Capítulo 2

Marca de agua digital y su robustez

2.1 *Introducción*

Actualmente las tecnologías de la información tienen una gran importancia en la vida diaria. El desarrollo de las mismas ha permitido que de manera sencilla sea posible intercambiar información digital perteneciente o no a un propietario. Incurrir en un atentado contra el derecho de autor que junto con la propiedad industrial conforman la propiedad intelectual, es muy factible. Es por ello que se requieren técnicas para proteger la información de usuarios no autorizados o para demostrar quién es el propietario legítimo. Una de esas técnicas es la criptografía la cual permite proteger la información para que sólo la pueda descifrar el destinatario a quien va dirigido, sin embargo una vez descifrada la información pierde su protección. Otra técnica es la marca de agua digital, la cual permite esconder información de manera electrónica en fotografías, videos o música que usualmente son referidos como trabajos y al conjunto de todos ellos como contenido. Tomemos como ejemplo un billete, en el cual la marca de agua permanece oculta y sólo es visible luego de un proceso especial para observarla (como poner el billete a contraluz). En este caso, la marca de agua porta información referente a la autenticidad del contenido en el que se mantiene oculta.

Un trabajo que se encuentra marcado puede ser víctima de diversos tipos de alteraciones intencionales o no intencionales. Para que la marca de agua no pierda su utilidad es necesario que pueda sobrevivir a dichas modificaciones. Una forma de hacerla más resistente es codificándola con un código detector y corrector de errores antes de insertarla en el trabajo.

2.1.1 Criptografía

El encriptado o cifrado es un proceso para transformar la información, también llamada texto plano, en un criptograma, el cual se caracteriza por ser ininteligible con la finalidad de transmitir un mensaje de forma segura a través de un canal donde la intercepción es posible. Para convertir el criptograma nuevamente en texto plano es necesario realizar el proceso inverso conocido como desencriptado. Para realizar el encriptado y desencriptado se requiere de una clave que debe ser conocida por el transmisor y receptor del mensaje [4].

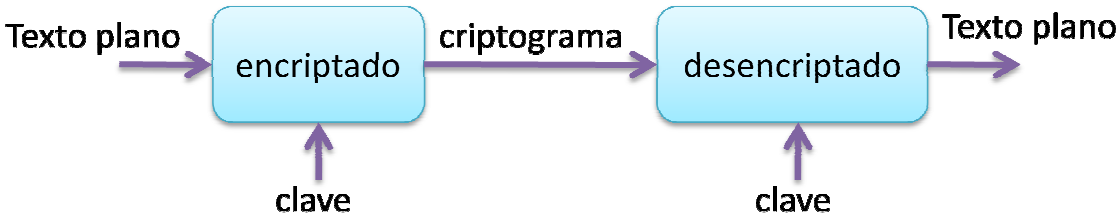

Figura 2.1 Esquema general del encriptado y desencriptado.

La criptografía por sí misma no es suficiente para proteger la información dado que el mensaje ya no permanece seguro luego de haber sido desencriptado.

2.1.2 Esteganografía

Proveniente del griego *steganos* que significa "encubierto" y *graphia* que significa "escrito", la esteganografía es el arte de la comunicación oculta. En la esteganografía la existencia del mensaje está oculta, excepto para el receptor y el transmisor, por lo que debe de añadirse al trabajo de manera que no haya una considerable alteración perceptual en el mismo. Generalmente la incrustación del mensaje no es tan robusta como la que se pudiese realizar con una marca de agua.

2.1.3 Marca de agua digital

El proceso de incrustación de una marca de agua digital consiste en la inserción de información en una señal portadora la cual puede tratarse de cualquier documento digital. El objetivo principal de insertar información en los contenidos es la protección de la propiedad intelectual. El trabajo marcado puede ser publicado, distribuido a través de una red, comercializado y/o radio emitido; situaciones en las cuales el trabajo puede sufrir alteraciones intencionales o no intencionales. En cualquier momento se puede realizar el proceso de detección y extracción de la marca de agua para demostrar, por ejemplo, la autoría o autenticidad del trabajo. Por lo tanto, de manera general, el sistema de marcado de agua consta de una etapa de inserción y una etapa de recuperación de la marca de agua.

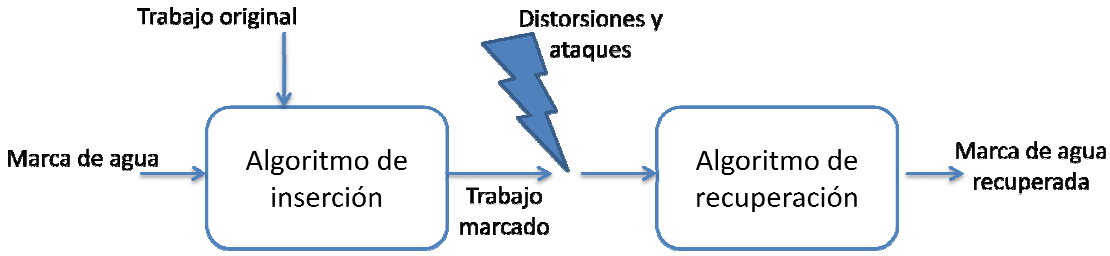

Figura 2.2 Sistema de marcado de agua.

El sistema de marcado de agua digital es similar a un sistema de comunicaciones donde se tiene un transmisor, canal de comunicación y un receptor. En este modelo la inserción de la marca de agua en la señal portadora corresponde a la transmisión de la señal, las modificaciones realizadas por ataques y distorsiones al trabajo marcado corresponden a la transmisión a través del canal de comunicaciones y la recuperación de la información escondida es análoga a la recepción. En los sistemas de cómputo el mensaje a ocultar en la señal portadora es un vector binario de dimensión n de la forma:

$$\bar{b} = (b_1, b_2, \dots, b_n) \tag{2.1}$$

donde $b_k \in \{0,1\}$, $k = \{1,2, \dots, n\}$

Usualmente no se incrusta de manera directa, sino que primero se transforma en una señal $\overline{w}$ con mejores características aleatorias, o de dispersión para ser incrustada.

$$\overline{w} = (w_1, w_2, \dots, w_h) \tag{2.2}$$

donde $w_i \in \{0,1\}$, $i = \{1,2, \dots, h\}$

Al igual que en un sistema de comunicaciones se pueden agregar procesos tales como la codificación de fuente, codificación de canal, espectro disperso, detección y decodificación, como se verá más adelante.

2.2 Características de la marca de agua

- *Robustez:* Es la capacidad que tiene la marca de agua de resistir a las manipulaciones intencionales o no intencionales realizadas a la señal que la porta [1]. El nivel de robustez que debe poseer la marca de agua depende de cada aplicación en particular, sin embargo se pueden considerar cuatro tipos generales de marca de agua de acuerdo a su robustez, los cuales se describirán en el subtema *2.3*.

- *Imperceptibilidad:* Es la capacidad que tiene la marca de agua de permanecer oculta en la señal portadora sin alterar ésta última de manera notable.

- *Capacidad:* Cantidad de información en bits que el algoritmo de marcado de agua puede incrustar de tal manera que la marca de agua se pueda recuperar.

Las características de un sistema de marcado de agua digital varían dependiendo de la aplicación. En ocasiones es deseable contar con una marca de agua robusta que sea resistente, en el caso de imágenes, a los ataques como la compresión JPEG, el ruido aditivo, filtrado, recortes o rotaciones. No obstante, al aumentar la robustez de la marca de

agua también aumenta la información de ésta que se añade en el trabajo, produciendo una mayor alteración perceptual.

2.3 Tipos de marca de agua

De acuerdo a su robustez:

- *Marca de agua segura:* Este tipo de marca de agua es resistente a ataques maliciosos y no maliciosos. Los ataques maliciosos son aquellos en los que el atacante conoce el algoritmo de incrustación de la marca de agua e intenta modificarla o extraerla del trabajo. Las manipulaciones no maliciosas son técnicas comunes de procesamiento digital como la compresión con pérdidas, el filtrado y la adición de ruido; en el caso de imágenes las manipulaciones también pueden ser técnicas para aumentar el contraste, modificaciones en el histograma o recortes. Es posible perder la información incrustada sólo después de una significativa alteración de la señal portadora. Esta marca de agua es utilizada en la protección de los derechos de autor.

- *Marca de agua robusta:* Esta marca de agua es resistente sólo a ataques no maliciosos por lo que es usada en aplicaciones donde se espera que nadie intente manipular a la señal portadora con el fin de remover la marca de agua; a pesar de ello también puede ser utilizada en aplicaciones de protección a la propiedad intelectual.

- *Marca de agua semi frágil:* Se utiliza en aplicaciones donde la señal portadora no sufrirá severas alteraciones, sólo algunas ligeras modificaciones como mejoras en la calidad o compresión moderada.

- *Marca de agua frágil:* Es aquella en la que la información incrustada se pierde o altera cuando la señal portadora sufre cualquier tipo de modificación. Es utilizada principalmente en aplicaciones de autenticación.

De acuerdo a su perceptibilidad:

- *Marca de agua visible:* Este tipo demarca de agua es utilizada en imágenes y video. Consiste en la superposición espacial de la imagen portadora y la marca de agua (tipo "logotipo").

- *Marca de agua invisible:* La marca de agua no es observable en la señal portadora.

(a)

(b)

(c)

(d)

Figura 2.3 (a) Imagen original. (b) Marca de agua. (c) Imagen con marca de agua visible. (d) Imagen con marca de agua invisible.

De acuerdo a su técnica de detección:

- *Algoritmo de marca de agua heurístico:* Es aquel en el que se requiere comparar el trabajo original con el trabajo marcado en el proceso de recuperación de la marca de agua.

- *Algoritmo de marca de agua ciego:* Es aquel que no requiere del trabajo original para extraer la información contenida en el trabajo marcado.

De acuerdo al dominio de inserción

- *Marca de agua en el dominio espacial*

  Se trata de una función que modifica directamente los pixeles que componen la imagen de acuerdo a la información que la marca de agua contiene. Usualmente sólo un subconjunto de la imagen es marcado en este dominio.

- *Marca de agua en el dominio transformado*

  La marca de agua se inserta en los coeficientes de la transformada de la señal o imagen portadora. Usualmente debido a los otros procesos digitales comunes, se escoge la transformada de Fourier o la transformada discreta coseno para la inserción de la marca de agua en el dominio transformado [1], aunque también se pueden utilizar otras transformaciones tales como la transformada *wavelet*, o la transformada *contourlet*. Comúnmente el dominio transformado proporciona mayor robustez a los ataques.

2.4 Distorsiones y ataques

- *Ataques no maliciosos:* Son los ataques que pueden ocurrir en el uso normal del trabajo.

- *Ataques maliciosos:* Son aquellos cuyo objetivo principal es remover la marca de agua del trabajo o hacerla irrecuperable. Los ataques maliciosos pueden o no explotar el conocimiento del algoritmo de marcado de agua.

2.4.1 Ruido aditivo

Algunos procesos realizados en el trabajo pueden tener como efecto la adición de una señal aleatoria considerada ruido aditivo el cual sigue cierta función de densidad de probabilidad.

$$I_N(x,y) = I(x,y) + N(x,y) \tag{2.3}$$

donde I es el trabajo original de tamaño $m \times n$, N es el ruido aditivo, e I_N es el trabajo con rudio.

Este ruido es independiente del trabajo, por lo que las manipulaciones causadas al trabajo por el ruido pueden asumirse como si el trabajo fuese transmitido por un canal con ruido aditivo.

2.4.2 Filtrado lineal

En imágenes, se trata de un filtrado espacial que realiza operaciones con los coeficientes de un filtro, también llamado *kernel* del filtro, y los pixeles de una imagen. Es lineal porque la respuesta del filtro está dada por la suma del producto de los coeficientes del filtro y los correspondientes pixeles de la imagen que se enciman con la máscara del filtro, es decir, es una combinación lineal. En el dominio espacial corresponde a una convolución entre los coeficientes del filtro K y la imagen I.

$$I_f(x,y) = I(x,y) * K(i,j) \tag{2.4}$$

donde I es la imagen original de tamaño $m \times n$, I_f es la imagen filtrada, K los coeficientes del filtro de tamaño $I \times J$ y $*$ denota convolución.

(a)

(b)

Figura 2.4 Ejemplo de filtrado. (a) imagen original. (b) Imagen suavizada con filtro paso bajas Gaussiano de tamaño 5x5 y σ=1.

2.4.3 Recorte de la imagen

Este ataque altera el trabajo mediante la remoción de los bordes o alguna parte de la imagen marcada. Se pueden utilizar técnicas de espectro disperso o de dispersión espacial de la marca de agua al momento de incrustarla para hacerla más resistente a este tipo de ataques.

Figura 2.5 Ejemplo de una imagen recortada.

2.4.4 Distorsiones geométricas

Estas distorsiones incluyen la rotación, escalamiento espacial, y traslación de la imagen. El atacante puede hacer una distorsión geométrica a la imagen con el propósito de dificultar la extracción de la marca de agua. Al hacer una transformación en las coordenadas de la imagen el algoritmo de detección tendrá dificultad para recuperar la marca de agua a menos que realice una búsqueda exhaustiva con el fin de alinear y sincronizar el trabajo marcado con el patrón de referencia para el cual fue diseñado el detector.

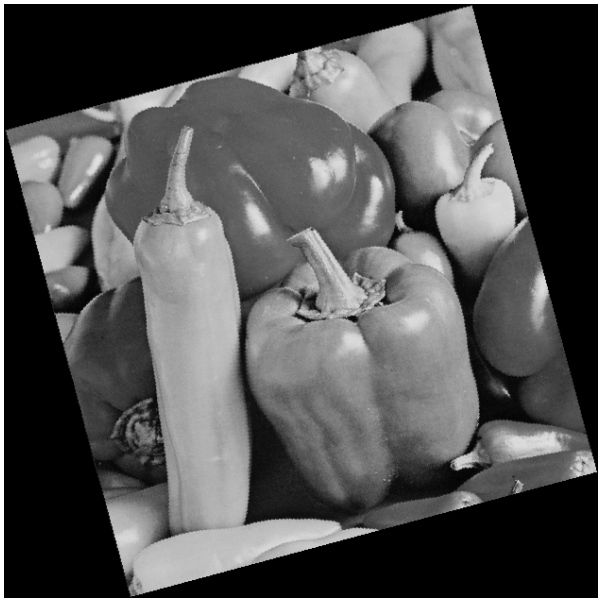

Figura 2.6 Ejemplo de distorsión geométrica: rotación.

2.4.5 Compresión

En la compresión con pérdidas el trabajo original no es igual al trabajo descomprimido. En teoría de la compresión, se puede hacer compresión sin considerable degradación debido a que la información que representa la señal electrónica contiene redundancia con respecto a la información necesaria para la percepción humana. Se pueden lograr diferentes niveles de compresión, como en el caso de la compresión de imágenes JPEG, especificando el factor de calidad de la imagen. Cada valor del factor de calidad corresponde a una matriz

de cuantización. Dependiendo de la aplicación, se puede buscar la resistencia de la marca de agua a la compresión JPEG.

2.4.6 Modificaciones del histograma

En algunas aplicaciones será necesario que la marca de agua sea resistente a este tipo de manipulaciones, pues son muy comunes en el mejoramiento de imágenes. Por ejemplo, si una imagen es muy oscura, se buscará aumentar el rango dinámico de los niveles de gris de los pixeles oscuros y disminuir el rango dinámico de los pixeles claros para conseguir un mayor contraste; es decir, se expande el rango de los valores de pixeles oscuros de una imagen y se contrae el de los valores de mayor intensidad. Esto se realiza mediante una operación puntual logarítmica sobre la intensidad de luz de cada pixel que conforma la imagen. También es común obtener el negativo de una imagen para resaltar ciertas figuras presentes en ella. Ambas técnicas realizan modificaciones en el histograma.

2.5 Aplicaciones del marcado de agua

2.5.1 Monitoreo de emisiones de contenido

Existen diferentes organizaciones e individuos interesados en el monitoreo de emisiones de contenido. Las compañías que anuncian sus productos en televisión necesitan asegurar que se emita su publicidad en los espacios "al aire" que rentaron y los propietarios de trabajos con derechos de autor deben asegurar que no se distribuyan ilegalmente sus contenidos. Para ello se puede realizar el monitoreo incrustando una marca de agua que porte información de identificación capaz de ser recuperada por una computadora en la recepción de las emisiones.

2.5.2 Demostración de propiedad

Para demostrar que alguien es el propietario de un trabajo al momento de su creación se puede incrustar una marca de agua que identifique al autor con dicho trabajo. El autor

también puede detectar la distribución de su trabajo en *Internet* mediante un motor de búsqueda que localice su trabajo en la red y así saber si es víctima de fraude.

2.5.3 Control de copias

Para evitar la copia ilegal de contenidos con derechos de autor, como los DVD's, se ha añadido al encabezado MPEG un conjunto de bits llamado bits CGMS (Copy Generation Management System) que le indican a un grabador de DVD's que el disco está protegido contra la copia ilegal abortándose el proceso de copiado. Sin embargo este sistema no ha funcionado pues se puede reproducir el DVD y al mismo tiempo convertir la señal analógica de televisión nuevamente a digital utilizando otro grabador de DVD's creando un disco que ya no posea los bits CGMS, el cual el reproductor de DVD's considerará como disco de libre distribución y todas las copias ilegales de éste podrán ser reproducidas.

Una mejor solución al problema de copias ilegales es migrar a un sistema en el que los bits CGMS también sean incrustados en el video como una marca de agua digital segura. De esta manera si se intenta reproducir un DVD grabado a partir de la señal analógica de video, el reproductor recuperará la marca de agua y al no existir bits CGMS en el encabezado MPEG, el reproductor sabrá que se trata de una copia ilegal [1].

2.5.4 Autenticación

La facilidad de procesar señales de manera digital permite alterar trabajos sin dejar rastros perceptibles de la modificación. Recuperar una marca de agua previamente incrustada en el trabajo original permite saber si el trabajo es el legítimo o se trata de una versión alterada.

2.6 *Esquema general del marcado y recuperación de la marca*

La fase de inserción de marca de agua digital consiste en una función f que transforma el trabajo original $I(x,y)$ de acuerdo a la marca de agua $\overline{w}$ utilizando una clave K dando como resultado el trabajo marcado $I'(x,y)$. K sirve para dar seguridad al sistema pues aunque se conozca el algoritmo será indispensable emplear la clave correcta para realizar la extracción de la marca de agua. Desde el punto de vista del procesamiento, la marca de agua $\overline{w}$ se trata del vector $\bar{b}$ codificado.

$$I'(x,y) = f(I(x,y), \overline{w}, K) \tag{2.5}$$

donde: $I(x,y)$ es el trabajo original de tamaño $m \times n$, $\overline{w}$ es la marca de agua codificada de dimensión h, y K es la llave.

Como se mencionó anteriormente, existe la detección ciega y la detección informada de la marca de agua. En la detección informada la recuperación se realiza a partir del trabajo marcado, el trabajo original, y la clave, como se muestra en la ecuación 2.6.

$$\overline{w}_r = f(I'(x,y), I(x,y), K) \tag{2.6}$$

donde: $\overline{w}_r$ es la marca de agua recuperada, $I'(x,y)$ es el trabajo marcado, $I(x,y)$ es el trabajo original y K es la clave.

En la detección ciega la recuperación se realiza sin necesidad de utilizar el trabajo original.

$$\overline{w}_r = f(I'(x,y), K) \tag{2.7}$$

donde: $\overline{w}_r$ es la marca de agua recuperada, $I'(x,y)$ es el trabajo marcado y K es la clave. En el sistema de marcado de agua que proponemos la recuperación es ciega, pues no se requiere de la imagen original para hacer la extracción de la marca de agua.

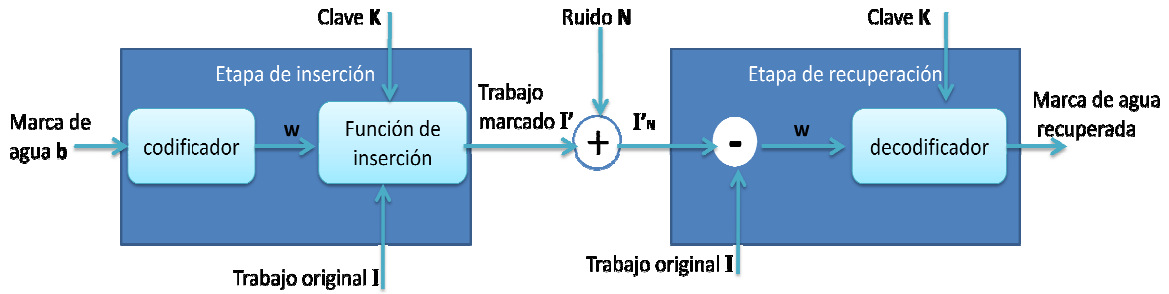

Figura 2.7 Esquema general del sistema de marcado y recuperación de la marca de agua informado.

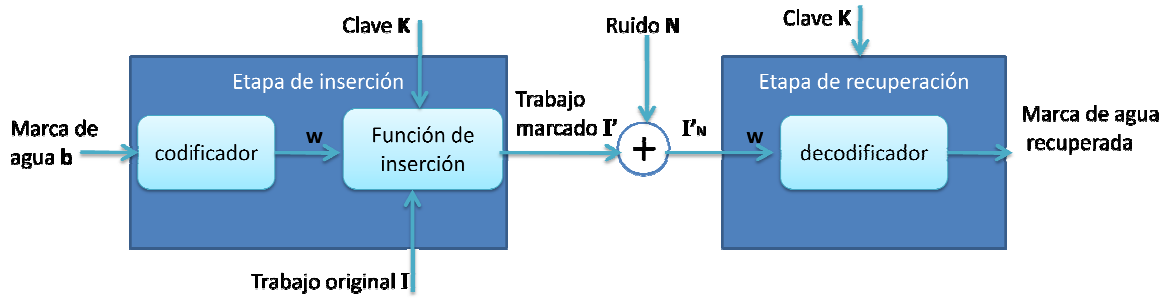

Figura 2.8 Esquema general del sistema de marcado y recuperación de la marca de agua ciego.

Estos esquemas tienen sus variantes como se mostrará más adelante al introducir el diagrama de bloques del algoritmo de inserción y recuperación utilizados en este trabajo.

2.7 Codificación de la marca de agua

Con el propósito de aumentar la robustez del sistema de marcado de agua se puede recurrir a técnicas que impliquen añadir información redundante a la marca de agua que

se desea incrustar. Una de esas técnicas es la codificación de la marca de agua, la cual consiste en transformar el vector binario $\bar{b}$ que contiene los bits de la marca de agua en una versión codificada $\bar{w}$ de mayor longitud. La información redundante adicional permite realizar la detección y corrección de errores en la etapa de extracción de la marca de agua, aunque el costo que se debe pagar por incrustar una mayor cantidad de información es una mayor alteración en el trabajo original.

2.8 Codificación de canal

La codificación de canal consiste en modificar la señal a transmitir de tal forma que sea menos susceptible a sufrir alteraciones en el canal de comunicaciones. Desde un punto de vista de un sistema de transmisión de información, el objetivo es obtener la menor tasa de bits en error posible en el receptor. En el caso de la marca de agua, conseguir una alteración mínima por un posible ataque.

2.9 Detección y corrección de errores

Una forma de disminuir la tasa de bits en error en un sistema de comunicaciones es mediante el uso de códigos que permitan la detección y corrección de errores FEC (*Forward Error Correction)*. Algunos de ellos son los códigos de bloque lineales los cuales mapean un bloque de bits de tamaño k en otro de tamaño n, donde $n > k$. Esto se denota comúnmente como código (n, k), lo cual indica que por cada bloque de k bits que entra al codificador salen n bits codificados. Es decir, se pasa de un espacio vectorial S_k de dimensión k, a otro S_n de dimensión n, mediante el mapeo de cada uno de los 2^k vectores binarios de S_k en su correspondiente vector dentro de S_n. Dado que $n > k$, resulta evidente que el número de vectores posibles (también binarios) dentro de S_n es mayor al de S_k pero en realidad, dado que el mapeo es uno a uno, el número de palabras del código en S_n es igual a 2^k; por lo

tanto, hay un subespacio vectorial dentro de S_n que contiene las 2^k palabras del código. El resto de los vectores que puede haber en S_n ($2^n - 2^k$ vectores) son vectores que se encuentran dispersos entre las 2^k palabras del código. Esos vectores dispersos hacen que las palabras del código se encuentren más distantes entre sí, lo cual implica que se necesita ruido con mayor potencia para "confundir" palabras del código y es por ello que es posible detectar errores y corregirlos hasta cierto límite. Estos códigos son lineales porque cualquier palabra de código se puede obtener a partir de la combinación lineal de palabras base de código dentro de S_n. Adicionalmente a los códigos de bloque correctores de errores están los códigos convolucionales, también lineales.

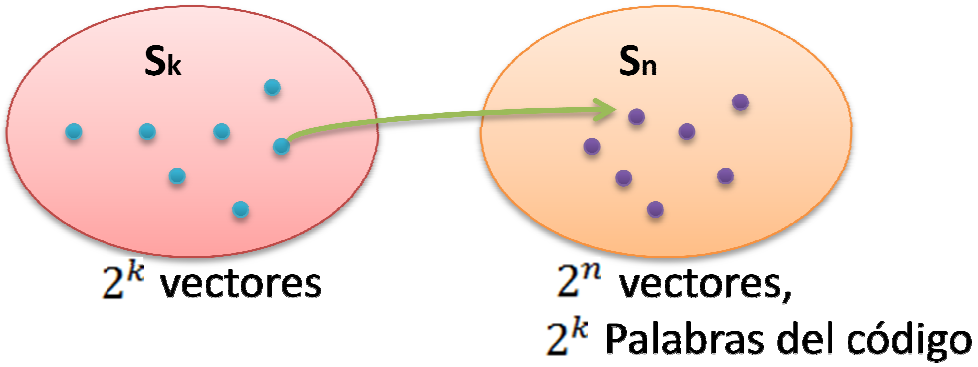

Figura 2.9 Codificación

2.10 Códigos convolucionales

Los códigos convolucionales funcionan con secuencias binarias semi-infinitas tanto a la entrada como a la salida, y por lo general son más rápidos que los códigos de bloque, razón por la cual se prefieren en aplicaciones donde el tiempo de codificación y decodificación es prácticamente en tiempo real respecto a la tasa de transmisión. En este tipo de códigos la codificación se realiza en función del bit que recién acaba de entrar al codificador y algunos bits que han entrado con anterioridad; para ello, el codificador cuenta con registros de corrimiento y sumadores de aritmética módulo-2, los cuales realizan la suma binaria sin acarreo, y producen un efecto de memoria. Cada código

convolucional está definido por su longitud restringida, la tasa del código, y la conexión entre los registros de corrimiento y los sumadores módulo 2. La tasa del código indica el número de bits V que salten del codificador por cada bit que entra: $tasa = \frac{1}{V}$. La longitud restringida K es el número de bits que son considerados para obtener la salida del codificador en un instante de tiempo t, produciendo una memoria de tamaño $K - 1$.

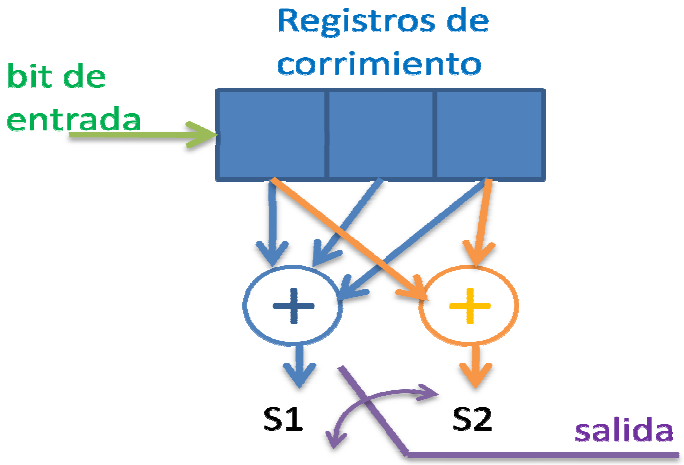

Figura 2.10 Ejemplo de codificador convolucional. $V = 2$, $K = 3$. Por cada bit que entra al codificador salen 2 bits codificados.

La relación entre cada registro de corrimiento y los sumadores está dada por los polinomios generadores del código. Nótese que el número de sumadores del codificador es igual a V. A cada sumador módulo-2 corresponde un polinomio de grado máximo $K - 1$. Para el ejemplo de la Figura 2.10, al sumador $S1$ corresponde el polinomio generador $g_1(x) = 1 + x + x^2$, y al sumador $S2$ el polinomio $g_2(x) = 1 + x^2$. El coeficiente del término de menor orden corresponde a la entrada del registro de corrimiento. También es común expresar los polinomios especificando simplemente sus coeficientes como vectores binarios, es decir, $g_1 = [1\ 1\ 1]$ y $g_2 = [1\ 0\ 1]$.

Es posible representar el codificador mediante diagrama de bloques, diagrama de árbol, diagrama de estados, y enrejado. El diagrama de árbol aunque nos permite seguir la codificación en el tiempo no es muy utilizado por el rápido crecimiento del número de

ramas que se tienen que dibujar para hacerlo. El diagrama de estados es útil para conocer la salida del codificador para cada bit de entrada dependiendo del estado en que se encuentre el registro de corrimientos, pero no nos permite visualizar su evolución en el tiempo. El enrejado es similar al diagrama de estados pero extendido en el tiempo.

2.10.1 Diagrama de estados

Es una representación gráfica de los 2^{K-1} estados en que puede encontrarse los registros de corrimiento del codificador. También muestra la transición de un estado a otro y los bits de salida del codificador dependiendo del bit que entre. Es decir, la entrada de un "1" al codificador se representa con una línea de transición punteada y los bits de salida del codificador se escriben sobre dicha línea; si entra un "0", la línea de transición es continua y se escriben los bits de salida sobre ésta. En la figura 2.11 se muestra el diagrama de estados del codificador convolucional de la figura 2.10.

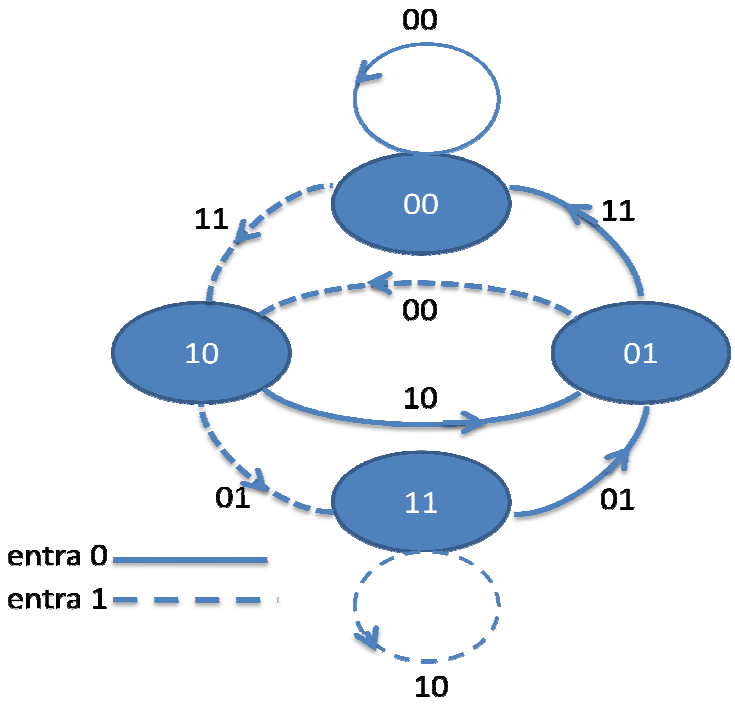

Figura 2.11 Diagrama de estados del código convolucional $V = 2, K = 3$.

2.10.2 Diagrama de trellis o enrejado

El diagrama de *trellis*, también conocido como enrejado, es un diagrama que muestra cómo se pasa de un estado a otro en el tiempo dependiendo del bit de entrada que llega al codificador. Al igual que en el diagrama de estados, la entrada de un "0" al codificador se representa con línea continua y la de un "1" con línea puntada. Los estados son representados por un arreglo horizontal de nodos siendo cada renglón de nodos del enrejado correspondiente a un mismo estado. En la figura 2.12 se muestra como ejemplo el enrejado del código convolucional de la figura 2.10.

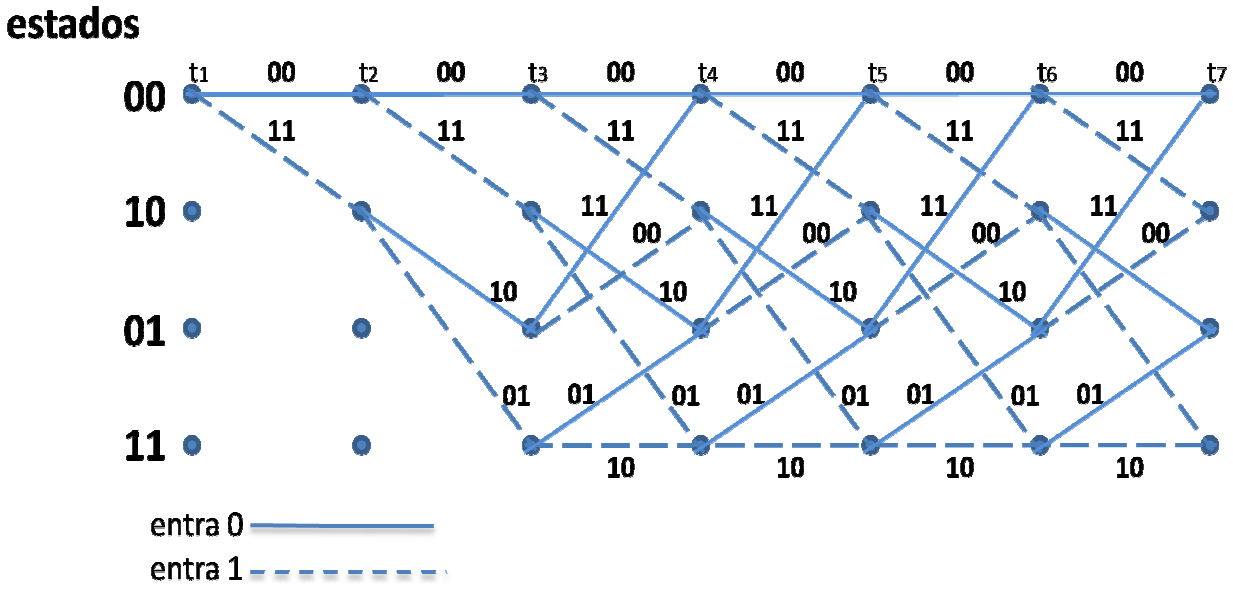

Figura 2.12 Diagrama de *trellis* del código convolucional $V = 2, K = 3$.

Se puede hacer la codificación con el enrejado de la siguiente manera: se parte de un estado inicial en el cual todos los bits del registro de corrimientos del codificador son cero lo cual corresponde al nodo más a la izquierda del primer renglón del enrejado en el tiempo t_1. Posteriormente se va definiendo un camino dependiendo de los bits de la secuencia que entra al codificador. Los bits sobre las líneas del camino definido corresponden a los bits que salen del codificador.

El enrejado resulta muy útil al realizar la decodificación, como se verá más adelante.

2.10.3 Corrección de errores

La corrección de errores es posible gracias a la redundancia que se introduce al hacer la codificación. En los códigos de bloques, como los códigos Hamming o los códigos BCH, se tiene el concepto de distancia mínima $d_{mín}$ (también conocida como distancia de Hamming) la cual está estrechamente relacionada con la capacidad de corregir errores. Tomemos como ejemplo una codificación mediante un código de bloques (n, k). Supóngase un conjunto de vectores correspondiente a las palabras del código dentro del espacio S_n. Los vectores se encuentran al centro de esferas de radio t, como se muestra en la figura 2.13. La separación entre dos de los vectores está dada por el número de bits que tienen diferentes, esto es la distancia de Hamming. A mayor distancia de Hamming entre dos vectores, mayor será su separación.

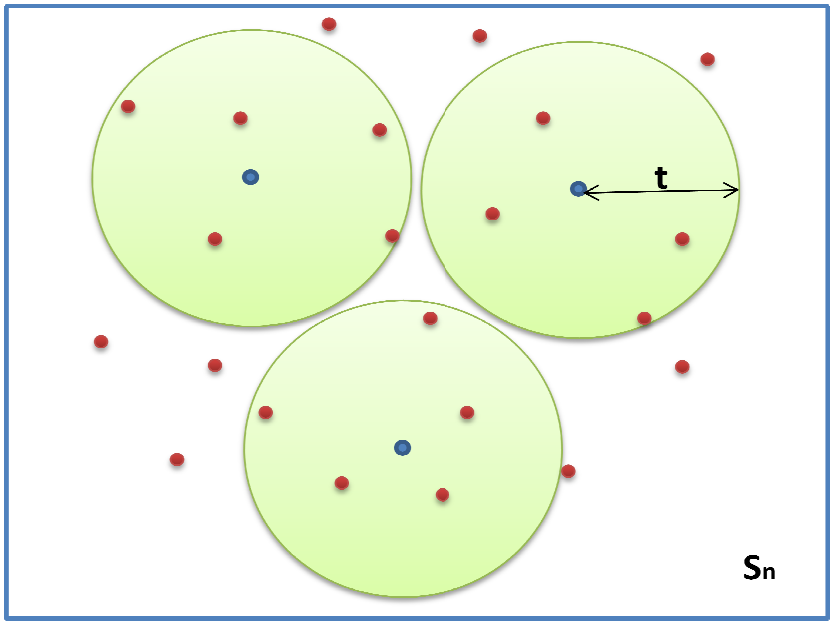

Figura 2.13 En azul palabras del código, en rojo vectores de S_n.

La distancia t máxima del radio de cada esfera sin que se superponga con esferas vecinas se calcula a partir de la distancia mínima que exista entre dos palabras del código. De esta

forma la separación de esferas de igual tamaño en cuyo centro se encuentran las palabras del código está dada por:

$$t = \frac{d_{mín}-1}{2} \tag{2.8}$$

En los códigos de bloques, el valor de t es el número de errores que el decodificador es capaz de corregir en cada bloque.

Si el ruido presente en el canal modifica alguna de las palabras del código, mientras la palabra modificada se encuentre dentro de la esfera de la palabra del código correcta, entonces se podrán corregir esos errores al hacer la decodificación. Para que se tenga error en la decodificación el ruido tendría que sacar una palabra del código fuera de su esfera de pertenencia desplazándola hacia el interior de la esfera correspondiente a otra palabra del código.

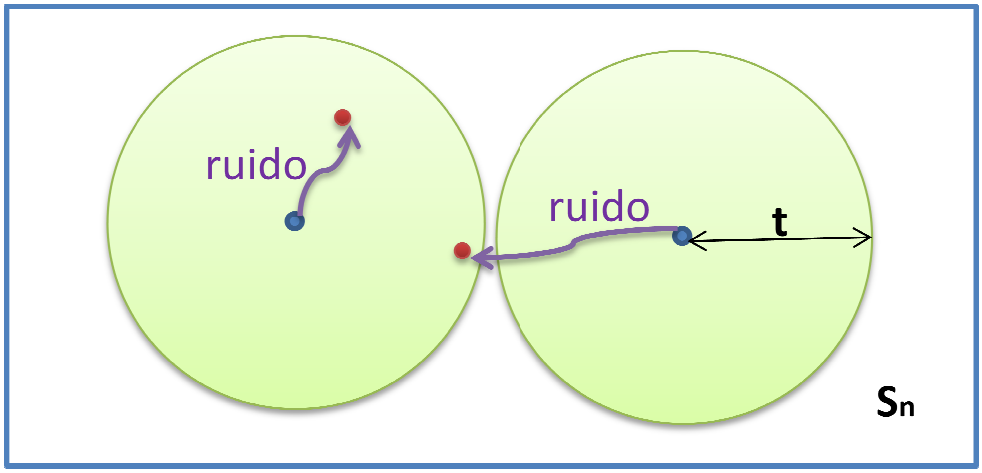

Figura 2.14 En azul palabras del código, en rojo palabras del código luego de transmisión por canal AWGN.

En los códigos convolucionales el equivalente a la distancia mínima se conoce como distancia libre d_{free} y su significado es similar al de la distancia mínima: la separación entre vectores. Como en códigos convolucionales no hay bloques de tamaño fijo definido, se calculan distancias de Hamming acumuladas en el camino del enrejado al hacer la decodificación. La distancia libre se obtiene de la siguiente manera: Se asume que se transmite una secuencia de puros ceros (secuencia nula) y que existe un error en la

secuencia recibida, la distancia libre es la mínima distancia de Hamming acumulada entre el camino de bits codificados con error en el enrejado y el camino correspondiente a la codificación de sólo ceros (secuencia nula). Mientras más grande sea la distancia libre, más robusto será el código convolucional frente al ruido y en nuestro trabajo, más resistente será la marca de agua a los ataques.

Es posible conocer el número de errores que puede corregir un código convolucional de igual forma que se hace con un código de bloque, sólo hay que cambiar $d_{mín}$ por d_{free} en la ecuación 2.8.

$$t = \frac{d_{free}-1}{2} \tag{2.9}$$

Sin embargo, en los códigos convolucionales, dado que no se manejan bloques, no se conoce con exactitud cuántos errores se pueden corregir en un bloque de bits determinado. En la práctica, el valor de t es el número máximo de errores que se pueden corregir en una secuencia de tamaño igual a entre 3 y 5 longitudes restringidas [8]. Es por ello que para poder conocer mejor la capacidad de corregir errores y el rendimiento de los códigos convolucionales resulta útil realizar simulaciones y obtener gráficas que relacionan el BER vs SNR.

2.10.4 Codificación de la marca de agua

El código convolucional propuesto para la codificación de la marca de agua fue tomado de [10], en donde se muestran las características y resultados de simulaciones de 27 códigos convolucionales diferentes. Fue escogido por haber presentado las menores tasas de bit en error en las simulaciones [10]. Es un código de tasa $\frac{1}{3}$ y longitud restringida $K = 5$. Los polinomios generadores son $g_1 = [1\ 1\ 1\ 1\ 1]$, $g_2 = [1\ 1\ 0\ 1\ 1]$ y $g_3 = [1\ 0\ 1\ 0\ 1]$. El codificador se muestra en la figura 2.15.

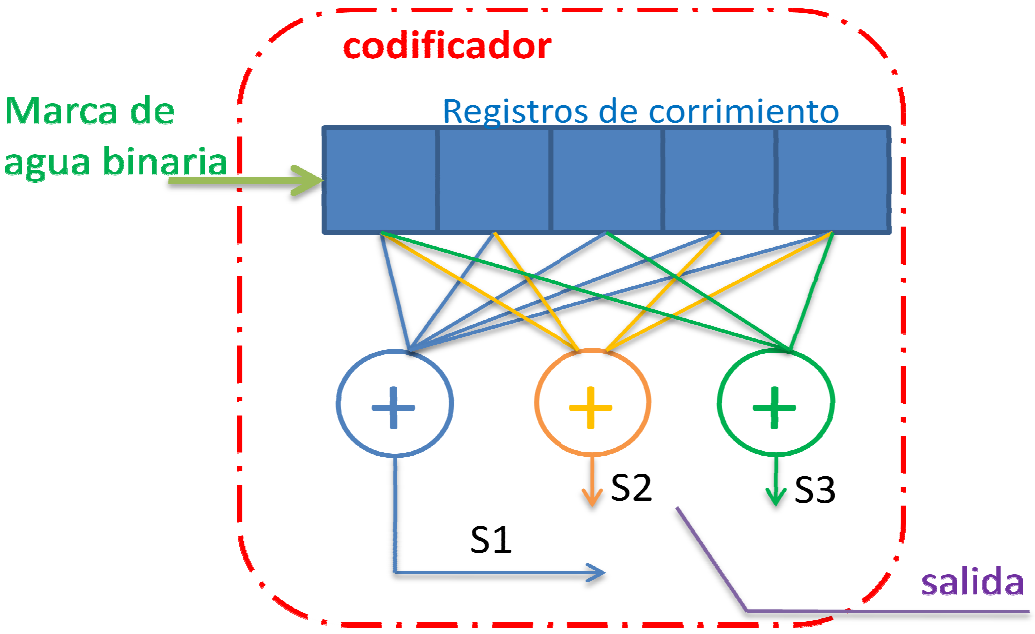

Figura 2.15 Codificador de marca de agua binaria. $V = 3$, $K = 5$.

La distancia libre de este código convolucional es $d_{free} = 12$ [8]. Por lo tanto, el número de errores que puede corregir son 5.

En el proceso de marcado de agua utilizado en esta tesis, la marca de agua binaria es primero codificada utilizando el código convolucional de la figura 2.15 y posteriormente es incrustada utilizando una técnica de espectro disperso, la cual será descrita en el siguiente capítulo.