Nuevo algoritmo para la detección de bordes en imágenes para esteganografía

Pablo Martí Méndez Naranjo, Henry Mauricio Villa Yánez, Andrés Santiago Cisneros Barahona

Universidad Nacional de Chimborazo, Avda. Antonio José de Sucre, Km. 1.5 vía a Guano & Escuela Superior Politécnica de Chimborazo, Km 1.5 Panamericana Sur, Riobamba, Ecuador.

Autores para correspondencia: {pmendez, hvilla, ascisneros}@unach.edu.ec

Fecha de recepción: 11 de abril 2017 - Fecha de aceptación: 2 de agosto 2017

ABSTRACT

The present research investigation addresses information security in the area of Software Engineering. Using steganography and cryptography, an improvement was proposed to the Canny Edge detection algorithm to hide information in a multimedia environment, encrypting the message with the symmetric cryptographic algorithm Advanced Encryption Standard (AES) to increase security. Netbeans was applied as the development environment and the following tools to perform the tests on the images: IonForge ImageDiff to compare pixel to pixel differences, Beyond Compare to compare hex code, StegSecret to perform test steganos and Digital Invisible Ink Toolkit to perform Benchmark tests. Two prototypes were developed: in Prototype I the standard Canny Edge detection algorithm was used, and in Prototype II the new proposal for improvement of the Canny Edge detection algorithm. Both prototypes were incorporated in the AES symmetric cryptographic algorithm. Results revealed that Prototype II performs better because the information incorporated in the multimedia environment is more diffuse, resistant to the analysis, and the results of the metrics related to the quality of the image Peak Signal To Noise Ratio (PSNR) Mean Square Error (MSE) are more optimal.

Keywords: Advanced Encryption Standard (AES), Canny Edge, computer security.

RESUMEN

La presente investigación corresponde al tipo de track científico, del área de Ingeniería de Software referente a la seguridad de la información. Utilizando la esteganografía y la criptografía se propuso una mejora al algoritmo de detección Canny Edge para ocultar información en un medio multimedia, cifrando el mensaje con el algoritmo criptográfico simétrico Advanced Encryption Standard (AES) para incrementar la seguridad. Se utilizó Netbeans como ambiente de desarrollo y las siguientes herramientas para realizar las pruebas en las imágenes: IonForge ImageDiff para comparar pixel a pixel las diferencias, Beyond Compare para comparar el código hexadecimal, StegSecret para realizar pruebas de estegoanálisis y Digital Invisible Ink Toolkit para realizar pruebas de benchmark. Se desarrolló dos prototipos: en el Prototipo I se utilizó el algoritmo de detección Canny Edge estándar y en el Prototipo II se utilizó la nueva propuesta de mejora del algoritmo de detección Canny Edge, a los dos prototipos se les incorporó el algoritmo criptográfico simétrico AES. De los resultados obtenidos de las pruebas realizadas, se concluye que el Prototipo II es mejor debido a que la información incorporada en el medio multimedia es más difusa, es resistente a estegoanálisis y los resultados de las métricas relacionadas a la calidad de la imagen Peak Signal to Noise Ratio (PSNR) Mean Square Error (MSE) son más óptimas.

Palabras clave: Advanced Encryption Standard (AES), Canny Edge, seguridad informática.

1. INTRODUCCIÓN

La seguridad en el envío de la información por canales de comunicación inseguros es de mucha importancia para disminuir la vulnerabilidad contra posibles ataques realizados para conseguirla, debido a este problema es necesario garantizar la seguridad informática combinando los campos de la esteganografía y criptografía para complementar sus fortalezas (Gaba & Kumar, 2013).

La esteganografía trata del estudio de técnicas para ocultar información tras un medio multimedia cumpliendo los parámetros fundamentales que son capacidad, imperceptibilidad y robustez (Jabbar, Alaa, Sahib & Zamani, 2013), esta técnica puede ser utilizada en varios ámbitos relacionados a la seguridad informática y poder enviar información que pase inadvertida entre el emisor y receptor (Rodríguez, Navas & Eterovic, 2014). Puede aplicarse a varios medios multimedia como documentos, audios, imágenes, videos, entre otros. Para la investigación realizada se ha utilizado formato de imagen de tipo mapa de bit (BMP). La técnica más utilizada en la esteganografía en imágenes es la del Bit Menos Significativo (Least Significant Bit – LSB) que reemplaza el último bit menos significativo de cada byte del portador con los bits del mensaje que se desea ocultar, de tal forma que exista una mínima distorsión visual para el ojo humano, el resultado final es una imagen esteganografiada con la información embebida en ella. La limitación del método es la capacidad de almacenar la información del mensaje en relación al tamaño de la imagen, para lo cual se podría utilizar hasta los 2 últimos bits menos significativos (Rodríguez & Navas, 2016).

El uso de determinados filtros en las imágenes permite detectar discontinuidades existentes en las imágenes como por ejemplo determinadas líneas, secciones de una imagen o sus bordes. La forma más eficiente de detectar las discontinuidades más relevantes en los niveles de grises entre los pixeles es resaltando los bordes de la imagen, para lo cual el algoritmo Canny es el más utilizado ya que utiliza el coeficiente de escala espacial fijo del filtro de Gauss y los valores empíricos de los umbrales altos y bajos (Qiang, Guoying & Hongmei, 2016). El principal problema es que el método Canny Edge estándar tiene su funcionamiento definido, en el cual se define los bordes de la imagen y se ubica la información en los primeros pixeles del borde recorriendo la imagen de izquierda a derecha y de arriba hacia abajo, con lo cual se podría obtener la información al conocer su conducta, por lo que se plantea una nueva forma de funcionamiento.

El estegoanálisis permite detectar mensajes ocultos en los medios multimedia usando la esteganografía, se considera que un sistema esteganográfico es vulnerado cuando se determina la existencia información oculta en la imagen (Lerch-Hostalot & Megías, 2014).

La criptografía es el arte de utilizar procedimientos para transformar datos en criptogramas, el origen cifra el mensaje y realizando el proceso inverso, el destino puede descifrarlos con la clave respectiva (simétrica o asimétrica) para obtener el mensaje original. Las principales propiedades de la criptografía son: integridad, confidencialidad, autenticación, vinculación (Saini & Verma, 2013). El algoritmo Advanced Encryption Standard (AES) es un tipo de criptografía simétrica que utiliza la misma clave de 128 bits, 192 bits o 256 bits para el cifrado y descifrado de los datos (Nurhayati & Ahmad, 2015). El criptoanálisis es una parte de la criptografía que estudia sistemas criptográficos para encontrar debilidades y vulnerarlas para obtener la información cifrada (Comunidad OWASP, 2009). Las principales métricas para medir la calidad de las imágenes son Peak Signal to Noise Ratio (PSNR) que mide la señal de ruido y Mean Square Error (MSE) que mide el promedio de errores de la imagen (Gaba & Kumar, 2013).

No se conoce las formas de optimizar el proceso de funcionamiento de los métodos esteganográficos existentes, de tal forma de que el mensaje pase desapercibido y no sea descubierto. El objetivo de la presente investigación es proponer un nuevo método para ocultar información basado en el algoritmo de detección Canny Edge, con el proposito de difuminar la información cifrada que será embebida en toda la imagen, mejorando parámetros de calidad de imagen, obteniendo valores altos de PSNR y valores bajos de MSE, para que la modificación realizada en el embebido de la información sea imperceptible.

En la investigación de Singla & Juneja (2014), se usa la esteganografía para ocultar información de terceras personas no autorizadas para conocer la información utilizando otras áreas de la imagen, además combinando técnica de bordes Canny, fuzzy y LSB para embeber la información en la imagen ocupando 1 bit de rojo, 4 bits de verde y 8 bits de azul. En la investigación de Mishra & Bhanodiya (2015) se

comprimen los datos utilizando el algoritmo LAW para disminuir el tamaño, posteriormente se cifra y oculta la información en un medio digital, para incrementar la seguridad utiliza el algoritmo de borde Canny para ocultar la información en los bordes detectados de la imagen y embeberlos con funciones hash. Se realizan pruebas con el programa Matlab y claves fuertes para demostrar la robustez y seguridad de la propuesta planteada. En la investigación de Nurhayati & Ahmad (2015), el origen cifra la información y la embebe en un medio multimedia y el destino lo descifra utilizando el proceso inverso extrayéndola y descifrándola, con lo que se obtiene el mensaje original. Se realizan pruebas utilizando el programa Matlab para analizar la calidad de las imágenes con la propuesta planteada. Estos estudios han permitieron analizar el conocimiento existente y determinar las ventajas, desventajas y funcionamiento de la esteganografía y criptografía para mejorar la seguridad de la información.

2. MATERIALES Y MÉTODOS

Para la presente investigación, se realizaron las siguientes actividades: se desarrolló el algoritmo esteganográfico Canny Edge estándar, se desarrolló una propuesta para el nuevo algoritmo esteganográfico, desarrolló del algoritmo criptográficos AES, se integró los algoritmos esteganográficos con el algoritmo criptográfico, se compararon los pixeles y el código hexadecimal de las imágenes esteganografiadas, se realizó estegoanálisis a las imágenes esteganografiadas y se calculó el PSNR - MSE para determinar la calidad de las imágenes esteganografiadas para validar el método propuesto en comparación con el estándar. Se utilizó *Netbeans* como ambiente de desarrollo (Netbeans, 2015) y las siguientes herramientas para realizar las pruebas en las imágenes: *IonForge ImageDiff* para comparar pixel a pixel las diferencias entre las imágenes (ionForge, 2014), *Beyond Compare* para determinar las variaciones en el código hexadecimal (Beyond Compare, 2016), *StegSecret* para realizar pruebas de estegoanálisis (Muñoz, 2007) y *Digital Invisible Ink Toolkit* para realizar pruebas de benchmark (University of Waikato Computer Science Department, 2016). En la Figura 1 se ilustra el proceso de la metodología utilizada y las herramientas usadas en cada proceso realizado.

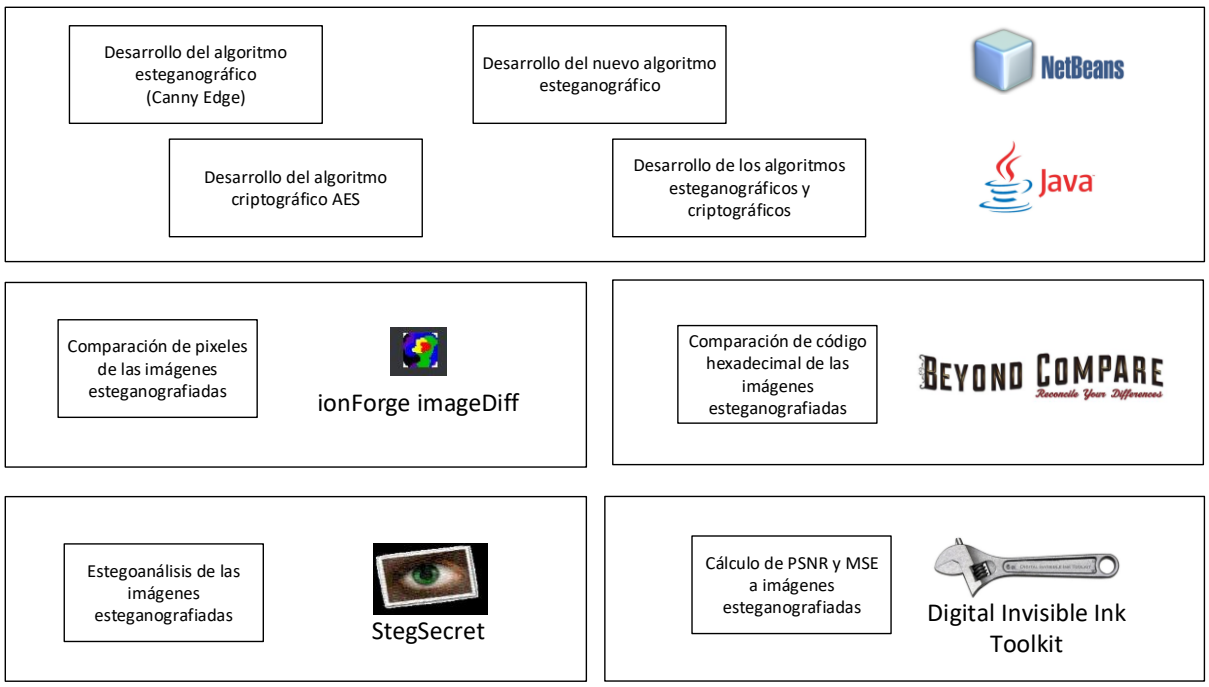


Figura 1. Metodología y herramientas utilizadas.

Desarrollo del algoritmo esteganográfico Canny Edge

Se desarrolló el algoritmo Canny Edge para detectar los bordes de la imagen en la cual se ocultará la información, comenzando desde la esquina superior izquierda y se desplaza de arriba hacia abajo y de izquierda a derecha. Se utilizó la técnica esteganográfica del Bit Menos Significativo (LSB) debido a su sencillez, rapidez, distorsión mínima de la imagen y mantiene el tamaño de la imagen sin alteración.

Desarrollo del nuevo método esteganográfico

Se propone una mejora al algoritmo de Canny Edge, realizando modificaciones en la forma para determinar los bordes de la imagen y utilizando la técnica de LSB para embeber la información cifrada de forma más dispersa en toda la imagen, pasando desapercibida en el medio digital; considerando la dimensión total para definir el salto que abarque toda su extensión. Adicionalmente se genera un archivo en formato XML en el que se almacenan las coordenadas de los pixeles de la imagen en los que se oculta la información. Como ambiente de desarrollo se utilizó Netbeans con el lenguaje de programación Java cumpliendo con las normativas de programación. Para la demostración de la hipótesis se realizaron pruebas de estegoanálisis y benchmark de las imágenes esteganografiadas con el algoritmo estándar y con el nuevo algoritmo para determinar que el nivel de seguridad es superior y la calidad de la imagen no ha sido afectada.

Desarrollo del algoritmo criptográfico simétrico AES

Se seleccionó el algoritmo criptográfico simétrico AES para incrementar la seguridad de la información considerando sus ventajas, entre las principales: resistencia a criptoanálisis, fuerza bruta, tamaño variable del bloque, utiliza claves de 128 bits, 192 bits y 256 bits; con el número de rondas para cada uno de 10, 12 y 14 rondas respectivamente. Se desarrollaron las funciones AddRoundKey, SubBytes, ShiftRows, MixColumns y para descifrado las funciones: AddRoundKey, InvSubBytes, InvShiftRows, InvMixColumns (Mikiazo, 2013).

Integración de los algoritmos esteganográficos y criptográficos

Para combinar la esteganografía con la criptografía se desarrollaron dos prototipos: el Prototipo I incluye el algoritmo Canny Edge estándar y el algoritmo AES, el Prototipo II incluye la nueva propuesta de mejora del algoritmo Canny Edge y el algoritmo AES.

Comparación de pixeles de las imágenes esteganografiadas

Para determinar la modificación visual existente en las imágenes esteganografiadas generadas con el Prototipo I y Prototipo II en comparación con la imagen original se realizó el análisis pixel a pixel.

Comparación código hexadecimal de las imágenes esteganografiadas

Para determinar la modificación existente en las imágenes esteganografiadas generadas con el Prototipo I y Prototipo II en comparación con la imagen original se realizó el análisis de su código hexadecimal.

Estegoanálisis a las imágenes esteganografiadas

Para determinar la información cifrada que esta oculta en las imágenes esteganografiadas generadas con el Prototipo I y Prototipo II en comparación con la imagen original se realizó pruebas de estegoanálisis.

Calculo de métricas para calidad de la imagen

La métrica PSNR (Peak Signal to Noise Ratio) permite medir el pico señal ruido de los datos ocultos, los valores típicos de este parámetro están entre 30 y 50 dB, siendo mayor cuanto mejor es la codificación, un alto valor de PSNR corresponde a una mejor calidad de imagen. (National Instruments, 2015)

La métrica MSE (Mean Square Error) mide el error cuadrático medio entre la imagen original y la esteganografiada. Un valor bajo de MSE significa un error menor entre las dos, cuanto menor sea el valor de MSE, menor será el error. (Kamdar, Kamdar & Khandhar, 2013)

Matemáticamente MSE se computa como se muestra en la ecuación (i) (Kaur & Verma, 2013):

$$MSE = \frac{1}{M * N} \sum_{i=1}^{M*N} (o_i - r_i)^2 \quad \text{(i)}$$

y el PSNR se calcula usando la ecuación (ii):

$$PSNR = 10 \log_{10} \frac{MAX_I^2}{MSE} \quad \text{(ii)}$$

donde:
o_i y r_i: imagen original y la imagen reconstruida respectivamente.
$M * N$: número de filas y columnas de la imagen de entrada respectivamente.
MAX: máxima cantidad de píxeles de la imagen.

Proceso de Cifrado - embebido y Extracción - descifrado

Para realizar las pruebas de criptografía se utilizan claves de 128 bits, 192 bits y 256 bits para el proceso de cifrado-embebido y extracción-descifrado con los siguientes datos:
Clave de 128 bits: @TDA&10RcrsSdBcD
Clave de192 bits: @TDA&10RcrsSdBcDcrsSdBcD
Clave de 256 bits: @TDA&10RcrsSdBcDcrsSdBcDcrsSdBcD

Mensaje (original): La esteganografía oculta información tras un medio multimedia de tal forma que pase inadvertido por terceras personas y al combinarlo con la criptografía simétrica, la información es cifrada con una clave, lo que fortalece el nivel de seguridad de la información y únicamente el destino que conozca los procesos inversos podrá extraer la información del medio multimedia y descifrarla con la misma clave, con lo que podrá obtener el mensaje original.

3. RESULTADOS Y DISCUSIONES

Prototipos desarrollados

Se diseñaron los diagramas de flujo respectivos para cada uno de los prototipos planteados, el algoritmo Canny Edge estándar combinado con el algoritmo criptográfico simétrico AES para el proceso de cifrado-embebido del Prototipo I se muestra en la Figura 2, y el diagrama propuesto para la mejora del algoritmo Canny Edge combinado con el algoritmo criptográfico simétrico AES para el proceso de cifrado-embebido del Prototipo II se muestra en la Figura 3.

Resultados de mensajes cifrados

Se realizaron las pruebas cifrando los mensajes originales con claves de 128 bits, 192 bits y 256 bits, cuyo resultado obtenido son caracteres imprimibles y no imprimibles, los cuales se muestran en la Tabla 1.

Validación de pruebas de imágenes esteganografiadas

Para la validación de la propuesta realizada de la mejora del algoritmo Canny Edge en las imágenes esteganografiadas generadas con el Prototipo I y Prototipo II en comparación con la imagen original, se realizan pruebas visuales pixel a pixel, la modificación realizada en el código hexadecimal y pruebas de estegoanálisis y pruebas de benchmark.

Figura 2. Diagrama del proceso de cifrado y embebido - extracción y descifrado con el Prototipo I.

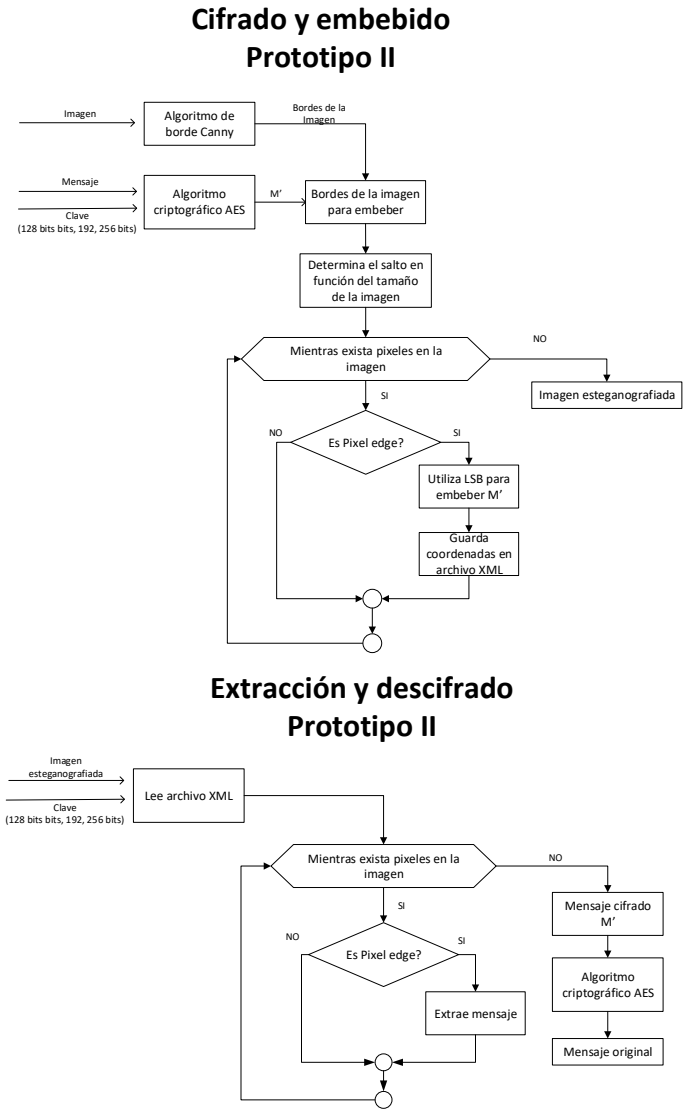


Figura 3. Diagrama del proceso de cifrado y embebido - extracción y descifrado con el Prototipo II.

Tabla 1. Mensajes cifrados con claves de 128, 192 y 256 bits con el algoritmo AES.

Clave	Mensajes cifrados
128 bits	ánU□ÃÂÂ¬÷□") 7üÐüõPçkÑ□e□□æic´õXí=□\B□Ý©MûJ□ÑB□Ã□Tÿ/□6¨3fÎ□ÎL÷«f□□TK° D ¹"Ø[MOS□□R□Â(»□×□□?-0□v□ó□/Î□ªgaõ□´ÒíE□áõ□Î□ø-¹ã|3Ðõ¿ÀDg□Ú□s]üJq!RªT êyÝ□ôÂâÊ5□p£vú□QQõ ß5©BÙE·SªñH~□ÌØö¼□¨□□Ç>qqtüí·;ifû"n-(ô□N©□^□Þ□ù□÷□ ³□v□hK □MàyIÜ□Ä□µ4GÒ`rMy)òAViæW¾□Yw□ □l"-¬G□1ÊìQl®þ±)ë%Ä□'þÓ¼³áx¬Ê{Ó5 ¹H$ÝÇ*□¶□¦¹²|þýYVæÜþ□ÐA□□ôõH½Ü%Ð□□□□□□□□Ùsc6□UãQ1[èLD□□è¨ ^ò]Õ\z·½O□¸S½f{ô]þ□ä1Bù¶ºÄÊâõÜä»ü¨ÓcmO(□P2^6½£8u□´*□G□ØØ□ZüvN½Ú¿Òp□? □?□úy+?□§ê1□9Úïø Øµ V□,*ÝHdxìÊ]Ê□#□È□6ºU¿k□xàÑ´Y?Ý|\□□«!y¼×¼Ñ¯
192 bits	ÂY□ô□¢ ¶ÓØH□□¯Ë°□Mn½K*gqº].Ú□°Õi□g□X9□NTùÑqQ=º□□□-Ö»&□Ì□B±q.%¤ïÔÈí□ó°□□K □[bî4]ë□»É+[r)□I□<□V□#Ì□3wü□ºÒ□h"Ø´Hßû[ï*ß□p 9i□□□□ïfF¾¿□\7□□D!Ñ□bóä□!ÜzMã,w□Ù□!□4□□Qif□ð/ùõ□□>ø G±¬H□×¾fØ=G□?□ @J□¦£R{s□W□uº □C2□fCAuy¼□y¯¶ÓWéZ8ÍAb□□Â+ÑÝ§ KÒmôW□;7µº3□ÖÐ¨ø;ä□¼¥q□PfÄIKp gß□□.d¿□ú#\F7¶Û□ÿL□â□0Ú1¥;n□7□□Ld ÉL□fT□ßTô8Æmø7Àr,□□ñ□□MC{*jeiLÀ O□ÐÎ□ÿ1□Ì□Äúùûë□±ïé□^□ùô£³cf□û>WÜ□xÞ¯¿u□□]Ö¯s©iKQ£m[□egYG□r&ó µ□Ú°9□ª□"□ r=Fé_1□®□□ÜKÔzY□¹~□êz□æîø9□NÂû□Lqr□ðî□□1□ù□É)ÙË¶ÊWÉ·$ô¥Ðâ □□rSO¸R(Í?Ï
256 bits	^□)□ÂDÐPjj!883D@□□pVm]□mR9±í Ò□£□Ö·.Á□□□[hWü□□□D}4¶q ÷Ô□□□n¦A□¶-ë· □âÆºæ□i□□§¤X´°0i□È□£ìæðúÈC?□□A4□m¿D□CcÚ4□□I□M%EEói□ò\Â=áü.K`õ¨ßÃÿ□□ b{¶□ñ□·ÿ□□U±½□□pZ½Æ¸R^-újÃ□IñhÍ+ò□ãSâÝºº□¶B□B □¿z□«¼ËɛI@□Càw□□□ü qÍÊ□□xça□Ø □ãJ"?i©□ Ùuæ)r□U÷®¶ò°&by N8□□□E□HV¬¸'ùÙò□T«»&□□W□:uï&¨´5□à&\±ÿh ºxµusJíu/□zp□¨□k□óP□eÝÀ□□_O□G!&k¶MºMy3□□□¤0k.¿Ù;Z±□ÉløÂU:98□□p□|Qø}«³ □y□wi □ÏÚBfêvº u&Në÷÷□T□X□bç0b!□\□¹iAÙ□Nz2□w+7a□ìÆ□.□À□¯□i□Kuö□□ □Ýªg¹□C□□ê-¯.□t□'TÂ,v□:□ýc□ □□□□□ý□Ä□□S¶□□T,□Âª È□:À ¨''²Ýu

Comparación de resultados esteganográficos

Se comparan de forma visual la imagen original “Paisaje.bmp” con las imágenes esteganografiadas “Paisaje_embebido.bmp” que fueron generadas con el Prototipo I y Prototipo II, con el mensaje que fue cifrado con claves de 128 bits, 192 bits y 254 bits, el resultado obtenido no puede ser notorio a simple vista, como se muestra en la Figura 4.

Figura 4. Comparación visual de imágenes.

Se determinan los bordes de la imagen “Paisaje.bmp” como se muestra en la Figura 5, para ocultar la información cifrada con el algoritmo criptográfico simétrico AES. Además, se analiza el tamaño de las imágenes “Paisaje.bmp” y “Paisaje_embebido.bmp” que fueron generadas con el Prototipo I y Prototipo II, con el mensaje cifrado con claves de 128 bits, 192 bits y 256 bits, con lo que se determina que los tamaños son exactamente iguales, lo que cumple con el principio de la esteganografía de imperceptibilidad.

Para determinar las diferencias en el código hexadecimal se utiliza el programa Beyond Compare, con lo que se compara la imagen original Paisaje.bmp” y las imágenes “Paisaje_embebido.bmp” que

fueron generadas con el Prototipo I y Prototipo II, mostrando con color rojo las diferencias encontradas entre ellas, como ejemplo se muestran las primeras diferencias en la Figura 6.

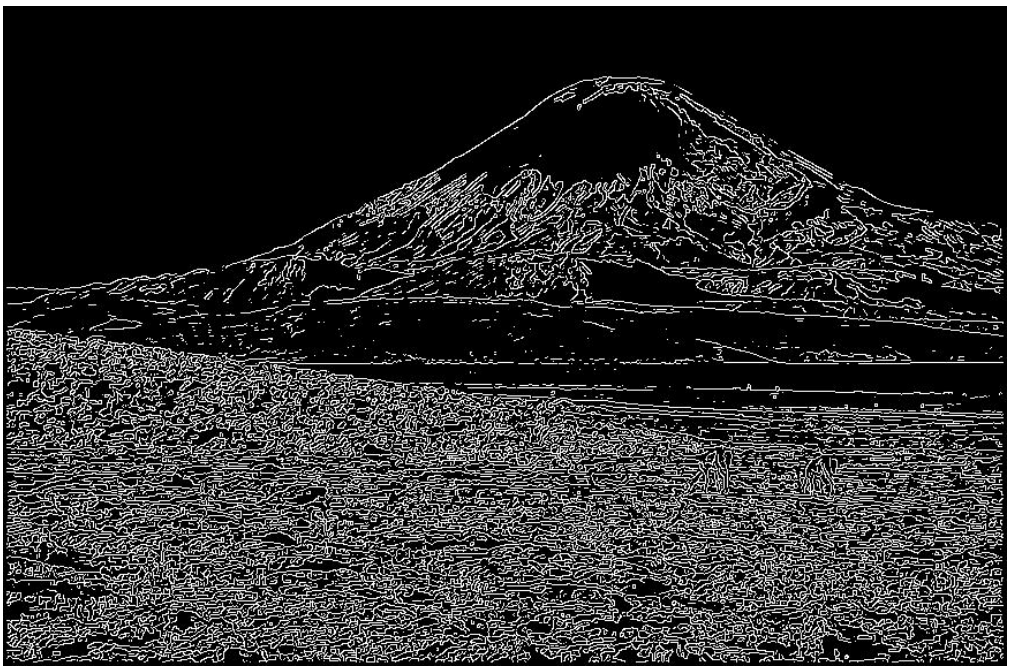
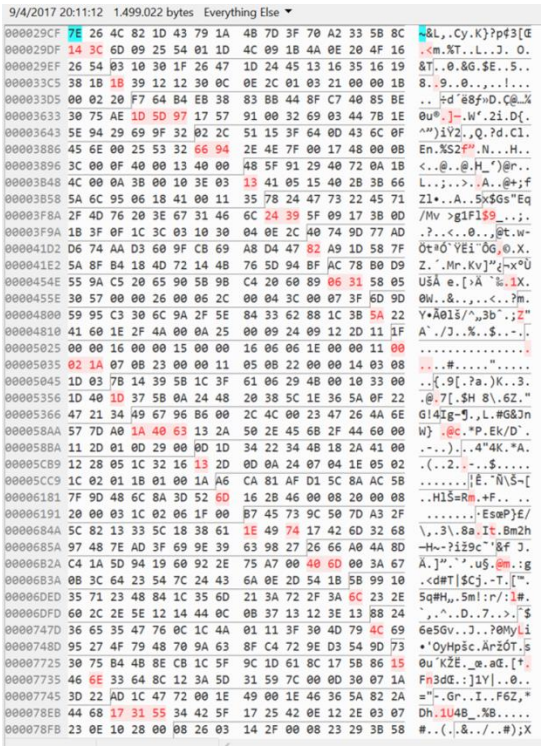


Figura 5. Bordes de la imagen determinados con algoritmo Canny Edge.

Paisaje_embebido PI.bmp



Paisaje_embebido PII.bmp

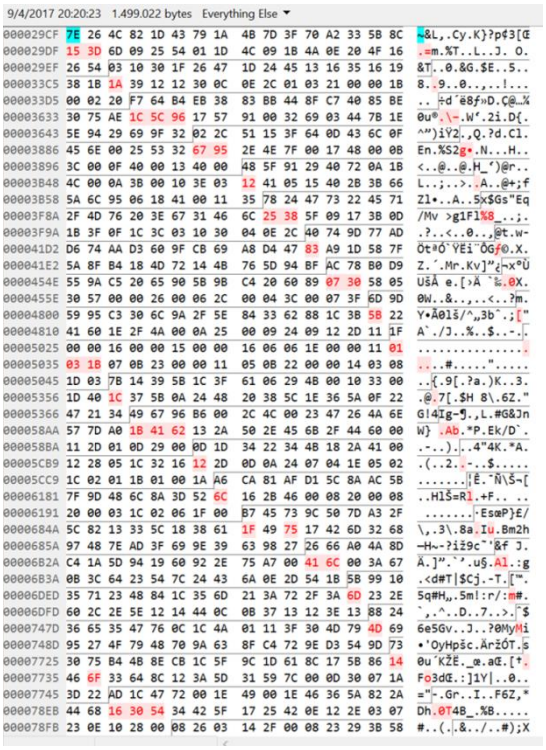
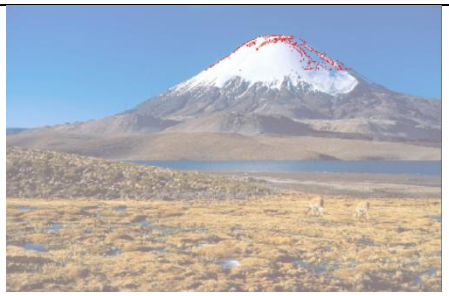
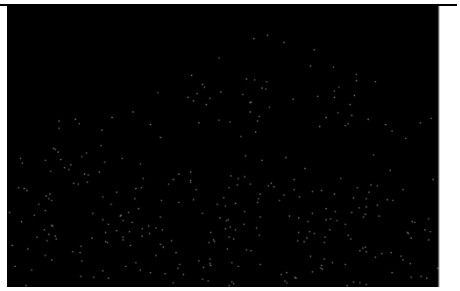


Figura 6. Comparación de código hexadecimal de las imágenes esteganografiadas.

Para determinar las diferencias visuales se utiliza el programa IonForge ImageDiff, con lo que se compara pixel a pixel la imagen original Paisaje.bmp" y las imágenes "Paisaje_embebido.bmp" que fueron generadas con el Prototipo I y Prototipo II, mostrando los pixeles que fueron modificados con la

información cifrada oculta dentro de la imagen, como se muestra en la Tabla 2. Se puede visualizar que la información oculta en la imagen generada con el Prototipo II se encuentra dispersa en toda la imagen del borde lo que incrementa el nivel de seguridad e imperceptibilidad, en comparación con la información oculta en la imagen generada con el Prototipo I que se encuentra ubicada en la parte superior siguiendo las primeras posiciones del borde, lo cual concentra la información en esta sección de la imagen.

Tabla 2. Análisis de pixeles modificados con el Prototipo I y Prototipo II.

	Prototipo I	Prototipo II
Con fondo		
Sin fondo		

Para almacenar las posiciones en las que se encuentran los bordes de la imagen en los que se oculta la información cifrada con la nueva propuesta de Canny Edge se generaron los archivos key.xml, con el salto establecido determinando en la dimensión de la imagen las posiciones varían por lo que son más dispersas en comparación con el algoritmo estándar de Canny Edge, ejemplos de esta comparación se muestran en la Figura 7.

Para realizar las pruebas de estegoanálisis de las imágenes se utilizó el software StegSecret para ejecutar un ataque RS que es un algoritmo para la detección de LSB-pseudoaleatorio y poder estimar el tamaño de la información oculta (Fridrich, Goljan & Du, 2001). Para realizar pruebas de benckmark de las imágenes esteganografiadas se utilizó el software Digital Invisible Ink Toolkit con el cual se calculó las métricas PSNR y MSE, las pruebas de estegoanálisis y benchmark se muestran en la Figura 8. Los resultados obtenidos fueron que la imagen generada con el Prototipo II no fue distorsionada visualmente debido a que el valor de PSNR es alto y el valor de MSE es bajo en comparación con la imagen generada con el Prototipo I.

4. DISCUSIÓN

Comparando los resultados obtenidos en la presente investigación, con la de otros autores mencionados en los trabajos relacionados, el principal beneficio de la propuesta de mejora del algoritmo Canny Edge es que difumina el mensaje cifrado en los bordes determinados de toda la imagen y no únicamente en los primeros pixeles del borde como en el algoritmo estándar, por lo que es menos detectable ante estegoanálisis sin afectar su calidad. Al incluir la criptografía simétrica con claves de 128 bits, 192 bits o 256 bits, los mensajes son cifrados incrementando la seguridad de la información. Se utilizaron herramientas de estegoanálisis y benchmark para determinar variaciones en las imágenes esteganografiadas y compararlas con la imagen original con el objetivo de validar el nuevo algoritmo

esteganográfico planteado. El nuevo método puede ser utilizado para acreditar la validez de un documento publicado electrónicamente debido a que oculta información en el medio multimedia pasando desapercibido por terceras personas y únicamente el destino que conozca su proceso inverso podrá verificar la autenticidad del documento.

Prototipo I

key1.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<coordenadas>
  <pos col="520" row="61" />
  <pos col="521" row="61" />
  <pos col="522" row="61" />
  <pos col="523" row="61" />
  <pos col="525" row="61" />
  <pos col="526" row="61" />
  <pos col="527" row="61" />
  <pos col="550" row="61" />
  <pos col="551" row="61" />
  <pos col="552" row="61" />
  <pos col="517" row="62" />
  <pos col="518" row="62" />
  <pos col="519" row="62" />
  <pos col="524" row="62" />
  <pos col="525" row="62" />
  <pos col="527" row="62" />
  <pos col="528" row="62" />
  <pos col="529" row="62" />
  <pos col="530" row="62" />
  <pos col="531" row="62" />
  <pos col="532" row="62" />
  <pos col="533" row="62" />
  <pos col="534" row="62" />
  <pos col="535" row="62" />
  <pos col="536" row="62" />
  <pos col="537" row="62" />
  <pos col="538" row="62" />
  <pos col="539" row="62" />
  <pos col="540" row="62" />
  <pos col="541" row="62" />
  <pos col="542" row="62" />
  <pos col="543" row="62" />
  <pos col="544" row="62" />
  <pos col="545" row="62" />
  <pos col="546" row="62" />
  <pos col="547" row="62" />
  <pos col="548" row="62" />
  <pos col="549" row="62" />
  <pos col="553" row="62" />
  <pos col="554" row="62" />
  <pos col="555" row="62" />
  <pos col="556" row="62" />
```

Prototipo II

key2.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<coordenadas>
  <pos col="520" row="61" />
  <pos col="524" row="64" />
  <pos col="565" row="67" />
  <pos col="496" row="71" />
  <pos col="558" row="73" />
  <pos col="477" row="76" />
  <pos col="557" row="78" />
  <pos col="484" row="81" />
  <pos col="581" row="84" />
  <pos col="578" row="89" />
  <pos col="617" row="93" />
  <pos col="591" row="99" />
  <pos col="438" row="104" />
  <pos col="427" row="110" />
  <pos col="652" row="114" />
  <pos col="416" row="118" />
  <pos col="615" row="122" />
  <pos col="610" row="126" />
  <pos col="656" row="128" />
  <pos col="567" row="131" />
  <pos col="575" row="133" />
  <pos col="697" row="134" />
  <pos col="383" row="137" />
  <pos col="620" row="138" />
  <pos col="706" row="139" />
  <pos col="609" row="141" />
  <pos col="643" row="142" />
  <pos col="702" row="143" />
  <pos col="372" row="145" />
  <pos col="415" row="146" />
  <pos col="415" row="147" />
  <pos col="378" row="148" />
  <pos col="717" row="148" />
  <pos col="687" row="149" />
  <pos col="646" row="150" />
  <pos col="554" row="151" />
```

Figura 7. Comparación de archivos XML de coordenadas del Prototipo I y Prototipo II.

5. CONCLUSIONES

- La imagen esteganografiada que fue generada con el Prototipo II (aplicando nuevo método esteganográfico) distribuye la información cifrada en toda la imagen, proporcionando mayor difusión sin modificar el tamaño del archivo, sin distorsionar la imagen; lo que garantiza la imperceptibilidad de la información, en comparación con la imagen generada con el Prototipo I (método esteganográfico original).

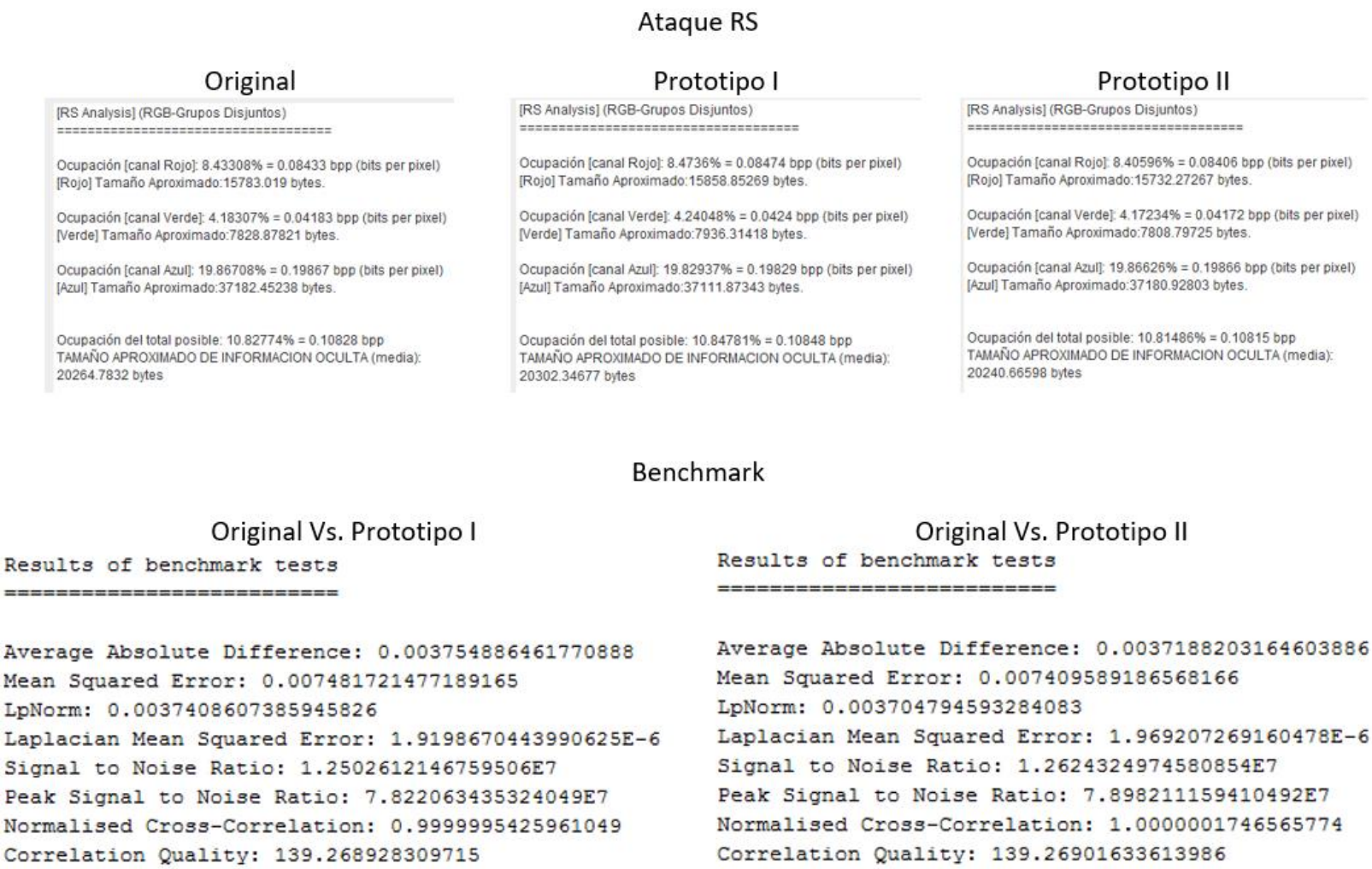


Figura 8. Resultados de estegoanálisis y benchmark de las imágenes esteganografiadas.

- Se incrementó la seguridad con la generación del archivo XML que almacena las posiciones en las que se embebe el mensaje cifrado debido a que sin él no se puede obtener el mensaje original.
- Al cifrar la información con el algoritmo esteganográfico AES utilizando claves de 128 bits, 192 bits y 256 bits, se incrementa la seguridad en el mensaje contra posibles ataques de fuerza bruta.
- Se recomienda utilizar imágenes con tamaños mayores a la información que va a ser embebida para que pueda ser embebida sin dificultad.
- Como trabajos futuros se puede considerar:
  - Utilizar o modificar diferentes algoritmos criptográficos simétricos o asimétricos para incrementar la seguridad de la información.
  - Cifrar el archivo XML para que no pueda ser accedida por terceras personas no autorizadas.

BIBLIOGRAFÍA

Beyond Compare (2016). *What's New in Beyond Compare 4*. Obtenido de https://www.scootersoftware.com/download.php

Comunidad OWASP (2009). *Owasp*. Obtenido de Cryptanalysis https://www.owasp.org/index.php/Cryptanalysis

Fridrich, J., Goljan, M., & Du, R. (2001). *Reliable detection of LSB steganography in color and grayscale images*. New York. Obtenido de https://www.ws.binghamton.edu/fridrich/Research/acm_2001_03.pdf

Gaba, J., & Kumar, M. (2013). *Implementation of steganography using CES technique*. IEEE Second International Conference on Image Information Processing (ICIIP) (págs. 395-399). Shimla: IEEE.

ionForge (2014). *ionForge ImageDiff*. Obtenido de http://www.ionforge.com

Jabbar, A., Alaa, A., Sahib, S., & Zamani, M. (2013). *An introduction to image steganography techniques*. Advanced Computer Science Applications and Technologies (ACSAT), 2012 International Conference on (pág. 5). IEEE.

Kamdar, N., Kamdar, D., & Khandhar, D. (2013). Performance Evaluation of LSB bases Steganography for Optimization of PSNR and MSE. *Journal of Information, Knowledge and Research in Electronics and Communication Engineering*, *2*(2), 505-509.

Kaur, J., & Verma, H. (2013). *A hybrid approach for image security by combining encryption and steganography*. Image Information Processing (ICIIP) (págs. 607-611). IEEE.

Lerch-Hostalot, D., & Megías, D. (2014). *Esteganografía en zonas ruidosas de la imagen*. Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información (págs. 173-178). Alicante: Universidad de Alicante.

Mikiazo (2013). *Cryptography*. Obtenido de http://crypto.stackexchange.com/questions/8043/aes-addroundkey

Mishra, R., Mishra, A., & Bhanodiya, P. (2015). *An edge based image steganography with compression and encryption*. Computer, Communication and Control (IC4). IEEE.

Muñoz, A. (2007). *Stegsecret.* Obtenido de http://stegsecret.sourceforge.net/

National Instruments (2015). *What is the Peak Signal to Noise Ratio (PSNR) measurement in NI Picture Quality Analysis (PQA)?* Obtenido de http://digital.ni.com/public.nsf/allkb/CA0C16A29F6C089586257E2000773991

Netbeans (2015). *Netbeans*. Obtenido de https://www.netbeans.org

Nurhayati & Ahmad, S. S. (2015). *Steganography for inserting message on digital image using least significant bit and AES cryptography algorithm*. Cyber and IT Service Management, International. IEEE.

Qiang, S., Guoying, M., & Hongmei, Z. (2016). *An edge-detection method based on adaptive canny algorithm and iterative segmentation threshold*. Control Science and Systems Engineering (ICCSSE) (págs. 64-67). IEEE.

Rodríguez, G., & Navas, S. (2016). *Esteganografía: Sustitución LSB 1 bit utilizando Matlab*. XVIII Workshop de Investigadores en Ciencias de la Computación (WICC 2016, Entre Ríos, Argentina), (págs. 859-864).

Rodríguez, M., Navas, S., & Eterovic, J. (2014). *Aplicación del filtro de Canny en la esteganografía digital*. WICC 2014 XVI Workshop de Investigadores en Ciencias de la Computación, (págs. 806-811).

Saini, J., Verma, H. (2013). *A hybrid approach for image security by combining encryption and steganography*. IEEE Second International Conference on Image Information Processing (ICIIP) (págs. 607-611). Shimla: IEEE.

Singla, D., & Juneja, M. (2014). New information hiding technique using features of image. *Journal of Emerging Technologies in Web Intelligence, 6*(2), 237-242.

University of Waikato Computer Science Department (2016). *Digital Invisible Ink Toolkit*. Obtenido de http://diit.sourceforge.net/