**UNIVERSIDAD NACIONAL DEL CENTRO DE LA PROVINCIA DE BUENOS AIRES  
FACULTAD DE CIENCIAS EXACTAS**

**INGENIERÍA DE SISTEMAS**



**Trabajo Práctico Especial   
3ra Entrega**

**TEORÍA DE LA INFORMACIÓN**

**Integrantes:**

Sanchez Casco, Angelo : [angelo16s.as@gmail.com](mailto:angelo.16s@gmail.com), LU: 249116

Bertino, Ariel: [eugenioingenio10@gmail.com](mailto:eugenioingenio10@gmail.com), LU: 247865

### Resumen

En el presente trabajo se pretende analizar diversas características de nuevas imágenes provistas por la cátedra mediante las aplicaciones desarrolladas en las entregas anteriores.

### Introducción

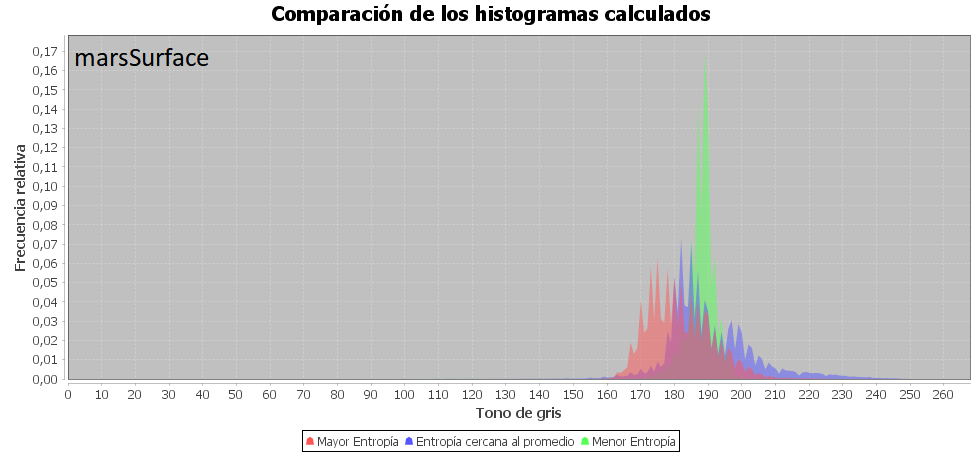
Sobre las nuevas imágenes se pretende analizar la proporción de cada tono en las imágenes (histograma), la media y desvío estándar del los tonos de gris, luegos se precisa comprimir las imágenes para estudiar el rendimiento del compresor en cada caso, estos estudios se realizan sobre las imágenes *citySurface*.bmp y *linesSurface*.bmp. Por otra parte a partir de las imágenes *cS1*.bmp y *cS2*.bmp que funcionarán como resultado de transmitir la imagen *marsSurfaceTp2*.bmp se calculará el ruido y la pérdida de los respectivos canales asociados.

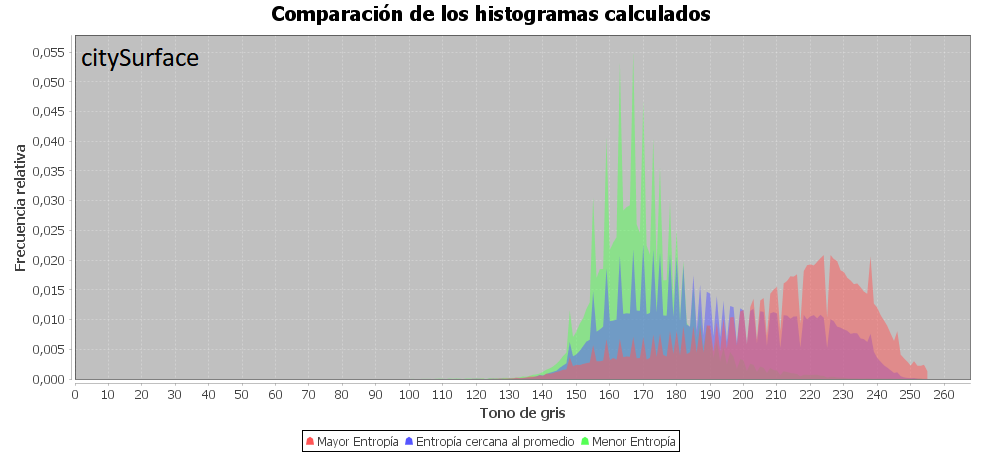
### Desarrollo

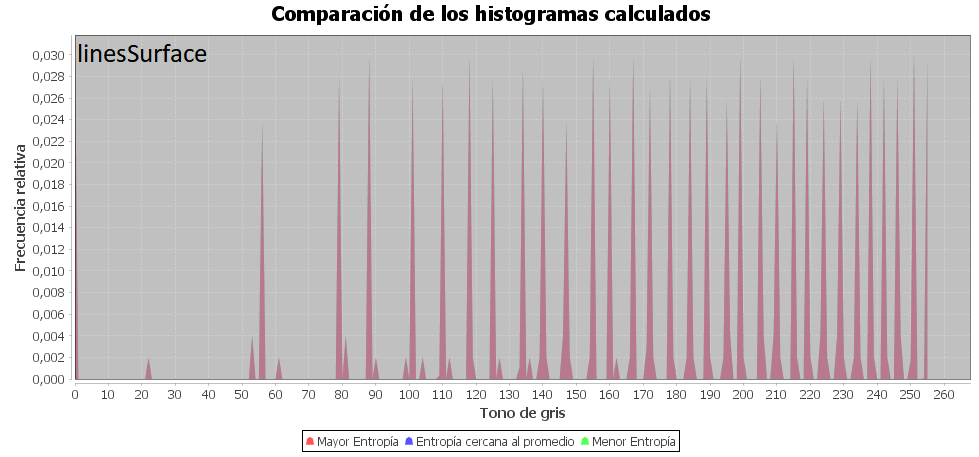
El desarrollo de este este trabajo consiste en analizar el comportamiento de los algoritmos de compresión baja diferentes imágenes de entrada. Luego se necesita evaluar dos canales.

### Resultados

En primer lugar se muestran los histogramas de *marsSurface*.bmp, *citySurface*.bmp y *linesSurface*.bmp seguido de un breve análisis de las mismas.







### Las medias y desvío estándar para las imágenes son:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Entropía menor | | Entropía mayor | | Valor Medio | | Desvío Estándar | |
| Bloque | Valor | Bloque | Valor | Bloque de menor entropía | Bloque de mayor entropía | Bloque de menor entropía | Bloque de mayor entropía |
| *marsSurface* | 5 | 3.61 | 4 | 4.88 | 182.77 | 183.47 | 6.54 | 5.75 |
| *citySurface* | 17 | 4.26 | 14 | 5.50 | 182.21 | 205.25 | 22.17 | 27.67 |
| *linesSurface* | 4 | 0 | 3 | 0 | x | x | x | x |

Se observa a partir del histograma de *marsSurface* y de los valores medios obtenidos los tonos presentes en la imagen tienden a ser los que se encuentran en el medio del espectro de grises con tendencia a grises un poco más oscuros y además posee un desvío de aproximadamente de 6.5 y 5,75 tonos con lo cual hay poca variación de tonos(de entre 11 a 14 tonos aproximadamente) en la imagen, lo cual concuerda a simple vista donde no se ven cambios bruscos en las tonalidades de grises. los valores de entropía (3,61 y 4,88) nos indican que como mucho se deberá hacer de 4 a 5 preguntas en promedio para poder determinar el tono de cualquier píxel del bloque. Se podría establecer una relación entre la entropía y el desvío en base a la imagen de entrada, que en este caso es una relación directa, a mayor entropía, mayor el desvío.

En lo que respecta a la imagen *citySurface* los tonos medios para los bloques de menor entropía(4.26) y mayor entropía(5.50) son 182.21 y 205.25 respectivamente. Mayormente se observan tonos claros de grises incluso pudiendo identificar blanco y en menor medida tonos oscuros de grises. Se puede resaltar el hecho de que hay una variación muy grande de tonos no solo de bloque a bloque sino incluso de pixel a pixel esto nos lleva a valores muy altos de entropía ya que se requieren muchas evaluaciones para poder identificar el tono de un píxel cualquiera de la imagen. Esto se verifica no solo observando la entropía sino también prestando especial detalle al desvío estándar de tonos que son bastante altos.

Por último, analizando la imagen *linesSurface* que presenta una repetición de un patrón de degradado horizontal de tonos de gris desde el blanco hasta el negro. Por las propiedades de la imagen se observa que todos los bloque poseen la misma entropía(0), pero en todas las tiradas se obtiene una media y un desvío diferente. Esto se debe a que el grafo de transición tiene ciclos de alta probabilidad en cada nodo, pudiendo saltar a otro nodo muy rara vez, ajustar el error de convergencia en este caso sería en vano, ya que en realidad estos valores van a tender al estado absorbente (tono 0) sólo cuando la cantidad de tiradas tienda a infinito, aumentando a su vez el tiempo de ejecución considerablemente.

Ahora se presentan los valores obtenidos por el algoritmo híbrido de la compresión: para estos cálculos los valores de umbral(Ht) que utilizamos son 3,2, 3,8 y 4,2. El tamaño de las imágenes sin comprimir es de 4,76 MB

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Tamaño de las compresiones | | | Cantidad de bloques | | | | | |
|  | Huffman | | | RLC | | |
| Ht | 3.2 | 3.8 | 4.2 | 3.2 | 3.8 | 4.2 | 3.2 | 3.8 | 4.2 |
| *marsSurface* | 3.1MB | 3.89MB | 6.53MB | 20 | 17 | 7 | 0 | 3 | 13 |
| *citySurface* | 3.95MB | 3.95MB | 3.95MB | 20 | | | 0 | | |
| *linesSurface* | 80KB | 80KB | 80KB | 0 | | | 20 | | |

Aquí se muestran distintos tamaños de compresiones realizadas con los distintos valores de Ht para comprimir los bloques de cada imagen con dos métodos distintos de compresión (método de Huffman semi-estático y Run Length Code) y cuantos bloques se comprimieron con un dado método.

Al comprimir la imagen *marsSurface* con un umbral de 3.2 se comprimieron todos los bloques con huffman, obteniendo un tamaño en la compresión de 3.1MB, subiendo el umbral a 3.8 ya aparecen bloques con RLC, teniendo 17 bloques codificados con Huffman y 3 con RLC con un peso de 3.89MB y por último con un umbral de 4.2 se obtienen 7 bloques Huffman y 13 RLC, luego la compresión pesa unos 6.53MB y podemos observar que a medida que aumenta la cantidad de bloques codificados con RLC tenemos mayor tamaño en la compresión, lo que indica que si bien los tonos presentes son similares no es conveniente codificar con RLC porque no hay tiradas largas con tonos uniformes.

La imagen *citySurface* posee mucha variación de tonos con lo cual sería poco intuitivo comprimir utilizando RLC ya que seria un desperdicio de espacio innecesario y no se estaría cumpliendo el objetivo que es reducir el tamaño mediante una compresión. Es por eso que bajo cualquier valor de umbral con los que probamos el algoritmo híbrido opta por codificar mediante Huffman la imagen completa, dando como resultado una compresión con un peso de 3.95 MB, dando un factor de compresión de 1.2 : 1.

Luego mirando *linesSurface* es inevitable pensar que debería codificarse con RLC. En efecto la codificación se realiza siempre mediante RLC. A simple vista se podrá deducir esto ya que la imagen se compone de líneas horizontales de un mismo tono con lo cual habrá en la comprensión solo 500 duplas que serán (tono-500), donde el 500 indica la longitud de la línea de ese tono. Se reduce muchísimo el tamaño de la compresión (59 : 1) que al fin y al cabo es lo que se está buscando. La imagen comprimida consta de 20 bloques codificados con RLC y con un peso total de 80 KB para los tres valores ht.

A continuación se muestra el ruido y la pérdida del canal tomado como imagen de salida *marsSurfaceTp2*.bmp y de llegada las imágenes *cS1*.bmp y *cS2*.bmp para evaluar dos posibles canales distintos. Llamaremos C1 y C2 a los canales que tienen como llegada las imágenes Cs1 y Cs2 respectivamente para poder diferenciarlos de manera clara.

|  |  |  |
| --- | --- | --- |
| Canal | Ruido | Pérdida |
| C1 | 0.4267218606189754 | 0.5469859643061109 |
| C2 | 0.0 | 3.814544564481346 |

Observando el canal C1 por cada tono que se envía de la imagen original(*marsSurface*) se introduce un ruido de 0.4267 esto indica que es sumamente fácil determinar el tono de gris que llegará a la salida conociendo el tono que se emite. Por otro lado la perdida de informacion tambien es pequeña siendo esta de 0.5469 esto indica que cuesta poco trabajo discernir mirando el tono de un pixel de la imagen de llegada de qué tono proviene desde el píxel original. Esto se debe a que la imagen de llegada es la negativa exacta de la original, con a lo sumo un poco de compresión al guardar la imagen. Ahora si bien yo se que la imagen me llega negativa no es tan grave, es un defecto del canal que puede ser salvado tratando la imagen de llegada.

Tomando como imagen de llegada *Cs2* vemos que el canal C2 no introduce ruido en la recepción de la imagen esto se traduce en que a la hora de enviar un tono se sabe que a su llegada se espera un sólo tono de gris, con lo cual no habrá información redundante al enviar cada pixel por este canal. Pero en contraposición existe un pérdida de información muy grande siendo esta de 3.8145 con esto nos damos cuenta que si yo quisiera reconstruir la imagen de partida mediante la imagen de llegada no tengo forma de saber o sería casi imposible determinar cuál es la imagen original. Contextualizando estos datos esta canal provoca pérdidas insalvables ya que nunca conocemos la imagen de salida siempre se ve la de llegada, con lo cual no es una canal apto para transmitir las imágenes de la sonda desde marte hacia la tierra.

### Conclusiones

El algoritmo utilizado para calcular media y desvío por muestreo computacional presenta comportamientos no deseados cuando la imagen tiene tiradas muy largas de tonos uniformes, esto se debe a que el algoritmo se basa en el grafo de transición de estados y que este tiene muchos ciclos.

Para imágenes con con poca variación de tonos el algoritmo optara por comprimir cada uno de sus bloques con RLC, este tipo de imágenes consta por lo general de de una seguidilla de píxeles del mismo tono.

Cuando se trata de imágenes donde existen muchas tonalidades de grises se selecciona el método de huffman.

Por otro lado se podría definir a los canales según la precisión de imágenes con la que se esté trabajando. Si lo que se desea es estudiar una imagen compleja con un nivel de detalle alto optaria por el canal 1, por otro lado si se desea enviar una imagen que no posee tantos tonos similares optaria por el canal 2.