

Proyecto Final  
Clase: Programación Multinúcleo  
Profesor: Octavio Navarro  
César Armando Valladares Martínez  
A01023506  
Juan Pablo Güitrón Leal  
A01019936

## Resumen

El siguiente documento es un reporte sobre el proyecto final de programación multinúcleo. Este proyecto consiste en editar los píxeles de una imagen para poder esconder un mensaje (Esteganografía) dentro de esta, utilizando CPU, CPU con hilos y GPU. En este documento se muestran análisis y comparaciones entre los tiempos de ejecución de cada uno de los algoritmos utilizados así como la explicación de la resolución del problema.

## Introducción

La esteganografía es la aplicación de técnicas para ocultar mensajes u objetos dentro de otros objetos, como imágenes, videos o audios. La esteganografía, a diferencia de la criptografía, esconde la mera existencia de la información y es indetectable en un análisis tradicional de información.

Actualmente existen varios métodos para la esteganografía, pero en este proyecto queremos enfocarnos en el método del bit menos significativo (LSB). Este método consiste en modificar el último bit dentro de los bytes de un objeto o imagen con el fin de alterarlo lo menos posible al momento de agregar el mensaje.

Ejemplo:

- En el caso de una imagen RGB se tienen 3 bytes por cada pixel, suponiendo que se quisiera esconder la letra 'A' cuya representación en binario es 1 0 0 1 0 1 1 dentro de la imagen se deben utilizar 8 bytes los cuales serian 3 pixeles de la imagen, esto se veria asi:

(1 1 0 1 1 0 1 1) (0 1 0 0 1 0 0 0) (0 1 0 0 0 0 1 0)

(0 0 0 1 1 1 1 1) (0 1 0 1 1 0 1 0) (1 1 0 1 1 1 1 1)

(0 0 0 0 1 1 1 1) (0 1 0 0 0 1 1 1) (0 0 0 0 0 1 1 1)

## ¿Por qué paralelismo?

El bit es la unidad más pequeña utilizada en la computación, cuya representación únicamente es 1 y 0, por lo que para poder representar mucha información se deben utilizar varios bits. Tener una cadena de caracteres o archivos escondidos en una imagen con gran información sería algo que tomaría mucho tiempo si se hace de forma secuencial, ya que se accedería individualmente a cada pixel para editar los bits, mientras que utilizando

paralelismo este tiempo se podría reducir, pues la edición de los píxeles no depende uno de otro.

### **Aplicaciones:**

- Pasos información: Se pueden mandar mensajes ocultos dentro de las imágenes, vídeos, etc. Información importante la cual es bastante peligroso que se transporte de manera que cualquier persona la pueda ver.
- Ciberseguridad: Además de ocultar la imagen muchas veces, antes de esconder el mensaje, se encripta, esto le da una capa más de seguridad al paso de información importante. Desde el punto de vista empresarial las aplicaciones principales son la detección de malware que usa esteganografía para ocultarse y la detección de usuarios maliciosos intentando extraer secretos. Desde el punto de vista de las agencias de seguridad nacional, las aplicaciones principales del estegoanálisis son la detección de comunicaciones terroristas o de espionaje.
- Derechos de autor en el contenido: Una marca de agua digital es un código de identificación que se introduce directamente en el contenido de un archivo multimedia, normalmente para incluir información relativa a los derechos de autor o de propiedad del contenido digital en cuestión. La presencia de esta marca de agua debe ser inapreciable para el sistema de percepción humano a la vez que fácilmente extraíble por una aplicación telemática que conozca el algoritmo para recuperarla. Como consecuencia, en la actualidad se están empleando técnicas que en algunos casos pueden ser consideradas esteganografía para conseguir dicho propósito.
- Reconocimiento de imágenes por buscadores: Como en la aplicación de derechos de autor, aquí la imagen guarda un tag o etiqueta que clasifica a la imagen.

### **Proyectos similares:**

- Fileinjector - pc
- Our secret - pc
- Steganography master - móviles
- Steganographia - móviles
- Hakros concealer - pc

### **Librerías:**

- OpenCV: Está es una librería que existe para C++, Python, y Java. En nuestro caso será utilizado en C++, está optimizada en C/C++ y puede tomar ventaja del procesamiento multinúcleo. OpenCV nos permite trabajar con imágenes RGB y obtener los valores RGB de cada píxel.

### **Desarrollo:**

- Para poder codificar y decodificar un mensaje dentro de una imagen utilizando paralelismo cada hilo se encarga de:
  - Al codificar el mensaje:
    - Cada hilo escribe un carácter dentro de la imagen, para hacer esto cada hilo edita 8 colores cambiando el bit menos significativo del color para afectar lo menos posible la imagen.

Decodificar el mensaje:

- Cada hilo obtiene un carácter de la imagen, para hacer esto cada hilo lee el último bit de 8 colores consecutivos los cuales se van agregando a un carácter empezando por el bit menos significativo hasta el más significativo.

Nota: Después de realizar pruebas con diferentes configuraciones de bloque decidimos utilizar bloques de 32 \* 32 ya que no había una diferencia significativa entre los tiempos de ejecución.

## Resultados:

Se ejecutaron un total de 60 pruebas en 23.82 segundos.

### - Características del sistema utilizado

- Device: Laptop
- Processor (CPU): Intel Core i7-8750H
  - Frequency: 2.2 GHz
  - Cores: 6
- Video card (GPU): NVIDIA GeForce GTX 1050 ti
  - Memory: 4 GB GDDR5
  - Cores: 768 cuda cores
  - Frequency: 1290 MHz

### Comparación de tiempos (Tiempos en milisegundos):

Las siguiente tabla muestra una comparación de tiempos de ejecución entre el CPU, CPU con hilos y GPU utilizando una imagen con resolución 8192\*5461 codificando un mensaje de 16Mb.

	CPU cipher	CPU decipher	OMP cipher	OMP decipher	GPU cipher	GPU decipher
1	626.996582	576.602478	576.250732	613.618469	0.012879	0.017623
2	615.271606	582.297791	576.981262	614.635559	0.013339	0.035305
3	612.596436	577.236633	575.41571	620.694153	0.017088	0.01701
4	609.701416	575.124329	570.856323	612.069702	0.01418	0.012245
5	617.102722	580.171265	574.444397	616.65802	0.014144	0.01505
6	617.787415	581.327515	575.981323	612.77655	0.012785	0.013966
7	611.763977	578.442322	575.237915	614.312439	0.014319	0.019709
8	609.847961	570.68103	576.50592	611.834839	0.012241	0.013928
9	609.44397	574.186218	575.03241	623.065002	0.016021	0.012908

10	606.436157	572.360962	575.824707	622.784302	0.013715	0.017391
	613.694824	576.8430543	575.2530699	616.2449035	0.014071	0.0175135
					1	

SpeedUP	cipher	decipher
CPU / OMP	1.06682581338798	0.93606137920782
CPU / GPU	43613.8485406258	32937.0516630028
OMP / GPU	40881.8834277349	35186.8503440203

### Análisis:

Como se puede observar en las tablas anteriores, los resultados de los speedup's son los esperados, también se puede ver que la mejora entre CPU y CPU con hilos no es tan significativa, esto puede ser debido al tamaño de la imagen y el mensaje. El tiempo de ejecución en GPU es considerablemente mucho menor a los dos casos en CPU.

### Referencias:

Computerworld. (2002). QuickStudy: Steganography: Hidden Data. 26/09/2018, de Computerworld Sitio web: <https://web.archive.org/web/20080226100159/http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=71726>

Manuel López Michelone. (2012). Esteganografía: para cifrar mensajes en imágenes. 26/09/2018, de UNOCERO Sitio web: <https://www.unocero.com/noticias/esteganografia-para-cifrar-mensajes-en-imagenes/>

Enciclopedia Universal. (2012). Bit menos significativo. 26/09/2018, de ACADEMIC Sitio web: [http://enciclopedia\\_universal.esacademic.com/65202/Bit\\_menos\\_significativo](http://enciclopedia_universal.esacademic.com/65202/Bit_menos_significativo)

Edgardo Fernandez. (2014). Cómo ocultar información en imágenes: Herramientas para esteganografía. 26/09/2018, de Neoteo Sitio web: <https://www.neoteo.com/herramientas-para-esteganografia/>

NVIDIA. (2018). OpenCV. 29/09/2018, de NVIDIA Sitio web: <https://developer.nvidia.com/opencv>

Daniel Lerch. (2017). La esteganografía es una herramienta de gran interés para los criminales. 30/29/2018, de PANDA SECURITY Sitio web:

<https://www.pandasecurity.com/spain/mediacenter/panda-security/esteganografia-daniel-lerch/>

Observatorio de la Seguridad de la Información . (2001). ESTEGANOGRAFÍA, EL ARTE DE OCULTAR INFORMACIÓN . 30/09/2018, de INTECO Sitio web:

<http://www.egov.ufsc.br/portal/sites/default/files/esteganografia1.pdf>