Esteganografía en zonas ruidosas de la imagen

Daniel Lerch-Hostalot
Universitat Oberta de Catalunya,
Internet Interdisciplinary Institute (IN3),
Estudis d'Informàtica, Multimèdia i Telecomunicació,
Rambla del Poblenou, 156,
08018 Barcelona,
Email: dlerch@uoc.edu

David Megías
Universitat Oberta de Catalunya,
Internet Interdisciplinary Institute (IN3),
Estudis d'Informàtica, Multimèdia i Telecomunicació,
Rambla del Poblenou, 156,
08018 Barcelona,
Email: dmegias@uoc.edu

***Resumen*—La mayor parte del estegoanálisis en el estado del arte se basa en el uso de técnicas de *machine learning*, es decir, en entrenar clasificadores para que sean capaces de diferenciar una imagen portadora de una imagen con mensaje oculto. Las investigaciones realizadas en este campo muestran que las zonas de la imagen más difíciles de modelar y, en consecuencia, aquellas en las cuales es más difícil detectar un mensaje incrustado, son las zonas ruidosas. Estas corresponden a líneas y texturas. En este artículo presentamos un nuevo método de esteganografía que permite ocultar información en dichas zonas, dificultando así su detección. La efectividad del método se ha comprobado usando dos bases de datos de imágenes diferentes y dos estegoanalizadores recientes. Los experimentos demuestran que el algoritmo propuesto mejora significativamente la indetectabilidad estadística respecto al sistema LSB *matching* para la misma capacidad de incrustación.**

***Palabras clave*—Esteganografía, Estegoanálisis.**

I. Introducción

La esteganografía estudia diferentes técnicas para la ocultación de datos en otros objetos, conocidos como objetos portadores. Actualmente, estos objetos portadores suelen ser medios digitales, como por ejemplo imágenes, vídeos o archivos de sonido. No obstante, sin lugar a dudas, el medio más utilizado en la actualizad son las imágenes, por su amplia difusión en Internet.

Uno de los métodos más usados para ocultar información en imágenes de mapas de bits es la sustitución del bit menos significativo (*Least Significant Bit*, LSB). Este método divide el mensaje original en bits y oculta cada uno de ellos en un píxel de la imagen. La variación en el valor del píxel es tan poco significativa que no puede ser detectada visualmente, pero resulta suficiente para ocultar información. Sin embargo, como puede verse en [16], esta técnica presenta algunos inconvenientes. La sustitución del LSB es una operación asimétrica, pues los píxeles con un valor par tenderán a incrementar su valor (cuando se incruste un '1'), mientras que los píxeles con un valor impar tenderán a disminuirla (cuando se incruste un '0'). Esto crea anomalías estadísticas en la imagen, como por ejemplo parejas de barras (frecuencias) que tienden a igualarse en el histograma de luminosidad de la imagen [16]. Finalmente, esta debilidad de la sustitución del LSB ha culminado en ataques como el RS [5] o el SPA [4], los cuales han conseguido detectar la presencia de información oculta incluso cuando el número de bits incrustados apenas alcanza el 3 % del número de píxeles de la imagen. Debido a estos ataques, el método de sustitución del LSB ha dejado de ser considerado como seguro.

El sistema que ha tomado el relevo de la sustitución del LSB es el conocido como LSB *matching* [15]. Este método es muy similar al anterior, pues solo tiene una pequeña diferencia: en lugar de sustituir el valor del LSB directamente por el bit a incrustar, lo que hace es modificarlo sumando o restando uno al valor total del píxel cuando el bit a incrustar no coincide con el LSB del píxel correspondiente. El efecto sobre el LSB es el mismo, así como la dificultad para detectarlo visualmente. Sin embargo, al proceder de esta manera, ya no se trata de una operación asimétrica y no se introducen anomalías estadísticas tan evidentes. De hecho, con este método resulta muy difícil diferenciar un mensaje oculto del ruido existente en todas las imágenes que aparece como consecuencia del proceso de captura.

Para detectar este método de ocultación de información en imágenes los estegoanalistas han recurrido al uso de técnicas de *machine learning* [2]. Para ello es necesario preparar una base de datos de imágenes que se usarán para entrenar un clasificador y verificar que este funciona correctamente. Este clasificador será el encargado de diferenciar las imágenes con mensaje oculto (imágenes esteganográficas) de las imágenes no alteradas (imágenes portadoras). En este tipo de estegoanálisis el trabajo del estegoanalista se basa principalmente en detectar aquellas características de la imagen que son más susceptibles de ser alteradas cuando se oculta información. Estas características son las usadas para entrenar al clasificador. Si bien se han propuesto diferentes métodos de estegoanálisis basados en clasificadores que ofrecen buenos resultados [13], [8], [11], todavía queda mucha investigación para avanzar en este campo.

Una de las lecciones aprendidas en los últimos años de investigación en estegoanálisis usando clasificadores es que existen zonas que son mucho más difíciles de modelar que otras: los bordes y las texturas. Estas zonas contienen mucho ruido y, en ellas, es muy difícil extraer características ade-

Este trabajo está financiado parcialmente por el Ministerio de Economía y Competitividad a través de los proyectos TIN2011-27076-C03-01/02 "CO-PRIVACY" y CONSOLIDER INGENIO 2010 CSD2007-0004 "ARES".

ISBN: 978-84-9717-323-0

cuadas para entrenar al clasificador. Esta es la base que usan algunas técnicas modernas de esteganografía, como por jemplo las presentadas en [14], [9]. En este artículo se presenta una nueva técnica de ocultación de información que usa estas zonas ruidosas de la imagen para incrustar los bits del mensaje.

El resto del artículo se organiza de la manera siguiente. En la Sección II se presenta el método de esteganografía propuesto en el artículo. En la Sección III se analiza experimentalmente el método y se comprueba su indetectabilidad usando software de estegoanálisis. Finalmente, la Sección IV presenta las conclusiones extraídas de este trabajo.

II. Método propuesto

II-A. Motivación

Los métodos modernos de estegoanálisis usan clasificadores para modelar las propiedades estadísticas de las imágenes, pero existen zonas que son especialmente difíciles de modelar, como los bordes o las texturas. Por ello, la investigación en nuevos sistemas de esteganografía se centra, en gran parte, en la construcción de métodos que permitan ocultar información en esas zonas.

Sistemas de estegoanálisis como [13], [8], [11] modelan las características extraídas de la imagen como diferencias entre píxeles vecinos. Tomemos como ejemplo el método basado en patrones de diferencias de píxeles (*patterns of pixel diferences*, PPD) presentado en [11]. En ese artículo se usan cinco píxeles vecinos para modelar la imagen, como se muestra en la Fig.1, tomando uno de ellos como referencia a restar de los demás.

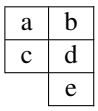


Figura 1: Extracción de características basadas en bloques PPD

De esta forma pueden obtenerse vectores formados por cuatro posibles diferencias. Por ejemplo, si tomamos b como base, podemos obtener una representación en cuatro dimensiones: $v = [a-b, c-b, d-b, e-b]^{\mathrm{t}}$, donde $[\cdot]^{\mathrm{t}}$ denota el operador de transposición de un vector (o una matriz).

Suponiendo el caso más sencillo, es decir, usando imágenes en escala de grises con una profundidad de color de 8 bits, cada píxel puede tomar un valor de 0 a 255. Por lo tanto, en el peor de los casos, la diferencia entre dos píxeles vecinos será de 255. Así pues, con el modelo presentado, podrían generarse hasta 255^4 características, lo que resulta numéricamente impracticable para los clasificadores actuales. Con los sistemas propuestos en [13] y [8] la situación es similar. No obstante, no es habitual que un píxel con valor 0 sea el vecino de un pixel con valor 255, dado que en las imágenes el valor de los píxeles suele cambiar en forma de degradado. Por lo tanto, ignorar diferencias muy grandes entre valores vecinos no suele perjudicar a los sistemas de estegoanálisis. Es por ello que se utiliza un parámetro para reducir el número de características. Por ejemplo, en [11] se sugiere el uso de un umbral $S = 4$. De esta manera, en lugar de obtener 255^4 características, se obtienen 4^4, que es una cifra mucho más manejable. Con esta aproximación lo métodos de estegoanálisis pueden atacar el problema sin que la explosión en el número de características les impida modelar la imagen.

Sin embargo, este no es el único motivo para el uso de un umbral para reducir el número de dimensiones. Otro problema asociado a la dimensionalidad, y que perjudica seriamente al estegoanálisis, es el de la obtención de muestras insuficientes. Este tipo de problema, detectado previamente en estegoanálisis [7], se produce al usar modelos de grandes dimensiones. Cuantas más dimensiones tenga el modelo, más difícil es encontrar muestras en la imagen para todas las posibilidades que ofrece. Durante la extracción de características, se reparten todas las muestras extraídas de la imagen entre cada uno de los patrones de los que dispone el modelo. En el caso de PPD, por ejemplo, existen T^4 patrones, la frecuencia de los cuales dependerá de la imagen y de su contenido. Dado que el número total de muestras es fijo y a medida que crece el valor de T aumenta el número de patrones, la frecuencia de cada patrón será cada vez más pequeña. Los patrones menos frecuentes serán los primeros en llegar a frecuencias tan bajas que su valor no será representativo y perjudicarán al entrenamiento del clasificador. Por este motivo existe un límite en el valor del umbral a partir del cual los métodos de estegoanálisis dejan de ser efectivos. El método que se propone en este artículo, pretende explotar esta debilidad.

II-B. Zonas de inserción

Como se ha describe en el apartado anterior, el objetivo del método presentado en este artículo es ocultar la información en las zonas más difíciles de modelar de la imagen. El problema que se presenta cuando se desea ocultar información únicamente en unas zonas concretas de la imagen es cómo comunicar al receptor del mensaje (de la imagen) en qué zonas debe leer y en qué zonas no. Si se desarrolla un procedimiento para identificar las zonas ruidosas, estas pueden cambiar (dejar de ser ruidosas) al ocultar información, por lo que el receptor puede acabar leyendo en zonas donde no hay mensaje e ignorando zonas donde sí lo había.

Un enfoque válido, tomado en los métodos [14], [9], es el uso de *Wet Paper Codes* (WPC) [6]. Estos métodos, que son muy adecuados para el problema presentado, son relativamente complejos, lo que implica un procesamiento muy lento en la inserción del mensaje. En este trabajo se expone un procedimiento alternativo, que resulta muy rápido y sencillo en comparación con el uso de WPC.

Para detectar las zonas de inserción se establecerá un umbral T que nos indicará las zonas difíciles de modelar, de la misma forma que lo hacen los sistemas de estegoanálisis. Se agruparan los píxeles en parejas de píxeles vecinos (a, b),de manera que solo los tomaremos en consideración para ocultar información si su diferencia es mayor o igual al umbral T, es decir si $|a-b| \geq T$.

(a) Imagen Lenna en escala de grises (b) Zonas de ocultación para $T = 4$ (c) Zonas de ocultación para $T = 10$

Figura 2: Imagen Lenna y zonas donde se oculta la información para diferentes valores de T

En la Fig.2(b) se muestran, con píxeles negros, las zonas donde se ocultaría la información en caso de usar $T = 4$ para la imagen Lenna de la Fig.2(a). Análogamente, la Fig.2(c) muestra las zonas de ocultación de información para el caso $T = 10$. En ellas podemos ver cómo se evitan las zonas más uniformes de la imagen, mientras que los bordes, principalmente, y algunas texturas son las zonas que se usan para ocultar información.

Para ocultar información el procedimiento consiste en recorrer la imagen tomando cada vez una pareja diferente de píxeles vecinos. Las parejas se forman sin solapamiento, de manera que, dados cuatro píxeles vecinos, (a, b, c, d), se formarán las parejas (a, b) y (c, d), mientras que la pareja (b, c) no se tendrá en consideración. Las parejas así formadas no se usarán para incrustar información si su diferencia esta por debajo de umbral T. La imagen se puede recorrer tomando las parejas de forma horizontal o vertical, aunque esto no es determinante para el funcionamiento del método propuesto.

II-C. Procedimiento de incrustación

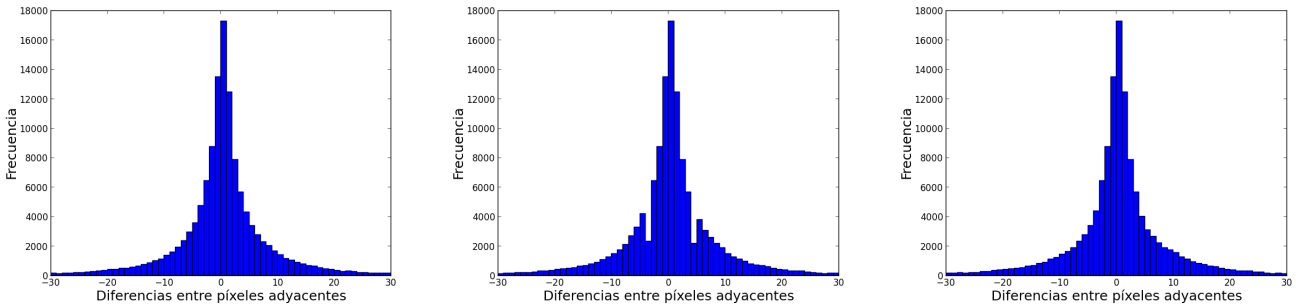
El método propuesto usa cada pareja que supera el umbral para ocultar un bit. Concretamente se oculta alterando el píxel de la izquierda, es decir, el etiquetado como a de la pareja (a, b). Para ello, se usa el LSB de a como bit de información, dejándolo tal y como está si su valor es igual al del bit del mensaje que se quiere ocultar y modificándolo si su valor no coincide. Esta modificación se realizará siempre incrementando la diferencia entre los valores de los píxeles de la pareja. Así pues, a', el nuevo valor de a, vendrá determinado por la ecuación siguiente:

$$a' = \begin{cases} a, & \text{si } a \bmod 2 = m, \\ a+1, & \text{si } a > b \\ a-1, & \text{si } a < b. \end{cases}$$

La idea es incrementar siempre la diferencia entre los píxeles de la pareja, nunca disminuirla. El motivo es que incrementar o disminuir aleatoriamente, de forma similar a como se hace en LSB *matching*, llevaría en el caso $|a - b| = T$ a dejar de cumplir el umbral establecido para algunas parejas, que pasarían a tener una diferencia $T - 1$. Al no cumplir el umbral de lo que consideramos un píxel adecuado para ocultar información, el receptor no sabría que debe leer información de él y se perdería la información oculta en ese píxel.

Lógicamente, esta operación para ocultar información introduce una anomalía estadística, pues no se aplica de forma equitativa sobre todas las parejas de píxeles. Al incrementar la diferencia entre parejas con un valor superior a $T+1$, parte de esas parejas (en las que se quiere ocultar un bit diferente al LSB de a) se convierten en parejas con diferencia $T + 2$. Análogamente, algunas parejas con diferencia $T + 2$ pasan a tener diferencia $T+3$ y así sucesivamente. En general, aunque varias parejas con diferencia $T + n$ pasan a tener diferencia $T + n + 1$, este hecho se ve compensado por el número de parejas nuevas con diferencia $T+n$ que aparecen al incrustar información en parejas con diferencia $T + n - 1$. Pero hay un caso especial, el de las parejas con diferencia T, pues mientras que parte de estas pasan a tener diferencia $T + 1$, el número de parejas con diferencia T no se ve retroalimentado y solo decrece. Esta anomalía puede observarse en el histograma de los valores de las diferencias entre píxeles adyacentes representado en la Fig.3(b), en comparación con el histograma de la imagen original que aparece en la Fig.3(a).

Para eliminar esta anomalía en el histograma, se puede repartir la responsabilidad de ser la primera pareja (la que tiene diferencia igual a T) entre diferentes parejas. De esta forma, se consigue que las barras del histograma correspondientes no decrezcan lo suficiente como para generar una anomalía. Para ello, se usa un valor de T dinámico, que dependerá del pixel que se esté modificando. La idea es inicializar un generador de números pseudoaleatorios (*Pseudo-Random Number Generator*, PRNG) con una semilla (por ejemplo una contraseña) que deberán conocer tanto el emisor del mensaje como el receptor. Este PRNG se utiliza para generar una secuencia de valores dinámicos para el umbral T. Se usa un rango de



(a) Imagen original

(b) Imagen esteganográfica con $T = 4$

(c) Imagen esteganográfica con $T_{\text{mín}} = 4$ y $T_{\text{máx}} = 9$

Figura 3: Histograma de diferencias entre parejas de píxeles adyacentes para la imagen original, la imagen esteganográfica con $T = 4$ y la imagen esteganográfica con umbral dinámico entre $T_{\text{mín}} = 4$ y $T_{\text{máx}} = 9$

valores de T entre un mínimo, $T_{\text{mín}}$, y un máximo, $T_{\text{máx}}$. De esta manera los valores dinámicos de T se generarán con la función $T = \text{PRNG}(T_{\text{mín}}, T_{\text{máx}})$, que devuelve un número pseudoaleatorio con distribución uniforme en el intervalo $[T_{\text{mín}}, T_{\text{máx}}]$. De esta manera, como se puede apreciar en la Fig.3(c), la anomalía en el histograma desaparece.

El uso del PRNG para seleccionar un valor dinámico del umbral no solo sirve para eliminar la anomalía estadística, sino que además ofrece una capa de seguridad adicional que dificulta el estegoanálisis, dado que no se puede saber con exactitud en qué píxeles se ha ocultado información sin disponer de la semilla.

II-D. Algoritmos de incrustación y de extracción

Tal y como se explica en los apartados anteriores, ya tenemos todas las piezas necesarias para el algoritmo de inserción de datos de la imagen (Algoritmo 1).

La extracción de datos es similar. Basta con inicializar el PRNG, recorrer la imagen extrayendo parejas de píxeles e ir leyendo los LSB del primer píxel de cada pareja que cumple con el umbral T (Algoritmo 2).

III. Resultados experimentales

El método se ha verificado con dos sistemas de estegoanálisis: PPD [11] y SPAM [13] (en la versión más efectiva de este, que usa características de segundo orden). Estos sistemas permiten extraer características de las imágenes. Como clasificador, se ha usado una implementación de una *Support Vector Machine* (SVM) [3], por ser uno de los clasificadores que ofrece mejores resultados en estegoanálisis.

La SVM debe ser ajustada para que proporcione unos resultados óptimos. Concretamente, es necesario seleccionar valores para los parámetros C y γ. Estos valores serán escogidos para dar al clasificador la capacidad de generalizar. Para escoger dichos parámetros, se ha seguido el proceso especificado en [10], es decir, realizando una validación cruzada en el conjunto de entrenamiento de todos los posibles valores de los parámetros C y γ que se especifican a continuación:

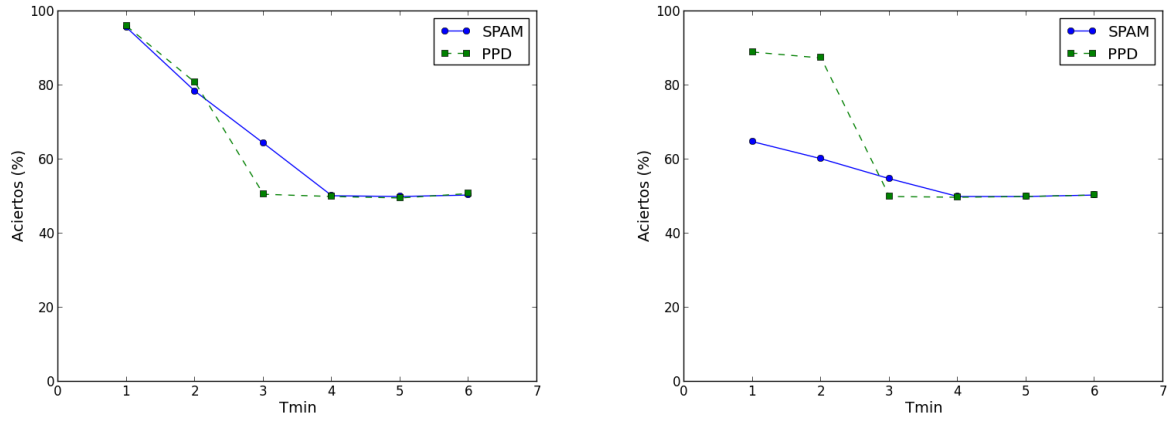
Algorithm 1 Ocultar mensaje

Input: $M, I, \text{Seed}, T_{\text{mín}}, T_{\text{máx}}$
M: Mensaje a ocultar
I: Matriz $[1..H, 1..W]$ que contiene la imagen original
Seed: Semilla del generador PRNG
$T_{\text{mín}}$: Valor mínimo para el cálculo de T
$T_{\text{máx}}$: Valor máximo para el cálculo de T
Output: I'
I': Matriz que contiene la imagen con el mensaje oculto

1: InitializePRNG(Seed)
2: **for all** $i \in [1, H]$ **do**
3: **for all** odd $j \in [1, W-1]$ **do**
4: $T \leftarrow \text{PRNG}(T_{\text{mín}}, T_{\text{máx}})$
5: $a \leftarrow I[i, j]$
6: $b \leftarrow I[i, j+1]$
7: **if** $|a - b| \geq T$ **then**
8: $m \leftarrow \text{nextBit}(M)$
9: $I'[i,j] \leftarrow \begin{cases} a, & \text{if } a \bmod 2 = m, \\ a+1, & \text{if } a > b, \\ a-1, & \text{if } a < b. \end{cases}$
10: **end if**
11: **end for**
12: **end for**

$$C \in \left\{2^{-5}, 2^{-3}, 2^{-1}, 2^{1}, 2^{3}, \ldots, 2^{15}\right\},$$
$$\gamma \in \left\{2^{-15}, 2^{-13}, 2^{-11}, \ldots, 2^{-1}, 2^{1}, 2^{3}\right\}.$$

Los experimentos se han realizado con la base de datos BOSS, presentada en [1], por ser una de las más usadas en esteganografía, y también en la base de datos pública NRCS [12], por disponer de imágenes de alta resolución muy ruidosas, significativamente diferentes de las de BOSS. Para cada base de datos, se han creado dos grupos de imágenes, uno que se usa como conjunto de entrenamiento y otro que se usa como conjunto de verificación. Cada uno de ellos esta formado por 500 imágenes, 250 de ellas sin incrustar (portadoras) y



(a) Base de datos BOSS

(b) Base de datos NRCS

Figura 4: Porcentajes de detección correcta en función de $T_{\text{mín}}$, usando las bases de datos de imágenes BOSS y NRCS

Algorithm 2 Extraer mensaje

Input: I, Seed, $T_{\text{mín}}, T_{\text{máx}}$
I: Matriz $[1..H, 1..W]$ que contiene la imagen con el mensaje oculto
Seed: Semilla del generador PRNG
Output: M
M: Mensaje extraído

```
InitializePRNG(Seed)
for all i ∈ [1, H] do
    for all odd j ∈ [1, W − 1] do
        T ← PRNG(T_mín, T_máx)
        a ← I[i, j]
        b ← I[i, j + 1]
        if |a − b| ≥ T then
            M ← addBit(M, a)
        end if
    end for
end for
```

las otras 250 con información incrustada (imágenes esteganográficas). El umbral usado por defecto en PPD es $T = 4$ (parámetro S especificado en [11], mientras que en SPAM es de $T = 3$. Los experimentos se han diseñado para verificar que, marcando con umbrales superiores a los establecidos por las herramientas de estegoanálisis, el método presentado no se detecta. Se ha incrustado información en las imágenes usando diferentes valores para $T_{\text{mín}}$ y $T_{\text{máx}}$, tal y como se muestra en el Cuadro I.

Como se puede ver en la Fig.4, el porcentaje de detección cae al 50 % (equivalente a decisión aleatoria, o sea, a no detección) aproximadamente al llegar a $T_{\text{mín}} = 4$. En los gráficos se aprecia como los métodos de estegoanálisis fallan cuando la información esta oculta en zonas que no pueden modelar. Los experimentos se han realizado sobre dos bases de datos de imágenes muy diferentes, y en ambos casos, el algoritmo propuesto no es detectado cuando $T_{\text{mín}}$ supera el umbral usado por los métodos de estegoanálisis.

Cuadro I: Valores mínimos y máximos del umbral usados en los experimentos

$T_{\text{mín}}$	$T_{\text{máx}}$
1	6
2	7
3	8
4	9
5	10
6	11

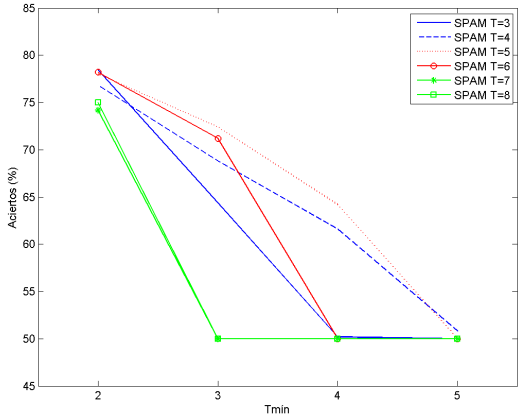


Figura 5: Porcentaje de detección correcta en función de $T_{\text{mín}}$, usando la base de datos de imágenes BOSS y diferentes valores de T, para el método de estegoanálisis SPAM

Sin embargo, podría parecer que el motivo por el que los métodos de estegoanálisis no detectan al método de esteganografía propuesto es por la elección de un umbral superior

al que ellos usan y que bastaría con subir también el umbral usado en estos métodos. Pero esto no es así dado que, si se incrementa el umbral, aumenta el número de dimensiones y aparecen combinaciones para las que no existen muestras o para los que existen muy pocas. Esto, como se ha comenta en la Sección II, empeora considerablemente los resultados del estegoanálisis. En la Fig.5 se puede observar que no existe ningún umbral T que permita detectar el método propuesto con SPAM si $T_{\text{mín}} \geq 5$. Lo mismo sucede para PPD si $T_{\text{mín}} \geq 4$.

La cantidad de información que puede incrustarse (es decir, la capacidad del método) en una imagen depende de las zonas ruidosas de esta, por lo que no es sencillo de determinar *a priori*. A nivel orientativo, usando $T_{\text{mín}} = 4$ y $T_{\text{máx}} = 9$ en las imágenes de BOSS, se ha realizado la inserción con una ratio media del 9 %, es decir incrustando un bit en el 9 % de los píxeles (0,09 bits por píxel). En las imágenes de NRCS la ratio de incrustación es del 13 %. Para mostrar la efectividad del sistema propuesto, se han comparado los resultados de indetectabilidad de este con los obtenidos usando la esteganografía LSB *matching* tradicional [15]. Para ello, se han usado los estegoanalizadores PPD y SPAM con el objetivo de calcular los porcentajes de detección cuando se incrusta en LSB *matching* usando una ratio del 9 % en BOSS y del 13 % en NRCS. De esta manera se puede realizar una comparación en igualdad de condiciones en cuanto a capacidad se refiere. Los resultados de detección se muestran en II. Como se puede observar, para las mismas ratios de inserción que no se detectan (porcentaje de aciertos del 50 %) con el algoritmo presentado, la esteganografía LSB *matching* se detecta con los estegoanalizadores SPAM y PPD (porcentaje de aciertos superior al 50 %).

Cuadro II: Detección de la esteganografía LSB *matching*, para la misma capacidad que el método propuesto, usando PPD y SPAM

Base de datos	Método de detección	Porcentaje de aciertos
BOSS	SPAM	85.00 %
BOSS	PPD	81.60 %
NRCS	SPAM	58.00 %
NRCS	PPD	64.00 %

IV. Conclusión

En este artículo se presenta un nuevo método para ocultar información en zonas difíciles de modelar de la imagen. Para detectar estas zonas, el método propuesto intenta explotar dos debilidades de los sistemas de estegoanálisis existentes: el crecimiento exponencial del número de características y la imposibilidad de extraer información útil de patrones con pocas muestras. Ambas debilidades tienen en común un umbral T, usado como base en el método presentado.

Por otra parte, se trata de un método que no requiere de ningún cálculo complejo, a diferencia de otros que persiguen objetivos similares, como los basados en WPC, por lo que es adecuado para entornos en los que la velocidad de ejecución o el rendimiento sean un factor clave. Los resultados muestran la indetectabilidad del método ante dos sistemas de estegoanálisis: PPD [11] y SPAM [13], viendo como la selección de un umbral T adecuado es suficiente para eludir la detección. También se comprueba que el ajuste del parámetro T en los métodos de estegoanálisis no permite la detección del método propuesto.

En futuros trabajos sería interesante estudiar si existen otros modelos similares que tengan en cuenta grupos de píxeles mayores que una pareja y si esto puede mejorar el algoritmo. Además, sería recomenable realizar un estudio teórico para el cálculo del valor óptimo de T.

Referencias

[1] T. Filler, T. Pevný, and P. Bas, *Break our steganographic system (BOSS)*, 2010. [Online]. Disponible: http://exile.felk.cvut.cz/boss/ [Accedido el 24 de julio de 2014].

[2] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics Series)*. New York, Secaucus, NJ, USA: Springer, 2006.

[3] C.-C. Chang and C.-J. Lin, "LIBSVM - A Library for Support Vector Machine," [Online]. Disponible: http://www.csie.ntu.edu.tw/~cjlin/libsvm [Accedido el 24 de julio de 2014].

[4] S. Domitrescu, X. Wu and N. D. Memon, "On Steganalysis of Random LSB Embedding in Continuous-tone Images," In *Proc. International Conference on Image Processing, ICIP 2002*, Rochester, NY, USA: IEEE, pp. 324-339.

[5] J. Fridrich, M. Goljan and R. Du, "Detecting LSB steganography in color and grayscale Images," In *Proc. ACM Workshop on Multimedia and Security*, Ottawa, Canada: ACM, pp. 22-28, 2001.

[6] J. Fridrich et al., "Writing on Wet Paper," *IEEE Trans. on Signal Processing*, vol. 53, no. 10, Oct. 2005, pp. 3923-3935.

[7] J. Fridrich, J. Kodovský, V. Holub, M. Goljan. "Breaking HUGO - the process discovery ," *Information Hiding, 13th International Workshop, Lecture Notes in Computer Science*.

[8] J. Fridrich and J. Kodovský, "Rich Models for Steganalysis of Digital Images," *IEEE Trans. on Information Forensics and Security*, vol. 7, no. 3, June 2012, pp. 868-882.

[9] V. Holub and J. Fridrich, "Designing Steganographic Distortion Using Directional Filters," In: *Proc. IEEE Workshop on Information Forensic and Security (WIFS)*, Tenerife, Spain: IEEE, pp. 234-239, 2012.

[10] C. W. Hsu, C. C. Chang, and C.J. Lin, "A practical guide to support vector classification," Department of Computer Science, National Taiwan University, 2003. [Online]. Disponible: http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf [Accedido el 24 de julio de 2014].

[11] D. Lerch-Hostalot and D. Megías, "LSB matching steganalysis based on patterns of pixel differences and random embedding," *Computers & Security*, vol. 32, Feb. 2013, pp. 192-206.

[12] National Resource Conservation Service, *NRCS Photo Gallery*, [Online]. Disponible: http://photogallery.nrcs.usda.gov [Accedido el 24 de julio de 2014].

[13] T. Pevný, P. Bas and J. Fridrich, "Steganalysis by Subtractive Pixel Adjacency Matrix," In *Proc. ACM Multimedia and Security Workshop*, Princeton, NJ, USA: ACM, pp. 75-84, 2009.

[14] T. Pevný, T. Filler, P. Bas, *Using High-Dimensional Image Models to Perform Highly Undetectable Steganography*. Information Hiding. Lecture Notes in Computer Science Volume 6387, 2010, pp 161-177.

[15] T. Sharp, "An Implementation of Key-Based Digital Signal Steganography," In *Information Hiding, Lecture Notes in Computer Science*, vol. 2137, Berlin-Heidelberg, Germany: Springer, 2001, pp. 13-26.

[16] A. Westfeld and A. Pfitzmann, "Attacks on Steganographic Systems," In *Information Hiding, Lecture Notes in Computer Science Volume*, vol. 1768, Berlin-Heidelberg, Germany: Springer, 2000, pp. 61-76