

Universidad Nacional del Centro de la Provincia de Buenos Aires

Facultad de Ciencias Exactas

Teoría de la Información

Trabajo Final

Alumnos:

Lacoste Yamil yamil.lacost@gmail.com

Luti Alberto betomdq.93@gmail.com

Resumen

El presente trabajo consiste en el análisis e interpretación de los resultados obtenidos en la implementación de diversos algoritmos desarrollados durante la cursada de la materia.

En los trabajos previos se consideró una imagen en escala de grises como una fuente de información sin memoria, y se desarrollaron aplicaciones para: obtener la distribución de probabilidades de los símbolos de la fuente, obtener la codificación de huffman asociada a la fuente en estado estacionario, obtener la longitud media de código (Entrega 1); obtener la extensión a orden n de la fuente, codificar la imagen a nivel de bit, comprimir y descomprimir la imagen (Entrega 2). Para la tercer entrega, se debió analizar un canal de comunicación dado, para el cual se calculó: la matriz de dicho canal, el ruido, y la pérdida e información mutua (computacionalmente).

El objetivo de este trabajo es utilizar todo lo que hemos desarrollado para poder analizar y comparar dos imágenes. En primer lugar se codifican las imágenes mediante el algoritmo de huffman y se calcula la longitud media por símbolo de cada imagen, además se analiza lo que sucede luego de extender la fuente.

En segundo lugar se realiza la compresión de las imágenes y se calcula la tasa de compresión de las mismas. Finalmente, se verifica el teorema de Shannon y se estudia el ruido, pérdida e información mutua por medio de la simulación computacional.

Introducción

En el contexto de nuestro problema, se desea transmitir la imagen Star_8.bmp desde el observatorio donde ha sido adquirida hasta la central donde será procesada. Se sabe que el canal de transmisión tiene ruido. El archivo nebula bmp contiene la imagen luego de ser transmitida por este canal.

A partir de esta imagen y la original vamos a determinar por el algoritmo de Huffman la codificación de cada símbolo, la entropía y longitud media de la fuente para cada imagen, para poder lograr una compresión o reducción más eficiente de los datos enviados sobre la cantidad de información original, así como también las diferentes posibilidades de extensión de la fuente que permitan reducir el tamaño del mensaje enviado en la comunicación.

Además, vamos a evaluar lo que pasa con los resultados si se extiende el orden de las fuentes. Luego de obtener la codificación de Huffman y comparar los resultados se comprimió cada imagen y se analizó la tasa de compresión de las imágenes.

Se verificó el primer teorema de Shannon para la codificación de cada una de las imágenes.

Debido a que la transmisión de la imagen por el observatorio a la central no es ideal. Es por eso, que se desea estudiar por simulación computacional el ruido, la pérdida e información mutua de cada canal, teniendo en cuenta la probabilidad de ocurrencia de cada tono gris.

De esta manera, el problema central de este trabajo radica en un análisis y un estudio detallado de cada una de las imágenes enviadas por los distintos integrantes de la organización, comparando los resultados obtenidos entre cada una de ellas e interpretando la razón de los mismos.

Desarrollo

1)

El algoritmo de Huffman semi estático se basa en generar el árbol de Huffman a partir de las probabilidades de ocurrencia de los símbolos de la fuente. Cada valor de probabilidad pasa a ser un nodo hoja del árbol. A continuación, dado el conjunto de nodos, se eliminan los dos nodos que poseen las probabilidades más pequeñas, se genera y agrega al conjunto un nodo padre que contiene la suma de las probabilidades de los nodos hijos. Se

repite el paso anterior hasta que quede un solo nodo (su probabilidad asociada debería ser igual a uno), que pasa a ser llamado el nodo raíz. Cada nodo del árbol tendrá un cero o un uno como valor asociado dependiendo de si su probabilidad es menor o mayor con respecto a su nodo hermano respectivamente. Por último, el código asociado de un símbolo, es el código obtenido partiendo desde su nodo hoja asociado hacia la raíz.

Conociendo el código asociado a cada símbolo, se puede calcular la longitud media de código de la siguiente manera: $\langle L \rangle = \sum_i P_i * l_i$, donde P_i y l_i son la probabilidad y longitud de código asociados a un símbolo respectivamente.

2)

La compresión de la imagen consta de dos partes: la obtención del código de Huffman asociado a cada símbolo para luego codificar cada uno de los símbolos que forma la imagen.

Como resultado de este proceso se genera un archivo comprimido que consta de un encabezado con la información de la fuente (probabilidad de ocurrencia de cada símbolo y las dimensiones de la imagen), seguido por la codificación de la imagen a nivel de bits.

La tasa de compresión del algoritmo es igual al tamaño del archivo original sobre el tamaño del archivo comprimido y se expresa de la siguiente manera: (N:1).

3)

Shannon dedicó parte de sus estudios a determinar hasta cuánto puede ser reducida la longitud media de la codificación de una fuente sin perder información. También se dedicó a buscar alternativas de codificación que mejoren la longitud media por símbolo, como las extensiones de orden. En particular, para las fuentes con memoria y sin memoria, el primer Teorema de Shannon establece lo siguiente: $H(S) \leq \frac{\langle L_n \rangle}{N} \leq H(S) + \frac{1}{N}$

Siendo $H(S)$ la entropía de la fuente S , N el orden de extensión de la fuente, y $\langle L_n \rangle / N$ la longitud media de código para n símbolo de S . De dicha expresión podemos concluir que la longitud media de código de un símbolo de la fuente no puede ser menor a la entropía, lo cual tiene mucho sentido si consideramos a la entropía como la cantidad mínima de bits en promedio necesaria para representar los símbolos de la fuente.

Para la fuente sin memoria el N es igual a 1.

Podemos ver que la entropía se mantiene constante independientemente del orden de extensión, ya que sólo depende de las probabilidades de emisión de los símbolos. Pero si extendemos indefinidamente el orden de la fuente (N), podemos ver que el cociente $\langle L_n \rangle / N$ se hará cada vez más pequeño. En particular, con N tendiendo a infinito, el cociente alcanzará el mínimo posible: $\lim_{n \rightarrow \infty} \frac{\langle L_n \rangle}{N} = H(S)$

4)

Lo que se hizo fue calcular por muestreo computacional el ruido, pérdida e información mutua, y se analizó lo que sucede si se cambia la cota de convergencia.

Los pasos que se siguen hasta que converja es emitir un símbolo de entrada, aumentar en uno la cantidad de elementos de ese símbolo en la matriz conjunta y marginales, se aumenta en uno la cantidad de elementos y actualizó los valores del ruido, pérdida e información mutua del estado actual, se itera hasta que el error sea menor e igual a la cota de convergencia.

Resultados:

1)

Los resultados obtenidos al generar la codificación de Huffman para cada imagen fue la siguiente:

Para la imagen original Star para una fuente sin memoria se obtuvo:

- Longitud Media de la fuente 1.2673
- Entropía de la fuente 0.8719 bits.

Para la imagen Nebula para una fuente sin memoria se obtuvo:

- Longitud Media de la fuente 2.0481
- Entropía de la fuente 2.0239 bits.

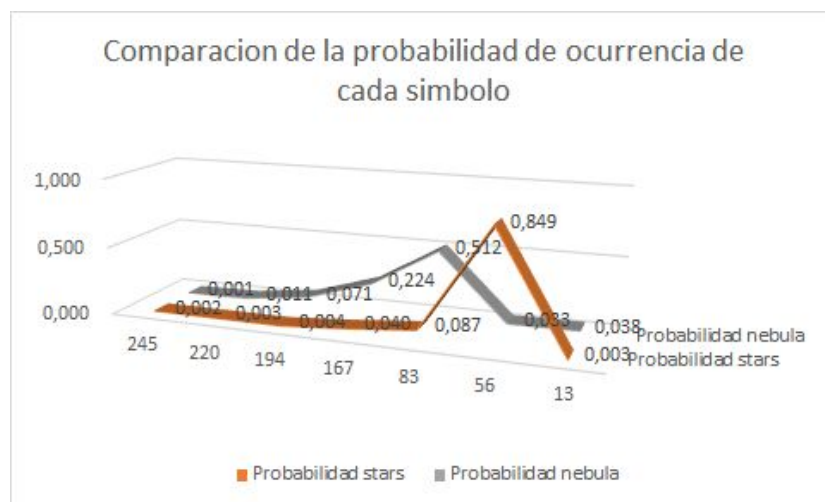
Para la imagen original Stars para una fuente con memoria se obtuvo:

- Longitud Media de la fuente para 2 símbolos: 1.8456
- Longitud Media de la fuente para 1 símbolo: 0.9228
- Entropía de la fuente 0.1268 bits.

Para la imagen Nebula para una fuente con memoria se obtuvo:

- Longitud Media de la fuente para 2 símbolos: 2.8299
- Longitud Media de la fuente para 1 símbolo: 1,4149
- Entropía de la fuente 1.2497 bits.

Se puede observar en el siguiente gráfico la comparación de las probabilidades de ocurrencia de cada símbolo para cada una de las imágenes:



Al comparar las probabilidades de ocurrencia de cada símbolo, se puede observar que la imagen Stars tiene el tono 56 con casi un 85% de probabilidad de ocurrencia y el 15% de las veces es muy probable que salga cualquier tono ya que tienen probabilidades bastante similares.

Por otra parte, en la imagen Nebula se observa que los símbolos tienen mayor

diversidad en cuanto a la probabilidad de ocurrencia, si bien podemos determinar que la mitad de las veces sale el tono 83, un 22% el tono 167, el resto de tonos tienen probabilidades muy parejas de ocurrencias.

Por lo tanto, es correcto que la entropía de Stars con respecto a la de Nebula sea mucho menor, ya que la cantidad de preguntas binarias para saber el tono que salió en Stars va a ser entre cero y una pregunta, debido a que el tono 56 sale un 85% de las veces.

En cambio en Nebula no se puede determinar con tanta precisión el tono que salió, debido a que solo se sabe que el 50% de las veces sale el tono 83, pero se hace muy difícil encontrar el otro 50%.

La longitud media de la fuente para la imagen Star resultó ser bastante superior que el límite mínimo teórico que es la entropía, pero comparado con la otra imagen la longitud media por símbolo resulta ser mucho menor que la imagen Nebula, debido a que en Star la probabilidad de representar con menor longitud los tonos más probables es mucho mayor.

Para una fuente con memoria, la longitud de la fuente por símbolo resulta menor que la longitud sin extender el orden, lo cual resulta beneficioso ya que lo máximo que vamos a poder reducir la longitud de la fuente es la entropía tomando una extensión la fuente n igual a infinito. Además al aumentar el orden de la fuente y tomar de a dos los símbolos emitidos, se obtiene ciertas similitudes con respecto a una fuente sin memoria, debido a que el análisis para Star y Nebula resultara similar, ya que la imagen Star al tener un símbolo con

mayor probabilidad de ocurrencia, la probabilidad de que un símbolo se repita va a ser alta, entonces vamos a poder representar dos símbolos en uno obteniendo mucha menor longitud y entropía que para la imagen Nebula y que para una fuente sin memoria.

2)

Al codificar la imagen Stars con la codificación de Huffman se obtuvo la siguiente tasa de compresión:

$\text{Factor de compresión} = \frac{215kb}{34kb} = 6,32$ dando por resultado el siguiente factor de compresión: 6,32:1

Al codificar la imagen Nebula con la codificación de Huffman se obtuvo la siguiente tasa de compresión:

$\text{Factor de compresión} = \frac{213kb}{78kb} = 2,73$ dando por resultado el siguiente factor de compresión: 2,73:1

Resultó mucho más favorable la compresión de Start debido a que tiene símbolos con mayor probabilidad de ocurrencia y representado con un tamaño óptimo.

3)

Verificación del primer teorema de Shannon sin memoria y con memoria respectivamente:

$$H(S) \leq \frac{\langle L_n \rangle}{N} \leq H(S) + \frac{1}{N} \quad H_1(S) + \frac{H(S) - H_1(S)}{N} \leq \frac{\langle L \rangle}{N} \leq H_1(S) + \frac{H(S) - H_1(S) + 1}{N}$$

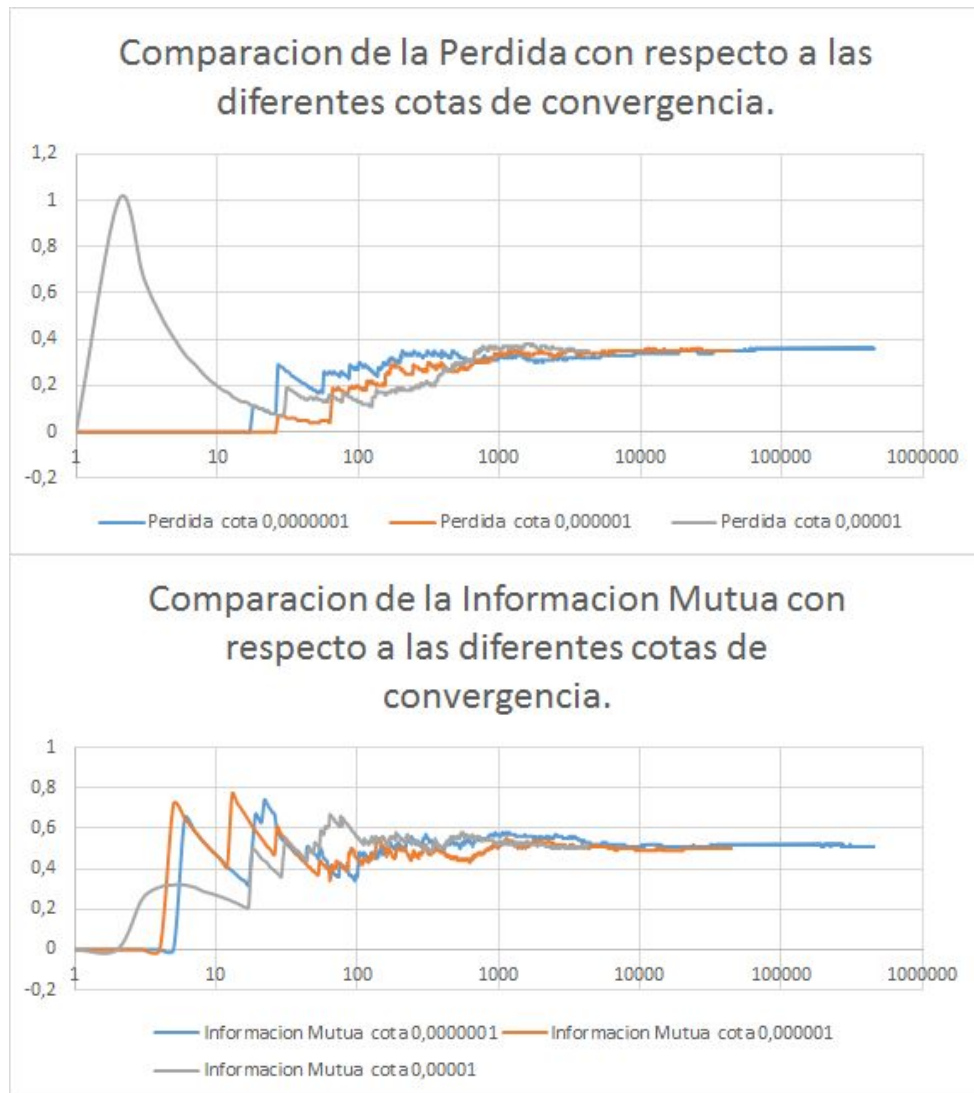
Reemplazando los valores obtenidos tenemos lo siguiente (valores aproximados):

- Imagen Stars sin memoria:
 $0.8719 \leq \frac{1.2672}{1} \leq 0.8719 + \frac{1}{1}$
 $0.8719 \leq 1.2672 \leq 1.8719$
- Imagen Nebula sin memoria:
 $2.0238 \leq \frac{2.0480}{1} \leq 2.0238 + \frac{1}{1}$
 $2.0238 \leq 2.0480 \leq 3.0238$
- Imagen Stars con memoria:
 $0.7450 + \frac{0.8719 - 0.7450}{2} \leq \frac{1.8455}{2} \leq 0.7450 + \frac{0.8719 - 0.7450 + 1}{2}$
 $0.8085 \leq 0.9227 \leq 1.3085$
- Imagen Nebula con memoria:
 $0.7742 + \frac{2.0238 - 0.7742}{2} \leq \frac{2.8299}{2} \leq 0.7742 + \frac{2.0238 - 0.7742 + 1}{2}$
 $1.399 \leq 1.4149 \leq 1.899$

4)



En el siguiente apartado se observa los gráficos de convergencia para cada valor calculado, y su interpretación de los mismos. Podemos notar, que para la menor cota se puede obtener una primer aproximación del ruido más



rápidamente que las demás cotas, pero con un valor de error mas elevado. En cambio cuando la cota de convergencia es mayor se necesita mayor cantidad de iteraciones para que pueda el ruido converger y por lo tanto obtener mayor grado de exactitud del ruido, esto sucede ya que se busca una cota de error menor para que pueda converger. Entre la cota azul y la naranja el resultado obtenido no varía mucho pero el costo computacional

aumento demasiado.

La pérdida, se comporta de manera similar al ruido, a menor error se necesita más iteraciones debido a que se necesita mayor cantidad de experimentos para que pueda converger y mejor va a ser el resultado

La información mutua mide la dependencia entre las variables de entrada y salida, podemos observar que a mayor error de convergencia la información mutua de las variables es menor y menos precisa, pero al aumentar las iteraciones, el valor se va estabilizando debido a que la cantidad de datos aumenta, hasta llegar a la convergencia.

Conclusion:

El grupo termina la experiencia de forma satisfactoria, aprendiendo el funcionamiento de diferentes algoritmos de codificación, compresión de imágenes, así como también el uso de la simulación computacional para la obtención del ruido, pérdida e información mutua de un canal. Se trabajó sobre dos imágenes en escala de grises para luego sacar conclusiones y determinar que representaban los resultados obtenidos.