

***“Perfeccionamiento de los Medios de Trabajo en el
proceso de enseñanza- aprendizaje en la asignatura
Televisión Digital”***

Autora: Yeilanis Romero Tápanes

Tutora: Ing. Irina Siles Siles

Contenido

1. Digitalización de la señal de video.....	5
Ejercicio #1.....	5
Resolución	5
Ejercicio #2.....	6
Resolución	6
Ejercicio #3.....	7
Resolución	7
Ejercicio #4.....	8
Resolución	8
Ejercicio #5.....	9
Resolución	9
Ejercicio #6.....	11
Resolución	11
Ejercicio #7.....	11
Resolución	12
Ejercicio #8.....	13
Resolución	13
Ejercicio #9 (Norma DTMB)	14

	2
Resolución	14
Ejercicio #10 (Norma DTMB)	16
Resolución	16
Ejercicio #11.....	19
Resolución	19
Ejercicio #12.....	20
Resolución	20
Ejercicio #13.....	21
Resolución	21
Ejercicio #14.....	22
Resolución	23
Ejercicio #15.....	23
Resolución	23
Ejercicio #16.....	23
Resolución	24
Ejercicio #17.....	27
Resolución	27
2. Digitalización de la señal de audio.....	27
Ejercicio #1.....	27
Resolución	27
Ejercicio #2.....	28
Resolución	28
Ejercicio #3.....	29
Resolución	29
Ejercicio #4.....	30
Resolución	30

	3
Ejercicio #5.....	30
Resolución	31
Ejercicio #6.....	32
Resolución	32
Ejercicio #7.....	33
Resolución	33
Ejercicio #8.....	33
Resolución	33
Ejercicio #9.....	33
Resolución	34
Ejercicio #10.....	34
Resolución	34
Ejercicio #11.....	35
Resolución	35
3. Métodos de codificación.....	36
Ejercicio #1.....	36
Resolución	36
Ejercicio #2.....	36
Resolución	37
Ejercicio #3.....	37
Resolución	38
Ejercicio #4.....	38
Resolución	38
Ejercicio #5.....	39
Resolución	40
Ejercicio#6.....	40

	4
Resolución	40
Ejercicio #7.....	43
Resolución	43
Ejercicio #8.....	45
Resolución	45
Ejercicio#9.....	47
Resolución	47
Ejercicio#10.....	48
Resolución	48
Ejercicio#11	49
Resolución	49
Ejercicio#12.....	49
Resolución	49
Ejercicio#13.....	50
Resolución	50
Ejercicio#14.....	51
Resolución	51
Ejercicio#15.....	54
Resolución	55
Ejercicio#16.....	56
Resolución	57
Ejercicio#17.....	58
Resolución	58
Ejercicio#18.....	58

1. Digitalización de la señal de video.

Ejercicio #1

Determinar el número de píxeles o muestras por línea activa, en el sistema NTSC con norma 525/60, si la frecuencia utilizada para el muestreo es 13,5 MHz. Considere solamente la señal de luminancia.

Resolución

NTSC (sistema de codificación y transmisión de televisión en color analógico)

- 30 imágenes por segundo formadas por 486 (492) líneas horizontales visibles.
- Video en modo entrelazado dividido en 60 campos por segundo.
- Total de 525 líneas horizontales y una banda útil de 4.25 MHz.

$$N_{\text{líneas/cuadro}} = 525$$

$$N_{\text{líneas/campo}} = 525/2 = 262,5$$

$$f_{\text{línea}} = N_{\text{líneas/cuadro}} \times f_{\text{cuadro}} = 525 \times 30 = 15750 \text{ Hz}$$

$$H = 1/f_{\text{línea}} = 1/15750 \text{ Hz} = 6,349206 \times 10^{-5} = 63,5 \mu\text{s}$$

$$K_H = H_a/H$$

K_H : Resolución Horizontal (0,84)

H_a : Tiempo activo de línea

H : Tiempo de línea

$$H_a = K_H \times H = 0,84 \times 63,5 \mu\text{s}$$

$$H_a = 53,3 \mu\text{s}$$

$$t_m = \frac{H_a}{N_{\text{muestras}} / \text{línea activa}}$$

t_m : Tiempo de una muestra.

$$N_{\text{muestras}} / \text{línea activa} = H_a/t_m = H_a \times f_m$$

$$\begin{aligned} N_{\frac{\text{muestras}}{\text{línea activa}}} &= 53,3\mu\text{s} \times 13,5 \text{ MHz} = 719.55 \text{ muestras/LTVA} \\ &\approx 720 \text{ muestras/LTVA} \end{aligned}$$

Si se considera solamente la señal de luminancia (Y) se va a tener una señal monocromática con diversas tonalidades de grises que estarán contenidas en 720 muestras por cada línea activa. Como la información referente al color está ausente (Cr y Cb) se puede decir que la cantidad de pixeles coincide con la cantidad de muestras por línea.

Ejercicio #2

Determinar el número de muestras por línea activa, en el sistema NTSC con norma 525/60, si la frecuencia utilizada para el muestreo es 13,5 MHz. Considere solamente la información de relacionada a la crominancia y que el formato de muestreo es 4:2:2.

Resolución

Tomando como referencia la resolución del ejemplo 3.2.1 en el que se calculaba la cantidad de muestras por línea activa para iguales condiciones se puede calcular de manera sencilla la cantidad de muestras de crominancia teniendo como premisa que por cada 4 muestras de luminancia existen 2 Cr y 2 Cb.

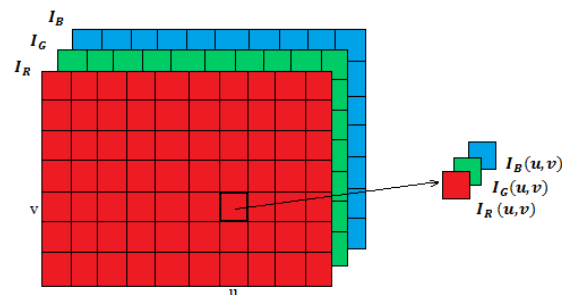


Fig. 1.1 Imagen en color RGB en el orden de los componentes. Los tres componentes de color se representan en matrices independientes del mismo tamaño

$$I_R(u, v) \quad I_G(u, v) \quad I_B(u, v)$$

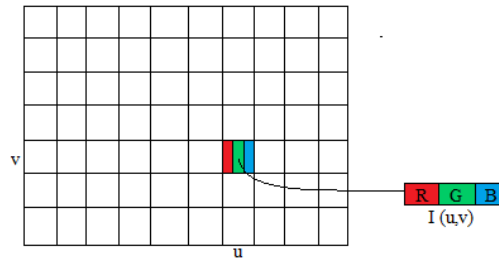


Fig. 1.2 Imagen de color RGB. Los tres componentes de color R, G, y B se colocan juntos en un solo elemento de matriz.

$$N_{\frac{muestras}{LTVA}} = 720/2 = 360 \frac{muestras}{LTVA} \text{ Tanto para Cr como para Cb.}$$

Ejercicio #3

La resolución de una cámara fotográfica digital está limitada por el sensor de la cámara que responde a las señales de luz. Este se compone de millones de “cubos” llamados píxeles que se cargan en respuesta a la luz, generalmente responden solamente a una gama limitada de longitudes de onda ligeras, debido a un filtro del color sobre cada uno. Calcule la cantidad de megapíxeles contenidos en una imagen, en una cámara digital que tiene una resolución de 2048×1536 utilizando como formato JPEG.

Resolución

En el mundo de la fotografía digital es muy común hablar de megapíxel, resolución y DPI. Una imagen digital está conformada por píxeles, dispuestos cual si fuera un rompecabezas, cuando todos los píxeles o elementos de imagen se juntan, forman una foto o cuadro. Cada megapíxel está constituido por un millón de píxeles, mientras que DPI significa “puntos por pulgada”, detalle que representa una computadora al procesar la imagen. Entre más megapíxeles tenga una imagen, será más clara y nítida y a su vez tendrá mayor tamaño de archivo.

En el caso del ejemplo, para calcular la cantidad de megapíxeles simplemente se debe multiplicar las dos dimensiones: $2048 \times 1536 = 3.1$ millones de píxeles, o 3.1 megapíxeles.

Ejercicio #4

Determine el flujo digital expresado en Mbit/segundos para un muestreo de tipo 4:2:2 si la frecuencia de muestreo para la señal de luminancia es de 13,5 MHz, considere que cada muestra se codifica como una palabra digital de 8 bit. Realice el mismo procedimiento para los formatos 4:2:0 y 4:4:4.

Resolución

Como se implementa el formato 4:2:2, por cada 4 muestras de luminancia se muestrean 4 de crominancia 2Cr y 2 Cb:

$$R_y = 13,5 \times 10^6 \text{ muestras /seg} \times 8 \text{ bits/muestra} = 108 \text{ Mbit/seg}$$

R_y : Razón de muestreo de luminancia.

R_{Cr} y R_{Cb} : Razón de muestreo de canales de crominancia

$$R_{Cr} = R_y / 2 = 54 \text{ Mbits/segundo}$$

$$R_{Cr} = R_{Cb} = 54 \text{ Mbit/segundo}$$

$$R_T = R_y + R_{Cr} + R_{Cb}$$

R_T : Flujo digital total.

$$R_T = 108 \frac{\text{Mbits}}{\text{segundo}} + 54 \frac{\text{Mbits}}{\text{segundo}} + 54 \frac{\text{Mbits}}{\text{segundos}} = 216 \text{ Mbits/segundo}$$

Para la norma 4:2:0

$$R_y = 108 \text{ Mbit/seg}$$

$$R_{Cr} = R_y / 4$$

$$\begin{aligned} R_T &= R_y + R_y / 2 = R_y + 2 \times R_{Cr} = 108 \frac{\text{Mbits}}{\text{segundo}} + 108 \frac{\text{Mbits}}{\text{segundo}} / 2 \\ &= 162 \frac{\text{Mbits}}{\text{segundo}} \end{aligned}$$

Para la norma 4:4:4

$$R_y = R_{Cr} = R_{Cb} = 108 \frac{\text{Mbits}}{\text{segundo}}$$

$$R_T = 3 \times R_y = 324 \frac{\text{Mbits}}{\text{segundo}}$$

Evidentemente, como se esperaba el formato 4:4:4, contiene mayor cantidad de información lo que se traduce en mayor ancho de banda, esto lo hace útil solo para aplicaciones de alta calidad y fidelidad de la imagen como por ejemplo; se utiliza en postproducción TV para realizar efectos espaciales digitales; por su parte 4:2:0 y 4:2:2 se auxilian de las características del SHV eliminando aquella parte de la información que resulta difícil o imposible de distinguir dentro de una imagen como son las altas frecuencias (detalles); manteniendo una buena calidad de audio y video logrando que las razones binarias no sean tan elevadas y que se pueda lograr la decodificación en el receptor sin una complejidad de hardware muy elevada.

Ejercicio #5

Determine el número de total de macrobloques y la cantidad de muestras que conforman una imagen cuya resolución es 640×480 pixeles para los casos en que se usen las teorías de muestreo 4:2:2, 4:2:0 y 4:4:4.

Resolución

De acuerdo con el epígrafe 2.1 la imagen está formada por pixeles que se agrupan dentro de esta en arreglos de 8×8 elementos denominados bloques:

- Un bloque está formado por 8 × 8 pixeles.
- Tenemos una resolución de 640 × 480 pixeles.

Para el formato de muestreo 4:2:0:

$$\frac{640_{\text{muestras/linea}}}{8_{\text{muestras/bloque}}} = 80 \text{ bloques/linea} \quad \frac{480_{\text{muestras/linea}}}{8_{\text{muestras/bloque}}} = 60 \text{ bloques/linea}$$

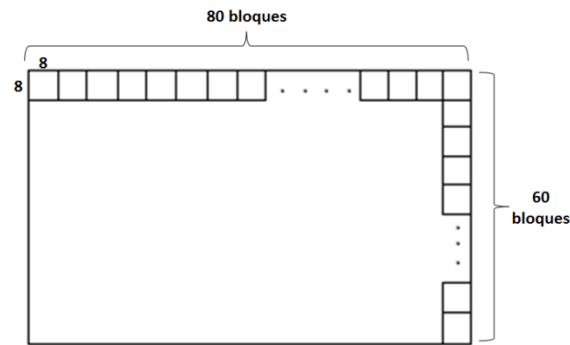


Figura 1.3 Cantidad de bloques en la imagen

- Los bloques de información se organizan en macrobloques, formados por 4 bloques de luminancia, constituyendo matrices de 16x16 elementos.

$$N_{MBh} = \frac{640}{16} = 40$$

$$N_{MBv} = \frac{480}{16} = 30$$

N_{MBh} : Número de macrobloques horizontales.

N_{MBv} : Número de macrobloques verticales.

- Total de macrobloques/imagen: $40 \times 30 = 1200$.

4 Bloques de luminancia (Y) = $16 \times 16 = 256$ pixeles/macrobloque

2 Bloques de Cr = $8 \times 8 \times 2 = 128$ pixeles/macrobloque

2 Bloques de Cb = $8 \times 8 \times 2 = 128$ pixeles/macrobloque

- Total de pixeles por macrobloque: $256 + 128 + 128 = 512$
- Total de pixeles por imagen: $1200 \frac{\text{macrobloques}}{\text{imagen}} \times 512 \frac{\text{pixeles}}{\text{macrobloque}} = 614400$.

Para la norma 4:2:0.

4 Bloques de luminancia (Y) = $16 \times 16 = 256$ pixeles/macrobloque

1 Bloques de Cr = $8 \times 8 = 64$ pixeles/macrobloque

1 Bloques de Cb = $8 \times 8 = 64$ pixeles/macrobloque

- Total de pixeles por macrobloque: $256 + 64 + 64 = 384$
- Total de pixeles por imagen $1200 \frac{\text{macrobloques}}{\text{imagen}} \times 384 \frac{\text{pixeles}}{\text{macrobloque}} = 460800$.

Para la norma 4:4:4

4 Bloques de luminancia (Y) = $16 \times 16 = 256$ pixeles/macrobloque

4 Bloques de Cr = $16 \times 16 = 256$ pixeles/macrobloque

4 Bloques de Cb = $16 \times 16 = 256$ pixeles/macrobloque

- Total de pixeles por macrobloque: $256 + 256 + 256 = 768$
- Total de pixeles por imagen $1200 \frac{\text{macrobloques}}{\text{imagen}} \times 768 \frac{\text{pixeles}}{\text{macrobloque}} = 921600$.

Ejercicio #6

SMPTE 274M es una de las normas más importantes de HDTV definidas por la SMPTE (Society of Motion Picture and Television Engineers, Sociedad de ingenieros de cine y televisión). Esta norma define una resolución de 1920x1080 píxeles con una alta fidelidad de color en formato 16:9 y utiliza barrido progresivo o entrelazado a 25/30 Hz y 50/60 Hz. Calcular la velocidad de muestreo teniendo los siguientes datos correspondientes a la norma:

- ✓ Total de muestras/línea: 2200 muestras/línea
- ✓ Total de líneas/cuadro: 1125 líneas/cuadro
- ✓ Cantidad de cuadros/segundo: 30 cuadros/segundo

Resolución

Multiplicando estos tres elementos y realizando un dimensionamiento obtenemos:

$$V_m = 2200 \text{muestras/linea} \times 1125 \text{lineas/cuadro} \times 30 \text{cuadros/segundo}$$

Por lo tanto la velocidad de Muestreo = 74.25×10^6 *muestras/segundo*. Esta velocidad de muestreo se traduce en que la frecuencia del reloj de muestreo tiene que ser de 74.25 *MHz*.

Ejercicio #7

Analice la influencia del muestreo en los pixeles en los sistemas de 625 y 525 líneas.

Resolución

Las muestras de luminancia y diferencia de color de un cierto valor de la señal de video se traducen, en la presentación, en el elemento de imagen más pequeño llamado pixel. En el caso del sistema 625 LTV con 576 líneas de televisión activas (LTVA) y cada una contiene 720 muestras de luminancia. Si el tamaño de un elemento de imagen (pixel) es $a \times b$, se tendrá que verificar lo siguiente si la relación de aspecto de la pantalla es $4/3$:

$$\frac{a \times 720}{b \times 576} = \frac{4}{3}$$

De donde se obtiene:

$$\frac{a}{b} = 1,07$$

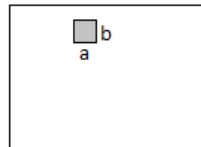


Figura 1.4 Tamaño de un pixel de dimensiones $a \times b$

Por lo tanto, el pixel no es cuadrado, es un 7% mayor en horizontal que en vertical.

Análogamente, en el estándar 525 LTV (480 LTVA):

$$\frac{a' \times 720}{b' \times 480} = \frac{4}{3}$$

$$\frac{b'}{a'} = 1,125$$

El pixel tampoco es cuadrado, es un 12,5% mayor en vertical que en horizontal. Mencionar que las resoluciones horizontal y vertical de un elemento espacial (pixel) son inversamente proporcionales a sus dimensiones horizontal y vertical respectivamente.

Por otra parte, el sistema visual tiene una acuidad o poder separador ligeramente mayor en horizontal que en vertical. Por lo tanto, las resoluciones horizontal y vertical de los pixeles del estándar 525 LTV se adaptan mejor que las de 625 LTV a las acuidades respectivas del sistema visual. Además, las resoluciones espaciales de ambos tipos de pixeles son:

$$Res(625) = \frac{K}{a \times b}, \text{ K es una constante de proporcionalidad.}$$

$$Res(525) = \frac{K}{a' \times b'}$$

Suponiendo que el ancho de los barridos de las líneas TV es igual en ambos estándares (b=b'):

$$\frac{Res(625)}{Res(525)} = \frac{1}{1.125 \times 1.07} = 0.83$$

La resolución espacial de un pixel en el estándar 625 LTV es inferior a la del pixel del estándar 525 LTV. (Cubero Enrici)

Ejercicio #8

Una imagen digital tiene una resolución espacial de 150×250 pixeles, donde cada uno de los mismos puede tomar uno de 64 valores posibles dentro de la escala de grises. ¿Qué cantidad de información contiene dicha imagen y cuánto tarda en transmitirse a 1000 bits/segundo?

Resolución

Para tener la información que contiene la imagen necesitamos saber la información que hay en un pixel, por consiguiente:

Si cada pixel puede tomar uno de 64 valores posibles, tenemos que:

$$i(pixel) = \log_2 64 = 6 \text{ bits}$$

$i(pixel)$: Información del pixel

Como la imagen tiene 150×250 pixeles, la cantidad de información de dicha imagen sería:

$$i(imagen) = i(pixel) \times 150 \times 250$$

$$i(imagen) = 6 \times 150 \times 250 = 225\,000 \text{ bits}$$

Para saber cuánto tarda en transmitirse la imagen a una velocidad de 1000 bits/segundo:

$$t = \frac{i(imagen)}{V} = \frac{225\,000 \text{ bits}}{1000 \text{ bits/seg}} = 225 \text{ seg}$$

Ejercicio #9 (Norma DTMB)

Compare para el sistema DTMB de 6MHz, la trama de señal para encabezados de trama de 420, 525 y 945 símbolos.

Resolución

La estructura de trama del sistema DTMB obedece a una disposición TDS-OFDM (Time Domain Synchronous -Orthogonal Frequency-Division Multiplexing) y consiste de dos partes, el encabezado y el cuerpo de trama. El encabezado está formado por un preámbulo, una secuencia PN, y un epílogo para un total de 420, 595 o 945 símbolos. El cuerpo de trama consiste de 36 símbolos que contienen información del sistema y 3744 símbolos portadores de datos. (Ver figura 2.14)

Para el caso de las norma DTMB de 6MHz y 8MHz se sabe que comparten el mismo diagrama básico de Tx/Rx, que el ancho de banda y la carga útil (payload) se pueden ajustar por un factor de $\frac{3}{4}$ y que la estructura de la trama, sincronización, esquema de modulación, estimación de canal y FEC son compatibles, de ahí que como se vio en el capítulo 2 se puede calcular la razón de símbolos como:

$$V_{T(6MHz)} = \frac{3}{4} V_{T(8MHz)} \rightarrow V_{T(6MHz)} = \frac{3}{4} (7.56 \text{ Msps}) = 5.67 \text{ Msps}$$

$$V_T = \frac{\#Simbolos}{T} \rightarrow T = \frac{\#Simbolos}{V_T} = \frac{3780}{5.67} \rightarrow T = 666.667 \mu s$$

Para el Modelo 1, cuando el encabezado de trama incluye 420 símbolos se puede calcular el intervalo de encabezamiento como:

$$V_T = \frac{\#Simbolos}{T} \rightarrow T = \frac{\#Simbolos}{V_T} = \frac{420}{5.67} \rightarrow T = 74.1 \mu s$$

Para el Modelo 2, cuando el encabezado de trama incluye 595 símbolos se puede calcular el intervalo de encabezamiento como:

$$V_T = \frac{\#Simbolos}{T} \rightarrow T = \frac{\#Simbolos}{V_T} = \frac{595}{5.67} \rightarrow T = 104.9 \mu s$$

Para el Modelo 3, cuando el encabezado de trama incluye 945 símbolos se puede calcular el intervalo de encabezamiento como:

$$V_T = \frac{\#Simbolos}{T} \rightarrow T = \frac{\#Simbolos}{V_T} = \frac{945}{5.67} \rightarrow T = 166.6 \mu s$$

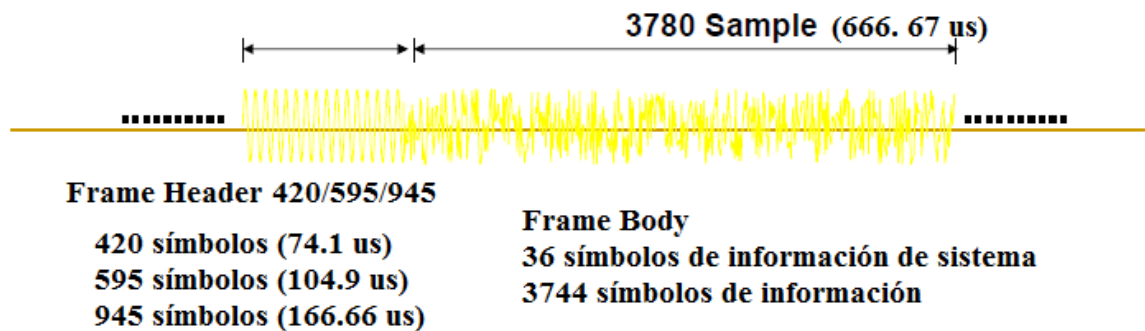


Figura 1.5 Disposición TDS-OFDM

La modulación por multiplexación por división de frecuencias ortogonales (OFDM), envía un conjunto de portadoras de diferentes frecuencias, moduladas a su vez en QAM o PSK, donde cada una de ellas transporta unidades pequeñas de información, es la elegida en el estándar principalmente por su robustez frente al multitrayecto, además de por la buena eficiencia espectral, que constituye uno de los más importantes objetivos a cubrir por el nuevo sistema de televisión. Esta modulación, unida a la inserción de intervalo de guarda y el uso de codificación de canal son los tres mecanismos provistos para mitigar los posibles efectos negativos del canal radio terrestre y asegurar la protección necesaria contra los altos niveles de interferencia cocanal y de canal adyacente que puedan surgir como consecuencia de la presencia de otras emisiones o servicios.

El principio básico de la modulación OFDM, como hemos adelantado ya, consiste en dividir la información a transmitir entre un gran número de portadoras independientes de más baja velocidad, que se suman de forma ortogonal, es decir, cada subportadora tiene un cero en la frecuencia de la subportadora adyacente. Cada una de ellas está modulada con uno de los tipos de modulación posibles, desde QPSK hasta 64-QAM. La separación de frecuencias de subportadoras es exactamente el inverso de la duración de los símbolos de baja velocidad.

Tener una menor tasa de símbolos por portadora se traduce en un periodo de símbolo más grande, lo que proporciona protección contra los ecos producidos por los múltiples caminos que toma la señal en su propagación, sobre todo en ámbitos urbanos como consecuencia de las reflexiones producidas en edificios.

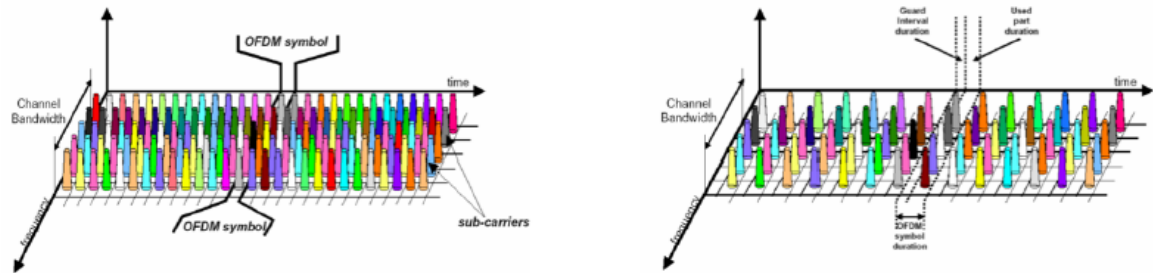


Figura 1.6 Estructura de flujo OFDM con y sin intervalo de guarda

Por otro lado, la inclusión del intervalo de guarda conlleva evidentemente un pequeño (o no tan pequeño, según la relación de su longitud frente a la longitud de símbolo) descenso en la tasa de bits, dado que no todo el símbolo transmite ahora información útil.

Los resultados obtenidos evidencian que los modelos que utilizan menor cantidad de símbolos en el frame header aunque presenta mayores razones de transmisión tienen el pequeño inconveniente de otorgar un menor intervalo de guarda, lo que trae como consecuencia que esté presente el fenómeno conocido como interferencia inter símbolo.

Ejercicio #10 (Norma DTMB)

Tabule para la norma DTMB, para canales de 6 y 8 MHz, con esquemas de modulación 4, 16y 64 QAM, encabezado de trama de 420/595/945 símbolos (PN) y FEC 0.4/0.6/0.8 la tasa de carga útil de datos o tasa de transmisión.

Resolución

La fórmula para calcular la tasa de carga útil de datos en el estándar DTMB es la siguiente:

$$Rate = \frac{3744}{PN + 3780} \times Ri \times Rm \times Fb \text{ (Mbps)}$$

Rm: Constituye la eficiencia que depende de la modulación QAM a utilizarse (2/4/6/8).

Fb: Tasa de símbolos banda base del encabezado y cuerpo de la trama (5.67 y 7.56Mbps).

Ri: Valor definido por una de las tres tasas de codificación FEC (LDPC+BCH)

PN: Secuencia de encabezado de trama (420/595/945)

Las tres razones de código que utiliza el FEC, son 0.4/0.6/0.8/ y están dadas por la cantidad de grupos BCH que tienen que concatenarse para entrar al codificador LDPC. El codificador externo utilizado como se mencionó en el capítulo 2 es del tipo BCH el cual ha de adecuarse como función adicional a la corrección en sí, el tamaño de la trama LDPC, para ello toma 752 bits de información (correspondientes a la mitad de la trama TS de 188Bytes (1504 bits)), le agrega 261 bits de relleno de ceros y a continuación se implementa la corrección una vez que se le agregan los 10 bits de paridad obteniéndose 1023 bits codificados. Cuando se obtiene la palabra de código se extraen los 261 ceros en la cabecera, esto es posible puesto que el código es sistemático y coloca los bits de información por delante de los de chequeo. En el caso cuando se concatenan 4 grupos BCH ($762 \times 4 = 3048$), entran al codificador LDPC un total de 3048 bits, en este bloque son eliminados 40 bits que correspondían a los 10 bits de paridad introducidos por el código BCH producto de la concatenación en 4 grupos, eliminando estos bits se codifican entonces 3008 bits, el codificador LDPC previamente sabe dónde encontrar estos bits de paridad puesto que el codificador BCH los sitúa después de los bits de información. El código LDPC es del tipo LDPC (7493,3048), se obtiene una palabra de código de 7493 bits de información que seguidamente son eliminados los primeros cinco símbolos de paridad de la palabra código LDPC. Así que el total de símbolos de la palabra de código LDPC es 7488 finalmente. La razón de codificación se obtiene de dividir $3008/7488=0,4$

Cuando se concatenan 6 grupos BCH, entran al codificador LDPC un total de 4572 bits, en este bloque son eliminados 60 bits que correspondían a los 10 bits de paridad introducidos por el código BCH producto de la concatenación en 6 grupos, eliminando estos bits se codifican entonces 4512 bits, se obtiene una palabra de código de 7493 bits de información que seguidamente son eliminados los primeros cinco símbolos de paridad de la palabra código LDPC. Así que el total de símbolos de la palabra de código LDPC es 7488 finalmente. La razón de codificación se obtiene de dividir $4512/7488=0.6$.

Al concatenarse 8 grupos BCH entran al codificador LDPC un total de 6096 bits, en este bloque son eliminados 80 bits donde codifican entonces 6016 bits, se obtiene una palabra de código de 7493 bits de información después de codificar LDPC, eliminados los primeros cinco símbolos de paridad de la palabra código LDPC queda un total de

símbolos de 7488. La razón de codificación se obtiene de dividir $6016/7488=0.8$. (Ver figura 2.12)

Code Rate	Block Length (bits)	Information Bits
0.4	7488	3008
0.6	7488	4512
0.8	7488	6016

Figura 1.7 Razones de codificación

Para ilustrar el modo de implementar el cálculo de la carga útil se toma como ejemplo un canal de 8 MHz, modulación 16 QAM, encabezado de 420 símbolos y una tasa de codificación a 0.4, los valores mostrados en la tabla se llevan a cabo de una manera similar:

$$Rate = \frac{3744}{420 + 3780} \times 0.4 \times 4 \times 7.56 (Mbps)$$

$$Rate = 10.78 Mbps$$

CANAL 8MHz				
Encabezado de Trama	Constelaciones Mapeadas	FEC		
		0.4	0.6	0.8
PN 420	4 QAM	5.414	8.122	10.829
	16 QAM	10.829	16.243	21.658
	64 QAM	16.243	24.365	32.486
PN 595	4 QAM	5.198	7.797	10.396
	16 QAM	10.396	15.593	20.791
	64 QAM	15.593	23.390	31.187
PN 945	4 QAM	4.813	7.219	9.626
	16 QAM	9.626	14.438	19.251
	64 QAM	14.438	21.658	28.877

CANAL 6MHz				
Encabezado de Trama	Constelaciones Mapeadas	FEC		
		0.4	0.6	0.8
PN 420	4 QAM	4.061	6.092	8.122
	16 QAM	8.122	12.281	16.244
	64 QAM	12.281	18.274	24.365
PN 595	4 QAM	3.882	5.823	7.764
	16 QAM	7.764	11.645	15.527
	64 QAM	11.645	17.468	23.29
PN 945	4 QAM	3.610	5.414	7.219
	16 QAM	7.220	10.829	14.438
	64 QAM	10.829	16.244	21.658

Tabla 1.1 Tasas de datos en el estándar DTMB para canales de 8 y 6MHz

La idea que persigue mostrar la tabla, es que con 5 valores de modulación, 3 valores de FEC, y 3 intervalos de guarda diferentes se pueden implementar más de 30 combinaciones posibles que arrojan tasas desde 4.813Mbps hasta 32,486Mbps para canales de 8 MHz y de 3.610Mbps hasta 24.365Mbps en canales de 6MHz. Estas combinaciones en la práctica son escogidas por los operadores de los equipos transmisores manualmente, teniendo en cuenta la C/N, características geográficas que condicionan el multitrayecto e IIS, la potencia de trasmisión, etc.

Ejercicio #11

Responda V o F según sea el caso. Justifique cualitativa y cuantitativamente aquellos marcados con (*).

- a) En MPEG se definen tres tipos de imágenes: I, P y B. Las imágenes tipo P y B ofrecen una menor tasa de compresión debido a que en ellas se aplican técnicas de compresión espacial y redundancia temporal. (*)
- b) La slice es la unidad mínima de resincronización después de un error.
- c) La DTC aunque no soporta compresión alguna, permite la aplicación de otros métodos de compresión más contundentes. (*)
- d) En el formato 4:2:0, las frecuencias de muestreo/LTVA de luminancia y de crominancia coinciden. (*)
- e) En la codificación reversible (lossy) no se pierde ningún tipo de información.

Resolución

- a) F. La tasa de compresión de imágenes I es relativamente pequeña, comparable con la de JPEG con pérdidas. Ellas consisten ante todo de los coeficientes transformados y no contienen vectores de movimiento. En las imágenes P, su tasa de compresión es claramente mayor que las imágenes I. Las imágenes P requieren aproximadamente la mitad de los datos de las imágenes I, mientras que la imágenes B ofrecen el factor de compresión más alto que generalmente es de una cuarta parte de los datos de las imágenes I. (Rivas, 2006). Los cuadros I codifican redundancia espacial, mientras que los cuadros B y P codifican redundancia temporal. Debido a que los marcos adyacentes son a menudo bien correlacionados, los cuadros P pueden ser del 10% del tamaño de un cuadro I, y el cuadro B al 2% de su tamaño.

- b) V. La “slice” es la unidad mínima de resincronización después de un error. De aquí se deriva que mientras más “slices” haya en el flujo de datos, mejor va a ser la recuperación del sincronismo después de ocurrir el error, pero va a disminuir la eficiencia de la compresión de video debido al uso de bits adicionales para indicar los comienzos de “slices”
- c) F. La DCT tiene una buena capacidad de compactación de la energía al dominio transformado, es decir, consigue concentrar la mayor parte de la información en pocos coeficientes que se concentran en zonas de baja frecuencia, esta consigue un error prácticamente nulo en los dos extremos de la señal. (Tarrés Ruiz, 2012)
- d) F. Las señales diferencia de color Cb y Cr tienen cada una, una banda pasante nominalmente plana hasta 2.75 MHz. Este menor ancho de banda que el de la luminancia, se debe a la menor resolución de los conos fotocaptadores de los colores en el ojo humano, en relación con la de los bastones fotocaptadores de la luminancia. Por este motivo se decidió que la frecuencia de muestreo de cada componente Cb y Cr fuese la mitad de 13.5 MHz frecuencia de muestreo de la luminancia, es decir, 6.75 MHz. (Cubero Enrici)
- e) F. La compresión lossy, es decir, con pérdidas de datos, es más bien una reducción en la cantidad de información y no una compresión de la misma (eliminación de datos irrelevantes). Esto significa que en el proceso de compresión-reducción, parte de la información se pierde irremediamente. Se trata entonces de un proceso irreversible; no es posible recuperar la señal original en su totalidad efectuando el proceso inverso porque parte de la misma se perdió para siempre. Un método lossy es, por ejemplo el proceso de compresión denominado ADPCM.

Ejercicio #12

En el caso de las imágenes, existen tres maneras de reducir el número de datos redundantes. Menciónelas y explíquelas brevemente.

Resolución

En el caso de las imágenes, existen tres maneras de reducir el número de datos redundantes:

- Eliminar el código redundante: El código de una imagen representa el cuerpo de la información mediante un conjunto de símbolos. La eliminación del código redundante consiste en utilizar el menor número de símbolos para representar la información. Las técnicas de compresión por codificación de Huffman y codificación aritmética utilizan cálculos estadísticos para lograr eliminar este tipo de redundancia y reducir la ocupación original de los datos. (Cubero Enrici)
- Eliminar píxeles redundantes: La mayoría de las imágenes presentan semejanzas o correlaciones entre sus píxeles. Estas correlaciones se deben a la existencia de estructuras similares en las imágenes, puesto que no son completamente aleatorias. De esta manera, el valor de un píxel puede emplearse para predecir el de sus vecinos. Las técnicas de compresión Lempel-Ziv implementan algoritmos basados en sustituciones para lograr la eliminación de esta redundancia. (ej RLC). (Cubero Enrici)
- Eliminar redundancia visual: El ojo humano responde con diferente sensibilidad a la información visual que recibe. La información a la que es menos sensible se puede descartar sin afectar a la percepción de la imagen. Se suprime así lo que se conoce como redundancia visual. La eliminación de la redundancia está relacionada con la cuantificación de la información, lo que conlleva una pérdida de información irreversible. Técnicas de compresión como JPEG, EZW o SPIHT hacen uso de la cuantificación. (Cubero Enrici)

Ejercicio #13

Determine el número de byte necesario para almacenar una imagen a color sin comprimir, de tamaño 640×480 empleando 8, 10, 12 y 14 bits por canal en color, respectivamente.

Resolución

Por ejemplo, una imagen típica de escala de grises usa $k=8$ bits (1 byte) por píxel y valores de intensidad en el rango de $[0\dots255]$, donde el valor cero representa mínimo brillo (menor luminancia) (negro) y 255 máximo brillo (blanco). Una imagen en color RGB-codificada con 8 bits requeriría 8 bits para cada canal en color, para un total de 24 bits, mientras la misma imagen con 12 bits por cada información de color requeriría un total de 36 bits por píxel.

Utilizando 8 bits por cada información de color (24 bits por pixel)

1 pixel → 3 byte (información de rojo, azul y verde)

$$\text{cantidad de bytes} = \frac{\text{cantidad de pixeles} \times \text{cantidad de bits por pixel}}{8}$$

$$307200 \times 24 / 8 = 921600 \text{ byte}$$

Utilizando de 10 bits por cada información de color

1 pixel → 30 bits

$$\text{cantidad de bytes} = \frac{\text{cantidad de pixeles} \times \text{cantidad de bits por pixel}}{8}$$

$$\text{cantidad de bytes} = \frac{640 \times 480 \times 30}{8}$$

$$\text{cantidad de byte} = 1152000 \text{ byte}$$

Utilizando de 12 bits por cada información de color

1 pixel → 36 bits

$$\text{cantidad de bytes} = \frac{640 \times 480 \times 36}{8}$$

$$\text{cantidad de bytes} = 1382400 \text{ byte}$$

Utilizando de 14 bits por cada información de color

1 pixel → 42 bits

$$\text{cantidad de bytes} = \frac{640 \times 480 \times 42}{8}$$

$$\text{cantidad de bytes} = 1612800 \text{ byte}$$

Ejercicio #14

La necesidad de representaciones de imágenes compactas considera la cantidad de datos necesarios para representar una definición estándar con 720×480 por 24 arreglos de pixeles. Teniendo en cuenta que una película digital (o video) es una secuencia de fotogramas de video en la que cada trama es una imagen fija a todo color, empleando una frecuencia de cuadro de 30 cuadros/segundo diga: ¿A cuánto se deben acceder los datos de video digital SD (GB) en una película de 2 horas de TV?

Resolución

$$30 \frac{\text{frames}}{\text{sec}} \times (720 \times 480) \frac{\text{pixeles}}{\text{frame}} \times 3 \frac{\text{byte}}{\text{pixel}} = 31\,104\,000 \text{ byte/seg}$$

$$31\,104\,000 \frac{\text{byte}}{\text{seg}} \times (60)^2 \frac{\text{seg}}{\text{hora}} \times 2 \text{hrs} = 2.24 \times 10^{11} \text{ bytes}$$

O 224 GB (gigabyte) de datos. Se necesitan 27 discos DVD de doble capa de 8.5 GB (Suponiendo discos convencionales de 12 cm) para almacenarlo. Para reproducir una película de 2 horas en un solo DVD, cada cuadro debe ser comprimido en promedio por un factor de 26.3. (Gonzalez & Woods, 2002)

Ejercicio #15

Se tiene una imagen que presenta 2376 pixeles, un número de líneas 1250 a una frecuencia de cuadro de 25 cuadros/segundo. Cada pixel está representado por una palabra de 10 bits. Determine la frecuencia de muestreo para la componente de luminancia, y con esta para las componentes de color Cr y Cb si se sabe se usa el formato 4:2:0.

Resolución

$$\text{Frecuencia de muestreo} = 2376 \text{pixeles/cuadro} \times 1250 \text{lineas/cuadro} \times 25 \text{cuadros/seg} \times 10 \text{bit/pixel} = 74.25 \text{ MHz.}$$

Para la componente de luminancia y la mitad de este valor 37.125 MHz, para los componentes de color Cr y Cb.

Estos flujos de datos presentan cadencias de 74.25 Mega palabras por segundo (MWps) para la luminancia (Y), y de 37.125 MWps para los colores Cr y Cb. La primera operación que se realiza con estos flujos de datos es la multiplexación en el tiempo (TDM-Time Division Multiplexing) de los flujos correspondientes a las dos componentes de crominancia. (Rivas, 2006)

Ejercicio #16

Tomando como ejemplo una imagen cualquiera evalúe usando métodos gráficos comparativos cuanto afecta la cantidad de bit a utilizar para albergar información de la profundidad del color teniendo en cuenta que se trata de una imagen de 640x480.

Resolución

La profundidad de color es el número de bits utilizados para describir el color de cada pixel de la imagen. Cuanto mayor sea la profundidad de color de una imagen, más colores tendrá esta y, por tanto, la representación de la realidad podrá hacerse con más matices y colores más sutiles. Por ejemplo, si se dispone de 1 bit para describir el color de cada pixel, tan sólo puede elegirse entre dos colores: un color si el bit tiene el valor 0 (habitualmente negro) y otro color si el bit vale 1 (habitualmente blanco).



Figura 1.8 Fotografía con una profundidad de color de 1 bit

Se observa la fotografía, con una profundidad de 1 bit, que tiene el aspecto de una fotocopia de mala calidad.

Si se dispone de 8 bit para describir el color de cada pixel, la elección se encuentra entre 256 colores, pues $2^8 = 256$. Esta es una profundidad de color suficiente para las imágenes construidas en el modo denominado escala de grises, porque con 8 bits cada pixel puede adoptar un tono entre 256 valores posibles de gris, entre el negro absoluto (00000000) y el blanco absoluto (11111111). La imagen siguiente es la misma fotografía, en modo escala de grises, con una profundidad de color de 8 bits, lo que le da una calidad tonal más que suficiente.



Figura 1.9 Fotografía en modo escala de grises con una profundidad de color de 8 bit

Si los 8 bit disponibles para la profundidad deben designar colores, entonces se utiliza una tabla con los 256 colores más frecuentes, que incluyen obviamente el negro, el blanco y varios tonos de gris, para componer la imagen. Cada una de las 256 combinaciones posibles de unos y ceros de los 8 bits es un índice que permite acceder a la tabla. Por eso, a este tipo de imágenes se les conoce como de color indexado. A continuación se muestra la misma fotografía en una imagen de color indexado, con 8 bit de profundidad:



Figura 1.10 Fotografía en una imagen de color indexado con una profundidad de color de 8 bits

Y así, cuanto mayor sea la profundidad se utilizará una cantidad mayor de colores para describir la imagen. En la tabla siguiente se muestra el cálculo de los colores disponibles para cada profundidad:

Profundidad	Colores
1 bits	2
4 bits	16
8 bits	256
16 bits	65536
32 bits	4294967296

Tabla 1.2 Cálculo de colores disponibles para cada profundidad

Una imagen con mapa de bit de calidad está compuesta por varias capas: una para cada color básico (rojo, verde y azul, por ejemplo) y una para la luminosidad (de oscuro absoluto a luz absoluta). Por encima de 16 bits de profundidad, la descripción del color se divide por capas. Si la profundidad de color es de 16 bits, por ejemplo, se dedican 4 bits (128 niveles) a cada capa. Y si la profundidad es de 32 bits, cada capa utiliza 8 bits (256 niveles) para ajustar el color.



Figura 1.11 Fotografía en una imagen de color indexado con una profundidad de color de 16 bits

Se observa la misma foto de la Alhambra, con una profundidad de 16 bits. La superposición del color de las cuatro capas proporciona la delicadeza de matices de la imagen.

Ejercicio #17

¿Cuántos colores están presentes en una imagen?

Resolución

Una tarea menor, pero frecuente en el contexto de las imágenes en colores es determinar la cantidad de colores contenidos en una imagen determinada. Una forma de hacer esto sería crear y llenar un conjunto de histogramas con elementos de números enteros para cada color y, posteriormente, contar todas las células histogramas con valores mayores que cero. Sin embargo, desde una imagen en color de 24 bits RGB contiene potencialmente $2^{24} = 16.777.216$ colores, el conjunto de histogramas resultante (con un tamaño de 64 megabytes) sería más grande que la propia imagen en la mayoría de los casos. Una solución simple a este problema sería ordenar los valores de los píxeles en una matriz de píxeles de tal manera que todos los colores idénticos se colocan uno junto al otro. El orden de clasificación es, por supuesto, completamente irrelevante, y el número de bloques de color contiguos en el vector de píxeles ordenados se corresponde con el número de diferentes colores de la imagen. Este número puede ser obtenido simplemente contando las transiciones entre bloques de color vecinas.

2. Digitalización de la señal de audio.**Ejercicio #1**

Se tiene una señal de audio analógica, para el proceso de digitalización y posterior codificación en PCM se necesita calcular el número de niveles de cuantificación uniforme, sabiendo que la relación señal a ruido mínima es de 96 dB.

- a) Calcule el flujo de bits para las frecuencias de muestreo de 44,1 kHz y 48 kHz.

Resolución

$$\frac{S}{N} = \frac{3}{2} (2^b - 1)^2$$

$$10 \log \frac{S}{N} = 96$$

$$-1,76 + 6b = 96$$

$$b \approx 16$$

$$M = 2^{16} = 65\,536 \text{ niveles}$$

Esto significa que la señal de audio codificada en PCM con 16 bits, introduce sobre el bit de menor peso un ruido de cuantificación del valor de ese bit y que se traduce en una SNR de cuantificación igual a:

$$\frac{96}{16} = 6 \text{ dB/bit}$$

El flujo de bits que resulta con una $f_m = 44.1 \text{ KHz}$ para un sonido estéreo será:

$Fb = 2 \times 44.1 \times 16 = 1.4112 \frac{\text{Mb}}{\text{s}}$ y para 48 KHz 1.536 Mb/s . Estos flujos son aceptables para un CD cuya capacidad es de 640 Mbytes/s o superior, lo que asegura alrededor de una hora de sonido estéreo sin ningún tipo de compresión, pero demasiado elevado para ser introducido como sonido de acompañamiento de una imagen comprimida de baja calidad de MPEG-1. (Cubero Enrici)

Ejercicio #2

Si tenemos una señal de audio con un margen dinámico de -1V a $+1\text{V}$, y se cuantifica con 8 bit, determine la mínima resolución que se puede distinguir.

Resolución

La cuantificación es el segundo proceso dentro de la digitalización de una señal, precedido por el muestreo y seguido por la codificación. De comunicaciones se conoce que:

$$\Delta = \frac{MD}{2^{NB}}$$

Donde MD corresponde al Margen Dinámico de la señal (diferencia entre la amplitud máxima y la mínima) y NB representa el Número de Bits con el que estamos cuantificando.

$$\Delta = \frac{2}{2^8} = 0,0078125 \approx 7,8 \text{ mV}$$

Tendremos un paso de cuantificación de $7,8\text{ mV}$, que indica la mínima resolución que podremos distinguir.

Ejercicio #3

MP3 es un formato de compresión de audio digital que usa un algoritmo con pérdidas para conseguir un menor tamaño del archivo. En este formato existen varias diferencias respecto a los estándares MPEG-1 y MPEG-2, entre las que se encuentra el llamado banco de filtros híbridos que hace que su diseño tenga mayor complejidad. Las muestras de salida del banco están cuantificadas y proporcionan una resolución en frecuencia variable, 6×32 o 18×32 subbandas, ajustándose mucho mejor a las bandas críticas de las diferentes frecuencias. Usando 18 puntos, calcule el número máximo de componentes frecuenciales, y a partir de este calcule la resolución frecuencial, si la frecuencia de muestreo es de 24KHz.

Resolución

El formato MP3 se convirtió en el formato utilizado para streaming y compresión de audio con pérdida de mediana fidelidad gracias a la posibilidad de ajustar la calidad de la compresión, proporcional al tamaño por segundo (bit rate) y por lo tanto el tamaño final del archivo que podía llegar a ocupar sería de 12 o 15 veces menos que el archivo original sin comprimir. MP3 se basa en el modelo psicoacústico y se auxilia en las características del sistema auditivo para la compresión. Consiste en analizar espectralmente cada cierto tiempo (T) el sonido que se está captando para confeccionar la curva del umbral de audibilidad enmascarada de dicho sonido. De este umbral se determinan las frecuencias enmascaradoras que hacen inaudibles el resto de los sonidos del espectro analizado. La compresión de audio se obtiene por consiguiente, conservando solamente las amplitudes correspondientes a dichas frecuencias (16 para el ejemplo) que son detectadas por el banco de filtros (32) cuyos anchos de banda se reparten por igual en todo el margen de audio (20Hz a 20KHz).

$$18 \times 32 = 576 \text{muestras/sub banda}$$

Por lo tanto, el número máximo de componentes frecuenciales es 576.

Dando lugar a una resolución frecuencial de:

$$\frac{24000\text{Hz}}{576} = 41.67\text{Hz si } f_s = 48\text{KHz}$$

Ejercicio #4

Se desea producir una digitalización de alta calidad de una fuente de música de 20 kHz de ancho de banda.

Resolución

Determinaremos una tasa de muestreo razonable para esta fuente. Para minimizar la razón de muestreo del sistema, deseamos que los filtros antialiasing tengan un ancho de banda de transición pequeño. Los filtros realizables requieren un ancho de banda distinto de cero para la transición para la parte pasa banda y la atenuación requerida por fuera de la banda. Esto se llama ancho de banda de transición. La complejidad y los costos de los filtros con ancho de banda de transición pequeño, requieren una relación de compromiso entre el costo del pequeño ancho de banda de transición y los costos de una alta razón de muestreo, “trade off”, los cuales son de mayor almacenamiento y altas tasas de transmisión. En muchos sistemas de respuesta ha sido hacer el ancho de banda de transición entre un 10 y un 20% del ancho de banda de la señal.

Si contamos un 20% del ancho de banda de transición del filtro anti aliasing, tenemos una versión de ingeniería de la tasa de muestreo de Nyquist:

$$f_s \geq 2.2 f_m$$

Por la versión de ingeniería de la tasa de Nyquist, la tasa de muestreo debería ser más grande que 44.0 kmuestras/s. En materia de comparación, la tasa de muestreo estándar para un reproductor de audio digital de discos compactos es 44.1 kmuestras/seg, y la tasa de muestreo estándar para audio con calidad de estudio es 48.0 kmuestras/seg.

Ejercicio #5

La modulación por código de pulsos (PCM) es el nombre dado a la clase de señales en banda base obtenidas de la señales PAM cuantizadas codificando cada muestra cuantizada en una palabra digital. La fuente de información es muestreada y cuantizada a uno de los L niveles; luego cada muestra cuantizada es codificada digitalmente en una palabra de código de l bits. Para transmisión en banda base, los bits de la palabra de código serán transformados a formas de onda de pulsos. Asumimos que una señal

analógica $x(t)$ es limitada en un rango de -4 a $+4$ V. El tamaño del paso entre los niveles de cuantización ha sido puesto a 1 V.

- ¿Cuántos niveles de cuantización se emplean y cuántos bits tiene la palabra de código?
- Los niveles de cuantización son localizados en $-3.5, -2.5, \dots, +3.5$ V, asignamos el código número 0 (cero) al nivel -3.5 V, el código número 1 al nivel -2.5 V, y así, hasta el nivel $+3.5$ V. Cada número de código tiene su representación en aritmética binaria, yendo desde 000 para el código número 0 hasta 111 para el 7. ¿Por qué han sido elegidos de esta manera los niveles de voltaje, comparado con el uso de una secuencia consecutiva de enteros, 1, 2, 3, ...?

Resolución

- Por consiguiente, se emplean 8 niveles de cuantización y se emplean $l = \log_2 L$ bits de la palabra de código. $l = \log_2 L = 3$
- La elección de los niveles de voltaje se basa en dos restricciones. Primero, el tamaño de los intervalos entre los niveles debe ser igual; y segundo, es conveniente para los niveles ser simétricos en cero.

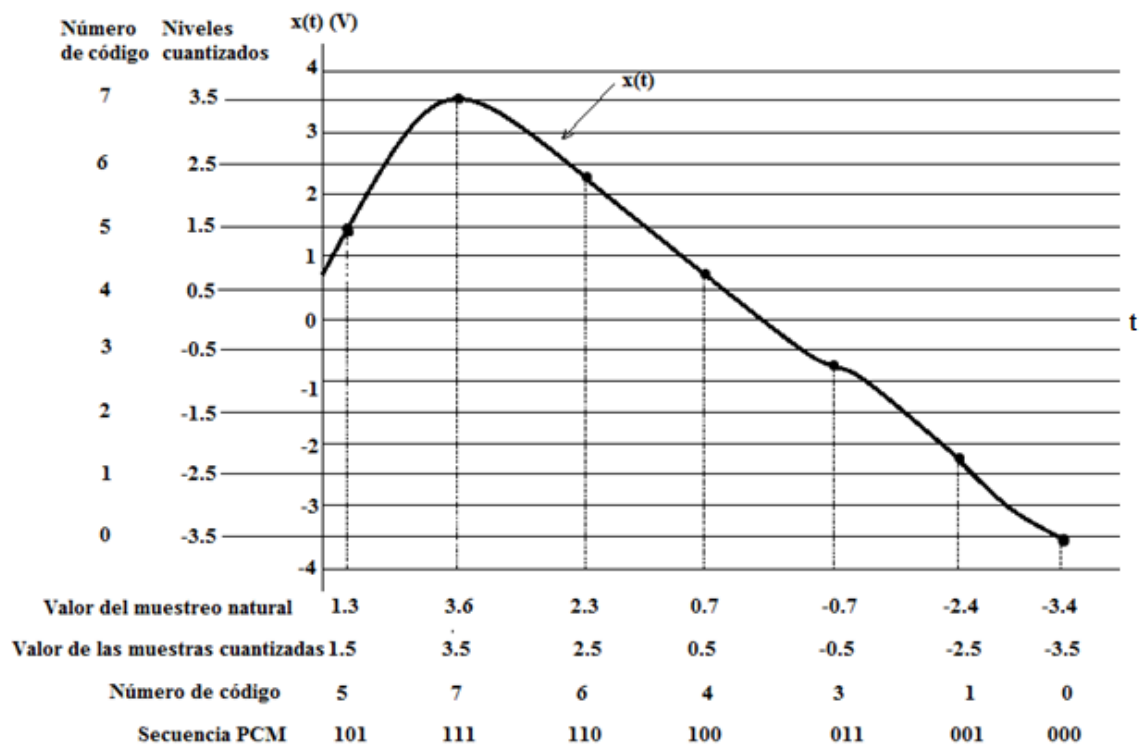


Figura 2.1 Muestreo Natural, muestras cuantizadas y modulación por código de pulsos

Ejercicio #6

La información en una forma de onda analógica, con una frecuencia máxima $f_m=3\text{kHz}$, es transmitida sobre un sistema PCM de M-niveles, donde el número de niveles de pulso es de $M=16$. La distorsión de cuantización se especifica que no excede el $\pm 1\%$ de la señal analógica.

- Cuál es el número mínimo de bits por muestra que debería usarse en el sistema PCM.
- Cuál es la mínima razón de muestreo requerida, y cuál es la razón de transmisión resultante.

Resolución

Una vez codificadas las muestras PCM, usando 8 bit, se tienen 2^8 o 256 posibles niveles de intensidad por muestra. La opción del número de niveles por muestra o bits por muestra depende de cuanta distorsión se tolere con el formato PCM. Es útil para desarrollar una relación general entre el número requerido de bits por muestra analógica (el tamaño de la palabra PCM) y la distorsión de cuantificación permitido. Se especifica la magnitud del error de distorsión de cuantificación, e , que no exceda de una fracción, p , de la tensión analógica de pico a pico, V_{pp} , de la siguiente manera:

$$|e| \leq pV_{pp}$$

$$e_{max} = \frac{q}{2} = \frac{V_{pp}}{2L}$$

Donde L es el número de niveles de cuantización. Entonces:

$$\frac{V_{pp}}{2L} \leq pV_{pp}$$

$$2^l = L \geq \frac{1}{2p} \text{ niveles}$$

$$l \geq \log_2 \frac{1}{2p} \text{ bits}$$

Es importante no confundir la idea de bit por palabra PCM, denotado por l , con el concepto de M-niveles de transmisión de k bits por símbolo.

- a) $l \geq \log_2 \frac{1}{0.02} = \log_2 50 \approx 5.6$. Por consiguiente, es apropiado $l = 6$ bits/muestra, para mantener el requisito de distorsión permitida.
- b) La mínima razón de muestreo es $f_s = 2f_m = 6000$ muestras/segundo. Del inciso a, cada muestra está compuesta por 6 bits de palabra PCM, por lo tanto la razón de bits es $R = lf_s = 36\,000$ bits/s

Ejercicio #7

Mencione la técnica más simple para reducir el volumen de datos en una señal digital.

Resolución

La técnica más simple es la reducción de la frecuencia de muestreo al de la cantidad de bits de cuantización. Si bien es un método bastante rudimentario y poco eficiente, todavía puede ser utilizado en algunas aplicaciones. Solamente hay que tener en cuenta que reduciendo la frecuencia de muestreo se reduce el ancho de banda que es posible procesar, y reduciendo los bits de cuantización disminuye el SQNR introduciéndose ruido en la señal.

Ejercicio #8

Si se observa que una determinada cadena de bits se repite con bastante frecuencia en la señal, no es necesario transmitirla todas las veces que aparezca. Explique.

Resolución

Se transmite la cadena una sola vez y luego se insertan bits de control que indican en qué lugares se deben reinsertar estas cadenas. Un ejemplo es la técnica de compresión Lossless que consiste en eliminar aquellos datos que son redundantes en la señal de audio (logran relaciones de compresión de 4:1).

Ejercicio #9

Mencione y explique el método de compresión que resulta exitoso ya que aprovecha cualidades perceptivas del oído humano para decidir qué información descartar.

Resolución

El método de compresión psicoacústica aprovecha cualidades perceptivas del oído humano para decidir qué información descartar. La compresión psicoacústica hace uso del fenómeno de la audición como “efecto de enmascaramiento” que consiste en una particularidad del oído humano por la cual escuchamos solamente algunos componentes de una señal acústica compleja, aquellos que tienen una mayor intensidad, y que ocultan otros que no se oyen. El efecto de enmascaramiento es más fuerte en las componentes del espectro más cercanas a la frecuencia enmascarante. (Ver epígrafe 2.2.1)

Ejercicio #10

Si se utiliza un codificador AC3 de 5.1 para la codificación de audio determine la tasa media de bits en sistemas multicanal, que se puede transmitir con comodidad utilizando la velocidad típica de trabajo del AC-3 (320 Kbps).

Resolución

El algoritmo AC3 considera los 6 canales como una entidad única incluyéndolos en una única trama de bit, con lo que se consigue una tasa de bit menor que separando cada canal en una trama distinta. En general la tasa media de bits en sistemas multicanal es directamente proporcional a la raíz cuadrada del número de canales. Si utilizamos 128 kb/s para codificar un único canal, una cantidad de 5.1 canales necesitará $128 \times \sqrt{5.1} = 289$ kb/s que se pueden transmitir con comodidad utilizando la velocidad típica de trabajo del AC-3 (320 kb/s). Por ello la mayoría de las veces será suficiente utilizar como método de compresión el algoritmo de asignación de bits. Sin embargo, cuando sea necesaria una mayor compresión se utiliza también el método de precombinación de portadora.

La principal ventaja de la codificación multicanal en una única trama es la posibilidad de distribuir la cantidad de bits con que se cuantificaran las mantisas de las componentes frecuenciales de cada canal, según las características de la señal, el modelo de enmascaramiento frecuencial de la audición humana y la velocidad de transmisión a la que se trabaje.

Ejercicio #11

Realice una comparación entre el ancho de banda de un canal telefónico analógico con uno digital. Determine el ancho de banda requerido sin interferencia inter-símbolo para transmitir R_s símbolos por segundo.

Resolución

Para el canal telefónico analógico se asigna un ancho de banda de 3 KHz. En el caso digital la voz es formateada en PCM con un conversor que toma 8000 muestras por segundo y cada muestra es cuantificada en 256 niveles. La secuencia de bits es transmitida en PCM y recibida con cero ISI.

El resultado de los proceso de muestreo y cuantización producen palabras PCM tal que cada palabra (representa una muestra) tiene uno de $L=256$ niveles diferentes. Si cada muestra fue enviada como un pulso PAM 256-ary, el ancho de banda requerido (sin ISI) para enviar R_s símbolos/seg será:

$$W \geq \frac{R_s}{2} \text{ Hz}$$

El signo igual será cierto solo para el caso de filtros ideales de Nyquist. Como el sistema telefónico digital usa formas de ondas PCM (binario), cada palabra PCM es convertida a $l = \log_2 L = \log_2 256 = 8$ bits. Por lo tanto el ancho de banda del sistema requerido para transmitir voz usando PCM es:

$$W_{\text{PCM}} \geq \frac{R_s}{2} = (\log_2 L) \frac{R_s}{2} = \frac{1}{2 \left(8 \frac{\text{bits}}{\text{símbolo}} \right) (8000 \text{ símbolos})} = 32 \text{ KHz}$$

Los 3 KHz del canal de voz analógico generalmente requieren 4KHz de ancho de banda, incluyendo la banda de guarda. Por lo tanto el formato PCM, usando cuantización de 8 bits y señales binarias con formas de ondas PCM, requiere por lo menos 8 veces el ancho de banda de un sistema analógico.

3. Métodos de codificación

Ejercicio #1

En la codificación RLC, el esquema de codificación se aplica en a aquellas fuentes de datos cuyos elementos se van transmitiendo a modo de secuencias repetidas de un mismo elemento. En la siguiente secuencia, si se usa un código ASCII de 7 bits.

accbbaaabb

- a) ¿Cuántos bits se transmitirían?
- b) ¿Cuántos bits se transmitirían usando Run Length Code?
- c) Determine el Factor de compresión.

Resolución

En vez de transmitir un código para cada elemento de la fuente, la idea es transmitir pares que, para el ejemplo dado serían:

$(1, a), (3, c), (2, b), (3, a), (2, b)$

Es decir, el primer término del par indica cuántas veces se transmite el segundo elemento del par. Es decir, la secuencia sería 1a3c2b3a2b, donde cada letra es reemplazada por el código ASCII correspondiente, por ejemplo:

00000011100001000001111000110000010110001000000111100001

Por lo tanto, si se usa un ASCII de 7 bits, se transmitirían $11 \times 7 = 77 \text{ bits}$, mientras que usando Run Length Code se transmiten $5 \times 2 \times 7 = 70 \text{ bits}$. Obviamente, en una secuencia más larga la relación de compresión se hace más grande. El factor de compresión ha sido $FC = 77\text{bits}/70\text{bits} = 1.1$

Esta idea es explotada en la codificación de mensajes de fax. Los mensajes de fax están formados por largas cadenas binarias que representan puntos blancos o negros.

Ejercicio #2

Mencione una variante para conseguir un flujo binario inferior al obtenido anteriormente con el código de longitud fija (RLC).

Resolución

Para conseguir un flujo binario inferior al obtenido con el código de longitud fija (RLC) la información obtenida se ha de recodificar con un nuevo código. Para lograr una nueva compresión se sigue el criterio de que la longitud de una palabra ha de ser función de la frecuencia o probabilidad de aparición de esa palabra en el mensaje; cuanto mayor sea esa probabilidad, menor ha de ser el número de bits que componen la palabra. Un ejemplo de esto es el código Morse.

Signo	Código	Signo	Código	Signo	Código
A	·—	N	—·	0	— — — — —
B	— · · ·	Ñ	— — — — —	1	· — — — —
C	— · — ·	O	— — — —	2	· · — — —
CH	— — — — —	P	· — — — ·	3	· · — — —
D	— · ·	Q	— — — —	4	· · · — —
E	·	R	· — ·	5	· · · · ·
F	· · — ·	S	· · ·	6	— · · · ·
G	— · — ·	T	—	7	— — · · ·
H	· · · ·	U	· — —	8	— — — · ·
I	· ·	V	· · — —	9	— — — — ·
J	· — — —	W	· — — —	.	· — — — —
K	— · —	X	— · · —	,	— — — — —
L	· — · ·	Y	— — — —	?	· · — — —
M	— —	Z	— · — ·	"	· — · · · ·
				!	— — — — —

Figura 3.1 Código Morse

El código Morse es difícil de aprender por lo que, para facilitar su aprendizaje, se suele utilizar una regla mnemotécnica, la cual permite aprenderlo mediante un código consistente en asignar a cada letra una palabra clave determinada, que comienza con la letra que se quiere recordar. Luego basta con sustituir cada vocal de la palabra clave por un *punto* o una *raya* según la siguiente regla.

Ejercicio #3

El código Reed Solomon es un subconjunto de los códigos BCH y son de bloques lineales; un código Reed Solomon se especifica como RS(n, k) con símbolos de s bits y añade símbolos de paridad para hacer una palabra de código de n símbolos. Un código popular Reed Solomos es RS (255, 223) con símbolos de 8 bits.

- a) Determine cuanto bytes contiene cada palabra de código
- b) Determine de estos bytes cuales son de datos y cuáles de paridad
- c) Determine la máxima longitud de la palabra de código

Resolución

- a) Cada palabra de código contiene 255 byte.
- b) De los 255 bytes, 223 bytes son datos y 32 de paridad.
- c) $n = 2^8 - 1 = 255 \text{ bytes}$

Ejercicio #4

Implemente la codificación de longitud variable (VLC) en el siguiente texto: “ata la jaca a la estaca”

Resolución

Se realiza un conteo de las veces que aparece cada carácter en el fichero a comprimir y se crear una lista enlazada con la información de caracteres y frecuencias.

“(5), a(9), c(2), e(1), j(1), i(2), s(1), t(2)

“: Representa el número de espacios entre cada palabra.

- Ordenamos por frecuencia de menor a mayor.
- Ordenar la lista de menor a mayor en función de la frecuencia.
- Convertir cada elemento de la lista en un árbol.
- Fusionar todos estos árboles en uno único, para hacerlo se sigue el siguiente proceso, mientras la lista de árboles contenga más de un elemento.
 - Con los dos primeros arboles formar uno nuevo, cada uno de los arboles originales en una rama.
 - Sumar las frecuencias. de cada rama en el nuevo elemento árbol.
 - Insertar el nuevo árbol en el lugar adecuado de la lista según la suma de frecuencias obtenida.

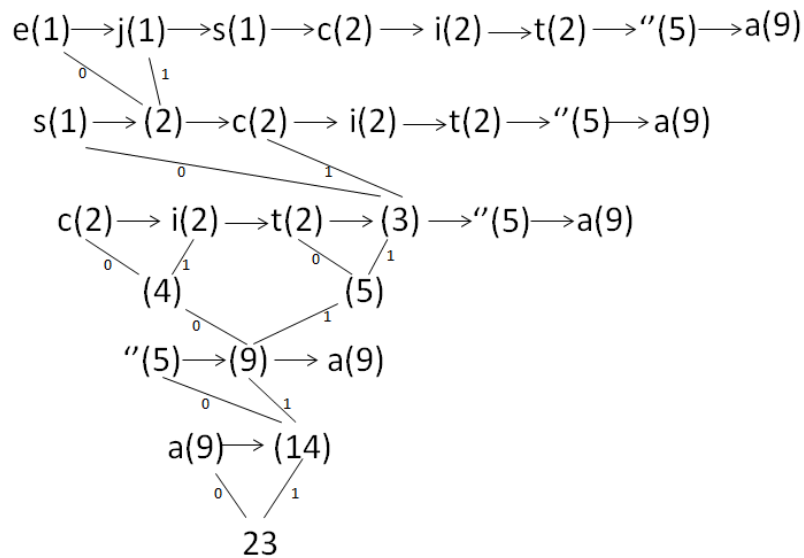


Figura 3.2 Diagrama del árbol

a	'	c	l	t	s	e	j
0	10	1100	1101	1110	11110	111110	111111

Traduciendo el texto tenemos:

a	t	a	'	l	a	'	j	a	c	a	'	a	'	l	a	'	e	s	t	a	c	a
0	111	0	10	110	0	10	1111	0	110	0	10	0	10	110	0	10	1111	111	111	0	110	0
	0			1			11		0				1			10	10	0		0		

El próximo paso sería empaquetar los bits en grupos de ocho:

0111001	1101010	1111101	0001001	1101010	1111011	1011100	1000000
0	1	1	0	1	1	1	0
0x72	0xD5	0xFB	0x12	0xD5	0xF7	0xb9	0x80

En total se envía la frase en 8 bytes, y al comienzo el texto original tenía 23 bytes.

Ejercicio #5

Para la decodificación de los códigos irregulares (LDPC) existen varios algoritmos, entre los cuales se pueden mencionar: suma de producto, etcétera. A partir de una matriz generadora G de la forma:

$$\mathbf{G} = \begin{bmatrix} 100001 \\ 010100 \\ 001010 \end{bmatrix}$$

Y una palabra de mensaje $s = [100]$ a transmitir.

- Obtenga la palabra de código.
- Describa brevemente lo que sucede con la palabra de código al ser enviada a través del canal de transmisión.

Resolución

- Al multiplicar s por \mathbf{G} se tiene la palabra de código: $y = [1001]$. La multiplicación se realiza en aritmética módulo 2.
- Esta palabra se envía a través del canal de transmisión, el cual le suma ruido a la señal transmitida; en el receptor se recibe como $r = y + n$, donde n , es el vector de ruido del canal de línea eléctrica de baja tensión. El receptor toma la palabra recibida r , y debe estimar la palabra transmitida.

Ejercicio#6

Se tiene la siguiente palabra de código $c = [11010101]$ a la entrada del receptor. Determine si esta fue enviada correctamente o presenta errores utilizando la decodificación LDPC si la matriz chequeo de paridad es la siguiente:

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Figura 3.3 Matriz de chequeo de paridad

Resolución

En el código de bloque lineal LDPC, se codifica efectuando la multiplicación del mensaje por la matriz generadora, mientras que la decodificación es en forma iterativa, usando un gráfico bipartito llamado “Gráfico Tanner”, descrito por la matriz de chequeo

de paridad. En general, estos algoritmos se pueden ver como algoritmos de envío de mensajes, llamados así porque en cada iteración se envía un mensaje desde cada check a cada variable relacionada y luego un mensaje de cada variable a cada check relacionado.

Para determinar la palabra de código válida es necesario construir el “Gráfico Tanner” correspondiente a la matriz de chequeo de paridad. La matriz H determina las conexiones entre los nodos-v ($C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7$ número de columnas) y los nodos-c (f_0, f_1, f_2, f_3 número de filas), en donde fluyen mensajes en forma iterativa desde los nodos-v a los nodos-c y viceversa. Adicionalmente en el grafico bipartito, el nodo de chequeo i se conecta al nodo variable j si el elemento (i, j) de la matriz de chequeo de paridad es uno.

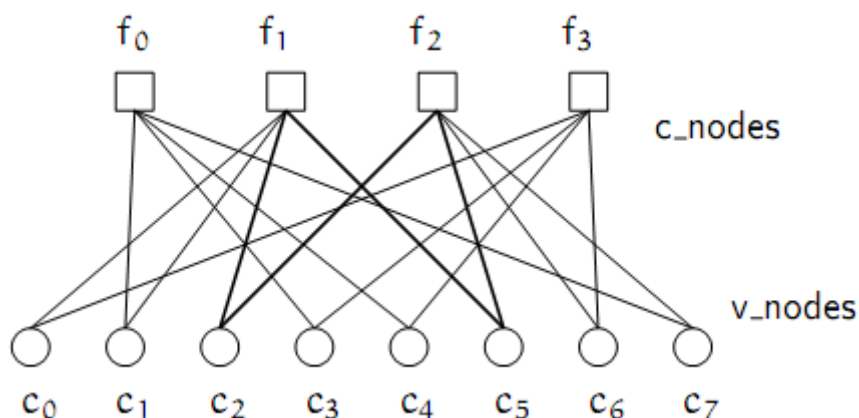


Figura 3.4 Gráfico Tanner correspondiente a la matriz de chequeo de paridad

Seguidamente, se sustituye el mensaje recibido en cada nodo variable, donde este fluye hacia los nodos de chequeo, y el proceso inverso se repite pero efectuando la suma módulo 2.

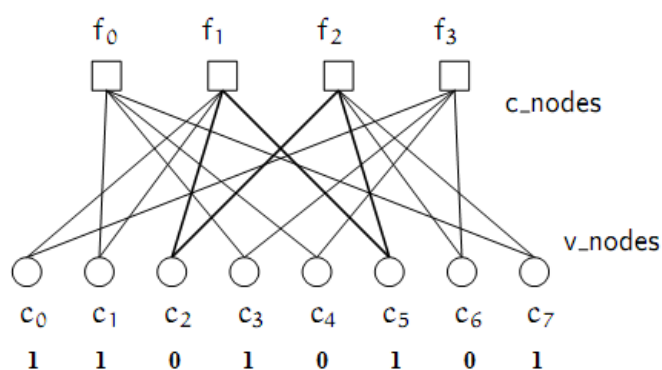


Figura 3.5 Sustitución del mensaje recibido a cada nodo variable

f_0 Recibe de los nodos variables c_1, c_3, c_4, c_7 , los bits 1, 1, 0, 1. El mensaje de respuesta de f_0 contiene el bit que este crea pueda ser correcto para los nodos variables correspondientes, mediante la operación suma módulo 2; por ejemplo para determinar qué bit le envía f_0 a c_1 se realiza la suma módulo 2 entre los bits enviados a f_0 (received) sin incluir el enviado por c_1 , para el caso de determinar el bit enviado por f_0 a c_3 se realiza la suma módulo 2 entre los bit enviados a f_0 sin incluir el enviado por c_3 , y así se realiza para los restantes nodos.

c-node	received/sent			
f_0	received: $c_1 \rightarrow 1$	$c_3 \rightarrow 1$	$c_4 \rightarrow 0$	$c_7 \rightarrow 1$
	sent: $0 \rightarrow c_1$	$0 \rightarrow c_3$	$1 \rightarrow c_4$	$0 \rightarrow c_7$
f_1	received: $c_0 \rightarrow 1$	$c_1 \rightarrow 1$	$c_2 \rightarrow 0$	$c_5 \rightarrow 1$
	sent: $0 \rightarrow c_0$	$0 \rightarrow c_1$	$1 \rightarrow c_2$	$0 \rightarrow c_5$
f_2	received: $c_2 \rightarrow 0$	$c_5 \rightarrow 1$	$c_6 \rightarrow 0$	$c_7 \rightarrow 1$
	sent: $0 \rightarrow c_2$	$1 \rightarrow c_5$	$0 \rightarrow c_6$	$1 \rightarrow c_7$
f_3	received: $c_0 \rightarrow 1$	$c_3 \rightarrow 1$	$c_4 \rightarrow 0$	$c_6 \rightarrow 0$
	sent: $1 \rightarrow c_0$	$1 \rightarrow c_3$	$0 \rightarrow c_4$	$0 \rightarrow c_6$

Figura 3.6 Corrección LDPC

A continuación, los nodos variables reciben el mensaje desde los nodos de chequeo y deciden sus bits originales mediante mayoría de voto.

Primeramente se coloca la palabra de código recibida que queremos decodificar para detectar los errores, en y_i received, seguidamente los bits que envía f_i a los nodos variables correspondientes, y la decisión tomada equivale por el bit predominante.

v-node	y_i received	messages from check nodes		decision
c_0	1	$f_1 \rightarrow 0$	$f_3 \rightarrow 1$	1
c_1	1	$f_0 \rightarrow 0$	$f_1 \rightarrow 0$	0
c_2	0	$f_1 \rightarrow 1$	$f_2 \rightarrow 0$	0
c_3	1	$f_0 \rightarrow 0$	$f_3 \rightarrow 1$	1
c_4	0	$f_0 \rightarrow 1$	$f_3 \rightarrow 0$	0
c_5	1	$f_1 \rightarrow 0$	$f_2 \rightarrow 1$	1
c_6	0	$f_2 \rightarrow 0$	$f_3 \rightarrow 0$	0
c_7	1	$f_0 \rightarrow 1$	$f_2 \rightarrow 1$	1

Figura 3.7 Determinación del valor real

Por consiguiente, la palabra de código enviada es [10010101].

Ejercicio #7

En el proceso de codificación de una imagen, se obtiene un bloque de 8x8 cocientes de DTC, donde los cuatro primeros valores 5, 4, 2, 1, 0 siguen el orden zig-zag, y los restantes son nulos. Si cada cociente se codifica PCM con 8 bits y la codificación es del tipo RLC. Determine el factor de compresión.

```

5 4 0 0 0 0 0 0
2 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

```

Resolución

Total de elementos - elementos no nulos = cantidad de ceros

$$\text{Cantidad de ceros} = 64 - 4 = 60$$

Como cada elemento se codifica con 8 bits: 60x8 bits = 480 ceros

ANTES DE LA CODIFICACION RLC (valor decimal)	5	4	2	1	0
ANTES DE LA CODIFICACION RLC (valor binario)	0 0 0 0 0 1 0 1	0 0 0 0 0 1 0 0	0 0 0 0 0 0 1 0	0 0 0 0 0 0 1	0 0 0 0 0 0 0 0
NUMERO DE BITS DESPUES DE LA CODIFICACION RLC	9 9 9 9	9 9	9 9	9 9	9 9

Figura 3.8 Resultado de la codificación RLC

Secuencias		Codificación RLC de b bits	
1ª	0	00	0
2ª	1	00	1
3ª	00	00	10
4ª	000	00	11
nª	0 000	11	11
	└─── n-1 ───┘		└─── b ───┘

Para expresar la cantidad de bits con que puedo representar el 480 para su codificación teniendo en cuenta el cero.

$$2^b \geq n + 1$$

$$2^b \geq 480 + 1$$

$$b = 9$$

$$FC = \frac{\text{número de bits antes de la codificación}}{\text{número de bits después de la codificación}}$$

$$FC = \frac{64 \times 8}{9 \times 11} = 5.2$$

Haciendo uso de un indicador de repetición sería lo siguiente:

$$0\rho 51010\rho 510\rho 810\rho 810\rho (480)$$

ρ : indica la cantidad de veces que se repite el número.

Codificando con 9 bits:

$$FC = \frac{64 \times 8}{21 \times 9} = 2,7$$

Codificando con 8 bits:

$$FC = 2,71$$

La codificación RLC, en cualquier versión, es rentable cuando abundan las redundancias, en caso contrario puede dar lugar a un aumento de datos (explosión de datos).

Ejercicio #8

Suponga la siguiente imagen binaria de 8×8 píxeles, considere grupos de 4 píxeles consecutivos como se muestra en los dos ejemplos de la figura. Utilice el método de Huffman para codificar los 16 grupos (bloques de cuatro píxeles que posee la imagen). ¿Cuántos bits se deberán transmitir?

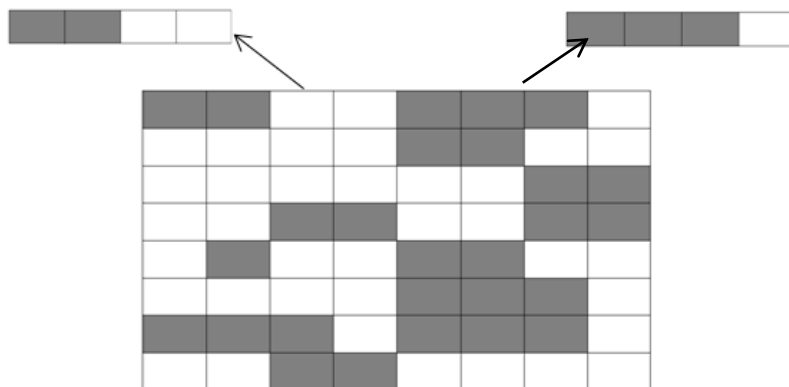


Figura 3.9 Imagen binaria de 8×8 píxeles

Resolución

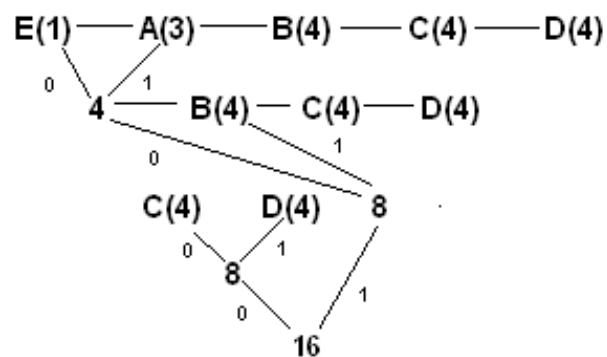
Siguiendo el procedimiento se cuenta las veces que aparece el carácter, y se ordena la lista de menor a mayor asignándole valores de cero y uno a cada rama.

Bloques	Cantidad de repeticiones	Código
A	3	1100
B	4	1110
C	4	0000
D	4	0011
E	1	0100
		64

Como se puede ver, la cantidad de bits transmitidos antes de la codificación es 64.

Aplicando la codificación:

3A	101
4B	11
4C	00
4D	01
1E	100



$$A = 3 * 3\text{bits} = 9 \text{ bits}$$

$$B = 4 * 2\text{bits} = 8 \text{ bits}$$

$$C = 4 * 2\text{bits} = 8 \text{ bits}$$

$$D = 4 * 2\text{bits} = 8 \text{ bits}$$

$$E = 1 * 3\text{bits} = 3 \text{ bits}$$

Total a transmitir= 36 bits de un total de 64 bits.

Obviamente, con la codificación, se reduce grandemente la cantidad de bits a transmitir, donde antes enviaban cuatro bits tres veces seguidas ahora con la codificación se mandan tres bits con una secuencia de tres.

$$36 + (4 + 3) + (4 + 2) + (4 + 2) + (4 + 2) + (4 + 3) = 68$$

Se observa que en este caso la tabla más los datos suman 68 bits. Por lo tanto se puede afirmar que la codificación resulta eficiente solo cuando existen algunos elementos que se repiten muchas veces.

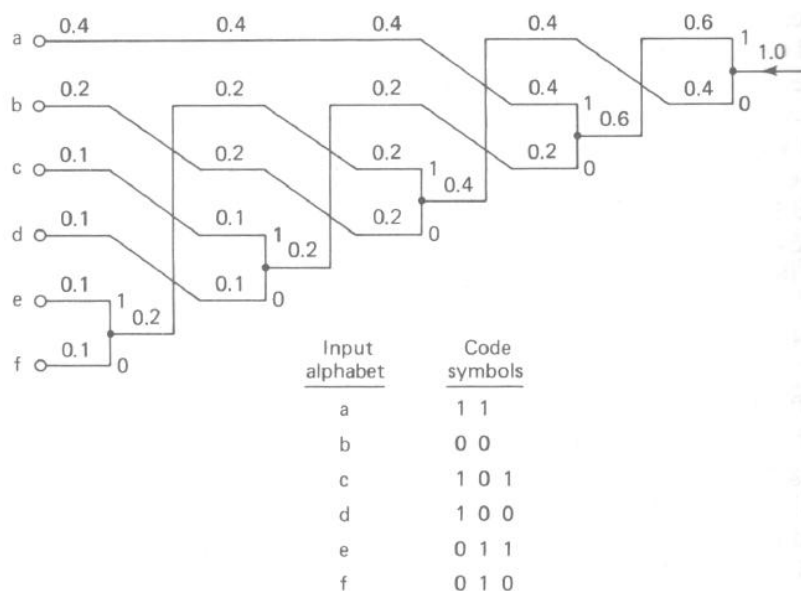
Ejercicio#9

En la tabla siguiente se muestran los seis símbolos de una fuente discreta sin memoria junto con sus respectivas probabilidades, ordenadas de manera decreciente. También se incluye el código resultante de la aplicación del algoritmo de Huffman, la longitud de cada código y el valor medio de la longitud de palabra de código. Calcule a longitud media y la relación de compresión de la palabra de código.

Símbolo	Probabilidad	Código resultante	l_k	$l_k \times p_k$
<i>a</i>	0,4	11	2	0,8
<i>b</i>	0,2	00	2	0,4
<i>c</i>	0,1	101	3	0,3
<i>d</i>	0,1	100	3	0,3
<i>e</i>	0,1	011	3	0,3
<i>f</i>	0,1	010	3	0,3
				$\bar{L} = 2,4$

Resolución

Durante el proceso de codificación, en cada columna se va eliminando un elemento, como consecuencia de la combinación de los dos últimos elementos de la columna anterior, que son los de menor probabilidad.



La longitud media de palabra de código es:

$$\begin{aligned}
 L &= 0,4 \times 2 + 0,2 \times 2 + 0,1 \times 3 + 0,1 \times 3 + 0,1 \times 3 + 0,1 \times 3 \\
 &= 2,4 \text{ bits/símbolo}
 \end{aligned}$$

La longitud media de palabra de código L , es 2,4 bits por carácter. Esto no significa que se transmite un número no entero de bits. Significa que en promedio, para transmitir 100 símbolos de la fuente, se transmiten 240 bits por el canal de comunicación.

Utilizando un sistema de codificación de longitud fija, se necesitarían 3 bits para poder codificar los 6 elementos de la fuente. Es decir, en promedio, por cada 100 símbolos se transmitirían 300 bits. Por lo tanto, la relación de compresión es $(3/2,4) = 1,25$.

Es importante tener en cuenta que el proceso de codificación de Huffman no es único (es decir, se puede llegar al mismo resultado por otro camino). Primero, en cada etapa es arbitraria la asignación del 1 y el 0 a las dos últimas ramas pudiendo invertirse el orden. Segundo, si la suma de estos dos elementos da como resultado una probabilidad que es igual a la de un elemento de la siguiente columna, este elemento resultante puede ubicarse ya sea lo más arriba posible como lo más abajo posible o en otro puesto intermedio en la columna siguiente (en el ejemplo anterior se ubicó primero), siempre respetando el orden decreciente de probabilidad para realizar el ordenamiento. Estas alternativas conducen a resultados diferentes en cuanto a la longitud de las palabras de código de cada símbolo, aunque la longitud media sigue siendo la misma.

Ejercicio#10

El formato de la cadena de bits que debe entregar el codificador MPEG está normalizado, este bloque se encarga de cumplir con dicho requerimiento, además de añadir cualquier información adicional que el usuario desee. El bitstream resultante está dividido en pequeños paquetes denominados frames (cuadros o marcos). Cada frames es totalmente independiente de cualquier otro, la razón de bits y la frecuencia de muestreo pueden contener entre 20 y 70ms de audio digital comprimido. Supóngase una frecuencia de muestreo típica de 44.1 kHz y una razón de bits de 128 Kbps. ¿Cuál será el tamaño del paquete?

Resolución

$$Frame\ Size = \frac{144 \times Bit\ Rate}{Sample\ Rate} [byte]$$

Frame Size: Tamaño del frame en byte

Bit Rate: Cadencia de bits elegida a la salida del compresor

Sample Rate: Frecuencia de muestreo de la señal de audio digital original

$$Frame\ Size = \frac{144 \times 128\ 000}{44\ 100} = 417\ byte$$

Represente aproximadamente 26, 12 ms de audio.

Ejercicio#11

Una fuente de mensajes se caracteriza por el número de mensajes que puede producir y por la probabilidad con la que se produce cada mensaje. Calcule la información asociada a cada uno de los mensajes de una fuente que puede transmitir 8 mensajes equiprobables.

Resolución

$$I_{(m)} = \log_2 \frac{1}{1/8} = \log_2 8 = 3\ bits$$

Ejercicio#12

El algoritmo de predicción utilizado en el proceso de compresión de la señal de video (según JPEG y MPEG) es el DPCM (Diferencial Pulse Code Modulation) que consiste en que el elemento estimado (E) es el elemento previo (P) anterior al actual (A).

$E=P$.

La diferencia D que se transmite será: $D=A-P$. Esta predicción se aplica a los cocientes no cero de la cuantificación de los coeficientes de la DTC, específicamente a los coeficientes DC.

- a) ¿Qué le sucede a la predicción DPCM según crece el movimiento de la escena?
- b) Puede suceder que el primer paso (A) para realizar una predicción interframe eficiente no sea nulo ($DPCM\ 3D \neq 0$) y sin embargo nada se ha movido en la escena actual ya que el elemento previo no existió, la predicción interframe se hace impracticable. ¿Cuándo se dará esta situación?

Resolución

- a) Cualquier movimiento de la escena es mayor que dos muestras en el tiempo de un cuadro (movimiento muy frecuente), da como resultado la pérdida de

correlación o identidad entre dos elementos de igual ubicación en la imagen previa y en la actual respectivamente y por lo tanto una interferencia total en la predicción inter (DPCM). Por consiguiente si el movimiento es importante, habría que adaptar la predicción interframe a las variaciones temporales de la escena (Predicción por compensación de movimiento).

b) Esta situación se dará cuando:

- La iluminación de la escena varía.
- El ruido de la señal de video falsee una imagen respecto a otra.
- Un objeto de la escena se deforme o sufra una rotación variando su forma y luminancia.
- Un objeto oculto en la imagen previa aparezca en la actual.
- La imagen previa esté difuminada con respecto a la actual.

Ejercicio#13

¿Bajo qué condiciones será posible la predicción interframe con compensación de movimiento?

Resolución

La predicción interframe con compensación de movimiento será posible cuando:

- Haya desplazamientos en la escena (rectilíneos, uniformes y en un plano perpendicular al eje óptico de la cámara) y sin que haya variaciones de la luminancia.
- Sabiendo que la luminancia de una escena sin movimientos no es uniforme, se aplique la corrección de la variación de la luminancia (Gain compensation interframe), en cuyo caso la predicción inter no tendrá necesidad de compensación de movimiento puesto que no lo hubo.
- Cuando haya oclusiones, rotaciones, deformaciones, desplazamiento de cámara y difuminaciones, al no ser iguales las imágenes previa y actual, se creará que hay movimiento en la escena y se buscara la imagen previa, al no encontrarla al cabo de un cierto tiempo, el sistema se inhibirá del problema sin realizar predicción interframe alguna.

Aunque parezca que la predicción inter con compensación de movimiento queda muy restringida con estas limitaciones, los resultados que se obtienen con los algoritmos son satisfactorios. El consumo de potencia de cálculo en una estimación de movimiento es comparable al de una compresión DTC.

Ejercicio#14

Comente cuanto se ve afectada una porción de tamaño 8×8 , de una imagen en el proceso de compresión y reconstrucción, con el estándar JPEG si se usa la DCT como algoritmo de compresión. (Gonzalez & Woods, 2002)

Se tiene en cuenta la matriz de cuantización:

52 55 61 66 70 61 64 73	16 11 10 16 24 40 51 61
63 59 66 90 109 85 69 72	12 12 14 19 26 58 60 55
62 59 68 113 144 104 66 73	14 13 16 24 40 57 69 56
63 58 71 122 154 106 70 69	14 17 22 29 51 87 80 62
67 61 68 104 126 88 68 70	18 22 37 56 68 109 103 77
79 65 60 70 77 63 58 75	24 35 55 64 81 104 113 92
85 71 64 59 55 61 65 83	49 64 78 87 103 121 120 101
87 79 69 68 65 76 78 94	72 92 95 98 112 100 103 99
a	b

Fig. a) Fragmento de la imagen de tamaño 8×8 , b) Matriz de cuantización

Resolución

La compresión se lleva a cabo en tres pasos secuenciales: cálculo de la DCT, cuantificación, y la asignación de códigos de longitud variable. La imagen a codificar es primero subdividida en bloques, arreglos de pixeles que conforman una matriz de 8×8 , los cuales son procesados de izquierda a derecha, de arriba abajo. Cada uno de estos bloques o sub-imágenes de 8×8 , 64 pixeles en conjunto, como parte del proceso de compresión se les desplaza el nivel de intensidad, sustrayéndole a cada valor cuantificado del arreglo 2^{n-1} (128) pues 2^n (256) es el máximo número de niveles de intensidad cuantificado, dejando solo el nivel de intensidad mínimo necesario para representar la información (se le quita toda la información referente al brillo) (c).

-76 -73 -67 -62 -58 -67 -64 -55	-415 -73 -67 -62 -58 -64 -64 55
-65 -69 -62 -38 -19 -43 -59 -56	-65 -69 -62 -38 -19 -43 -59 -56
-66 -69 -60 -15 16 -24 -62 -55	-66 -69 -60 -15 16 -24 -62 -55
-65 -70 -57 -6 -26 -22 -58 -59	-65 -70 57 -6 26 -22 -58 -59
-61 -67 -60 -24 -2 -40 -60 -58	-61 -67 -60 -24 -2 -40 -60 -58
-49 -63 -68 -58 -51 -65 -70 -53	-49 -63 -68 -58 -51 -65 -70 -53
-43 -57 -64 -69 -73 -67 -63 -45	-43 -57 -64 -69 -73 -67 -63 -45
-41 -49 -59 -60 -63 -52 -50 -34	-41 49 -59 -60 -63 -52 -50 -34
c	d

Fig. c) Matriz referente a un nivel de intensidad, d)

Se realiza entonces la transformada discreta del coseno directa el bloque cuantificado (c), y el redondeo de cada elemento al número entero más cercano, con el objetivo de concentrar la mayor parte de la información en pocos coeficientes transformados. Además, este proceso produce pocos errores en los límites de los bloques imagen. La minimización de los errores a los bloques imagen permite reducir el efecto de bloque en las imágenes reconstruidas. Una vez aplicada la DCT se obtiene la matriz mostrada en la figura (d).

El ojo humano es muy bueno detectando pequeños cambios de brillo en áreas relativamente grandes, pero no cuando el brillo cambia rápidamente en pequeñas áreas (variación de alta frecuencia). Aprovechando este defecto, se puede eliminar las altas frecuencias, sin pérdida excesiva de calidad visual. Esto se realiza dividiendo cada componente en el dominio de la frecuencia por una constante para ese componente (matriz de cuantización), y redondeándolo a su número entero más cercano. Este es el proceso en el que se pierde la mayor parte de la información (y calidad) cuando una imagen es procesada por este algoritmo. El resultado de esto es que los componentes de las altas frecuencias, tienden a igualarse a cero, mientras que muchos de los demás, se convierten en números positivos y negativos pequeños. (Gonzalez & Woods, 2002)

-26 -3 -6 2 2 0 0 0
1 -2 -4 0 0 0 0 0
-3 1 5 -1 -1 0 0 0
-4 1 2 -1 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
e

Haciendo el proceso de zig-zag, obtenemos el siguiente orden:

[-26 -3 1 -3 -2 -6 2 -4 1 -4 1 1 5 0 2 0 0 -1 2 0 0 0 0 -1 -1 EOB]

Donde el símbolo EOB indica la condición de fin de bloque. La construcción del código JPEG por defecto para la secuencias de coeficientes reordenados comienza por el cálculo de la diferencia entre el coeficiente DC actual y de la subimagen previamente codificada.

Para descomprimir el archivo comprimido JPEG, el codificador debe primero recrear los coeficientes de transformación normalizados que llevaron a la corriente de bits comprimidos. Fig (e)

Se lleva a cabo el proceso de desnormalización realizando el proceso inverso en cada elemento de la matriz de acuerdo con la ecuación $\hat{T}(u, v) = \hat{T}(u, v)Z(u, v)$. (En el proceso de codificación se corresponde al proceso inverso).

```

-416 -33 -60 30 48 0 0 0
 12 -24 -56 0 0 0 0 0
 -42 13 80 -24 -40 0 0 0
 -56 17 44 -29 0 0 0 0
 18 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0

```

f

Por ejemplo, el primer coeficiente DC se calcula como:

$$\hat{T}(0,0) = \hat{T}(0,0)Z(0,0) = (-26)(16) = -416$$

```

-70 -64 -61 -64 -69 -66 -58 -50
-72 -73 -61 -39 -30 -40 -54 -59
-68 -78 -58 -9 13 -12 -48 -64
-59 -77 -57 0 22 -13 -51 -60
-54 -75 -64 -23 -13 -44 -63 -56
-52 -71 -72 -54 -54 -71 -71 -54
-45 -59 -70 -68 -67 -67 -61 -50
-35 -47 -61 -66 -60 -48 -44 -44

```

g

La sub-imagen completamente reconstruida se obtiene tomando la transformada del coseno discreta inversa de la matriz sin normalizar. Fig. (g)

Seguidamente se procede al cambio de nivel de cada pixel sumándole 128 para producir:

```

58 64 67 64 59 62 70 78
56 55 67 89 98 88 74 69
60 50 70 119 141 116 80 64
60 51 71 128 149 115 77 68
74 53 64 105 115 84 65 72
76 57 56 74 75 57 57 74
83 69 59 60 61 61 67 78
93 81 67 62 69 80 84 84

```

h

Cualquier diferencia entre la sub imagen original y la reconstruida representa un resultado de la naturaleza con pérdidas de la compresión y descompresión JPEG. En este ejemplo los errores van desde -14 hasta -11 (Tarrés Ruiz, 2012) y se distribuyen de la siguiente manera:

```

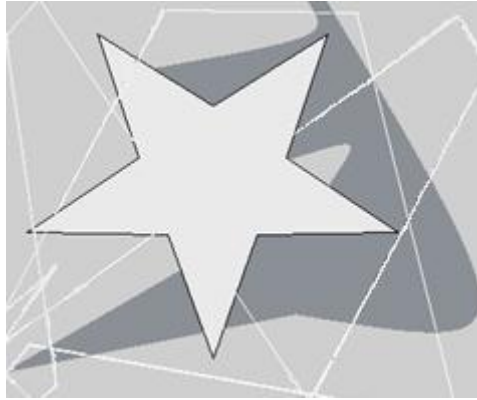
-6 -9 -6 2 11 -1 -6 -5
7 4 -1 1 11 -3 -5 3
2 9 -2 -6 -3 -12 -14 9
-6 7 0 -4 -5 -9 -7 1
-7 8 4 -1 6 4 -3 2
3 8 4 -4 2 6 1 1
2 2 5 -1 -6 0 -2 5
-6 -2 2 6 -4 -4 -6 10

```

i

Ejercicio#15

Un generador de imágenes de una computadora recrea la siguiente imagen de 256×256 8-bit por pixel. La distribución de probabilidad de ocurrencia de cada símbolo y la palabra de código empleada en la codificación de cada elemento (codificación PCM con 8 bits) se muestra en la tabla. Responda:

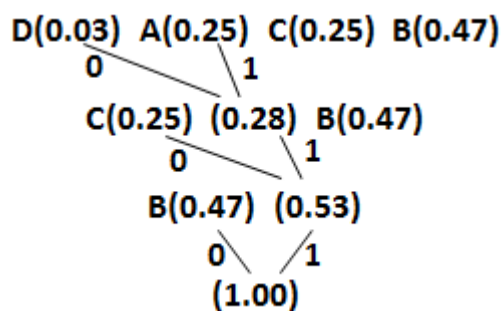


- ¿Qué tipo de redundancias se manifiestan en la imagen.
- ¿Cuál es el factor de compresión una vez que se usa codificación VLC?
- ¿Cuál es el número promedio de bits para representar cada pixel?

r_k	$p_k(r_k)$	Code 1	$L(r_k)$
R_{87}	0.25	01010111	8
R_{128}	0.47	10000000	8
R_{186}	0.25	11000100	8
R_{255}	0.03	11111111	8

Resolución

- Redundancia de codificación y redundancia espacial.
- $R_{87} \rightarrow 0.25 \rightarrow A$
 $R_{128} \rightarrow 0.47 \rightarrow B$
 $R_{186} \rightarrow 0.25 \rightarrow C$
 $R_{255} \rightarrow 0.03 \rightarrow D$



$$R_{87} \rightarrow 0.25 \rightarrow A \rightarrow 111$$

$$R_{128} \rightarrow 0.47 \rightarrow B \rightarrow 0$$

$$R_{186} \rightarrow 0.25 \rightarrow C \rightarrow 10$$

$$R_{255} \rightarrow 0.03 \rightarrow D \rightarrow 110$$

$$FC = \frac{\text{Valor fijo antes de VLC}}{\text{Valor medio de } \frac{\text{bit}}{\text{palabra}} \text{ después de VLC}}$$

$$FC = (256 \times 256 \times 8) / (256 \times 256 \times 3 \times 0.25 + 256 \times 256 \times 1 \times 0.47 + 256 \times 256 \times 2 \times 0.25 + 256 \times 256 \times 3 \times 0.03)$$

$$FC = 4.42$$

$$c) L = 0.25 \times 3 + 0.47 \times 1 + 0.25 \times 2 + 0.03 \times 3$$

$$L = 1.81 \text{ bits}$$

Ejercicio#16

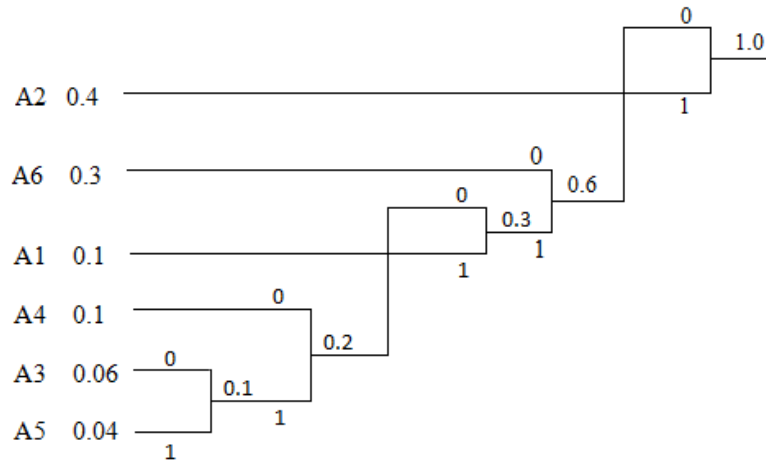
A continuación se muestra un conjunto de fuentes de símbolos y sus probabilidades están ordenadas de arriba abajo en términos de disminución.

- ¿Cuál es el número promedio de bits para representar cada pixel?
- Determine el factor de compresión una vez que se usa código Huffman.
- Calcule la entropía de la fuente.

A1	0.1
A2	0.4
A3	0.06
A4	0.1

A5	0.04
A6	0.3

Resolución



A2	0.4	1	1
A6	0.3	00	2
A1	0.1	011	3
A4	0.1	0100	4
A3	0.06	01010	5
A5	0.04	01011	5

$$L = 0.4 \times 1 + 0.3 \times 2 + 0.1 \times 3 + 0.1 \times 4 + 0.06 \times 5 + 0.04 \times 5$$

$$L = 2.2 \text{ bits/pixel}$$

FC

$$= \frac{512 \times 512 \times 8}{512 \times 512 \times 0.4 \times 1 + 512 \times 512 \times 0.3 \times 2 + 512 \times 512 \times 0.1 \times 3 + 512 \times 512 \times 0.1 \times 4 + 512 \times 512 \times 0.06 \times 5 + 512 \times 512 \times 0.04 \times 5}$$

$$FC = \frac{2097152}{104857.6 + 157286.4 + 78643.2 + 104857.6 + 78643.2 + 52428.8}$$

$$FC = \frac{2097152}{576716.8} = 3.6$$

Basándonos en el ejercicio anterior la cadena original sería:

gtrrrrrjuli0000jklppphj



Referencias Bibliográficas

Cubero Enrici, M. (s.f.). *La Televisión Digital: Fundamentos y Teorías*.

Gonzalez, R. C., & Woods, R. E. (2002). *Digital Image Processing*. New Jersey.

Rivas, I. A. (2006). *Introducción a la Tecnología de Video Moderna*. México.

Tarrés Ruiz, F. (2012). *Sistemas Audiovisuales. Televisión Analógica y Digital Parte II*.
La Habana: Félix Varela.