteganografía

Sistema de Sustitución: Sustituye redundantes o innecesarios bits con una cubierta de los bits del mensaje secreto (LSB).

Métodos del Dominio Frecuencia: Estos otros están relacionados con algoritmos de modificación y transformación de la imagen. Ocultan los mensajes en las áreas más significativas de la tapadera y pueden manipular las propiedades de la imagen como es la luminosidad. Normalmente se crean máscaras indicando los lugares más idóneos para la ocultación de información. Estas técnicas son más complejas de realizar y más robustas que las anteriores.

1.3 Esteganografía Pura

Denominamos a un sistema de Esteganografia puro cuando no es requerido, un previo cambio de alguna información secreta (Stego - Key), formalmente puede ser expresado como. $E : CxM \to C$, donde *C* son las posibles conversiones y M son los posibles mensajes.

El proceso de extracción consiste en la siguiente función, $D : C \to M$, en donde definimos la extracción del mensaje secreto, claramente esto es necesario si $|C| \geq |M|$. Pero al enviar y recibir deben de ser accedidos por los algoritmos de implantar y extraer, el algoritmo no debe ser publico. [1]

Definición 1: **(Esteganografía Pura)** la cuádrupla $G = (C, M, D, E)$, donde C son las posibles transiciones, M es el mensaje secreto, con $|C| \geq |M|, E : CxM \to C$ es la función de inserción y $D : C \to M$ es la función extracción, con la propiedad de $D(E(c, m)) = m$ par todo $m \in M$ y $c \in C$ a esto lo nombramos como Sistema de Esteganografía pura. [1]

1.4 Esteganografía con Clave Secreta.

La Esteganografía con clave secreta, es un sistema similar a sistemas de cifras simétricas: al enviar escogemos a *c* en la cual llevará la clave secreta k. si la clave es usada en el proceso de implantación y sabemos como recuperarla, nosotros podemos revertir el proceso de extracción del mensaje secreto. Cualquiera que no sepa la clave secreta no deber ser capaz de obtener de la información codificada.

Definición 2. (**Esteganografía de Clave Secreta**) La quíntupla $G = (C, M, K, D_k, E_k)$, donde C son las posibles transiciones, M es el mensaje secreto con $|C| \geq |M|$, K la clave secreta, $E_k : CxMxK \to C$ y $D_k : CxK \to M$ con la propiedad de $D_k(E_k : (c, m, k)) = m$ para todo $m \in M$ y $k \in K$ esto es nombrado sistema esteganográfico de clave secreta. [1]

1.5 Esteganografía de Clave Pública

La Esteganografía de clave pública es un sistema que requiere usar dos, claves; una privada y una pública, la clave pública es libre en la bases de datos, de manera que la clave pública es usada para en el proceso de implantación y la clave privada es usada en el proceso de reconstrucción del mensaje, se usa de la misma forma la clave pública de sistema esteganográfico que de un sistema criptográfico.

1.6 Aplicaciones de la Información Escondida

La comunicación discreta es requerida por los militares y agencias de inteligencia.

Monitoreo Automático de materiales protegidos bajo ciertos derechos en la Web: Un robot que busca en la Web material marcado y por lo tanto identifica usos ilegales [2].

Monitorización de la difusión: mediante la inserción de marcas en anuncios comerciales, un sistema automatizado puede supervisar si los anuncios están siendo difundidos como se contrató. También puede servir para controlar la emisión de algunos programas de televisión solamente en las emisoras acordadas.

Argumentación de datos: La información es agregada para beneficio del grupo, estas pueden ser detalles del trabajo, anotaciones, otros canales, o comprar información.

Interferir en Pruebas: la información se esconde en objetos digitales, puede ser una señal, “summary”, la cual puede ser usada para prevenir o a detectar modificaciones sin autorización.

Protección de los derechos de autor: destinadas a proteger la propiedad intelectual. Los datos originales llevan insertados una marca conteniendo toda la información sobre los derechos de autor. Así, el dueño de los datos puede demostrar ante un tribunal si alguien ha infringido estos derechos.

1.7 Estegoanalisis

El Estegoanálisis es un campo de investigación que crea herramientas autónomas para detectar el uso de esteganografia. La dificultad del estegoanalisis depende de que tanto sepamos, de la existencia de un mensaje oculto, que técnica o medio se usó para ocultar el mensaje y en donde podemos buscar el mensaje. Por tal motivo el nuevo objetivo a alcanzar es el “Estegoanálisis a ciegas” el cual tiene la habilidad de reconocer esteganografía en archivos digitales sin saber nada acerca de la herramienta de software usado para el ocultamiento de datos. El estado actual del estegoanáslisis a ciegas no es bien conocido y muchas de las investigaciones son confidenciales.

Capítulo 2

Esteganografía en el Dominio Espacial

2.1 Imagen Digital

El término imagen se refiere a una función bidimensional de la luz y la intensidad, a la que indicamos por $f(x,y)$, donde el valor o amplitud de f en las coordenadas espaciales (x,y) da la intensidad (iluminación) de la imagen en este punto. [6].

Para ser utilizable para el procesamiento por medio de computadora, una imagen $f(x,y)$ debe ser digitalizada tanto espacialmente como en su amplitud. La digitalización de las coordenadas espaciales (x,y) se denomina muestreo de la imagen y la digitalización de la amplitud se conoce bajo el nombre de cuantificación del nivel de gris.

Se supone que una imagen continúa $f(x,y)$, se describe de forma aproximada por una serie de muestras iguales espaciales organizadas en forma de una matriz *M x N* como se indica en la ecuación 2.1, donde cada elemento de la matriz es una cantidad discreta:

$$f(x,y) = \begin{pmatrix} f(0,0) & f(0,1) & .... & f(0,2) \\ f(1,0) & f(1,1) & .... & f(1,M-1) \\ .... & .... & .... & f(1,M-1) \\ f(N-1,0) & f(N-1,0) & .... & f(N,1,M-1) \end{pmatrix} \quad \textbf{Ec. 2.1}$$

El término de la derecha de la ecuación representa lo que comúnmente se denomina una imagen digital. A cada elemento de la matriz se le puede denominar elemento de la imagen, píxel. Los términos imagen y píxeles van a emplearse para indicar una imagen digital y sus elementos. [6]

2.2 Dominio Espacial

El término domino espacial se refiere al conjunto del píxeles que componen una imagen, los métodos de domino espacial implican la generación de una nueva imagen modificando el valor del píxel en una simple localización, basándose en una regla global aplicada a cada localización de la imagen original. El proceso consiste en obtener el valor del píxel de una localización dada en la imagen, modificándolo por una operación lineal o no lineal y colocando el valor del nuevo píxel en la correspondiente localización de la nueva imagen. El proceso se repite para todas y cada una de las localizaciones de los píxeles en la imagen.

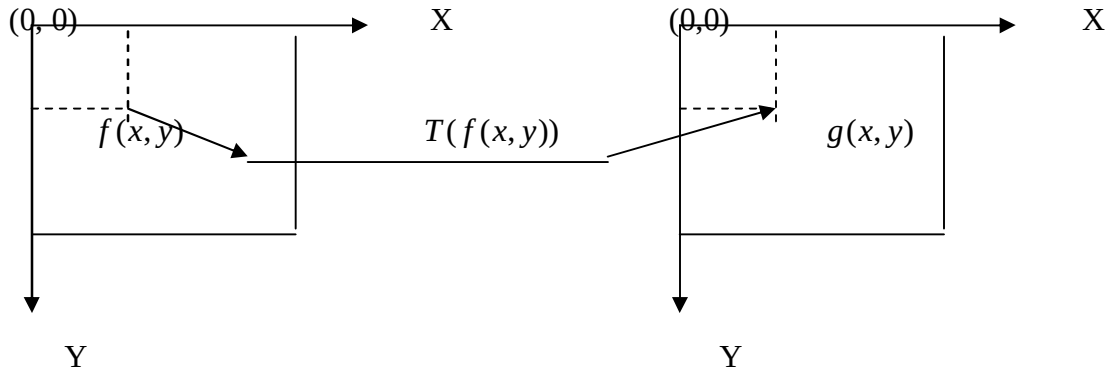


Fig. 2.1 Transformación de la Imagen

Como se aprecia en la figura 2.1, el operador es una transformación T uno a uno. El operador T se aplica a cada píxel en la imagen o sección de la imagen y la salida depende únicamente de la magnitud del correspondiente píxel de entrada; la salida es independiente de los píxeles adyacentes. Determinado por la ecuación.

$$g(x,y) = T\left(f\left(x,y\right)\right)$$

De tal forma $f(x,y)$ es la imagen de entrada y $g(x,y)$ es la imagen procesada, y T es un operador que actúa sobre f definido en algún entorno de (x,y). En este caso g depende solo del valor de f en el punto (x,y) y T se convierte en una función de transformación del nivel de gris, también denominada correspondencia de la forma:

$$s = T(r)$$

Donde, para simplificar la notación, r y s son variables que indican el nivel de gris de $f(x,y)$ $g(x,y)$ en cada punto (x,y). [6]

2.3 Ocultando la información

Cuando se oculta información en archivos gráficos (imágenes) se aprovechan los bits menos significativos de los colores RGB, para introducir en ellos la información (con lo que se hace una reducción de colores respecto a la imagen original, si es necesario). Si la relación entre la información a ocultar, el tamaño de la imagen y el número de colores es buena, resulta prácticamente imposible diferenciar la imagen original de la imagen con información oculta.

Por lo tanto, un mensaje se puede ocultar dentro de una imagen cambiando los bits menos significativos utilizando métodos del domino espacial, o de frecuencias, teniendo en cuenta los diferentes formatos de imágenes digitales, ya que en unos formatos la inserción de la información es mas fácil, por ejemplo BMP, en este trabajo se utilizo el formato JPEG el cual pose una calidad de imagen buena, con ello la imagen portadora o Stego-Imagen, en el proceso de inserción no sufre cambios notables en la calidad de la imagen, por tal motivo los cambios son imperceptibles para el ojo humano, de manera general este proceso se describe en la figura 2.2. Es importante que la estego-imagen no contenga características, que faciliten su detección, ya que se modificaran por la inserción del mensaje, de esta forma pueden ser detectadas por vigilancia electrónica, utilizando técnicas de estegoanálisis para detectar las imágenes que contienen mensajes ocultos. Una vez que se alcance esto, la herramienta esteganografía deja de ser útil.

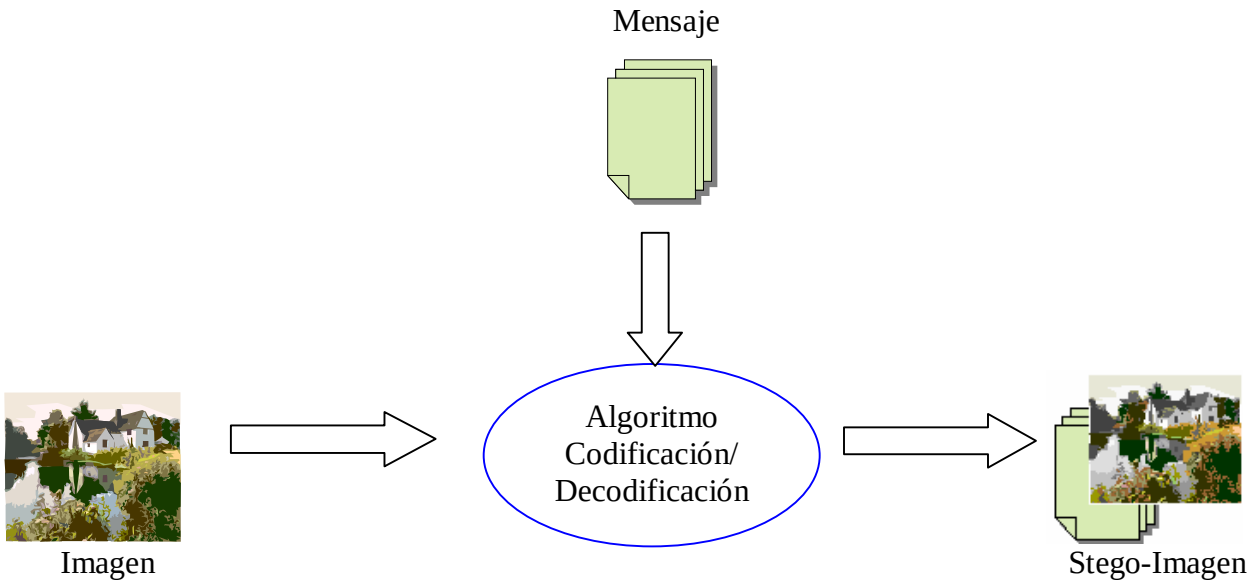


Fig. 2.2 Proceso general

2.4 Métodos de Sustitución LSB (Less Significant Bit)

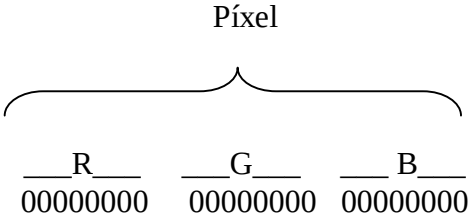
Este método es uno de los más comunes por su fácil implementación, ya que trabaja directamente con los bits de la imagen, que cosiste en modificar el bit menos significativo de cada píxel, insertando el valor que se desea almacenar. Sin ninguna alteración previa del píxel. En nuestro caso el mensaje que se insertara tendrá un proceso previo de encriptación, si el usuario a si lo requiere.

Por tal motivo se denominan métodos del domino espacial, a diferencia de los métodos del domino de frecuencias, que si tienen un proceso previo de los píxeles antes de implantar el valor. Por tal motivo les proporcionan una seguridad más alta. Los métodos de domino espacial no son muy robustos en seguridad, por lo cual se consideran de primera generación en la esteganografía, dichos métodos tienen diferentes formas de implementación.

En este proyecto se implementara el método de LSB, aplicando una máscara con las operaciones lógicas *AND (&) y OR* (|), al bit menos significativo de cada píxel, el cual se ubica en la parte extrema derecha de los píxeles.

Los tamaños de las imágenes son muy variados, por ejemplo una imagen con una dimensión 256 x 256 píxels y 256 colores (o 8 bits por píxel), el tamaño de la imagen es de unos 23.4 KB (23,965 bytes), las imágenes digitales normalmente se almacenan un una calidad de 24 bits (16.000.000 de colores). Una imagen de 24 bits es lo ideal para esconder información, pero estas imágenes pueden llegar a ser bastante grandes en tamaño. Las imágenes de 24 bits utilizan 3 bytes por cada píxel para representar un valor de color. Estos bytes podrían se representados en decimal, los valores que podrían tomar cada uno de estos bytes van desde 0 a 255. Cada uno de estos bytes representa los colores de RGB.

Características de un píxel:



De modo que un píxel se forma con 24 bits

Los bits que se sustituirán con el método LSB son:

___R___	___G___	___ B___
0000000 |**0**|	0000000 |**0**|	0000000 |**0**|

Supongamos que se quiere insertar el mensaje “HOLA”, en un pequeño pedazo de una imagen se hace necesario cuatro bytes para esconderlo ya que el mensaje cuenta con cuatro letras, consultado la tabla ASCII, se obtiene el código en decimal 72, 79, 76, 65, que en binario corresponde a 01001000, 01001111, 01001100, 01000001, respectivamente.

Alterando el último bit de cada byte resulta la imagen después de insertar el mensaje los bits que tienen una marca abajo, son los que se sustituyeron por los bits del mensaje y que juntos forman la cadena HOLA en binario

```
00010100 10100101 01010100 00110100 01110101 01000010 01010010 01101010
       -        -        -        -        -        -        -        -
00001010 01010101 10100100 01010110 11010111 10000101 01010011 01010011
       -        -        -        -        -        -        -        -
10101000 10101011 10001000 10100100 00010101 10100101 00010100 10100100
       -        -        -        -        -        -        -        -
10101100 10101011 00001000 10100100 00010000 10100100 11010100 10100101
       -        -        -        -        -        -        -        -
```

2.4.1 Proceso de Codificación:

El algoritmo de codificación es el que realiza, la tarea de ocultar el mensaje en la imagen digital, el cual va enmascarando los RGB de la imagen. En este algoritmo se usan los operadores & (and) y | (or), que efectúan las operaciones lógicas and y or con los bits respectivamente y El operador >>> que realiza corrimientos de bits a la derecha.

Denotamos con l_m y l_im la longitud del mensaje y la longitud de la imagen respectivamente a continuación se describe el algoritmo de codificación.

```
Si  ( l_m>  l_im ) entonces
            Letrero ("El mensaje excede el tamaño de la imagen")
  Fin _ si

Si_ no
      Para  i = 0  hasta n
            dato ← mensaje  ( i )

          Para  bit  = 7 hasta bit-1,  j + 1
                b ← (dato >>> bit ) &  1
           imagen ( j ) ←(( imagen ( j ) & 0xff) | b)
        Fin _ para
    Fin _ para
 Fin _ si no
```

El algoritmo de decodificación es similar al de codificación,

2.4.2 Proceso de Descodificación

```
Para  i = 0 hasta  n hacer
        arregló _ auxiliar  ← (arregló _ auxiliar << 1) | (imagen( i ) & 1)
fin_ para

      resultado(tamaño)

        para  b = 0 hasta( resultado(arregló _ auxiliar) )  hacer
          para  x = 0  hasta n, j hacer
                      resultado ( b) ← ((resultado ( b) <<1) |  (imagen(j) & 1))
         fin _ para
      fin _ para
```

Utilizando los algoritmos descritos anteriormente, tanto para codificación y decodificación obtenemos la stego-imagen, la cual es una imagen que es muy similar a la original, no teniendo cambios significativos. Estos tipos de métodos tienen la característica que pueden almacenar un numero mayor de información con respecto a los de frecuencias.

2.5 Localización del mensaje en la imagen

Las técnicas de esteganografía LSB permiten insertar la información en las imágenes, de diferentes formas siguiendo un patrón determinado en el proceso de inserción del mensaje, los cuales pueden ser desde algoritmos que van seleccionando los bytes de forma aleatoria, seguir una trayectoria o bien usando fractales. En este trabajo, el algoritmo que se usa, inserta el mensaje en las primeras filas de la imagen, de arriba hacia abajo y de izquierda a derecha, el número de las filas que se usan depende del tamaño del mensaje, con lo cual el este se sitúa en la parte superior de la imagen como se muestra en la figura 2.3.

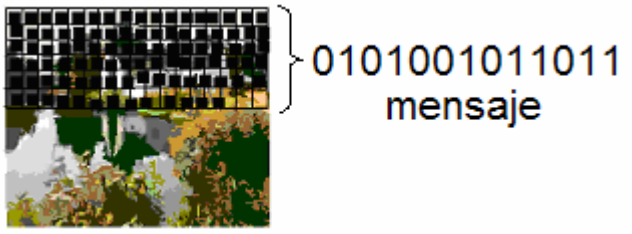


Fig. 2.3 Imagen portadora del mensaje

De igual forma podemos aproximar el número de información a insertar por (m * n * 3)/8 caracteres donde m * n representan el número total de píxeles de la imagen, la razón de multiplica por tres es debido a que en cada píxeles se almacenan tres bits y se divide entre ocho porque ocho bits representan un byte.

Capítulo 3

Implementación del Método de Esteganografía en el Dominio de las Frecuencias

3.1 Formato JPEG

El formato JPEG, es uno de los formatos mas utilizados para el intercambio de información gráfica vía Internet. Esto es así porque se trata de un formato comprimido, precisamente utilizando la misma compresión de su mismo nombre. Las razones de compresión son elevadas y, en consecuencia el tamaño de los datos se reduce considerablemente. Utiliza un método de compresión con perdida basado en la transformada de coseno.

La DCT (al igual que las demás transformadas) puede aplicarse a codificación de imágenes, desde un punto de vista de relación de ancho de banda o compresión de datos. El objetivo es conseguir que la imagen (dominio espacial) o secuencia de imágenes (dominio espacial-temporal), se traslade a un dominio transformado de tal manera que se reduzca el ancho de banda para la transmisión o para el almacenamiento, para la recuperación de la imagen mediante la transformada inversa, no presenta una distorsión perceptible. El paso final de la codificación basada en la DCT es la codificación por entropía. Este paso consigue una compresión sin pérdidas adicionales al codificar los coeficientes cuantificados de la DCT de una forma compacta basándose en sus características estadísticas. La propuesta de JPEG especifica dos métodos, de codificación, por entropía la cual proporciona un mínimo teórico para determinar el número medio bits/píxel; o también llamada Codificación Huffman, y codificación Aritmética. El modelo base para codificación y descodificación se ejemplifica en la figura 3.1 y figura 3.2 respectivamente.

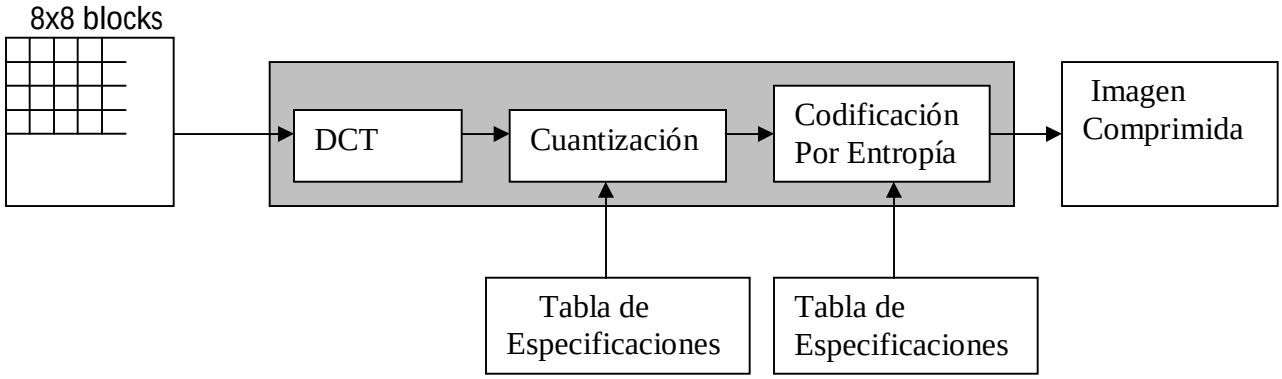


Fig. 3.1 Codificación Base DCT (JPEG)

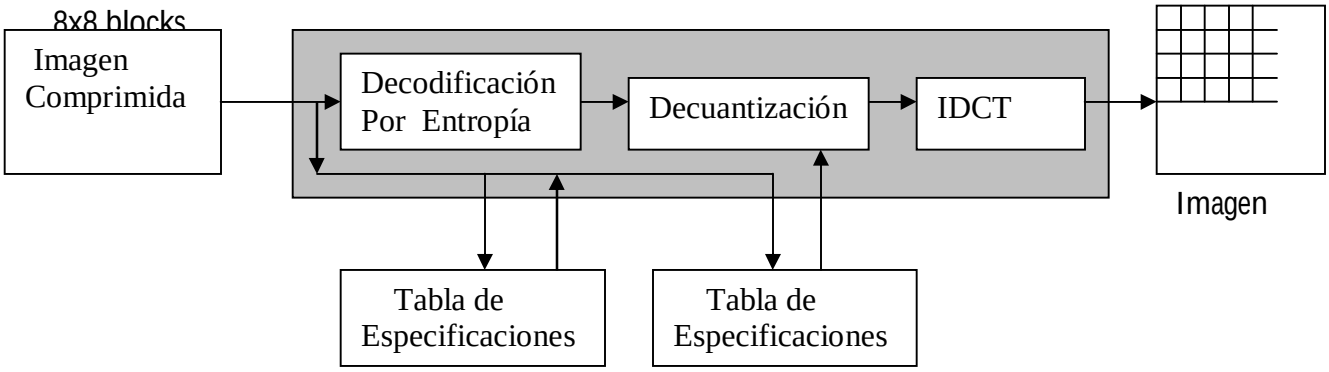


Fig. 3.2 Decodificación Base IDCT (JPEG)

La codificación Huffman requiere que la aplicación especifique uno o más conjuntos de tablas de códigos Huffman, estas deben ser las mismas para comprimir una imagen y para descomprimir. Las tablas de códigos Huffman pueden ser predefinidas por la aplicación o bien calculadas específicamente para una imagen. La codificación Huffman convierte los valores de brillo de los píxeles de la imagen original en nuevos códigos de longitud variables, basado en sus frecuencias de ocurrencias en la imagen. De esta manera, los valores de brillo que ocurren más frecuente se les asignan los códigos más cortos y los valores de brillo que ocurren con menos frecuencia se les asignan los códigos más largos. El resultado es que la imagen comprimida requerirá de menos bits para describir la imagen original.

3.2 Fundamentos del color

El modo que percibe los colores el cerebro del humano, es un fenómeno psicofisiológico, que aún no se ha llegado a entender completamente, la naturaleza física del color se puede expresar en una base formal corroborada por los resultados experimentales y teóricos. El uso del color en el procesamiento de imágenes, define dos factores principales. En primera instancia el análisis de la imagen, el color es un potente descriptor que a menudo simplifica la identificación y extracción de objetos de una escena. En segundo lugar, el ojo humano puede distinguir una amplia gama de colores comparando con los niveles de gris.

En el procesamiento de imágenes, el color se divide en dos áreas fundamentales: color y pseudocolor, en la primera categoría se procesa las imágenes obtenidas con un sensor de color. Estas imágenes constan básicamente de tres bandas; rojo, verde y azul. En la segunda las imágenes monocromáticas.

3.2.2 El modelo RGB

El objetivo de un modelo de color es facilitar la especificación de los colores de una forma normalizada y aceptada genéricamente. En esencia, es la especificación de un sistema de coordenadas tridimensional y de un subespacio de este sistema en el que cada color quede representado por un único punto. [15]

El modelo RGB cada color aparece en sus componentes espectrales primarias; rojo, verde, azul. Este modelo esta basado en el sistema de coordenadas cartesianas, el subespacio de color de interés es el tetraedro mostrado en la figura 3.3

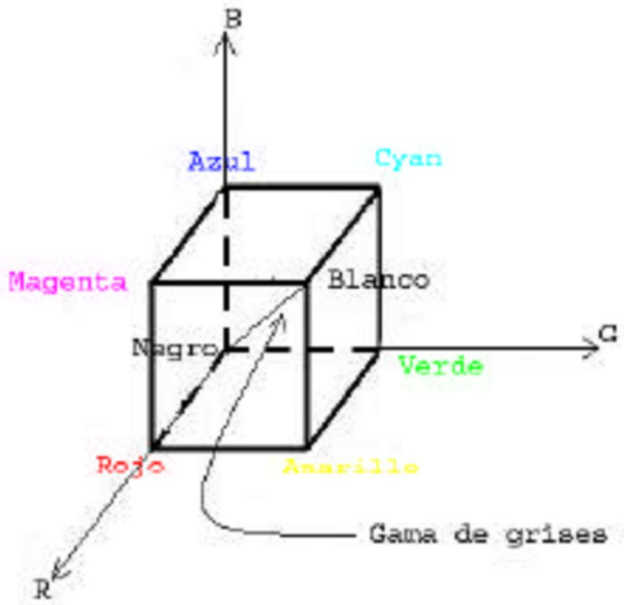


Fig.3. 3 representación gráfica del modelo de color RGB

La representación grafica del modelo RBG (ver figura 3.3) se realiza mediante un cubo unitario con los ejes R, G Y B. El origen (0,0,0) representa el negro y las coordenadas (1,1,1) el blanco. Los valores del cubo en cada eje R, G y B, de coordenadas (1,0,0), (0,1,0) y (0,0,1) representan los colores primarios rojo, verde y azul, los restantes tres vértices (1,0,1), (0,1,1) y (1,1,0) al magenta, cian y amarillo respectivamente. La diagonal del cubo representa la gama de grises desde el negro al blanco. En esta diagonal cada punto o color se caracteriza por tener la misma cantidad de cada color primario. [13]

3.2.3 Modelo YCbCr

YCbCr es una codificación no lineal del espacio de color RGB, usada comúnmente por los estudios de televisión europeos y en la compresión de imágenes. El color es representado por la luminancia (Y) y por dos valores diferentes de color (Cb y Cr) que son características calorimétricas del color. La luminancia es la cantidad lineal de luz, directamente proporcional a la intensidad física, y ponderada por la sensibilidad de la percepción humana visible al espectro. La luminancia puede ser calculada como la suma ponderada de los componentes lineales del espacio de color RGB. La obtención de este espacio de color a partir del RGB es la siguiente:

$$Y = 0.299\,R + 0.587\,G + 0.114\,B$$
$$Cb = -0.1687\,R - 0.3313\,G + 0.5\,B + 128 \qquad \text{Ec. 2 (conversión de RGB a YCbCr)}$$
$$Cr = 0.5\,R - 0.4187\,G - 0.0813\,B + 128$$

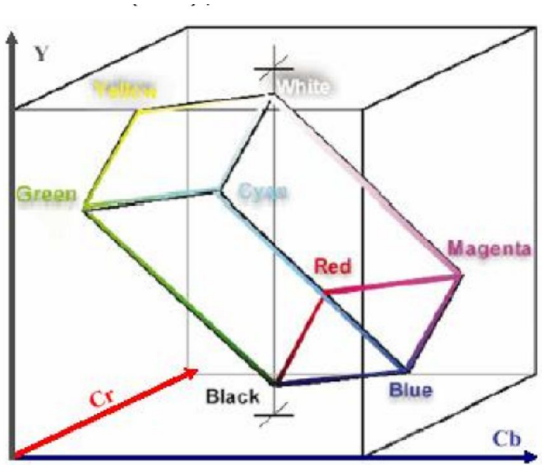


Figura (3.4) Modelo YCbCr

El espacio YCbCr permite una elevada eficiencia de codificación, pero provoca una distorsión en la colorimetría original debido a los redondeos de la transformación. El modelo RGB presenta alta redundancia que provoca baja eficiencia de codificación. La forma inversa para obtener el modelo de color RGB apartir del modelo YCbCr es la que a continuación se presenta:

$$R = Y + 1.402Cr$$
$$G = Y - 0.34414Cb - 0.71414Cr \qquad \text{Ec. 2 (conversión de YCbCr a RGB)}$$
$$B = Y + 1.722C$$

3.3 Características Generales DCT

La Transformada Coseno esta basada en la Transformada de Fourier Discreta (DFT), pero utilizando únicamente números reales, es decir, se trata de una transformada real debido a que los vectores base se componen exclusivamente de funciones coseno. La DCT toma un conjunto de puntos de un dominio espacial y los transforma en una representación equivalente en el dominio de frecuencias. Además minimiza algunos de los problemas que surgen con la aplicación de la DFT a series de datos.

La Transformada Discreta del Coseno cuenta con una buena propiedad de compactación de energía, que produce coeficientes no correlacionados, donde los vectores base de la misma dependen sólo del orden de la transformada, y no de las propiedades estadísticas de los datos de entrada.

La decorrelación de coeficientes es muy importante para la compresión, ya que así, el tratamiento de cada coeficiente en un momento posterior puede realizarse independientemente unos de otros, sin que se pierda eficiencia de compresión. Otro aspecto importante de la DCT es la capacidad de cuantificar los coeficientes utilizando valores de cuantificación que se eligen de forma visual.

Esta transformada ha tenido un gran éxito en el campo del tratamiento digital de imágenes, debido a que, para los datos de una imagen convencional, se tiene una alta correlación entre los elementos. Otro motivo de utilizar la transformada del coseno en lugar de la de Fourier, de mayor uso y aplicación, radica en que la primera puede codificar mejor funciones lineales con menos componentes.

3.3.1 Transformada Discreta de Coseno Unidimensional (DCT)

La DCT es aplicada sobre cada matriz de 8x8 píxeles y nos devuelve una matriz de 8x8 con los coeficientes de la frecuencia. Es decir, se utiliza para codificar los valores de la imagen, devolviendo un campo de valores transformados que serán una serie de coeficientes que multiplicarán a funciones coseno para reconstruir la imagen.

La respuesta del sistema visual humano depende de la frecuencia espacial. Si pudiéramos, de algún modo descomponer una imagen en un conjunto de subimágenes, cada una con una frecuencia espacial particular, podríamos separar la estructura de la imagen que el ojo puede ver a partir de la estructura que es imperceptible.

La DCT puede proporcionar una buena aproximación a esta descomposición. Para comprender cómo una imagen puede ser descompuesta en sus frecuencias espaciales fundamentales, primero consideraremos el caso unidimensional. [5]

La DCT unidimensional se define como:

$$F(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cos\left(\frac{(2x+1)u\pi}{2N} \right)$$

La inversa DCT se define como:

$$f(x) = \sum_{u=0}^{N-1} \alpha(u) F(u) \cos\left(\frac{(2x+1)u\pi}{2N} \right)$$

donde:

$$C(u) = 1/\sqrt{2} \; para \;\; u = 0$$
$$C(u) = 1 \;\; para \; u > 0$$

La figura 3.6 representa un conjunto de ocho funciones base cosinusoidales de amplitud uniforme, cada una muestreada en ocho puntos. La forma de onda superior izquierda (u = 0) es simplemente una constante, mientras que las otras siete (u = 1, ... , 7) presentan un comportamiento alterno a frecuencias más altas progresivamente.

Estas formas de onda (que se denominan funciones base cosinusoidales), son ortogonales. El coeficiente que pondera la función base constante (u=0) se denomina coeficiente DC. El resto de coeficientes se denominan coeficientes AC. Observar que el término DC proporciona el promedio sobre el conjunto de muestras.

El proceso de descomponer un conjunto de muestras en un conjunto ponderado de funciones base cosinusoidales se denomina Transformada Discreta del Coseno Directa (DCT), y al método de reconstruir el conjunto de muestras a partir del conjunto ponderado de funciones base cosinusoidales se le conoce como Transformada Discreta del Coseno Inversa (IDCT). Si la secuencia muestreada es mayor de ocho muestras puede dividirse en grupos de ocho muestras y la DCT puede calcularse independientemente para cada grupo.

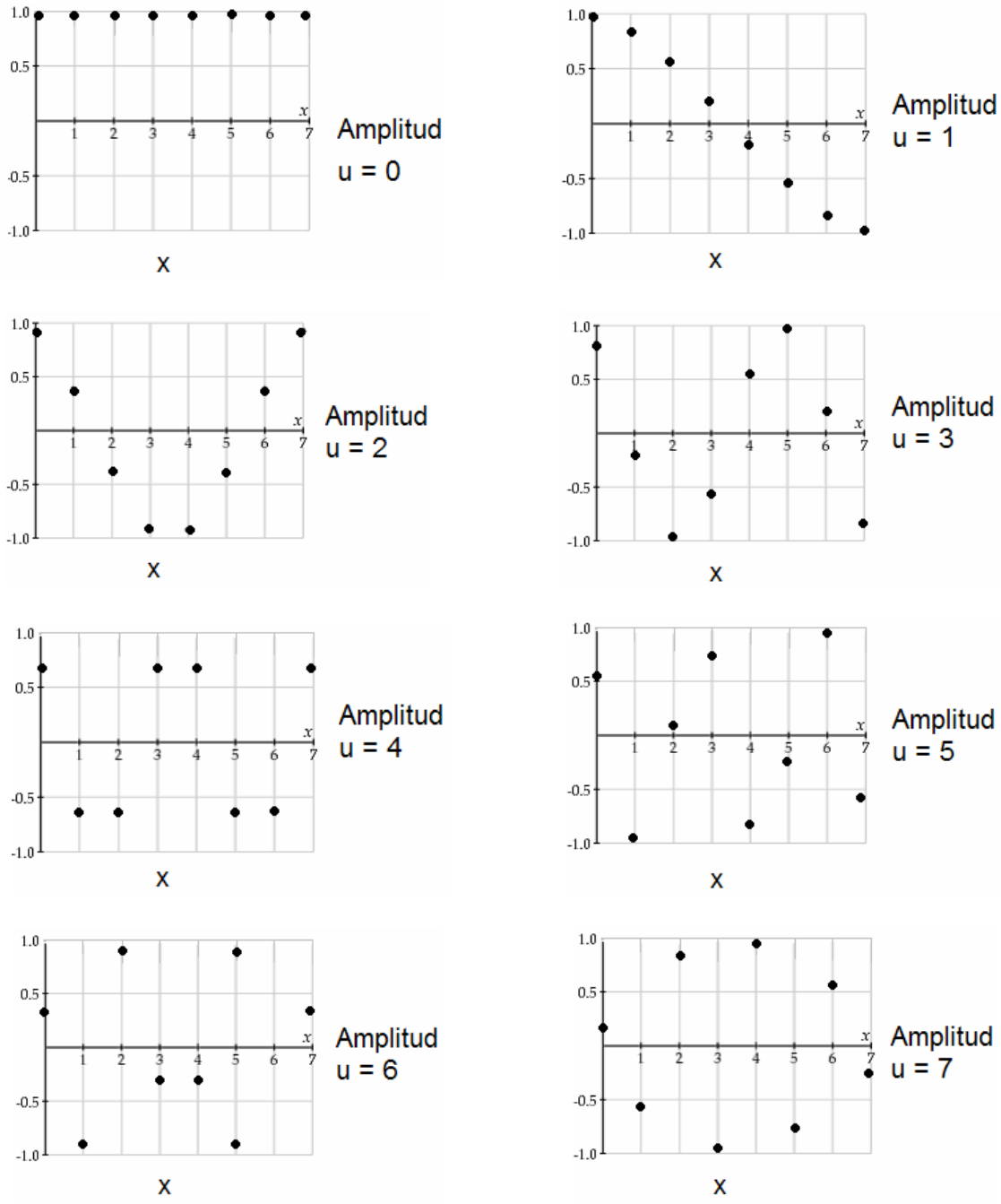


Fig. 3.6 Funciones base cosenosoidales

3.3.2 Transformada Discreta de Coseno Bidimensional (DCT)

La DCT unidimensional puede extenderse a dos dimensiones para su aplicación a imágenes. la Fig. 3.7 muestra un conjunto de 64 funciones base cosinusoidales bidimensionales (imágenes base) que se generaron multiplicando un conjunto de funciones base unidimensionales (de ocho puntos).

Las imágenes base orientadas horizontalmente representan las frecuencias horizontales y las orientadas verticalmente representan las frecuencias verticales. Por convenio, el término DC de las funciones base horizontales esta situado a la izquierda, y arriba en el caso de las funciones base verticales. Por consiguiente, la fila superior y la columna de la izquierda tienen variaciones de intensidad en una dimensión, que si se representan, serían las mismas que las de la figura 3.7 (Para propósitos de ilustración, un gris neutro representa cero en estas figuras, el blanco representa amplitudes positivas, y el negro representa amplitudes negativas.)[5]

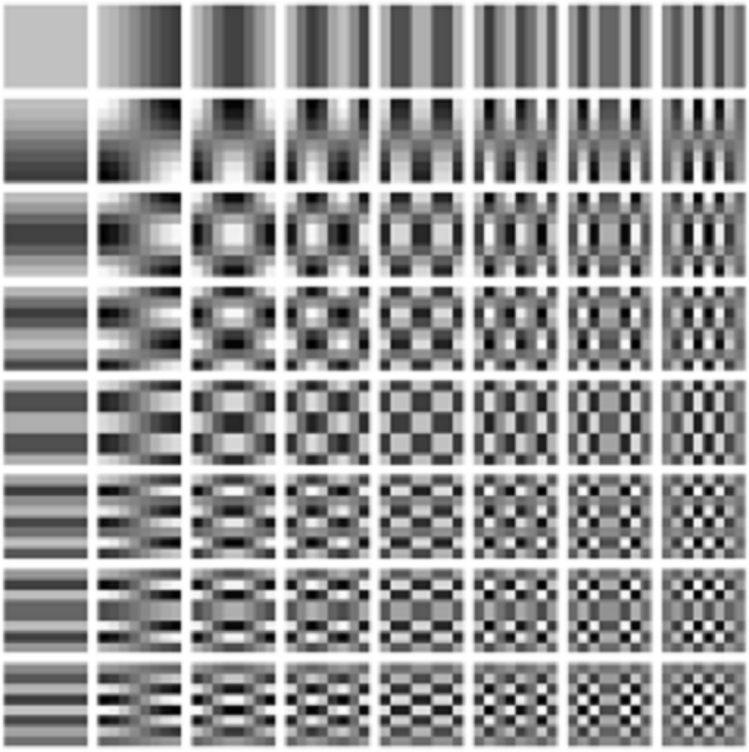


Fig. 3.7 Funciones base DCT

3.3.3 Definición de la DCT y la IDCT

La 2-D DCT e IDCT puede ser construida por productos, con los términos horizontales de la 1-D DCT (usando *u* y *x*) y los términos verticales 1-D DCT (usando *v* y *y*, donde *v* representa las frecuencias verticales y, *y* representa los desplazamientos verticales).

Transformada DCT directa bidimensional

$$F(u,v) = \alpha(u)\alpha(v)\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} f(x,y)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

Transformada IDCT Inversa

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\alpha(u)\alpha(v)F(u,v)\cos\left(\frac{(2x+1)u\pi}{2N}\right)\cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

donde:

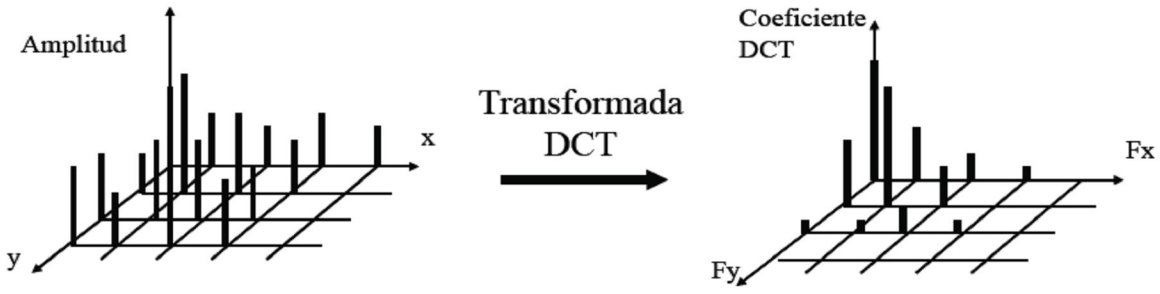
$$\alpha(u) = \frac{1}{\sqrt{2}} \quad para\ u = 0$$

$$\alpha(u) = 1 para\ u > 0$$

$$\alpha(v) = \frac{1}{\sqrt{2}} \quad para\ v = 0$$

$$\alpha(v) = 1 para\ v > 0$$

Transformada DCT directa bidimensional



Bloque en el dominio Del espacio

Bloque en el dominio De las frecuencias

3.4 Cuantización

La cuantificación nos permite reducir la precisión con la que los coeficientes de la DCT se representan cuando se convierte la DCT a una representación entera. Esto puede ser muy importante en compresión de imágenes, en donde se tiende a anular muchos coeficientes, especialmente los de altas frecuencias espaciales. Los valores cuantificados pueden ser asignados individualmente para cada coeficiente DCT, utilizando el criterio basado en la visibilidad de las funciones base. Si medimos el umbral de visibilidad de una función base y determinar la amplitud del coeficiente a partir de cual, función base es detectable por el ojo humano, podemos dividir (cuantificar) los coeficientes por ese valor (con el apropiado redondeo a valores enteros). Para la decuntificación, multiplicamos los coeficientes así obtenidos por ese valor antes de la reconstrucción, crearemos una condición en la que el ojo no podría detectar alguna diferencia entre los coeficientes DCT cuantificados y sin cuantificar. Este proceso de ponderación y truncamiento de los coeficientes de la DCT a valores enteros se denomina cuantificación, y el restablecimiento aproximado de la magnitud de los coeficientes originales de la DCT recibe el nombre de decuantificación.

El estándar define varias tablas de cuantización basadas en experimentos de percepción visual, aunque no son obligatorias las cuales se muestran a continuación.

$$Q_l = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Tabla 3.1 de Cuantizacion para Luminancia

$$Q_c = \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

Tabla 3.2 de Cuantización para Crominancia

La tabla de cuantización se aplica de la siguiente forma:

Se calcula los coeficientes de cuantizacion de la siguiente forma,

Cuantizacion:

$$redondeo\left[\frac{F(i,j)}{Q(i,j)}\right]$$

Decuantizacion:

$$F(i,j)*Q(i,j)$$

Donde $F(i,j)$ es el bloque trasformado y $Q(i,j)$ es la tabla de Luminancia o Crominancia .

Cuando un archivo es creado con el formato JPEG, el algoritmo pide un parámetro para controlar la calidad de la imagen y lo mucho que la imagen se comprime. Este parámetro, que se llama *q*, es un entero de 1 a 100, las tablas de Luminancia (tabla 3.1) y Crominancia (tabla 3.2) mostradas anteriormente tienen un grado de calidad de 50.

En este paso es donde se produce la mayor perdida de información, el análisis de error de cuantificación, hay que tener en cuenta la forma de representación de los valores en la computadora, según se haga en punto fijo o en punto flotante, y, asimismo, también tendrá influencia la forma de realizar el paso de un número cualquiera, equivalente a infinitos dígitos, según sea por truncamiento o por redondeo.

3.5 Recorrido en Zig-zag

Para aprovechar el patrón ordenado de componentes de frecuencia que se obtiene a partir de la DCT se utiliza el recorrido de Zig_zag, para hacer un mejor ordenamiento de las componentes de frecuencia. Con dicho recorrido se logra obtener un vector en el cual las primeras componentes son las de baja frecuencia mientas que las últimas representan las componentes de alta frecuencia mostrado en la figura 3.8.Con ello se efectúa un paso previo a los algoritmos de codificación que aprovechan tal circunstancia.

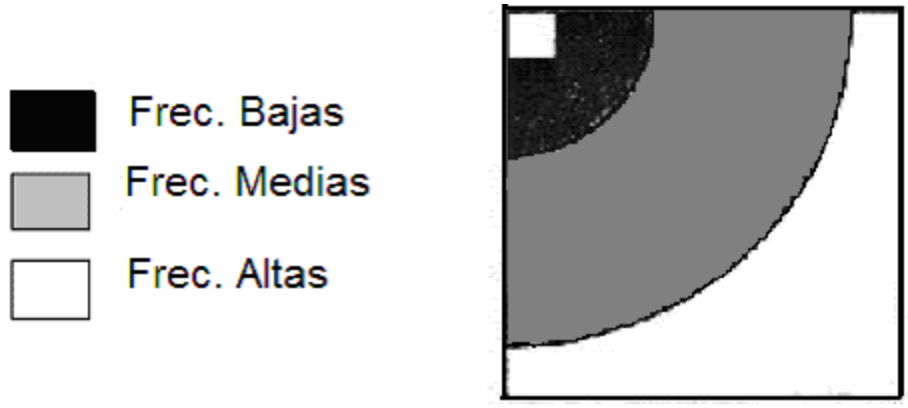


Fig. 3.8 Distribución en frecuencia de los coeficientes de la DCT bidimensional y las características de bloque que representan; el coeficiente DC es el cuadradito de la esquina superior izquierda.

Los coeficientes de alta frecuencia suelen ser pequeños, los últimos coeficientes se convierten en ceros. Se transmite un símbolo EOB (end-of-block) indicando el punto a partir del cual el receptor debe poner a cero los coeficientes.

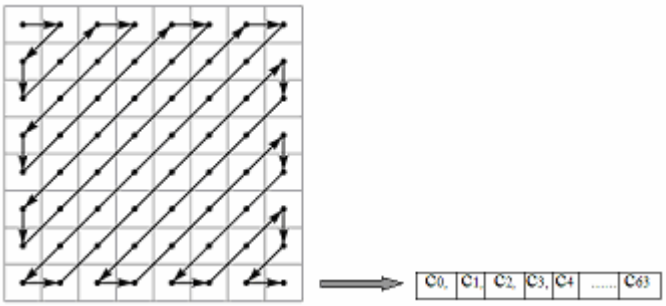


Fig.3.9 Recorrido en Zig-zag

El recorrido se realiza de forma de Zig-zag, de modo que obtenemos un vector unidimensional, el cual contiene todos los coeficientes DC y AC, tal como lo muestra la figura 3.9

3.6 Compresión de una Imagen

Cuando se muestrea y cuantifica una función bidimensional de la intensidad para crear un imagen digital, se produce una enorme cantidad de información de datos. De hecho esta cantidad de datos generados puede ser tan grande que su almacenamiento, procesamiento y su transmisión puede llegar a ser desmesurados para cualquier aplicación práctica.

La compresión de imágenes afronta el problema de la reducción de la cantidad de datos necesarios para representar una imagen digital. La base del proceso de reducción de datos consiste en:

La eliminación de los datos redundantes. Desde el punto de vista matemático, equivale a trasformar una distribución bidimensional de píxeles en un conjunto de datos estadísticos sin correlacionar. [15]

Además, hay muchas aplicaciones que utilizan la compresión con distintas expectativas: Compresión de datos, video y voz, compresión en tiempo real, etc. A su vez, cada uno de estos requiere unas características de velocidad, reversibilidad (que el algoritmo pueda se aplicado de forma reversible para obtener los datos originales), pérdida mínima de información.

Debido a esto, se ha elegido una clasificación muy general tomando como características de división la reversibilidad de algoritmo. Así, los algoritmos de compresión se pueden dividir en dos tipos:

3.6.1.1 Compresores Lossless o sin pérdida, en el sentido de que guardan absolutamente toda la información original. Se utilizan para la compresión de datos, en los que no se pueden dar perdida de información.

3.6.1.2 Compresores Lossy o con perdida, la compresión hace que se pierda información de la fuente original. Sin embargo, esta perdida es insignificante en comparación con la ganancia en compresión. Se utiliza sobre todo en imágenes y sonido, donde se puede "engañar" a los sentidos, donde la perdida de calidad a penas es percibida (pero ocasiona una razón de compresión mucho mayor).

3.7 Redundancia

Las técnicas de compresión son posibles porque todo conjunto de datos normalmente contiene redundancias. En el caso de una imagen dicha redundancia existe en forma de pautas de repetición y otras formas de información de brillo común entre varios píxels de la imagen. El objetivo de la compresión de la imagen reside en caracterizar estas redundancias y codificarlas de una forma distinta que requiera menos datos que la original.

Existen tres tipos de redundancia en las imágenes digitales:

Redundancia Espacial: Se debe a la correlación existente entre los píxeles adyacentes de una imagen.

Redundancia Espectrales: Se presenta debido a la correlación existe entre los píxeles asociados a la bandas espectrales de las imágenes.

Redundancia Codificación: Se presenta cuando la escala de grises de una imagen, se encuentra codificada de tal manera que se emplean más símbolos que los estrictamente necesarios para representar cada uno de los niveles de grises.

Redundancia Psicovisual: El ojo humano responde con diferente sensibilidad a la información visual que recibe. La información a la que es menos sensible se puede descartar sin afectar a la percepción de la imagen. Se suprime así lo que se conoce como redundancia visual. La eliminación de la redundancia esta relacionada con la cuantificación de la información, lo que con lleva una pérdida de información irreversible.

3.8 Codificación Huffman

Es una de las técnicas de entropía mas importante, se basa en la construcción de un árbol que representa la codificación de los mensajes de la fuente, de manera que los nodos hojas contienen cada uno de los mensajes emitidos por la fuente. El caso más simple, el alfabeto de salida en el que se realiza la codificación es binario. Esto quiere decir que de cada nodo partirán dos ramas, una para el 0 y otra para el 1. El código para cada mensaje se construye siguiendo el camino desde el nodo raíz hasta la hoja que representa el mensaje.

Su principal característica es que a los mensajes resultantes del algoritmo se le asignan longitudes de código variables, con respecto a la probabilidad de aparición de cada mensaje, es decir, el mensaje que mas aparezca tendrá una codificación mas corta, con lo que se ahorrará espacio en la transmisión.

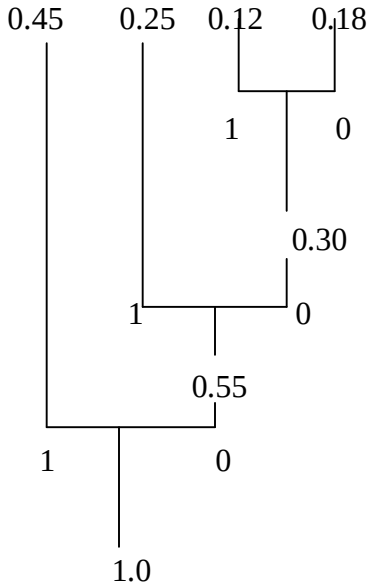
Huffman recoge los dos nodos con menor probabilidad del árbol. Construye entonces un nodo padre de ambos y se le asigna la probabilidad sumando ambos hijos. Este proceso hace que el árbol crezca y los nodos con menor probabilidad queden al fondo. Además, es óptimo y alcanza la máxima eficiencia cuando las probabilidades de los mensajes son potencias exactas de dos.

Por ejemplo: podemos considerar cuatro valores dentro de la fuente A con sus siguientes probabilidades de ocurrencia.

s1 = 0.45 s3 = 0.12
s2 = 0.25 s.4 = 0.18

Simbolo	s1	s2	s3	s4
Código Huffman	1	01	001	000

Valores de la ocurrencia De la probabilidad



3.9 Codificación de los coeficientes DC

Esta codificación se realiza mediante el método, DPCM (modulación por código de pulso, de tipo diferencial) saca ventaja de la redundancia entre muestras sucesivas, que en este caso se aplica a los coeficientes DC obtenidos en la etapa de la transformada DCT.

Como la diferencia entre muestras es menor que el valor de los coeficientes, individualmente, se ocupan menos bits para codificar la diferencia entre muestras vecinas, por lo tanto, hay un menor espectro.

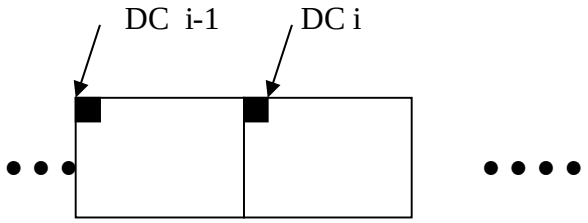


Fig. 3.10 Coeficientes DC

Los coeficientes DC varían ligeramente entre bloques sucesivos, figura 3.10. La codificación diferencial de los coeficientes DC, codificación DPCM, explota esta propiedad.

Esta técnica codifica la diferencia entre el coeficiente DC cuantificado del bloque actual y el coeficiente DC cuantificado del bloque anterior. Por ejemplo, considerando el k-ésimo bloque cuantificado, la fórmula para el código de DPCM está dada por:

$$DPCMCODE_k = C_k(0,0) - C_{k-1}(0,0)$$

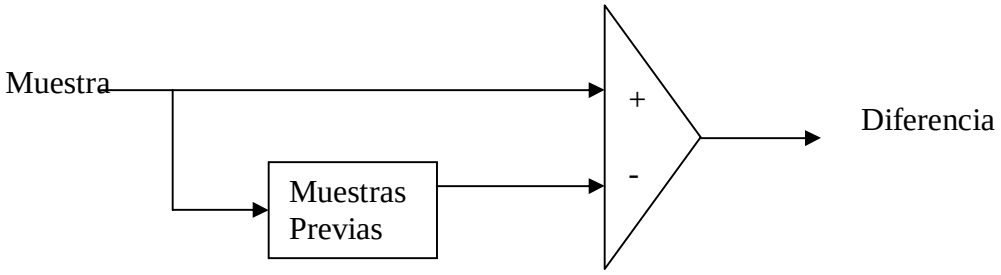


Fig. 3.11 Modelo de Codificación DPCM

La inversa del DPCM calcula el coeficiente DC actual sumando el código DPCM actual con el código del coeficiente DC previo, según la ecuación:

$$C_k(0,0) = DPCM_k + C_{k-1}(0,0)$$

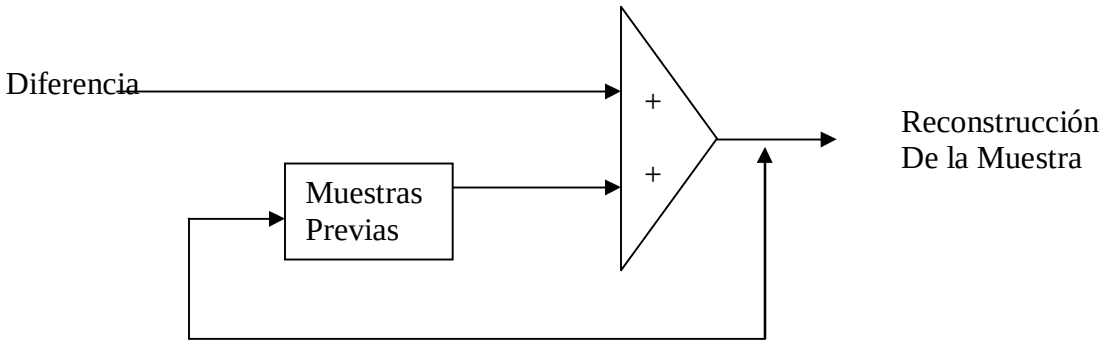


Fig. 3.12 Modelo de Decodificación DPCM

El proceso de codificación de los coeficientes Diff (DC), es la siguiente:

1. Determinar la cantidad (SSSS), bits necesarios para representar Diff valores de los coeficientes DC.

2. Utilizar la tabla 3.3 para codificar (SSSS). a esto se lo conoce como codificación VLC.

3. Si Diff ≥ 0, entonces tomar los (SSSS), bits menos significativos. Si Diff < 0, entonces tomar el complemento a dos de la representación de Diff - 1 y tomar los (SSSS) bits menos significativos.

Categoría (SSSS)	Diferencias DPCM (DIFF valores)
0	0
1	-1, 1
2	-3, -2, 2, 3
3	-7,..-4, 4,..7
4	-15,..-8, 8,..15
5	-31,..-16, 16,..31
6	-63,..-32, 32,..32
7	-127,..-64, 64,..127
8	-255,..-128, 128,..255
9	-511,..-256, 256,..511
10	-1023,..-512, 512,..1023
11	-2047,.. 1024, 1024,.. 2047

Tabla 3.3: Categoría de diferencias de magnitudes para Codificar coeficientes DC [4]

Categoría	Código
0	010
1	011
2	100
3	00
4	101
5	110
6	1110
7	11110
8	111110
9	1111110
10	11111110
11	111111110

Tabla 3.4: Categoría DC para códigos Huffman [4]

Después de ser aplicada la DPCM a los coeficientes resultantes de la DCT, éstos pasan por un codificador "ENTRÓPICO" que le asigna un determinado número de bits con un largo variable. Luego la información queda lista para ser almacenada o transmitida para su posterior recuperación. Una vez que tenemos las etiquetas no nulas reordenadas, el siguiente paso es codificarlas La codificación es distinta para coeficientes DC y AC.

3.10 Codificación de los coeficientes AC

Una vez que se dispone del arreglo resultante del recorrido en zig-zag, en series de pares longitud, valor (RRRRSSSS). La componente longitud indica la cantidad de coeficientes nulos y la componente valor indica el siguiente coeficiente no nulo en el arreglo.

RS = binario ‘RRRRSSSS’

Los cuatro bits mas significativos ‘RRRR’, representan la cantidad de ceros que precede al valor no nulo, y ‘SSSS’, identifica a la categoría que es la cantidad de bits necesarios para representar el componente, valor no nulo en la secuencia Zigzag, figura 3.13. Iniciando en el índice J (0 < J < 63)

ZZ (J + 1) ,……. ,ZZ (63)

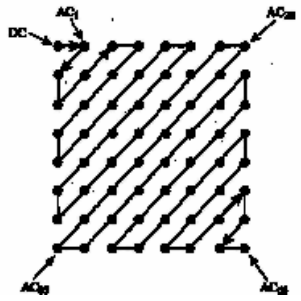


Fig. 3.13 Recorrido en Zig-zag

Los coeficientes AC se codifican de modo similar a los coeficientes Diff, pero teniendo en cuenta que los coeficientes AC estarán representados por pares (longitud, valor).

Proceso de codificación

1. Teniendo en cuenta el par (longitud, valor), determinar el par (RRRR, SSSS), donde longitud es la cantidad de ceros que precede al valor no nulo y (SSSS) es la cantidad de bits necesarios para representar la componente valor del primer par. A (SSSS) se le determina del mismo modo que en el paso 1 de la codificación de los coeficientes DC.
2. Mapear la componente valor en una secuencia de (SSSS) bits, del mismo modo que en el paso 3 de la codificación de los coeficientes Diff, ver tabla 3.5.
3. Utilizar una tabla de códigos Huffman para coeficientes AC, para mapear el par (RRRR, SSSS) en un código Huffman, ver tabla 3.6.
4. Concatenar los códigos para formar una cadena de bits que representa la codificación del par original (RRRR SSSS).

Categoría (SSSS)	AC Coeficientes ZZ (k)
1	-1, 1
2	-3, -2, 2, 3
3	-7,..-4, 4,.. 7
4	-15,..-8, 8,.. 15
5	-31,..-16, 16,... 31
6	-63,..-32, 32,... 32
7	-127,..-64, 64,.. 127
8	-255,..-128, 128,… 255
9	-511,..-256, 256,.. 511
10	-1023,..-512, 512,… 1023

Tabla 3.5: Categoría de diferencias de magnitudes para Codificar coeficientes AC [4]

Run	Category	Code	Run	Category	Code	Run	Category	Code
0	0	1010	6	0		12	0	
0	1	00	6	1	1111011	12	1	1111111010
0	2	01	6	2	11111111000	12	2	1111111111011010
0	3	100	6	3	1111111110100111	12	3	1111111111011011
0	4	1011	6	4	1111111110101000	12	4	1111111111011100
0	5	11010	6	5	1111111110101001	12	5	1111111111011101
0	6	111000	6	6	1111111110101010	12	6	1111111111011110
0	7	1111000	6	7	1111111110101011	12	7	1111111111011111
0	8	1111110110	6	8	1111111110101100	12	8	1111111111100000
0	9	1111111110000010	6	9	1111111110101101	12	9	1111111111100001
0	10	1111111110000011	6	10	1111111110101110	12	10	1111111111100010
1	0		7	0		13	0	
1	1	1100	7	1	1111001	13	1	1111111010
1	2	111001	7	2	11111111001	13	2	1111111111100011
1	3	1111001	7	3	1111111110101111	13	3	1111111111100100
1	4	111110110	7	4	1111111110110000	13	4	1111111111100101
1	5	11111110110	7	5	1111111110110001	13	5	1111111111100110
1	6	1111111110000100	7	6	1111111110110010	13	6	1111111111100111
1	7	1111111110000101	7	7	1111111110110011	13	7	1111111111101000
1	8	1111111110000110	7	8	1111111110110100	13	8	1111111111101001
1	9	1111111110000111	7	9	1111111110110101	13	9	1111111111101010
1	10	1111111110001000	7	10	1111111110110110	13	10	1111111111101011
2	0		8	0		14	0	
2	1	11011	8	1	11111010	14	1	11111110110
2	2	11111000	8	2	1111111111000000	14	2	1111111111101100
2	3	1111110111	8	3	1111111110110111	14	3	1111111111101101
2	4	1111111110001001	8	4	1111111110111000	14	4	1111111111101110
2	5	1111111110001010	8	5	1111111110111001	14	5	1111111111101111
2	6	1111111110001011	8	6	1111111110111010	14	6	1111111111110000
2	7	1111111110001100	8	7	1111111110111011	14	7	1111111111110001
2	8	1111111110001101	8	8	1111111110111100	14	8	1111111111110010
2	9	1111111110001110	8	9	1111111110111101	14	9	1111111111110011
2	10	1111111110001111	8	10	1111111110111110	14	10	1111111111110100
3	0		9	0		15	0	11111111110111
3	1	111010	9	1	111111000	15	1	1111111111110101
3	2	111110111	9	2	1111111110111111	15	2	1111111111110110
3	3	11111110111	9	3	1111111111000000	15	3	1111111111110111
3	4	1111111110010000	9	4	1111111111000001	15	4	1111111111111000
3	5	1111111110010001	9	5	1111111111000010	15	5	1111111111111001
3	6	1111111110010010	9	6	1111111111000011	15	6	1111111111111010
3	7	1111111110010011	9	7	1111111111000100	15	7	1111111111111011
3	8	1111111110010100	9	8	1111111111000101	15	8	1111111111111100
3	9	1111111110010101	9	9	1111111111000110	15	9	1111111111111101
3	10	1111111110010110	9	10	1111111111000111	15	10	1111111111111110
4	0		10	0				
4	1	111011	10	1	111111001			
4	2	1111111000	10	2	1111111111001000			
4	3	1111111110010111	10	3	1111111111001001			
4	4	1111111110011000	10	4	1111111111001010			
4	5	1111111110011001	10	5	1111111111001011			
4	6	1111111110011010	10	6	1111111111001100			
4	7	1111111110011011	10	7	1111111111001101			
4	8	1111111110011100	10	8	1111111111001110			
4	9	1111111110011101	10	9	1111111111001111			
4	10	1111111110011110	10	10	1111111111010000			
5	0		11	0				
5	1	1111010	11	1	111111010			
5	2	1111111001	11	2	1111111111010001			
5	3	1111111110011111	11	3	1111111111010010			
5	4	1111111110100000	11	4	1111111111010011			
5	5	1111111110100001	11	5	1111111111010100			
5	6	1111111110100010	11	6	1111111111010101			
5	7	1111111110100011	11	7	1111111111010110			
5	8	1111111110100100	11	8	1111111111010111			
5	9	1111111110100101	11	9	1111111111011000			
5	10	1111111110100110	11	10	1111111111011001			

Tabla 3.6, Códigos Huffman para Coeficientes AC

Los métodos de esteganografia en el domino de las frecuencias, tiene una secuencia de pasos mas elaborados para insertar la información, en este capítulo se describe la DCT, la cual utilizamos para decorrelacionar los datos de la imagen, con ello se obtiene una compactación de energía, visto de otra forma se compacta un numero mayor de información en un menor numero de datos o byte , aclarando que a través de los datos se trasmite la información.

Otro punto que se trato en este capitulo, es la Cuantificación la cual es muy importante en la compresión de imagen, en donde se tiende a anular muchos coeficientes especialmente los de altas frecuencias espaciales. Hasta este momento no se a insertado la información solamente se prepararon los coeficientes AC para poder efectuar el proceso de inserción de información.

3.11 Método del dominio de las frecuencias

La base de las técnicas en el dominio de las frecuencias es el teorema de la convolución. Sea $g(u,v)$ una imagen formada por la convolucion de una imagen $f(x,y)$ y un operador (filtro) lineal invariante de posición $h(x,y)$, es decir

$$g(u,v) = h(u,v) * f(u,v)$$

Entonces y por el teorema de la convolución, se cumple la siguiente relación en el dominio de las frecuencias:

$$G(u,v) = H(u,v)F(u,v)$$

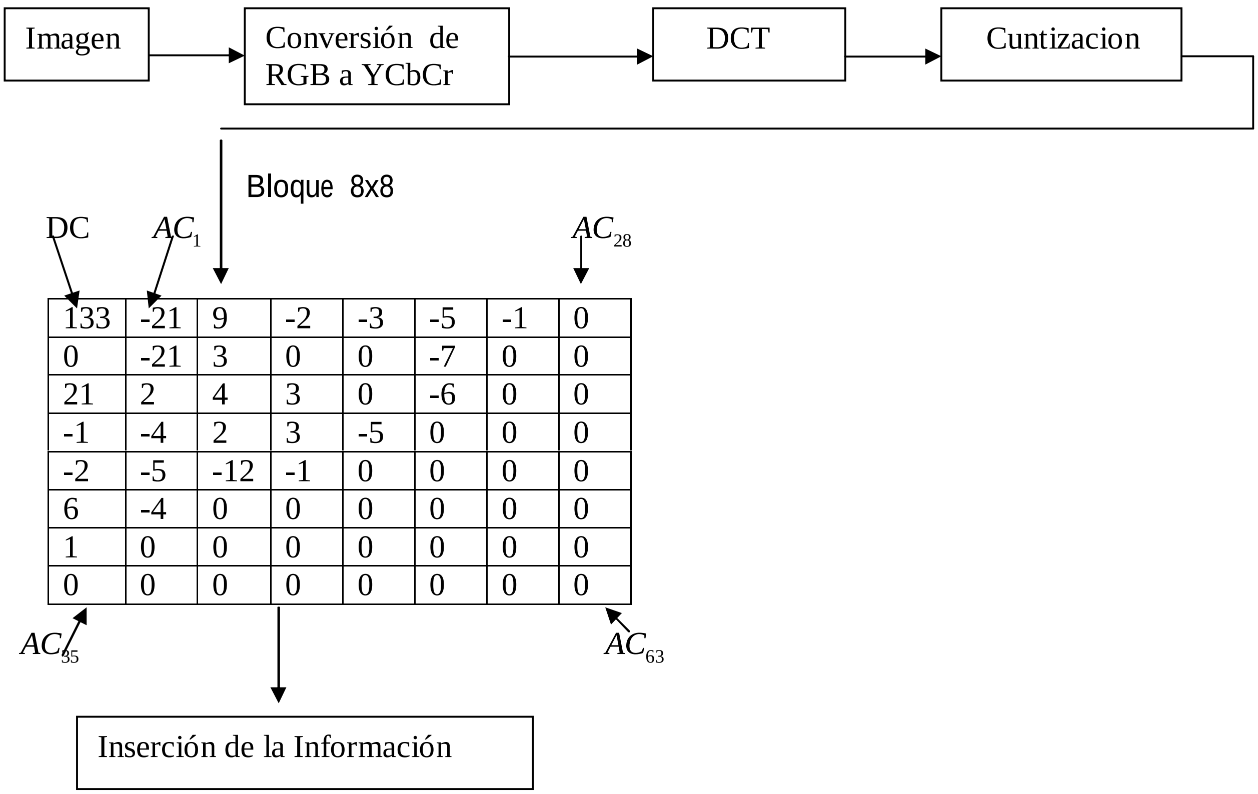
Donde *G*, *H*, y *F* son respectivamente las trasformadas de Fourier de *g*, *h* y *f*. En la terminología de teoría de sistemas lineales, la transformación $H(u,v)$ se denomina la función de transferencia del proceso. En óptica $H(u,v)$ se denomina la función de transferencia óptica, y su magnitud es la modulación de la función de transferencia. [6]

3.11.1 Esteganografía en el dominio de la DCT

Cabe destacar que en los últimos años, se ha dedicado un gran esfuerzo por el desarrollo de técnicas que operan en el dominio de las frecuencias, con lo que es posible que el estegomedio más idóneo y más comunes sean las imágenes JPEG, usadas para ocultar información. Se han publicado diferentes técnicas para aprovechar los coeficientes cuantificados (DCTs) de estas imágenes, para ocultar información de una forma más robusta que operando con una simple técnica LSB.

La inserción de la información en esta clase de métodos, no se realiza directamente como en el caso de los métodos, del dominio espacial. En este caso son varios los pasos que se tienen que seguir, los cuales son: dada una imagen convertirla de modelo de color RGB al modelo YCbCr, aplicarle la transformada de coseno (DCT), seguida de la cuantización, una vez tendiendo las matrices de coeficiente cuantificados DC, AC, en este paso es donde se efectúa la inserción de la información, en los coeficientes DC y AC, descartando los coeficientes nulos.

El proceso de codificación de JPEG es muy elaborado, como planteamos anteriormente, el proceso inicia con la conversión del modelo de color de RGB a YCbCr, posteriormente la aplicación de la DCT a los bloques de 8x8, después viene la cuantificación, y la codificación de los coeficientes DC y AC, pudiendo así obtener la imagen con formato JPEG. La esteganografía en el domino de las frecuencias, sus métodos no se aplican directamente a los bits de la imagen se muestra a continuación en forma esquemática en la figura 3.14.



133	-21	9	-2	-3	-5	-1	0
0	-21	3	0	0	-7	0	0
21	2	4	3	0	-6	0	0
-1	-4	2	3	-5	0	0	0
-2	-5	-12	-1	0	0	0	0
6	-4	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Fig. 3.14 Proceso de la inserción de la información

Recorrido en Zig-zag

133, 21, 0, 21, 21, 9, -2, 3, 2, -1, -2, -4, 4, 0, -3, -5,0, 3, 2,-5, 6, 1,-4,-12, 3,0,-7,-1,0,0,6,5,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

Una vez realizado el recorrido en zig-zag los coeficientes AC, están ordenados conforme a sus frecuencias. Todos los coeficientes iguales a 0 son descartados para insertar la información, el coeficiente DC que en este caso es 133 también es descartado.

133,21, 21, 21, 9, -2, 3, 2, -1, -2, -4, 4, -3, -5, 3, 2,-5, 6, 1,-4,-12, 3,-7,-1-6,-5,1

Obteniendo los coeficientes, en los cuales la información se va a insertar, en los bit menos significativos LSB

3.11.2 Proceso de Inserción del Mensaje

1. Entrada: password, bloques de la DCT después de haber pasado por la cuantización.

2. Inicializamos las permutaciones de coeficientes de la DCT .

3. Calculamos el mensaje secreto con, la matriz para su codificación.
   a) Con el arreglo de la de la función suma (generamos un valor suma con k bit-lugares)
      G←{n coeficientes AC no nulos}
      s←k-bit funcion suma de LSB en G
   b) Agregamos el siguiente bit del mensaje con el valor determinado por la suma (bit a bit con un xor).
      s←s⊕k-bit bloque del mensaje.
   c) Si la suma es cero, el buffer es inalterado. De otro modo si la suma del buffer es diferente de cero,el valor de elemento es decrementado.
   d) Examinaos para la reducción, si se produce un cero ajustamos el arreglo, de otro modo elegimos otro coeficientes.

6. Continúa la compresión (codificación Huffman).

3.11.3 Proceso de extracción del mensaje

1. Entrada: password, bloques de la DCT después de haber pasado por la cuantización.

2. Inicializamos las permutaciones de los coeficientes de la DCT.

3. Calculamos el mensaje secreto.
   a) Con el arreglo de función suma (generamos un valor suma con k bit-lugares)
      G←{n coeficientes AC no nulos}
      s←k-bit la función suma de LSB en G
      x←s⊕k-bit bloque del mensaje.
         donde x contendrá el mensaje secreto.

Capítulo 4

Encriptación del Mensaje

El objetivo principal de la utilización de encriptación, es brindarle al sistema una mayor seguridad. Con ello el mensaje contará con una codificación independiente con respecto a la imagen, si en dado caso que la imagen fuera detectada, el mensaje tendrá que ser sometido a una decodificación diferente, para poder interpretar su información. Dicho proceso se realiza con el algoritmo RSA, el cual es uno de los más importantes algoritmos de encriptación, y ha demostrado ser viable para encriptación de llave pública. Mucha de la teoría de criptosistemas llave pública esta basada en teoría números.

4.1 Algoritmo de Euclides

Sean *a, b* números enteros, con *a* distinto de cero decimos que ***a divide a* b** si y solo si existe un entero m tal que $a*m=b$ y se denota como *a/b*

Propiedades:

- Si $a|1$, entonces $a = \pm 1$.
- Si $a|b$ y $b|a$, entonces $a = \pm b$.
- Cualquier $b \neq 0$ divide a 0.
- Si $b|g$ y $b|h$, entoces b|($mg + nh$) para algún entero, *m* y *n.*

Sen *a, b* números enteros, se define ***el máximo común divisor de* a, b** como el entero positivo c que cumple

1. *c/ a* y *c/b*
2. Si hay un entero *r* tal que *r/a* y *r/b* entonces ***r/c***

Y se denota así mcd(a,b)=c, así el mcd cumple con las siguientes propiedades:

1. mcd(a,b)=mcd(|a|,|b|)
2. mcd (a,0)=|a| , para todo a distinto de cero

Una de las técnicas básicas de la teoría de números, es el algoritmo de Euclides, el cual es un simple procedimiento para determinar el máximo común divisor de dos enteros positivos.

Algoritmo

Euclides(a, b)

1. A ← a; B← b
2. si B = 0 regresa a = mcd(a, b)
3. R = A mod B
4. A←B
5. B←R
6. salto 2

El algoritmo tiene la siguiente secuencia:

$$A_1 = B_1 \times Q_1 + R_1$$
$$A_2 = B_2 \times Q_2 + R_2$$
$$A_3 = B_3 \times Q_3 + R_3$$
$$A_4 = B_4 \times Q_4 + R_4$$

Decimos que a, b números enteros son **coprimos o primos relativos** si mcd(a,b)=1

4.2 Aritmética modular

Si *a* es un entero y *n* es un entero positivo se define, ***a mod n*** como es el resto, cuando *a* es dividido por *n*, esto es

$$11 \bmod 7 = 4; \qquad -11 \bmod 7 = 3$$

A continuación se muestra propiedades de aritmética modular:

$$1.\left[(a \bmod n) + (b \bmod n)\right] \bmod n = (a+b) \bmod n$$
$$2.\left[(a \bmod n) - (b \bmod n)n\right] \bmod n = (a-b) \bmod n$$
$$3.\left[(a \bmod n)\, x\, (b \bmod n)\right] \bmod n = (a*b) \bmod n$$

Definiendo algunos un ejemplo para las anteriores propiedades:

$$11 \bmod 8 = 3$$
$$15 \bmod 8 = 7$$
$$[(11 \bmod 8) + (15 \bmod 8)] \bmod 8 = 10 \bmod 8 = 2$$
$$(11 + 15) \bmod 8 = 26 \bmod 8 = 2$$
$$[(11 \bmod 8) - (15 \bmod 8)] \bmod 8 = -4 \bmod 8 = 4$$
$$(11 - 15) \bmod 8 = -4 \bmod 8 = 4$$
$$[(11 \bmod 8) \times (15 \bmod 8)] \bmod 8 = 21 \bmod 8 = 5$$
$$(11 \times 15) \bmod 8 = 165 \bmod 8 = 5$$

La exponenciación se realiza por medio de multiplicaciones repetidas, como en la aritmética ordinaria. Para encontrar $11^7 \bmod 13$, se procede de la siguiente forma:

$$11^2 = 121 \equiv 4 \bmod 13$$
$$11^4 \equiv 4^2 \equiv 3 \bmod 13$$
$$11^7 \equiv 11 * 4 * 3 \equiv 132 \equiv 2 \bmod 13$$

4.3 Algoritmo Extendido de Euclides

Si mcd(m, b) = 1 entonces b tiene un inverso multiplicativo modulo m, esto es, para un entero positivos
$b < m$, existe un $b^{-1} < m$ que cumple $bb^{-1} = 1 \bmod m$, $bb^{-1} = 1$.
El algoritmo de Euclides puede extenderse a fin de que, si el mcd(m,b) es 1, el algoritmo regrese el inverso multiplicativo de *b*.

Euclides_extendido (m, b)

1. (A1, A2, A3) ← (1, 0, m); (B1, B2, B3) ←(0, 1, b)
2. si B3 = 0 regresa A3 = mcd(m, b); entonces b no tiene inverso
3. si B3 = 1 regreso B3 = mcd(m, b);entonces B2 = $b^{-1} \bmod m$
4. $Q = \left\lfloor \frac{A3}{B3} \right\rfloor$
5. (T1, T2, T3) ← (A1, QB1, A2 – QB2, A3 – QB3)
6. (A1, A2, A3) ← (B1, B2, B3)
7. (B1, B2, B3) ←(T1, T2, T3)
8. salta 2

[8]

A continuación se muestra un ejemplo del uso del Algoritmo de Euclides Extendido con los siguientes datos de entrada m=256 y b=117

Q	A1	A2	A3	B1	B2	B3
--	256	117	1	0	0	1
2	117	22	0	1	1	-2
5	22	7	1	-5	-2	11
3	7	1	-5	16	11	-35
7	1	0	16	117	-35	256

Así el inverso de b=117, b^{-1}= -35 mod 256 , esto es b^{-1}=221 , lo que significa que 117*221 mod 256 =1

4.4 Criptografía

El termino criptografía provine de griego kriptos, que significa esconder y graphien, escribir, es decir escritura oculta. La Criptografía es la ciencia que se ocupa del cifrado seguro de mensajes.

4.4.1 Criptografía de clave sencilla o de clave secreta

Criptografía simétrica, que resulta útil en muchos casos, aunque tiene limitaciones significativas. Los algoritmos simétricos, o de clave secreta, se caracterizan por ser altamente eficientes (en relación al tamaño de su clave) y robustos. Se les llama así porque se emplean la misma clave para cifrar y para descifrar. Se basan en el uso de claves secretas que previamente hay que intercambiar mediante canales seguros, con los riesgos que ello supone. Todas las partes deben conocerse y confiar totalmente la una en la otra. [9]

4.4.2 Clave pública o criptografía asimétrica.

Al contrario que los anteriores, los algoritmos asimétricos tienen claves distintas para cifrado y descifrado. Por ello, también se les llama algoritmos de clave pública. Permiten eliminar el gran inconveniente de cómo hacer llegar al remitente la clave de cifrado. En el caso de los algoritmos asimétricos se usan una clave pública (para cifrar) y una secreta (para descifrar).

La primera se publica en un tipo de directorio al que el público en general tiene acceso, mientras que la privada se mantiene en secreto. Las dos claves funcionan conjuntamente como un curioso dúo. De esa manera, una intercepción de la clave pública es inútil para descifrar un mensaje, puesto que para ello se requiere la clave secreta. Cualquier tipo de datos o información que una de las claves cierre, sólo podrá abrirse con la otra.

De entre todos los algoritmos asimétricos se eligió, *RSA* quizá por que sea el mas sencillo de comprender e implementar y se le tiene como uno de los algoritmos asimétricos más seguros. Como ya se ha dicho, sus claves sirven indistintamente tanto para codificar como para autentificar. Debe su nombre a sus tres inventores: Ronald Rivest, Adi Shamir y Leonard Adleman, y estuvo bajo patente de los Laboratorios *RSA* hasta el 20 de septiembre de 2000. RSA se basa en la dificultad para factorizar grandes números. Las claves pública y privada se calculan a partir de un número que se obtiene como producto de dos primos grandes.

4.5.1 Algoritmo de RSA

1. Seleccionamos p y q donde p y q son n primos, $p \neq q$
2. Calculamos $n = p \times q$
3. Calculamos $\phi(n) = (p-1)(q-1)$
4. seleccionamos un entero e $mcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
5. Calculamos d $d \equiv e^{-1} \pmod{\phi(n)}$

Calve Pública $PU = \{e, n\}$

Clave Privada $PR = \{d, n\}$

4.5.2 Cifrado del mensaje

Si el usuario B desea enviar un mensaje m a otro usuario A, deberá realizar las siguientes operaciones

1. B obtener la clave pública de A, (n_A, e_A)
3. B calcula el valor del criptograma $c = m^{eA} (\mathrm{mod}\, n_A)$, y de este modo se envía a A.

4.5.3 Descifrado del Mensaje

Una vez que el usuario A recibe el criptograma c, para su recuperación el mensaje original m realiza lo siguiente:

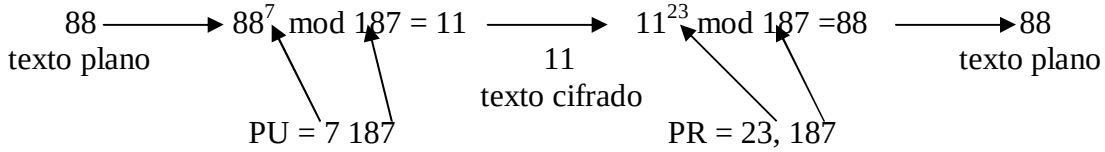
1. utilizar su clave privada d_A y calcular

$$c^{d_A} = (m^{e_A})^{d_A} (\mathrm{mod}\, n_A) = m^{e_A d_A} (\mathrm{mod}\, n_A) \equiv m (\mathrm{mod}\, n_A)$$

Ahora veamos como codificar el mensaje m=88 usando RSA:

1. Seleccionamos dos números primos, p = 17 y q = 11.
2. Calculamos n = pq = 17 x 11 = 187.
3. Calculamos $\phi(n) = (p- 1)(q -1) = 16 \times 10 = 160$.
4. Seccionamos *e* tal que e es relativamente a $\phi(n) = 160$ y menor que $\phi(n)$, escogemos *e* = 7.
5. Determinamos d tal que *d*e* mod 160=1 y *d* < 160, usando el algoritmo de Euclides Extendido, se tiene d=23 y 23*7=161 mod 160=1.

Por tanto la llave pública es PU={7,187} y la llave privada es PR ={23,187}



Para encriptar se usa la llave pública e =7 y se calcula $C = 88^7 \bmod 187$ usando las propiedades de aritmética modular.

$88^7 \bmod 187 = [(88^4 \bmod 187) \times (88^2 \bmod 187) \times (88^1 \bmod 187)] \bmod 187$

$88^1 \bmod 187 = 88$

$88^2 \bmod 187 = 7744 \bmod 187 = 77$

$88^4 \bmod 187 = 59{,}969{,}536 \bmod 187 = 132$

$88^7 \bmod 187 = (88 \times 77 \times 132) \bmod 187 = 894{,}432 \bmod 187 = 11$

Para desencriptar se necesita la llave privada d = 23 y ahora se calcula $M = 11^{23} \bmod 187$

$11^{23} \bmod 187 = [(11^1 \bmod 187) \times (11^2 \bmod 187) \times (11^4 \bmod 187) \times (11^8 \bmod 187) \times (11^8 \bmod 187)] \bmod 187$

$11^1 \bmod 187 = 11$

$11^2 \bmod 187 = 121$

$11^4 \bmod 187 = 14{,}641 \bmod 187 = 55$

$11^8 \bmod 187 = 214{,}358{,}881 \bmod 187 = 33$

$11^{23} \bmod 187 = (11 \times 121 \times 55 \times 33 \times 33) \bmod 187 = 79{,}720{,}245 \bmod 187 = 8$

En la figura 4.1 se describe de manera general el proceso de codificación del mensaje.

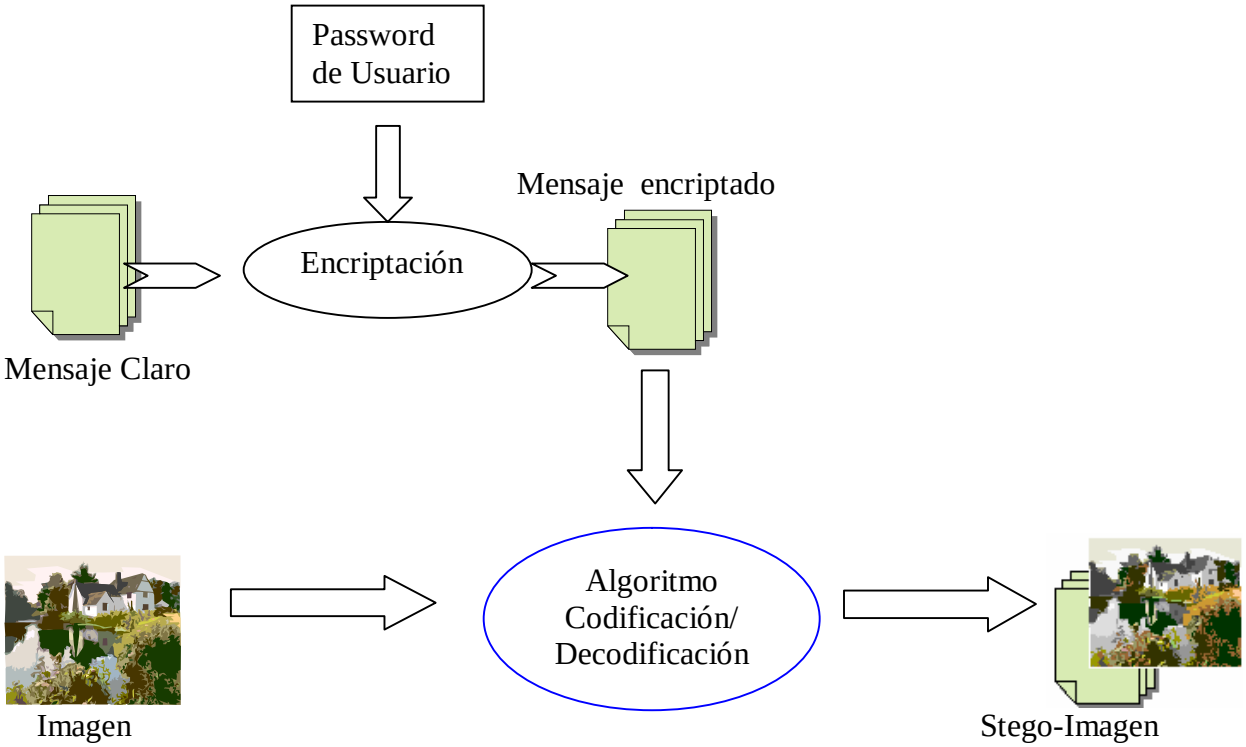


Fig. 4.1 Diagrama general de Esteganografía con Encriptación

Capítulo 5

Análisis, Diseño e Implementación del Sistema

5.1 Análisis del Sistema

Este trabajo tiene como objetivo implementar, técnicas de esteganografía en imágenes digitales, utilizando el método de la Transformada Discreta de Coseno (DCT), el cual trabaja en el domino de las frecuencias y, el método del Bit Menos Significativo (LSB), que trabaja en el domino espacial. Los cuales consisten en insertar información en la imagen digital mediante la realización de modificaciones sobre la misma, sin afectar su calidad, la información que se insertara tendrá un proceso, de encriptación, si el usuario a si lo requiere, utilizando RSA.

En esta primera etapa de análisis consideramos a nuestro sistema como una "caja negra" reaccionando a las peticiones del usuario. Tenemos un solo usuario que interactúa con el sistema, para la inserción de texto en imágenes.

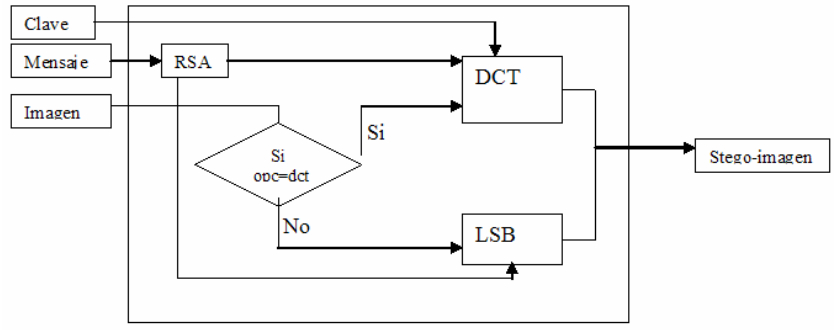


Fig. 5 Prototipo del Sistema

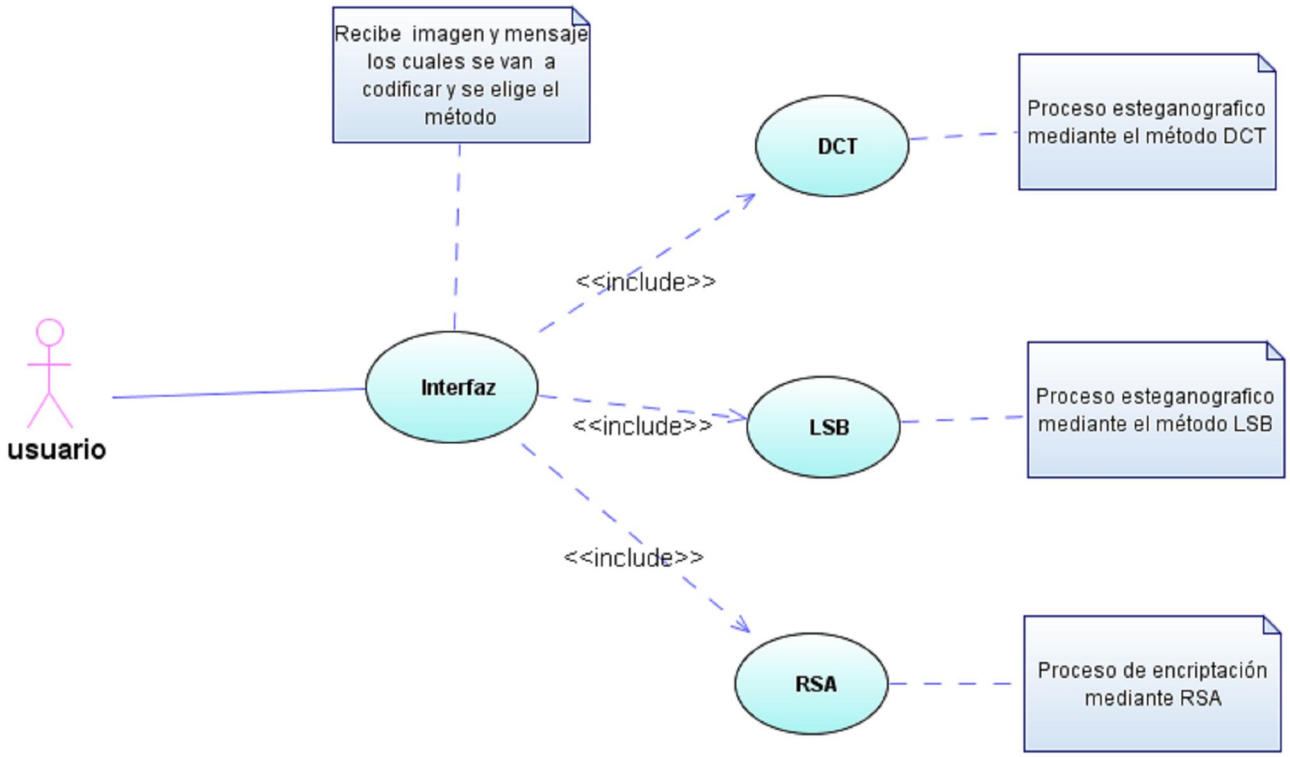
Como se muestra en la figura 5, se describe el proceso general de codificación del sistema, para una imagen. El sistema contará con un interfaz amigable, que le permita al usuario realizar, las siguientes operaciones:

- Ingresar imagen con formato JPG, abrir y guardar la imagen.
- Ingresar mensaje, y generar claves públicas y privadas, abrir y guardar el archivo, con formato txt.
- Ingresar clave para el método DCT.

5.2 Diseño del Sistema

Debido a las facilidades que ofrece UML para la realización de un Análisis y Diseño de un sistema de software, se tomara como herramienta principal. Se utiliza el modelo en cascada para el desarrollo del sistema, este es el más básico de todos los modelos.

5.2.1 Diagrama de Casos de Usos

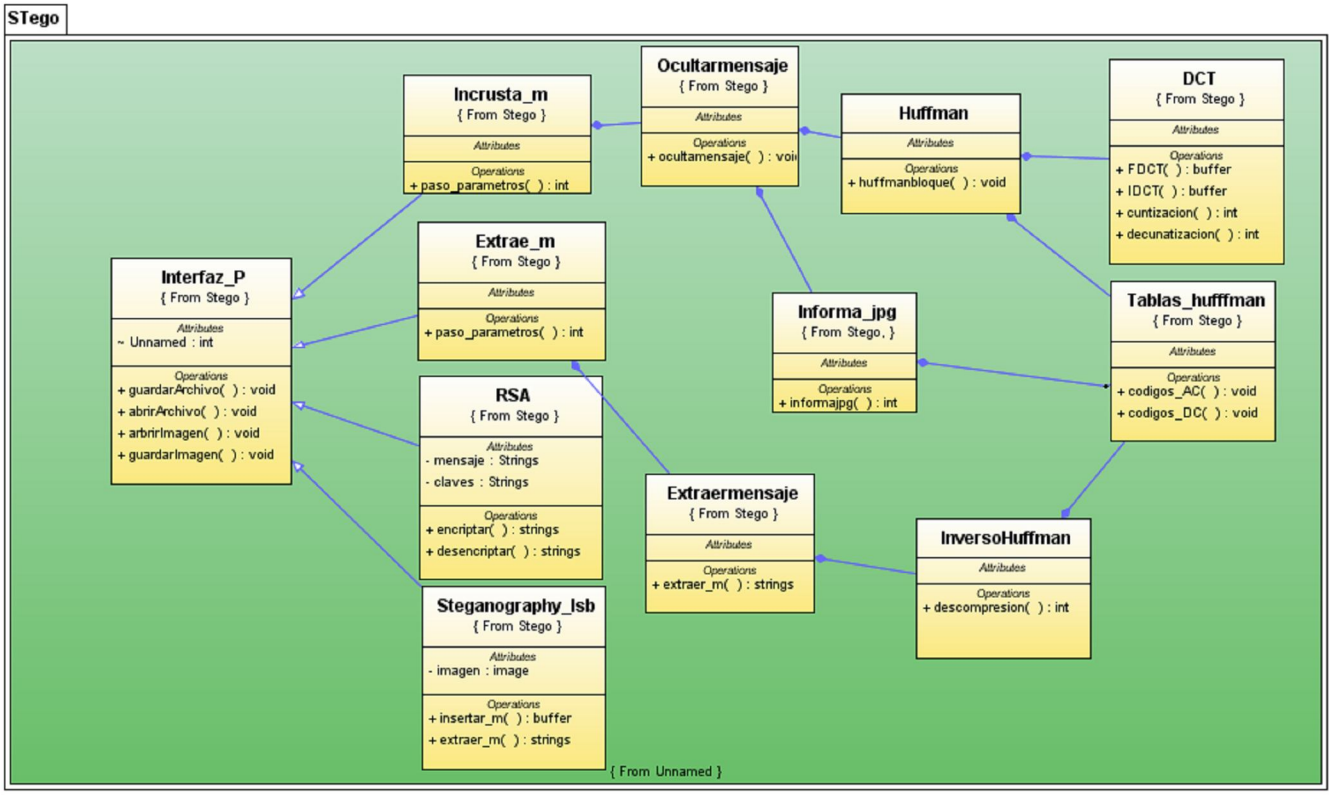


Descripción

El usuario selecciona el método, con el cual se insertará el mensaje en la imagen, ya sea DCT o LSB, si el usuario quiere que el mensaje se codifique mediante encriptación, se tendrá que hacer antes de insertarlo en la imagen. Este proceso se realiza con el método RSA.

5.2.2 Diagrama de Clases

Paquete Stego



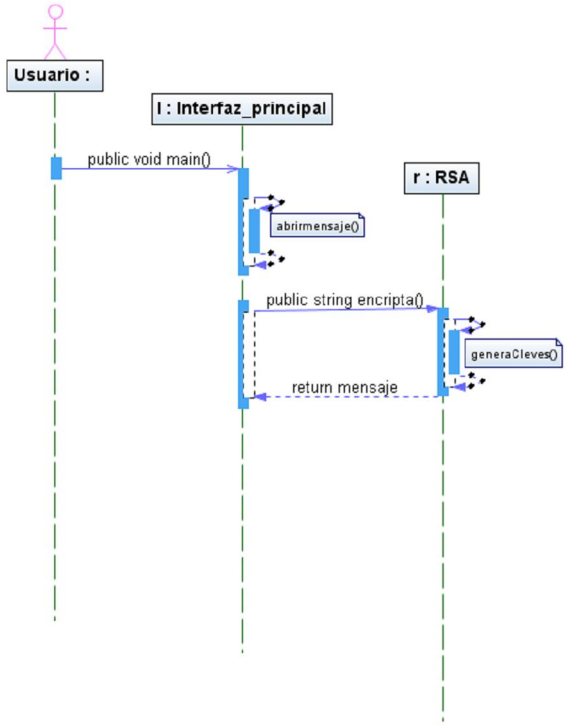
Descripción

Se muestra las relaciones principales de la clase Interfaz_P, pertenecientes al paquete de clases Stego. A su vez asocia las clases incrusta_m, ocultamensaje, Huffman, DCT, Tablashuffman, informajpg, que se encargan de insertar el mensaje en la imagen con el método DCT, las clases para extraer el texto, son Extrae_m Extraermensaje, InversoHuffman, IDCT, Tablas_huffman. Con respecto a método de LSB que utiliza la clase Steganography para su codificación y decodificación. El método de encriptación RSA utiliza la clase RSA que se encarga de la codificación del mensaje.

5.2.3 Diagramas de secuencias

Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada método de la clase.

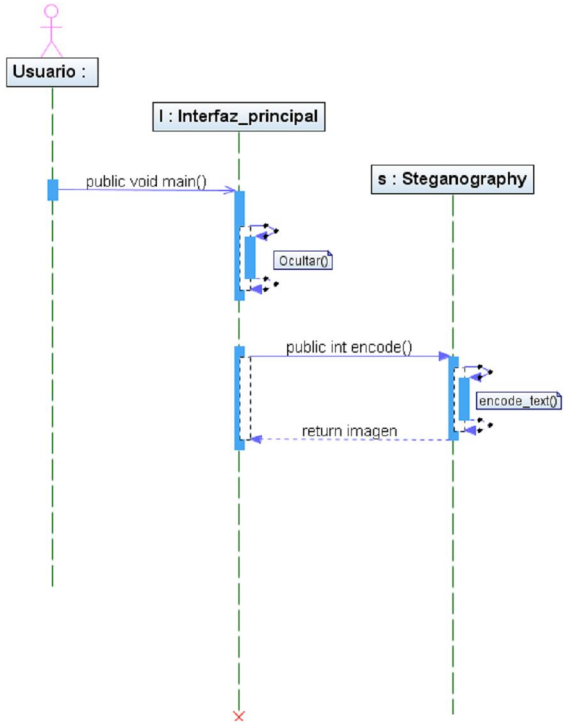
5.2.3.1 Diagrama de Secuencia del Método RSA



Descripción

1. El usuario inicia el método main de la clase principal
   1.1 Se llama al método abrirArchivo( ), de la clase Interfaz_principal, para abrir al archivo del mensaje
2. Se llama al método encriptar( ), de la clase RSA, para encriptar el mensaje.
   2.1 Se llama al método generaClaves( ), de la clase RSA, de este modo obtenemos los números primos
   2.2 La clase RSA envía un mensaje que la codificación fue exitosa

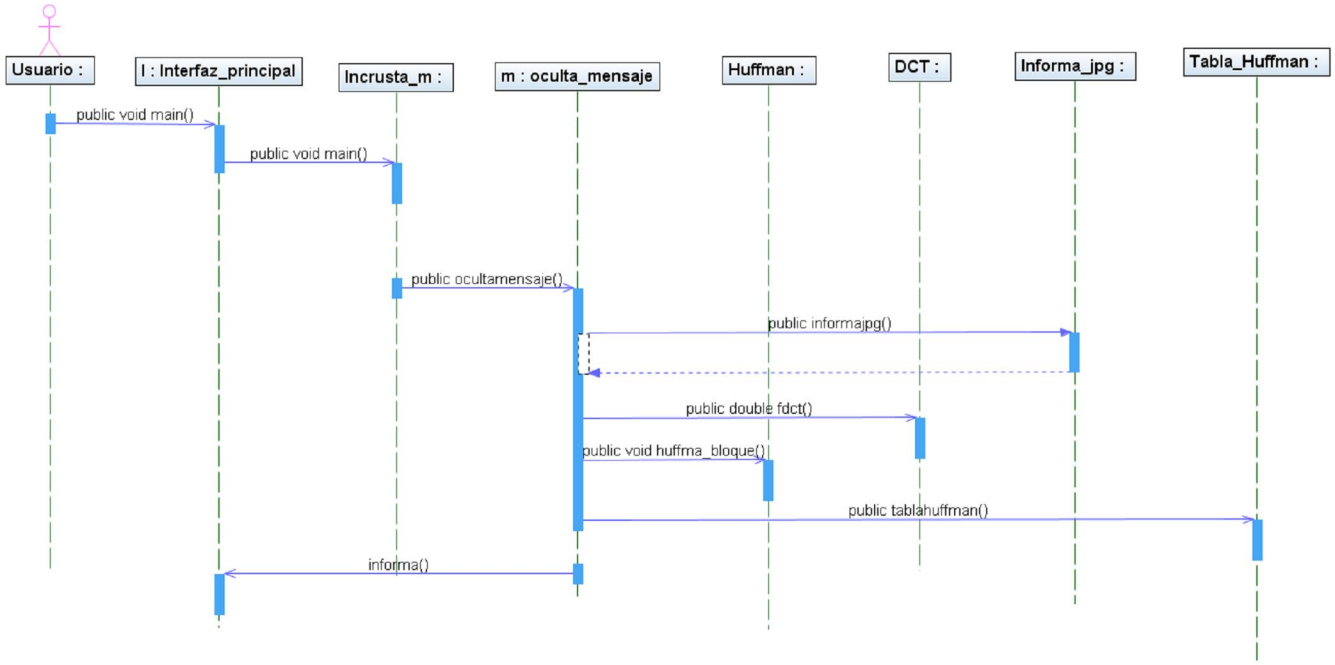
5.2.3.2 Diagrama de Secuencia del Método LSB



Descripción

1. El usurario inicia el método mian
   1.1 Se llama al método ocultar( ), de la clase Interfaz _ principal, el cual carga la imagen y el mensaje
2. Se llama al método encode( ), de la clase Steganography , el que se encarga de validar los datos.
   2.1 Se invoca al método encode_text( ), de la clase Steganography con ello se procede a la codificación de la imagen.
   2.2 La clase Steganography envía un mensaje que la codificación fue exitosa

5.2.3.1 Diagrama de Secuencia del Método DCT



Descripción

1. El usuario inicia el método main de la clase principal

2. Se llama el método main( ), de la clase Incrusta_m, pasando como parámetros los datos den entrada.

3. Se llama al método oculta _ mensaje( ), y se inicia la codificación.
   3.1 Se invoca al método infojpg( ), filtra la imagen, y regresa información de la misma .
   3.2 Se llama la método fdct( ), de la clase DCT. que efectúa la transformación de las frecuencias.
   3.3 Se invoca al método huffman_bloque( ) de la clase Huffman que realiza la compresión.
   3.5 Se llama la método de tablas_huffman( ) de la clase Tablas_Huffman, para obtener los coeficientes.

4. La clase oculta _ mensaje envía un mensaje que la codificación fue exitosa

5.3 Implementación del Sistema

Para la implementación se ha utilizado Java que es un lenguaje orientados a objetos, una de las características mas importantes es que los programas ejecutables, creados por el compilador java, son independientes de la arquitectura. Se ejecutan indistintamente en una gran variedad de equipos con diferentes microprocesadores y sistemas operativos.

- Cuenta con una librería de java JCA (Java Crytography Architecture) la cual contiene los paquetes con los algoritmos de encriptación con clave Simétrica y Asimétrica.
- Dispone de un API, de procesamiento de imágenes JAI (Java Advanced Imaging).
- Es público, puede conseguirse un JDK(Java Developer s Kit) o Kit de desarrollo de aplicaciones Java gratis.

A continuación se muestran los principales pseudocódigos y los diagramas de flujo.

5.3.1 Pseudocódigos

Pseudocódigo de la Trasformada Discreta de Coseno

```
Para  v = 0  hasta n
       Para  u  =0 hasta n
                 Para  k = 0  hasta  n
                      para  l  = 0 hasta n
                            ad(v, n ) +=  af( k, l )  *  coseno(( 2  *  k  + 1)  *   u  *  Pi) /  16)
                            * coseno ((2 * l  + 1)  *  v  *  Pi / 16)
                      fin _ para
                  ad( v, u) *=  0.25  * ( u  !=  0  ?  1  :  1 / raíz( 2))  *  (v  !=  0  ?  1  : 1/raíz( 2));
                 fin _ para
        fin _ para
 fin _ para
```

Pseudocódigo de la Transformada Inversa del coseno

```
Para  x = 0 hasta  8
    Para  y = 0 hasta  8

                suma = 0.0
              Para  i = 0 hasta  8

                      m1 = (2.0  *  x  + 1.0) *  i  *  PI  / 16.0
                     ci = (i = 0) ? 1.0 /  raiz(2.0)  :  1.0

                           Para  j = 0  hasta  8

                                m2 = (2.0 * (y  + 1.0) *  j  *  PI  / 16.0
                                cj = ( j = 0 )  ?  1.0 /  raiz(2.0) : 1.0
                                m3 = ci * cj * 0.25;
                               suma += m3 *  coseno(m1) *  coseno(m2) *  ad(i, j)

                          fin _ para
              fin _ para

             ad(x, y) = suma;
    fin _ para

fin _ par
```

5.3.2 Diagramas de flujo

Diagrama de flujo de para decodificación de los coeficientes AC

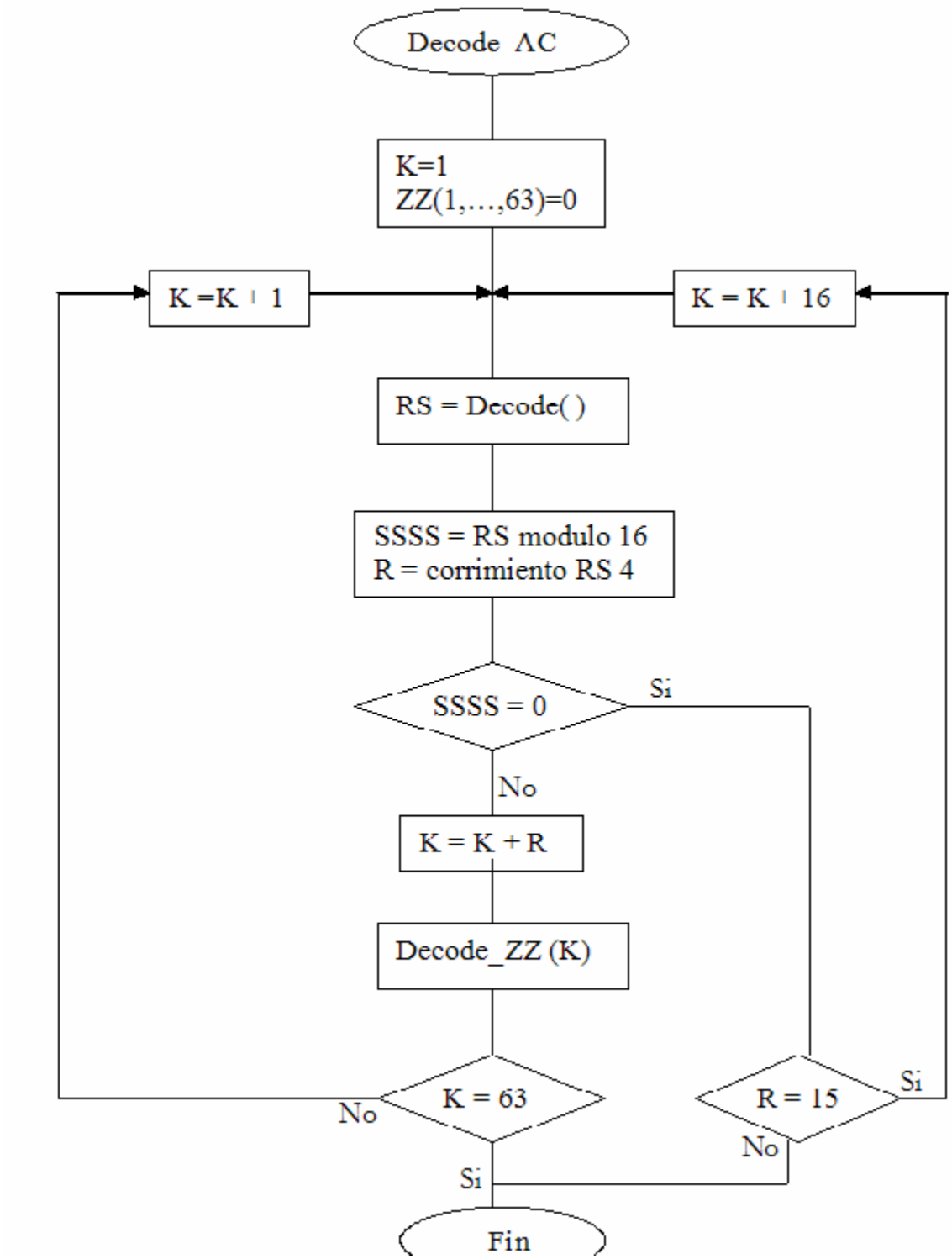


Diagrama de flujo para recibir la Bits SSSS de la categoría

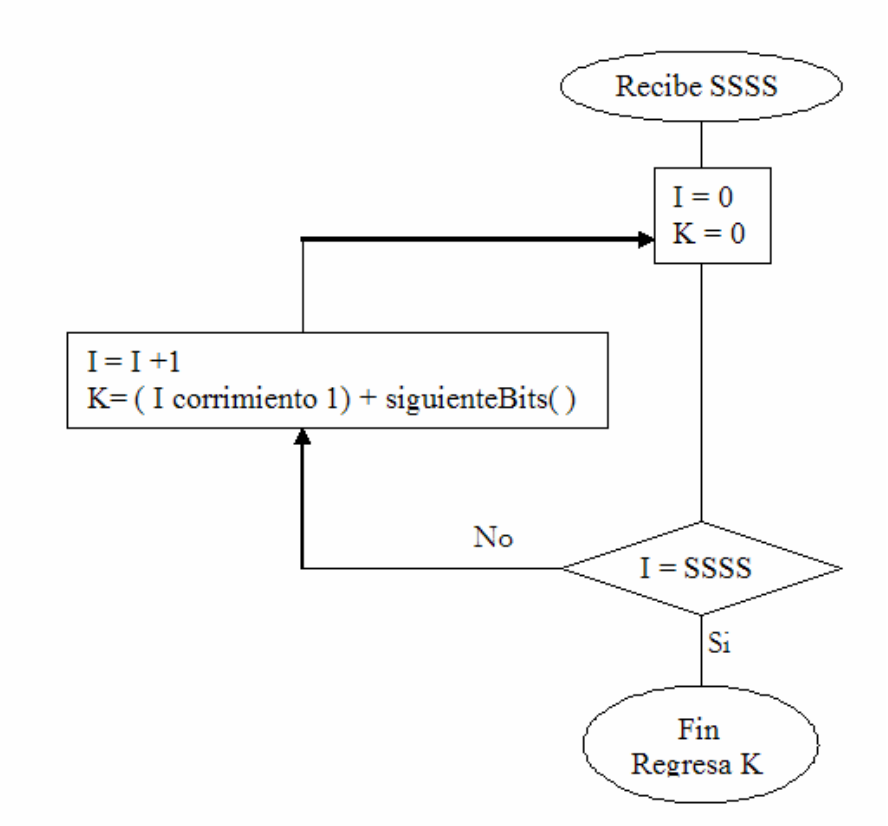


Diagrama de flujo para la Decodificación

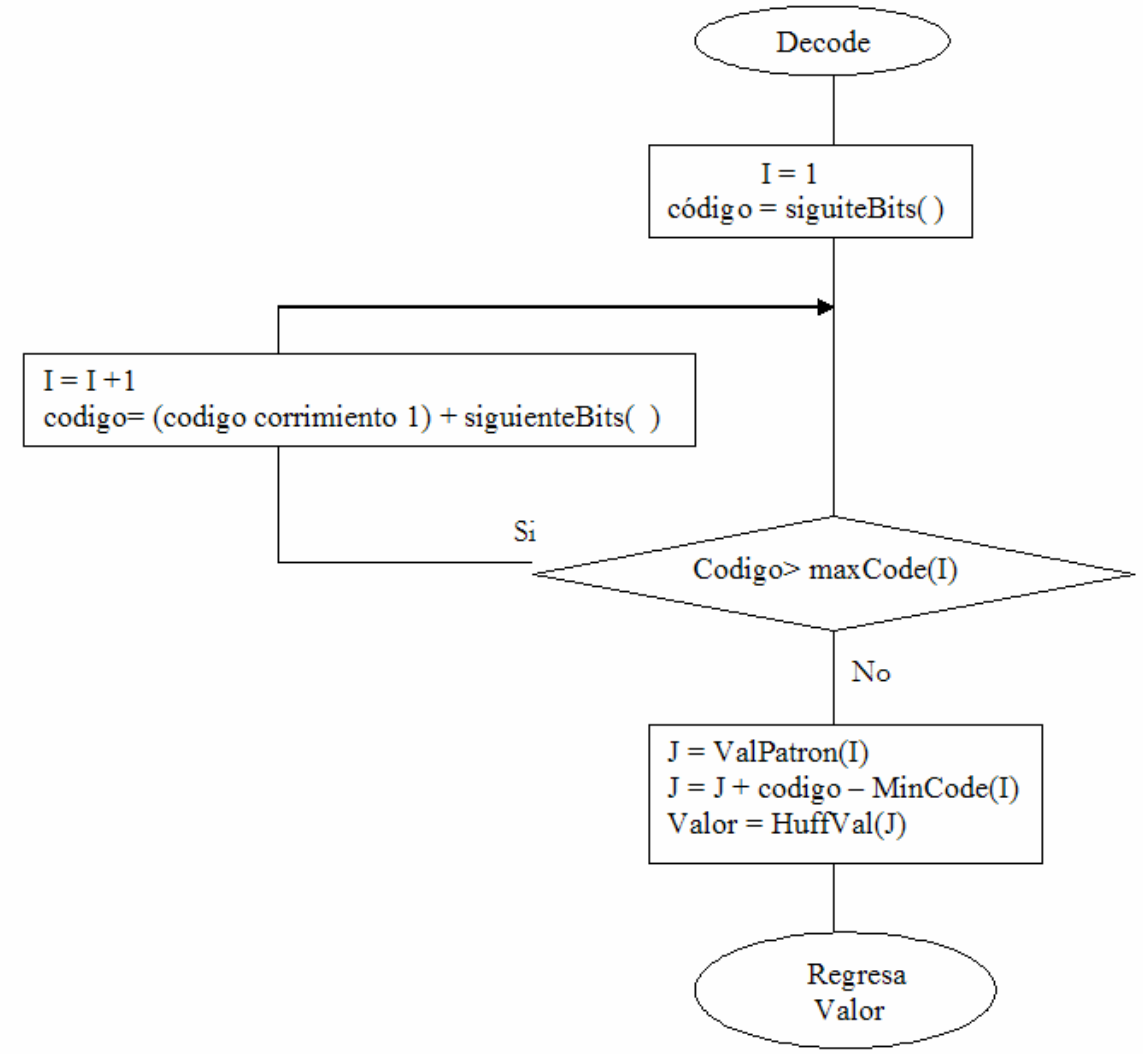


Diagrama de flujo para Generar el tamaño de la tabla de los códigos Huffman

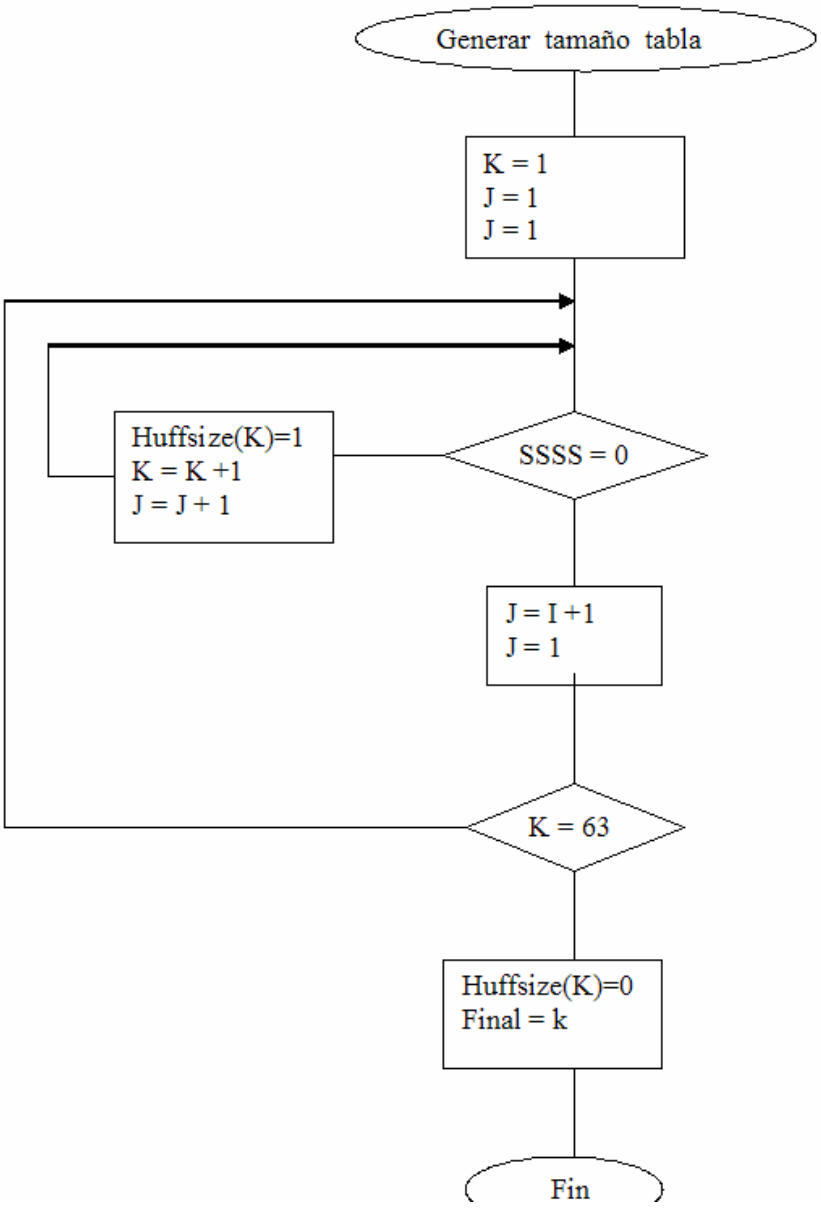


Diagrama de flujo para generar la tabla de código Huffman

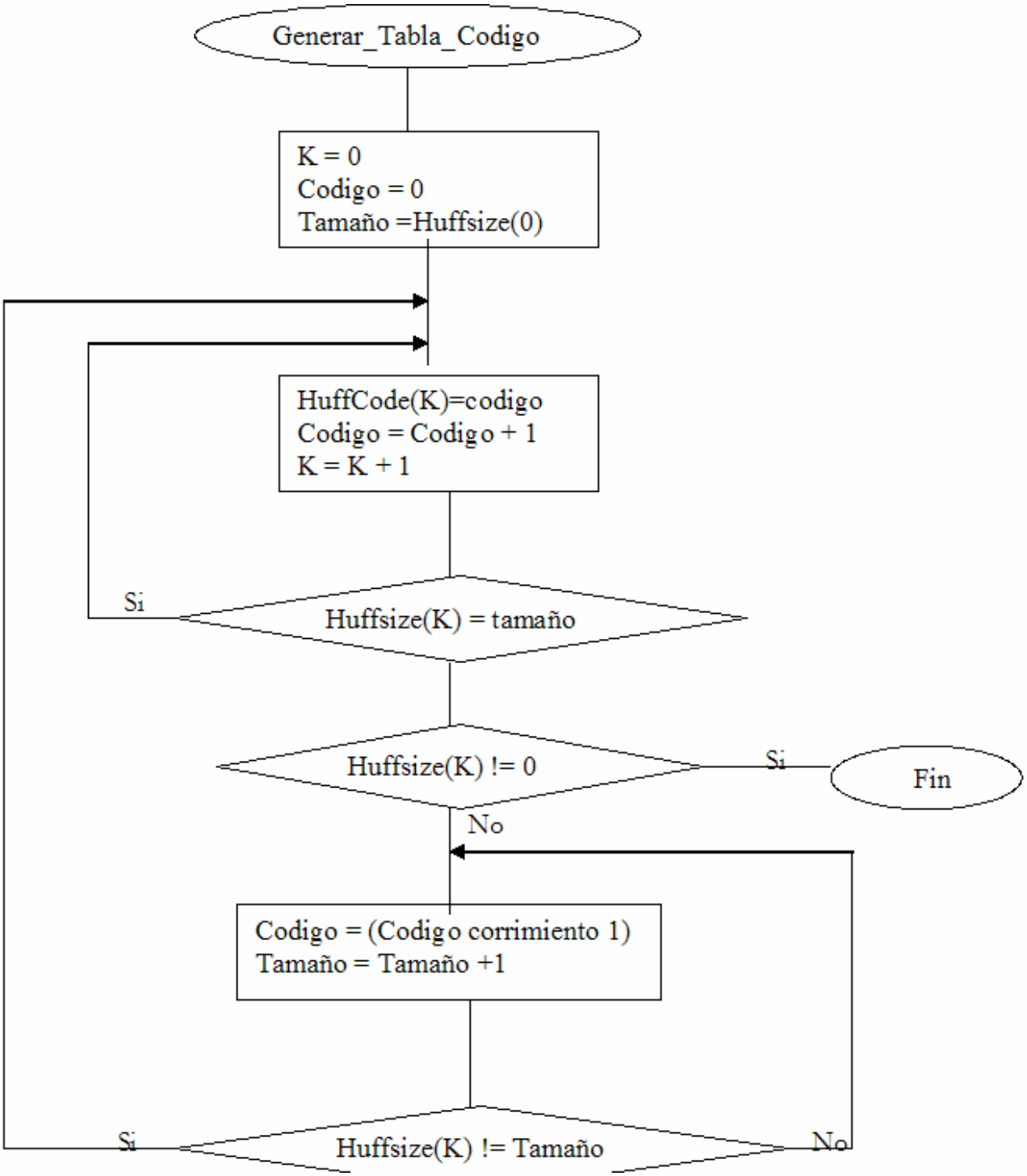
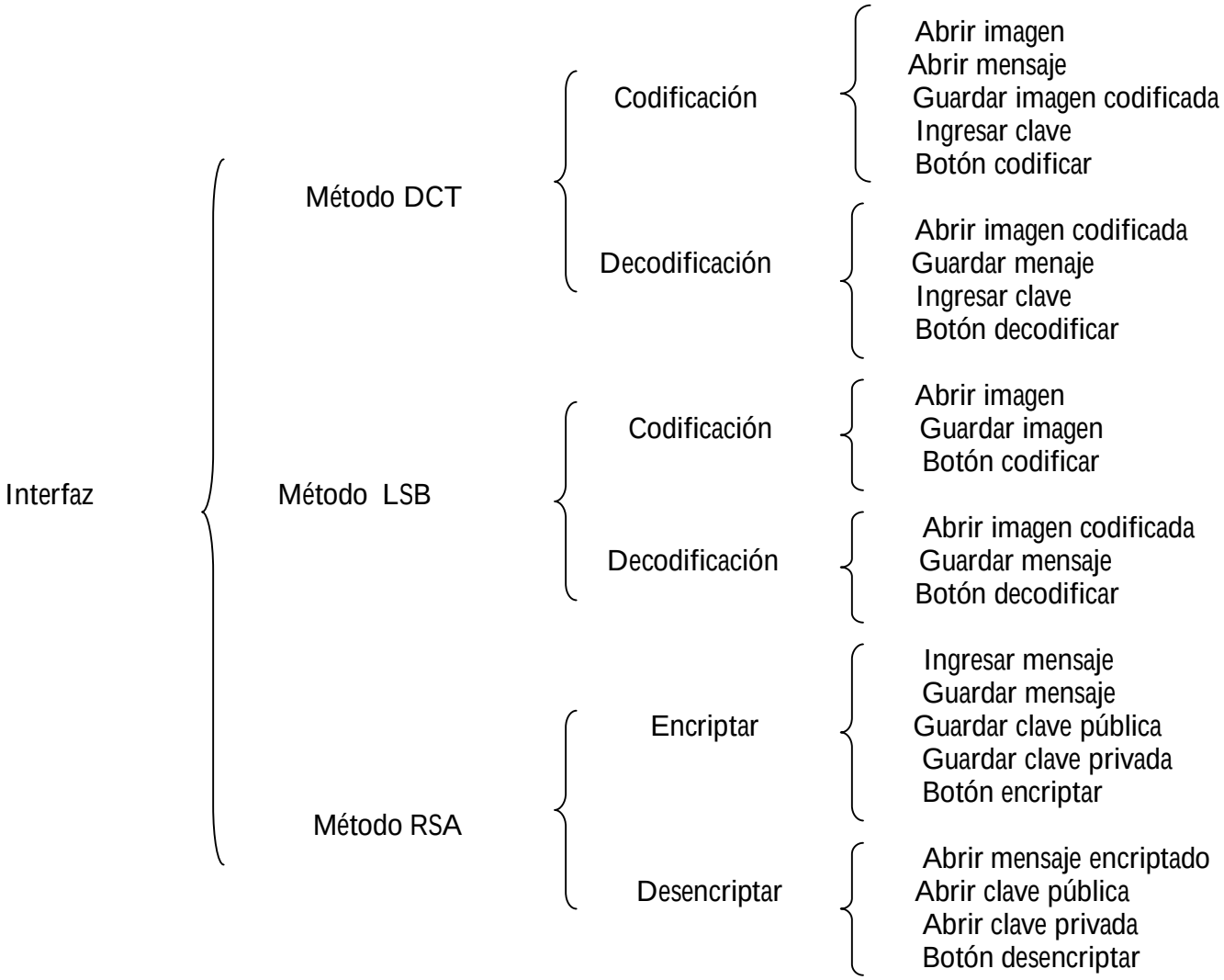


Diagrama de Menú de la Interfaz

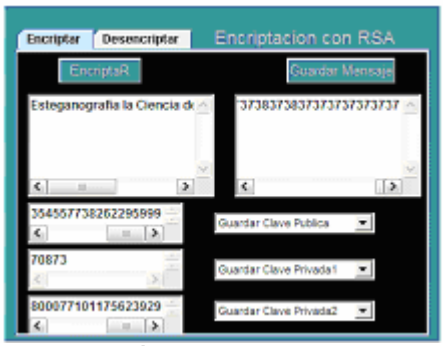
A continuación se describe el diagrama general del menú del sistema



5.3.2 Interfaces del Sistema

Encriptación RSA

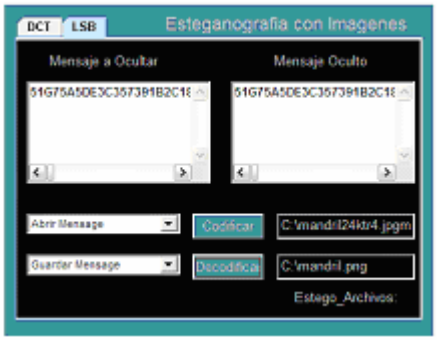
Desencriptacion RSA



Codificación DCT

Decodificación DCT

Codificación y Decodificación LSB



Interfaz General

Capítulo 6

Pruebas y Resultados

A continuación se presentan ejemplos significativos, para cada uno de los métodos, LSB y DCT usando un mensaje ecriptado con RSA.

Mensaje a Encriptar con el Método RSA

Mensaje: *Esteganografía la Ciencia de la Información Oculta*

Mensaje encriptdo con RSA generado:

51G75A5DE3C357391B2C180F0B789207D628755A1F7323079B21A56D8EE24480A7B9CFC31945B39A
03F4A5F4A296A53EDB2AC64812BCC77B88BBDF5B199C1C6D180BF54C357C07D31D30AA5C1A65
DA954C77C13602A724B2515BE067F92E42E7C70C215F4EEB49B2D27157A837C74B7DD77140D4B5
38C21B2C4F73F8C0DF402A8A422E12E9954167F1BBA08D8DDDC2525DE5F2D8F30E891A65DA954
C77C13602A724B2515BE067F92E42DC9DB225B038FD1FBAA6DA5038C5967B158B8E7C70C215F4E
EB49B2D27157A837C74B7DD77315E4DAFB7778E29A23B96432083B3999EC4818A3F302DB76EAA3
20CE774BB79593C97B84B1E7C70C215F4EEB49B2D27157A837C74B7DD7786715262B04DA36EA6F
F6522A004E184E15BF86F581126494EBBF886A4DB9358ED352928AEE7C70C215F4EEB49B2D27157
A837C74B7DD7786715262B04DA36EA6FF6522A004E184E15BF179E20DEF62886DDC49D5F96D21
A72AD9AD82418A3F302DB76EAA320CE774BB79593C97B84B1BDF5B199C1C6D180BF54C357C07D
31D30AA5C140D4B538C21B2C4F73F8C0DF402A8A422E12E1302C354F38166B2A7250595DF3836960
668A18A3F302DB76EAA320CE774BB79593C97B84B1E7C70C215F4EEB49B2D27157A837C74B7DD7
786715262B04DA36EA6FF6522A004E184E15BF2182AA846FB3AF46ED7A82A94295D4D02D4F76BD
F5B199C1C6D180BF54C357C07D31D30AA5C86715262B04DA36EA6FF6522A004E184E15BF86F5811
26494EBBF886A4DB9358ED352928AEE7C70C215F4EEB49B2D27157A837C74B7DD7786715262B04
DA36EA6FF6522A004E184E15BF20F024A9C7C5B5387CAB8AA7AD0CDC89171A12140D4B538C21B
2C4F73F8C0DF402A8A422E12E315E4DAFB7778E29A23B96432083B3999EC489954167F1BBA08D8D
DDC2525DE5F2D8F30E89DC9DB225B038FD1FBAA6DA5038C5967B158B8E7C70C215F4EEB49B2
D27157A837C74B7DD7722EAB785E85CDC86771EF38138F9E4779BAC8EE7C70C215F4EEB49B2D2
7157A837C74B7DD771302C354F38166B2A7250595DF3836960668A18A3F302DB76EAA320CE774BB7
9593C97B84B19954167F1BBA08D8DDDC2525DE5F2D8F30E89140D4B538C21B2C4F73F8C0DF402

A8A422E12E86715262B04DA36EA6FF6522A004E184E15BFCCAF7196E6AD7BFAEC7A8FECA94AE
4A13E8B71302C354F38166B2A7250595DF3836960668ADDC96EE0BE955B841FF8F13BA2FD77D3A8
66586F581126494EBBF886A4DB9358ED352928AE9A03F4A5F4A296A53EDB2AC64812BCC77B88BE
7C70C215F4EEB49B2D27157A837C74B7DD77373837373837383738373738373737373837383738
37373837373737373837373737383737383738373737373737

Tamaño del archivo txt 1.95 Kb

Con Clave Pública:

833434210354745749921001220354557738262295999

Y Clave Privada:

70873

Clave dos n:

525722204621213487448896158800077101175623929

Desencriptado el mensaje utilizando las claves

Clave Pública: 833434210354745749921001220354557738262295999

Clave Privada: 70873

Métodos Esteganográficos

Se utilizó la imagen de la figura 6.1, ya que contiene tonos de colores muy variados, y eso es un punto que hay que considerar para insertar información en una imagen. El mensaje que se inserta en las imágenes, fue encriptado con RSA, en cual se describe en el ejemplo anteriormente.

Fig. 6.1 Imagen sin Información Oculta
Dimensiones 256 x 256
Tamaño: 13.8 KB

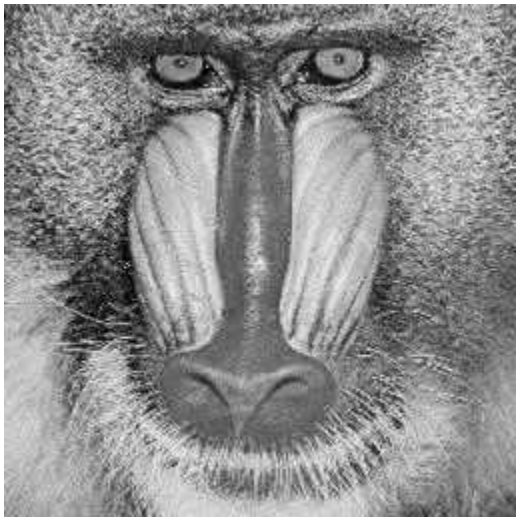


Fig. 6.2 Imagen con Información Oculta generada con el método LSB
Dimensiones 256 x 256
Tamaño: 88.2 KB

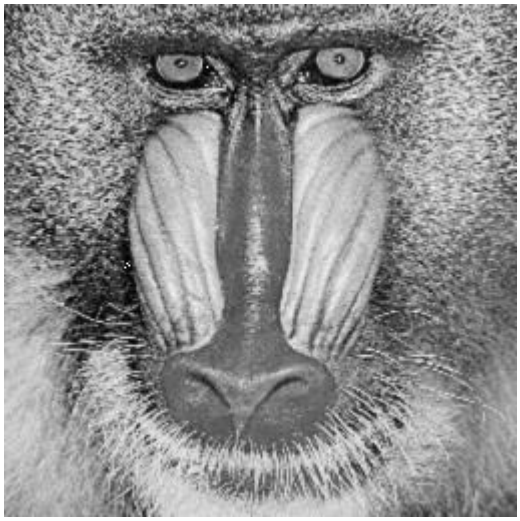
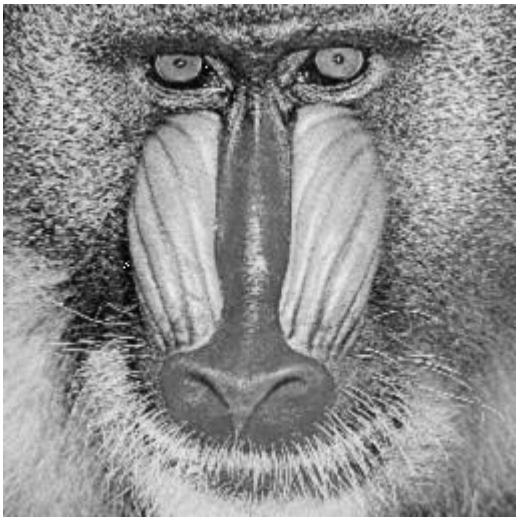


Fig. 6.3 Imagen con Información Oculta generada con el método DCT
Dimensiones 256 x 256
Tamaño: 13.9 KB

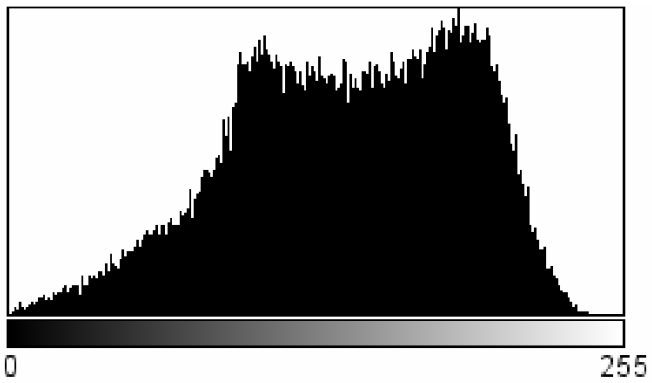


Las imágenes que se obtuvieron, visualmente no sufrieron cambios notables, con respecto a su peso el método de LSB Fig. 6.2 lo incremento notablemente, el método DCT Fig. 6.3 mantuvo el tamaño de la imagen.

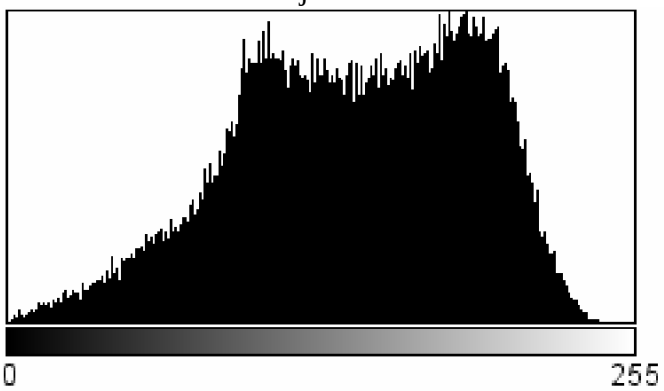
Resta de histogramas del método LSB

Sin embargo se ha realizado la resta de los histogramas de la figura 6.1 y figura 6.2, con el objeto de analizar cuantitativamente el efecto de insertar la información en la imagen.

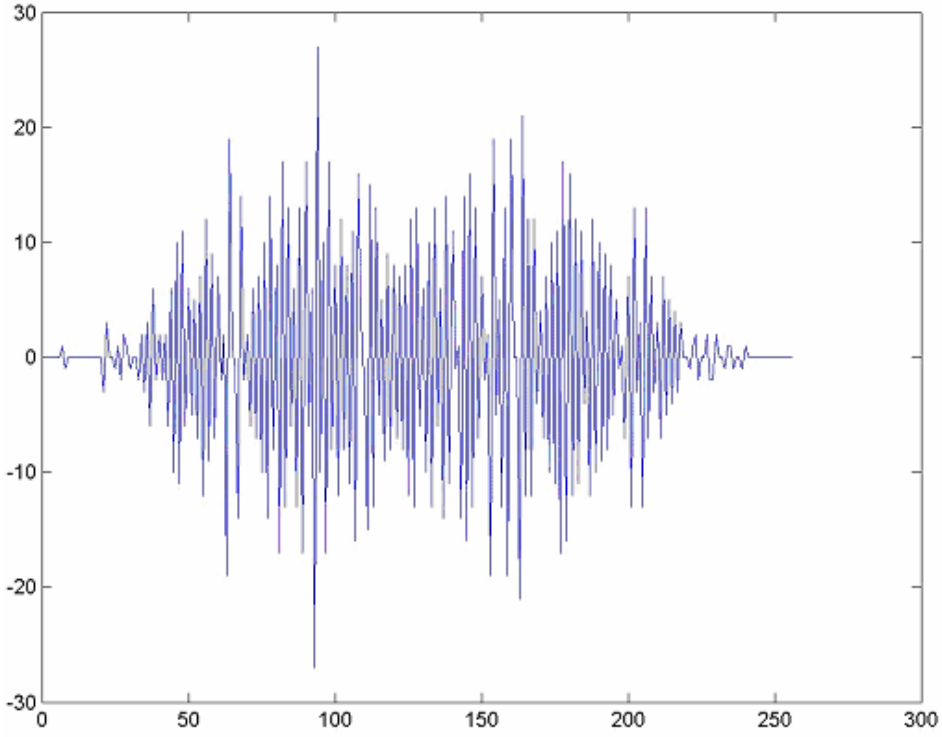
Histograma de la imagen Fig. 6.1
Sin mensaje



Histograma de la imagen Fig. 6.2
Con mensaje utilizando LSB



Gráfica 1, resta de histogramas Fig. 6.1 y Fig. 6.1

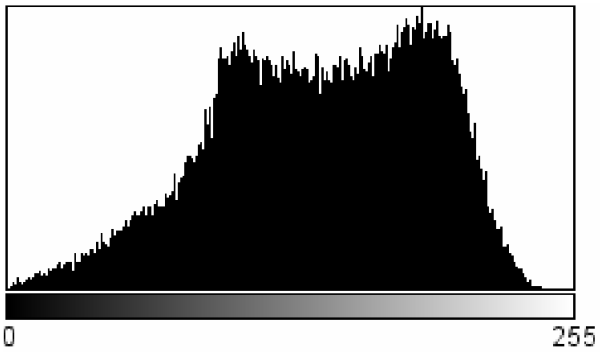


Observando la gráfica 1, podemos notar, que la imagen se modificó en los tonos de colores, claros y oscuros pero a simple vista no se notan.

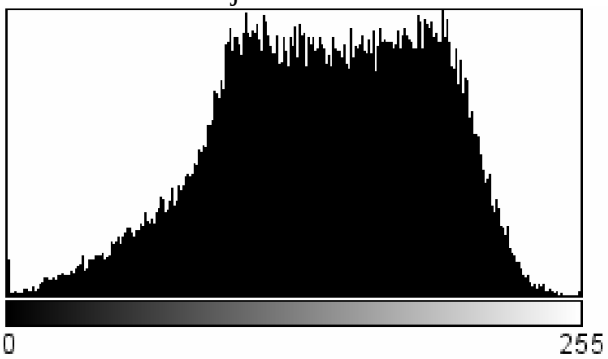
Resta de histogramas del método DCT

Se efectúa la misma operación, pero a hora con los histogramas de la figuras 6.1 y la figura 6.3, en este caso se utilizo el método DCT para insertar el mensaje.

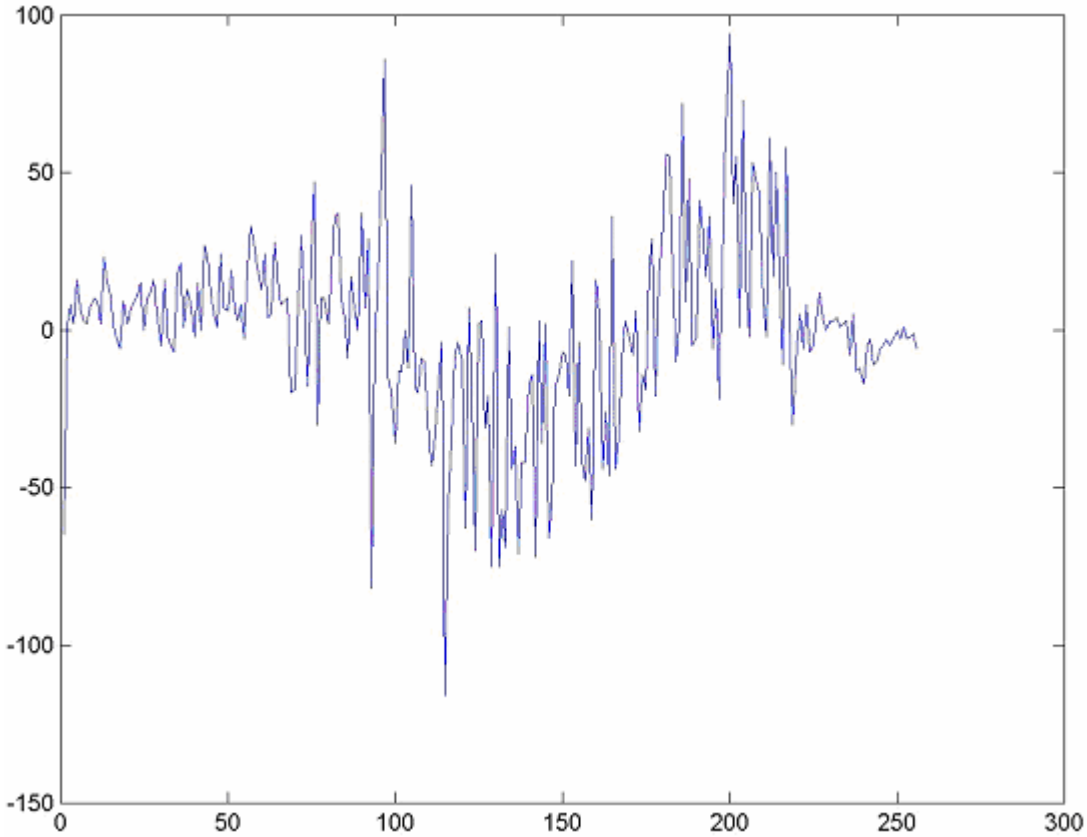
Histograma de la imagen Fig. 6.1
Sin mensaje



Histograma de la imagen Fig. 6.3
Con mensaje utilizando DCT



Gráfica 2, resta de histogramas Fig. 6.1 y Fig. 6.3



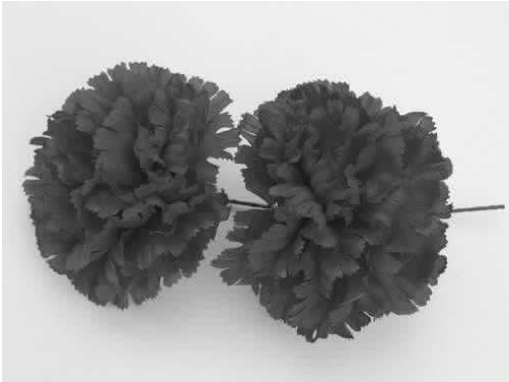
A diferencia de la gráfica 1, la gráfica 2 presenta cambios más drásticos esto significa que la imagen se modificó de un forma mas aleatoria, en los tonos de colores tanto claros como oscuros, ya que el mensaje se inserta en diferentes áreas de la imagen.

Para este ejemplo se toma una imagen figura 6.4, con tonos de colores uniformes, con respecto a la imagen de ejemplo anterior, este tipo de imágenes no son muy recomendables. Por que sufre cambios notables a simple vista

Fig. 6.4 Imagen sin Información Oculta
Dimensiones 500 x 375
Tamaño 66.4 KB

Fig. 6.5 Imagen con Información Oculta generada con el método LSB
Dimensiones 500 x 375
Tamaño: 124 KB

Fig. 6.6 Imagen con Información Oculta generada con el método DCT
Dimensiones 500 x 375
Tamaño: 10.8 KB

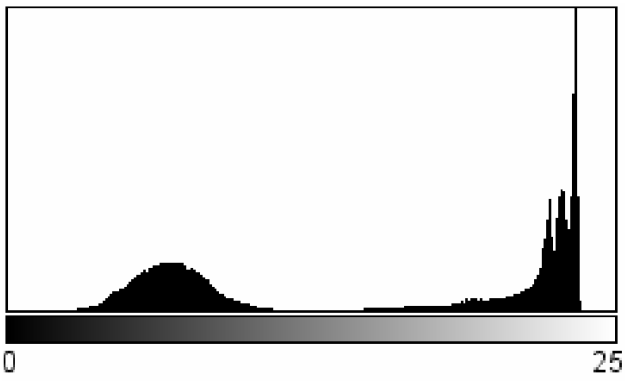


El mensaje que se inserto en las figuras 6.2 y 6.3, es el mismo que se inserta en las imágenes de la figuras 6.5 y 6.6, los resultados obtenidos en este caso son notables, la imagen figura 6.5 se opaco, esto es por que el método de DCT la comprimió demasiado ya que existe demasiada redundancia, por lo tanto su calidad es menor, con respecto al método LSB la imagen figura 6.6, no sufre cambios aparentes pero su tamaño aumenta considerablemente.

Resta de histogramas del método LSB

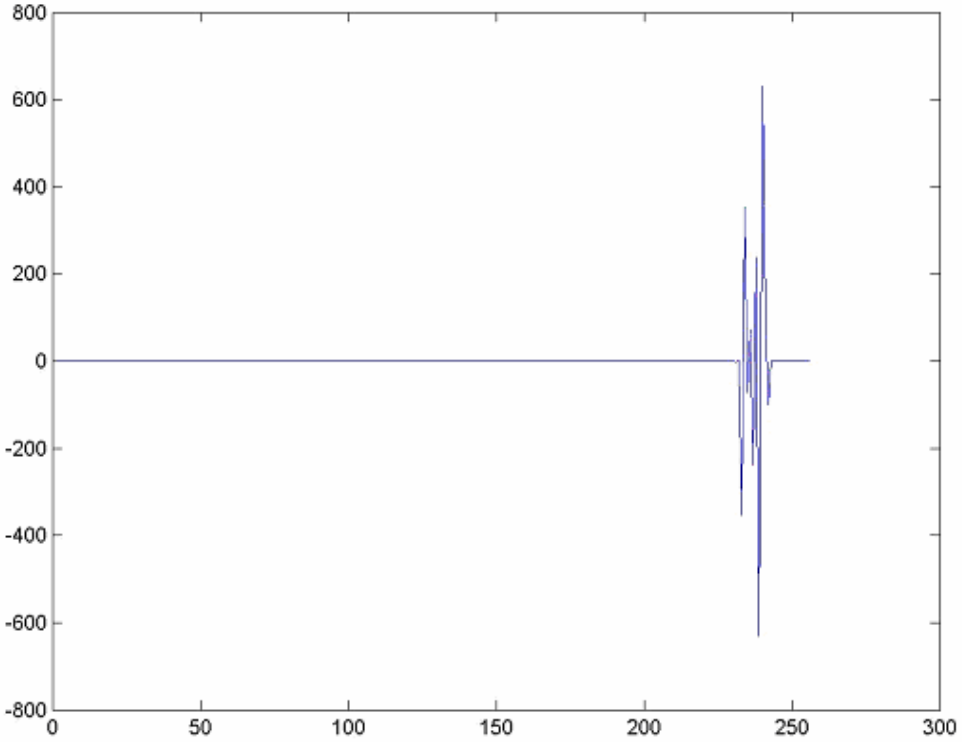
Sin embargo se ha realizado la resta de los histogramas de la figura 6.4 y figura 6.5, con el objeto de analizar cuantitativamente el efecto de insertar la información en la imagen.

Histograma de la imagen Fig. 6.4
Sin mensaje



Histograma de la imagen Fig. 6.5
Con mensaje utilizando LSB

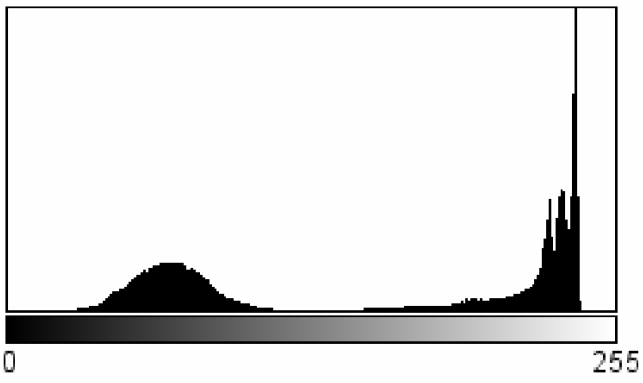
Gráfica 3, resta de histogramas Fig. 6.4 y Fig. 6.5



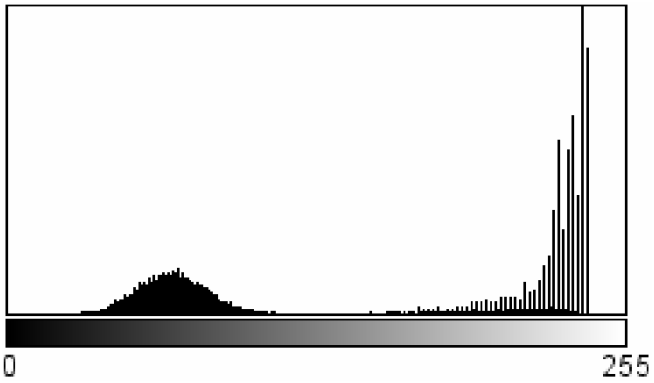
Como la gráfica 3 lo ilustra, este tipo de imágenes con tonos de colores uniformes, los cambios se concentran en los colores claros debido que las imágenes no tienen mucha variación de colores. Con una observación minuciosa de la imagen se pueden notar los cambios.

Resta de histogramas del método DCT

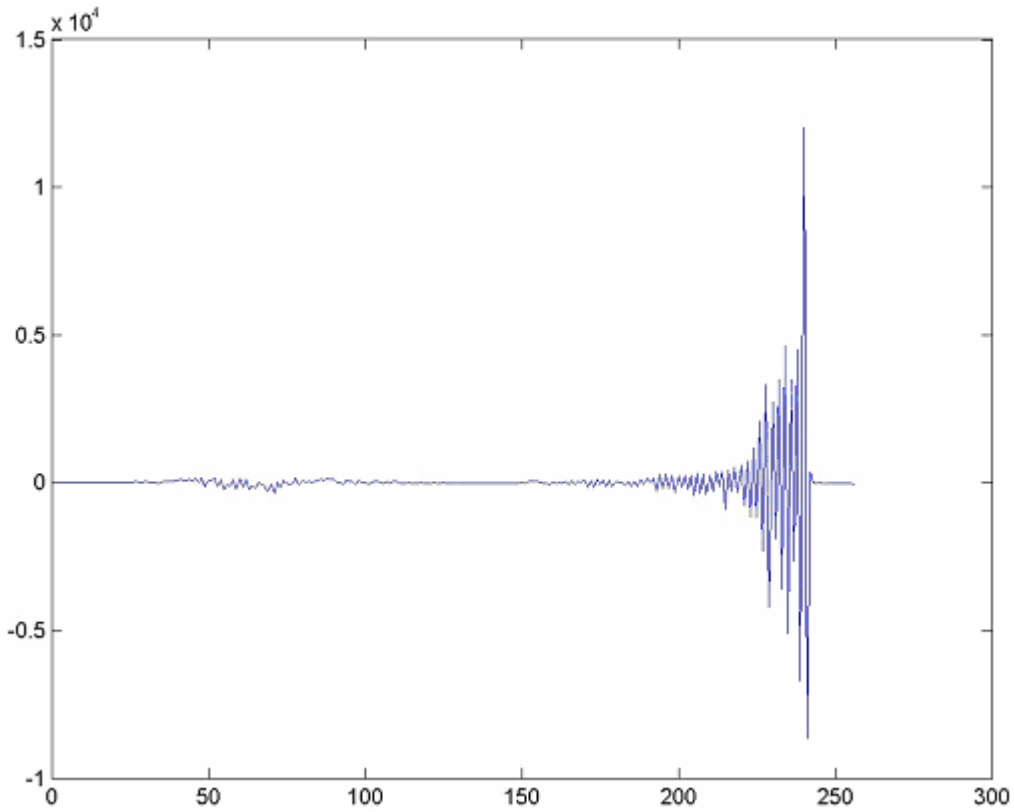
Histograma de la imagen Fig. 6.4
Sin mensaje



Histograma de la imagen Fig. 6.6
Con mensaje utilizando DCT



Gráfica 4, resta de histogramas Fig. 6.4 y Fig. 6.6



Para este caso utilizando el método DCT, como la gráfica 4 lo ilustra, la imagen sufre mínimos cambios en los tonos de colores claros, pero debido a la compresión de la imagen pierde calidad ya que existe mucha redundancia y es eliminada.

Para medir el desempeño de cada uno de los algoritmos, se emplean los siguientes parámetros:

El parámetro pico de razón señal a ruido PSNR (peak singnal to reconstruction noise), que es una medida relativa de calida de imagen. Se basa en el error cuadrático medio RMSE (root mean squared error) y se calcula con la siguiente fórmula:

$$MSE = \sqrt{\frac{1}{mn}\sum_{y=1}^{m}\sum_{x=1}^{n}\left(A(x,y) - B\left(x,y\right)\right)^2}$$

Donde m, n representan el largo y ancho de la imagen, A(x, y) es la imagen original y B(x, y) es la imagen reconstruida.

Una vez calculado el MSE se pude obtener PSNR.

$$PSNR = 20\log_{10}\left(\frac{256-1}{\sqrt{MSE}}\right)$$

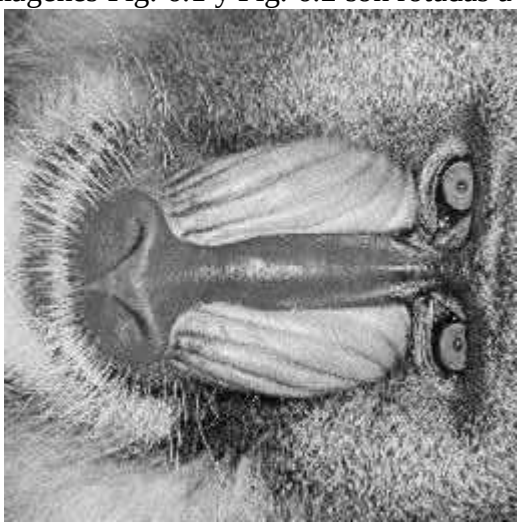
Un valor bajo de MSE, significa que hay menos error en la señal reconstruida con respecto a la señal original; lo cual se traduce en un valor grande de PSNR (en decibeles). Es decir un valor grade de PSNR, es bueno ya que significa que la razón señal a ruido es grande.

Parámetros de Medición		
Método	**MSE**	**PSNR**
Método DCT aplicando a la imagen de la figura 6.2	**15.6883**	**36.17504475**
Método DCT aplicando a la imagen de la figura 6.4	**4.7521**	**41.3619479**

6.1 Pruebas y Ataques

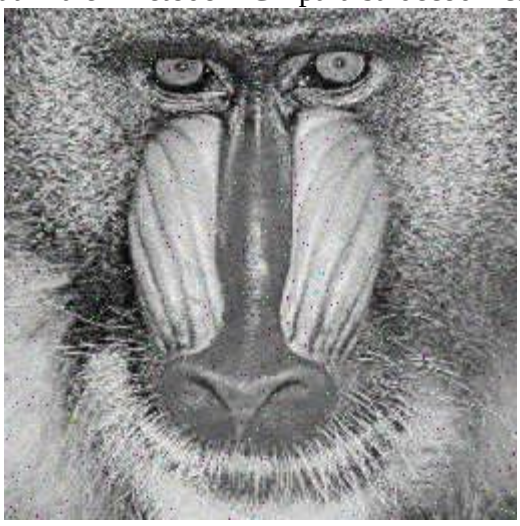
A continuación se realizan unas pruebas de ataques, a las imágenes generadas por los métodos de LSB y DCT, las cuales consisten en rotar a la imagen, agregar ruido a la imagen, escalar la imagen, y por último se somete a la imagen a un proceso de estegoanálisis.

Las imágenes Fig. 6.1 y Fig. 6.2 son rotadas a 90°



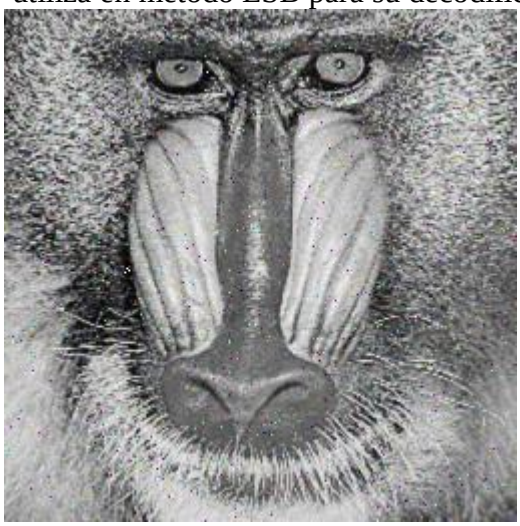
El mensaje no se pudo recuperar por se encuentra en diferente posición, se utilizo los dos métodos

Se le agrega ruido a la imagen Fig.6.2 de tipo salt & pepper con un porcentaje de 10 %
Se utiliza en método DCT para su decodificación



El mensaje no se pudo recuperar, debido a que se le insertan caracteres ajenos a la imagen, el archivo donde se guarda la información no contiene nada.

Se le agrega ruido a la imagen Fig.6.1 de tipo salt & pepper con un porcentaje de 10 %
Se utiliza en método LSB para su decodificación



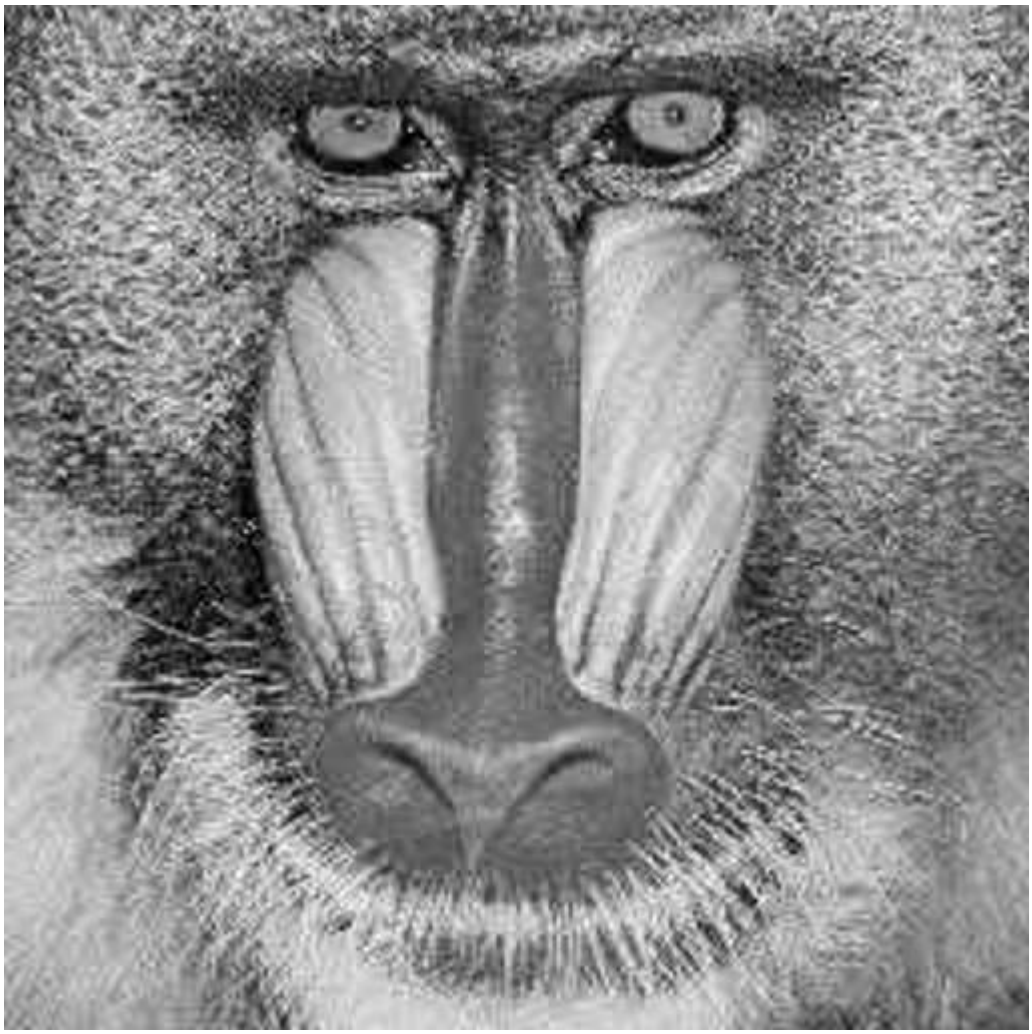
51G75A5DE‼ C357391B2C180F0B789207D628755A1F7323079B21A56D8EE24480A7B9CFC31945
B39A03F4A5F4A296A53MDB2AC64812BCC77B88ÂBDF5B199C1C6D1↑ 0BF54C357C07D31D30
AA5C1A65DA954C77C13>02A724B2515BE067F92E42E7C70C21%F4EEB49B2D271u7A83wC74B
7DD77140D4B538C21B2S4F73F8C0DF402A8A422E12E9954167F1BBA08D8DDDC2525DE5F2D8
F30E891A65DA954C77C13602A724B2515BE067F├ 2E42DC9DB225B038FD1FBAA6DA503↑ C59
67B158B8E7└ 70C215F4EEB49B2D27157A836C74B7DD77315E4DAFB7779E29A23B96432083B3
999EC4818A3F#02DB76EaA320CE774BB79593C97B84B1E7C70C215F4EEB49B2D27157A837C7
4B7DD7786715262B04DA36EA6FF6522A004E184U15BF86F581126494EBBG886A4DB935↑ ED³5
2928AEE7C70C215F4EEB49B2D27157E837C74B7DD7786755262B04DA37EA6FF6522A004E184
E15BF179E20DEF62886DDC49D5F96D21A72A@9AD¸2418A3Æ302DB76EAA320CE774BB79593
C97B84B1BDF5B199C1C6D180BF54C357C07D31D30AA5C14pD4B538C21B2C4F73F8C0DF402A
8A422E12E1302C354F3816┬ B2A6250595DF3(36960668A18A3F302DB76EAA320CE774BB78593
A97B84@1E7C70C215F4EEB49B2D27157A837C74B7DD7786715262B0¶ DA36EA6FF6522A004E
184E15BF2182AA846FB3AF56ED7A>↕ A94295D4D02D4F76BDF5B1=9C1C6D190BF54C357C07
D31D30AA5C86715262B04DA36EA6FF6522A004E184E15BF86F581126484EBBF886A5DB9358E
D352928AEE5C70C215F4EEB49B2D27157A837C74B7DD7786715262B04DA³6EA6FF6122A004E
104E15BF20F024A9C7C5B5387CAB8AA7AF0CDC89171A12140D4B538C21B2C4f73F8C0DF402
A8A422E12E71%E4DAFB7778E29A23B96432083B3999EC48895416'F1BBA08D8DDDC2505DE5F
2D8F30E89DC9DB225B038FD1FBAA6DA5038C5967B158B8E7C70C215F4EEB49B2┘ 27157A837
C74B7DD772"│ AB785E85KDC86771EF38138F9E4779B@C8EE7C70C31%F4ÅEÂ49B2D27157A8
37C74B7DD771302C354F38166B2A7250595DF3836960668A18A3F302DB76EAA320CE774BB795
¹3C97B84B19954167F1BBA08D<DDDC2525DE5F2D8– 30E89140D4B538C21B2C4F73F8C0DF402
A8A422E12E86715262B04DA36AA6FF6522A004E184E◀ 5BFCKAF7196E6AD7BFAEC7A8FECA
94AE4A13E8B71302C354F38166B2A'250595DF3836960268ADDC96EE0BE955B841FF8F13BA2F
D7·D#A866586F58112¶494EBBF8↑ 6C4FB9358ED352928AE9A03F4A5F4@296A53EDB2AC64812
BBC77B88BE7C?0C215F4EEB49F2D27157A837C74B7DD773738373738373837283737373837373737
373738383738373837373837373737373┴ 383837373737383737383737383737373┤ 373737

El mensaje que se recupera contiene caracteres basura, debido al ruido que se agrego

DCT

La imagen de la figura 6.2 se escala al doble de su tamaño.

Dimensiones originales de la imagen 256 x 256
Dimensiones de la nueva imagen 512 x 512

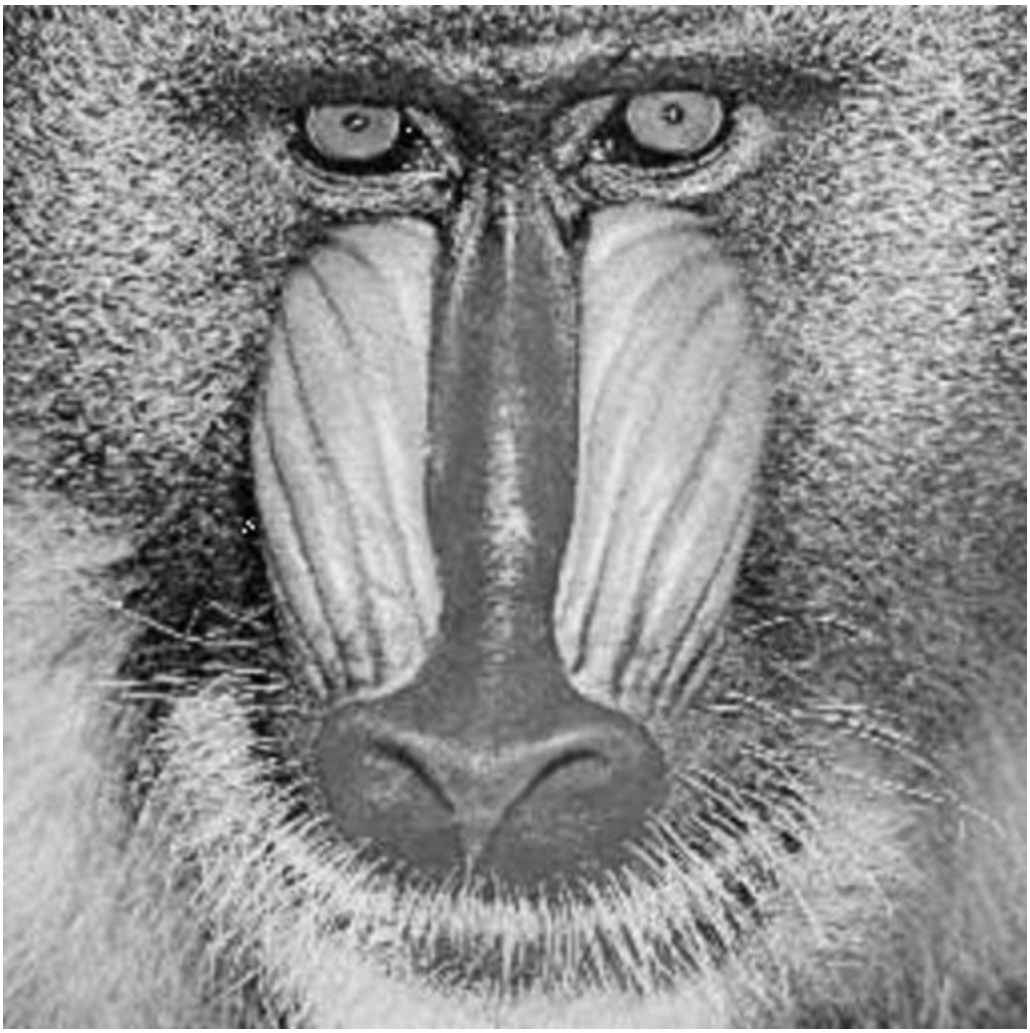


Utilizando el método DCT, el mensaje no se pudo recuperar, debido a que las posiciones no son las mismas, donde se insertó el mensaje con respecto a la imagen nueva, ya que a partir de unos bytes se generan otros bytes generando a si la imagen con dimensiones más grandes.

LSB

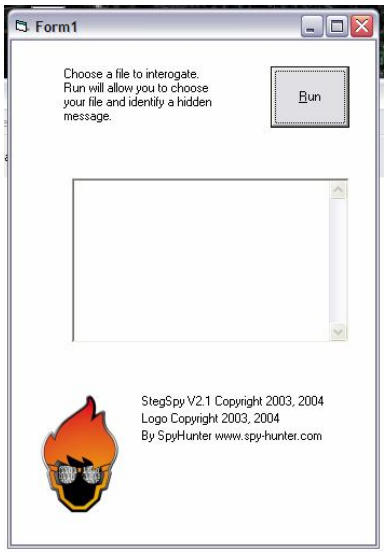
La imagen de la figura 6.2 se escala al doble de su tamaño.

Dimensiones originales de la imagen 256 x 256
Dimensiones de la nueva imagen 512 x 512



Utilizando el método LSB, el mensaje no se pudo recuperar, debido a que las posiciones no son las mismas, donde se inserto el mensaje con respecto a la imagen nueva, ya que a partir de unos bytes se generan otros bytes generando a si la imagen con dimensiones más grandes.

Utilizando el software de Estegoanális StegSpy V2.1 de estegánalisis, se sometió a las imágenes generadas pos los dos métodos, LSB, DCT. Dando como resultado.



Las imágenes generadas con el método **LSB,** el software detectó mensaje oculto**.**

Las imágenes generadas por el método **DCT,** el software no detectó mensaje oculto.

A manera de conclusión, determinamos que las imágenes son muy frágiles a cambios, rápidamente pueden ser modificadas, con ello el mensaje no se puede recuperar. Pero el objetivo de la esteganografía con imágenes, es permitir que las imágenes no sean detectadas ya sea por medios electrónicos o físicos, debido a que las imágenes estarán alojadas en sitios Web, serán envidas a través de correo electrónico estarán guardadas en discos duros, memorias USB, esto reducirá el riesgo que la imágenes sean adulteradas, auque no se descarta. Las imágenes estarán a la vista de público, sin saber que dichas imágenes contienen mensajes ocultos, solamente las personas que tengan la clave para su decodificación, podrán acceder al mensaje, o talvez la clave este implícita en el nombre de la imagen, como podría ser la fecha que fue tomada la foto, utilizando algunos números en específico de la fecha, se podrá determinar el patrón con que fue insertado el mensaje en la imagen.

Con respecto a Marcas de Agua, su objetivo es darle a las imagen una fortaleza para soportar ataques, con la intención de eliminar la marca el las imágenes, ya que si la marca es eliminada el algoritmo que se utilizó para su inserción no es confiable.

Conclusiones

A continuación se presentan las conclusiones de presente trabajo de tesis, el cual tuvo la tarea de implementar técnicas de Esteganografía con imágenes, utilizando la Transformada Discreta de Coseno y el método de sustitución del bit menos significativo. Denotado los siguientes puntos:

- La información que se inserta en las imágenes, para los dos métodos se recupera íntegramente, sin que se le apliquen modificaciones a la imagen con información.

- Se recomienda imágenes con tonos de colores variados, por el motivo, que cuando se inserta la información en las imágenes, pueden sufrir cambios visuales.

- Se realizó una red de pruebas de ataques a las imágenes con información escondida con el objeto de mostrar la resistencia de las imágenes con mensajes.

- Se realizaron pruebas con los histogramas, de las imágenes sin mensaje y con mensaje insertado, se elaboró la resta de dichos histogramas, con el fin de analizar cuantitativamente las imágenes.

- Se comprobó que los métodos del dominio de las frecuencias son más robustos, que los métodos del dominio espacial. Pero su capacidad de insertar información es mayor en los métodos en el dominio espacial LSB, pero se recomienda que los mensajes que se inserten en las imágenes, sean lo mas cortos posibles.

- Una de las características más interesantes que se tiene de este proyecto es el hecho de brindar una seguridad diferente a la información cuando viaja por canales inseguros, ya que los mensajes están ocultos en la información considerada como válida, en este caso imágenes.

De esta forma elegimos los algoritmos que tuvieran mejor desempeño, teniendo en cuenta las técnicas de Estegoanálisis Básicas, que cuentan con algoritmos muy eficientes para detectar la información en las imágenes. Se sometió a un proceso de estegoanálisis a los dos métodos LSB y DCT, la imagen que se codificó con el método LSB fue detectado el mensaje, a diferencia del método DCT en la imagen codificada no fue detectado el mensaje.

Información escondida es el término más general, el cual abarca a la Esteganografía y Marcas de Agua, la información escondida se esta aplicando principalmente en el área de autenticidad y derechos de autor, tanto en imágenes como en audio, aclarando que estos no son los únicos archivos que se utilizan, sin en cambio son los mas comunes. Otra área en la cual se esta trabajado es en la protección del Software, son muy diversas las áreas y los métodos de la información escondida, no limitándose a una sola área involucrándose varias disciplinas, al igual que en la criptografía, y aún mas. El uso de esta disciplina tiene diferentes objetivos, el principal es proteger la información.

Limitaciones y Perspectivas

Limitaciones

Utilizando el método LSB, que corresponde al domino espacial, debido que se esta utilizando el formato JPEG para las imágenes, el cual involucra compresión de datos, por tal motivo la inserción y extracción del mensaje no se da directamente como en las imágenes con formato BMP, la compresión se tiene que tratar con procesos diferentes ajenos a estos métodos, por ello se tomaron las siguientes medidas. La imagen de entrada es con formato JPEG, y la imagen de salida es con el formato PNG para evitar la compresión de datos y así poder realizar el proceso de extracción del mensaje.

El en proceso de Encriptación del mensaje, debido a que se ésta, utilizando encriptación Asimétrica RSA, los mensajes generados son muy grandes, esto hace que el mensaje que se inserta en la imagen tendrá un mayor peso aumentando el tamaño de la imagen, de otro modo si el mensaje se inserta sin encriptar su peso es menor, en las perspectivas se toman medidas para este inconveniente.

Perspectivas

Utilizando a Librería de java JCA (Java Crytography Architecture) la cual contiene los paquetes, con los algoritmos de encriptación con clave Simétrica y Asimétrica, como son el DES el triple DES, AES. De esta forma el sistema puede mejorar agregándole las opciones de encriptación simétrica y asimétrica, teniendo en cuenta que la simétrica genera mensajes cortos a comparación de la asimétrica, disminuyendo el peso del archivo que se insertara en la imagen.

Otra perspectiva que se tiene es migrarlo dispositivos móviles utilizando la plataforma J2ME de Java. Contemplando únicamente el método LSB debido que estos dispositivos su procesamiento es limitado, manejando únicamente números enteros, lo cual limita los métodos del dominio de las frecuencias. Con ello se pretende proteger la información que se encuentra en el Celular o PDA.

Bibliografía

[1]. Stefan Katzenbeisser Fabien A. P. Petitcolas “Information Hiding Techiques for Steganography and Digital Watermarking ”, Artech House. 2000

[2] Greg Kipper, Investigator's Guide to Steganography, Ed. AUERBACH PUBLICATIONS, 2004.

[3] Majid Rabbani, Paul W. Jones, Digital Image Compression Techniques Ed. SPIE OPTICAL ENGINEERING PRESS, Volumen TT7, Sexth Printing 1991

[4] Weidong Kou, DIGITAL IMAGE COMPRESSION Algorithns and Standards, Ed. KLUWER ACADEMIC PUBLISHERS, Second Printing. 1995

[5] William B. PenneBaker, Joan L. Mitchell, JPEG Still Image Data Compression Standard, Van Nostrand Reinhold, 1993.

[6] Rafael C. González, Richard E. Woods, Tratamiento Digital de Imágenes, Addision Wesley/Diaz de Santos, Ed. Primera 1992.

[7]. Gonzalo Pajares, Jesús M. de la Cruz, José M. Molina, Imágenes Digitales Procesamiento practico con Java, Alfaomega Ra-Ma 2003

[8] William Stallings, Cryptography and Network Security Principles and Practices, Fourth Edition Prentice Hall, 2005

[9] Amparo Fuster Sabater, Dolores de la Guía Martínez, Luis Hernández Ecinas, Fausto Montoya Vitini, Jaime Muñoz Masque, Técnicas Criptográficas de Protección de Datos Ed. Alfaomega RA-MA, Segunda Edición. 1998

[10] Scout Oaks, JAVA Security, Ed. O’REILLY , First Edition. 1998

[11] Jonathan Knudsen, JAVA Cryptography, Ed. O’REILLY, First Edition.

[12] Rafael C. González, Richard E. Woods, Digital Image Processing, Ed. Prentice Hall Second Edition 2002.

[13] Gonzalo Pajares, Jesús M. de la Cruz, Visión por Computador Imágenes digitales y aplicaciones, Ed. Alfaomega RA-MA, Primera Edición 2002.

[14] Samir S. Soliman Mandyam D. Srinath, Señales y Sistemas Continuos y Discretos Ed. Prentice Hall, Segunda edición 2000

[15] Jorge Lira Chávez, Introducción al Tratamiento de Imágenes, Instituto Politécnico Nacional, Universidad Autónoma de Puebla, Fondo de Cultura Económica. 2002

[16] Ross J. Anderson, Fabien A.P. Petitcolas, On The Limits of Steganography, IEEE Journal of Selected Areas in Communications, 16(4):474-481, May 1998. Special Issue on Copyright & Privacy Protection. ISSN 0733-8716.
http://www.infosyssec.org/infosyssec/security/stendig1.htm,

[17] Richard Bergmair, Stefan Katzenbeisser, Content-Aware Steganography: About Lazy Prisoners and Narrow-Minded Wardens,
http://www.infosyssec.org/infosyssec/security/stendig1.htm,

[18] Richard Popa, An Analysis of Steganographic Techniques, 1998
http://www.infosyssec.org/infosyssec/security/stendig1.htm,

[19] Dorian A, Xavier Flowers, Steganography 2000.
http://www.infosyssec.org/infosyssec/security/stendig1.htm,

[20] Alvaro Martin,Guillermo Sapiro, Gadiel Seroussi, Is Image Steganography Natural?, Information Theory Research Group HP Laboratories Palo Alto HPL-2004-39(R.1) August 10, 2004
http://www.infosyssec.org/infosyssec/security/stendig1.htm,

[21] Daniel L. Currie, Cynthia E. Irving, Surmounting the Effects of Lossy Compression on Steganography

[22] Johann Barbier, Eric Filio, Kichenakoumar Mayoura, Universal Detection of JPEG Steganography, JOURNAL OF MULTIMEDIA, VOL. 2, NO. 2, APRIL 2007

[23] Stephen Lau, An Analysis of Terrorist Groups' Potential Use of Electronic Steganography SANS Security Essentials GSEC Practical Assignment Version 1.3, February 18, 2003