

ESCUELA POLITECNICA NACIONAL

FACULTAD DE INGENIERIA ELECTRICA

**"ANALISIS Y SIMULACION DE
LA COMPRESION DE IMAGENES"**

**TESIS PREVIA A LA OBTENCION DEL TITULO DE
INGENIERO EN ELECTRONICA Y TELECOMUNICACIONES**

JIMMY EFRAIN RUIZ PROCEL

Quito, Diciembre de 1995

AGRADECIMIENTO

A todas las personas e instituciones que colaboraron pacientemente en toda mi vida estudiantil hasta la culminación de este trabajo.

INDICE

CAPITULO I CONCEPTOS GENERALES

1.1	INTRODUCCION A LA DIGITALIZACION DE IMAGENES	2
1.1.1	Antecedentes	2
1.1.2	Captura de datos y preproceso	4
1.1.3	Digitalización	10
1.1.4	Modulación por amplitud de pulsos	10
1.1.5	Velocidad de muestreo de Nyquist	11
1.1.6	Modulación por impulsos codificados	12
1.1.7	MIC codificado uniformemente	14
1.1.8	MIC codificado no uniformemente	15
1.1.8.1	Compansión de la ley μ	16
1.1.8.2	Compansión de la ley A	18
1.2	CRITERIOS DE TRANSMISION DIGITAL DE IMAGENES	19
1.2.1	Transmisión de pulsos	20
1.2.2	Interferencia intersímbolo(IIS)	23
1.2.2.1	Sincronización inexacta	23
1.2.2.2	Ancho de banda insuficiente	23
1.2.2.3	Distorsión de amplitud	23
1.2.2.4	Distorsión de fase	24
1.3	COMPARACION ENTRE MEDIOS DE ALMACENAMIENTO Y TRANSMISION	24
1.3.1	Comparación de medios	24
1.4	COMPRESION DE DATOS vs. COMPRESION DE ANCHO DE BANDA	26
1.4.1	Imágenes comprimidas	26
1.4.2	Reducción del flujo digital de datos	31
1.4.3	Reducción del ancho de banda	31
1.5	HERRAMIENTAS MATEMATICAS UTILIZADAS EN LA COMPRESION	33
1.5.1	La matemática de la compresión	33
1.5.1.1	Probabilidad y Estadística	33
1.5.1.2	Teoría de la Información y Codificación	34
1.5.1.2.1	Información	34
1.5.1.2.2	Entropía	36
1.5.1.2.3	Acerca del ruido	38
1.5.1.2.4	Procesos de descompresión-corrección de errores	39
1.5.1.3	Análisis en el tiempo, el espacio y la frecuencia	40
1.5.1.3.1	El concepto de modulación	40
1.5.1.3.2	Análisis por transformada	42

**CAPITULO II
METODOS DE COMPRESION**

2.1 CLASIFICACION DE LOS METODOS DE COMPRESION DE IMAGENES	2
(i) Compresión con pérdidas ³	4
(ii) Compresión con muchas pérdidas	4
Codificación por longitud variable(CLV)	6
2.1.1 CODIFICACIÓN DE PÍXELES	8
2.1.1.1 Modulación por impulsos codificados	8
2.1.1.2 Codificación de la entropía	9
2.1.1.2.1 El algoritmo de codificación de Huffman	10
2.1.1.3 Codificación longitudinal(CL)	11
2.1.1.4 Codificación planar	12
2.1.2 CODIFICACIÓN PREDICTIVA	12
2.1.2.1 MIC Diferencial(MICD)	14
2.1.2.2 Modulación delta(MD)	16
2.1.2.3 MICD línea por línea	19
2.1.2.4 MICD bidimensional	19
2.1.2.5 Codificación adaptiva	21
2.1.2.6. Técnicas inter-trama/intra-trama	22
2.1.2.6.1 Repetición de tramas	22
2.1.2.6.2 Cambio de resolución	22
2.1.2.6.3 Abastecimiento condicional	23
2.1.2.6.4 Codificación predictiva con compesación en el movimiento	23
2.1.2.6.5 Codificación adaptiva predictiva	27
(a) Predicción adaptiva	27
(b) Cuantificación predictiva	28
2.1.2.7 Otros métodos	29
2.1.3 CODIFICACIÓN POR TRANSFORMADA	30
(a) Transformada de Karhunen-Loève	31
(b) Transformada de Fourier	33
(c) Transformada de Hadamard	35
(d) Transformada de Haar	37
2.1.3.1 Codificación adaptiva por transformada	38
(a) Muestreo adaptivo	38
(b) Transformación adaptiva	39
(c) Cuantificación adaptiva	39
2.1.3.2 Codificación multidimensional	39
2.1.3.3 La transformada coseno	40
2.1.4 OTRAS TÉCNICAS DE CODIFICACIÓN	49
2.1.4.1 Codificación híbrida	49
2.1.4.2 Codificación de gráficos de dos tonalidades	51
2.1.4.2.1 Codificación por omisión de bloques blancos(OBB)	54
(a) Codificación unidimensional	54
(b) Codificación bidimensional	55
2.1.4.2.2 Cuantificación diferencial predictiva(CDP)	55
2.1.4.2.3 Codificación de la dirección relativa(CDR)	57
2.1.4.2.4 Codificación por designación de la dirección relativa del elemento(CDDRE)-CCITT modificada	58

2.1.4.3	Codificación de imágenes a color	59
2.1.4.3.1	Métodos de codificación de componentes	61
2.1.4.3.2	Codificación de la señal compuesta	62
2.1.4.4	Codificación por cuantificación vectorial	62
2.1.4.4.1	Cuantificación vectorial adaptiva de imágenes monocromáticas	64
2.1.4.4.2	Cuantificación vectorial de imágenes a color . . .	65
2.1.4.5	Misceláneos	66
2.1.4.5.1	Codificación por truncamiento de bloque	66
2.1.4.5.2	Formatos de video(Compresión a nivel informático)	70
	(a) El formato IMG	71
	(b) El formato PCX	73
	(c) El formato GIF	74
	(d) El formato TIFF	76
	(e) Otros formatos	77
	EPS(Encapsulated PostScript)	77
	BMP(BitMaP)	78
	MTF(MetaFiles)	78
2.2	COMPARACION CUANTITATIVA Y CUALITATIVA DE LOS METODOS	78
2.2.1	Desempeño de la técnica predictiva	80
2.2.1.1	Comparación MIC versus MICD	80
2.2.1.2	MD, MICD Línea a Línea y MICD Bidimensional	83
2.2.1.3	MICD Adaptivo-MICD no Adaptivo	85
2.2.2	Desempeño de la técnica de codificación por transformada	90
2.2.2.1	Comparación transformada versus transformada . . .	91
2.2.3	Desempeño de la técnica de codificación de gráficos de dos tonalidades	94
2.2.4	Desempeño de la técnica de codificación de imágenes a color	96
2.2.5	Desempeño de cuantificación vectorial en la codificación de imágenes	98
2.2.6	Desempeño de la técnica de codificación por truncamiento de bloque	100

CAPITULO III APLICACIONES

3.1	LA COMPRESION EN EL CAMPO DE LA GRABACION DE VIDEO	2
3.1.1	Sub muestreo	4
3.1.2	Un esquema práctico de video-grabador digital	6
3.1.2.1	Reducción de entropía	8
3.1.2.2	Alta densidad de grabación	9
3.1.3	Compresión de video comercial	9
3.1.3.1	Perfil de una aplicación profesional	13
3.2	LA COMPRESION EN EL CAMPO DE LA RADIODIFUSION TELEVISIVA	13
3.2.1	Transmisión de video en movimiento	14
3.2.2	Jerarquías de los servicios B-ISDN	15
3.2.2.1	Alto ritmo de transmisión(H_3 ;120-140Mbps)	16
3.2.2.2	Mediano ritmo de transmisión(H_2 ;60-70Mbps)	16
3.2.2.3	Bajo ritmo de transmisión(H_1 ;34-44Mbps)	16
3.2.3	Codificación digital a un ritmo primario- H_1	16
3.2.4	Transmisión de video en H_2	17
3.2.5	El sistema M.U.S.E.	19
3.2.6	Perfil de una aplicación comercial	20
3.3	ESTANDARES DE COMPRESION	24
3.3.1	El estándar de JPEG	25
3.3.2	El estándar del CCITT	28
3.3.3	El estándar propuesto para total movimiento	30
3.3.3.1	El estándar MPEG-1	31
3.3.3.2	El estándar MPEG-2	32
	Imágenes-I	33
	Imágenes-P	33
	Imágenes-B	34

CAPITULO IV SISTEMA DE TRANSMISION-RECEPCION DE IMAGENES

4.1	ESTUDIO DEL SISTEMA	2
4.1.1	Sistema simplex	3
4.1.2	Sistema semi-duplex	3
4.1.3	Sistema duplex	4
4.2	ETAPA DE TRANSMISION-COMPRESION	5
4.2.1	Transductor de entrada	5
4.2.2	Procesador de entrada	6
4.2.2.1	Protección contra errores	6
4.2.2.2	Esquemas de modulación	8
4.2.2.2.1	Esquemas de modulación de fase	9
4.2.2.2.2	Representación en cuadratura de señales	11
4.3	CANAL DE TRANSMISION, INCIDENCIA DE RUIDO Y PERTURBACION	12
4.3.1	Tipos de canales	12
4.4	ETAPA DE RECEPCION-DESCOMPRESION	14
4.4.1	Corrección de errores	14
4.4.2	Etapa de descompresión	15
4.4.3	Transmisión de imágenes en dos tonalidades	15

CAPITULO V
APLICACIONES DE LOS METODOS DE COMPRESION
A UNA IMAGEN DE PRUEBA

5.1	ESTUDIO DE LA IMAGEN A COMPRIMIR	2
5.2	DESCRIPCION Y ANALISIS DE LOS METODOS DE COMPRESION A UTILIZARSE	3
5.2.1	Codificación por Truncamiento de Bloque	3
5.2.2	Codificación por medio de la Transformada Coseno	5
5.2.3	Codificación por medio de la MICD	7
5.3	DESCRIPCION DEL PROGRAMA COMPUTACIONAL UTILIZADO	8
5.3.1	Programa principal: COMPIMAG.PAS	11
5.3.2	Unidad VARIOS.TPU	12
5.3.3	Unidad VARIABLE.TPU	12
5.3.4	Unidad INICIAL.TPU	13
5.3.5	Unidad CUADROS.TPU	13
5.3.6	Unidad PROCESO.TPU	14
5.3.7	Unidad ANALISIS.TPU	15
5.4	VALORACION CUANTITATIVA Y CUALITATIVA	16
5.4.1	Compresión sin ruido	17
5.4.2	Compresión con ruido	21
5.4.3	Análisis Gráfico	27

CONCLUSIONES Y RECOMENDACIONES

BIBLIOGRAFIA

ANEXOS

- ANEXO I: Documentos del CCITT para prueba de fax.
- ANEXO II: Ejemplos de aplicación.
- (a) MIC Ley A
 - (b) Huffman
 - (c) MIC-D
 - (d) MD
 - (e) Transformada de KL
 - (f) Transformada de Fourier
 - (g) Transformada de Walsh-Hadamard
 - (h) Transformada de Haar
 - (i) Transformada Discreta Coseno
 - (j) OBB Unidimensional
 - (k) Truncamiento de Bloque
 - (l) Formatos PCX y BMP
- ANEXO III: Recomendación 601 del CCIR.
- ANEXO IV: Formato eléctrico de una señal de TV a color.
- ANEXO V: Glosario de términos.
- ANEXO VI: Estructura de un prototipo para la compresión-descompresión de imágenes.
- Introducción
 - Estudio del sistema
 - Descripción del equipo

- Terminal receptor de imágenes-transmisor de señal
 - (a) Captura
 - (b) Digitalización
 - (c) Compresión
 - (d) Codificación
 - (e) Modulación
 - (f) Amplificación de potencia y transmisión.
- Terminal receptor de señal-expositor de imágenes
 - (a) Recepción
 - (b) Amplificación
 - (c) Demodulación
 - (d) Decodificación
 - (e) Descompresión
 - (f) Presentación
- Observaciones sobre tecnologías actuales

----- o -----

INTRODUCCION

Este trabajo se lo desarrolla para obtener una visión general de las varias técnicas de compresión de imágenes utilizadas en la actualidad y para realizar un análisis del fundamento científico de estas técnicas.

Se describe el problema del procesamiento digital de las imágenes cuando estas deben ser grabadas en medios tradicionales como cintas magnéticas, o memorias de semiconductor; o en el caso de ser transmitidas, a través de su radiodifusión televisiva. Además se realiza un estudio exhaustivo del uso de los distintos métodos de compresión de imágenes, así como de los estándares adoptados, mediante su presentación, descripción, estudio y análisis. Se desarrolla un estudio de un sistema real de transmisión-recepción digital de imágenes, terminal a terminal, señalando las partes más importantes, las implicaciones y restricciones que se hallarán en el trayecto, así como la participación de la compresión y descompresión en los respectivos terminales del sistema.

Por último se desarrollan tres algoritmos de compresión de una imagen de prueba, con los métodos:

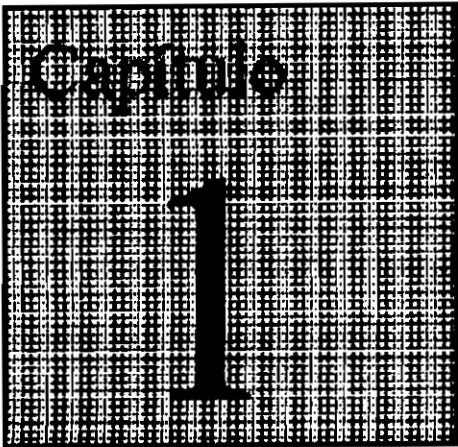
- Transformada Discreta Coseno.
- Truncamiento de Bloque.
- MIC Diferencial.

Y en base de los resultados, se realiza un estudio comparativo de estos métodos tanto cualitativa como cuantitativamente. Se presenta además dentro de un anexo un ejemplo de prototipo que realiza compresión y descompresión de señales de video convencional, indicando la circuitería necesaria y la denominación comercial de sus componentes.

Toda una introducción al proceso de digitalización de imágenes se encuentra en el Capítulo I. Los métodos de compresión se presentan en el Capítulo II. Las aplicaciones actuales de la compresión de imágenes así como sus estándares

se hallan en el Capítulo III. El estudio de un sistema de compresión y descompresión se lo encuentra en el Capítulo IV y en el Anexo 6. Finalmente el Capítulo V describe el desempeño de tres técnicas de compresión (TDC, CTB, y MICD) al ser aplicadas sobre una imagen de prueba.

Este trabajo cubre las diferentes técnicas de compresión que se ha investigado hasta la actualidad, exceptuando a técnicas demasiado avanzadas como la *Compresión por Redes Neuronales*, *Compresión Orientada a Objetos*, y así por el estilo, que merecen una investigación más amplia.



Conceptos Generales

CAPITULO I

Conceptos Generales

1.1 INTRODUCCION A LA DIGITALIZACION DE IMAGENES

1.1.1 ANTECEDENTES:

Debido a que la tendencia de la tecnología desde hace algunos años, se ha dirigido hacia la digitalización de todo tipo de señales, la digitalización de señales analógicas que representan a una imagen ya no es más que una de tantas aplicaciones, tal como lo demuestra el apareamiento del teletipo (telex), en los años cincuenta, o del facsímil a mediados de los setenta, cuya tendencia actual es hacia su extinción (reemplazado con las modernas tarjetas de FAX-MODEM).

Ya en los antiguos sistemas de telex, la imagen de una página de texto, por ejemplo, se la digitalizaba a través de un código de "marca" y "espacio", es decir ya aparecía la "bit-ificación" de los códigos. Los equipos utilizados eran sumamente lentos, pues transmitían los "bits", a razón de 50bps (bits por segundo), velocidad que con el tiempo hizo aparecer al telex, como un medio obsoleto de transmisión de datos. La generación siguiente al telex sería la del teletext, versión más rápida del original telex, con este se podía alcanzar velocidades de transmisión de hasta 2400bps.

El facsímil era la novedad de su época, pues no solo digitalizaba texto, sino que la imagen presentada a este aparato, podía ser de cualquier tipo, como un dibujo o una fotografía. El fax (como ahora se lo llama), resultó ser más eficiente respecto al telex, pues podía transmitir los bits de la imagen digitalizada a mayores velocidades, que la tradicional de 2400bps, alcanzando transmisiones de hasta 19200bps en casos especiales, pudiendo digitalizar una página de revista en 10 segundos.

La tendencia siguiente en la tecnología de digitalización de señales, es hacia la cuantificación digital de imágenes no solo bidimensionales, sino de naturaleza multi-dimensional, tanto en la captación de modelo y de sus características visuales como de sus colores; por ejemplo la digi-

talización de una imagen captada por un cámara de video a color, o de la captación tridimensional de la orografía de un terreno, en el cual es muy importante conocer profundidades, cimas, picos, así como también aunque en segundo plano, su cromatismo; es decir se trata de emular el proceso que sigue el sentido de la VISION para reconocer objetos (imágenes).

Los problemas en el tratamiento del video son complejos. Considérese una secuencia de imágenes de media resolución (512x512 pixeles) y una velocidad de trama estándar (30 cuadros por segundo) de color (3 bytes por pixel). Esto representa una velocidad de casi 24 millones de bytes de datos por segundo. Un simple algoritmo de extracción de características puede requerir de miles de operaciones básicas por pixel, y un típico sistema de visión, requeriría de más complejos cálculos. En estos casos, el problema se resuelve con el llamado "Procesamiento Paralelo", algoritmo de cálculo que se utiliza en el procesamiento y reconocimiento de imágenes en Cibernética.

La mayoría de algoritmos de procesamiento de imágenes recurren a técnicas más complejas de digitalización que los hasta ahora indicados en estos párrafos. Este hecho ha inspirado el desarrollo de procesadores matriciales, lográndose no solo procesadores de imágenes bidimensionales (norte, sur, este y oeste), sino que la arquitectura del procesamiento va ya más allá hasta utilizarla en levantamientos geofísicos de terrenos, o en posicionadores globales.

Dentro del proceso de imágenes, se pueden distinguir tres niveles de procesamiento: Bajo, Intermedio y Alto, diferenciándose uno de otro, por medio del nivel de complejidad de los algoritmos utilizados, la depuración de las señales obtenidas, compresión de su información, etc.¹

(1) G. Weems et al., "The DARPA Image Understanding Benchmark for Parallel Computers," J. Parallel and Distributed Computing, Jan. 1991, pp. 1-24.

1.1.2 CAPTURA DE DATOS Y PREPROCESO:

El barrido óptico (scanning) es el método usual de captura de datos de un documento. Los datos adquiridos por este método se almacenan en un archivo de elementos de imagen, llamados "píxeles" (pixel = picture element) a manera de un arreglo matricial. Estos píxeles tienen valores: 0 o 1 para imágenes binarias, 0 a 255 para imágenes con escala de grises, y tres canales de 0 a 255 valores de color para imágenes a color.

Es importante comprender que la imagen de un documento contiene los datos "crudos" para luego ser analizados y seleccionar así una imagen. Por ejemplo una letra "a" del tamaño de una décima de pulgada, será capturada como 1 o 0, por una matriz de 30x30 píxeles.

El preproceso de video incluye la binarización, reducción de ruido, y amplificación de la señal. Para imágenes que se componen de varias tonalidades de gris, con información inherentemente binaria, tal como un texto o un gráfico, la binarización tiene primera prioridad.

Un "scanning" bastante conocido es el que realiza una cámara de video al interpretar electrónicamente las diferentes tonalidades de alguna imagen que se le ponga frente a su sensor. En la Figura 1.1 se indican los componentes principales de una cámara de video, que son el sistema óptico, el dispositivo sensor de imagen y los circuitos electrónicos. Por medio de las lentes del sistema óptico, se enfoca la imagen sobre el sensor de imagen, el cual puede ser un tubo de vacío (vidicón o plumbicón), o un semiconductor como el dispositivo de carga acoplada (CCD). Este dispositivo produce una señal de voltaje que es proporcional a la intensidad de la luz emitida por los elementos de imagen. Luego esta señal de voltaje es procesada por ciertos circuitos electrónicos, produciéndose la señal de video.

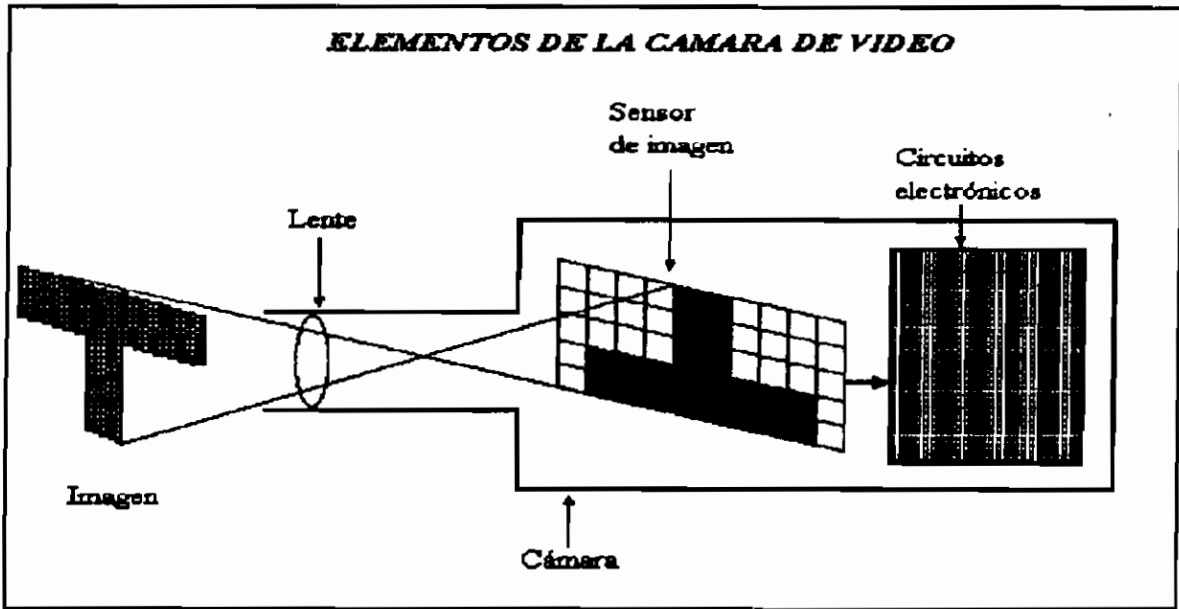


Figura 1.1 Elementos de la cámara de video.

El dispositivo sensor de imagen está compuesto de una gran cantidad de fotosensores, como se muestra en la Figura 1.2, donde cada cuadrado representa un fotosensor. En la práctica, el dispositivo sensor de imagen contiene cientos de miles de estos fotosensores. Para entender como se produce la señal eléctrica de voltaje, hay que imaginarse que a este arreglo de pequeños fotosensores se le agrega un gran switch. Al conmutar este switch, se tendrá una secuencia de muestras de voltaje, las cuales forman la señal eléctrica.

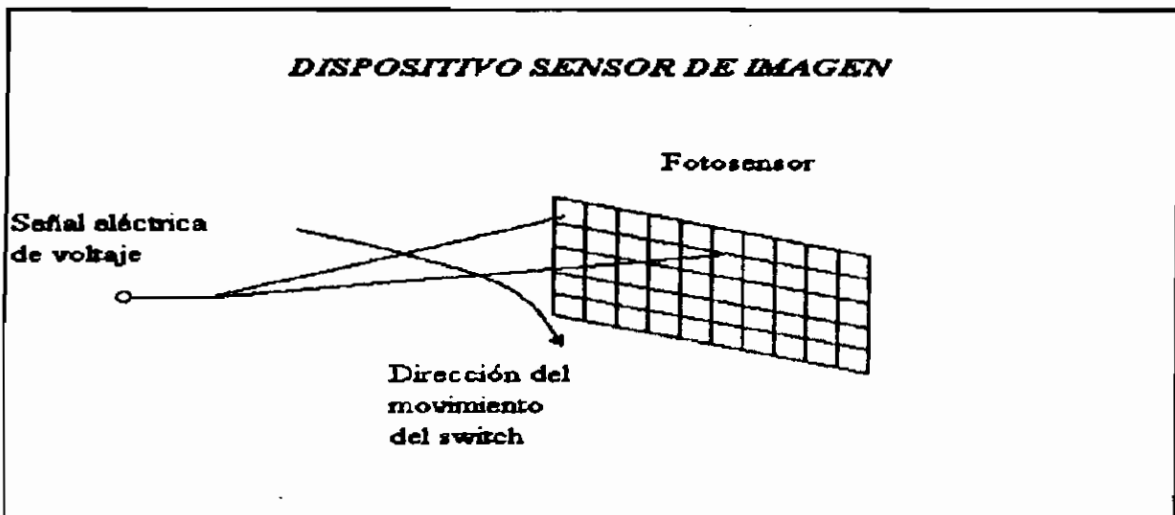


Figura 1.2 Dispositivo sensor de imagen.

Este switch se desplaza por una línea de izquierda a derecha. Cuando se acabe la línea, se pasa a la siguiente con el mismo desplazamiento de izquierda a derecha. Así sucesivamente se continúa hasta la última línea; cuando esto sucede, se regresa el switch a la parte superior izquierda, repitiéndose el proceso antes descrito.

En la Figura 1.3 se observa como mediante el uso de una lente, se forma la imagen (letra T) sobre el dispositivo sensor de la imagen; el fondo de la letra es blanco, y la lente invierte la imagen. En la Figura 1.4 se muestra la forma de onda de la señal Vout. En los fotosensores donde incidió mucha luz (zonas blancas) el nivel de voltaje es alto y en los puntos de poca luz (zonas negras), el nivel de voltaje es bajo. Con un ejemplo sencillo, se puede apreciar la relación que existe entre la imagen y los componentes de frecuencia de la señal eléctrica de voltaje que esta genera. En la Figura 1.5 se tienen tres imágenes compuestas de franjas verticales blancas y oscuras.

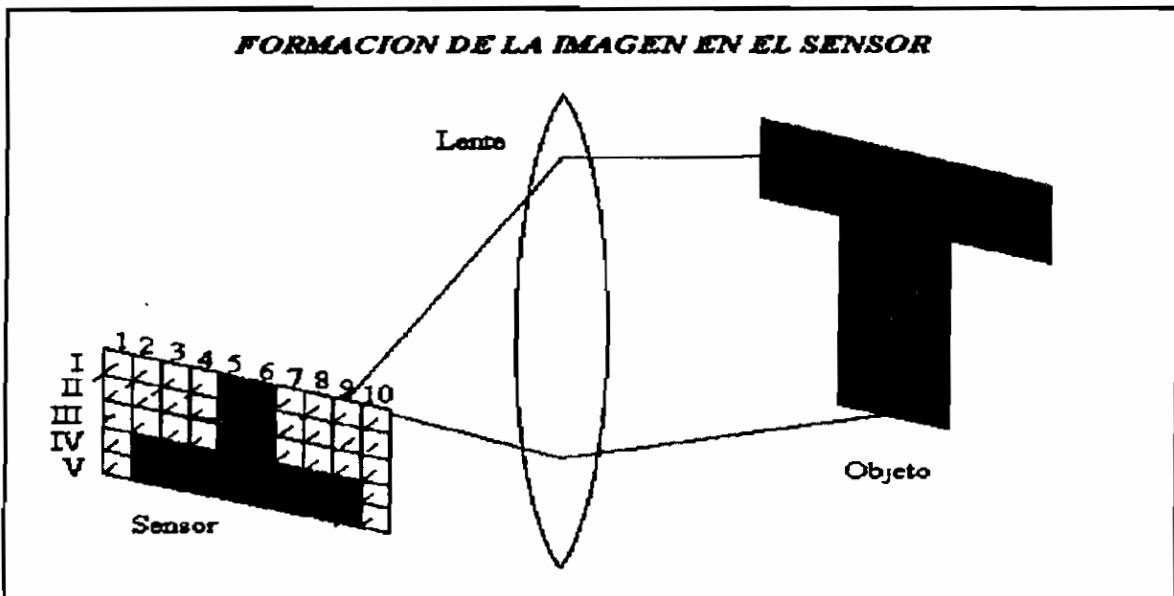


Figura 1.3 Formación de la imagen en el sensor.

La figura (a) corresponde a la imagen con las franjas más anchas, esta figura produce la señal de voltaje que se muestra en la parte inferior de la imagen, la cual completa un ciclo en cada línea.

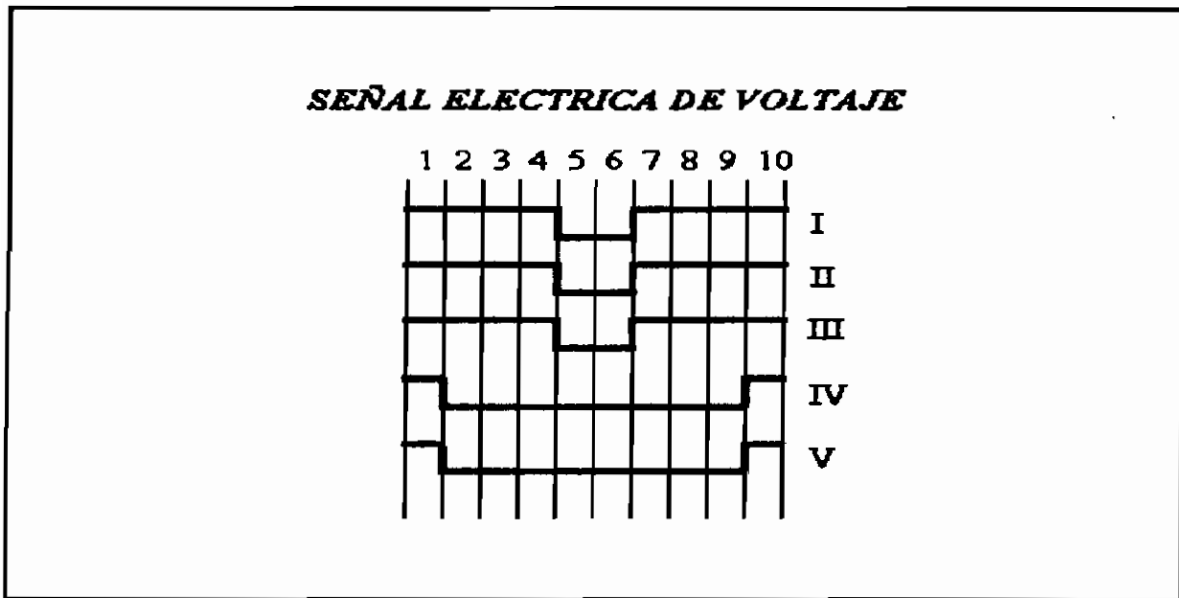


Figura 1.4 Señal eléctrica de voltaje.

La figura (b) muestra una imagen con el doble de líneas que la anterior, nótese en la señal de voltaje tiene el doble de frecuencia que la señal anterior. Por último se tiene la figura (c) con las franjas más delgadas, la cual genera una señal que tiene una frecuencia más alta.

Si se sigue disminuyendo el ancho de las franjas, en la misma proporción se iría incrementando la frecuencia de la señal eléctrica que representa una línea de esa imagen. A partir de este ejemplo, se puede hacer una generalización: *toda imagen compuesta por elementos de área pequeña, como el cabello, el follaje en un paisaje, las tramas de la ropa, los bordes, etc., hace que el sensor de imagen genere una señal con componentes de alta frecuencia. Y una imagen que contenga elementos de área grande, como el cielo azul, el mar, etc, conlleva a que el sensor de imagen genere una señal con componentes de baja frecuencia.*

La digitalización de imágenes también se aplica al tratamiento de las imágenes de televisión. El problema de los sistemas de televisión digital es la enorme cantidad de bits de información involucrada en este tratamiento.

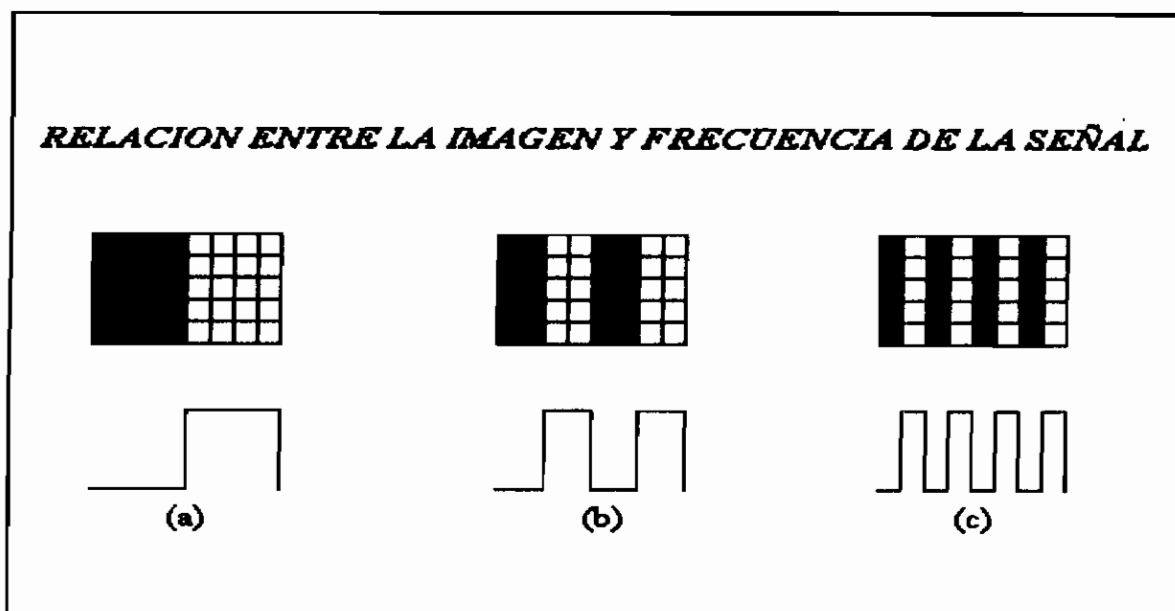


Figura 1.5 Relación entre la imagen y la frecuencia.

-Una sola imagen de HDTV (Televisión de Alta Definición) es equivalente a casi 1000 millones de bits de datos digitales y esta cantidad es virtualmente imposible de transmitir en forma normal-

Aparece entonces la necesidad de llevar a cabo un *proceso de compresión* que reduce grandemente la cantidad de bits por pixel digitalizado. Además, en lugar de transmitir palabras binarias formadas por unos y ceros, se recurre a métodos más complejos que abarcan una cantidad uniforme de números distintos a los binarios 1 y 0. Estos tipos de codificación usados ampliamente en sistemas informáticos y son más eficientes en cuanto a cantidad de dígitos a transmitir, por este motivo se requiere de un ancho de banda sustancialmente menor, (pues como se verá el número de datos digitales transmitidos por segundo tiene una relación directa con el ancho de banda utilizado para transmitirlos). Un diagrama esquemático del proceso de digitalización de imágenes se muestra en la Figura 1.6. Como se puede apreciar, la etapa inicial es la captura de la imagen, por algún medio transductor que cambia la imagen análoga visual, a imagen análoga eléctrica (por ejemplo una escena captada por una cámara). Después de esta etapa, se procede a realizar una conversión

de analógico a digital. Este paso es el más crítico de la digitalización. Una vez obtenido el tren de pulsos que representan a la señal digitalizada, el siguiente paso corresponde al proceso particular para cada aplicación, por ejemplo, este bloque correspondería a un almacenamiento de la señal digital, a su transmisión, a su multiplexaje con otras señales, etc.

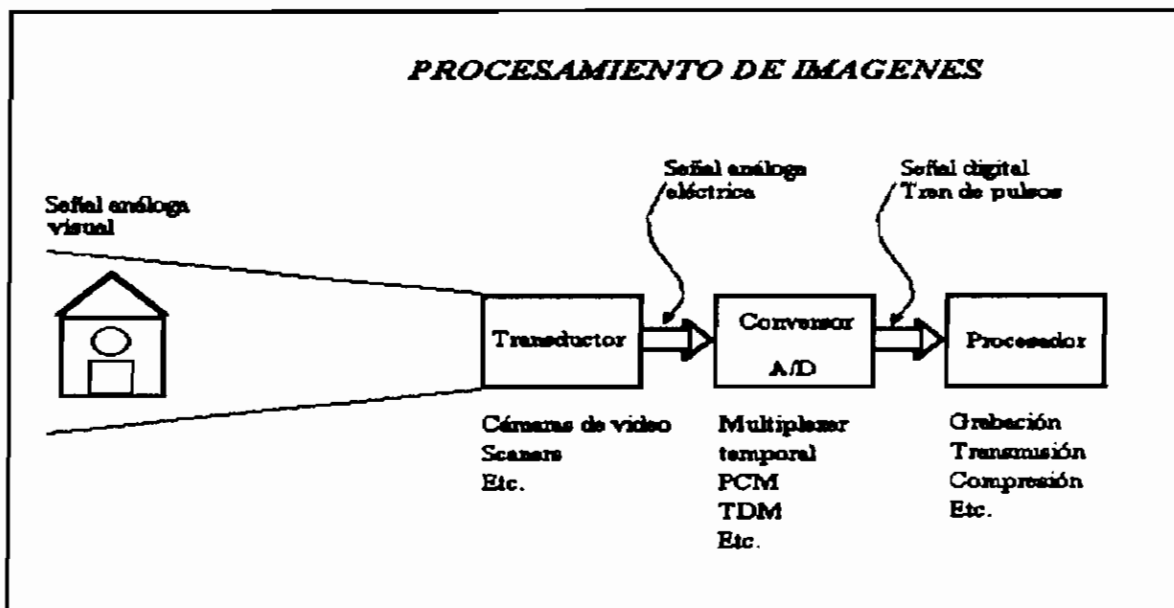


Figura 1.6 Procesamiento de imágenes.

Para este estudio en particular, luego de la captación de la imagen, el proceso se centrará solamente en dos aspectos fundamentales:

- Digitalización de la señal de video que representa la imagen; y,
- Compresión del código digital que representa a dicha imagen.

Con el fin de capturar a la imagen, se pueden emplear métodos específicos como el uso de scaners, o cámaras de video, teniendo estos métodos un propósito en común: Tomar la imagen visual, retenerla por algún medio, para obtener así una interpretación eléctrica de esta señal de video, en virtud de que a través de medios eléctricos la tarea de análisis es más objetiva y práctica para cualquier tipo de señal (acústica, visual, sonora, etc.).

1.1.3 DIGITALIZACION:

Debido a su interesante naturaleza y a la variedad de campos de aplicación, el proceso de la digitalización de imágenes ha sido objeto de intensos estudios en los últimos años. Estos estudios han producido muchos tipos de digitalizadores de imágenes con numerosas variaciones de cada tipo. El optar por uno u otro depende de la aplicación y el nivel de calidad deseada.

Las técnicas de digitalización de imágenes pueden ser categorizadas dos clases: las que codifican digitalmente formas de onda análogas tan fielmente como sea posible, y las que procesan las formas de onda para codificar solamente aspectos significantes de la imagen.

Las técnicas más comunes usadas para codificar una forma de onda de imagen son la modulación de impulsos codificados (MIC en todas sus versiones como la diferencial o adaptiva) y la modulación delta (MD igualmente en todas sus versiones), con una característica en común: Estas técnicas utilizan el recurso de la "predicción" de bits, tema que se tratará ampliamente en el capítulo referido a los métodos de compresión de imágenes.

1.1.4 MODULACION POR AMPLITUD DE PULSOS (MAP):

El primer paso en la digitalización de una forma de onda analógica es establecer un conjunto de tiempos discretos en los cuales la señal analógica de entrada es *muestreada*. Técnicas de digitalización muy conocidas se basan en el uso de intervalos de muestreo periódicos y regularmente espaciados. Si las muestras ocurren muy a menudo (alta densidad de muestreo), la forma de onda original puede ser completamente recuperada de la secuencia de muestreo usando un filtro pasa bajos para interpolar o seguir los valores muestreados. Estos conceptos básicos se ilustran en la Figura 1.7.

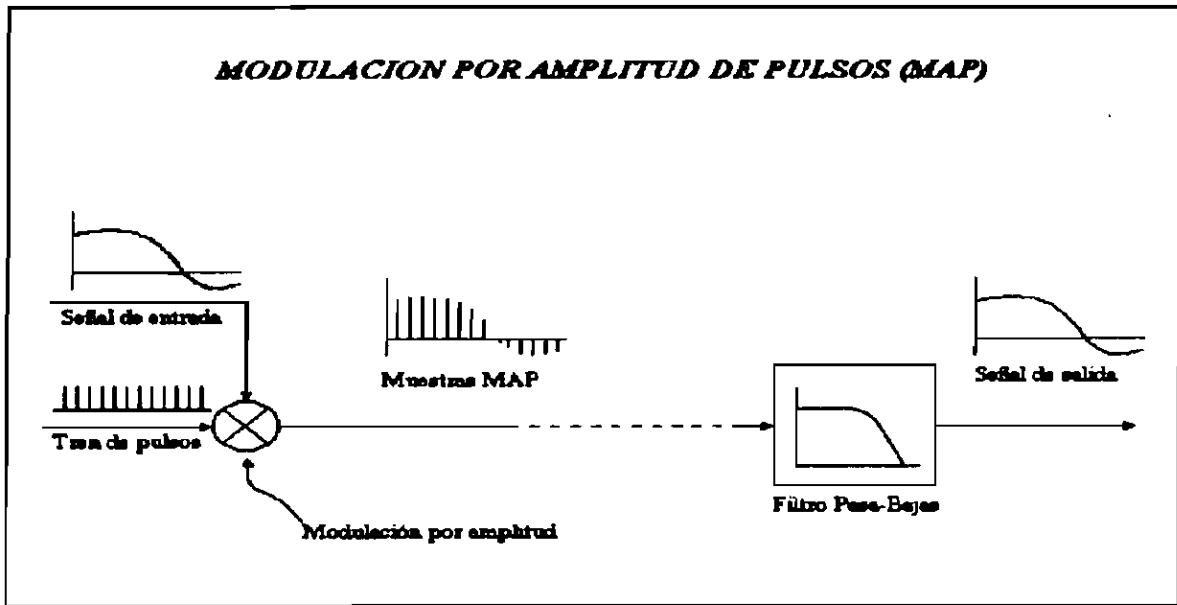


Figura 1.7 Modulación por amplitud de pulsos.

Una onda analógica es muestreada a una frecuencia constante $f_s = 1/T$ y reconstruida usando un filtro pasa bajos, nótese que el proceso de muestreo es equivalente a la modulación por amplitud de un tren de pulsos de amplitud constante. Por esta razón esta técnica es usualmente conocida como modulación por amplitud de pulsos (MAP).

1.1.5 VELOCIDAD DE MUESTREO DE NYQUIST:

Un resultado clásico en sistemas de muestreo fue establecido en 1933 por Harry Nyquist cuando dedujo la mínima frecuencia de muestreo requerida para extraer toda la información en una forma de onda continua variable en el tiempo. Este resultado-*el criterio de Nyquist*-se define por la relación:[29]

$$f_s > 2 \cdot BW \quad (1.1)$$

donde, f_s = frecuencia de muestreo.

BW = ancho de banda de la señal de entrada.

Este resultado se infiere de la Figura 1.8, que describe el espectro de la señal de entrada y al espectro resultante del tren de pulsos MAP.

1.1.6 MODULACION POR IMPULSOS CODIFICADOS (MIC):

La sección precedente describe la modulación por amplitud de pulsos, la cual usa intervalos de muestreo simples con amplitudes de muestras analógicas para extraer la información en una señal análoga variable continuamente. La modulación por impulsos codificados (MIC) es una extensión de MAP en donde cada valor muestreado es cuantificado a un valor discreto para su representación digital en una palabra código, como se muestra en la Figura 1.8. Un sistema MAP puede ser convertido a un sistema MIC si se le aumenta un convertidor de análogo a digital en la fuente, y un convertidor de digital a análogo en el destino.

La Figura 1.9 describe un proceso de cuantificación típico en el que un grupo de intervalos de cuantificación se asocian en una relación de uno a uno con una palabra código binaria. Todos los valores muestreados cayendo en un intervalo de cuantificación particular, son representados por un valor discreto localizado en el centro del intervalo de cuantificación.

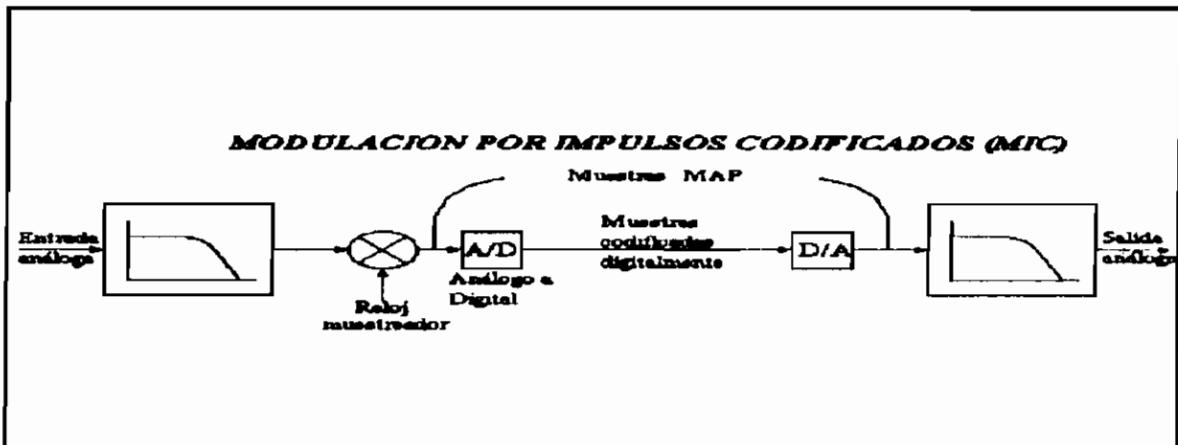


Figura 1.8 Proceso MIC.

De esta manera el proceso de cuantificación introduce una cierta cantidad de error o distorsión en las señales muestreadas. Este error, conocido como "Ruido de Cuantificación", es minimizado al establecer un gran número de pequeños intervalos de cuantificación.

Por supuesto, que si el número de intervalos de cuantificación se incrementa, el número de bits se incrementa también.

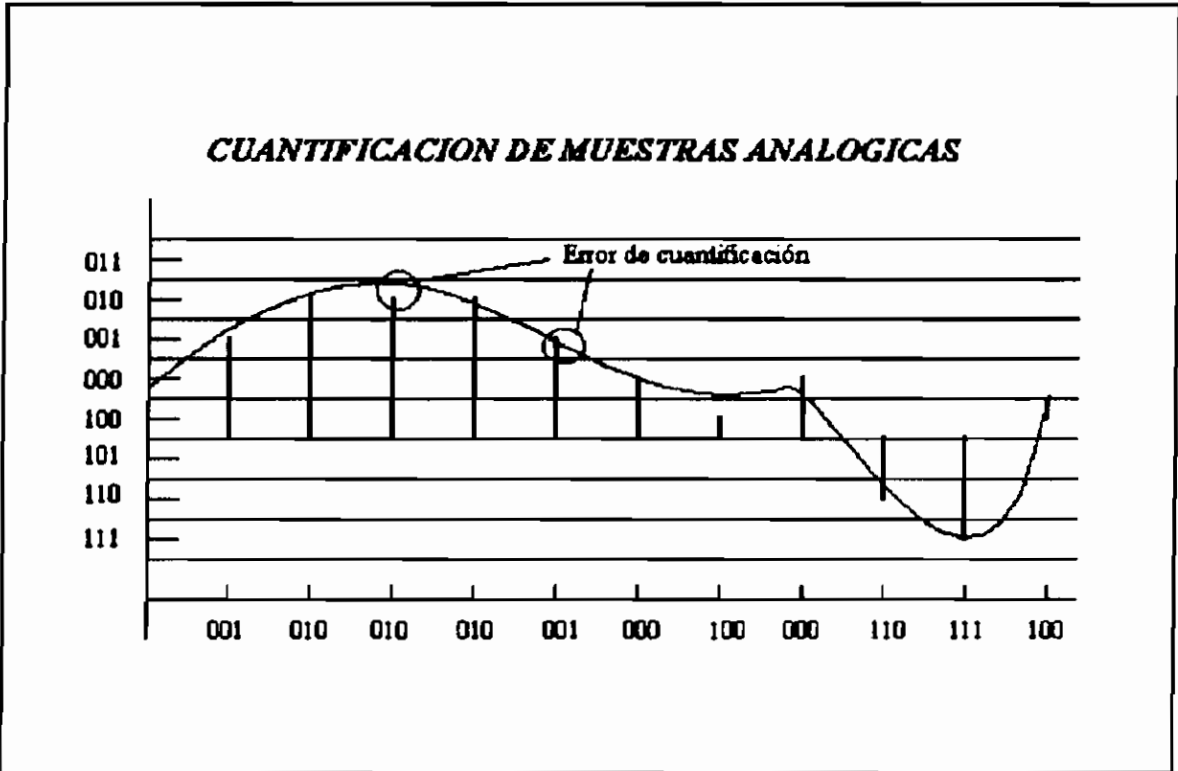


Figura 1.9 Cuantificación de muestras analógicas.

El error de cuantificación o la distorsión creada por la digitalización de una señal analógica es expresada funcionalmente como una potencia promedio de ruido relativo a la potencia promedio de la señal. Así la razón de señal a ruido de cuantificación SNR ("Signal-Noise Ratio" también llamada una relación de señal a distorsión o razón señal a ruido), puede ser determinada por: [29]

$$SNR = \frac{E\{x^2(t)\}}{E\{[y(t) - x(t)]^2\}} \quad (1.2)$$

donde, $E\{\cdot\}$ = valor esperado.

$x(t)$ = la señal analógica de entrada.

$y(t)$ = la señal de salida DECODIFICADA.

Al determinar el valor esperado del ruido de cuantificación, son necesarias tres observaciones:

- 1) El error $y(t)-x(t)$ está limitado en amplitud para $q/2$ donde q es la magnitud del intervalo de cuantificación (Las muestras decodificadas son idealmente posicionadas en la mitad del intervalo de cuantificación).
- 2) Un valor de muestra tiene una probabilidad de caer en cualquier punto del intervalo de cuantificación, dicha densidad de probabilidad es de $1/q$.
- 3) Las amplitudes de las señales-se asumen-confinadas al rango máximo del codificador. Si un valor de muestra excede el rango del intervalo de mayor cuantificación, se presenta la llamada *distorsión de sobrecarga*.

Asumiendo (por conveniencia) un nivel de resistencia de 1Ω , la potencia de ruido de cuantificación queda determinada por:[29]

$$\text{Potencia de ruido de cuantificación} = \frac{q^2}{12} \quad (1.3)$$

Si todos los intervalos de cuantificación son de igual longitud (*cuantificación uniforme*), el ruido de cuantificación es independiente de los valores muestreados y la relación señal a ruido es:[29]

$$SNR(dB) = 10\log_{10}\left(\frac{v^2}{q^2/12}\right) \quad (1.4)$$

donde v es la amplitud RMS de la señal de entrada.

1.1.7 MIC CODIFICADO UNIFORMEMENTE:

Un codificador que utiliza iguales intervalos de cuantificación para todas las muestras, produce una palabra código linealmente relacionada con el valor de la muestra análoga. Esto es, el equivalente numérico de cada palabra código es proporcional al valor de la muestra cuantificada que representa. En esta forma, un sistema MIC uniforme usa un

convertidor convencional de análogo a digital para generar los códigos binarios de la muestra. El número de bits requerido para cada muestra es determinado por la máxima potencia de ruido aceptable. El desempeño de un sistema de MIC uniforme se determina según la relación:[29]

$$q = \frac{2A_{\max}}{2^n} \quad (1.5)$$

donde, A_{\max} es la amplitud (no sobrepasable) máxima.
 n es el número de bits por muestra codificada.

1.1.8 MIC CODIFICADO NO UNIFORMEMENTE:

En un sistema MIC uniforme, el tamaño de cada intervalo de cuantificación se determina por los requerimientos del SNR del nivel de la señal más pequeña a ser codificada. Para señales más grandes, estas son codificadas con el mismo intervalo de cuantificación. De esta manera un sistema MIC uniforme provee calidad innecesaria para grandes señales. Más aún las señales grandes son muy raras de ocurrir. Por estas razones el espacio código en un sistema MIC uniforme es ineficientemente utilizado.

Un proceso de codificación más eficiente se alcanza si los intervalos no son uniformes pero permiten incrementarse con el valor de la muestra. Cuando los intervalos de cuantificación son directamente proporcionales al valor de la muestra, el SNR es constante para todos los niveles de las señales. Con esta técnica, muy pocos bits por muestra dan un específico SNR para pequeñas señales y un adecuado rango dinámico para las señales grandes. Cuando los intervalos de cuantificación no son uniformes, una relación no lineal existe entre las palabras código y las muestras que representan. Históricamente, la función no lineal que se implementó en un principio en señales análogas fue usando dispositivos especiales como diodos.

El proceso básico se muestra en la Figura 1.10 donde la muestra análoga de entrada es primero comprimida y luego cuantificada con intervalos de cuantificación uniforme. El efecto de la operación de compresión, se muestra en la Figura 1.11.

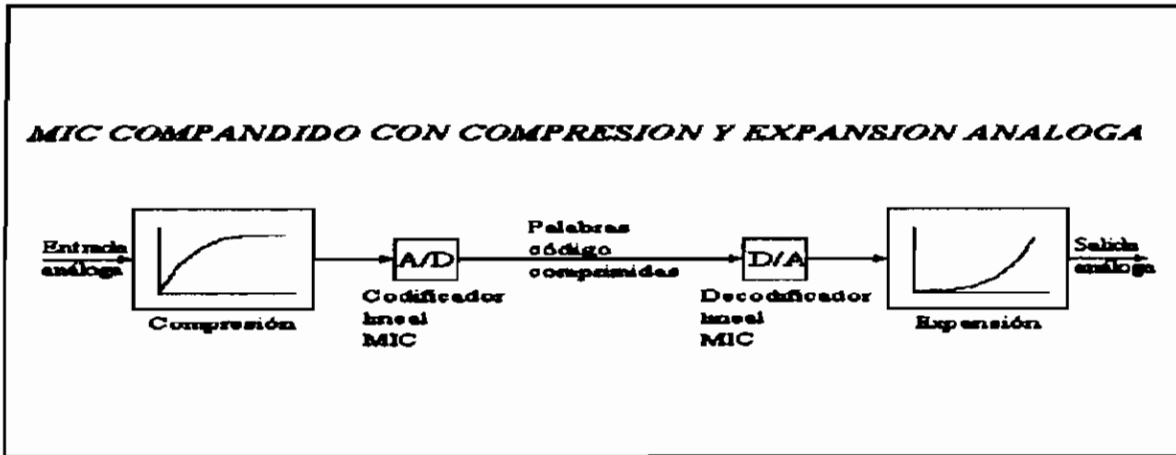


Figura 1.10 Compansión en MIC.

Nótese que sucesivamente los intervalos de la señal de entrada grande son comprimidos dentro de intervalos de cuantificación de longitud constante. Así la muestra más grande es comprimida antes de la codificación. Como se ve en la Figura 1.10, un decodificador no uniforme MIC expande el valor comprimido usando una característica de compresión inversa para recobrar el valor de la muestra original.

-El proceso de primero comprimir y luego expandir una señal se conoce como COMPANSION-.

Cuando se digitaliza, la compansión asigna intervalos de cuantización pequeños a muestras pequeñas y grandes intervalos de cuantización a señales grandes.

1.1.8.1 Compansión de la Ley μ :

Varias características de compresión-expansión pueden ser escogidas para implementar un compansor. Al incrementar la cantidad de compresión, se incrementa el rango dinámico a expensas de la relación señal a ruido para señales de gran amplitud.

Una familia de características de compresión usadas en Norte América y Japón es la ley μ cuya característica está definida por:[29]

$$F_{\mu}(x) = \text{sgn}(x) \frac{\ln(1+\mu|x|)}{\ln(1+\mu)} \quad (1.6)$$

donde, x es la amplitud de la señal de entrada ($-1 \leq x \leq 1$), $\text{sgn}(x)$ es la polaridad de x , y μ es un parámetro usado para definir la cantidad de compresión.

Debido a la naturaleza matemática de la curva de compresión, el MIC compandido es a veces conocido como un MIC logarítmico. Una curva de compresión logarítmica es ideal en el sentido de que los intervalos de cuantificación y así el ruido de cuantificación es proporcional a la amplitud de la muestra. La característica inversa o de expansión de la ley μ se define por:[29]

$$F_{\mu}^{-1}(y) = \text{sgn}(y) \left(\frac{1}{\mu}\right) [(1+\mu)^{|y|}-1] \quad (1.7)$$

donde, y es el valor comprimido = $F_{\mu}(x)$ ($-1 \leq x \leq 1$), $\text{sgn}(y)$ es la polaridad de y , y μ es el parámetro de compansión, cuyo valor es (Recomendación 732 del CCITT):

$$\mu = 255$$

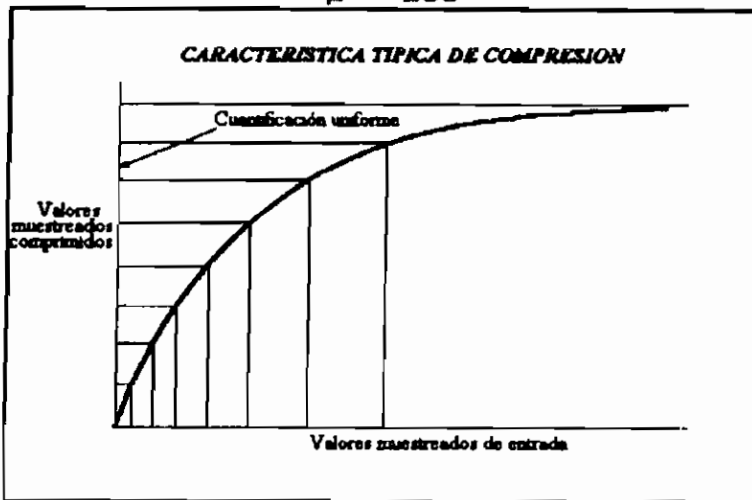


Figura 1.11 Curva de compresión.

1.1.8.2 Compansión de la ley A:

La característica de compansión recomendada por el CCITT (Comité Consultivo Internacional de Telefonía y Telegrafía) se refiere a la ley A. Esta característica tiene las mismas particularidades básicas y ventajas implementadas en la característica de la ley μ . En particular, la característica de la ley A puede ser aproximada por segmentos de líneas continuas para facilitar una compansión directa o digital y puede ser fácilmente convertida a un formato lineal. La característica de compresión de la ley A normalizada está definida por:[29]

$$F_A = \operatorname{sgn}(x) \left(\frac{A|x|}{1+\ln(A)} \right) \quad 0 \leq |x| \leq \frac{1}{A} \quad (1.8)$$

$$F_A = \operatorname{sgn}(x) \left(\frac{1+\ln(A|x|)}{1+\ln(A)} \right) \quad \frac{1}{A} \leq |x| \leq 1 \quad (1.9)$$

La característica inversa o de expansión es:[29]

$$F_A^{-1} = \operatorname{sgn}(y) \frac{|y|[1+\ln A]}{A}; \quad 0 \leq |y| \leq \frac{1}{1+\ln(A)} \quad (1.10)$$

$$F_A^{-1} = \operatorname{sgn}(y) \frac{(e^{|y|[1+\ln A]} - 1)}{A}; \quad \frac{1}{1+\ln(A)} \leq |y| \leq 1 \quad (1.11)$$

Donde,

$$y = F_A(x);$$

y (Recomendación 732 del CCITT),

$$A = 87.6$$

Nótese que la primera porción de la característica de la ley A mostrada en la Figura 1.12 es lineal por definición. Así mismo, existen ocho segmentos positivos y ocho segmentos negativos, los primeros pares de segmentos de cada polaridad (cuatro en total) son colineales y por lo tanto a veces son considerados como un solo segmento de línea continua. Así la aproximación segmentada de la característica de la ley A es a veces referida como una aproximación de "13 segmentos" (Recomendación 732 del CCITT). Existen otros tipos de procesos digitalizadores de señales analógicas dentro de la técnica de codificación binaria o por pulsos, así se tienen:

- MIC diferencial(DPCM en inglés).
- Modulación Delta(DM).
- Modulación Delta lineal(LDM).
- Modulación Delta Adaptiva(ADM).

Todas ellas basadas en el mismo principio de la optimización del número de bits de palabra código que representa una muestra analógica, reduciendo el número de bits de 8 a 1!. Este tipo de codificación se la explicará más adelante cuando se profundice en materia de compresión de imágenes.

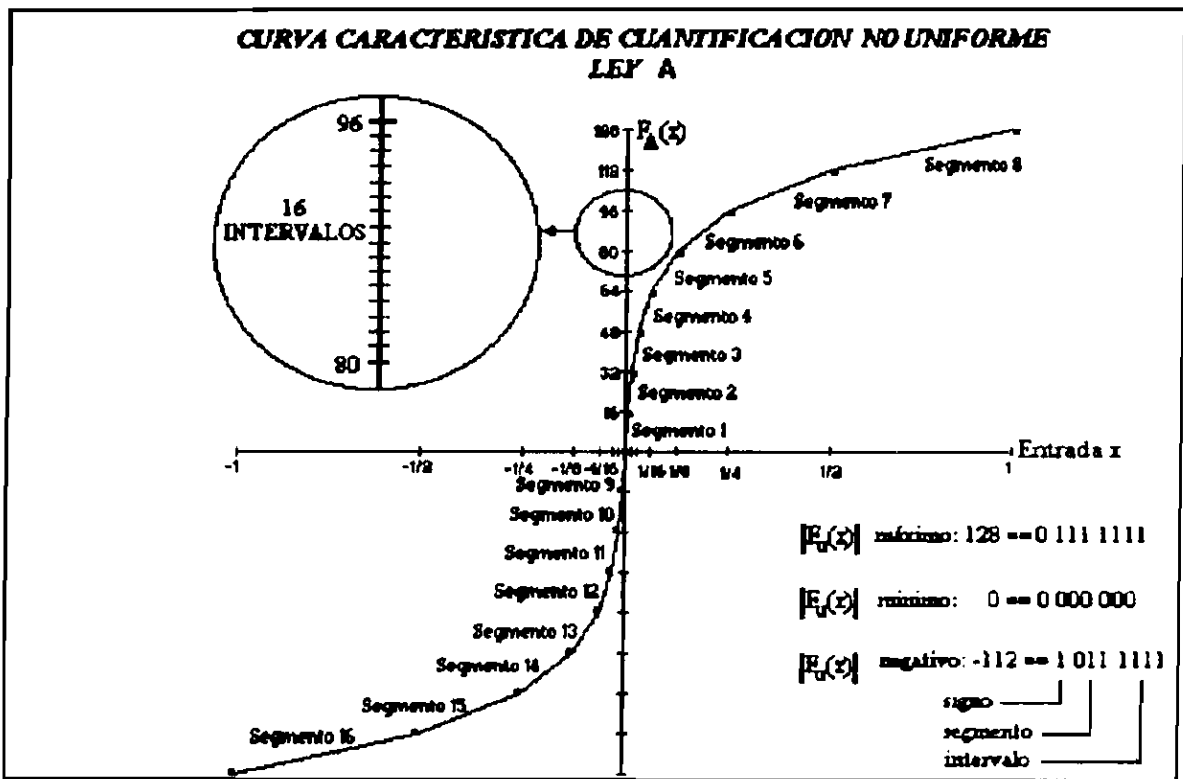


Figura 1.12 Curva de compansión- Ley A.

1.2 CRITERIOS DE TRANSMISION DIGITAL DE IMAGENES

Se requieren aún de grandes velocidades de transmisión, rangos que están entre las decenas y cientos de megabits por segundo para una transmisión de alta calidad. Por ejemplo 168Mbps ha sido propuesto por el CCIR (Comité Consultivo Internacional de Radio) como un estándar, para transmisión de este tipo de señales.

Una consideración fundamental en el diseño de sistemas de transmisión digital es la selección de un conjunto finito de formas de ondas eléctricas discretas para codificar la información. Según los conceptos de la teoría de comunicación digital, estas formas de onda se las conocen como señales. En la terminología de la teoría de comunicaciones, "el procesamiento de señales" se refiere al filtraje, formación y transformación de formas de onda eléctricas.

1.2.1 TRANSMISION DE PULSOS:

Todos los sistemas de transmisión digital son diseñados sobre una forma particular de respuesta de pulso cuadrado como el mostrado en la Figura 1.13.

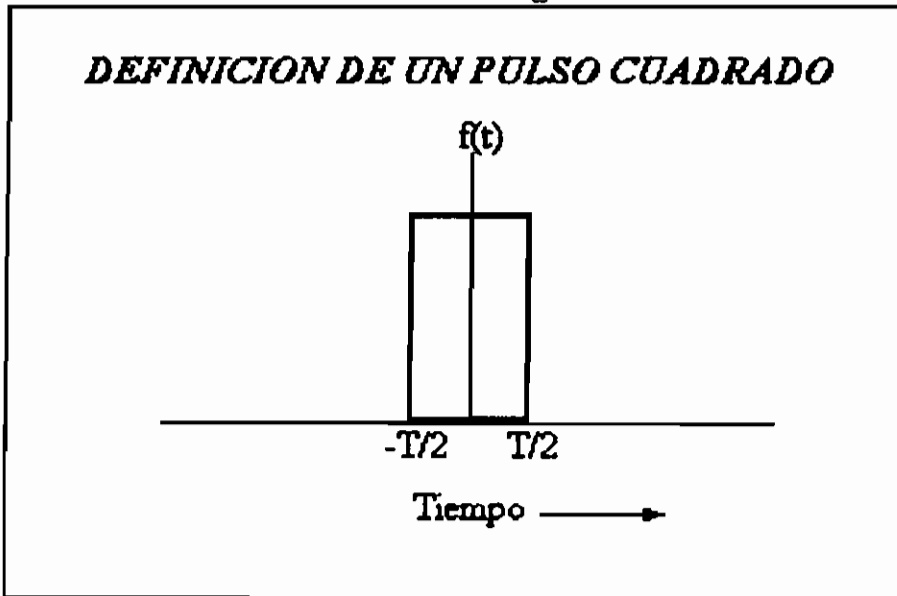


Figura 1.13 Pulso cuadrado.

El espectro de frecuencia correspondiente al pulso cuadrado se puede deducir por medio de la aplicación de la transformada de Fourier, obteniéndose:[29]

$$F(\omega) = (T) \frac{\sin(\omega T/2)}{\omega T/2} \quad (1.12)$$

esto es, referido a una respuesta de la forma:[29]

$$\frac{\sin(x)}{x} \quad (1.13)$$

donde, $\omega = 2\pi f$ está en radianes por segundo, y T es la duración de un intervalo de una señal. Se muestra el espectro de frecuencia del pulso cuadrado en la Figura 1.14.

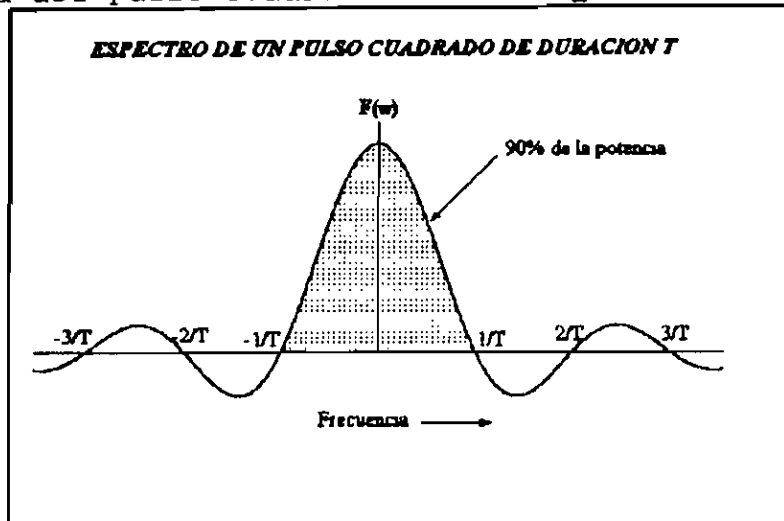


Figura 1.14 Espectro de un pulso cuadrado de duración T .

En la Figura 1.14 también se da el porcentaje de potencia del espectro. Como se indica, el 90% de la energía de la señal está contenida dentro del primer espectro que se anula en $f=1/T$. El gran porcentaje de energía dentro de esta banda indica que la señal puede ser confinada a un ancho de banda de $1/T$ y se obtendrá una buena aproximación a la forma de onda original. En teoría, si el valor de la muestra a la mitad de cada intervalo de señal, se mantiene, el ancho de banda puede ser confinado a $1/(2T)$. De este hecho la velocidad máxima de señal en un ancho de banda específico se determina por:[29]

$$R_{\max} = 2 \cdot BW \quad (1.14)$$

Donde R_{\max} es la velocidad de transmisión $=1/T$, y BW es el ancho de banda disponible. La ecuación anterior especifica un resultado fundamental de la teoría de las comunicaciones acreditado a Harry Nyquist: "*La máxima velocidad de muestreo alcanzable a través de un ancho de banda de un filtro pasa bajos, sin interferencia intersímbolo es igual a dos veces el ancho de banda.*"

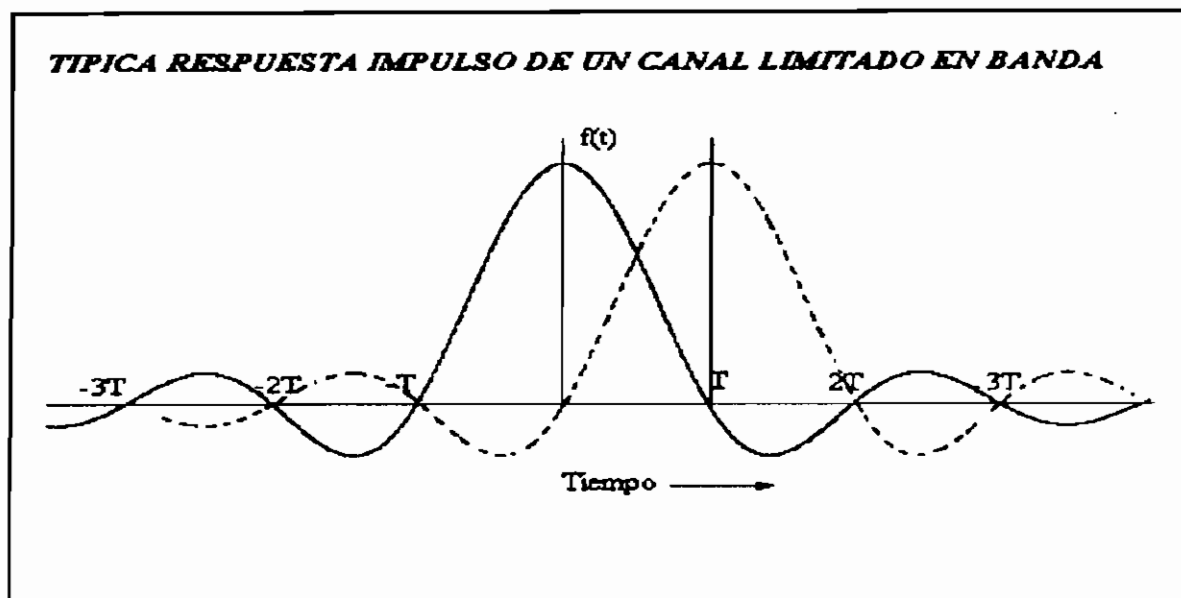


Figura 1.15 Respuesta impulso de un canal limitado en banda.

Esta velocidad R_{max} es a veces llamada como la "*velocidad de Nyquist*". Aunque discretos, los pulsos formados de onda cuadrada son fáciles de visualizar, la preservación de la forma cuadrada requiere de amplios anchos de banda y son por lo tanto indeseables. Una forma típica de un pulso sencillo se muestra en la Figura 1.15. El rizado a ambos lados de la parte principal del pulso (ondulaciones) es un acompañamiento necesario para un canal con una banda limitada. Normalmente en una transmisión digital los pulsos cuadrados (o modulados equivalentes de ellos), pero filtrados por banda y el propio medio de transmisión se combinan para producir una respuesta como la mostrada en la Figura 1.15.

Una importante característica de la respuesta del pulso mostrado en la Figura 1.18 es que si no se considera la ondulación, un pulso puede transmitirse una vez cada T segundos y ser detectado en el receptor sin interferencia de pulsos adyacentes. Obviamente, el tiempo de muestreo coincide con los cruces por cero de los pulsos adyacentes. La respuesta del pulso como la mostrada en la Figura 1.18 puede ser obtenida en canales con anchos de banda aproximados al mínimo (Nyquist), igual a la mitad de la velocidad de muestreo.

1.2.2 INTERFERENCIA INTERSÍMBOLO (IIS):

Las pequeñas perturbaciones en la respuesta del canal o en los tiempos de muestreo producen un solapamiento no nulo en los tiempos de muestreo llamado interferencia intersímbolo. Las principales causas para esta interferencias son:

- 1) Sincronización inexacta.
- 2) Ancho de banda insuficiente.
- 3) Distorsión de amplitud.
- 4) Distorsión de fase.

1.2.2.1 Sincronización inexacta:

Esta ocurre en el transmisor o en el receptor. En el transmisor, ocurre IIS si la rata de transmisión no se adapta a la frecuencia de las oscilaciones diseñadas en el canal. Mientras que la inexacta sincronización en el receptor se deriva del ruido y posiblemente distorsione las señales recibidas.

1.2.2.2 Ancho de banda insuficiente:

La frecuencia de las oscilaciones mostradas en la Figura 1.15 es exactamente igual al ancho de banda teórico mínimo del canal. Si se reduce el ancho de banda más aún, la frecuencia de la oscilación se reduce y esto resulta en un aumento de la IIS.

1.2.2.3 Distorsión de amplitud:

Cuando un medio de transmisión con determinadas características es utilizado, estas características pueden ser incluidas en el diseño general de un filtro. Una parte de la respuesta de frecuencia deseada causa distorsiones de pulso (reduce picos de amplitudes y crea frecuencias impropias de oscilación) en el dominio del tiempo. La compensación de irregularidades en la respuesta de frecuencia del canal se conoce con el nombre de equalización de amplitud.

1.2.2.4 Distorsión de fase:

Cuando se examina el dominio de la frecuencia, un pulso, está representado como la superposición de componentes de frecuencia con una determinada relación de amplitudes y fases. Si las amplitudes relativas de las componentes de frecuencia son alteradas, se produce, la distorsión de amplitud. Si las relaciones de fase de los componentes se altera, la distorsión de fase ocurre. Básicamente, las distorsiones de fase resultan cuando los componentes de frecuencia de una señal experimental difieren del retardo en el enlace de transmisión.

1.3 COMPARACION ENTRE MEDIOS DE ALMACENAMIENTO Y TRANSMISION

1.3.1 COMPARACION DE MEDIOS:

La transmisión, el almacenamiento, y el procesamiento de las señales digitales de video requieren el manejo de un flujo de datos digitales. La velocidad de ese flujo de datos es llamado la **velocidad binaria**, la cual mide la cantidad de datos en bits por segundo (bps). A efectos de comparación, en lo concerniente a transmisión, basta con analizar la fórmula obtenida por el criterio de Nyquist (Ecuación 1.14), que confirma la necesidad de reducir el ritmo de transmisión (bits por segundo) a fin de lograr anchos de banda de transmisión reducidos como los que requieren las leyes de radiodifusión.

Con respecto al almacenamiento de información, si se utilizan medios magnéticos para guardar información digital, como cintas o discos flexibles, se tropieza con limitantes físicas de estos medios, limitantes que son resultado del ancho de banda de las señales a almacenar. Por ejemplo al comparar una grabadora de audio, con una de video, la principal diferencia es el ancho de banda, porque la señal de audio contiene frecuencias que van desde los 20Hz hasta los 20KHz, mientras que la señal de video contiene frecuencias desde DC hasta 4.2MHz.

La relación entre las frecuencias más altas que contiene cada señal es:

$$F_{\max\text{-video}} = 4.2\text{MHz}$$

$$F_{\max\text{-audio}} = 20\text{KHz}$$

$$F_{\max\text{-video}} / F_{\max\text{-audio}} = 210$$

Así si se intenta grabar la señal de video en una grabadora de audio, solo se grabaría el sincronismo de vertical (60Hz) y el sincronismo de horizontal (15.75KHz). Para modificar la grabadora de audio y convertirla en una de video, se debe modificar lo siguiente:

- La velocidad de la cinta.
- El entrehierro (gap) de la cabeza.
- Los circuitos electrónicos para que trabajen a más alta frecuencia.
- La calidad de la cinta.
- Las dimensiones de la cinta.

Nótese que con este pequeño ejemplo de almacenamiento y comparación entre una grabadora de audio y una de video, se puede dar una idea del problema que se tiene al almacenar video digital y video analógico en una grabadora de video convencional, puesto que la relación entre el ancho de banda y la velocidad de transmisión es una relación directa. Al relacionar el ancho de banda de video convencional (4.2-MHz) y el ancho de banda de video digital (según Nyquist, $R_{\max}/2 = 168\text{Mbps}/2 = 84\text{MHz}$ ---CCIR 601), se tiene una relación, no de 210 como en el ejemplo anterior, sino de:

$$BW_{\text{dig}} / BW_{\text{conv}} = 84 / 4.2 \approx 20$$

que de todas formas es considerable, y más si se hiciese una comparación con el ancho de banda de la señal HDTV ($BW \approx 1\text{GHz}$).

Debido a que una imagen digitalizada es un flujo continuo de bits, estos se pueden agrupar en paquetes de 8bits llamados "bytes" teniéndose por lo tanto un flujo constante de bytes, que sin duda se los puede almacenar en otros medios no previstos para la grabación de imágenes, como son los discos magnéticos y las memorias de semiconductor.

Con el interés incrementado en la introducción de servicios digitales de alta velocidad para el consumidor, existe una necesidad de interfaces baratos entre redes digitales y analógicas para servicios de video. Serán menores los requerimientos para la calidad de imagen, que los de transmisión entre redes afiliadas de televisión. El requerimiento básico es tener una imagen visualmente no defectuosa luego de una razonable ruta de transmisión. Una consideración importante de este sistema es que sus equipos terminales deben ser baratos. Los errores de transmisión pueden producir una degradación visible, pero sin ser fatal.

Luego de que NHK (Corporación de Radiodifusión de Japón) comenzó investigando el sistema HDTV (Televisión de Alta Definición) de 1125 líneas, los componentes y subsistemas para el sistema HDTV ya se han desarrollado conjuntamente con el progreso de esta tecnología. Por otro lado, las imágenes de HDTV que contienen un gran aval de información, se consideran para proveer un importante medio de transmisión dedicado para esta información. A fin de concertar el alto potencial del sistema HDTV en la transmisión de datos de imágenes fijas, se ha creado el sistema de disco digital de HDTV. Este sistema de información HDTV de alta calidad desde una cámara, convierte tal información analógica de video, a señal digital, y se la edita, graba, y reproduce sus datos. La cuantificación de esta señal a través de una conversión lineal de 8bits, necesita una alta velocidad de transmisión de cerca de 800Mbits/seg.

1.4 COMPRESION DE DATOS VERSUS COMPRESION DE ANCHO DE BANDA

1.4.1 IMAGENES COMPRIMIDAS:

La compresión de imágenes se relaciona con la idea de minimizar el número de bits requerido para representar a una imagen. Quizás la más simple y más dramática forma de compresión de datos es el muestreo de imágenes limitadas en banda, donde un número infinito de pixeles por unidad de área se reduce a una muestra sin ninguna pérdida de información

para el ojo humano (asumiendo que se dispone de un filtro pasa bajos ideal). Las aplicaciones de la compresión de datos se hallan principalmente en la transmisión y el almacenamiento de información. Las aplicaciones de transmisión de imágenes están en la radiodifusión televisiva, sensores remotos vía satélite, comunicaciones militares, radar y sonar, teleconferencias, comunicaciones computacionales vía redes como la INTERNET, transmisión de facsímiles, y otras aplicaciones de ese estilo. El almacenamiento de imágenes se requiere en los documentos educativos y de negocios, imágenes médicas que se obtienen de las tomografías por computador, imágenes generadas por resonancia magnética y de radiología digital, películas en movimiento, imágenes satelitales, mapas terrestres, agrimensura y Geología.

La aplicación de la compresión de datos también es posible en el desarrollo de algoritmos rápidos donde el número de operaciones requeridas para la implementación de un algoritmo se reduce al trabajar con imágenes comprimidas.

Al tener un cuadro de una imagen, un sensor, se desplaza a lo largo y a lo ancho de ella, realizando un "Barrido" primero horizontal y luego vertical; cada barrido horizontal representa una línea de imagen, así la imagen se la divide en un grupo de líneas horizontales que contienen información, el barrido de una línea, se lo hace identificando puntos de luz, punto por punto, en forma discreta, así un punto tendría un tamaño físico muy reducido, con el fin de captar la mayor cantidad de información de ese pequeño pedazo de imagen llamado "Pixel". Debido a que el barrido es discreto, existe un cierto número de pixeles por línea barrida(n), y así a lo ancho (verticalmente) de la imagen se barrerá un determinado número de líneas(m), teniéndose por lo tanto:

$$\# \text{total de puntos que confoman un cuadro} = n \cdot m = N$$

En realidad no interesa sobremanera la transmisión, o el almacenamiento de un solo cuadro de imagen, sino, como en televisión, una serie de imágenes, que se presentan una a continuación de otra, para dar la idea de movimiento de la imagen, dicho movimiento continuo de imágenes tiene un ritmo(v) de aparición de un cuadro por unidad de tiempo, lo que en conjunto, con el valor de N hallado anteriormente, indica un resultado interesante, el "Ritmo de aparición de un pixel", o los pixeles por unidad de tiempo:

$$R = N v \text{ (pixeles/segundo)}$$

Ahora se debe asignar niveles lógicos a la celda de imagen (pixel) captada, pues un pixel no necesariamente es idéntico a otro pixel de la imagen barrida, así entra en juego la idea ya mencionada anteriormente, la "DIGITALIZACION DEL PIXEL", es decir llevar a la señal visual contenida en un pixel de imagen, por el proceso de:

- Interpretarlo eléctricamente,
- Muestrear esta señal eléctrica;
- Cuantificar dicha señal; y,
- Codificarla (aquí entra la compresión).

Ya se han discutido estos procedimientos por separado, los cuales convergen al final para obtener una codificación a ocho bits de las señales muestreadas y cuantificadas. Esto por efectos de regulaciones y consensos que se han establecido internacionalmente para la codificación de señales analógicas. A fin de aumentar la calidad de imagen digitalizada (rendimiendo en la cuantificación) se ha concluido que al aumentar el número de bits que representarán a una muestra digitalizada, se aumenta la fiabilidad de la muestra, mas nó el rendimiento de cuantificación. Si a fin de cuentas lo que se quiere es tener una imagen lo más nítida posible, la opción es el aumento del número de bits, y así se lo ha hecho. En la actualidad ya existen normativos, recomendaciones y postulados que dirigen el diseño de codificadores-decodificadores de video, con las normas de los 8, 12 y 16

bits para codificación. Así si "s" es el número de bits por muestra asignado, según sea s=8 o s=12, o cualquier otro estándar, el ritmo de transmisión de Nyquist es:

$$R_s = N \cdot v_s \text{ (bits por segundo)}$$

Por ejemplo:

Una señal video-telefónica en la forma digital abarca 360 pixels por línea, a la vez 288 líneas por trama de video, y 12 bit de datos por pixel, a razón de 10 tramas por segundo:

$$360 \times 288 \times 12 \times 10 \approx 13 \text{ Mbps (Mega bits por segundo)}$$

Una componente de calidad tipo estudio de una señal digital de video, tal como una componente 4:2:2 (Capítulo III) conforme a la recomendación 601 del CCIR, requiere una velocidad binaria muy grande: 720 pixels por línea, 486 líneas por trama, 16 bits por pixel, 30 tramas por segundo:

$$720 \times 486 \times 16 \times 30 \approx 168 \text{ Mbps}$$

Y en el punto más alto del espectro digital, la señal de TeleVisión de Alta Definición(HDTV) requiere velocidades en el rango de los 1.5Gbps-o gigabits por segundo!

En líneas anteriores se halló que para un sistema MIC uniforme:[29]

$$q = \frac{2A_{\max}}{2^n} \quad (1.15)$$

donde A_{\max} es la amplitud (no sobrepasable) máxima. Y que sustituyendo en la ecuación del SNR (Relación de señal a ruido de cuantificación) se tiene:[29]

$$SNR(dB) = 10 \log_{10} \left(\frac{v^2}{q^2/12} \right) \quad (1.16)$$

donde v es la amplitud RMS de la señal de entrada.

Entonces: [29]

$$SNR_{dB} = 1.76 + 6.02n + 20 \log_{10} \left(\frac{A}{A_{max}} \right) \quad (1.17)$$

Es decir, se puede hallar una relación entre el SNR y los bits(n) utilizados para representar las muestras, así:

- Si el número de intervalos(q) aumenta, el SNR aumenta.
- Conviene aumentar el número de niveles de cuantificación para evitar que el ruido afecte de sobremanera a las pequeñas señales. Por lo que se debe tener un SNR uniforme, así se tendrá una calidad justa y necesaria en la cuantificación-codificación.
- Para cuantificación no uniforme (Ley A y μ), independientemente del tamaño de las amplitudes, el SNR es constante.
- A fin de compensar el bajo rendimiento de la cuantificación para señales pequeñas, lo que se pudiera hacer es asignar a señales pequeñas un número de intervalos(N_1) y a las señales grandes otro número de intervalos(N_2), con $N_2 < N_1$, o haciendo variables los intervalos.

La ley A o μ se basa en la codificación a 8 bits, teniendo sus curvas características de compresión 256 niveles de cuantificación (2^8 , 8 =número de bits). Se escoge 8 porque se debe tener una SNR aceptable (para cuantificación no uniforme), se debe mantener un margen de SNR constante, dependiendo del número de bits, y/o el número de intervalos. Como se mencionó existen estándares establecidos con respecto al uso de un determinado número de bits para representar una muestra, en el caso de escoger 12 bits, el tipo de cuantificación es uniforme (igual número de intervalos tanto para señales pequeñas o grandes), así el SNR es aceptable, es decir al aumentar el número de bits, con cuantificación uniforme, el SNR no se afecta demasiado.

1.4.2 REDUCCION DEL FLUJO DIGITAL DE DATOS:

Esta reducción se la consigue con la **reducción de la velocidad binaria**: reduciendo la alta velocidad binaria de la información digital de la imagen sin comprometer la calidad del cuadro resultante. A pesar de que el campo de la compresión del video ha sido activamente estudiado por muchos años, solo recientemente se ha logrado un avance significativo en este campo y las técnicas VLSI han hecho posible la reducción sustancial de los bits/pixel de una señal de video digital sin dañar su calidad.

1.4.3 REDUCCION DEL ANCHO DE BANDA:

Existe un número de atributos estadísticos y psico-visuales de las imágenes, los que convenientemente aprovechados por varias técnicas de procesamiento de señales, permiten reducciones muy significantes en la cantidad de información y la necesidad de ancho de banda de transmisión.

Estos son:

- El ojo no necesita un alto grado de resolución de color en una imagen. Para detalles muy pequeños es suficiente la señal de brillo (luminancia). Todos los sistemas de TV a color de hoy (NTSC, PAL y SECAM) toman ventaja de este fenómeno al reducir el ancho de banda de la señal de crominancia en un cincuenta por ciento o más.
- Los detalles muy finos no se observan mientras las imágenes están en movimiento. (Para ilustrar esto, mueva esta página rápidamente a través de su campo visual varias veces y note que los detalles de la impresión no se distinguen claramente hasta que el movimiento baje o cese totalmente).

Aunque es importante que un sistema de HDTV sea capaz de reproducir detalles muy finos, estos rara vez son parte importante de toda la trama. Es necesario fijarse que en imágenes de TV, cuadros o escenas se notará que la cantidad de información de alta frecuencia (bordes de los objetos y detalles finos) es relativamente pequeña; la

proporción de componentes de alta frecuencia en las imágenes de TV actuales es poco significativa.

- En la mayoría de imágenes de TV existe alta correlación entre elementos de imagen (píxeles) adyacentes uno del otro, tanto vertical como horizontalmente.

Este último aspecto no es inusual para porciones significativas de imágenes de TV como todo un cuadro, que se mantiene estático durante muchas tramas, y aún cuando está presente cierta cantidad de movimiento, muchas áreas permanecen sin cambio alguno durante períodos relativamente largos. Al tomar ventaja de estos factores psíquicos y estadísticos, es posible comprimir grandemente a las señales de TV. Mientras la fuente de datos se tome directamente de una cámara de TV, el flujo de datos alcanza a 1 gigabit por segundo, las técnicas de compresión logran reducir estos flujos por un factor de 30 a 40, o más. El simple proceso de convertir una señal análoga a una señal digital, resulta en el incremento del ancho de banda para la transmisión (Véanse Figuras 1.16, 1.17 y 1.18 para visualizar el ancho de banda de un solo pulso cuadrado). Por ejemplo, una señal de televisión de 4.5-MHz, muestreada a un ritmo de muestreo de Nyquist con 8 bits por muestra, requiere un ancho de banda de 36MHz (72Mbps) cuando se transmite utilizando un esquema de modulación digital, tal como la modulación PSK, con 1Hz por cada 2 bits. Así, aunque la información digitalizada tiene sus ventajas sobre su forma análoga en términos de flexibilidad de procesamiento, el acceso aleatorio al almacenar, alta relación Señal/Ruido para la transmisión con la posibilidad de la comunicación libre de errores, y así por el estilo, se tiene que pagar el precio en términos de incremento en el ancho de banda (de 4.5MHz a 36MHz). La compresión digital busca minimizar este costo y a veces tratar de reducir el ancho de banda de la señal digital, incluso por debajo de sus requerimientos mínimos.

La investigación en el área de la compresión del ancho de banda se ha intensificado desde mediados de este siglo. La necesidad de reducir el ancho de banda de imágenes digitales es debido a los excesivos requerimientos de este parámetro de las imágenes digitales y la continua demanda de capacidad de los canales de comunicación. De otro lado, el procesamiento, el almacenamiento, la transmisión, y la discriminación de imágenes en la forma digital, ofrece muchas ventajas distintivas con respecto a estas operaciones en la forma analógica. Estas son la flexibilidad del procesamiento, la transmisión libre de errores, precisión controlada, registramiento preciso, facilidad de encriptamiento y codificación, facilidad de almacenamiento y recuperación, así como también la compatibilidad con redes digitales y computadores digitales.

1.5 HERRAMIENTAS MATEMATICAS UTILIZADAS EN LA COMPRESION

1.5.1 LA MATEMATICA DE LA COMPRESION:

Un requisito importante para entender las técnicas de compresión de imágenes es el de tener conocimiento sobre:

- Probabilidad y Estadística.
- Teoría de la Información y la Codificación.
- Análisis en los dominios temporales, espaciales y de frecuencia.

Es decir, en general, un conocimiento de las bases del procesamiento digital de señales analógicas.

1.5.1.1 Probabilidad y Estadística:

Se ha observado que en diferentes áreas que el resultado de un experimento depende de la combinación de muchos factores impredecibles pero sin embargo ciertos *promedios* o características medias se aproximan a un *valor constante* cuando el número de observaciones se incrementa. Se observa también que este valor constante se mantiene si los promedios son nuevamente evaluados sobre cualquier conjunto de realizaciones en términos de las *probabilidades de eventos* o *verosimilitud de eventos*.

Los resultados obtenidos de la probabilidad y la estadística y que se los halla en algún proceso digital de señales son ya típicos como las permutaciones, las combinaciones, la esperanza matemática($E\{.\}$), la varianza(σ^2), el valor cuadrático medio($E\{.^2\}$), el BER, las funciones de distribución de probabilidad($F(.)$) y las funciones de densidad de probabilidad($f(.)$), la correlación y sus coeficientes de correlación, entre los más utilizados. Si se habla de esperanza matemática de los valores digitales de una imagen, por ejemplo, se está hablando del valor del pixel más probable, o la media de los valores de pixeles; en cambio si se escucha nombrar a la varianza de la imagen, se dice que aquella es la medida de cuan cerca están los valores de los pixeles del valor medio, si se tiene una varianza pequeña (relativa) los valores de los pixeles están más cerca al valor medio que si la varianza es grande (relativa). Al hablar de valor cuadrático medio se está entrando ya al aspecto energético de una señal, en efecto la definición de valor cuadrático medio es:[42]

$$E\{x^2\} = \sigma^2 [x] + (E\{x\})^2 \quad (1.18)$$

Y si x se considera un voltaje de 1V sobre una resistencia de 1Ω , la anterior ecuación representaría:

$$\text{Potencia Total} = \text{Potencia AC} + \text{Potencia DC} \quad (1.19)$$

Si se considera un proceso estacionario, el término **DC** normalmente no representa mucho contenido de información por lo que se lo desecha y toda la potencia se la halla dentro de la varianza.

1.5.1.2 Teoría de la información y la codificación:

1.5.1.2.1 Información:

Existe una mínima cantidad de datos que serán suficientes para describir a la imagen sin pérdida visible de información? La respuesta depende de hasta donde se define una útil medida de INFORMACION. La cantidad de información en

un cuadro es inversamente proporcional a la probabilidad de de su ocurrencia. En otras palabras, un evento menos probable incorpora más información que un evento más probable. La información, en este sentido, es similar a un valor "sorpresa".

"El sol saldrá mañana" es una expresión con una probabilidad(P) que es virtualmente del 100%. Se podría decir que para este evento, $P(\text{evento})=1$: el evento siempre ocurre. Ninguna información se le atribuye a este evento pues ninguna incertidumbre está asociada a él.

Se pudiera formular la expresión que dice "El sol no saldrá mañana", y así se estaría tomando un evento extremadamente raro-en el cual $P(\text{evento})\approx 0$. La ocurrencia de tal evento conducirá a un alto grado de sorpresa (y alarma), y así a una gran cantidad de información. Otra propiedad en esta definición de información de imagen es que la gran cantidad de información es la suma de informaciones acarreadas por cada parte. Para una imagen de video, estas partes son los llamados **pixeles**.

Combinando estas dos propiedades, se puede producir una medida de información expresada por la siguiente relación:[35]

$$I(x_1) = \log_2 \frac{1}{P(x_1)} \quad (1.20)$$

donde x_1 es un evento particular. Esta ecuación define el concepto de medida de información en unidades de bits, o "UN BIT DE INFORMACIÓN"-el cual es muy diferente a "UN BIT DE DATOS". En realidad los bits de datos son la medida de comunicación de los bits de información. Varias cantidades de datos pueden ser utilizados para comunicar la misma cantidad de información.

1.5.1.2.2 Entropía:

Solo en el caso particular de que los eventos tengan igual probabilidad de ocurrencia, los bits de datos son iguales a los bits de información. Cuando uno de dos eventos posibles igualmente probables ocurre, 1 bit de información es transportado- y solo 1 bit de datos es necesario para comunicar el resultado. La importancia de definir a la información de este modo, es que permite calcular el *contenido promedio de información* de una fuente. Una fuente, X (por ejemplo, una imagen de video), se puede describir como un arreglo finito de posibles símbolos fuente (es decir, píxeles: x_1, x_2, x_3 , etc.) y sus probabilidades de ocurrencia. La información promedio de una fuente se expresa usualmente por $H(X)$, y se la llama *entropía* de la fuente. La expresión 1.21 muestra como la entropía de una fuente puede ser calculada: [35]

$$H(X) = \sum_{i=1}^N P(x_i) \cdot I(x_i) = \sum_{i=1}^N P(x_i) \cdot \log_2 \frac{1}{P(x_i)} \quad (1.21)$$

donde las unidades de $H(X)$ se dan en:

$$H(X) \equiv \text{bits/símbolo}$$

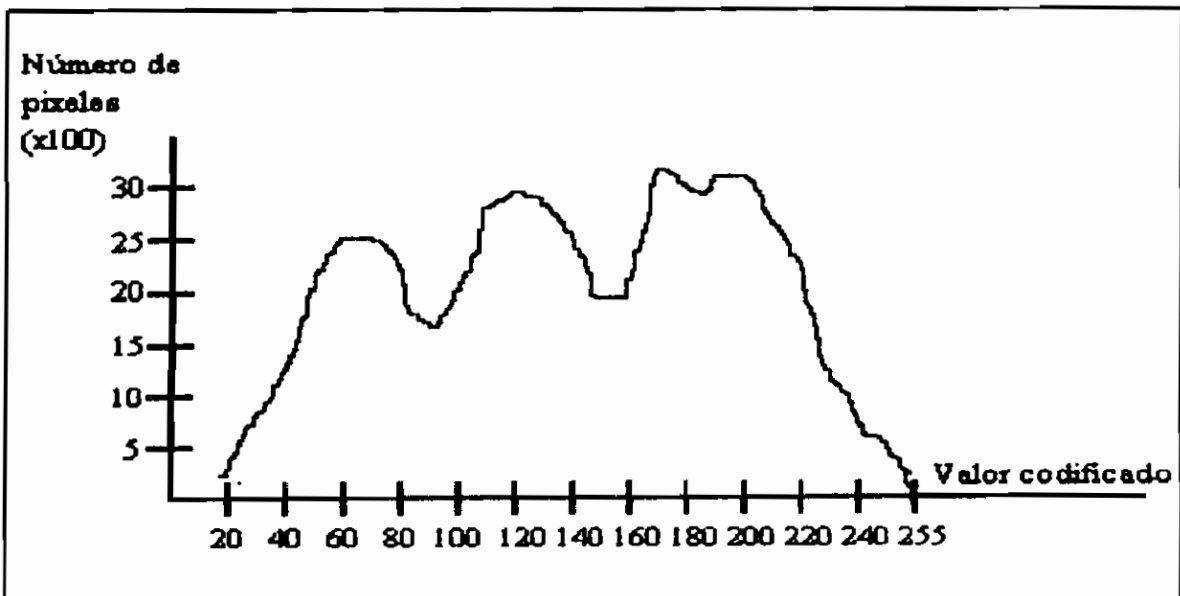


Figura 1.16 Histograma de una imagen en blanco y negro.

La Figura 1.16 representa un "Histograma" para una imagen en blanco y negro. En el caso de una imagen de tonos continuos de gris con 8 bits por pixel, el arreglo de símbolos fuentes contiene 256 valores (del 0 al 255, los cuales se describen por medio de un código binario de 8 bits de longitud). La probabilidad de cada ocurrencia por símbolo (o el valor de un pixel individual) se calcula por el número de ocurrencias de ese valor particular sobre el número total de pixeles en la imagen.

De acuerdo al *teorema de Shannon para la codificación sin ruido* es posible realizar una codificación sin distorsión, para una fuente de entropía H bits/símbolo, usando $H+\epsilon$ bits por símbolo, donde ϵ es una cantidad positiva arbitraria muy pequeña. Así se define la máxima compresión alcanzable C , [27]

$$C = \frac{\text{vel. promedio de la data original}(B)}{\text{vel. promedio de la data codificada}(H+\epsilon)} \quad (1.22)$$

$$C = \frac{B}{H + \epsilon} \approx \frac{B}{H} \quad (1.23)$$

El cálculo de tal razón de compresión de imágenes es impráctico. Por ejemplo en una imagen digital de $N \times M$ pixeles, con B bits/pixel, la razón daría 1 entre $L = 2^{BNM}$ posibles formas de imágenes a obtener. Así si p_i es la probabilidad del i -ésimo patrón de la imagen, se podría computar la entropía, es decir la razón de información para imágenes de $N \times M$ pixeles y B bits por pixel. Así se podría almacenar todos los L patrones de la imagen, y codificar a la imagen según sus direcciones-utilizando un método adecuado, el cual requerirá aproximadamente H bits por imagen o H/NM bits por pixel. Tal método de codificación se denomina *codificación bloque o cuantificación vectorial*. La principal dificultad de este método es que aún para pequeños valores de N y M , L puede ser prohibitivamente grande. Por ejemplo para $B=8$, $N=M=16$ se tiene: [27]

$$L = 2^{2048} \approx 10^{614}.$$

La entropía de una imagen puede ser estimada de su entropía condicional. Para un bloque de N pixeles u_0, u_1, \dots, u_{N-1} con B bits por pixel, y arreglado en un orden arbitrario, La entropía condicional de orden N se define como:

$$H_N \Delta \sum_{u_0} \dots \sum_{u_{N-1}} p(u_0, u_1, \dots, u_{N-1}) \log_2 \frac{1}{p(u_0 | u_1, \dots, u_{N-1})} \quad (1.24)$$

donde cada u_i , $i = 0, 1, \dots, N-1$, toma 2^n valores, y $p(\dots)$ representa las probabilidades relevantes.

1.5.1.2.3 Acerca del ruido:

En una imagen digital de tonos continuos de gris con ocho bits por pixel, el máximo número de bits de información por pixel (o la máxima tasa de bits/pixel generada por la imagen) puede producirse cuando los pixels son enteramente independientes unos de otros y sus valores ocurren con igual probabilidad. En realidad, un valor de entropía (o información) de 8 bits, significa que la cantidad promedio de bits por pixel podría no ser reducida sin la pérdida de información visual.

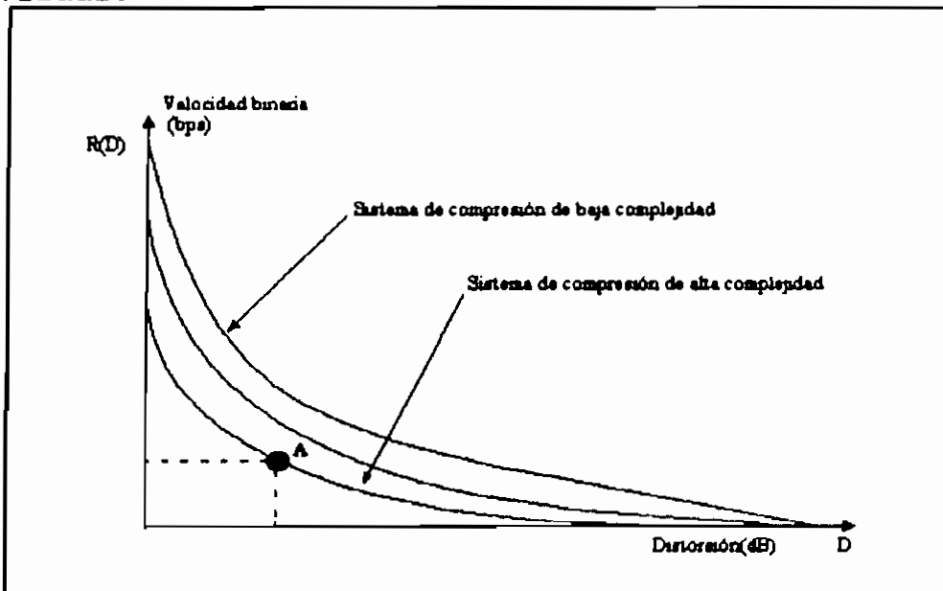


Figura 1.17 Bits/seg vs. Distorsión.

Las imágenes de video normales, afortunadamente, exhiben una alta correlación entre píxeles. Si el objetivo de la compresión de video es la reducción de la tasa binaria, sin distorsión, entonces la entropía de la secuencia del video no impone un limitante. Pero si se considera tolerable una cierta cantidad de distorsión, esto puede dar lugar a una significativa reducción en la tasa binaria bajo el valor de entropía de una determinada secuencia. En la Figura 1.17 se muestra como se relacionan los rangos de compresión respecto a la distorsión.

En la Figura 1.17 se muestra la curva de la razón de distorsión $R(D)$, que fue primeramente introducida en 1948 por Claude Shannon de los laboratorios Bell, y nos muestra el menor valor de velocidad binaria para el cual la información de una fuente puede transmitirse (bajo determinado criterio de distorsión permitida) por medio de una técnica de codificación eficiente. Por ejemplo, el punto A en la curva $R(D)$ corresponde a las condiciones de operación óptima de un sistema de compresión con una determinada tasa de velocidad binaria mínima para una medida de distorsión dada.

1.5.1.2.4 Procesos de descompresión-corrección de errores:

Una teoría general sobre los métodos de compresión de imágenes no puede estar completa si no se realiza un estudio adecuado sobre sus respectivos métodos de descompresión. Así como existe una infinidad de métodos de compresión (aunque cada uno de ellos puede ser englobado en un tipo, de un pequeño grupo de categorías), deben existir sus contrapartes de descompresión. Como complemento a la descompresión el siguiente proceso debería ser la corrección o por lo menos la detección de errores que se hayan producido en los datos codificados al pasar por el medio de transmisión o al reproducirlos cuando estos han ingresado a un medio de grabación.

En el Capítulo IV se analizará con más en detalle la generación y los efectos de los errores que en un sistema real de transmisión-recepción se producen. Las técnicas de corrección de errores se han desarrollado ya desde mucho antes de que existiese la idea misma de la compresión de imágenes, pues anteriormente los métodos de corrección de errores se implementaban en los codec (codificadores - decodificadores) de transmisión de datos, no precisamente de imágenes digitales sino de teletextos, de canales telefónicos multiplexados en el tiempo (MIC 30+2), etc.

Se verá en el Capítulo III en el que se tratará la radiodifusión digital televisiva, que la corrección de errores no es indispensable, resultando en un aumento de precio innecesario del equipo de recepción, debido a que las imágenes a proyectar en las pantallas de los televisores no necesitan ser en extremo precisas y detallistas, y que talvez lo único necesario es una corrección de croma, una eliminación de parpadeos, compensación del matiz, etc., es decir macrocorrectivos de cada cuadro presentado. Por otro lado, si las imágenes a receptor necesariamente deben ser idénticas a las transmitidas, en el mayor grado posible como son las fotografías tomadas por un satélite, las tomografías digitales, imágenes para aplicaciones militares, etc. En este caso es realmente cuando la corrección de errores es vital en el equipo de recepción.

1.5.1.3 Análisis en el tiempo, el espacio y la frecuencia:

1.5.1.3.1 El concepto de modulación:

Ya se ha tratado el tema de la modulación en párrafos anteriores (Secciones 1.1.6 y 1.1.4), debido al criterio de que para digitalizar a una señal análoga es necesario muestrearla con otra señal de características diferentes también es natural, e inmediato. La finalidad de ese tipo de modulación es entonces generar la digitalización de una señal análoga.

Pero existe otra finalidad para la modulación, orientada a la transmisión de información digital por un medio físico adecuado para ésta. La MAP en el dominio del tiempo equivale a la multiplicación de la señal modulante y a la portadora (Figura 1.7). En el dominio de la frecuencia, equivale a una traslación de frecuencia. Los esquemas de modulación mencionados anteriormente se refieren a la denominada transmisión en *banda base* (Figura 1.7 y Figura 1.11) en tales esquemas se reproduce la señal original, por medio de un filtro pasa bajos es decir en las cercanías de frecuencia DC (muy baja). La mayoría de canales de comunicación real tienen una muy pobre respuesta en las vecindades de la frecuencia cero. Al transmitir información digital por medio de canales de paso de banda, se transfiere la información a una onda portadora de frecuencia apropiada. La información digital puede incluirse dentro de una onda portadora de formas distintas, así se tienen los esquemas (Figura 1.18):

- *ASK*: Modulación en amplitud.
- *FSK*: Modulación en frecuencia.
- *PSK*: Modulación en fase.
- *QAM*: Modulación en cuadratura.

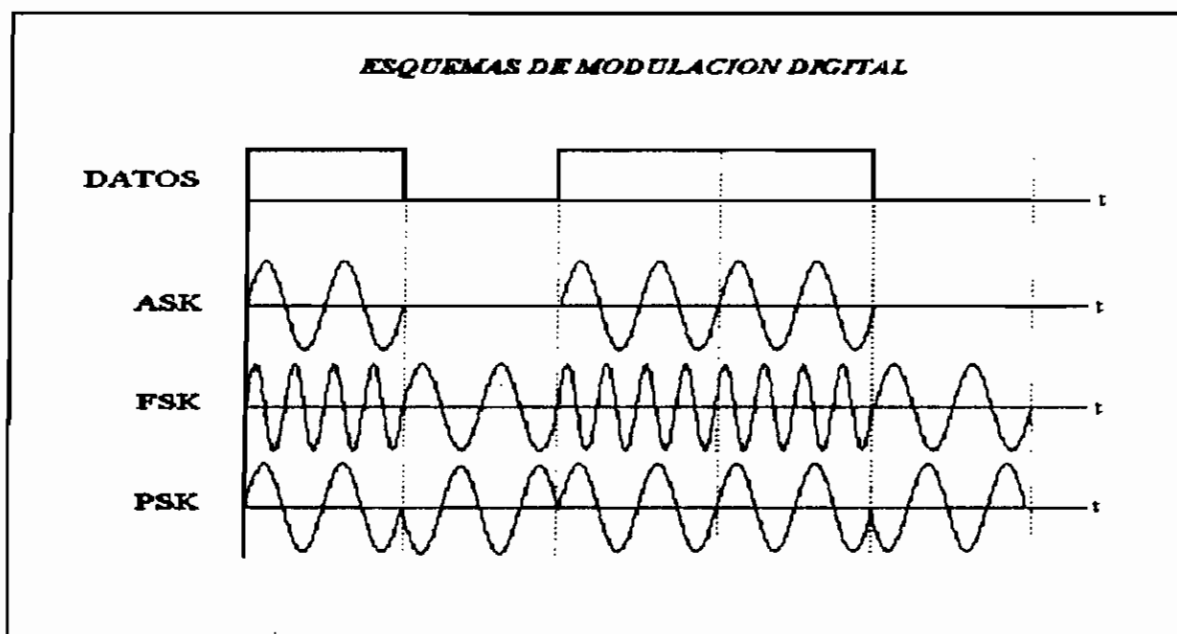


Figura 1.18 Esquemas de modulación digital.

1.5.1.3.2 Análisis por transformada:

La codificación por transformada involucra a transformaciones lineales en las cuales la señal espacial es mapeada a otro espacio donde las muestras transformadas luego se comprimen para su almacenamiento o transmisión. La operación de reconstrucción involucra a una transformación inversa de las muestras transformadas y comprimidas. Esta secuencia de operaciones se muestran en la Figura 1.19. La secuencia envuelve a dos operaciones en el lado de la fuente del canal: *Transformación y Cuantificación*; y dos operaciones en el lado receptor del canal: *Decodificación y Transformación Inversa*. Cada una de estas operaciones requiere cierta cantidad de memoria, y esta necesidad de memoria-particularmente en el terminal fuente del sistema- está envuelta en una de las consideraciones importantes en su complejidad. La selección de la transformación (y así mismo la transformación inversa) está a menudo dictada por consideraciones de implementación práctica, aún así es posible diseñar una transformación *óptima*, la transformada de componente principal o KL (Karhunen-Loève, véase 2.1.3(a) del Capítulo II).

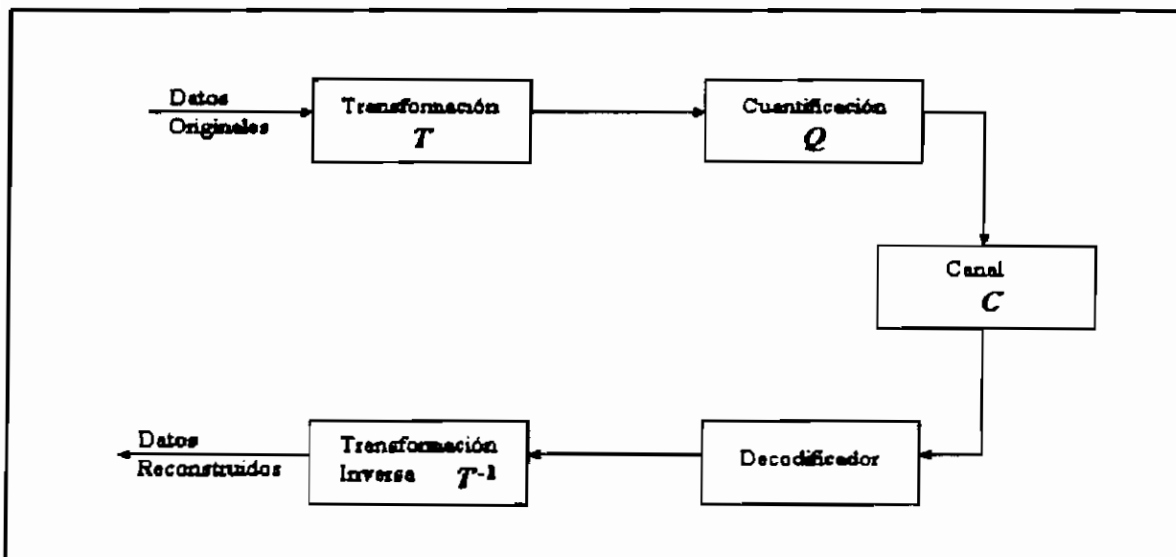


Figura 1.19 La transformada en codificación digital.

Capítulo

2

Métodos de Compresión

CAPITULO II

METODOS DE COMPRESION

2.1 CLASIFICACION DE LOS METODOS DE COMPRESION DE IMAGENES:

La compresión de datos es la reducción de la cantidad de la señal en el espacio que debe ser asignado a un conjunto de mensajes o conjuntos de muestras de datos. La señal en el espacio puede estar en un medio físico de almacenamiento o puede estar también en un intervalo de tiempo, tal como el tiempo requerido para transmitir un mensaje dado, o en una porción del espectro electromagnético tal como el ancho de banda requerido para transmitir un conjunto de mensajes.

De esta forma, la reducción de este volumen de información puede conseguirse disminuyendo el tiempo de transmisión o el ancho de banda. El parámetro a reducir o comprimir generalmente está determinado por el lugar del sistema en el cual la operación de compresión será realizada. La *compresión* de datos puede designarse con otros nombres tales como la *compactación* de datos o *codificación de fuente*.

Han existido muchos intentos para agrupar las técnicas de compresión de datos dentro de varias clases, pero, todavía no existe una clasificación en la que todos estén de acuerdo; debido a la infinidad de interpretaciones que se dan a las técnicas empleadas en la compresión, o porque periódicamente se generan más métodos de compresión, o a su vez porque se generan versiones mejoradas de los métodos existentes.

Los métodos de compresión de imágenes caen dentro de dos categorías principales. En la primera categoría, llamada *codificación predictiva*, están los métodos que explotan la *redundancia* en los datos.

La redundancia es una característica que involucra factores como la *predictividad (los datos pueden predecirse)*, *aleatoriedad ("randomness")*, y *uniformidad ("smoothness")* en los datos. Las técnicas tales como la modulación delta (MD) y la modulación de impulsos codificados diferencial (MICD) caen dentro de esta categoría. En la segunda categoría, llamada *codificación por transformada*, la compresión se lleva a cabo al transformar la imagen dada en otro arreglo de información a fin de que una gran cantidad de información sea *empaquetada* en una pequeña cantidad de muestras.

Existen otros algoritmos de compresión de imágenes que son combinaciones de los métodos ya señalados. Del proceso de compresión inevitablemente resulta alguna distorsión debido a la conversión de análogo a digital así como del rechazo de cierta cantidad de información que generalmente resulta insignificante. Las técnicas eficientes de compresión tienden a minimizar esta distorsión. Las técnicas de compresión de imágenes, en general, pueden ser clasificadas en dos grupos principales, según la magnitud de pérdidas que se produce:

- Sin pérdidas.
- Con muchas pérdidas.

(i) **COMPRESION SIN PERDIDAS:**

También conocida como compresión *reversible*, o codificación sin ruido. En este tipo de compresión, no se pierden datos. La señal de video comprimida puede ser descomprimida sin pérdida de ningún tipo de información. La compresión sin pérdidas, sin embargo permite una modesta reducción de velocidad binaria (a la mitad, o a un tercio, de la tasa original), la cual también depende del tipo de imagen de video. En este tipo de compresión, se reduce la redundancia de los datos. Algunas técnicas de codificación representativas de este grupo son por ejemplo la codificación de Huffman, la codificación secuencial o la codificación de la longitud de palabra.

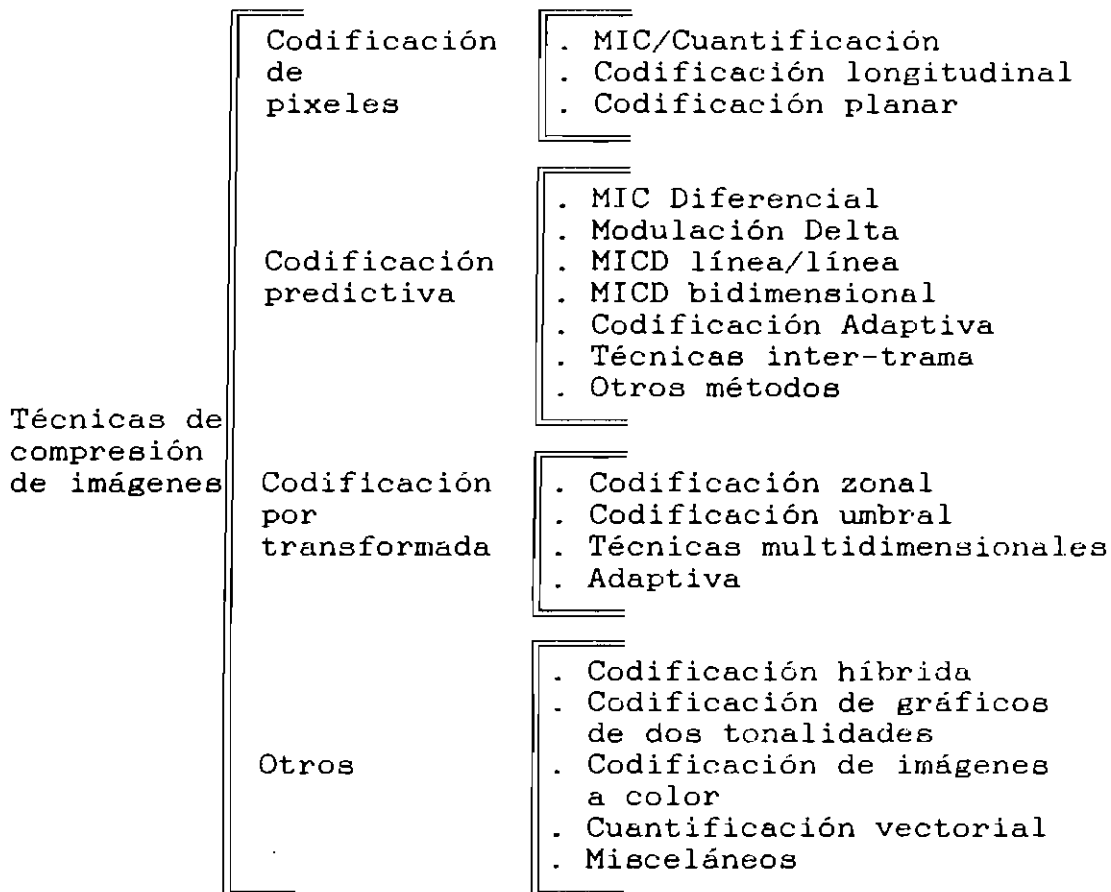
(ii) COMPRESION CON MUCHAS PERDIDAS:

Es *irreversible*, reduce la fidelidad, y la entropía: La imagen reconstruida, perderá algunos de los pixeles de datos originales. La compresión irreversible permite una alta reducción de velocidades binarias (reducciones en el orden de las decenas y centenas). En general a más compresión se crea más distorsión, pero con técnicas de compresión seleccionadas cuidadosamente, esta distorsión puede hacerse casi imperceptible al ojo humano. Esto es posible, en virtud de que los procesos visuales del ojo y del cerebro humano son imperfectos. Las pequeñas diferencias en la amplitud de los pixeles se pueden confundir cuando se acercan a grandes transiciones de bordes. En áreas de cuadros con intensa actividad espacial, la compresión de imágenes puede remover datos de pixeles sin introducir distorsión (*artefactos de codificación*) que el ojo humano perciba. Por otro lado, el sistema visual humano es muy sensible a las distorsiones que aparecen en áreas de intensidad uniforme de una imagen, especialmente si esta es el resultado del entremezclado de tramas de video. En estas áreas, la compresión de imágenes debe actuar más conservadoramente. Un aspecto estandarizado en todos los sistemas de televisión alrededor del mundo es el *entrelazado* de líneas: las líneas impares se barren en un primer campo, y las líneas pares en un segundo campo. Los dos campos constituyen un cuadro completo de televisión. Aunque la información de un cuadro en la señal de televisión no es realmente continua, el cerebro humano así lo interpreta debido a la alta velocidad de barrido de las líneas.

Lo que es válido para video, también lo es para la compresión de video. El ojo y el cerebro humanos tienen una menor respuesta a la distorsión de los artefactos de codificación de objetos en movimiento al seguir un cambio inmediato de escena. Usando técnicas que reducen la redundancia entre campos visuales y cuadros a lo largo del tiempo, la compresión de video puede ser aplicada sin pérdidas visibles en la calidad de la escena.

Explotando las características del sistema visual humano en ambos procesamientos (espacial y temporal) de la señal, se puede lograr una compresión de video que es *visualmente libre de pérdidas*: produciendo pérdidas de calidad no perceptibles bajo condiciones visuales normales.

Una reducción de entropía resulta en una reducción de información. La información perdida nunca podrá ser recuperada. Una reducción de redundancia, remueve o al menos reduce la redundancia de tal forma que pueda ser nuevamente reinsertada en los datos. Dentro de este grupo están por ejemplo la codificación por bloques como las codificaciones por transformada (Fourier, Hadamard, Haar, etc.). El siguiente cuadro presenta una clasificación sumaria de las varias técnicas de compresión de imágenes:



Antes de entrar de lleno a desarrollar una explicación de los métodos de compresión de imágenes es necesario recalcar los objetivos a donde apuntan todos estos métodos y técnicas de compresión: *La disminución de velocidades binarias (bits/segundo)*, pues dicha reducción implica un abaratamiento en anchos de banda de transmisión; y el aumento de capacidades de almacenamiento en dispositivos tradicionalmente "estrechos" de volumen de almacenamiento, al disminuir la cantidad de símbolos unidad (bits) para representar a una muestra digital (pixel), es por eso que se busca *la disminución de entropía* de una fuente de muestras pictóricas que componen a una imagen. En resumen lo que las técnicas de compresión de imágenes logran es:

- (1) Disminuir los *bits por segundo (bps)* en una transmisión digital.
- (2) Disminuir los *bits por símbolo* transmitido o almacenado según sea la aplicación que se busque.

En realidad la disminución de los bits por segundo en una transmisión es nada más que la reducción de la velocidad digital a la que se transmiten los símbolos cuya representación binaria se busca disminuir por los métodos de compresión, es decir se está hablando de la misma reducción, solo que con una característica de lenguaje distinta.

CODIFICACION POR LONGITUD VARIABLE(CLV):

Uno de los primeros métodos para alcanzar una reducción es por medio del mejoramiento de la *eficiencia de la codificación* de los pixeles en la imagen. Se procede en primer lugar a calcular la frecuencia de la ocurrencia de los diferentes valores de pixeles (cuantos pixeles son negros, cuantos son verdes, cuantos celestes, cuantos azules, etc.), luego se asignan menos bits para los valores más probables (los de mayor ocurrencia) y un mayor número para los menos probables (los que existen en menor cantidad). Este proceso es conocido como *Codificación por Longitud Variable*.

Si los valores de pixeles son codificados en una manera que, en promedio exceda el mínimo número de bits por pixel requerido por el valor de la entropía de la imagen, entonces la representación digital resultante se dice contiene *Codificación Redundante*. La codificación redundante resulta cuando los códigos asignados a valores de pixeles no toman toda la ventaja de la distribución de probabilidad de los pixeles. La codificación binaria directa de la amplitud de pixeles, como la codificación producida por un convertidor análogo a digital, es ineficiente, pues esta asigna el mismo número de bits de datos para el más y el menos probable de los valores de pixeles, conteniendo así una gran cantidad de código redundante. Para capturar toda la información en una imagen tipo "ruido blanco gaussiano" (imagen donde todos los valores de los pixeles son igualmente probables), la codificación binaria directa podría ser apropiada. Pero imágenes comunes tienen una estructura más aleatoria; éstas, sin ser similares a la imagen con ruido blanco tienen sus elementos organizados.

Para imágenes comunes, la codificación por longitud variable es más eficiente que la codificación binaria directa. *La Codificación por Longitud Variable es también una técnica de compresión libre de pérdidas*, debido a que los valores muestreados originales se pueden recuperar exactamente vigilando que no hayan ocurrido errores en la transmisión o en el almacenamiento del flujo de bits.

Se pueden generar tablas de codificación por longitud variable (CLV) por medio de la distribución específica de probabilidad que describe cada imagen. Si los valores CLV de una imagen se usan para codificar otra imagen diferente, con más o menos movimiento, no habría distinción entre objetos, variaría la textura de los cuadros, y las estadísticas de los pixeles no serían compatibles. En realidad, podría darse el caso de que la velocidad binaria de la nueva imagen sea aún más alta que la que se alcanzaría con

una codificación binaria directa. Varias *transformaciones* de la imagen o *mapificación* han sido estudiadas a fin de reducir los altos niveles de redundancia de imagen y así hacer a la técnica CLV más independiente de la clase de imagen a ser codificada. En la dimensión espacial, por ejemplo, la diferencia promedio entre pixeles adyacentes va a ser pequeña. Solo cuando ocurran altas transiciones de contraste, parecerá que la diferencia promedio de una imagen visual, o *señal diferencia*, toma grandes valores. La mayor diferencia entre valores codificados será para las pequeñas muestras, y esta diferencia estará centrada alrededor de cero, haciendo a la señal diferencia altamente no uniforme (Véase la Figura 2.28). Debido a esta gran desviación estadística, una señal diferencia mostrará baja entropía y puede ser eficientemente codificada usando una técnica de codificación por longitud variable.

2.1.1 CODIFICACION DE PIXELES:

En esta técnica, cada pixel es procesado en forma independiente, ignorando las dependencias que éste tenga con sus pixeles vecinos conformadores de la imagen.

2.1.1.1 Modulación por Impulsos Codificados(mic):

Este tipo de codificación ya fue expuesta en el Capítulo I. Se analizó que esta modulación es una extensión de la modulación por amplitud de pulso(MAP), en la cual cada valor muestreado es cuantificado a un valor discreto para su representación digital en una palabra código. Fueron expuestas además las definiciones de ruido de cuantificación, relación Señal/Ruido(SNR), MIC uniforme y no uniforme, además de las leyes A y μ de cuantificación no uniforme.

Estas leyes son de mucha utilidad en los sistemas codificadores de imágenes. Con la MIC, la señal de video entrante es *muestreada, cuantificada, y codificada* a una palabra código adecuada (antes de alimentarla a un modulador digital para su transmisión) (Figura 2.1).

La salida del cuantificador es generalmente codificada en una palabra código de longitud binaria fija de B bits. Comúnmente, son suficientes 8 bits para la radiodifusión monocromática o para video-conferencia con imágenes de buena calidad, mientras que para imágenes médicas o señales de video a color pueden requerirse de 10 a 12 bits por pixel.

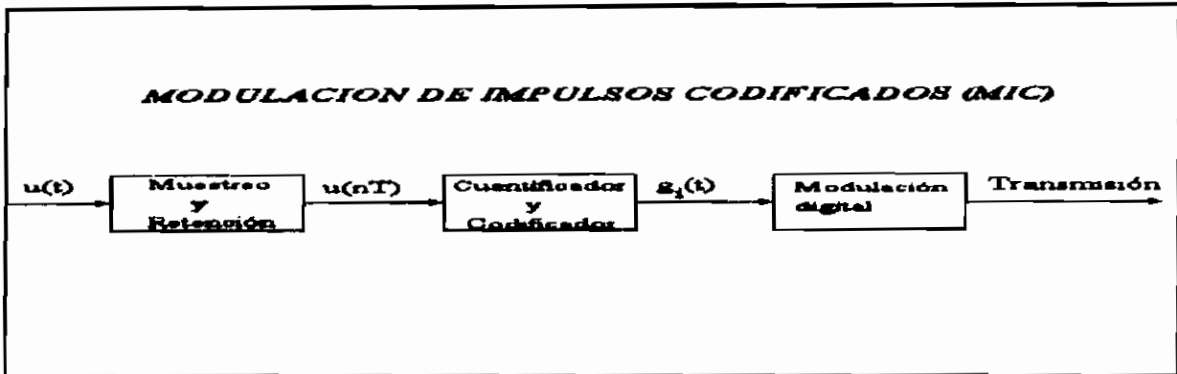


Figura 2.1 Modulación de Impulsos Codificados(MIC).

El número de bits de cuantificación necesarios para representar a una imagen se puede reducir a 4 o a 8 bits por pixel al usar compensación, cuantificación del contraste, o las técnicas bi-tono que reducen la salida de los cuantificadores a 1 bit por pixel, para esto la velocidad de muestreo debe incrementarse en factores de 2 a 16. La compresión alcanzada por estas técnicas generalmente no supera la razón de 2 a 1. En términos de la distorsión cuadrática media, la mínima tasa de velocidad binaria de transmisión de la MIC (R_{MIC}) está dada por la fórmula de la tasa de distorsión:[27]

$$R_{MIC} = \frac{1}{2} \log_2 \frac{\sigma_u^2}{\sigma_q^2}, \quad \sigma_q^2 < \sigma_u^2 \quad (2.1)$$

donde,

σ_u^2 , es la varianza de la entrada al cuantificador; y,

σ_q^2 , es la distorsión cuadrática media del cuantificador.

2.1.1.2 Codificación de la entropía:

Si los pixeles cuantificados no son distribuidos uniformemente, entonces su entropía será menor a B bits/pixel, así existirá un código que utilizará menor cantidad de B bits por pixel.

En la codificación de la entropía el objetivo es codificar un bloque de M pixeles conteniendo $M \cdot B$ bits con probabilidades p_i , $i=0,1,2,\dots,L-1,L$; $L=2^{M \cdot B}$, por $-\log_2 p_i$ bits, así que la tasa binaria promedio (bits/muestra) es:[27]

$$\sum_I^L p_i \log_2 \frac{1}{p_i} = H \quad (2.2)$$

Esto resulta en una *codificación por palabras código de longitud variable, CLV* para cada bloque, donde el bloque de la más alta probabilidad se representa por códigos de pequeña longitud, y viceversa. Para un bloque de tamaño dado, una técnica llamada *Codificación de Huffman* es el método más eficiente de codificación por longitud variable.

2.1.1.2.1 El algoritmo de codificación de Huffman:

Este algoritmo es de naturaleza probabilística cuyo desarrollo se indica a continuación:

- 1.- Ordenar las probabilidades de los símbolos p_i , en un orden decreciente y considerarlos como los nodos de las hojas de un árbol.
- 2.- Mientras haya más de un nodo:
 - Emparejar los nodos con la menor probabilidad a fin de formar un nuevo nodo cuya probabilidad es la suma de las probabilidades de los dos nodos emparejados.
 - Arbitrariamente asignar 1 y 0 a cada par de bifurcaciones confluyentes al nodo.
- 3.- Leer secuencialmente desde el nodo raíz al nodo de la hoja, en donde el símbolo se localiza. La codificación y decodificación se la realiza simplemente al mirar valores en una tabla formada de 1's y 0's.

Otra alternativa es el llamado *código de Huffman modificado*, donde el entero i se representa como:[27]

$$i = qL_1 + j, \quad 0 \leq q \leq \text{Ent}\left(\frac{L-1}{L_1}\right), \quad 0 \leq j \leq L_1 - 1 \quad (2.3)$$

Los primeros L_1 símbolos son codificados por Huffman. El resto de símbolos son codificados por un código de prefijo, representando el cociente q , seguido de un *código terminal*, el cual es el mismo que el código Huffman para los restantes j símbolos, $0 \leq j \leq L_1 - 1$.

2.1.1.3 Codificación longitudinal:

Considérese una fuente binaria constituida de unos y ceros cuya salida está codificada como el número de 0's entre dos 1's, esto es, se codifica la longitud de secuencias de 0's. Este proceso se llama *Codificación Longitudinal (CL)* pues representa la codificación de una palabra código, por la longitud de las secuencias de 0's que en ella existen. Esta codificación es útil cuando se espera codificar largas secuencias de ceros. Tales situaciones ocurren en documentos impresos, gráficos, mapas climáticos, y así por el estilo, donde p , la probabilidad de un cero (el que representa a un pixel blanco) se aproxima a la unidad. Supóngase que las secuencias se codifican en longitudes máximas de M , y por simplicidad, sea que:[27]

$$M = 2^m - 1 \quad (2.4)$$

Esto es, se tomarán m bits para codificar cada secuencia por un código de longitud fija. Si los sucesivos 0's ocurren independientemente, entonces la distribución de probabilidad de las longitudes de las secuencias produce la distribución geométrica:[27]

$$g(l) = \begin{cases} p^l(1-p), & 0 \leq l \leq M-1 \\ p^M, & l = M \end{cases} \quad (2.5)$$

Como una secuencia de longitud $l \leq M-1$ implica una secuencia de l 0's seguida de un 1, esto es, $(l+1)$ símbolos, el número promedio de símbolos por secuencia será:[27]

$$\mu = \frac{1-p^M}{1-p} \quad (2.6)$$

Así, se toman m bits para establecer un código longitudinal para una secuencia de μ símbolos binarios, en promedio. La compresión alcanzada es, por lo tanto:

$$C = \frac{\mu}{m} = \frac{1-p^m}{m(1-p)} \quad (2.7)$$

2.1.1.4 Codificación planar:

Una imagen de 256 niveles de gris se puede considerar como un arreglo de ocho *planos de 1 bit*, cada uno de los cuales puede ser codificado longitudinalmente. Para imágenes monocromáticas a 8 bits, se alcanzan rangos de compresión de 1.5 a 2. Este método es muy sensitivo a los errores de canal a menos que los planos binarios sean cuidadosamente protegidos. Se debe utilizar una técnica eficiente de codificación para las "diferencias" entre imágenes, las que toman ventaja del hecho de que para pequeñas diferencias la tarea se facilita más que para las grandes diferencias. Cada plano se codifica por algún código bidimensional binario, y si se considera que se tiene un cuantificador de L niveles, se requerirán de $1+\text{Log}_2 L$ bits para cada valor de error de predicción.

En el ANEXO 2 se hallan ejemplos ilustrativos de codificación de una imagen por medio de la técnica MIC-Ley A (Anexo 2a), y por medio de la técnica de codificación de Huffman (Anexo 2b).

2.1.2 CODIFICACION PREDICTIVA:

La filosofía que se subraya en las técnicas predictivas es la de remover redundancia mutua entre pixeles sucesivos y codificar solamente la nueva información. Considérese una secuencia muestreada $u(m)$, la cual se ha codificado hasta $m=n-1$ y sea $u'(n-1)$, $u'(n-2)$,..... la secuencia de valores reproducidos (DECODIFICADOS). En $m=n$, cuando $u(n)$ llega, una cantidad $u''(n)$, una estimación de $u(n)$, es predicha de las muestras decodificadas anteriores $u'(n-1)$, $u'(n-2)$,....., así, [27]

$$u''(n) = \Phi(u'(n-1), u'(n-2), \dots) \quad (2.8)$$

donde $\Phi(\dots)$ denota la regla de predicción. Esto es suficiente para codificar el *error de predicción*: [27]

$$e(n) = u(n) - u''(n) \quad (2.9)$$

Si $e'(n)$ es la cuantificación de $e(n)$, entonces el valor reproducido de $u(n)$ es: [27]

$$u'(n) = u''(n) + e'(n) \quad (2.10)$$

el proceso de codificación continúa recursivamente de esta manera. Para una codificación predictiva *unidimensional*, un *dato puntual* x_1 se puede predecir por: [14]

$$x(n)'' = \sum_{i=1}^N a(i) x'(n-i) \quad (2.11)$$

donde los coeficientes se escogen a fin de minimizar el error cuadrático medio: [14]

$$E \{ [x''(n) - x'(n)]^2 \} \quad (2.12)$$

N es el orden de la secuencia predictora. Las diferencias entre el valor actual y el valor predicho, se cuantifican y entonces se almacenan o transmiten.

A fin de obtener una *compresión preservadora de información* esta diferencia a veces se codifica utilizando el código Huffman. Para este código son necesarios los modelos de función de densidad de probabilidad para computar la probabilidad de las diferencias y para implementar a dicho código. A menudo las diferencias son transmitidas utilizando una codificación "cruda" (MIC) para evitar las complejidades de la codificación Huffman. Mientras este procedimiento es efectivo, no garantiza una preservación de la información.

2.1.2.1 MIC Diferencial (MICD):

La técnica de codificación antes descrita se la conoce como *Modulación por Impulsos Codificados Diferencial*, o *MICD*.

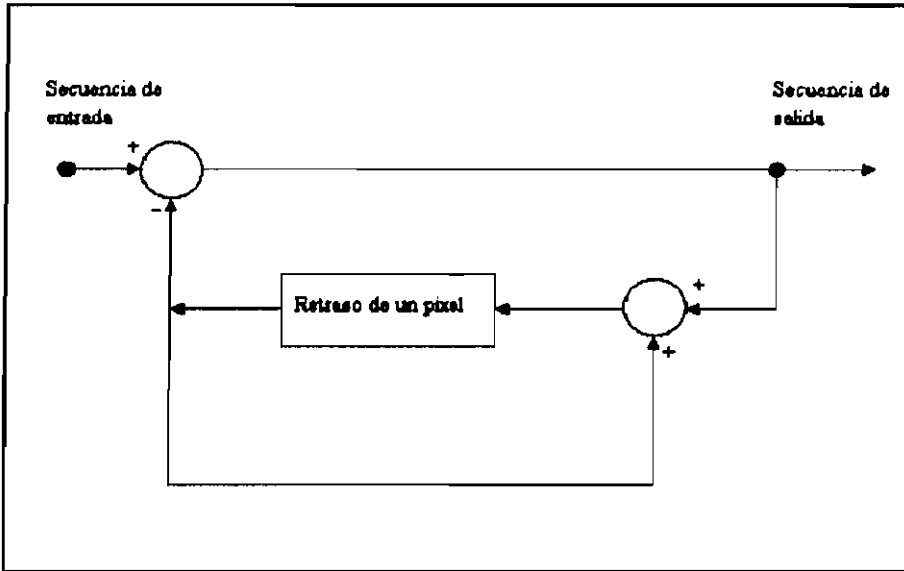


Figura 2.2 Codificación MIC Diferencial.

Esta técnica permite obtener un valor de predicción para cada pixel utilizando una combinación de valores de pixeles de localidades vecinas. A esta, se la conoce como *predicción espacial* debido a que aprovecha de la redundancia espacial de una imagen normal.

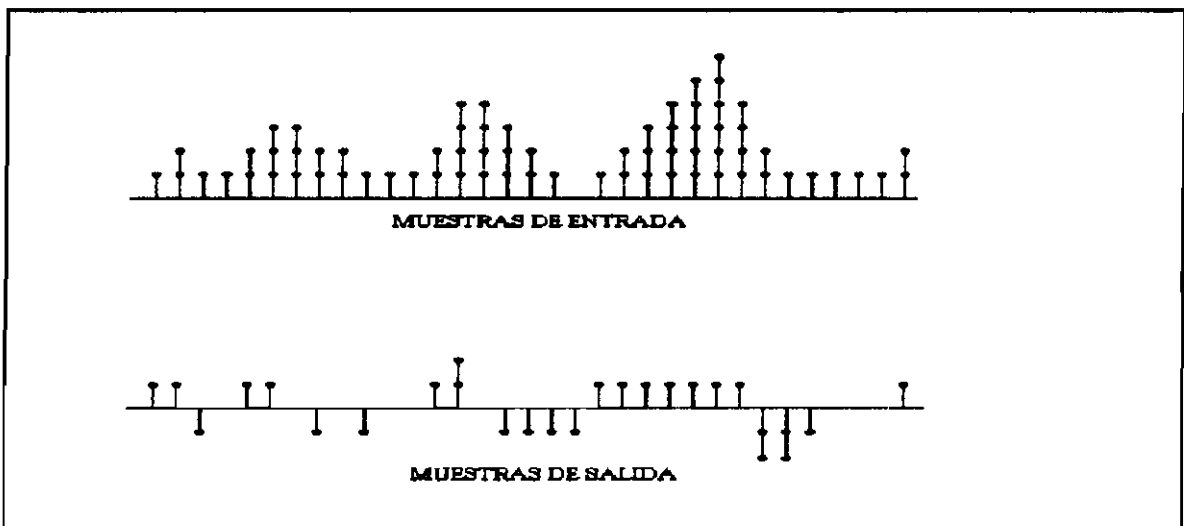


Figura 2.3 Resultados de una codificación MICD.

Los valores actuales de los pixeles de la señal de video entrante son comparados para predecir sus valores futuros.

El resultado de esta comparación constituye el *error de predicción*. La Figura 2.3 muestra los errores de predicción, la comparación del valor del pixel predicho y del valor actual del pixel entrante, puede ser codificado más eficientemente que la señal de entrada. La técnica CLV reduce la tasa binaria (bits/muestra) de la secuencia a la salida.

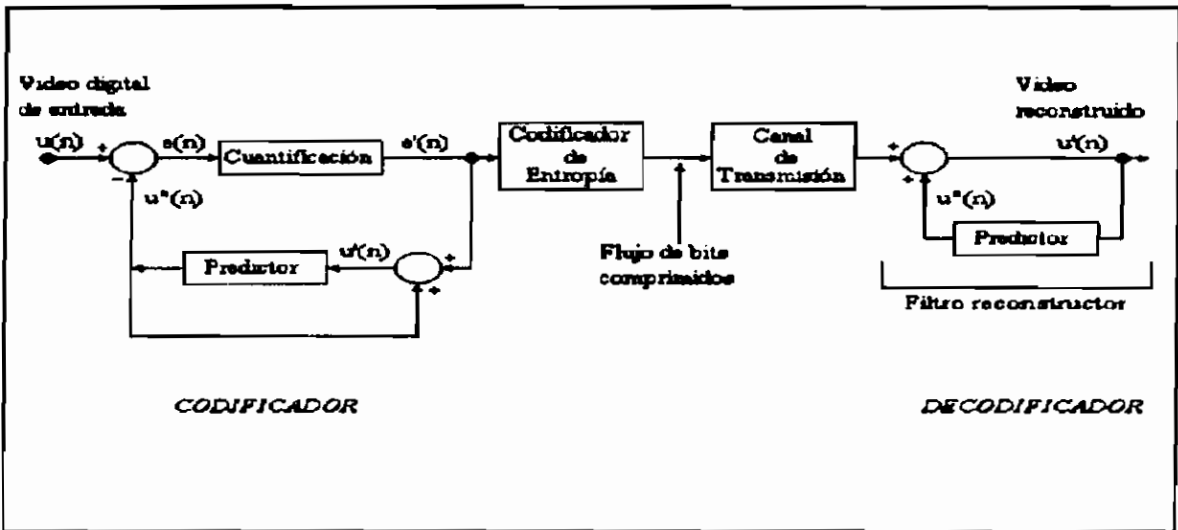


Figura 2.4 Comunicación con MIC Diferencial.

En la figura 2.3 se ilustra además que el rango de amplitud de la señal de error es más pequeño que el rango de la señal de entrada. La Figura 2.4 muestra el diagrama de una sistema MICD que emplea la cuantificación de los errores de predicción de los pixeles, se pueden observar dos resultados de cuantificación que se hallan en la Figura 2.5.

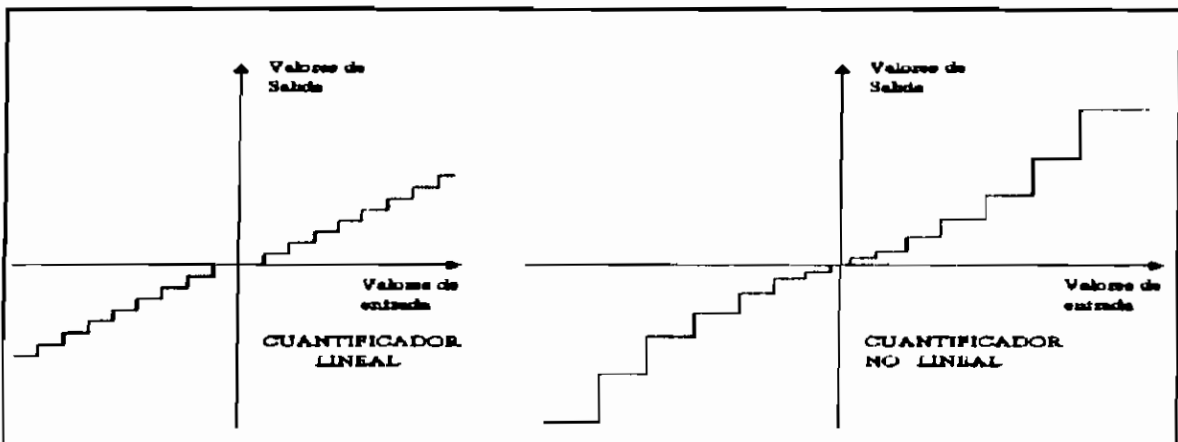


Figura 2.5 Dos tipos de cuantificadores.

En la Figura 2.5 se describen gráficamente un cuantificador lineal, tal como el usado en una conversión directa A/D, y un cuantificador no-lineal. Una mayor compresión puede alcanzarse, limitando el rango dinámico de la señal de error de predicción. Este proceso, introduce un ruido de cuantificación irreversible para la señal digital. En áreas de una imagen con alta actividad en el espacio o en el tiempo, son posibles grandes errores de predicción. Los valores grandes de estos errores son cuantificados toscamente debido a que ellos rara vez son esperados. En algunos casos, estos valores toscos de cuantificación llevarían a la salida del cuantificador no lineal a la saturación. La *sobrecarga temporal o espacial* resultante puede producir defectos de codificación altamente visibles: detalles como bordes temblorosos o desvanecimiento de ellos en la imagen (sobrecarga espacial), excesivo desvanecimiento o movimiento de objetos de la imagen (sobrecarga temporal), o efectos como ruido de distorsión "sal y pimienta" en áreas amplias de la imagen. Durante los años setenta y ochenta, las técnicas de compresión de video MICD fueron estudiadas extensamente para la transmisión de video-teléfono y televisión. Estos estudios permitieron un avance considerable en la reducción de redundancias no solo espaciales sino temporales en la señal de video.

2.1.2.2 Modulación Delta (MD):

La modulación delta (MD) es la más simple de las codificaciones predictivas. Esta utiliza una función de retardo escalón como un predictor y a la vez cuantificador a 1 bit, dando a la señal una representación de 1 bit. Así:[27]

$$u''(n) = u'(n-1); \quad e(n) = u(n) - u'(n-1) \quad (2.13)$$

Un sistema MD práctico, que no requiere de muestreo de la señal de entrada está representado en la Figura 2.6. El predictor integra la salida del cuantificador, la cual es una secuencia de pulsos binarios. El receptor es simplemente un integrador.

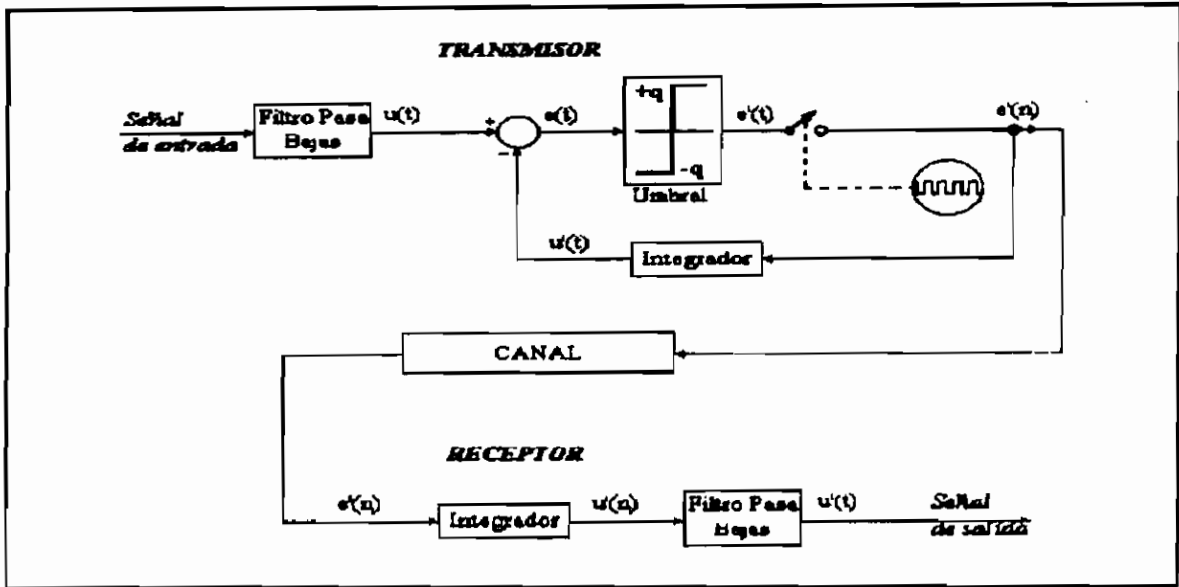


Figura 2.6 Comunicación con Modulación Delta.

La Figura 2.7 muestra a las típicas señales de entrada-salida de un modulador delta. Las principales limitaciones de la modulación delta son:

- Sobrecarga de pendiente.
- Ruido de granulación.
- Inestabilidad para canales con error.

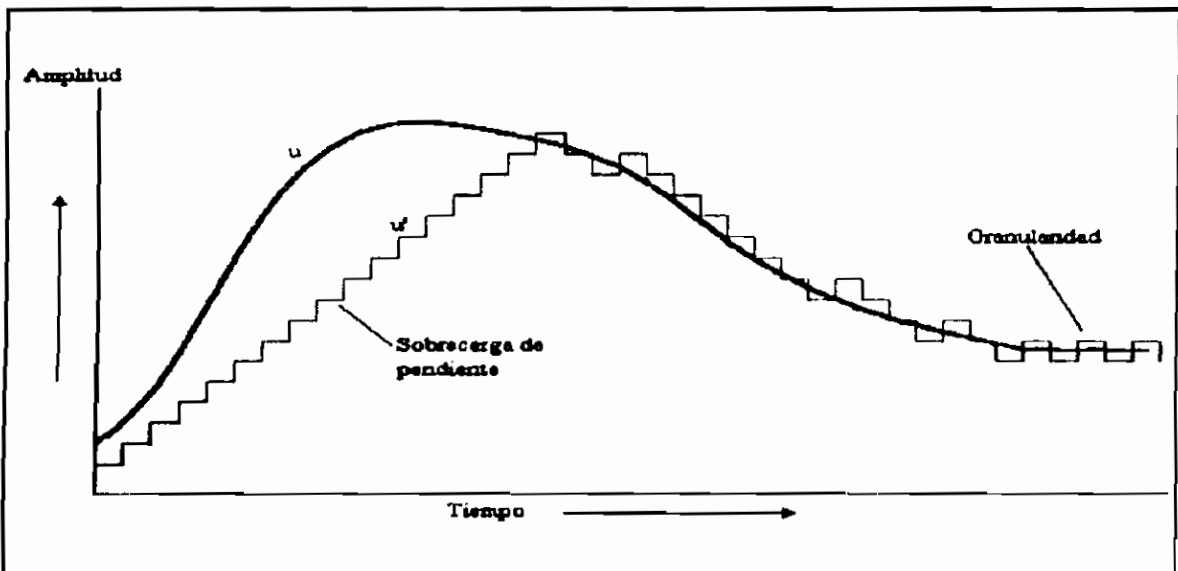


Figura 2.7 Defectos de la Modulación Delta.

La sobrecarga de pendiente ocurre siempre que existan grandes separaciones entre puntos (discontinuidades) en la señal, a lo cual el cuantificador puede responder solo dando varios pasos de diferenciación (deltas). Por otro lado *el ruido de granulación* es la naturaleza misma de la salida cuando la señal de entrada es casi constante. Así las imágenes que se tratan por este método de codificación pueden presentar el efecto de borrosidad debido a la sobrecarga por pendiente cerca de los bordes y el efecto de granularidad en los fondos de niveles constantes de gris.

Estos dos tipos de errores se pueden compensar hasta cierto punto por un filtrado pasa-bajos de las señales tanto de entrada como de salida. A la sobrecarga de pendiente también se la puede reducir al incrementar la tasa de muestreo, la cual reducirá las diferencias interpixeles. Sin embargo, a mayor tasa de muestreo se obtendrá una menor razón de compresión. Una alternativa para reducir el ruido por granularidad mientras se mantiene la simplicidad del modulador es tender al denominado *modulador delta a tres estados*. La ventaja es que un gran número de pixeles (del 65% al 85% del total) se encuentran en el *estado 0*, o, *nivel 0*, mientras que el resto de pixeles se halla en los estados *+1* o *-1*. La codificación Huffman de los tres estados o la codificación longitudinal de los estados 0 con un código de 2 bits para los otros estados permiten tener codificaciones de alrededor de 1 bit por pixel.

El filtro de reconstrucción—que es un simple integrador—, es inestable. Por lo tanto en presencia de *canales con error*, a la salida del receptor se puede acumular gran cantidad de error la cual se puede estabilizar atenuando la salida del predictor por una constante positiva:[27]

$$p < 1 \quad (2.14)$$

Esto sin embargo significa que no se mantendrá la simplicidad en la realización del sistema como la de la

Figura 2.6. Para la modulación delta de imágenes, la señal se presenta línea por línea y no se toma ventaja de la correlación bidimensional existente en los datos.

2.1.2.3 MICD línea por línea:

En este método cada línea barrida de la imagen se codifica independientemente por la técnica MICD. Generalmente se utiliza una representación *auto-regresiva* (AR) de orden N que a menudo se utiliza en la teoría de filtraje y la estimación lineal:[27]

$$u(n) - \sum_{k=1}^N a(k) u(n-k) = e(n) \quad (2.15)$$

donde por definición:[27]

$$E\{[e(n)]^2\} = \beta^2 \quad (2.16)$$

donde, β^2 es la varianza (potencia) del error de predicción.

Las ecuaciones del MICD son:[27]

Predictor:

$$u''(n) = \sum_{k=1}^N a(k) u'(n-k) \quad (2.17)$$

Entrada al cuantificador:

$$e(n) = u(n) - u''(n) \quad (2.18)$$

Salida del cuantificador:

$$e'(n) \quad (2.19)$$

Filtro reconstructor (salida reproducida):

$$u'(n) = u''(n) + e'(n) \quad (2.20)$$

2.1.2.4 MICD bidimensional:

Para el caso bidimensional, el pretérito de un pixel de una imagen se lo define de una manera obvia. Así si x_{ij} es un pixel donde i indica la posición-fila (línea) en la imagen y j indica la columna, un estimado lineal se lo halla de la siguiente relación:[14]

$$x(m, n)'' = \sum_{i=1}^{M'} a(0, i) x'(m, n-i) + \sum_{j=1}^M \sum_{i=1}^{M'} a(j, i) x'(m-j, n-i) \quad (2.21)$$

Para este caso más general, encontrar coeficientes óptimos es muy dificultoso a menos que se asuman particulares correlaciones espaciales. Mientras que la búsqueda de valores apropiados para M , N , o M' puede no ser posible, la exactitud de la predicción basada en unos razonables y pequeños valores de estos parámetros podría ser la adecuada para obtener una buena compresión. Para imágenes comunes, se ha visto que el incremento de los órdenes de los predictores N , M , y M' no dan reducciones muy apreciables en la varianza del error de predicción. Así para imágenes línea por línea barrida, es suficiente considerar predictores de la forma:[27]

$$u'(m, n) = a_1 u(m-1, n) + a_2 u(m, n-1) + a_3 u(m-1, n-1) + a_4 u(m-1, n+1) \quad (2.22)$$

En esta relación, a_1 , a_2 , a_3 , a_4 , y β^2 se obtienen al resolver las ecuaciones lineales:[27]

$$\begin{aligned} r(1, 0) &= a_1 r(0, 0) + a_2 r(1, -1) + a_3 r(0, 1) + a_4 r(0, 1) \\ r(0, 1) &= a_1 r(1, -1) + a_2 r(0, 0) + a_3 r(1, 0) + a_4 r(1, -2) \\ r(1, 1) &= a_1 r(0, 1) + a_2 r(1, 0) + a_3 r(0, 0) + a_4 r(0, 2) \\ r(1, -1) &= a_1 r(0, 1) + a_2 r(1, -2) + a_3 r(0, 2) + a_4 r(0, 0) \end{aligned} \quad (2.23)$$

$$\beta^2 = E \{ [e(m, n)]^2 \}$$

$$\beta^2 = r(0, 0) - a_1 r(1, 0) - a_2 r(0, 1) - a_3 r(1, 1) - a_4 r(1, -1)$$

Donde $r(k, l)$ es la función covarianza. Estudios prácticos han demostrado que la solución del sistema (2.23) puede producir un modelo inestable. Esto significa que mientras a la varianza del error de predicción se la minimiza, el filtro reconstructor puede ser inestable causando que cualquier error en el canal de transmisión se pueda amplificar mucho en la etapa receptora. Por tanto, la estabilidad del predictor debe ser probada y, si no es estable, tiene que ser modificada (incrementando el orden del predictor). Afortunadamente, para imágenes digitales monocromáticas comu-

nes (tales como las imágenes de televisión), este problema se lo halla muy raramente. Las ecuaciones del MICD bidimensional son: [27]

Predictor:

$$u''(m, n) = a_1 u'(m-1, n) + a_2 u'(m, n-1) + a_3 u'(m-1, n-1) + a_4 u'(m-1, n+1) \quad (2.24)$$

Entrada al cuantificador:

$$e(m, n) = u(m, n) - u''(m, n) \quad (2.25)$$

Salida del cuantificador:

$$e'(m, n) \quad (2.26)$$

Filtro reconstructor (salida reproducida):

$$u'(m, n) = u''(m, n) + e'(m, n) \quad (2.27)$$

2.1.2.5 Codificación Adaptiva:

La eficacia de MICD puede ser mejorada al adaptar las características del cuantificador y del predictor a las variaciones en la estadística local de la imagen digital. Las técnicas adaptivas utilizan un rango de características de los predictores y/o cuantificadores de los cuales un "grupo óptimo" se selecciona de acuerdo a la propiedades locales de la imagen. Para eliminar la sobreestimación debido al proceso de adaptación, se utilizan píxeles codificados previamente para determinar el modo de operación del codificador adaptivo.

En ausencia de errores de transmisión, el receptor puede realizar la misma secuencia de decisiones que se realizan en la transmisión. Los predictores adaptivos se diseñan generalmente para mejorar la calidad subjetiva de la imagen, especialmente en los bordes. Una técnica popular es utilizar varios predictores, cada uno de los cuales permite determinar si la imagen está altamente correlacionada en una cierta dirección. La dirección de máxima correlación es calculada de píxeles codificados previamente y de aquí se escoge el predictor correspondiente.

2.1.2.6 Técnicas inter-trama / intra-trama:

Las teleconferencias, la radiodifusión, y muchas imágenes médicas se reciben como secuencias de dos tramas de imágenes bidimensionales. Las técnicas de codificación *INTER-TRAMA* explotan la redundancia que existe entre tramas sucesivas (Redundancia Temporal), mientras que las técnicas *INTRA-TRAMA* explotan la redundancia interna de una sola trama (Redundancia Espacial).

2.1.2.6.1 Repetición de tramas:

Más allá de los métodos de entrelazado de líneas verticales y/o horizontales discutidos anteriormente, un simple método de compresión intertrama es submuestrear y repetir tramas de cuadros entrelazados. Esto, sin embargo, no produce imágenes móviles de buena calidad. Una alternativa es un abastecimiento selectivo, donde las tramas se transmiten a una velocidad reducida de acuerdo a un algoritmo de actualización fijo y predeterminado. En el receptor, cualquier dato no actualizado se refresca con datos de la trama previamente almacenada en la memoria de tramas. Este método se lo utiliza en áreas de bajo movimiento.

2.1.2.6.2 Cambio de resolución:

La respuesta del sistema visual humano es pobre para escenas dinámicas que simultáneamente contienen altas frecuencias espaciales (Ciclos/Grado) y temporales (Ciclos/Segundo). Así, las áreas de una escena que cambian rápidamente se pueden representar con amplitud y resolución espacial reducidas cuando se comparan con las áreas estacionarias.

Esto permite el cambio de resolución espacial con resolución temporal y se lo puede utilizar para producir imágenes de buena calidad a razones de 2 a 2.5 bits por pixel. Uno de tales métodos segmenta la imagen en áreas estacionarias introduciendo el valor de la señal diferencia de tramas.

En áreas estacionarias, la diferencia de trama se transmite para cualquier otro pixel y los pixeles restantes se repiten desde la trama previa. La principal distorsión ocurre en el movimiento de bordes agudos con velocidad moderada.

2.1.2.6.3 Abastecimiento condicional:

Esta técnica está basada en la detección y codificación de las áreas en movimiento, las que son abastecidas de trama a trama. Sea que $u(m,n,i)$ denota al pixel en la localidad (m,n) en la trama i . La señal diferencia intertrama es[27]:

$$e(m,n,i) = u(m,n,i) - u'(m,n,i-1) \quad (2.28)$$

donde $u'(m,n,i-1)$ es el valor reproducido de $u(m,n,i)$ en la $(i-1)$ -ésima trama. Siempre que la magnitud de $e(m,n,i)$ exceda un umbral δ , esta es cuantificada y codificada para su transmisión. En el receptor, un pixel se reconstruye por la repetición de la localidad del pixel de la trama previa si este vino de una area estacionaria, o si es reabastecida de la señal diferencia decodificada si este pixel proviene de un área móvil, esto es:[27]

$$u'(m,n,i) = \begin{cases} u'(m,n,i-1) + e(m,n,i); & \text{si } |e(m,n,i)| > \delta \\ u'(m,n,i-1); & \text{en otro caso} \end{cases} \quad (2.29)$$

Para la transmisión, se generan palabras código que representan a los valores cuantificados y sus direcciones. Se ignoran puntos aislados o muy pequeños grupos de áreas móviles para hacer del esquema de codificación de direcciones más eficiente.

2.1.2.6.4 Codificación predictiva con compensación en el movimiento:

En principio, si la trayectoria de movimiento de cada pixel pudiese ser medida, entonces solo la información de la trama inicial y la trayectoria serían necesarias codificar.

Para reproducir las imágenes se podría solamente propagar cada pixel a lo largo de su trayectoria. En la práctica, el movimiento de objetos en la escena puede ser aproximado por pequeños desplazamientos desde una trama a otra trama. El vector desplazamiento es utilizado para dirigir al predictor compensador de movimiento entre las tramas. La eficacia del codificador compensador de movimiento depende de una exactitud, velocidad, y robustez (con respecto al ruido) del estimador.

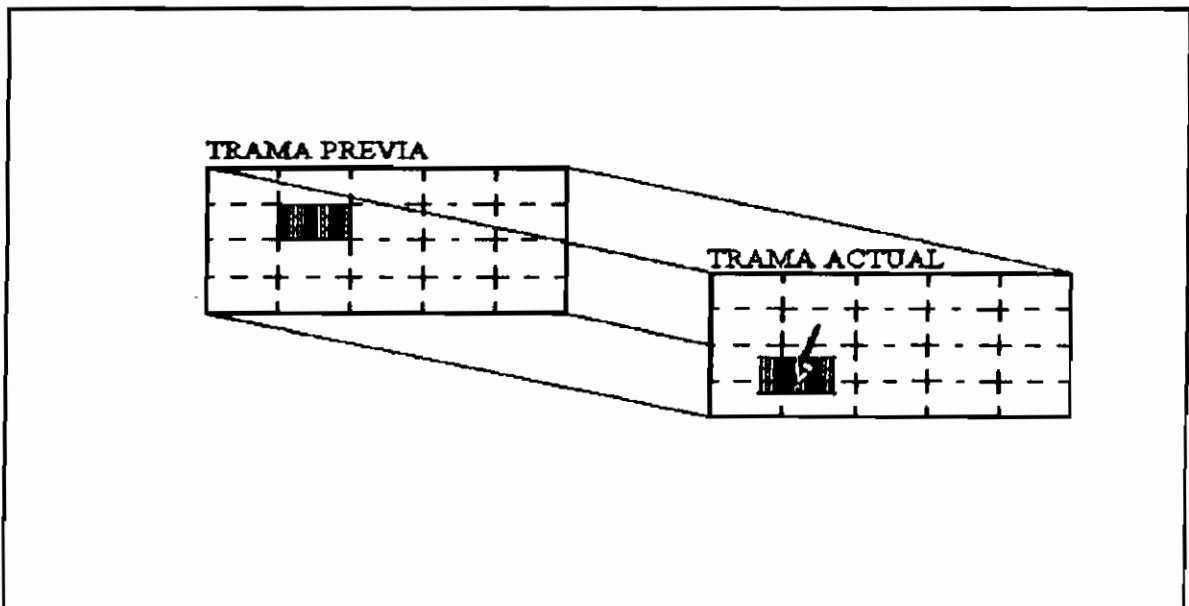


Figura 2.8 La compensación en el movimiento.

Cuando un área de imagen no muestra movimiento, el mejor predictor para un pixel o bloque de pixeles es el mismo grupo de pixeles en el campo o trama previo. En este caso el error de predicción será cero (o cercanamente cero si está presente ruido en la imagen), y una adecuada técnica de codificación de longitud variable se usará para reducir la velocidad binaria. Si existe movimiento en una área de imagen, la simple predicción de previas tramas y campos será incorrecta, y un valor alto de error de predicción incrementará la velocidad binaria de salida. Este problema puede ser solucionado efectivamente al incorporar la *compensación al movimiento*: una búsqueda dentro de una región confinada del campo o la trama previos del pixel o de los pixeles que vienen a conformar la presente señal de entrada.

La información digital transmitida o almacenada, consistirá ahora de valores de error de predicción codificados, más los vectores desplazamiento que localizan al pixel o grupo de pixeles en el campo o trama previos. La compensación al movimiento no se emplea para cada pixel en una imagen; esto demandaría muchos bits adicionales para el vector desplazamiento.

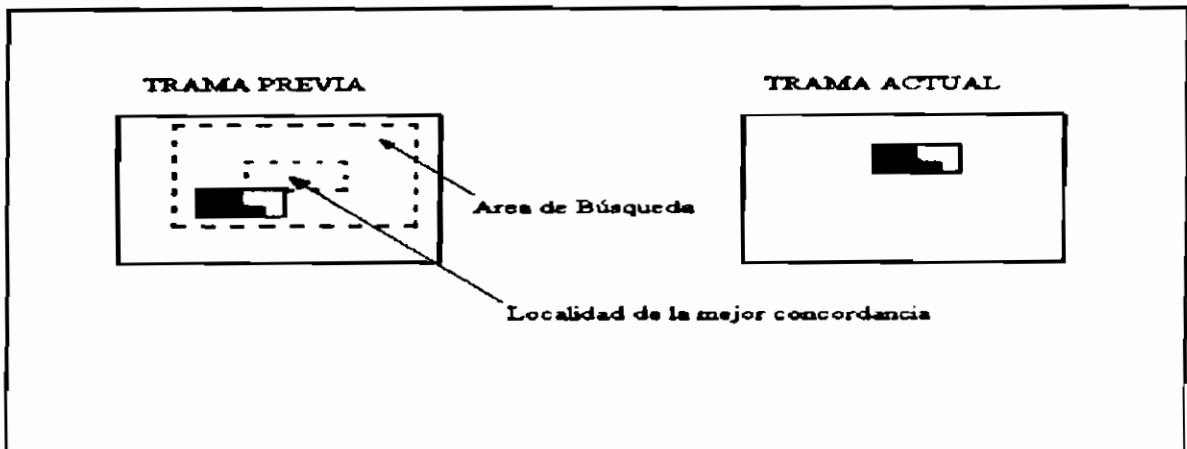


Figura 2.9 La búsqueda de la concordancia de bloques.

Una mejor aproximación es utilizar *la compensación al movimiento* de un bloque de píxeles. En la mayoría de casos, esto es lo suficientemente preciso para asumir que todos los píxeles dentro de un bloque experimentará el mismo desplazamiento de una trama a la siguiente. La técnica más común para calcular el vector desplazamiento de un bloque de píxeles es el método de *concordancia de bloques*. Este compara el bloque de píxeles de la imagen presente con bloques desplazados en el campo previo o trama para encontrar la concordancia inmediata. La Figura 2.8 muestra como trabaja la codificación con compensación de movimiento. El área de búsqueda para el proceso de concordancia de bloques se determina al suponer la magnitud de los desplazamientos horizontal y vertical que pueden ocurrir entre campos o tramas en una secuencia de video. Existe además un cierto número de criterios que se pueden usar para definir la "concordancia inmediata". Una técnica común selecciona el vector desplazamiento que resulta en el mínimo valor promedio de la diferencia al cuadrado entre los píxeles de los dos bloques.

El estimar los vectores de movimiento para todos los bloques de píxeles en una imagen, requiere de una gran cantidad de cómputo¹. Afortunadamente este proceso se lo hace en el momento de la codificación (o compresión); la parte de la decodificación simplemente utiliza vectores de movimiento recibidos para compensar las predicciones inter-cuadro/campo cuando la imagen es reconstruida. En la codificación con compensación de movimiento usando la técnica de concordancia de bloques, un bloque de píxeles de una trama previa se lo usa para predecir el valor del bloque de píxeles en la presente trama (Figura 2.9). El error de predicción resultante es luego comprimido y almacenado o transmitido como datos.

En una área de imagen móvil, el bloque de píxeles de la trama previa aparecerá desplazado una posición. El método de la concordancia de bloques busca la trama previa para encontrar un bloque de píxeles similar; el vector desplazamiento se lo calcula para cada bloque en la imagen. El bloque de error de predicción (diferencia entre los bloques entrantes y predichos) tiene un bajo contenido de información, particularmente si se aplica la técnica de compensación de movimiento.

Las correlaciones espaciales de estos bloques de error de predicción presentan nuevas oportunidades para una mayor compresión. En este caso, se aplica el proceso de la Transformada Discreta Coseno para "píxeles de error" que resultarán en "superficies con errores de predicción" producto del proceso de predicción. De nuevo el bloque transformado de coeficientes transformados es cuantificado y codificado para completar la fase final en la compresión de datos.

(1) ORIBSEN J. M., "Motion Estimation for Digital Video", PhD Thesis - Delft University of Technology, 1982.

2.1.2.6.5 Codificación adaptiva predictiva:

Hasta ahora, se han discutido varios puntos de vista teóricos concernientes al modelaje, eficacia, y estabilidad de predicciones lineales uni-dimensionales aplicadas a señales de imágenes monocromáticas. Siguiendo un procedimiento similar, se aplica la predicción lineal bi-dimensional a la codificación predictiva de imágenes todavía monocromáticas en una intertrama MICD-Adaptiva, con predicción adaptiva y cuantificación adaptiva.

(a) Predicción Adaptiva: El sistema estudiado se muestra en la Figura 2.10. El corazón de este sistema es un sistema básico MICD. Su predictor forma un estimado, $x''(m,n)$ de una muestra de imagen a ser codificada, $x(m,n)$, a partir de muestras pasadas reconstruidas en el receptor $x'(m,n)$. La diferencia entre $x(m,n)$ y su estimado $x''(m,n)$ es la señal diferencia $d(m,n)$, la que se cuantifica y codifica para su transmisión.

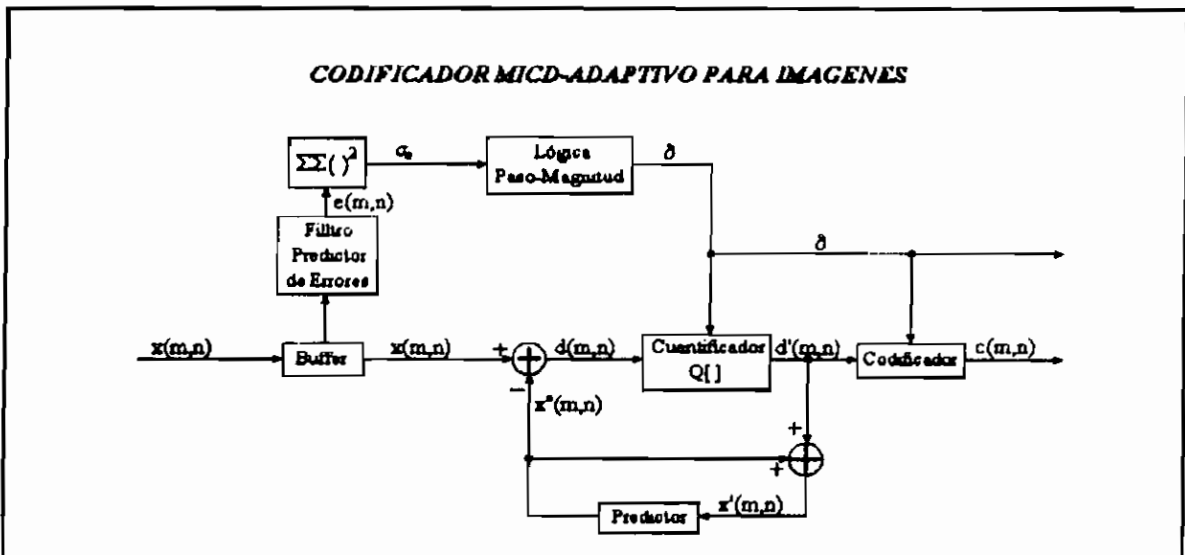


Figura 2.10 Codificador MICD Adaptivo de Imágenes.

Se debería notar que, aunque los coeficientes óptimos se obtienen de las muestras no cuantificadas $x(m,n)$, el predictor opera sobre las muestras reconstruidas $x'(m,n)$ (Véase la Figura 2.10). Sin embargo esto da como resultado una pequeña pérdida en la optimización cuando el error de cuantificación es pequeño; esto es: $x(m,n) \approx x'(m,n)$.

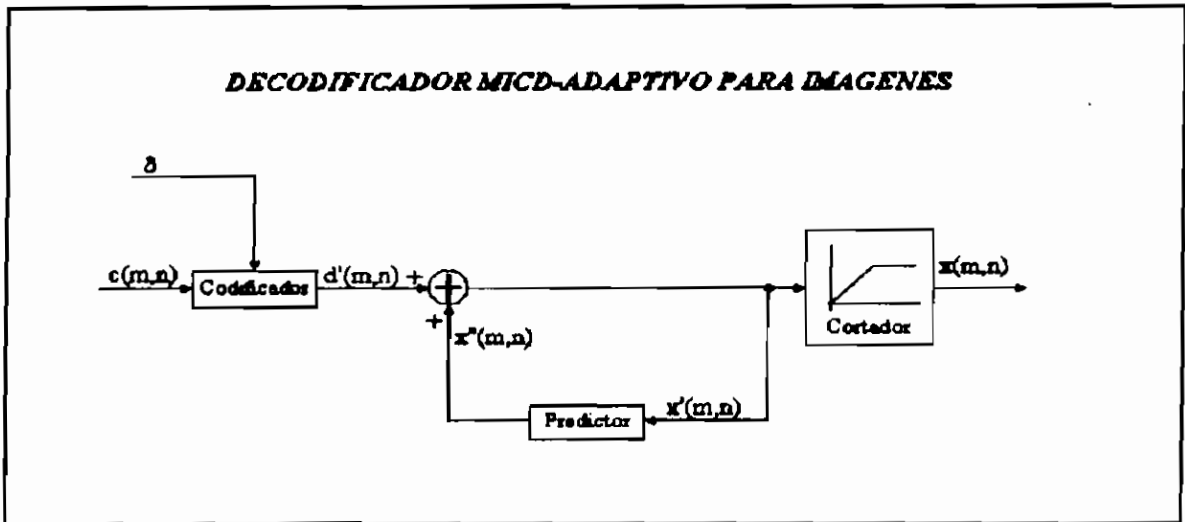


Figura 2.11 Decodificador MIC Adaptivo de Imágenes.

(b) **Cuantificación Predictiva:** A fin de alcanzar compresiones por debajo de 1 bit/píxel, la señal diferencia $d(m,n)$ se debe cuantificar a tres niveles. La característica de entrada-salida de este cuantificador de tres niveles está ilustrada en la Figura 2.11, y está dada por la ecuación:[9]

$$d(m,n) = \begin{cases} \delta, & d(m,n) \geq \theta \\ 0, & -\theta < d(m,n) < \theta \\ -\delta, & d(m,n) \leq -\theta \end{cases} \quad (2.30)$$

El umbral θ (Figura 2.12) determina el porcentaje de rango dinámico de la señal de entrada al cuantificador para asignarla al valor cero. El comportamiento del cuantificador se puede variar al variar θ proporcionalmente al tamaño de paso δ . Por ejemplo, si $\theta=0$, el cuantificador tiene solo dos niveles, mientras que $\theta=\delta/2$ corresponde a un cuantificador uniforme de tres niveles. Para $\theta>\delta/2$ se tendrá un cuantificador uniforme de más de tres niveles. A fin de alcanzar una tasa binaria para la señal diferencia de casi 1bit/píxel usando palabras código de longitud fija, es necesario utilizar un cuantificador a dos niveles. En contraste, un cuantificador a tres niveles ofrece la alternativa de tener el nivel medio de cero para pequeñas amplitudes de la señal diferencia más dos niveles laterales para el manejo de la porción de grandes amplitudes de la señal diferencia.

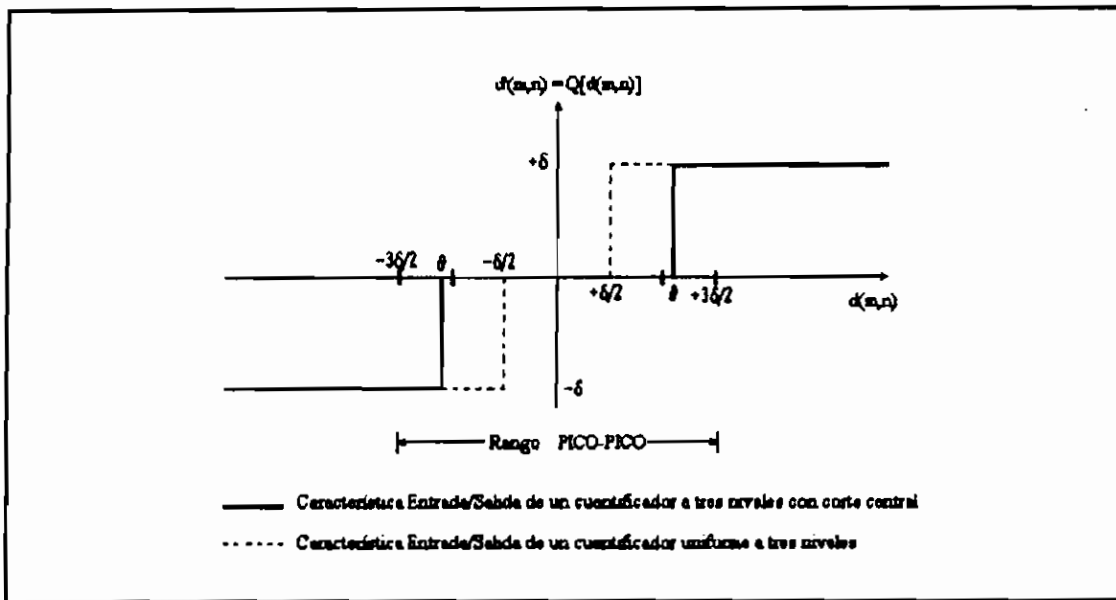


Figura 2.12 Cuantificador Predictivo.

La entropía de primer orden de la señal de salida de un cuantificador de dos niveles en la MICD-Adaptiva es muy cercana a 1bit/pixel. Sin embargo al incrementar el nivel de umbral θ del cuantificador a tres niveles, se produce una gran cantidad de valores 0 en la salida, y el valor de la entropía de la señal diferencia cuantificada puede hacerse significativamente menor a 1bit/pixel.

2.1.2.7 Otros métodos¹:

A bajas tasas binarias ($B=1$ bit/pixel) la eficacia de MICD se deteriora muy rápidamente. Una razón es que el predictor y el cuantificador, los cuales son diseñados independientemente, no tiene una gran operación en niveles cercanamente óptimos. Así las sucesivas entradas al cuantificador pueden tener una correlación significativa, y el predictor puede no ser suficientemente bueno.

(1) = J.W. MODESTINO & V. BHASKARAN, "Robust Two-Dimensional Tree Encoding of Images", IEEE Trans. Commun. COM-29, No. 12, Diciembre 1981.

= A.N. NETRAVALI, "On Quantizers for DPCM Coding of Picture Signals", IEEE Trans. Inform. Theory IT-23, Mayo 1977.

= L. H. ZETTERBERG, S. KRICSON, C. COUTURIER, "DPCM Picture Coding with Two-Dimensional Control of Adaptive Quantization", IEEE Trans. Commun. COM-32, No. 4, Abril 1984.

= H. M. HANG & J. W. WOODS, "Predictive Vector Quantization of Images", IEEE Trans. Commun., 1985.

Los dos métodos que pueden mejorar la eficacia de MICD son:

1. Codificación predictiva retardada.
2. Cuantificación vectorial predictiva.

En el primer método, un código-árbol se genera por el filtro de predicción excitado por niveles de cuantificación diferentes. Según los sucesivos píxeles son codificados, el predictor selecciona una *ruta* en el árbol (al contrario de un valor de bifurcación, como en MICD) a fin de minimizar el error cuadrático medio. Los retardos son introducidos en el predictor para facilitar el desarrollo de un árbol con suficientes rutas.

En el segundo método, las entradas sucesivas al cuantificador son dirigidas a un registro de desplazamiento, cuyo estado se lo utiliza para definir el valor de salida del cuantificador. Así la salida actual del cuantificador depende de sus previas salidas.

En el ANEXO 2 se hallan ejemplos ilustrativos de codificación de una imagen de video por medio de la técnica MICD (Anexo 2c) y por medio de la técnica MD (Anexo 2d).

2.1.3 CODIFICACION POR TRANSFORMADA:

La codificación por transformada, también llamada *cuantificación bloque* (debido a que la cuantificación es una parte integral de la codificación por transformada y su ejecución se realiza sobre un bloque de $n \times n$ píxeles), es una alternativa a la codificación predictiva. Un bloque de datos es unilateralmente transformado a fin de que una gran fracción de su energía total sea empaquetada en relativamente pocos coeficientes de transformación, los que se cuantifican independientemente. El *Codificador por Transformada Óptimo* está definido como aquel que minimiza la distorsión cuadrática media de los datos reproducidos para un número dado de bits del total. Una transformada que realiza esto es la transformada KL (Karhunen Loève).

Cuatro transformadas son las que se han implementado en los sistemas actuales de compresión. Estas son:

- (a) Karhunen-Loève (KL o Componente principal).
- (b) Fourier.
- (c) Hadamard.
- (d) Haar.

(a) Transformada de KARHUNEN-LOEVE:

La transformada de Karhunen-Loève o *transformada de componente principal* ha sido considerada una transformación óptima, y por esta razón muchas otras transformaciones han sido comparadas con aquella en lo que se refiere a su desempeño. Hotelling(1933) fue el primero en derivar y publicar esta transformación bajo el nombre de "Principal Component" (Componente Principal). Esta es una transformación discreta. Algunos años más tarde fue desarrollada la versión continua por Karhunen y Loève(1947, 1948). En la literatura, la transformada discreta se refiere a ambas, a la de componente principal y a la de Karhunen y Loève(KL). En este trabajo se usará la designación de componente principal cuando se trate de señales muestreadas (no continuas en el tiempo). Para simplificar la descripción, se asume una imagen "explorada" que se convertirá en un arreglo de datos unidimensionales mediante un nuevo vector largo de las sucesivas líneas exploradas como sigue: Para una imagen con L líneas y L pixeles por línea, sea que $f(x, y_j)$ represente todos los L pixeles en la j -ésima línea, donde $j=1, 2, \dots, L$. Entonces:[40]

$$\{f(z)\} = [f(x, y_1), f(x, y_2), \dots, f(x, y_L)] \quad (2.31)$$

es el L^2 vector compuesto por todos los pixeles tomados en la secuencia normal de barrido de patrones, en otras palabras, la imagen total. Luego se define una matriz $[A]$, $L^2 \times L^2$ tal que los pixeles transformados se definen como:[40]

$$[F(w)] = [A] [f(z)] \quad (2.32)$$

Cada componente de $[F(w)]$ (a veces llamado un *coeficiente*) puede ser expresado como una combinación lineal de todos los pixeles originales:

$$F(w_k) = \sum_{i=1}^{L^2} f(z_i) A_{ik}, \quad k=1, 2, \dots, L^2 \quad (2.33)$$

Los pixeles originales pueden ser reconstruidos de los pixeles transformados (transformación inversa):

$$[f(z_i)] = [A]^t [F(w)] \quad (2.34)$$

y similarmente, cada pixel original puede ser expresado como una combinación lineal de todos los coeficientes, $F(w_k)$:

$$f(z_i) = \sum_{k=1}^{L^2} F(w_k) A_{ki}, \quad i=1, 2, \dots, L^2 \quad (2.35)$$

Lo ideal debería ser que en la transformación, $[A]$, produzca *coeficientes independientes*, $F(w_k)$. Sin embargo, se puede tener esto, con una transformación que produzca *coeficientes no correlacionados*. Semejante transformación es una matriz cuyas columnas son los vectores propios de la matriz covarianza de los pixeles originales. La matriz covarianza se define como:[40]

$$C_f = E\{([f(z)] - E[f(z)]) ([f(z)] - E[f(z)])^t\} \quad (2.36)$$

Escribiendo esta expresión para C_f con la simplificación de que $E\{[f(x)]\} = 0$ y la notación $f(z_1) = f_1$,

$$C_f = \begin{bmatrix} E(f_1^2) & E(f_1 f_2) & E(f_1 f_3) & \dots & E(f_1 f_{L^2}) \\ E(f_2 f_1) & E(f_2^2) & E(f_2 f_3) & \dots & E(f_2 f_{L^2}) \\ E(f_3 f_1) & E(f_3 f_2) & E(f_3^2) & \dots & E(f_3 f_{L^2}) \\ \dots & \dots & \dots & \dots & \dots \\ E(f_{L^2} f_1) & E(f_{L^2} f_2) & E(f_{L^2} f_3) & \dots & E(f_{L^2}^2) \end{bmatrix} \quad (2.37)$$

Ahora los valores esperados de los varios cuadrados y productos cruz en C_f se obtienen típicamente de productos promediados de pixeles con la misma separación espacial y

relativa localización de una *imagen grande* de la cual el vector L^2 dimensional $[f(z)]$ es una *subimagen* $L \times L$. Los vectores propios de C_x son las soluciones ϕ para la ecuación matricial:[40]

$$C_x \phi = \alpha \phi \quad (2.38)$$

donde α representa a los valores propios. Primero se resuelven los valores propios de la ecuación característica:[40]

$$\det[C_x - \alpha I] = 0 \quad (2.39)$$

donde I es la matriz unitaria. Luego se ordenan los α 's en orden decreciente tal que: $\alpha_1 \geq \alpha_2 \geq \alpha_3 \geq \dots \geq \alpha_{L^2}$ y se sustituye en $(C_x - \alpha I)\phi = 0$ para hallar los vectores propios. Cuando la matriz $[A]$ (cuyas columnas son las soluciones de ϕ) se aplica a $[f(z)]$, la covarianza de los coeficientes resultantes $F(w_k)$ es una matriz diagonal con elementos en la diagonal $\alpha_1 > \alpha_2 > \alpha_3 > \dots > \alpha_{L^2}$. Así, los coeficientes, $F(w_k)$, no son correlacionados por definición.

(b) **Transformada de FOURIER:**

La *Transformada de Fourier* es probablemente la transformación más común. En los sistemas de compresión de imágenes se referirá a una transformación de Fourier especial, denominada, la *Transformada Discreta de Fourier* (TDF). Para señales continuas, existen funciones seno y coseno que, cuando se adicionan juntas, dan una buena aproximación de la señal continua (analógica) original, en el caso de la TDF existe un arreglo discreto de funciones ortogonales,[40]

$$w_{k,m} = e^{-\frac{j2\pi km}{N}}, \quad (j = \sqrt{-1}) \quad (2.40)$$

Las que, cuando se suman y ponderan convenientemente, dan una buena aproximación a la secuencia N -muestreada original de entrada. Se expresan los coeficientes transformados en la siguiente forma: Si $X(m)$ es una secuencia de N valores finitos de muestras reales o complejas, y $m=0,1,2, \dots, N-1$, luego los coeficientes transformados son:[40]

$$C_x(k) = \frac{1}{N} \sum_{m=0}^{N-1} X(m) e^{-\frac{j2\pi km}{N}}, \quad k=0,1,2,\dots,N-1 \quad (2.41)$$

Por ejemplo, para $k=0$, [40]

$$C_x(0) = \frac{1}{N} \sum_{m=0}^{N-1} X(m) \quad (2.42)$$

a veces llamado el valor promedio. Del mismo modo, para $k=1$, [40]

$$C_x(1) = \frac{1}{N} \sum_{m=0}^{N-1} X(m) e^{-\frac{j2\pi m}{N}} \quad (2.43)$$

Como se puede ver de esta formulación, cada coeficiente transformado, $C_x(k)$, es una suma ponderada de las muestras originales divididas por el número de muestras, N . En forma matricial, la TDF se ve como: [40]

$$[C] = \frac{1}{N} [W] [X], \quad (W_{k,m} = e^{-\frac{j2\pi km}{N}}) \quad (2.44)$$

o, [40]

$$\begin{bmatrix} C(0) \\ C(1) \\ \dots \\ \dots \\ C(N-1) \end{bmatrix} = \frac{1}{N} \begin{bmatrix} W_{0,0} & W_{0,1} & \dots & W_{0,N-1} & X(0) \\ W_{1,0} & W_{1,1} & \dots & W_{1,N-1} & X(1) \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ W_{N-1,0} & W_{N-1,1} & \dots & W_{N-1,N-1} & X(N-1) \end{bmatrix} \quad (2.45)$$

La correspondiente TDF inversa es: [40]

$$X(m) = \sum_{k=0}^{N-1} C_x(k) e^{\frac{j2\pi km}{N}} \quad (2.46)$$

Una versión afín a la TDF es la *Transformada Discreta Coseno* (TDC). La TDF y la TDC se pueden calcular con un algoritmo rápido, llamado la *Transformada Rápida de Fourier* (TRF).

(c) Transformada de HADAMARD:

La Transformada de Hadamard usa una matriz fija de transformación que toma la forma de una onda cuadrada muestreada a dos niveles. Los dos niveles son +1 y -1, y la matriz 2x2 se ve como:[40]

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.47)$$

La transformada de Hadamard tiene la propiedad de que las transformaciones de gran orden, múltiplos de 2, pueden ser generadas por medio de transformaciones de menor orden:[40]

$$H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix} \quad (2.48)$$

Por ejemplo,[40]

$$H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad (2.49)$$

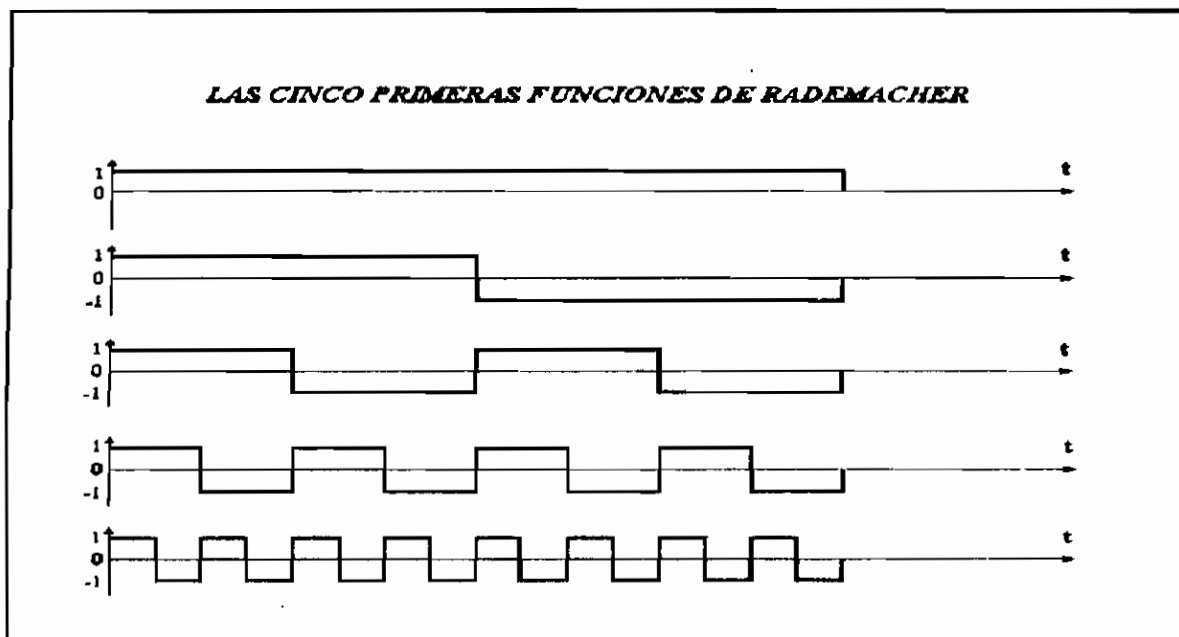


Figura 2.13 Funciones de Rademacher.

A fin de comprender como se usa a la transformada de Hadamard en la actualidad, se debe realizar una pequeña reseña. En 1893 Hadamard desarrolló la estructura matricial mostrada en (2.48). En 1922 Rademacher identificó a un arreglo de funciones rectangulares en el tiempo, un ejemplo de esto se muestra en la Figura 2.13. Estas funciones son ortonormales; esto es la integral del producto de dos funciones cualquiera es 0 y la integral del cuadrado de cualquier función es 1. En 1923, Walsh completó este arreglo de funciones al añadir funciones ortonormales entre las funciones de Rademacher, como se muestra en la Figura 2.14a (Las ocho primeras funciones de Walsh).

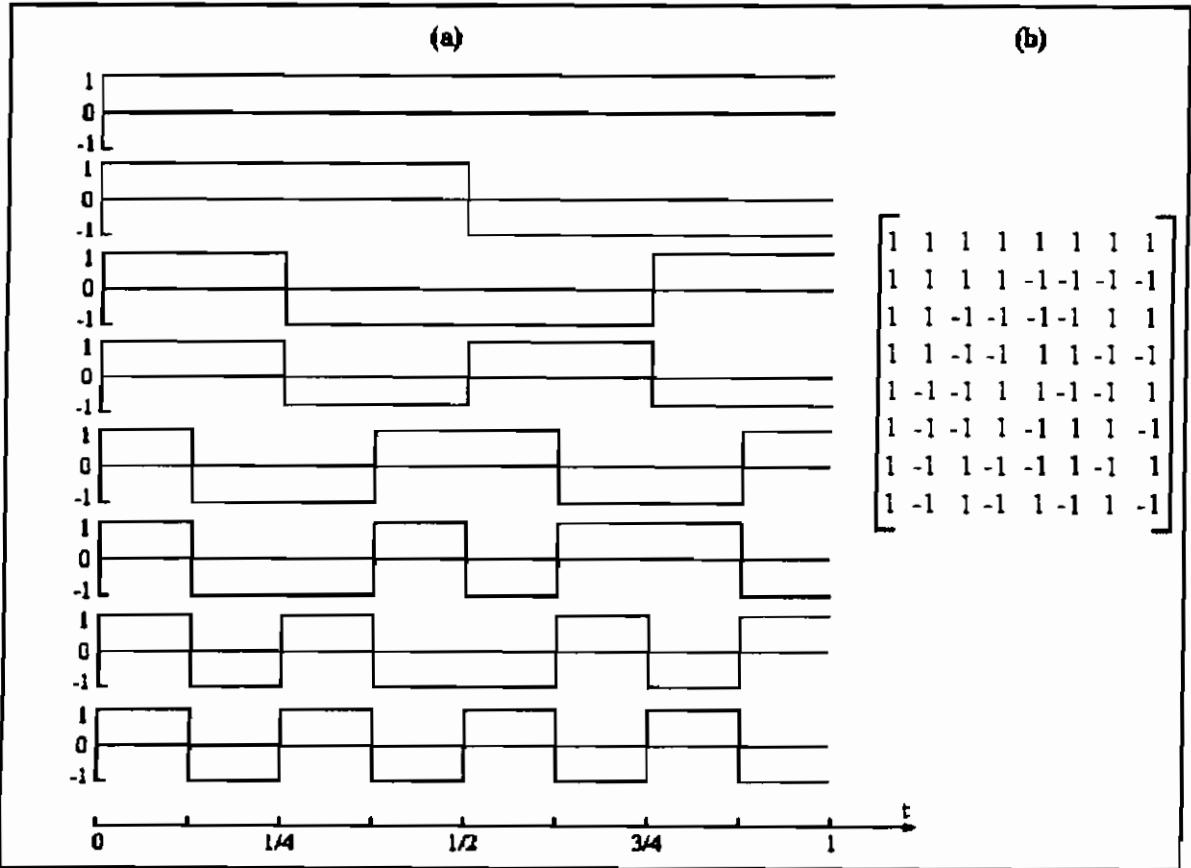


Figura 2.14 Funciones y Matriz de Walsh.

Si un sub arreglo de las funciones Walsh (comenzando en la menor) es muestreada a una frecuencia alta, suficiente como para detectar todos los cambios en el nivel, entonces se tiene una matriz muy parecida a una matriz de Hadamard, excepto por el ordenamiento de sus filas. Esto es

la matriz de Hadamard puede utilizarse para representar a las funciones de Walsh-reordenadas. La matriz Walsh para $N=8$ se muestra en la Figura 2.14b. Un término que se ha venido usando en conexión con estas matrices de dos niveles es la *secuencia*, la cual está definida como el número de cambios de signo en una fila dada. Este desarrollo histórico ha alcanzado lo que hoy se llama la *Transformación de Walsh-Hadamard* (TWH). Aunque a esta se la refiera en la literatura como la transformada de Hadamard, la matriz de transformación realmente representa a las funciones muestreadas de Walsh reordenadas por la forma matricial de Hadamard. En la literatura, a la TWH se hace referencia como la *transformada BIFORE* (REpresentación BInaria de FOurier). Como en el caso de la TDF, también existe la TWH rápida, la que puede calcularse utilizando una técnica similar a la utilizada por la TRF.

(d) **Transformada de HAAR:**

La *Transformada de Haar* utiliza las funciones ortonormales de Haar (Haar 1910), y una matriz 8×8 de transformación se puede ver como la Igualdad (2.50).

$$H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} & \sqrt{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix} \quad (2.50)$$

Existe una similitud entre la matriz de transformación de Haar y las matrices de transformación de Hadamard o de Walsh-Hadamard. Adicionalmente, las filas pueden visualizarse como ondas rectangulares (con amplitudes 0, 1, -1, multiplicadas por potencias de $\sqrt{2}$). Las primeras dos filas de la matriz de transformación de Haar son idénticas a las correspondientes de la matriz de Walsh (Figura 2.14b).

Las restantes filas contienen un cierto número de **ceros**, distintas a las filas de la matriz TWH. Así, puesto que la TWH tiene una propiedad *global* -esto es, cada coeficiente depende de *todas* las entradas muestreadas-la transformación de Haar tiene esta propiedad global solo en los dos primeros coeficientes. En forma matricial, la transformación de Haar de la secuencia de datos **X**, es:[40]

$$HAAR = \frac{1}{N} [H_N] [X] \quad (2.51)$$

y la transformación inversa es:[40]

$$X = [H_N]^t [HAAR] \quad (2.52)$$

2.1.3.1 Codificación adaptiva por transformada:

Se pueden identificar tres operaciones básicas:

- (a) Muestreo.
- (b) Transformación.
- (c) Cuantificación.

En ese orden, cada operación puede hacerse adaptiva, como se describe en las siguientes secciones.

(a) Muestreo adaptivo:

Con este la tasa de muestreo concatena por ella misma a las características del arreglo de datos. Donde exista alta actividad de los datos (tal como una escena con alta cantidad de detalles), se utiliza una alta tasa de muestreo. Por supuesto que, la tasa de barrido tendría que ser transmitida al receptor como información adicional (usualmente llamada "información lateral") a fin de mantener la sincronización. Aunque esta técnica es teóricamente muy simple, se la utiliza en pequeños sistemas de aplicación en la actualidad quizás debido a la necesidad de operar con patrones de sincronización cambiantes constantemente.

(b) Transformación adaptiva:

De las cuatro transformadas descritas en la sección 2.1.3, solo una es adecuada para hacerse adaptiva, y que es la transformación de componente principal (KL). Las otras tres son fijas e independientes de la estadística los datos fuente. En lugar de computar una nueva matriz de transformación para cada nuevo arreglo de datos, se desarrolla una simple matriz de transformación desde los vectores propios de una matriz covarianza "promedio", la cual será una concatenación limpia para la mayoría de los arreglos de datos esperados. Obviamente, algunos datos serán un arreglo muy pobre de transformaciones que mejor fijan ciertas clases de arreglos de datos.

(c) Cuantificación adaptiva:

En los esquemas de cuantificación adaptiva, la localización de bits se mantiene constante pero los niveles del cuantificador se ajustan de acuerdo a los cambios en las varianzas de los coeficientes transformados. Las varianzas en el dominio de la transformada se pueden estimar actualizando el parámetro estadístico del modelo de covarianza o por un promedio local de las magnitudes al cuadrado de las muestras en el dominio de la transformada.

2.1.3.2 Codificación multidimensional por transformada:

En muchas aplicaciones, (por ejemplo en imágenes multiespectrales, intertramas de imágenes de video, etc.), se tiene que trabajar con datos de tres o más dimensiones. Los esquemas de transformación son posibles para la compresión de tales datos extendiendo las ideas de las secciones anteriores (desde la teoría misma de la codificación por transformada). Una transformación tridimensional (separable) de una secuencia de dimensión $N_x M_x I$, $u(m, n, i)$ está definida como: [27]

$$v(k, l, j) = \sum_{m=0}^{M-1} \sum_{n=0}^{M-1} \sum_{i=0}^{I-1} u(m, n, i) a_M(k, m) a_M(l, n) a_I(j, i) \quad (2.53)$$

donde $0 \leq (k,m) \leq M-1$,
 $0 \leq (l,n) \leq N-1$,
 $0 \leq (j,i) \leq I-1$,

y, $\{a_{k,m}(k,m)\}$ son los elementos de una matriz $M \times M$ unitaria $A_{k,m}$.

Los coeficientes de transformación dados por (2.37) son simplemente el resultado de tomar la transformación A con respecto a cada índice y se requerirán $MNI - \log_2(MNI)$ operaciones para una transformación rápida.

Los requerimientos de almacenamiento para los datos es de MNI . Como antes, la aproximación práctica es particionar a los datos en pequeños bloques (como de $16 \times 16 \times 16$) y procesar cada bloque independientemente. El algoritmo de codificación después de la transformación es el mismo de antes excepto que se está trabajando con variables de triple-índice. También, la codificación híbrida adaptiva con compensación de movimiento se comporta mejor con codificación por transformada tridimensional. La siguiente tabla compara los varios esquemas de codificación por transformada en términos de su habilidad para comprimir. Las tasas de compresión listadas son tales que se alcanzan SNR's en el rango de 30dB a 36dB.

Método	Típica tasa de compresión
Uni-dimensional	2 - 4
Bi-dimensional	4 - 8
Bi-dimensional adaptiva	8 - 16
Tri-dimensional	8 - 16
Tri-dimensional adaptiva	16 - 32

2.1.3.3 La Transformada Coseno:

En el video digital, la información de un cuadro es muestreada y sus valores numéricos son almacenados, procesados o transmitidos. Estos valores numéricos pueden ser calculados de formas distintas. Un bloque de píxeles es transformado a un bloque de coeficientes de frecuencia a través del proceso de la *transformada discreta coseno (TDC)*.

La representación espacial de los píxeles se transforma en una representación de *estado frecuencial*; dando como resultado un bloque de coeficientes de frecuencia. El elemento en la parte superior izquierda del bloque de coeficientes corresponde al brillo promedio del bloque de píxeles, y es conocido como el *coeficiente DC*. Los otros coeficientes proveen información sobre el contenido de energía en otras frecuencias espaciales (horizontal, vertical, diagonal) del bloque original de píxeles. El sistema de la *Transformada Discreta Coseno*, o TDC, manipula un bloque de píxeles al cambiar sus valores a coeficientes de frecuencia. Las Figuras 2.15, 2.16, 2.17 y 2.18 muestran como un bloque de 8x8 píxeles se transforma a un bloque de 8x8 de coeficientes.

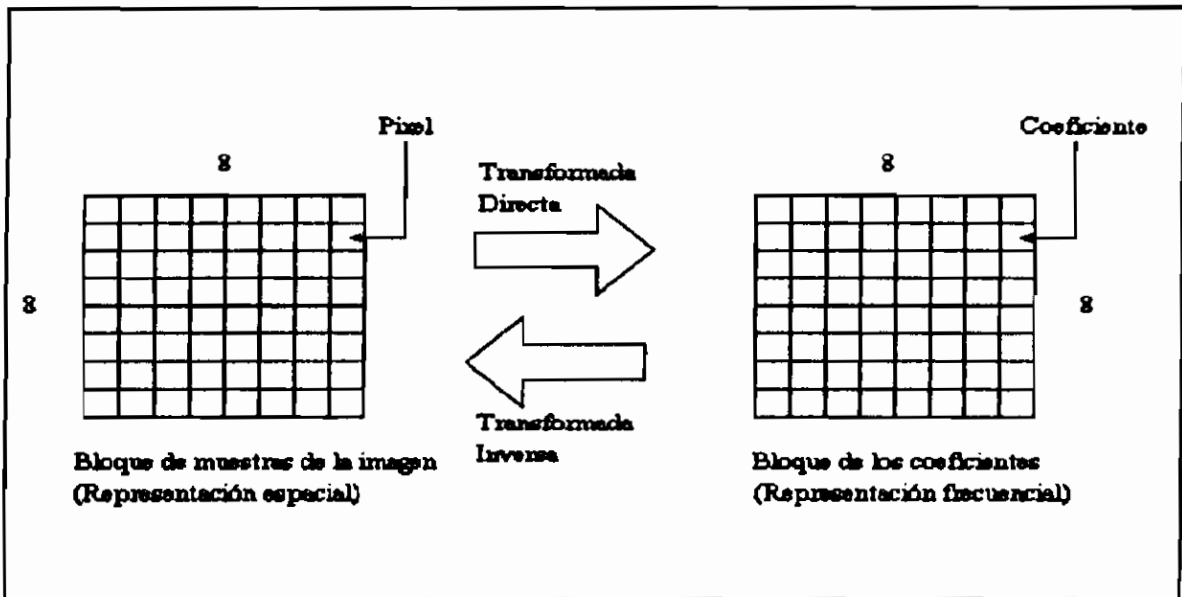


Figura 2.15 Codificación por Transformada de un bloque.

Estos coeficientes describen las características espaciales de frecuencia del bloque de píxeles originales. Sin embargo muestran grandes variaciones estadísticas con respecto a los píxeles originales-lo cual significa que el contenido de información del bloque de coeficientes es inferior al contenido del bloque de píxeles, que permitirá luego una codificación de información muy eficiente. La información espacial pasa a ser la información en frecuencia. Los coeficientes de mayor amplitud aparecen en la parte superior

izquierda del bloque. La mayoría de la información de la imagen se la llevan unos pocos coeficientes, los cuales representan ocasiones para hacer a la codificación eficiente. Esta propiedad de *Alta Compactación de Energía* le permite al proceso de la transformada discreta coseno, transformar la información de un bloque de pixeles a un pequeño número de coeficientes de frecuencia. Por ejemplo, si un bloque de pixeles consiste de franjas blancas y negras orientadas verticalmente (Figura 2.17), los coeficientes TDC correspondientes permanecerán exclusivamente a lo largo del eje horizontal de frecuencias.

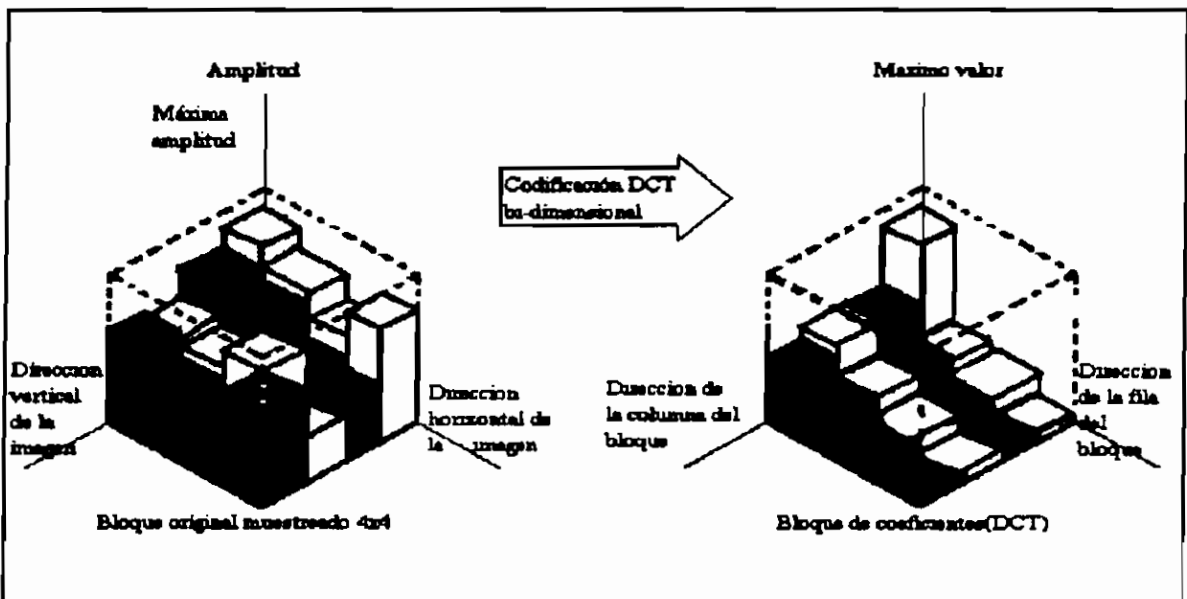


Figura 2.16 Codificación Bidimensional por Transformada.

La Transformada Discreta Coseno provee la misma información que la que lleva el bloque original de pixeles, pero de una forma que pueda ser *comprimida más fácilmente*. Los coeficientes de frecuencias bajas con valores significantes de amplitudes ocurren muy a menudo; los altos componentes de frecuencias ocurren con baja probabilidad. Si el bloque de pixeles consiste de filas alternantes de franjas blancas y negras (Figura 2.18), los coeficientes de la transformación permanecerán a lo largo del eje vertical solamente. El uso de la técnica de compresión TDC ha sido popularizada por el estándar *JPEG (Grupo Conjunto de Expertos en Fotografía)*

establecido en 1991 para la codificación de imágenes de tonalidades continuas. Al comienzo de la compresión de la nueva información, se pueden descartar los coeficientes con menores componentes de frecuencia. Solo los coeficientes con un gran contenido energético son realmente necesarios para codificar una imagen de alta calidad. Más aún la compresión es posible debido a que el sistema visual humano no es igualmente sensible a los coeficientes en las diferentes posiciones del bloque de frecuencias espaciales. Algunos medios y altos coeficientes de frecuencia pueden ser "toscamente" cuantificados asignándoles menos bits que los utilizados para las frecuencias bajas más importantes. El proceso TDC permite representar a un bloque de pixeles de una manera de conseguir un valor de baja entropía—una manera de hacer posible que la compresión sea eficiente.

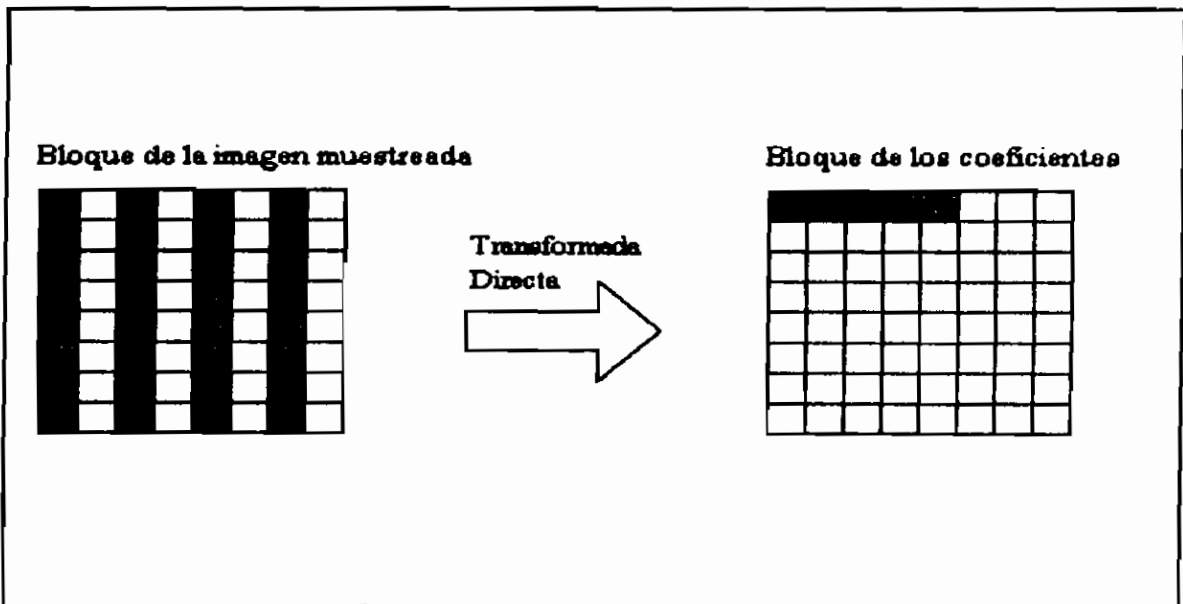


Figura 2.17 Transformación de formas verticales.

El proceso TDC no llega a reducir por si mismo la cantidad de datos; si el bloque de pixeles se describe a 8 bits/píxel, el bloque de coeficientes transformados requerirá típicamente de 11 o 12 bits por coeficiente debido a los requerimientos de precisión numérica en el procesador TDC. Sin embargo un gran número de coeficientes tendrán valores de cero o valores muy bajos, y así no necesitarán ser transmitidos o almacenados.

Dos procesos son introducidos para reducir aún más los datos de los coeficientes:

- La cuantificación de los valores de los coeficientes.
- La codificación por longitud variable (CLV) de estos coeficientes.

La cuantificación reduce aún más la velocidad binaria, al redefinir el rango dinámico (o el número de bits) permitido para cada coeficiente. En virtud de que el sistema visual humano es menos sensible a los errores de cuantificación para los coeficientes de alta frecuencia, estos pueden ser cuantificados con menos precisión. De esta manera el proceso de cuantificación puede realizarse de acuerdo a la localidad y la importancia de cada coeficiente en la percepción visual humana.

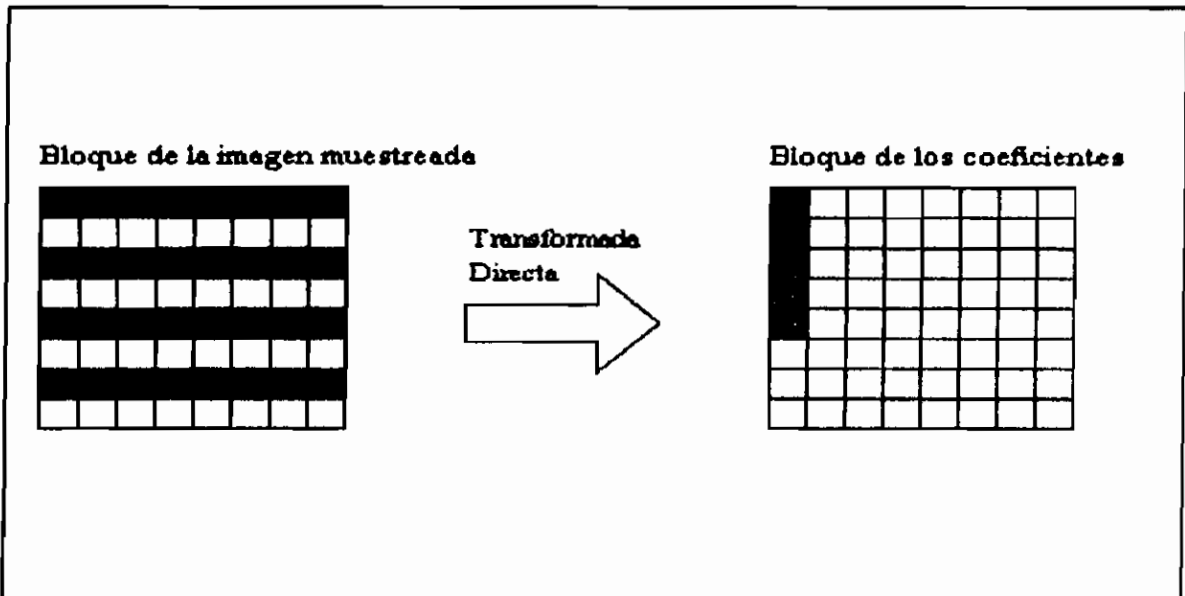


Figura 2.18 Transformación de formas horizontales.

La Figura 2.19 muestra una "matriz ponderada" como una ilustración del proceso de cuantificación usado en el estándar JPEG. El "Coeficiente DC" de un bloque transformado, el cual requiere la mayor precisión debido a que este representa al valor promedio de brillo del bloque de píxeles, debe ser cuantificado y codificado en forma diferente del resto de coeficientes bloque (o "coeficientes AC").

Cualquier error de cuantificación en el coeficiente DC será claramente visible al ojo humano como variaciones en los niveles de brillo del bloque; esto puede producir un *efecto mosaico*, en la imagen.

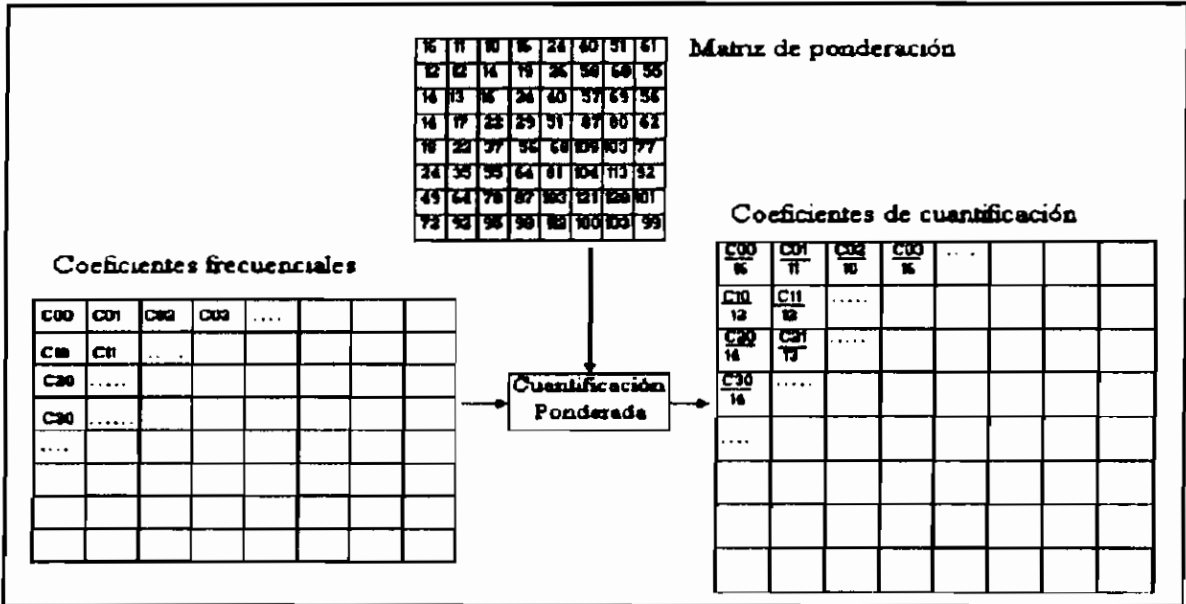


Figura 2.19 Cuantificación de la transformación.

En la matriz de cuantificación ponderada usada por el proceso JPEG, la cuantificación de los valores de coeficientes de frecuencia dan prioridad a los coeficientes que son importantes a la percepción visual humana. A esta matriz efectivamente se suma un *efecto umbral*: los coeficientes cuantificados de baja frecuencia son retenidos, mientras los coeficientes de alta frecuencia son descartados. Después de descartar los coeficientes de alta frecuencia que el sistema visual humano no alcanza a percibir, el proceso de codificación continúa. Las palabras códigos deben contener la información en la localidad de cada coeficiente, así el bloque de pixeles podrá ser reconstruido en el proceso de decodificación, una forma de dirigir esta reconstrucción podría ser enviando datos direccionales horizontales y verticales-pero estos datos requerirían muchos bits adicionales. Un esquema de codificación más eficiente, da un formato de un arreglo unidimensional, al arreglo bidimensional de coeficientes, usando un *reordenamiento en ZIG-ZAG* (Véase la Figura 2.20).

Para ayudar a prevenir el efecto de mosaico en la imagen decodificada, el coeficiente DC es codificado de una manera "predictiva" al usar el coeficiente DC del bloque de pixeles previo a manera de predictor. El valor diferencia entre estos dos términos coeficientes DC es luego codificado usando una tabla CLV (Codificación por Longitud Variable). El barrido en ZIG-ZAG reordena a los datos del bloque de coeficientes. Este proceso de barrido reagrupa los coeficientes en un orden aproximadamente decreciente de sus valores de amplitud y los coloca en orden incremental de frecuencias espaciales. Los coeficientes de los valores no-nulos, son típicamente separados por un flujo de coeficientes de valor cero. La codificación por longitud variable puede codificar eficientemente los valores de los coeficientes no-nulos en combinación con el "flujo de longitudes" de los términos cero.

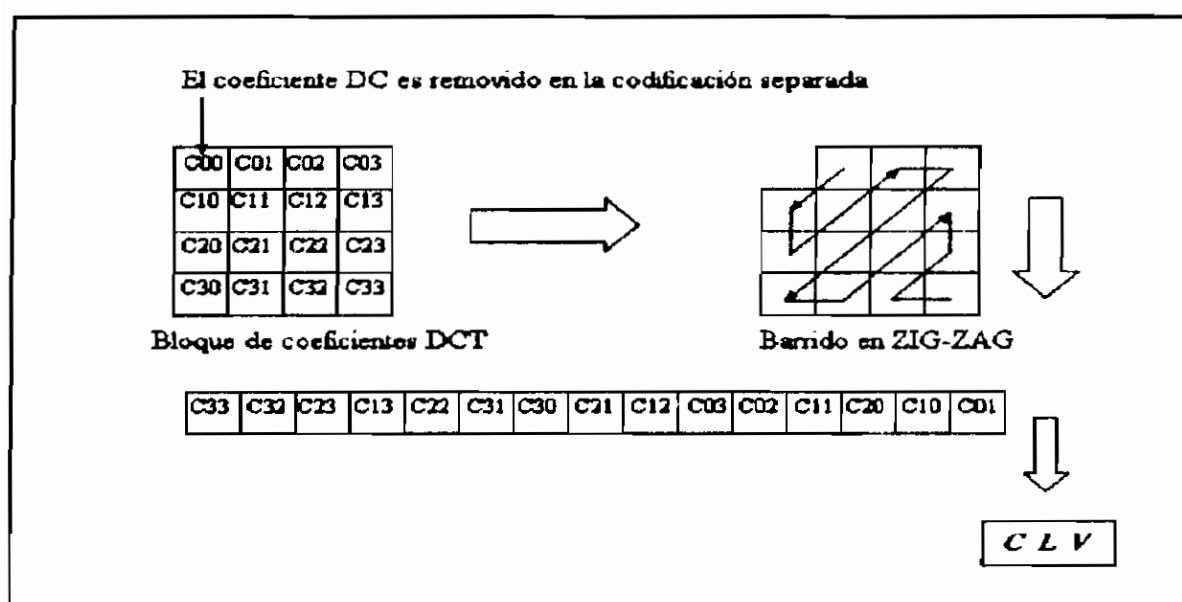


Figura 2.20 Proceso de codificación CLV de la transformada.

Un corto código CLV señala el "fin de bloque" (EOB), el cual indica que todos los siguientes coeficientes en el bloque, son cero. La transformada discreta coseno y el proceso de cuantificación toma ventaja de las redundancias espaciales que pueden existir en los bloques de pixeles de una imagen de video.

El resultado es un arreglo de coeficientes cuantificados y transformados que pueden convertir la información del bloque de pixeles original más eficientemente con una reducida velocidad binaria. Como siempre, existe un compromiso entre la reducción de la velocidad binaria generada por el bloque codificado y la prevención de distorsiones visibles introducidas por el proceso de selección y cuantificación. El proceso de codificación por transformada se puede visualizar desde otro "innovador" punto de vista, haciendo uso de las llamadas *Funciones Kernel*:

Sea $x(m,n)$ un porción de imagen evaluada, como un arreglo matricial de $M \times N$ de números con el elemento superior izquierdo (Coeficiente DC) denotado $x(0,0)$. Un par de transformadas bi-dimensionales del arreglo $x(m,n)$ se define de la siguiente manera:[8]

$$X(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n) \text{Kern}_d(k,l;m,n) \quad (2.54)$$

para, $k=0,1,\dots,M-1$
 $l=0,1,\dots,N-1$; y,

$$x(m,n) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X(k,l) \text{Kern}_i(k,l;m,n) \quad (2.55)$$

para, $m=0,1,\dots,M-1$
 $n=0,1,\dots,N-1$;

donde,

$$\sum_{k=0}^{M-1} \sum_{l=0}^{N-1} \text{Kern}_d(k,l;m,n) \text{Kern}_i(k,l;m',n') = \begin{cases} 1 & \text{si } m=m' \text{ y } n=n' \\ 0 & \text{en otro caso} \end{cases} \quad (2.56)$$

para, $m=0,1,\dots,M-1$.

El par transformador obtenido depende del escogimiento de los *Kernels* de transformación $\text{Kern}_d(k,l;m,n)$ y $\text{Kern}_i(k,l;m,n)$. La d y la i son los indicadores de Kernel directo o Kernel inverso, respectivamente. Dada esta definición general, se aplica a las transformadas antes estudiadas

como la Transformada Discreta de Fourier (TDF), la Transformada Discreta Coseno (TDC), la Transformada Discreta de Hartley (TDH) y para la Transformada Discreta Cas-Cas (TDCC):

Para la TDF los Kernel están definidos: [8]

$$Kern_d(k, l; m, n) = \frac{1}{\sqrt{MN}} e^{-j2\pi(\frac{mk}{M} + \frac{nl}{N})} \quad (2.57)$$

$$Kern_i(k, l; m, n) = \frac{1}{\sqrt{MN}} e^{j2\pi(\frac{mk}{M} + \frac{nl}{N})} \quad (2.58)$$

Para la TDC los Kernel están definidos: [8]

$$Kern_d(k, l; m, n) = \frac{1}{\sqrt{MN}} \alpha(k) \alpha(l) \cdot \text{Cos}\left(\frac{\pi k(2m+1)}{2M}\right) \text{Cos}\left(\frac{\pi l(2n+1)}{2N}\right) \quad (2.59)$$

$$Kern_i(k, l; m, n) = \frac{1}{\sqrt{MN}} \alpha(k) \alpha(l) \cdot \text{Cos}\left(\frac{\pi k(2m+1)}{2M}\right) \text{Cos}\left(\frac{\pi l(2n+1)}{2N}\right) \quad (2.60)$$

donde, [8]

$$\alpha(i) = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } i=0 \\ 1 & \text{en otro caso} \end{cases} \quad (2.61)$$

Para la TDH los Kernel están definidos: [8]

$$Kern_d(k, l; m, n) = \frac{1}{\sqrt{MN}} \text{Cas}\left(2\pi\left(\frac{mk}{M} + \frac{nl}{N}\right)\right) \quad (2.62)$$

$$Kern_i(k, l; m, n) = \frac{1}{\sqrt{MN}} \text{Cas}\left(2\pi\left(\frac{mk}{M} + \frac{nl}{N}\right)\right) \quad (2.63)$$

donde, [8]

$$\text{Cas}(\beta) = \text{Cos}(\beta) + \text{Sen}(\beta) \quad (2.64)$$

Para la TDCC los Kernel son:[8]

$$Kern_d(k, l; m, n) = \frac{1}{\sqrt{MN}} \text{Cas}\left(\frac{2\pi mk}{M}\right) \text{Cas}\left(\frac{2\pi nl}{N}\right) \quad (2.65)$$

$$Kern_l(k, l; m, n) = \frac{1}{\sqrt{MN}} \text{Cas}\left(\frac{2\pi mk}{M}\right) \text{Cas}\left(\frac{2\pi nl}{N}\right) \quad (2.66)$$

En el ANEXO 2 se hallan ejemplos ilustrativos de codificación de una imagen de video por medio de la transformada KL (Anexo 2e), y codificaciones de pequeñas secuencias por medio de la transformada de Fourier (Anexo 2f), de Walsh-Hadamard (Anexo 2g), de Haar (Anexo 2h), y TDC (Anexo 2i).

2.1.4 OTRAS TECNICAS DE CODIFICACION:

Ya se ha cubierto casi toda la teoría de compresión de imágenes en lo que respecta a métodos aislados de codificación, ahora la siguiente etapa será el estudio de la inter-relación de dichos métodos en una sola técnica de compresión como es la codificación híbrida o sobre aplicaciones básicamente de los procesos MICD a casos particulares como la codificación de imágenes de dos tonalidades (blanco y negro) o el caso más global de la codificación de imágenes a color, finalmente se mostrarán misceláneas técnicas aparecidas en los últimos años sobre compresión de imágenes.

2.1.4.1 Codificación híbrida:

En imágenes de video comunes, en el dominio temporal, ocurre otra clase de redundancia. Esta dependencia inter-imagen (similitud entre imágenes sucesivas) también puede aprovecharse para la compresión de video. En una imagen en movimiento, la mayoría de los cambios que toman lugar de un campo a otro campo son producidos por objetos moviéndose de un lugar a otro y cubren o descubren otros objetos y áreas de imágenes. Otros cambios ocurren cuando los objetos rotan, o los niveles de iluminación se incrementan o decrementan, etc. Las técnicas de codificación predictiva de la Sección 2.1.2 se basan en el barrido y en reglas de predicción recur-

sivas. La codificación híbrida es un método de implementación de un vector $N \times 1$ codificador MICD por medio de N codificadores escalares MICD desacoplados. Esto se logra al combinar técnicas de codificación predictiva y por transformada.

Típicamente la imagen es unitariamente transformada a una de sus dimensiones a fin de decorrelacionar sus píxeles en esa dirección. Cada coeficiente de transformación es entonces codificado secuencialmente en la otra dirección por un MICD uni-dimensional (Figura 2.21). Esta técnica combina las ventajas de la simplicidad en el hardware del MICD y la gran eficacia de la codificación por transformada. La complejidad en el hardware de este método está en el codificador uni-dimensional por transformada además de los N -canales MICD. En la práctica, el número de canales es significativamente menor que N debido a que muchos elementos del vector transformado se ponen a cero y por lo tanto no se codifican todos (Ver Sección 2.1.3.1).

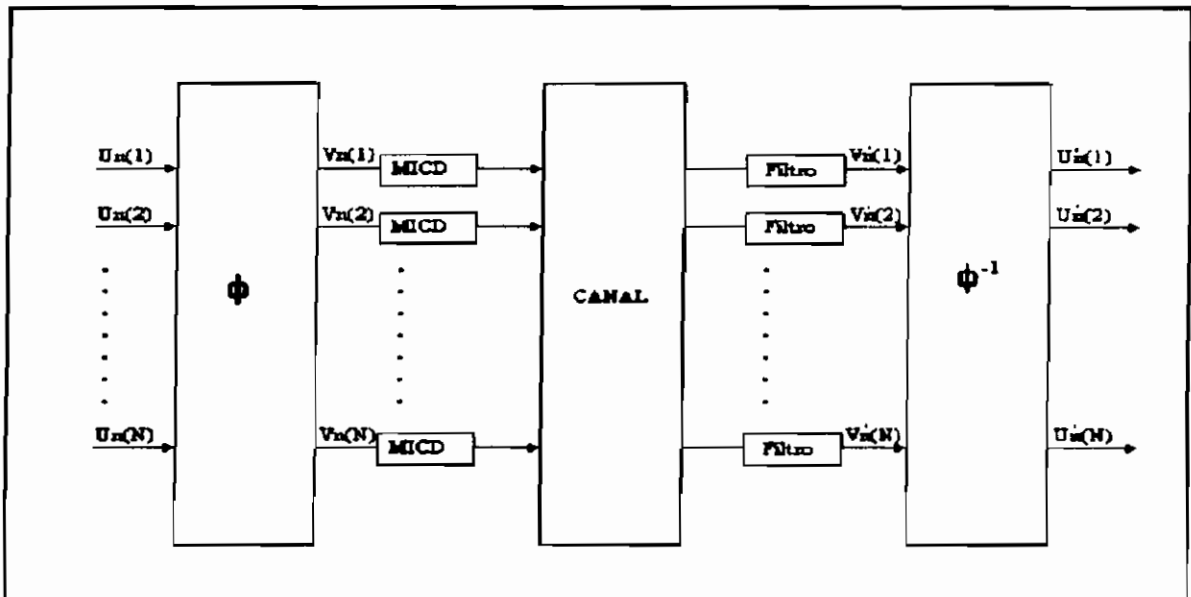


Figura 2.21 Codificación Híbrida.

Se eliminan redundancias en el espacio (por medio de la Transformada, Cuantificación, Codificación de entropía) y en el tiempo (por medio de la codificación predictiva compensadora de movimiento). Estas técnicas se combinan en un típico ejemplo de *Codificación Híbrida*.

La Figura 2.22 muestra un diagrama de bloques de un sistema Híbrido de compresión de video, simplificando un sistema de compresión MICD/TDC.

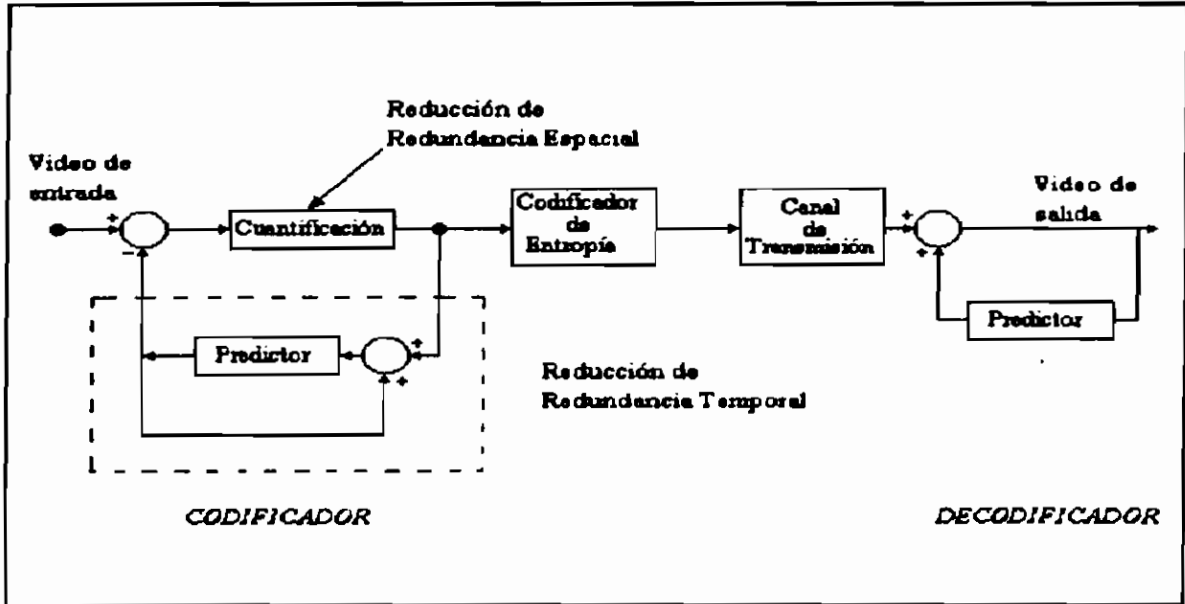


Figura 2.22 Codificación Híbrida MICD-TDC.

Las redundancias espaciales son reducidas mediante la Transformada Discreta Coseno (TDC) y el proceso de Cuantificación; las redundancias temporales se reducen por la predicción de errores y la compensación del movimiento. La codificación de la entropía luego crea un flujo de bits comprimido que requiere de pocos bits de datos para almacenar o transmitir la misma información de la imagen. En el terminal receptor, la TDC y la cuantificación restaura la característica espacial de los bloques de píxeles. Un predictor simple recibe los vectores de movimiento para compensar los errores de predicción inter campo/trama.

2.1.4.2 Codificación de gráficos de dos tonalidades:

La necesidad del almacenamiento electrónico y de la transmisión de gráficos de dos tonos tales como dibujos, cartas, mapas, y otros documentos se ha incrementado rápidamente, especialmente con el advenimiento de computadores personales. Los productos comerciales para la transmisión de documentos vía telefónica y de datos ya existe. El CCITT ha "recomendado" un grupo de ocho documentos para la comparación

y evaluación de algoritmos de codificación de imágenes binarias (Véase ANEXO 1). Los estándares del CCITT para los rangos de muestreo para un documento típico A4 (8½"x11") con respecto a su transmisión vía el así llamado-Grupo 3 de aparatos facsímiles digitales, son de 3.85 líneas por milímetro en la resolución normal y 7.7 líneas por milímetro en la resolución alta en la dirección vertical. La rata de muestreo horizontal es de 1728 pixeles por línea, la que corresponde a una resolución de 7.7 líneas por milímetro o de 200 puntos por pulgada (ppp). Para páginas de periódico y otros documentos que contienen tanto texto como imágenes a dos tonalidades, se usan tasas de muestreo de 400 a 1000 puntos por pulgada. Así para la página estándar de 8½ por 11 pulgadas, se requerirán de 1.87×10^6 bits a una densidad de muestreo de 200puntos por pulgada (ppp)x100líneas por pulgada (lpp). Transmitir esta información a 4800 bps vía línea telefónica, le tomaría a la transmisión cerca de 1.3 minutos.

Para una densidad de muestreo de 150 puntos por pulgada y cuantificación de 2 niveles (1 bit por muestra), una carta de 8½"x11" contiene aproximadamente 2 millones de bits. Al transmitirla como un documento a través de un canal telefónico con un modem de 2400 bits/segundo, se transmitiría en 15 minutos. Para reducir el tiempo de transmisión a, por ejemplo 1 minuto, se requeriría un factor de compresión de 15:1. Se puede, por supuesto, alcanzar este requerimiento al reducir la resolución y/o usar un modem de alta velocidad.

De aquí en adelante se asumirá que para la transmisión de este tipo de imagen (que es una imagen básica) el cuantificador utilizado será de dos niveles por muestra. Luego ha sido de considerable interés la transmisión digital de páginas de periódicos (transmisión por facsímil de boletines de prensa). Se ha encontrado que para material de texto, una densidad de muestreo de 400 puntos por pulgada es suficiente para una calidad de reproducción satisfactoria. De la imagen prototipo de dos tonos (negro y blanco) se puede pasar

entonces al tipo de imágenes que más interesa, las de varias tonalidades de gris. Para este tipo de imágenes se han extrapolado los criterios de cuantificación y resolución para la imagen prototipo, teniéndose así que se ha estimado que para el cuantificador de estas imágenes de tonos continuos de gris, 64 niveles son suficientes y 32 son satisfactorios en la mayoría de ocasiones. En la práctica, a fin de asegurar una calidad de imagen, se utiliza una resolución mucho más alta de la mencionada. Así por ejemplo el *Wall Street Journal* utiliza resoluciones de 1000 puntos por pulgada y superiores para la transmisión de sus boletines de prensa. Muchos algoritmos de compresión para imágenes binarias de característica visual a dos tonalidades explotan el hecho de que:

- 1.- Muchos de los pixeles son blancos; y,
- 2.- Los pixeles negros aparecen con una regularidad que se manifiesta en la forma de caracteres, símbolos, o linderos conectados.

Existen tres conceptos básicos que se emplean para la codificación de tales imágenes:

- (a) Codificación de solamente puntos de transición entre negro y blanco.
- (b) Saltos, evitando la codificación del blanco.
- (c) Reconocimiento de patrones.

Los algoritmos que se basan en estos conceptos, ya se los ha estudiado en párrafos anteriores, estos pueden ser:

- Codificación Lineal(Sección 2.1.1.2)
- Codificación Huffman(Sección 2.1.1.1.2)
- Codificación Predictiva(Sección 2.1.2)

Existe una gran cantidad de esquemas de codificación eficiente para imágenes de dos tonos; se basan en los llamados *Conceptos Heurísticos*, estos son: La omisión de blancos, la transmisión de solamente los puntos vecindades entre negro y blanco; y, el reconocimiento de patrones.

2.1.4.2.1 Codificación por Omisión de Bloques Blancos (OBB):

(a) Codificación unidimensional: El método, muy simple se lo realiza de la siguiente manera:

- Cada línea barrida y muestreada se divide en N bloques-píxel.
- Para un bloque que contiene solo píxeles blancos, se utiliza la palabra código "1".
- Para un bloque que contiene al menos un píxel negro, se utiliza una palabra código de $(N+1)$ bits, el primer bit es un "0" y los restantes " N " bits representan al patrón primario del N bloque (Blanco=1, Negro=0).

Así si $N=4$ y un bloque contiene dos píxeles negros seguidos de dos píxeles blancos, la palabra código para cada bloque es "00011".

La tasa de entropía de este método es:

$$R_N = 1 - P_N + \frac{1}{N} \left(\frac{\text{bits}}{\text{pixel}} \right) \quad (2.67)$$

donde,

N = Tamaño del bloque en píxeles.

P_N = Probabilidad de que un bloque contenga solo píxeles blancos.

Para una imagen o una clase de imágenes, la R_N puede medirse experimentalmente, y la ecuación (2.67) se utiliza para calcular la tasa de entropía, esta depende del tamaño del bloque N .

Algunos resultados experimentales se listan en la siguiente tabla la cual muestra las tasas medidas de ocho imágenes en dos diferentes resoluciones (Anexo 1). El tamaño del bloque fue de $N=10$, con el que probablemente se obtienen tasas cercanas a la mínima para todos los casos.

(bit/pixel)	Resolución	
	200ppp	100ppp
Imagen		
CCITT #1	0.117	0.134
CCITT #2	0.196	0.211
CCITT #3	0.240	0.264
CCITT #4	0.316	0.361
CCITT #5	0.243	0.270
CCITT #6	0.199	0.223
CCITT #7	0.313	0.371
CCITT #8	0.616	0.641

(b) **Codificación bidimensional:** El método de codificación OBB puede extenderse al caso bidimensional de una manera directa: Una imagen se divide en $M \times N$ bloques-píxel. Un bloque que contiene solamente píxeles blancos se representa por la palabra código "1" de un solo bit. Un bloque que contiene por lo menos un píxel negro se representa por una palabra de $(MN+1)$ bits, siendo el primer bit un "0", y los restantes MN bits el patrón binario del bloque.

2.1.4.2.2 Cuantificación Diferencial Predictiva (CDP):

La codificación longitudinal aprovecha de las restricciones estadísticas de los píxeles a lo largo de la dirección de barrido de la línea solamente. Las más altas compresiones pueden alcanzarse si también se hace uso de las restricciones estadísticas a lo largo de la línea barrida. Una manera de realiza esto es la así llamada técnica de cuantificación diferencial predictiva (CDP). Este esquema hace uso de la correlación entre líneas barridas, y puede ser considerada como una extensión de la codificación longitudinal a dos dimensiones. En lugar de corridas longitudinales, aquí se transmiten las diferencias entre las correspondientes corridas longitudinales de líneas barridas sucesivas.

En particular, las cantidades δ' y δ'' como se definen en la Figura 2.23 se transmiten, conjuntamente con los mensajes (*Nuevo inicio*, y *Fusión*) indicando el comienzo y fin de un área negra.

Así para cada línea barrida, se compara con la precedente línea barrida y se transmite básicamente los cambios en las localidades de transición de blanco a negro (δ') y los cambios en las corridas longitudinales del negro (δ''). Un esquema alternativo podría ser el transmitir los cambios en las localidades de transiciones de negro a blanco en lugar de δ'' .

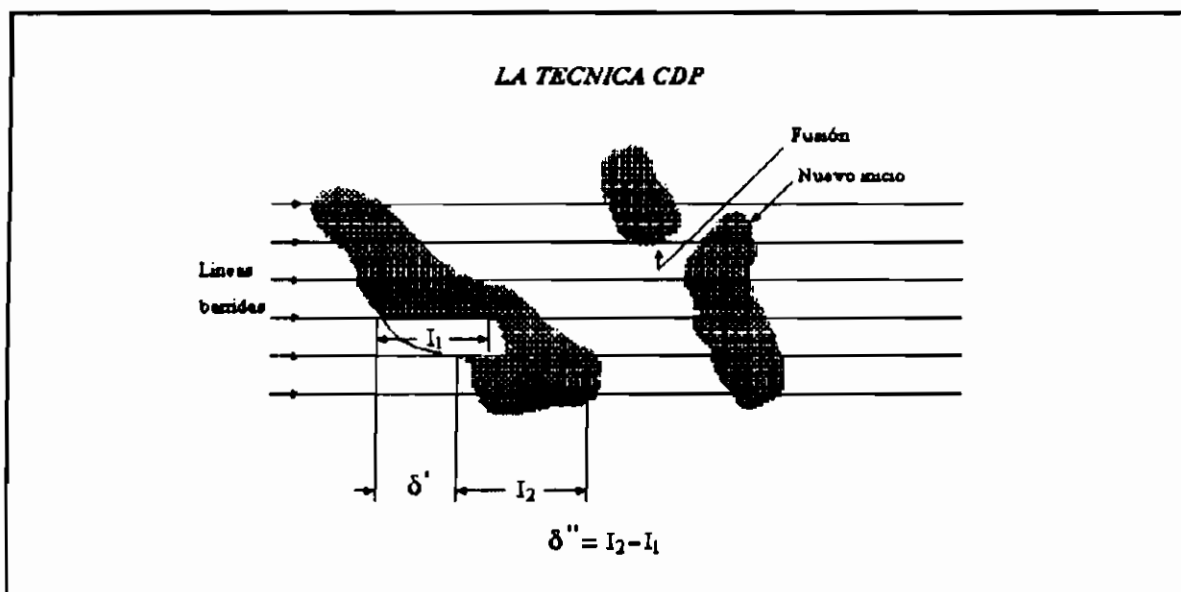


Figura 2.23 La técnica CDP.

Se observa que CDP es esencialmente un esquema de trazo de límites. Los parámetros δ' y δ'' señalan donde van los puntos limítrofes en la siguiente línea barrida.

Sin embargo, a diferencia de un verdadero esquema trazador de límites, el CDP requiere un almacenamiento equivalente de solamente dos líneas barridas en lugar del de todas las líneas de la imagen completa.

2.1.4.2.3 Codificación de la Dirección Relativa (CDR):

La codificación de la dirección relativa (CDR) utiliza el mismo principio de la CDP y computa las diferencias de corridas longitudinales de ceros o unos) por medio del rastreo de la última transición en una misma línea o la transición más cercana sobre la línea previa. Por ejemplo la transición Q de un pixel (Figura 2.24) se codifica por la distancia más corta PQ o $Q'Q$, donde P es el elemento de transición precedente en la línea actual, y Q' es el elemento de transición más cercano a la derecha de P en la línea previa, cuya dirección de transición es la misma que la de Q . Si P no existe, entonces se le considera como el pixel imaginario a la derecha del último pixel en la línea precedente. La distancia QQ' es codificada como $+N$ si Q' está ($N \geq 0$) pixeles a la izquierda o $-N$ si Q' está ($N \geq 1$) pixeles a la derecha de Q' sobre la línea precedente. La distancia PQ es codificada como ($N \geq 1$) si esta está N pixeles fuera. Una codificación de este tipo se puede ver como $xx\dots xx =$ representación binaria de (N):

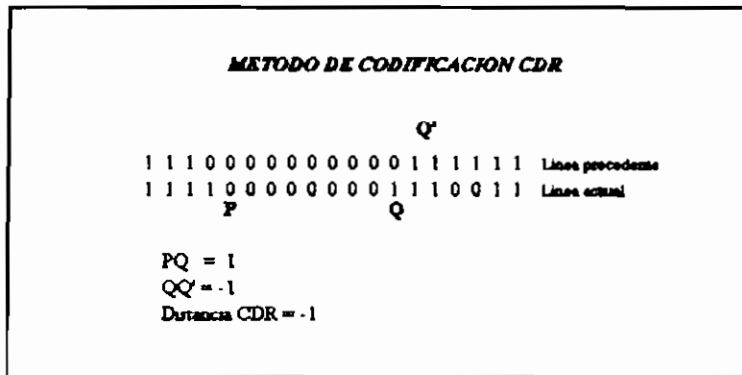


Figura 2.24 Codificación CDR.

Distancia	Código	N	F(N)
+0	0	1-4	0xx
+1	100	5-20	10xxxx
-1	101	21-84	110xxxxxx
N(N>1)	111F(N)	85-340	1110xxxxxxxx
+N(N>2)	1100F(N)	341-1364	11110xxxxxxxxxxx
-N(N>2)	1101F(N)	1365-5460	111110xxxxxxxxxxxx
:	:	:	:
:	:	:	:

2.1.4.2.4 Codificación por Designación de la Dirección Relativa de Elementos (CDDRE) - CCITT modificada:

El algoritmo de la codificación por designación de la dirección relativa de elementos (CDDRE) ha sido recomendada por el CCITT para la codificación bi-dimensional de documentos. Este es una modificación de la CDR y de otros códigos similares. Haciendo referencia a la Figura 2.25, se define a a_0 como el elemento de transición de referencia cuya posición está definida por el modo de codificación previo. Inicialmente, a_0 se lo toma como la transición de un imaginario pixel blanco situado a la izquierda del primer pixel sobre la línea de codificación.

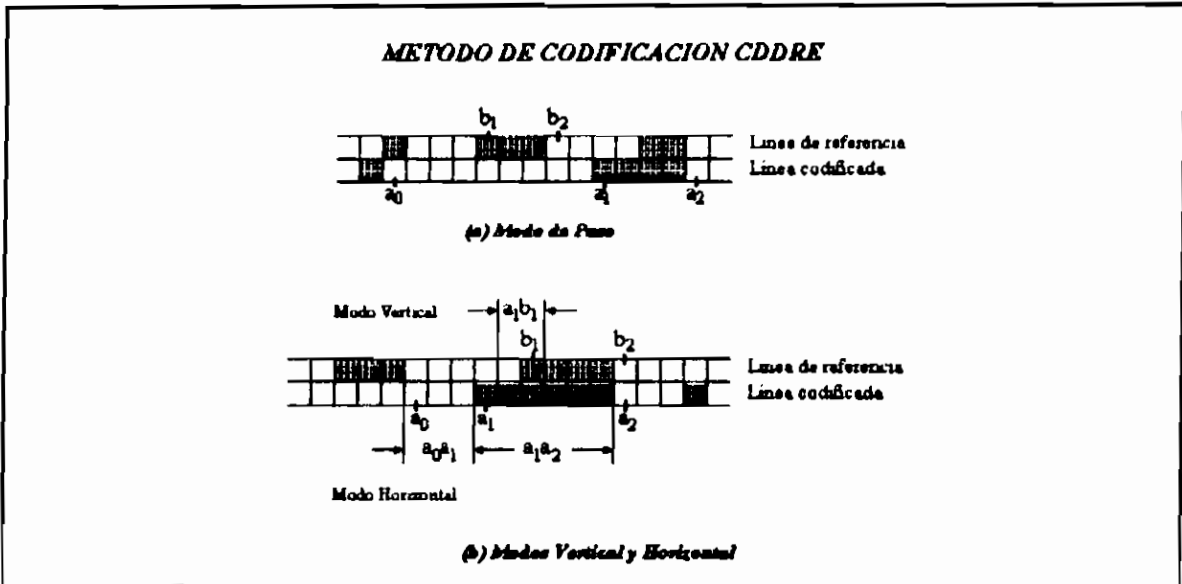


Figura 2.25 Codificación CDDRE.

El siguiente par de transiciones de píxeles a la derecha de a_0 se denotan como a_1 y a_2 sobre la línea de codificación y b_1 y b_2 sobre la línea de referencia y que tienen colores alternativos al de a_0 . Cualquiera de los elementos a_1 , a_2 , b_1 , b_2 no detectados para una línea codificada en particular se toma como el pixel imaginario a la derecha del último elemento de su respectiva línea barrida. El pixel a_1 representa al próximo elemento de transición a ser codificado.

El algoritmo tiene tres modos de codificación, como se indica a continuación:

Modo de Paso: b_2 está a la izquierda de a_1 Figura (2.25a). Esto identifica a las corridas blancas o negras sobre la línea de referencia que no se solapa (superpone) con la correspondiente corrida de blancos o negros de la línea codificada. El elemento de referencia a_0 es puesto debajo de b_2 en preparación a la siguiente codificación.

Modo Vertical: a_1 es codificado respecto a b_1 por la distancia $a_1 b_1$, la cual puede tomar valores de 0,1,2,3 a la derecha o izquierda de b_1 . Estos son representados por $V(0)$, $V(Rx)$, $V(Lx)$, $x=1,2,3$. En este modo a_0 es puesto como a_1 en preparación a la siguiente codificación.

Modo Horizontal: Si $|a_1 b_1| > 3$, el modo vertical no se lo utiliza y las secuencias $a_0 a_1$ y $a_1 a_2$ se codifican usando códigos Huffman. Luego de la codificación, la nueva posición de a_0 es puesta como a_2 . Si se necesita este modo para el primer elemento de la línea de codificación, entonces el valor $a_0 a_1 - 1$ es codificado antes que $a_0 a_1$. Así si el primer elemento es negro, entonces se codifica una secuencia de ceros.

2.1.4.3 Codificación de imágenes a color:

La codificación digital de las señales de video a color ha tenido considerablemente menos atención que la codificación de señales de video monocromáticas. Sin embargo dada la extensa proliferación de sistemas de televisión en color y la preferencia general por estas imágenes, es obvio que la codificación eficiente de señales en color es de primordial importancia. Desde los primeros experimentos en codificación del color, se han explotado principalmente dos aspectos: El primero consiste en operar directamente sobre la señal compuesta de televisión, mientras que el segundo divide a la señal en tres componentes, codifica a cada componente

separadamente y luego, después de la transmisión, las combina de nuevo para formar una señal compuesta, y se puede generalizar a imágenes de color y de múltiple espectro, como se muestra en la Figura 2.26. Cada pixel está representado por un $px1$ vector. En el caso de imágenes en color, la entrada es un vector de $3x1$ conteniendo los *Componentes R, G, B*.

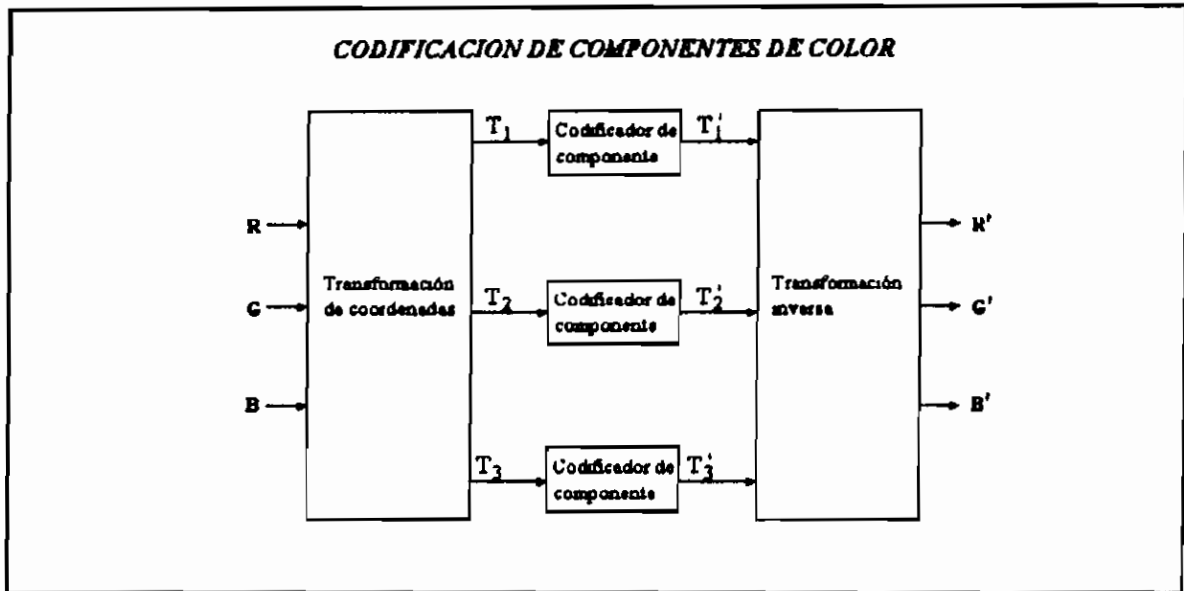


Figura 2.26 Codificación de componentes de color.

Este vector es transformado a otro sistema coordinado, donde cada componente puede ser procesada por un codificador espacial independiente. En la codificación de imágenes en color, se debe considerar lo siguiente:

- 1) La componente de luminancia (Y) tiene un mayor ancho de banda que las componentes de crominancia (I, Q) o (U, V) y,
- 2) La métrica en la diferencia de color es *No-Euclidiana* en esas coordenadas, esto significa que se percibe de manera diferente una igual potencia de ruido en distintos componentes de color.

En esquemas prácticos de codificación de imágenes, las señales de crominancia de bajo ancho de banda son muestreadas a un tercio o un sexto de la frecuencia de muestreo de la señal de luminancia.

2.1.4.3.1 Métodos de codificación de componentes:

El punto de partida para la codificación de las componentes de color es usualmente una señal **R,G,B** o los componentes de una señal compuesta, v.g. **Y,I,Q**. Estas señales se pueden transformar a otro espacio por medio de la digitalización. Esta digitalización puede envolver a una digitalización de cada componente o, alternativamente, cuantificación multidimensional por medio de un tratamiento simultáneo de más de un componente.

Si las señales **R, G, y B** se digitalizan, no es necesario cuantificar a cada una de ellas con la misma precisión debido a que ellas no son igualmente sensibles al ruido añadido. Por ejemplo, para un cuadro que tiene presentes los tres colores primarios, la relación señal/ruido presente es de 36dB para el Azul(**B**), 41dB para el Rojo(**R**), y 43 dB para el Verde(**G**), aunque estas figuras varían del mismo modo que la luminancia.

La diferencia en la sensibilidad al ruido de los componentes de color es aún muy grande para cuadros "normales". En la señal azul, el ruido es 10dB menos visible que en una señal roja y 20dB menos visible que en una señal verde. Para la cuantificación del ancho de banda total de estas señales sin errores visibles, se requieren al menos 4bits para la señal azul, 5bits para la roja, y 6 bits para la señal verde. El esquema de codificación de las componentes de la señal RGB muestra que se requiere una resolución relativamente alta para cada componente.

Para escenas típicas, se consigue ancho de banda de 2MHz para el verde, los anchos de banda para el rojo y el azul se reduce a 1.5MHz y 1.0Mhz respectivamente. Esta reducción relativa es menor que la que es posible por señales de crominancia. La luminancia y crominancia pueden ser directamente digitalizadas, y codificadas por métodos de longitud variable (como la codificación Huffman) así se puede reducir

aún más la tasa de entropía. Una pequeña reducción adicional a la entropía es alcanzada por las señales I y Q (V y U) que por las señales de luminancia debido a la mayor reducción en el rango dinámico de las señales de crominancia respecto a las señales de luminancia.

2.1.4.3.2 Codificación de la señal compuesta:

Se puede economizar en equipos utilizando codificación con multiplexaje en tiempo de las señales componentes por medio de un sencillo codec MIC-D. El multiplexaje podría involucrar el almacenamiento de las componentes de crominancia y colocarlas en el intervalo de borrado de líneas de la luminancia.

Alternativamente, la luminancia y la crominancia pueden ser entrelazadas muestra a muestra y se puede utilizar un almacenamiento separado en el caso de tener un lazo predictor realimentado más complejo. Extendiendo este concepto a sumas lineales de entrelazados de las muestras de crominancia y luminancia, se llega a una señal codificada la cual podía haber sido igualmente obtenida por medio de la codificación de la señal *compuesta* a un ritmo de muestreo tal que sea un múltiplo entero del de la subportadora.

2.1.4.4 Codificación por cuantificación vectorial:

La cuantificación vectorial es un esquema efectivo para la compresión de voz. Esta se ha aplicado también a la compresión de imágenes obteniéndose imágenes de calidad aceptable para codificaciones de 1.0 bits/pixel para señales monocromáticas y de 2.25 bits/pixel para las de color.

El primer paso en la cuantificación vectorial es la descomposición de la imagen en un arreglo vectorial. Estos vectores pueden definirse en un número de formas, por ejemplo, como los componentes de color de un pixel, los valores de intensidad espacial de un grupo de pixeles contiguos, o como componentes transformados de estos grupos.

El siguiente paso es generar un *listado* de vectores representativos utilizando un algoritmo iterativo de agrupamiento, que se lo utilizará para cuantificar al arreglo. La compresión se alcanza al reemplazar el vector representativo por una *etiqueta*. La reconstrucción de la imagen se produce por medio de una *observación* a la tabla utilizando a la etiqueta como una entrada a la tabla que contiene los vectores representativos del listado. Así, la memoria utilizada necesita almacenar solamente la imagen comprimida, la cual se reconstruye luego, durante el tiempo de refresco por medio de la observación a la tabla. Dos características atractivas tiene la cuantificación vectorial: Que no es necesario asumir criterios estadísticos; y, que se pueden utilizar distintas medidas de la distorsión.

En la cuantificación vectorial, generalmente se genera un listado fijo desde un arreglo de imágenes. Para mantener una baja distorsión sobre todas las posibles imágenes, se requiere por lo tanto un listado muy largo. Adicionalmente, para obtener una codificación eficiente, se deben utilizar vectores de gran dimensión.

Esto involucra dificultades prácticas no solo en la generación del listado, sino también en la codificación de las imágenes entrantes y su almacenaje en el listado que se encontrará en el decodificador. El asumir básicamente que las imágenes individuales, y más específicamente, las subpartes de las imágenes solo necesitarían direccionar un pequeño subarreglo del gran listado ya fijo. Las técnicas adaptivas presentadas involucran a la generación y transmisión separada de listados para cada imagen individual o subpartes de la imagen.

En sistemas de imágenes a color de alta calidad, se requiere normalmente una memoria de refresco con 24 bits/pixel. La cuantificación vectorial ha sido propuesta para reducir los requerimientos de memoria.

La memoria principal contendría las clases de etiquetas y una tabla de rápido acceso. En un espacio de color tri-dimensional, se divide, a fin de que cada color contenga aproximadamente el mismo número de píxeles.

2.1.4.4.1 Cuantificación vectorial adaptiva de imágenes monocromáticas:

La cuantificación vectorial no-adaptiva emplea un *Listado* sencillo generado desde un gran arreglo de imágenes. Para mantener fidelidad en la codificación el listado debe ser grande, el cual involucra más bits para codificar las etiquetas e incrementar la complejidad del codificador/decodificar. Para una imagen dada, el esquema adaptivo proviene del cómputo de un listado separado (o arreglos de listados) de los vectores de una imagen. Así, se requieren listados más pequeños que los del caso no-adaptivo para mantener un nivel dado de distorsión. El listado y la imagen codificada se deben transmitir al decodificador. Así la tasa efectiva se puede determinar de la cantidad de datos requeridos para transmitir el listado y los elementos codificados (etiquetas) de la imagen.

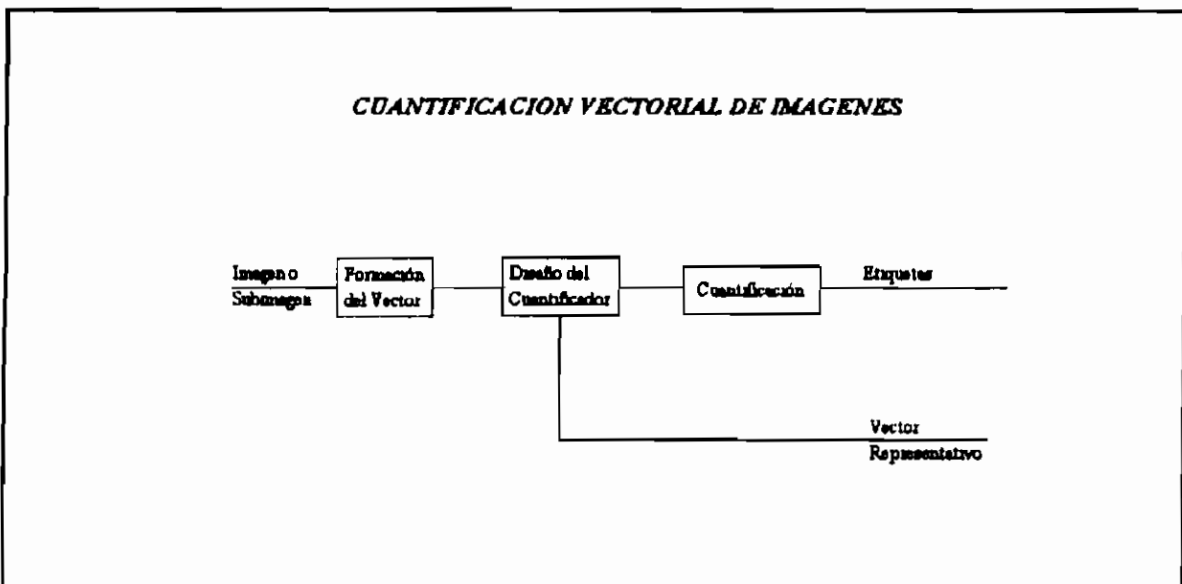


Figura 2.27 Cuantificación Vectorial de Imágenes.

Un diagrama de bloques de las operaciones involucradas en la cuantificación vectorial adaptiva de imágenes se encuentra en la Figura 2.27. La imagen es primero descompuesta en una secuencia de vectores. El listado es entonces computado por medio de algún algoritmo iterativo en los vectores dados; esto es, el arreglo se dibuja de la imagen que está siendo codificada. En la cuantificación vectorial espacialmente adaptiva, la imagen es primeramente particionada en regiones rectangulares separadas o subimágenes, y para cada una, se computan listados separados independientemente. Para cada subimagen, se transmiten los vectores representativos generados localmente del listado, seguido por las palabras código o etiquetas para los vectores de la imagen. La tasa de entropía promedio para la cuantificación vectorial de imágenes contiene dos componentes: r_c asociado con la transmisión de la palabra código individual, y r_o , el encabezado, relacionado a la transmisión de los vectores representativos del listado.

La tasa de entropía, en la siguiente relación:[15]

$$r = \frac{\log_2 K}{n} + K \cdot n \cdot \frac{b}{M} \quad \text{bits/pixel} \quad (2.68)$$

se relaciona con las siguientes parámetros,

M número de pixeles en la subimagen.

K número de vectores representativos, en potencias de 2.

n dimensión del vector.

b número de bits por vector componente representativo.

donde, $r = r_c + r_o$

2.1.4.4.2 Cuantificación vectorial de imágenes a color:

Las técnicas de cuantificación vectorial adaptiva se pueden extender a las imágenes a color representadas por las tres señales R, G, B. Antes de un procesamiento, estas señales van al sistema coordenado YIQ del NTSC. Esta transformación tiende a decorrelacionar a la información y compactar la mayor parte de la energía en la componente de luminosidad Y.

En el sistema NTSC las componentes de crominancia, I y Q son limitados en banda muy severamente así que para aplicar la codificación vectorial a estas componentes se las debe manipular y volcar la codificación a sus componentes alteradas I' y Q' modificadas en la resolución respecto a la de las señales I y Q originales, por un factor 2 de resolución.

2.1.4.5 Misceláneos:

Presumiblemente, los métodos y técnicas futuras orientadas a la compresión de imágenes se medirán con métodos y técnicas ya conocidas y su comportamiento será la medida de su eficacia frente a cualquier otro método existente, como es el caso de la Compresión de imágenes por medio del Truncamiento de Bloque (CTB) cuya calidad se mide con otras técnicas como la Transformada. Por otro lado, las futuras técnicas que se inventen serán orientadas a particulares aplicaciones, como es el caso del campo de la informática en donde el abaratamiento en el Hard-ware se logra con la optimización del Soft-ware, con los formatos de video PCX, TIFF, IMG, etc. Ambos tópicos se explican a continuación.

2.1.4.5.1 Codificación por Truncamiento de Bloque:

En estos párrafos se presentará una nueva técnica llamada Codificación por Truncamiento de Bloque (CTB) que ha sido desarrollada en la Universidad de Purdue a mediados de los años setenta. Esta técnica utiliza un cuantificador adaptivo no paramétrico de un bit sobre regiones locales de una imagen. El cuantificador que se muestra muy prometedor es uno que conserva una estadística global de la imagen. Para la explicación de como actúa este método, se tomará como muestra una imagen a la que se la dividirá en bloques de 4x4 pixeles, y se utilizará un cuantificador a 2 niveles. Como en general no es posible encontrar adecuados modelos de función de densidad para imágenes típicas, se han utilizado cuantificadores no-paramétricos para los esquemas de codificación.

Un cuantificador no-paramétrico se diseña de modo que minimice el error cuadrático medio (ECM) o el error absoluto medio (EAM). Se presenta el diseño de un cuantificador no-paramétrico que conserva los momentos de muestra (MP). Luego de dividir al cuadro en $n \times n$ bloques ($n=4$ para el ejemplo), los bloques son codificados individualmente, cada uno dentro de una señal de dos niveles. Los niveles para cada bloque se escogen de tal forma que los dos primeros momentos se conserven.

Sea $m=n^2$ y sean X_1, X_2, \dots, X_m los valores de los pixeles en un bloque del cuadro original. Entonces el primer y segundo momentos y la varianza de la muestra son, respectivamente:[6]

$$\bar{X} = \frac{1}{m} \sum_{i=1}^m X_i \quad (2.69)$$

$$\overline{X^2} = \frac{1}{m} \sum_{i=1}^m X_i^2 \quad (2.70)$$

$$\overline{\sigma^2} = \overline{X^2} - \bar{X}^2 \quad (2.71)$$

Como el diseño de cualquier cuantificador binario necesita de un umbral de comparación, se halla un umbral X_u , y dos niveles de salida, a y b, tal que:[6]

$$\begin{array}{ll} \text{Si } X_i \geq X_u & \text{Salida} = b \\ \text{Si } X_i < X_u & \text{Salida} = a \end{array} \quad (2.72)$$

para $i=1,2,\dots,m$. Para el primer cuantificador, se coloca a $X_u = \bar{X}$, este razonable criterio se puede modificar para mejorar la eficacia de la codificación. Los niveles de salida a y b se hallan de las siguientes ecuaciones: Sea q=número de X_i 's mayores que X_u entonces para conservar \bar{X} y $\overline{X^2}$:[6]

$$\begin{aligned} m\bar{X} &= (m-q)a + qb; \quad y, \\ m\overline{X^2} &= (m-q)a^2 + qb^2 \end{aligned}$$

Resolviendo para a y b:[6]

$$a = \bar{X} - \sigma \sqrt{\frac{q}{m-q}} \quad (2.73)$$

$$b = \bar{X} + \sigma \sqrt{\frac{m-q}{q}} \quad (2.74)$$

Cada bloque es luego descrito por los valores de \bar{X} , σ y un plano binario $n \times n$ consistente de 1's y 0's indicando si los pixeles están sobre o bajo X_u .

Asignando 8bits a \bar{X} y a σ , la codificación resulta en una tasa de 2 bits/pixel. El receptor reconstruirá la imagen bloque al calcular a y b con (2.73) y (2.74) y asignando estos valores de pixeles de acuerdo con el código en el plano binario. Debido a que los cálculos son relativamente simples y el dato a almacenar es pequeño, CTB es fácil de implementar. Un reciente estudio ha mostrado que CTB puede lograrse en un chip de circuito integrado.

En muchos esquemas de codificación se desea obtener tasas de entropía menores a 2bits/pixel. Una forma obvia de alcanzar esto es asignar menos de 8 bits a \bar{X} y a σ . La evidencia experimental muestra que codificando a \bar{X} con 6 bits y a σ con 4 bits introduce solo errores poco perceptibles. Ahora la tasa es de 1.63 bits/pixel. Se puede obtener mayor eficiencia con la cuantificación conjunta de \bar{X} y σ , utilizando 10 bits. A \bar{X} se le asigna más bits en bloques donde σ es pequeño y pocos bits donde σ es grande. Los cálculos de entropía indican que 0.85bits/pixel son suficientes para este plano. Tal pequeño mejoramiento no justifica el incremento de complejidad del sistema y la variabilidad incrementada al error que entraña una codificación eficiente.

(1) W. L. EVERSOLE, D. J. MAYER, F. B. FRAZKE & T.F. CHERR, "Investigation of VLSI Technologies for Image Processing". Proceedings: Image Understanding Work, Pittsburgh, PA, Noviembre 14-16 1972. Sponsored by the Defense Advanced Research Projects Agency p4ge. 181-186.

Colocando el umbral del cuantificador a \bar{X} , se remueve un posible grado de libertad para optimizar la codificación. Permitiendo ser al umbral una variable que logre una codificación que conserve no solamente el primer y segundo momento, sino el tercero también.

El tercer momento puede expresarse como:[6]

$$\bar{X}^3 = \frac{1}{m} \sum_{i=1}^m X_i^3 = \frac{1}{m} \sum_{i=1}^m Y_i^3 \quad (2.75)$$

El problema es entonces hallar a, b, y q a fin de conservar \bar{X} , \bar{X}^2 , y \bar{X}^3 (q define el umbral como aquel que especifica el número de X_i 's mayores que X_u):[6]

$$m\bar{X} = (m-q)a + qb; \quad y,$$

$$m\bar{X}^2 = (m-q)a^2 + qb^2; \quad y,$$

$$m\bar{X}^3 = (m-q)a^3 + qb^3$$

Teniendo las soluciones:[6]

$$a = \bar{X} - \bar{\sigma} \sqrt{\frac{q}{m-q}} \quad (2.76)$$

$$b = \bar{X} + \bar{\sigma} \sqrt{\frac{m-q}{q}} \quad (2.77)$$

$$q = \frac{m}{2} \left(1 + A \sqrt{\frac{1}{A^2 + 4}} \right) \quad (2.78)$$

donde,

$$A = \frac{3\bar{X}\bar{X}^2 - 2\bar{X}^3\bar{X}^3}{\bar{\sigma}^3} \quad (2.79)$$

Si $\sigma=0$, entonces $a=b=\bar{X}$.

2.1.4.5.2 Formatos de video(compresión a nivel informático):

Una imagen archivo (gráfico) en una resolución gráfica standard VGA de 640x480, tiene más de 300.000 píxeles. Si a cada pixel se le asigna a un byte, solo se podría archivar cuatro archivos gráficos en un disco flexible de alta densidad 5¼, 1.2MB. Para descargar el archivo a 2400 bits/seg, tomaría cerca de 20 minutos.

Si se quieren almacenar tantas imágenes como sea posible o transmitir las electrónicamente, se deduce que el mejor formato es aquel que almacena una imagen en la menor cantidad de bytes. Casi todos los formatos para archivos gráficos siguen el mismo esquema. Comienzan con una *cabecera*, la que identifica al tipo de archivo debido a que las extensiones del archivo por sí solas no son suficientes para garantizar esta información. Si un usuario renombra a algún archivo con una nueva extensión, un programa pudiera entrar en conflicto con la computadora.

La cabecera usualmente contiene otra información también. Por ejemplo, podría indicar: *el tamaño de la imagen (en bytes), número de colores utilizados, tipo de compresión de datos utilizado, tipo de computadora o tipo de aplicación utilizada para crear el archivo o el nombre original del archivo o su fecha de creación*. Por ejemplo, archivos IMG de imágenes monocromáticas siempre comenzarán con los tres enteros 1, 8 y 1. Esta señalización es seguida por la longitud de un patrón (usualmente 1), tamaño de cada pixel y tamaño de la imagen en píxeles. Otras cabeceras de archivo pueden ser mucho más complejas. Un archivo PCX a 256 colores, por ejemplo, tiene una cabecera de 128 bytes de largo más un byte marcador y 768 bytes de información de color al final del archivo. Siguiéndole a la cabecera muchos archivos gráficos almacenan a la imagen misma en un formato comprimido.

La técnica utilizada para esta compresión es un compromiso entre la longitud del archivo y el tiempo de compresión y descompresión que el archivo requiere. Los archivos PCX son muy fáciles de descomprimir pero tienden a ser relativamente grandes. Los archivos GIF por otra parte, son tan cortos como sea posible pero toman mucho más tiempo de cómputo al comprimirlos y descomprimirlos.

(a) El formato IMG:

Los archivos *monocromáticos* IMG, utilizados en aplicaciones como publicidad, comienzan con una cadena de seis bytes o de tres palabras llamada ID. Los bytes ID son seguidos por un patrón de longitud, el cual es de un tamaño fijo en bytes (o en pixeles) de todos los patrones incluidos en el archivo. Lo siguiente es la cabecera que define el tamaño de cada pixel de imagen. Si la imagen se proyecta a ser mostrada en una impresora láser en una resolución de 300-ppp (Puntos por pulgada), el tamaño del pixel se suele colocar a 85 micrones. Los programas de publicidad utilizan el tamaño del pixel a manera de medida de escalamiento. Los dos últimos valores en la cabecera son la longitud de una línea (en pixeles de imagen) y el número de líneas en la imagen. Los datos de la imagen están organizados por líneas, cada una de las cuales está dividida en uno o más paquetes de datos. El paquete más simple define hasta 127 pixeles de negro o blanco total utilizando una forma de codificación longitudinal (Sección 2.1.1.2). Un paquete de solo un color es siempre de una longitud de 1 byte y el byte no puede ser ni 0011 ni 8011. Si el byte es de cualquier otro valor, el bit más significativo es un blanco total(0) o un negro total(1). Los restantes bits contienen el número de veces que el pixel blanco o negro se repite. Por ejemplo, siete pixeles blancos se podrían codificar como 0711, mientras que siete pixeles negro se codificarían como 8711. A menudo, varias líneas de una imagen son idénticas, especialmente si representan un blanco, negro o un arreglo uniforme de estos a manera de patrón.

El segundo tipo de paquete, que comienza con los bytes 00H, 08H, 0FFH, le dicen al decodificador de imagen que la actual línea debe ser copiada un número específico de veces. A veces una imagen contendrá áreas de un patrón repetido, especialmente en imágenes que tienen un fondo punteado o de trazos.

Un archivo IMG puede tener solo una longitud patrón, la cual se define en la cabecera. Un patrón empaquetado viene con un byte de distinción 00H, el cual es seguido por el número de veces que el patrón se repite y luego los bytes en el patrón. Finalmente, una cadena empaquetada es de una imagen que no puede ser comprimida por ninguna de las técnicas que se explicarán luego. Después de un byte distintivo 80H, se almacena el número de bytes o píxeles en la cadena, seguido por los valores actuales de los píxeles. El formato IMG es una buena opción para el almacenamiento relativamente simple de dibujos y manuscritos. El siguiente es el formato de un archivo IMG:

HEADER:

```
WORD      0001H      ;Los primeros 3
WORD      0008H      ;bytes del
WORD      0001H      ;distintivo IMG
WORD      LONGTUD_PATRON
WORD      ANCHO_PIXEL
WORD      ALTURA_PIXEL
WORD      ANCHO_LINEA
WORD      ALTURA_IMAGEN
```

TIPOS_PAQUETES:

;Paquete de Corrida Sólida(Útilizado para hacer réplicas de negros y blancos sólidos - totales en una línea)

```
BYTE      80H      ;más la long. de negro total
BYTE      00H      ;más la long. de blanco total
```

;Paquete de Corrida de Patrón(Útilizado para hacer réplicas de blanco y negro)

```
BYTE      00H      ;Paquete ID
BYTE      Coo_Long      ;veces a duplicar
BYTE      Primer_Byte,....,Ultimo_Byte
```

;Replica de Paquetes(Útilizado para hacer réplicas de una línea)

```
BYTE      00H,00H,0FFH      ;Paquete ID
BYTE      Conteo_Réplicas
```

;Cadena Empaquetada(Útilizado para datos que no pueden ser comprimidos)

```
BYTE      80H      ;Paquete ID
BYTE      Long_Cadena
BYTE      Primer_Byte,....,Ultimo_Byte
```

(b) El formato PCX:

La popularidad del formato PCX es sorprendente, considerando su ineficiencia y dependencia del Hard-ware que le son propias. Los archivos PCX vienen en tres variedades: Monocromática, a 16 colore, y a 256 colores. Las variedades son similares una con otra, pero sus métodos de almacenamiento y compresión se reflejan de las tarjetas de video que estuvieron en boga cuando cada estándar fue desarrollado. Muchos de los campos en la cabecera son auto explicatorios. El byte OAH en el comienzo del archivo es el único byte distintivo, aunque un programa podría rechazar un archivo PCX si sus otros campos no tienen sentido. El siguiente es un formato de un archivo PCX:

```
CABECERA:
BYTE          creador          ;Siempre OAHn
BYTE          tipo_archivo
BYTE          esquema_codif    ;Siempre 1
BYTE          bits_pixel
WORD          xmin, ymin       ;Origen de la imagen
WORD          xmax, ymax       ;Dimensiones imagen
WORD          xres, yres        ;Resolucion
BYTES         (DUP48), paleta   ;Para los colores
BYTE          [reservado]
BYTE          planos_color
WORD          byte_linea
WORD          tipo_paleta
BYTE          DUP58, [llenado]
```

TIPOS_PAQUETES:

;Paquete de Corrida Sólida(Útilizado para hacer replicas de 1 byte)

```
BYTE          000h             ;mas la longitud
BYTE          byte_copia
```

;Cadena Empaquetada(Útilizado para datos que no pueden empaquetarse)

```
BYTE          Primer_byte      ;Cualquier bytes 000h
```

Los bits por pixel que se utilizan solo en imágenes a color, indican cuantos bits definen los valores de color. Los valores de **Xmin**, **Ymin**, **Xmax**, y **Ymax** muestran el tamaño de la imagen y su localización en la pantalla. Los valores de **Xres** y **Yres** muestran la resolución de la pantalla del dispositivo que creó el archivo. Los datos en un archivo PCX se almacenan utilizando un esquema inusual de codificación longitudinal.

Al igual que un archivo IMG, uno PCX es ordenado en filas. Contrario a un archivo IMG, uno PCX tiene solo dos empaquetamientos. Si los dos bits superiores de un byte son "seteados", el resto de bits son un índice de cuantas veces se repite el siguiente byte, mientras que si los dos bits superiores de un byte no se los ha "seteado", se utiliza el byte como está.

Un archivo PCX a 16 colores es ideal para un desempaquetamiento con una tarjeta de video EGA-y se produce cierto conflicto al tratar de mostrarlo en algún otro tipo de tarjeta (a menos que sea una EGA-compatible). Un archivo PCX a 256 colores se ha diseñado para un adaptador VGA. Debido a que no existe suficiente capacidad en la cabecera de 128 bytes para información de una paleta de colores (la tarjeta VGA necesita tres bytes para cada uno de los 256 registros de color), este dato es removido al final del archivo. A pesar de esto un archivo PCX a 256 colores es muy ineficiente debido a que cualquier pixel de valor 0COH o mayor se lo debe almacenar en dos bytes compuestos de una repetición de 1's seguido por el byte mismo.

(c) **El formato GIF:**

Si se quiere almacenar una gran biblioteca de imágenes, o si se quiere transmitir imágenes a través de un modem, se tiene que tomar en cuenta más el volumen de los archivos antes que cuan rápido y fácil se los puede presentar. Los archivos con formato GIF o *Formato de Intercambio de Gráficos* pueden, en general representar a una imagen en un tamaño mucho más pequeño que cualquier otro formato. El estándar GIF fue desarrollado por el *Servicio de Información CompuServe*, particularmente para crear un formato de imágenes que pudiese ser utilizado en cualquier computador. Una imagen GIF no necesita un tipo particular de computador o pantalla gráfica. Su Soft-ware gráfico es responsable de hacer que la imagen aparezca correctamente en la pantalla. La cabecera de un archivo GIF es muy sencilla.

Después de un distintivo, el computador que creó la imagen almacena el tamaño de esta en pixeles. Otra computadora puede leer esta información a fin de obtener la resolución correcta para su posterior presentación. La bandera global representa dos valores. Sus tres bits inferiores indican el número de colores utilizados. La fórmula es:[1]

$$\text{Colores} = 2^{[(\text{BANDERA} \& 07H)+1]}$$

donde el operador & es el "AND" lógico entre esos dos bytes. El bit más significativo, cuando se lo "setea", indica que los colores se almacenan en un mapa global de colores. El color de fondo indica que color debería utilizar un programa para decodificar la imagen entrante a este. El siguiente es el formato de un archivo GIF:

```

CABECERA:
    BYTE      "GIF87a"          ;Señalización
    WORD      Ancho_Pantalla
    WORD      Altura_Pantalla
    BYTE      Bandera_Global
    BYTE      Color_Fondo
    BYTE      0                  ;Para chequeo
    BYTE      Mapa_Color       ;3 bytes/color

TIPOS_PAQUETES:

;Imagen Empaquetada:
    BYTE      ','                ;Define una imagen
    WORD      Xmin                ;Localización
    WORD      Ymin
    WORD      Ancho_Imagen       ;Tamaño
    WORD      Altura_Imagen
    WORD      Byte_Band_Local
    BYTE      Mapa_Local_Color

;Paquete de archivo terminado:
    BYTE      ';'

;Paquete de extensión GIF:
    BYTE      '!'

```

Un archivo entrelazado es ideal si se tiene un visor lento o si un computador muestra el archivo tal como viene del modem. Después de la primera o segunda pasada se debería conocer si se quiere o no esperar al resto del archivo. Aunque todos los detalles del archivo no aparecerán hasta la última pasada, se puede almacenar la transmisión de la imagen hasta donde se quiera, si se decide parar la imagen tempranamente.

BMP(Bit MaP): Este es uno de los más populares formatos de archivos imagen dentro de *WINDOWS*. Como un archivo PCX, este es un registro de bits presentes en un archivo. Como tal, estos no pueden ser rescatados si se cambian las resoluciones de la pantalla. Los archivos BMP son rápidos de cargar y representar en *Windows*, pero también son inflexibles. Si se usa una pantalla con resolución super VGA (1024x840 pixeles) en *Windows* y se nota una distorsión con aplicaciones con necesidades no tan grandes, entonces se tiene problemas con gráficos Bit-Mapeados.

MTF(MetaFiles): Este es otro formato de *WINDOWS*. Los Meta-Archivos son extremadamente flexibles en comparación con los Archivo BMP. En esencia estos son una forma de manipular a los dispositivos de interfaz gráfica (GDI) de *Windows* para que dibujen a una imagen. Los GDI pueden compensar las diferentes resoluciones de pantalla, capacidades de color, etc.; así un meta-archivo es casi siempre mostrado correctamente. Si lo que se quiere es involucrarse seriamente con archivos imágenes, se necesitará un programa para traducir un tipo de archivo a otro.

En el ANEXO 2 se hallan ejemplos ilustrativos de codificación por medio de la técnica OBB-unidimensional (Anexo 2j), por medio de la técnica del Truncamiento de bloque (Anexo 2k), y muestras de formatos de video PCX, GIF, y BMP (Anexo 2l).

2.2 COMPARACION CUANTITATIVA Y CUALITATIVA DE LOS METODOS

Una confrontación de eficiencias entre diferentes métodos y técnicas de compresión es necesaria para obtener criterios para su selección cuando se requiera utilizar una técnica u otra según la aplicación que se busque. En el campo de la compresión de video, las comparaciones entre sus métodos se las realiza por medio de análisis en el tiempo y en el espacio, tomando en cuenta además criterios de transmisión y manipulación digital como reducción de ancho de banda, reduc-

ción de entropía, reducción de tasas de error, optimización de la relación Señal/Ruido, entre otros. A continuación se muestra una comparación cuantitativa y cualitativa de los diferentes métodos y técnicas de compresión. Las comparaciones se realizan tomando en cuenta como ellos actúan al aplicarlos sobre imágenes de prueba típicas o ya estandarizadas por el CCITT (Comité Consultivo Internacional de Telefonía y Telegrafía) y el IEEE (Instituto de Ingenieros Eléctricos y Electrónicos).

La identificación de un criterio de fidelidad para el diseño de sistemas de compresión del video ha sido una persistente dificultad para los investigadores. El criterio más ampliamente utilizado es el del error cuadrático medio:[10]

$$e_{\alpha}^2 = \frac{1}{N^2} \sum_{n=1}^N \sum_{m=1}^N [r(n,m) - i(n,m)]^2 \quad (2.80)$$

donde $r(n,m)$ es el pixel de una imagen reconstruida, y $i(n,m)$ es el pixel original

Este criterio, o funciones de él, tales como una relación señal a ruido (SNR), definida como:[10]

$$SNR = 10 \log_{10} \frac{\sigma_i^2}{e_{\alpha}^2} \quad (2.81)$$

donde σ_i^2 es la varianza de la imagen original, o como:[10]

$$SNR = 10 \log_{10} \frac{(255)^2}{e_{\alpha}^2} \quad (2.82)$$

(para imágenes originales cuantificadas con 8bits/pixel), se utiliza no solamente para la evaluación de las reconstrucciones de imágenes sino también como un criterio de diseño.

Un ejemplo muy conocido es la solución de las ecuaciones de Max-Lloyd a fin de encontrar la transición óptima y los niveles de reconstrucción de un cuantificador.

El criterio del error cuadrático medio no toma en cuenta mecanismos visuales humanos. Así no es necesario la métrica óptima para la evaluación de sistemas de compresión de imágenes, aunque esto es computacionalmente tratable. El problema de la evaluación también existe en el campo de la identificación de sistemas, en el cual la tarea es determinar un adecuado modelo paramétrico para representar un proceso. Por ejemplo, el tamaño de la varianza en el error (residual) de modelación en la identificación del sistema corresponde al tamaño del error de reconstrucción cuadrático medio en compresión de video.

2.2.1 DESEMPEÑO DE LA TECNICA PREDICTIVA:

Las técnicas predictivas, casi siempre se comparan con la técnica básica de codificación digital: la MIC. Las diferencias existentes entre esta técnica y otras (de la misma línea) se cuantificarán en eficacia y potencia.

2.2.1.1 Comparación MIC versus MICD:

Considerando las ecuaciones (2.8) a (2.12). Reescribiendo (2.9) como:[27]

$$u(n) = u''(n) + e(n) \quad (2.83)$$

y restando (2.83) de (2.10), se obtiene:[27]

$$\delta u(n) = u(n) - u'(n) = e(n) - e'(n) = q(n) \quad (2.84)$$

Así, desde el punto de vista de la codificación el error en la secuencia de entrada es exactamente igual a $q(n)$, el error de cuantificación en $e(n)$.

Con un predictor razonable el valor cuadrático medio de la señal diferencia $e(n)$ es mucho menor que $u(n)$. Esto significa, para el mismo error de cuantificación cuadrático medio, $e(n)$ requiere menos bits de cuantificación que $u(n)$. Si se denotan los valores cuadráticos medios del error de cuantificación $q(n)$ y el error de predicción $e(n)$ por σ_q^2 y σ_e^2 , respectivamente, y notando que (2.84) implica que:[27]

$$E\{[\delta u(n)]^2\} = \sigma_q^2 \quad (2.85)$$

la tasa mínima alcanzable por MICD está dada por la fórmula de la tasa de distorsión:[27]

$$R_{MIC} = \frac{1}{2} \log_2 \frac{\sigma_u^2}{\sigma_q^2}, \quad \text{bits/pixel} \quad (2.86)$$

Para la misma distorsión, $\sigma_q^2 \leq \sigma_u^2$, la reducción en la tasa de MICD comparada con la de MIC es:[27]

$$R_{MIC} - R_{MICD} = \frac{1}{2} \log_2 \frac{\sigma_u^2}{\sigma_u^2} \quad \text{bits/pixel} \quad (2.87)$$

Esto demuestra que la compresión alcanzada depende de la reducción de la tasa de varianza ($\sigma_u^2 \leq \sigma_u^2$), esto es, la habilidad de predecir $u(n)$ y, por lo tanto, la redundancia inter muestra en la secuencia. También, la naturaleza recursiva de MICD requiere de un predictor causal.

Para mínima varianza del error de predicción, el predictor óptimo es la media condicional $E\{u(n)/u'(n)\}$. Cuando el error de cuantificación es pequeño ($<10^3$), R_{MICD} alcanza un valor mínimo. La relación SNR correspondiente a σ_q es:[27]

$$(SNR)_{MICD} = 10 \log_{10} \frac{\sigma_u^2}{\sigma_q^2} = 10 \log_{10} \frac{\sigma_u^2}{\sigma_u^2 f(B)} \quad (2.88)$$

donde $f(B)$ es la función de distorsión cuadrática media del cuantificador para una entrada de varianza unitaria y B son los bits de cuantificación. Para un igual número de bits utilizados, la ganancia en SNR de MICD sobre MIC es:

$$(SNR)_{MICD} - (SNR)_{MIC} = 10 \log_{10} \frac{\sigma_u^2}{\sigma_u^2} \quad (\text{dB}) \quad (2.89)$$

El incremento en SNR es de 6dB ($R_{MIC} - R_{MICD}$), esto es, 6dB por bit de compresión disponible. De estas medidas, se nota que en el desempeño de los codificadores predic-

tivos depende del diseño de los cuantificadores y predictores. Por simplicidad el predictor se diseña sin considerar los efectos de la cuantificación. Esto significa que la regla de predicción óptima estimada para $u'(n)$ se aplica para estimar $u''(n)$. Por ejemplo, si $u'(n)$ está dada por:[27]

$$u'(n) = \sum_{k=1}^N a(k) u(n-k) \quad (2.90)$$

entonces el predictor MICD se diseña como:[27]

$$u''(n) = \sum_{k=1}^N a(k) u'(n-k) \quad (2.91)$$

Para dos (o más) dimensiones esta aproximación requiere encontrar las reglas más óptimas de predicción causales. Nótese que el codificador MICD resulta ser *no-lineal* aún con la predicción lineal de (2.91). Sin embargo, el decodificador será ahora un filtro lineal.

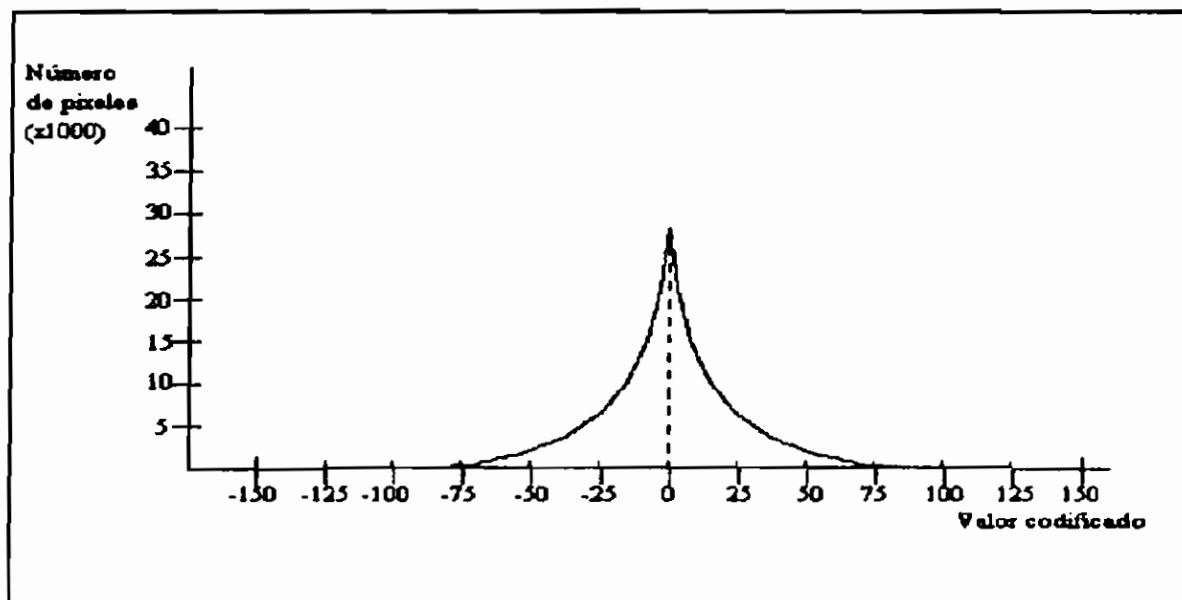


Figura 2.28 Histograma del error de predicción.

La Figura 2.28 muestra un histograma de la señal típica de error de predicción, en el eje de las abscisas se encuentra el valor codificado de este error antes de la cuantificación.

El error de predicción toma grandes valores cuando se acerca a los bordes de una imagen. A menudo el error de predicción se modela como una secuencia no correlacionada con una distribución de probabilidad Laplaciana, como la de (2.92):

$$p(\epsilon) = \frac{1}{\beta^2} e^{-\frac{\sqrt{2}}{\beta} |\epsilon|} \quad (2.92)$$

donde β^2 es su varianza (Ecuación 2.16).

2.2.1.2 MD, MICD línea a línea y MICD bidimensional:

En la Modulación Delta (Sección 2.1.2.2); y la relación (2.13) muestra la diferencia entre una muestra original y una decodificada. La predicción en este caso se plantea, tomando en cuenta el coeficiente de fuga relación (2.14):[27]

$$u(n) = \rho u(n-1) + e(n) \quad (2.93)$$

con la aproximación de que:[27]

$$E\{e(n)\} = 0$$

La SNR de la señal reconstruida está dada por,[27]

$$(SNR)_{MD} = 10 \log_{10} \frac{1 - (2\rho - 1) f(1)}{2(1 - \rho) f(1)} \text{ (dB)} \quad (2.94)$$

Asumiendo una distribución Gaussiana del error de predicción y utilizando cuantificación uniforme, con una fuga del 5% ($\rho=0.95$), la SNR es de 12.8dB, lo que es una mejora de 8.4dB sobre el MIC a 1 bit/pixel. Esto significa una compresión de 2.5, o un ahorro de 1.5 bits/pixel.

A fin de obtener resultados cuantitativos en MICD línea a línea, se procede de idéntica forma: Las ecuaciones (2.93) y (2.94) indican que el SNR de la modulación delta puede mejorarse al incrementar el coeficiente de fuga (reducir la fugas), lo cual puede hacerse al incrementar el ritmo de muestreo de la salida del cuantificador. Al mismo tiempo, sin embargo, el ritmo del flujo de datos también se duplica.

Mejores resultados se pueden obtener al utilizar técnicas adaptivas o al incrementar el número de bits de cuantificación. Del modelo de (2.93), la SNR de un sistema MICD a B bits, la salida se puede estimar de:[27]

$$(SNR)_{MICD} = 10 \log_{10} \frac{1 - \rho^2 f(B)}{(1 - \rho^2) f(B)} \text{ dB} \quad (2.95)$$

Para una fuga del 5%, aproximadamente existe una mejora de 6 a 10 dB en el SNR sobre MIC a tasas de 1 a 3 bits/pixel. Alternativamente, para pequeños niveles de distorsión $f(B) \approx 0$, la reducción de la tasa sobre MIC es de:[27]

$$R_{MIC} - R_{MICD} = \frac{1}{2} \log_2 \frac{1}{1 - \rho^2}, \text{ bits/pixel} \quad (2.96)$$

Esto significa, por ejemplo que la SNR de 6bits MIC se puede alcanzar con MICD línea a línea de 4bits para una fuga del 3%.

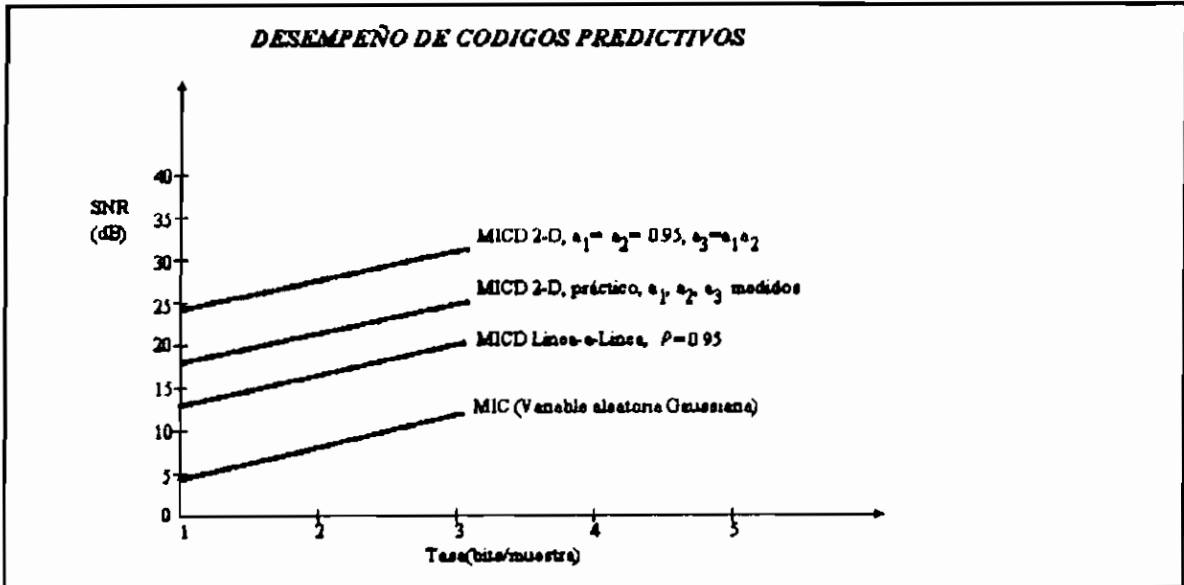


Figura 2.29 Desempeño de códigos predictivos.

La Figura 2.29 muestra los SNR teóricos versus la tasa de entropía (bits/muestra) de MICD bidimensional por medio de un modelo como el de (2.23), considerando:[27]

$$\begin{aligned} a_1 &= \rho_1, & a_2 &= \rho_2, & a_3 &= -\rho_1 \rho_2, & a_4 &= 0 \\ \beta^2 &= \sigma^2 (1 - \rho_1^2) (1 - \rho_2^2) \end{aligned} \quad (2.97)$$

y las ecuaciones (2.24), (2.25), (2.26) y (2.27) con $a_4 = 0$.

Además demuestran comparaciones hechas con MICD línea a línea unidimensional, y MIC. Notese que la modulación delta (MD) es la misma que una MICD de 1bit es esta curvas.

En la práctica, el MICD bidimensional no llega a los 20dB de mejoría sobre el MIC como se esperaría para campos aleatorios con los parámetros de (2.97). Esto es debido a que el modelo separable de la covarianza bidimensional es sobradamente optimista sobre la varianza del error de predicción. La subjetiva calidad de una imagen y su tolerancia a canales con error puede mejorarse por medio de predictores bi-dimensionales. Generalmente un codificador MICD de 3 bits/pixel puede dar imágenes de muy buena calidad. Con una codificación Huffman, el ritmo del flujo de salida de un cuantificador a 3 bits en un MICD bi-dimensional puede reducirse de 2 a 2.5 bits/pixel en promedio.

2.2.1.3 MICD Adaptivo-MICD No Adaptivo:

Al aplicar codificación predictiva adaptiva a la intensidad de la señal de video, se obtuvieron imágenes reconstruidas de alta inteligibilidad y buena fidelidad. La Figura 2.30 muestra a cuatro imágenes reconstruidas a las cuales se las codificó utilizando un cuantificador de dos niveles y diferentes tamaños de ventanas (bloques) así como también diferentes órdenes de predictores.



Figura 2.30 Cuantificación de la intensidad a tres niveles.

La siguiente es una tabla con los SNR resultantes[9], se puede observar que el uso de pequeñas ventanas (bloques de pixeles) y altos órdenes en los predictores nos dan un alto SNR. Sin embargo, para el caso de las ventanas de 16x16 se requiere una tasa de entropía cuatro veces más alta que para las de 32x32:

Imagen	Orden del Predictor	Tamaño de la Ventana	SNR (dB)
a	3	32x32	30.6
b	8	32x32	31.1
c	3	16x16	31.2
d	8	16x16	32.2

Las imágenes correspondientes al error de cuantificación se presentan a continuación en la Figura 2.31. Estas se han formado por un "mapeo" de las diferencias entre las imágenes originales y las reconstruidas en el rango original de 0 a 255 píxeles.

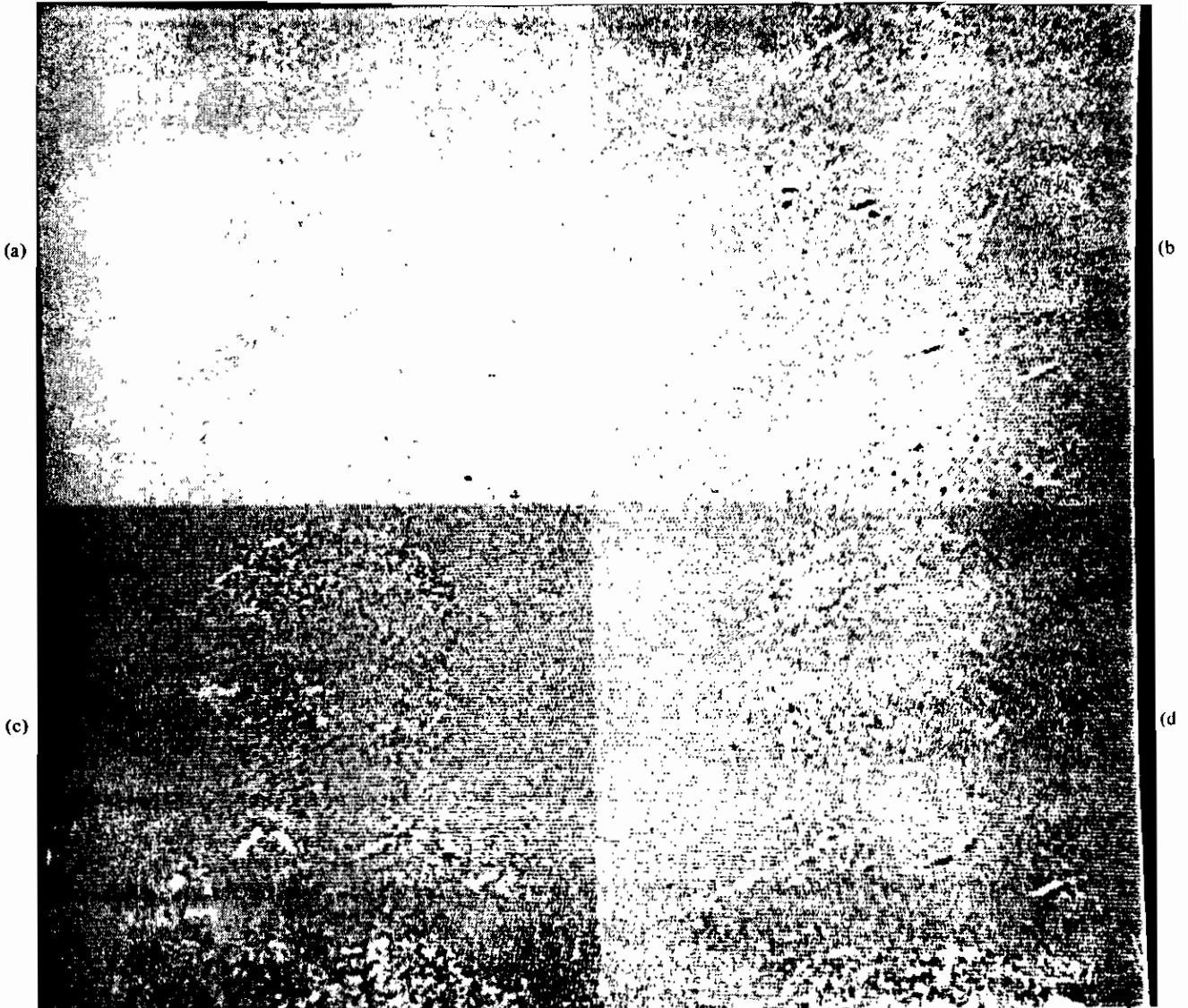


Figura 2.31 Imágenes error de cuantificación.

La Figura 2.32 muestra a un grupo de cuatro imágenes reconstruidas las cuales se obtuvieron al fijar el orden del predictor a $P=3$ y el tamaño de la ventana a $M=32$, al usar un cuantificador a tres niveles, centrado, y variando su umbral a fin de que la entropía sea de 0.7, 0.8, 0.9, y 1 bit/píxel.



Figura 2.32 Cuantificación centrada a tres niveles.

Una mejor observación a las imágenes de la Figura 2.32, muestra que los efectos de contorneado se hacen más obvios al reducirse los niveles de entropía. Consistente con esta observación es el hecho de que el SNR viene a ser más pequeño a bajas entropías, como se muestra en la siguiente tabla:[9]

Imagen	Entropía (bits/pixel)	SNR (dB)
a	0.707	30.3
b	0.804	31.6
c	0.900	32.6
d	1.002	33.4

Comparando los grupos de imágenes presentadas anteriormente, se llegan a las siguientes conclusiones:

- (a) Para una misma tasa de flujo de bits de ≈ 0.7 bits/píxel, orden de predictor, y tamaño de ventana, con los cuantificadores de dos y tres niveles se llega al mismo SNR. Más aún, en el caso de el cuantificador a tres niveles, la imagen reconstruida parece tener bordes afilados, menos ruido por granularidad, pero algunos efectos de contorno.
- (b) Para una misma tasa de flujo de bits de 1 bit/píxel usando palabras código de longitud variable para los tres niveles y de longitud fija para los cuantificadores de dos niveles, el cuantificador de tres niveles da imágenes casi libres de contorneo y algunos aspectos de alta fidelidad, tales como bordes afilados, menos ruido de granulación y alto SNR (aproximadamente 3 dB más).

Todos los resultados anteriormente discutidos se refieren a la MICD Adaptiva (Sección 2.1.2.6.5). Una cuestión natural a preguntarse en este punto es: Cuánta calidad de imagen es sacrificada por el uso de un simple MICD sin adaptación del predictor y/o el cuantificador?

Primero, se debe considerar el desempeño de MICD con cuantificación adaptiva. El predictor se fija, y sus parámetros se obtienen al aplicar una predicción lineal bidimensional a la imagen entera, la cual se puede considerar como una ventana de 256x256 píxeles. Los SNR's resultantes fueron aproximadamente los mismos (cerca de 1dB menos) como en el caso de MICD Adaptivo. Sin embargo, la calidad subjetiva de la imagen MICD reconstruida fue inferior debido al ruido por granularidad, el cual afecta particularmente a la tasa de entropía que está debajo de 1bit/píxel con un cuantificador a tres niveles.

prácticas (con componentes de color), de otro lado, aclaran que MICD da una eficiencia de codificación de 1 a 3dB sobre la CTH. Otro reporte recordatorio de la señal compuesta (-PAL), basado en una evaluación exclusiva, muestra que el MICD provee mejor calidad para imágenes con un relativamente alto ritmo de transmisión de más de 5 bits/pixel, aún si se toman en cuenta los bits adicionales que se añaden para la corrección de errores. De acuerdo a la experiencia de investigadores en pruebas con MICD y CTH, en el caso de una tasa de 5 bits/pixel o más, los dos métodos son capaces de proporcionar una imagen de alta calidad sin tener mucha diferencia de la imagen original. Al igual que para las técnicas anteriores, la relación SNR se puede definir:[18]

$$SNR = 20 \log \frac{S_{p-p} \times 0.7}{\frac{1}{IJ} \sqrt{\sum_{i=1}^I \sum_{j=1}^J (s_{ij} - s'_{ij})^2}} \quad (2.98)$$

donde s_{ij} y s'_{ij} representan los niveles de la imagen original y la imagen reconstruida, respectivamente. En el caso de imágenes ordinarias, 2, 5 y 15 bits/pixel no hay mucha diferencia entre MICD y CTH, aunque existe una tendencia para mejorar a la calidad de imagen por parte de MICD a tasas de 3 y 4 bits/pixel, y por medio de CTH a una tasa de 2 bits/pixel. Por otra parte, en el caso de las imágenes a color teniendo alta potencia de señal de crominancia y muy alta correlación de las señales de luminancia y crominancia, MICD muestra mejores resultados de evaluación a cualquier entropía (bits/pixel).

2.2.2.1 Comparación transformada versus transformada:

Las imágenes utilizadas para este estudio fueron imágenes de varios niveles de gris, y de un tamaño de 256x256 píxeles, con 8 bits/pixel. Se utilizaron cinco imágenes separadas: "Avión" (un F-16 en vuelo); "Comida" (una fotografía de un plato con pimientos verdes); "Lena" (un retrato de una chica); "Oficina" (una fotografía de una oficina); "Baboon" (un acercamiento al rostro de un mandril).

Las imágenes del avión, la comida, y Lena tienen una cantidad relativamente pequeña de bordes, y contiene muchas regiones de pixeles de intensidad constante. El resto de imágenes tienen una gran cantidad de bordes, regiones de texturas muy variadas de color. En este estudio se utilizaron reglas de distribución Gamma, Laplacianas y Gaussianas, usadas para modelar los coeficientes transformados. Dos sub-bloques de tamaños distintos se pusieron a prueba, de 8x8 y 16x16 pixeles. Finalmente, el número promedio de bits por pixel (bits/pixel) se hizo variar de 1 a 4. Los resultados se tabularon en términos de la relación de señal/ruido de cuantificación (SNR), y expresada en dB. Si se nota a $I(m,n)$ como un elemento de una imagen $M \times N$, y sea $I_q(m,n)$ denota su representación codificada, entonces la SNR en dB se define como: [8]

$$SNR = 10 \log_{10} \frac{\frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I^2(m,n)}{\frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [I(m,n) - I_q(m,n)]^2} \quad (2.99)$$

Las intensidades del pixel varían entre 0 y 255 donde 0 representa al negro, y 255 el blanco. El denominador del argumento de la función logarítmica es el error de cuantificación cuadrático medio.

El error de cuantificación cuadrático medio es ampliamente utilizado como una medida de la calidad de la imagen codificada. Este error tiene algunas ventajas: primero, un error pequeño usualmente implica una imagen aceptable, segundo, el error es fácil de computar, tercero, el error tiene una intuitiva interpretación de ser la potencia del error de la señal, y finalmente, no existe una interpretación universal de la medida de distorsión utilizada en estas aplicaciones. La Figura 2.33 presenta los resultados para la imagen del mandrill (BABOON); se utilizaron sub-bloques de 16x16 para obtener esta imagen.

La Figura 2.34 presenta los resultados para la imagen de "Lena"; un cuantificador Gamma y sub-bloques de 8x8 se utilizaron para obtener esta imagen. La siguiente relación muestra una buena guía para el diseño de codificadores basados en transformada:[8]

$$SNR_{TDC} > SNR_{TDCC} > SNR_{TDH} > SNR_{TDF} \quad (2.100)$$

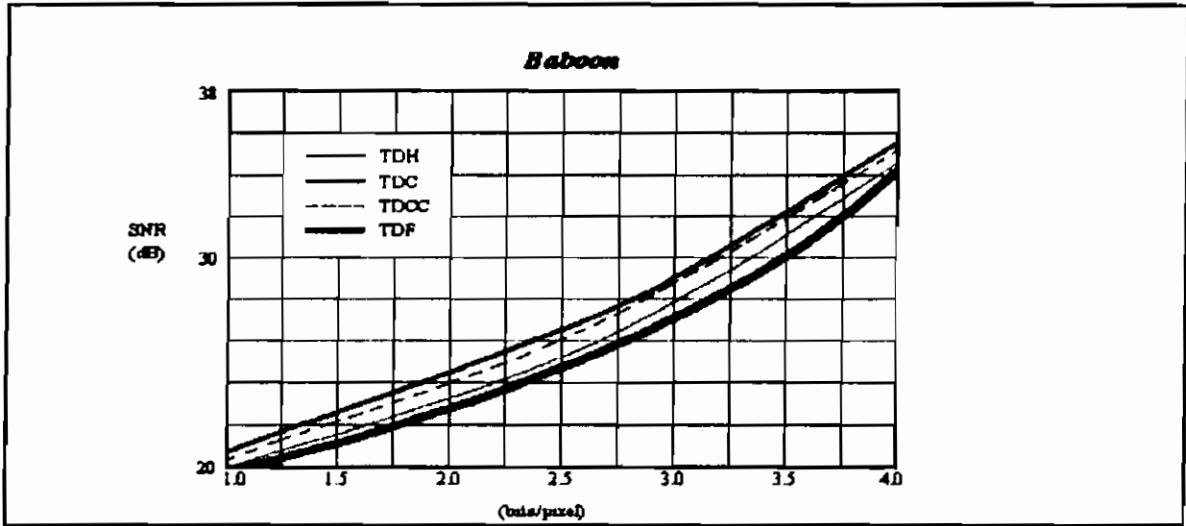


Figura 2.33 Comparación de transformaciones-BABBOON.

La Figura 2.35 compara el desempeño de las transformadas para bloques de 16x16 pixeles. El desempeño de la transformada Coseno es superior al de las otras transformadas rápidas, y es casi indistinguible de la transformada KL.

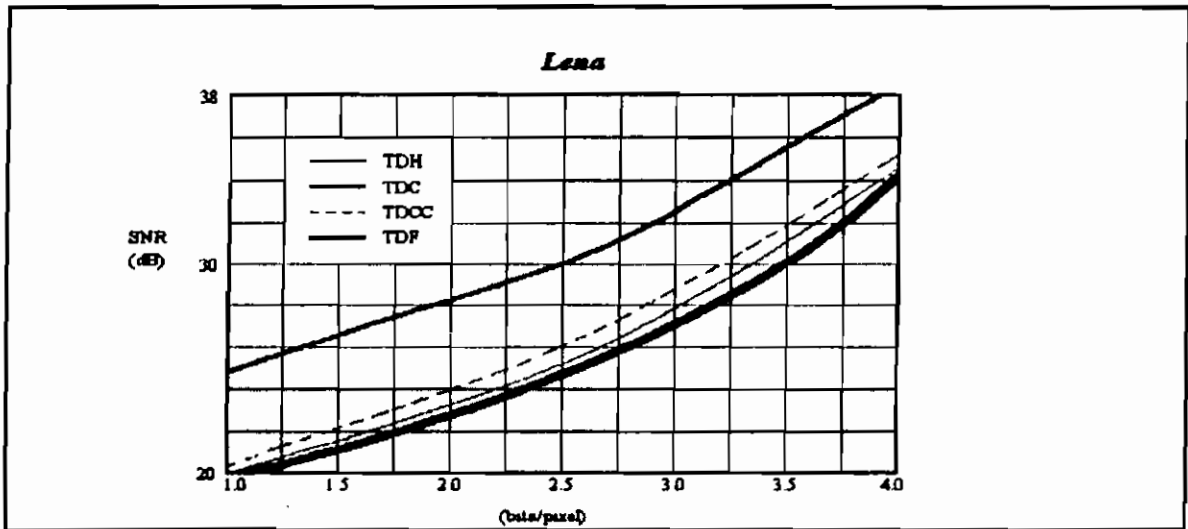


Figura 2.34 Comparación de transformaciones-LENA.

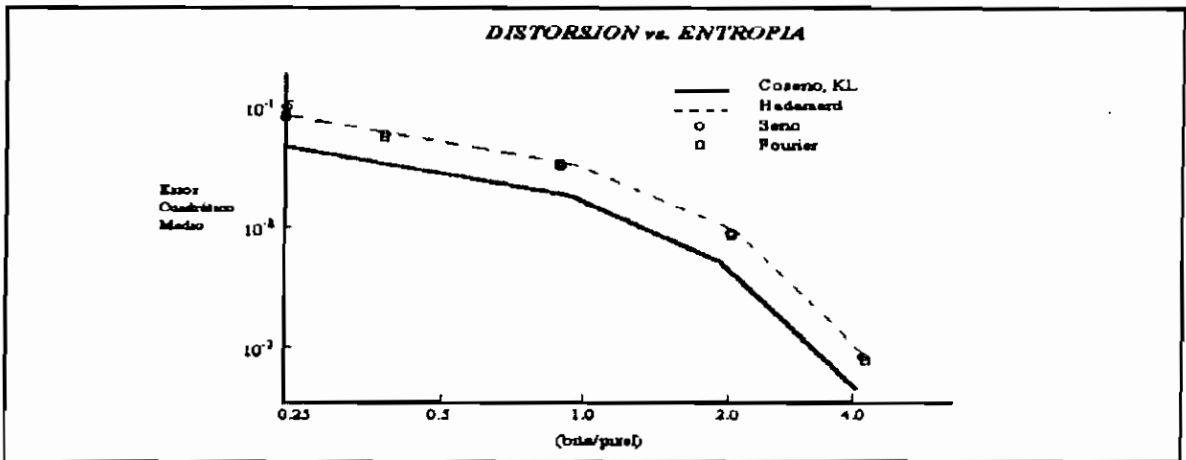


Figura 2.35 Distorsión vs. Entropía.

La siguiente tabla muestra ejemplos de valores de SNR a diferentes tasas de bits/pixel. [27]

Tamaño del Bloque	Tasa (bits/pixel)	SNR (dB)				
		kl	Coseno	Seno	Fourier	hadamard
8x8	0.25	11.74	11.66	9.08	10.15	10.79
	0.50	13.82	13.76	11.69	12.27	12.65
	1.00	16.24	16.19	14.82	14.99	15.17
	2.00	20.95	20.89	19.53	19.73	19.82
	4.00	31.51	31.54	30.17	30.44	30.49
16x16	0.25	No disponible	12.35	10.37	10.77	10.99
	0.50	No disponible	14.25	12.82	12.87	12.78
	1.00	No disponible	16.58	15.55	15.52	15.27
	2.00	No disponible	21.26	20.37	20.24	20.01
	4.00	No disponible	31.90	31.00	30.86	30.69

2.2.3 DESEMPEÑO DE LA TECNICA DE CODIFICACION DE GRAFICOS DE DOS TONALIDADES:

Ya se ha hablado (Sección 2.1.5.2) sobre las bondades que existen en utilizar un método de compresión de este tipo de imágenes (codificación binaria) con respecto a otro. Se explicó la estrategia que estos métodos utilizan en su desempeño de compresión. Así mismo, son famosas las imágenes del CCITT (Anexo 1) de prueba con las que se puede realizar un sondeo de calidad de compresión entre los métodos. Dichas imágenes de prueba (8 documentos de facsímil del CCITT) se prestan para realizar una comparación entre SNR's que se obtendrían al restaurar a estas imágenes comprimidas.

2.2.4 DESEMPEÑO DE LA TECNICA DE CODIFICACION DE IMAGENES A COLOR:

Para este análisis se empleó una imagen de prueba ("Girl"), la que fue codificada utilizando el método del filtraje zonal en el cual todas las muestras transformadas fuera de una zona central son descartadas. Se aplicó una cuantificación uniforme a 6 bits a los coeficientes restantes. Los ritmos binarios resultantes fueron de un total de 3.75 bits/pixel. La tabla siguiente muestra los resultados de la codificación por transformada de los componentes de una señal de color, con filtraje zonal y cuantificación MIC:[11]

Componentes	RGB	YIQ	YUV	YIQ	YIQ	YIQ
Transformada	Fourier	Fourier	Fourier	Hadamard	Hadamard	K-L
Cuantificación (bits)	R=1.25 G=1.25 B=1.25	Y=3.0 I=0.75 Q=0.75	Y=3.0 U=0.75 V=0.75	Y=3.0 I=0.75 Q=0.75	Y=3.0 I=0.75 Q=0.75	I=1.0 I=0.75 Q=0.75
Tasa Total (bits/pixel)	3.75	3.75	3.75	3.75	3.75	1.75
Error Cuadrático Medio Total	1.224	0.638	0.674	0.756	0.724	0.592

Otras imágenes han sido procesadas por medio de la transformada KL (Sección 2.1.3.a) aplicada sobre la señales de luminancia, con una codificación umbral de tasa resultante igual a 10:1, es de 1.0 bits para la señal de luminancia y de 0.75 bits para las componentes I y Q, dando un total de 1.75 bits/pixel. La transformación inclinada (Sección 2.1.3.3) también se la aplicó a la codificación de las señales de crominancia. Este tipo de transformación permite una óptima codificación casi tan buena como la transformada KL en términos de compactación de la energía de la señal en los componentes transformados de menor orden. Los componentes RGB de la imagen original se pueden primero transformar individualmente por Fourier y entonces combinados para formar una transformación tripartita y así obtener los coeficientes correspondientes a las señales de crominancia y luminancia.

Los componentes resultantes de Fourier se convierten en señales de amplitud y fase antes de ser codificadas. Los cuadros entonces son entonces codificados a tasas de 0.55 a 1.2 bits/muestra, pero la desventaja está en que se pierden algunos detalles en las imágenes reprocesadas. Los resultados de posteriores investigaciones sugieren que la transformación inclinada pudiera ser más eficiente que la de Hadamard, pero sus "codecs" (codificador-decodificador) bidimensionales se acentúan en la afinidad de la transformada vertical de Hadamard para la reversión de fase línea-a-línea de la señal subportadora de NTSC.

Esto podría ser presumiblemente ineficiente para la subportadora de PAL, o para la señal SECAM la cual utiliza diversidad de subportadoras en líneas adyacentes.

En términos de la ventaja en la transmisión del error de la codificación por transformada, los experimentos muestran que se obtiene solamente una degradación ligera en la probabilidad de error, de 10^{-6} y se considera usable aún para 10^{-3} .

Para los patrones de error más serios de la transformación de Hadamard de 32 elementos intra-línea, ya se ha establecido que una probabilidad de error de 10^{-6} fue definitivamente perceptible aunque no produjo disturbios, mientras que el umbral de percepción estuvo alrededor de una probabilidad de error de 10^{-7} . MICD por otra parte, es más sensible a los errores por cerca de dos órdenes de magnitud. De acuerdo con los resultados subrayados anteriormente, para llegar al máximo rendimiento en la codificación de componentes de color, podría darse una separación de la señal para estar acorde con los requerimientos de codificación, para las propiedades de cada componente y la relación entre ellos. Sin embargo, algunas de estas propiedades se las ha reconocido en diseños de componentes análogos existentes, y consecuentemente el margen resultante pudiese no ser tan largo.

2.2.5 DESEMPEÑO DE LA CUANTIFICACION VECTORIAL EN LA CODIFICACION DE IMAGENES:

Con la teoría ya cubierta en la sección 2.1.4.4 sobre codificación vectorial, se puede analizar su ejecución en lo que respecta a la codificación de imágenes.

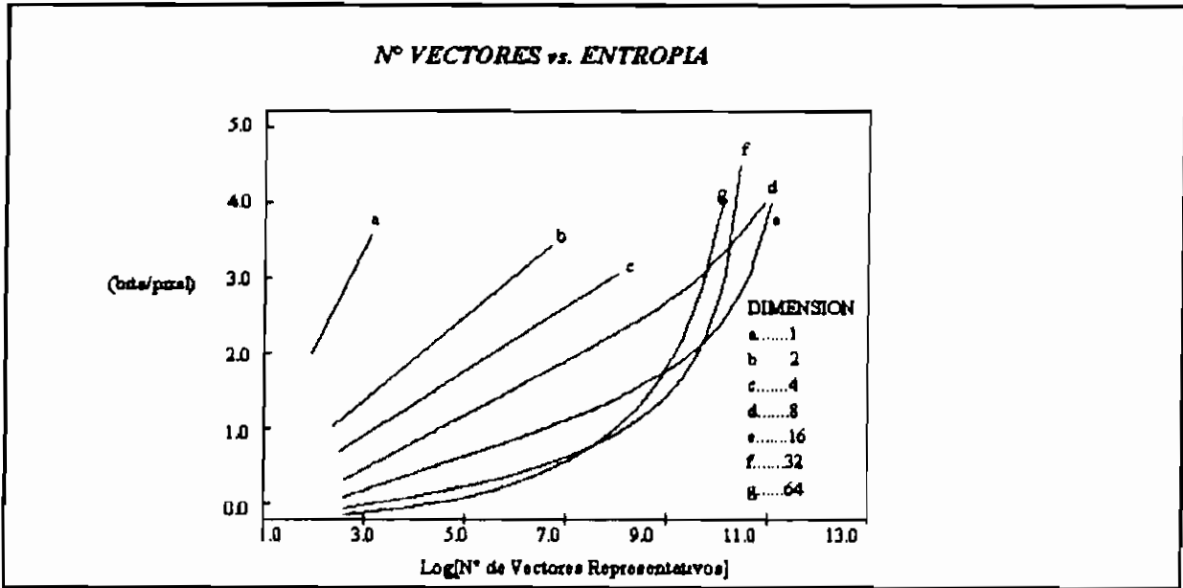


Figura 2.36 Número de vectores vs. Entropía.

La tasa de entropía r (Ecuación 2.68), como una función de K (número de vectores representativos: 2, 4, 8, 16, 32, 64, 128, ...etc), se muestra en forma gráfica en la Figura 2.36. Aquí $b=8$, $M=256 \times 256$, y n (dimensión del vector) varía de 1 a 32.

La acción del esquema de codificación por cuantificación vectorial fue comparada a la de los esquemas de codificación por transformada de Hadamard y Coseno. La imagen de prueba fue particionada en bloques no sobrepuestos de 16×16 y cada bloque fue transformado utilizando o la transformada coseno bidimensional o la transformada de Hadamard. Los coeficientes transformados fueron cuantificados escalarmente utilizando un número variable de bits escogidos de acuerdo a la energía promedio por coeficiente. El coeficiente DC fue simplemente cuantificado linealmente a 8bits, mientras que los componentes AC se cuantificaron por medio de un cuantificador MIC y modelados por medio de una distribución Laplaciana.

Los resultados obtenidos por medio de las dos transformadas (Coseno y Hadamard) lanzan a Hadamard como una muy poco notable transformación. El esquema de codificación por cuantificación vectorial adaptivo nunca sobrepasó a la transformada bloque Coseno, aunque se vuelve comparable a ciertas tasas binarias (bits/pixel) dependiendo hasta de la dimensionabilidad del vector, n .

Una comparación visual de imágenes monocromáticas reconstruidas, coseno-transformadas y cuantificadas vectorialmente no sugieren que técnica es superior a una tasa de 1.5bits/pixel, aunque se puede obtener un efecto de mosaico. A una tasa de 1.0bits/pixel, un contorneo y reproducción pobre de los bordes fueron los principales defectos del esquema de codificación por cuantificación vectorial. Los resultados presentados demuestran que, para vectores formados a partir de bloques de 2x2 pixeles, existe una zona potencial de operación a 1.0 y 1.5 bits/pixel, donde la generación de "listados" adaptivos es muy ventajoso.

A estas tasas binarias, la cuantificación vectorial adaptiva sobrepasa a los métodos no-adaptivos; más aún, las imágenes resultantes se pueden comparar con las obtenidas con la codificación por transformada coseno. En lo que respecta a la implementación, la generación de listados adaptivos requieren de menos cálculos que para la de las técnicas no-adaptivas y mantienen su potencial para una compresión en tiempo real. Los experimentos en imágenes a color demuestran que la cuantificación vectorial adaptiva da resultados comparables a la codificación por transformada coseno. Aquí los vectores utilizados corresponden a la información espacial y espectral. Las imágenes codificadas por la cuantificación vectorial muestran dos tipos de defectos: pobre reproducción de bordes y contorneo falso. Las técnicas son necesarias para el preprocesamiento de los vectores a fin de que los vectores correspondientes a estos errores son aislados y dan preferencia en la fase de generación de listados.

2.2.6 DESEMPEÑO DE LA TÉCNICA DE CODIFICACION POR TRUNCAMIENTO DE BLOQUE:

La codificación por truncamiento de bloque (CTB) se ha mostrado ya como una simple y efectiva técnica para la compresión de imágenes de video digitalizado. El cuantificador a 1bit inherente tiende a realzar los bordes al preservar las variaciones de intensidad dominantes a expensas de cambios secundarios los cuales son menos perceptibles al sistema de percepción humano. Imágenes (de 100x200 pixeles) en movimiento reconstruidas y codificadas a tasas binarias tan bajas como 0.9bits/pixel han mostrado pequeña distorsión y muy pequeñas pérdidas de resolución comparadas a las imágenes originales de 8bits/pixel. El movimiento reconstruido es continuo y claro, sin exhibir borrosidades o algún otro artefacto objetable presente en muchos sistemas de alta compresión de video. Las grandes traslaciones debido al movimiento de una cámara de video son reproducidos con precisión.

Considérese a tres imágenes de prueba (a), (b), y (c) ("Pareja", "Luna" y "Facsimil", respectivamente) y sus correspondientes imágenes reconstruidas (d), (e) y (f). La efectividad de la codificación aparece a continuación[5], presentada en términos de "Relación Pico Señal Ruido" (SNRp) para facilitar la comparación de las imágenes "Pareja" y "Facsimil" cuando se utiliza un esquema adaptivo (aproximado) de CTB y el mismo CTB.

Imagen	SNRp(dB)-Adaptivo	SNRp-Optimo
Pareja	32.92	31.69
Facsimil	28.23	27.64
Luna	32.74	No disponible

Quando se comparan estos resultados, se obtiene un mejor desempeño con una aproximación adaptiva, aún cuando se le compare con un óptimo escogitamiento de momentos. Por ejemplo, utilizando una cierta aproximación del método CTB, el SNRp para la imagen de la pareja mencionada antes, es de 29.90dB; así, la aproximación adaptiva permite un mejoramiento de 3.02dB en el SNRp.

Asumiendo que la imagen original se representa por pixeles tomando cada uno de un número de 2^b niveles de gris; así, la imagen original tiene b bits/pixel. También, asumiendo que los bloques son todos de $B \times B$ bloques de pixeles. Utilizando el anterior proceso adaptivo, para cada bloque de $B \times B$ pixeles, se transmitiría un "cero" o un "uno" para cada pixel en el bloque para denotar la salida del cuantificador para cada uno, y se transmitirían dos de 2^b niveles de gris para denotar los niveles de salida del cuantificador a 1bit; el primero de estos niveles de gris pudo, por ejemplo ser asignado a un "cero" y el segundo de esos niveles pudo ser asignado a un "uno", y en esta forma convenir el Soft-ware de transferencia a utilizar. Notese que el escogitamiento de los momentos no necesita ser transmitido. Así, se transmite:[5]

$$1 + \frac{2b}{B^2} \quad \text{bits/pixel} \quad (2.101)$$

y la tasa de compresión resultante es de:[5]

$$\frac{bB^2}{B^2 + 2b} : 1 \quad (2.102)$$

Con respecto a los ejemplos presentados anteriormente donde $b=8$ y $B=4$, se transmiten 2bits/pixel, y la tasa de compresión asociada con la transmisión es de 4:1. La implementación de un sistema CTB operando a 1.6bits/pixel requiere solo cuatro líneas de almacenamiento de video y produce un código de longitud fija de datos para cada bloque.

El transmisor necesita solo calcular una media y una desviación estándar de muestras, así utiliza la observación a una tabla para el caso de un cuantificador bi-dimensional, y así obtener el plano binario de muestras de la imagen muestra. Los sistemas CTB intertrama producen un código de longitud fija para cada arreglo de tres tramas. Esto puede ser útil en la sincronización del receptor y en la edición de video.

Muy pocos bits son bits de control los cuales requeririan de protección contra errores de canal. La protección contra errores de estos bits da menos de 0.01 bits/pixel de incremento en la tasa total de datos. La compresión del ancho de banda para video digitalizado utiliza las técnicas CTB ya estudiadas en 2.1.4.5.1 y su aplicación es muy práctica. La posibilidad de procesamiento en paralelo incrementa el atractivo para estos sistemas en lo que respecta a la codificación de imágenes secuenciales en tiempo real. Ya ha sido demostrado el desempeño robusto de los 0.9bits/pixel sobre canales ruidosos.



(a)



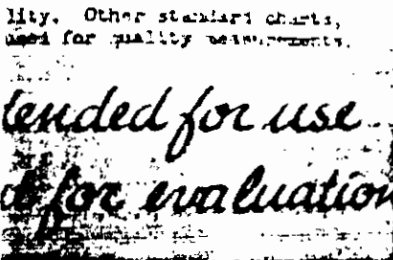
(d)



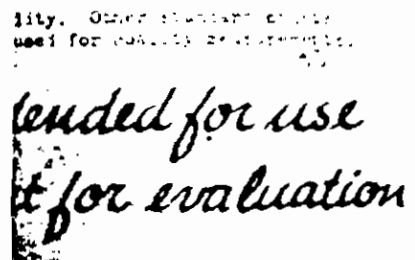
(b)



(e)



(c)



(f)

CAPITULO III

Aplicaciones

3.1 LA COMPRESION EN EL CAMPO DE LA GRABACION DE VIDEO

Los expertos en el campo del video siempre han buscado la máxima eficiencia, su mejor presentación y mayores aplicaciones, esto lo están logrando al manejar cada vez más y mejores sistemas de procesamiento. Estos sistemas se los puede encontrar tanto en los sistemas de video analógico como en los de video digital. En el primer tipo de sistema lo que se buscaba es presentar muestras "sintéticas" de imágenes reales a través de una pantalla o monitor, si lo que se quiere es almacenar estas imágenes para su posterior utilización (Video Grabadoras) lo que se busca es optimizar este proceso mediante adecuados esquemas de modulación analógica (Modular en frecuencia al audio, y al video), así se reduce el deterioro de las señales almacenadas por defectos propios del medio de grabación (Cintas magnéticas, Discos magnéticos, etc).

En el segundo tipo de sistema aún se está experimentando con diversas técnicas. Las que ya se utilizan, son resultado de muchos análisis y controles de eficacia. Se trabaja con las señales "sintéticas" del sistema analógico pero ahora surge un nuevo proceso, la digitalización de estas señales, proceso que hoy se aplica sobre todo por las aplicaciones que se puede lograr con una señal digital en especial el video digital; desde su manipulación para los estudios de grabación y post-producción, hasta la flexibilidad de manejo de equipos (Una señal digital cualquiera, puede ingresar al mundo de las telecomunicaciones más fácilmente que una analógica) de telecomunicación.

En lo que se refiere al almacenamiento y transmisión del video digital; la flexibilidad se ve opacada por la característica misma de estas señales, que al ser digitales se vuelven complicados algunos de sus tratamientos. Es por esto que la investigación del video digital se apuntó a la búsqueda de esquemas de procesamiento de estas señales a fin de compensar los inconvenientes que surgen en su tratamiento,

Las técnicas de compresión discutidas tienen una característica en común: Mantienen un compromiso entre la velocidad binaria y los niveles de calidad de la imagen. Estas técnicas se pueden combinar en un número de formas que dependen de las características funcionales de la *aplicación* requerida.

Algunos sistemas de compresión se diseñan a fin de producir una imagen que es subjetivamente "aceptable" para un observador en el punto final del canal de transmisión. En estos sistemas, la señal digital de video se comprimirá para concordar más precisamente con la tasa específica de transmisión. Las aplicaciones que no requieren de mucho procesamiento, se puede decir que usan esquemas simples de compresión. Ejemplos del *esquema de compresión sencillo*, incluyen los sistemas de distribución de video, tales como video-teléfonos, y sistemas futuros de radiodifusión terrestre digital. El siguiente cuadro describe las evaluaciones subjetivas de la calidad de una imagen que comúnmente se alcanzan luego de una aplicación simple de compresión DCT inter-campo (por ejemplo JPEG, que se estudiará más adelante) para varios rangos de compresión.

Bits/Fixe	Calidad de la Imagen	Razón de compresión
6.0	Método JPEG Sin pérdidas	2:1
1.5	Visualmente idéntico al original	11:1
0.75	Buena calidad	22:1
0.25	Imagen "Util"	64:1
0.1	Imagen reconocible	160:1

Otras aplicaciones pueden necesitar otros requerimientos en las tecnologías de compresión de video. Como ejemplo, al tomar el efecto de reducir costos en un *sistema de grabación de componente 4:2:2* con cassette compacto y tiempo de grabación extendido. En esta aplicación, el esquema de compresión escogido, no solo debe concordar con las características del sistema de grabación, sino que también debe garantizar un alto nivel de calidad de imagen e integridad de

la señal necesarios para editar y procesarla a través de múltiples generaciones. En los *servicios de telecomunicaciones*, la disponibilidad limitada por la capacidad de canal (56Kbps, 1.5Mbps, 45Mbps, etc.) requerirá altas tasas de compresión. En algunos casos, las señales de entrada se preprocesarán por filtrado y por sub-muestreo.

3.1.1 SUB MUESTREO:

En el Capítulo I se cubrió todo un estudio sobre la digitalización de las señales analógicas, y uno de sus puntos se ubicó en el muestreo de dichas señales. Esta sección se enfoca en las formas de como se puede implementar un muestreo si la frecuencia de este se disminuye (Muestreo Sub-Nyquist o *Sub-muestreo*) o si se incrementa (*Super-muestreo*) respecto de una referencia. En general el muestreo es una operación no invertible y si no se tiene cuidado en escoger la frecuencia del muestreador, se podría reconstruir una señal no deseada. Sea $x[n]$ una secuencia digital (discreta), su transformada de Fourier $X(e^{j\omega})$ se define como:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n} \quad (3.1)$$

$x[n]$ es la señal analógica (continua) $x_c(t)$ muestreada cada período T , esto es:

$$x[n] = x_c(nT) \quad (3.2)$$

así mismo la transformada de Fourier $x_c(t)$ puede definirse:

$$X_c(j\Omega) = \int_{-\infty}^{\infty} x_c(t) e^{-j\Omega t} dt \quad (3.3)$$

Se puede llegar a demostrar que:

$$X(e^{j\omega}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left[j\left(\frac{\omega - 2\pi k}{T}\right)\right] \quad (3.4)$$

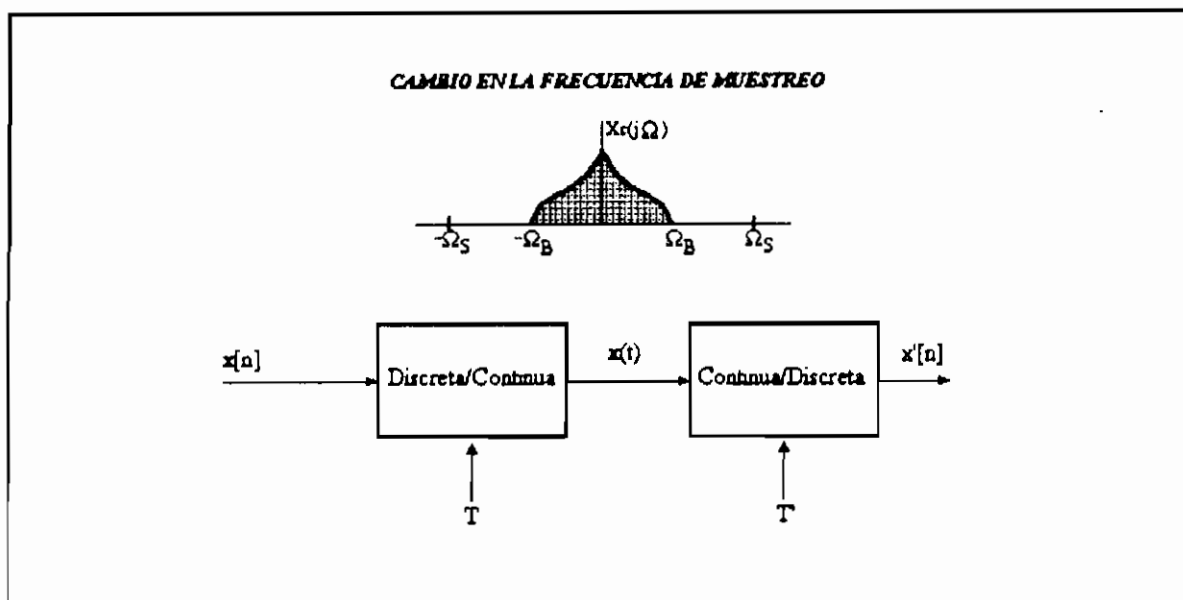


Figura 3.1 Cambio en la frecuencia de muestreo.

Sea Ω_m la frecuencia máxima (Véase la Figura 3.1) que podría tener la señal analógica, y sea que T es el periodo de muestreo que se utiliza para digitalizar a aquella señal, con $T = 1/\Omega_s$. Si lo que se quiere es SUB-MUESTREAR a la señal analógica, el nuevo periodo de muestreo T' debe ser un factor entero M del periodo original, esto es:

$$T' = M \cdot T \quad (3.5)$$

Considerando (3.2), la nueva secuencia discreta $x_d[n]$ es:

$$x_d[n] = x_c(nT') = x_c(nMT) = x[nM] \quad (3.6)$$

y la relación en frecuencia es entonces:

$$X_d(e^{j\omega}) = \frac{1}{M} \sum_{l=0}^{M-1} X(e^{j\frac{\omega-2\pi l}{M}}) \quad (3.7)$$

Es decir el espectro de la nueva señal digital se compone de una superposición de M espectros de ancho de banda **reducido** en un factor M . Si se cambia la frecuencia de muestreo de su valor original por un factor M , debe tenerse cuidado de no causar "Aliasing" en el espectro obtenido. Se evita este problema al tener siempre en cuenta el criterio de Nyquist:

$$\frac{1}{T'} = \Omega_{S'} > 2M\Omega_N \quad (3.8)$$

esto es, el mantener siempre Ω_N por debajo de la frecuencia de muestreo Ω_B :

$$\Omega_N < \frac{\pi}{T'} = \frac{\pi}{MT} = \frac{\pi\Omega_S}{M} \quad (3.9)$$

la frecuencia analógica Ω_N se relaciona con la frecuencia digital ω_N por:

$$\omega_N = \Omega_N \cdot T \quad (3.10)$$

con lo que la nueva señal muestreada se deberá filtrar con un filtro digital de frecuencia de corte:

$$\omega_c = \frac{\pi}{M} \quad (3.11)$$

Para el caso de elevar a la frecuencia de muestreo que no es de mayor interés en este trabajo el procedimiento es similar solo que en lugar de realizar un producto de T por un factor M lo que se realiza es una división por un factor entero N . Al proceso de bajar la frecuencia de muestreo se le conoce como *Down-Sampling*, *Sub-Nyquist*, o *Decimación*; y al de subirla, *Up-Sampling*, o *Interpolación*.

3.1.2 UN ESQUEMA PRACTICO DE VIDEO-GRABADOR DIGITAL:

Un estándar de HDTV (Televisión de Alta Definición) propuesto por la NHK (Corporación de Radiodifusión de Japón) con 1125 líneas barridas por trama a 30Hz (60 campos/segundo) requeriría de un amplio ancho de banda (20MHz) para la señal de luminancia y de 5.5MHz y 7MHz para las señales de crominancia, una grabación directa de estas señales en un disco de video con un usual formato FM resulta en una reproducción de corta duración. Una técnica de compresión del ancho de banda permitiría difundir el total de las señales de HDTV vía solamente un canal de satélite. Utilizando esta técnica, tales señales pueden comprimirse hasta 8.1MHz conservando un formato analógico dentro de un disco óptico.

larga duración. Una Video Casetera Digital (DVCR) graba señales de video digitalmente, y es superior a los VCR's domésticos en general en lo que respecta a la relación Señal/Ruido (SNR), resolución y manipulación. Sin embargo el VCR digital o DVCR tiene sus deméritos, el mayor de los cuales es una gran cantidad de consumo de cinta. En particular este demérito sería fatal para los VCR digitales domésticos. Por lo tanto, es muy importante establecer técnicas que hagan posible grabar señales de video digital tanto como sea posible sin un consumo excesivo de cinta. Desde este punto de vista, los investigadores se esfuerzan por desarrollar *técnicas* para reducir las *tasas de velocidad binaria* (Bits/Seg o bps) y tener *altas densidades de grabación*.

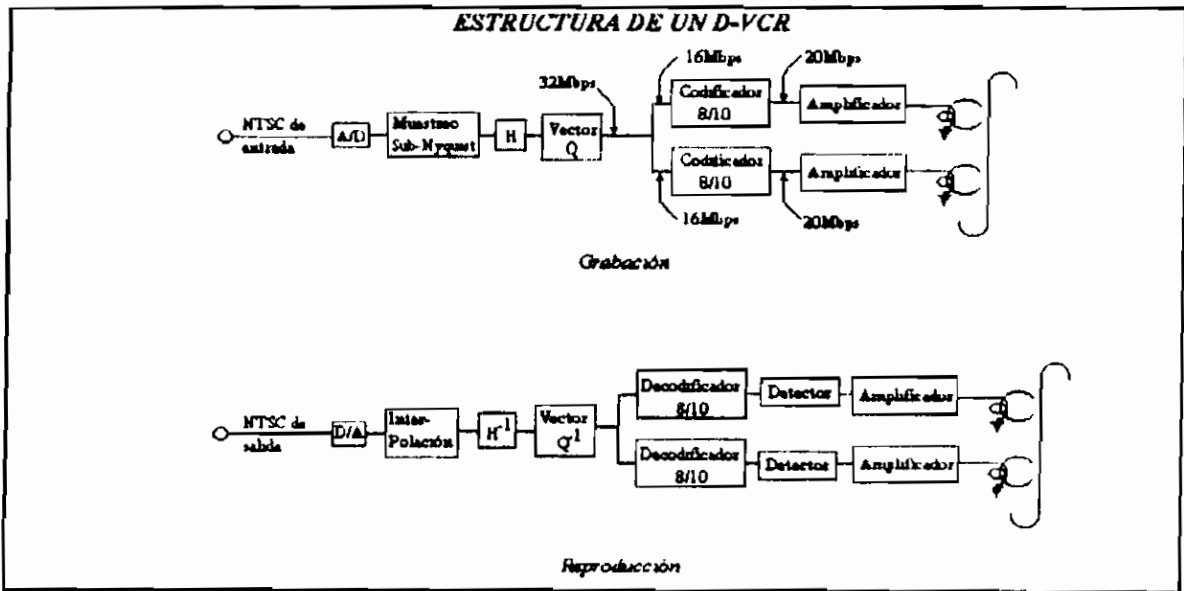


Figura 3.2 Estructura de un D-VCR.

Las técnicas de reducción de tasas binarias efectivamente eliminan la redundancia en las señales de video. Las técnicas de alta densidad de grabación guardan la información en una área reducida de cinta tanto como sea posible y reproducen la información lo más correctamente posible. Al usar estas dos clases de técnicas, se puede alcanzar grabaciones de larga duración de señales de video digital con un consumo de cinta tolerable. Por otra parte, el BER (Tasa de Bits Erróneos) en la grabación de alta densidad, necesaria para DVCR domésticos, es menor que la necesaria en un canal

de comunicación para la transmisión de imágenes. Más aún, se deben añadir datos redundantes para corregir errores aleatorios tan bien como los errores ráfaga en los DVCR's. Adicionalmente, no existe ninguna técnica de reducción de tasas binarias que no causen alguna propagación de errores. La estructura esquemática de un DVCR se muestra en la Figura 3.2[22].

La señal de entrada al DVCR es del formato NTSC. La resolución de un conversor A/D puede ser de 8bits, y la frecuencia de muestreo de 4 veces la mayor frecuencia de la sub-portadora de color. Se obtiene así una tasa de 115Mbps. Las técnicas de reducción de tasa binaria reducen la tasa original a 32Mbps. Los datos reducidos se distribuyen en 2 canales. Así la tasa para cada canal es de 16Mbps. Se añade paridad a los datos para controlar los errores. Todas las señales de paridad se insertan cada pulso de blanqueo de NTSC. Un codec mejorado se utiliza como codificador de canal. Se añaden dos bits de control a cada muestra para robustecer al sistema, de esta manera la tasa binaria se aumenta a 20Mbps. Estos datos se graban en una cinta de ME (Metal Evaporado) de 1/2" por medio de cabezas metálicas. La relación de velocidad cinta-cabeza es de 5.8 metros/seg. Los procesos se invierten en el modo de reproducción.

3.1.2.1 Reducción de entropía:

Dos clases de técnicas de reducción de tasas binarias se utilizan en un DVCR. La primera es un muestreo sub-Nyquist inter-campo (Dentro de la Sección 2.1.2.6 se trata la técnica intercampo y el muestreo sub-Nyquist o sub-muestreo que se trató en la Sección 3.1.1); y la segunda técnica es una cuantificación vectorial (Sección 2.1.4.4) combinada con la transformada de Walsh-Hadamard (Sección 2.1.3.c). El muestreo sub-Nyquist reduce la frecuencia de muestreo. La frecuencia original de muestreo de $4f_{sc}$ se reduce a $2f_{sc}$ en la primera etapa. Por lo tanto, la tasa después del muestreo sub-Nyquist es de 17Mbps.

La transformada bi-dimensional de Walsh-Hadamard de 8° orden se utiliza en un DVCR. El bloque Hadamard consta de 8 muestras luego del muestreo sub-Nyquist. La cuantificación de cada componente de frecuencia obtenido por la transformación de Hadamard es la mayor causa del demérito de la calidad del video. Se soluciona este problema por medio de la cuantificación vectorial de los componentes de frecuencia. La cuantificación vectorial se la utiliza con el fin de disminuir la distorsión y así mejorar la calidad del video. Para un DVCR, se le asigna 36bits al bloque de Hadamard compuesto de 8 muestras: el promedio de bits para expresar a una muestra es de 4.5. Como resultado, el ritmo de transmisión es de 32Mbps.

3.1.2.2 Alta densidad de grabación:

La velocidad relativa cinta-a-cabeza del DVCR anteriormente indicado es de 5.8m/seg, el ritmo de grabación está alrededor de los 20 Mbps, una densidad lineal de empaquetamiento es de 71Kbits/pulgada, la tasa de bits erróneos (incluyendo los errores ráfaga) está cercana a 10^{-4} .

3.1.3 COMPRESION DE VIDEO COMERCIAL:

En general, todos los sistemas de reducción de entropía tienen un nivel umbral (límite) de eficacia. A tasas de compresión más altas que las de este umbral, los *artefactos de codificación* (defectos como ruido y perturbaciones introducidas en la codificación) se incrementarán más allá de los que el sistema puede manejar. Estos efectos se pueden evitar por medio del filtrado y el sub-muestreo de la señal de entrada. La nueva combinación de densidades de muestreo de los componentes de la señal de entrada reducirán la tasa binaria de video digital original previa a la compresión; así se disminuirá la rata de compresión requerida.

Ejemplos de tales sistemas de muestreo, son los sistemas digitalizados de TV por componentes 4:2:0, 4:1:1, 3:1:1, o aún 3:1:0. En una señal por componentes 4:1:0, se

información de crominancia se filtra y sub-muestrea verticalmente. Esto significa que cada uno de los canales de croma (R-Y B-Y) contendrá solo 243 líneas, en comparación con las 486 líneas de una señal por componente de estudio en un sistemas de 525 líneas. Un sistema 4:2:0, por lo tanto puede describirse como un sistema de 126Mbps, lo que representa una reducción de un 25% en la tasa binaria (bits/muestra) previa a la compresión.

Aunque esta reducción de la tasa de entrada puede mejorar la ejecución de la compresión, el uso de filtraje vertical de la señal de croma puede degradar las características del procesamiento de los canales de croma en su manipulación y en las operaciones de efectos especiales. Esto hace de muchas imágenes filtradas en croma, indeseables para aplicaciones de producción. Un sistema 4:1:1 puede también alcanzar una reducción de un 25% en la tasa de entrada, esto por medio de una mayor reducción del ancho de banda de las señales de croma cerca de 1.5MHz(en términos análogos), o 180 pixeles por línea activa, comparada con los 2.75MHz, o 360 pixeles por canal de croma, en el sistema 4:2:2 original.

En la búsqueda de altas capacidades de almacenamiento sin una severa degradación en la calidad de la imagen, otros sistemas recurren a otro filtraje y sub-muestreo del canal de crominancia, además de la señal de luminancia. En los sistemas 3:1:1 por ejemplo, el filtraje y sub-muestreo da un resultado de 540 pixeles por línea activa y 180 pixeles para cada una de las señales de croma. Estos tres componentes juntos alcanzan un ritmo de aproximadamente 105Mbps, y un 38% de reducción en la tasa de entropía comparada con la componente 4:2:2. Estos sistemas digitales sub-muestreados están siendo considerados para las aplicaciones orientadas a los consumidores. Estos hacen uso de las técnicas inter-trama inter-campo para obtener tasas de alta compresión sin introducir objetables artefactos de codificación, pero con imágenes de calidad estos producen cierta pérdida en la resolu-

ción. Aunque se han hecho muchos intentos para encontrar una medida cuantitativa de calidad de imágenes para los sistemas de video comprimido, la comunidad de investigadores todavía no encuentra una forma de correlacionar los artefactos de codificación de imágenes de la misma manera en que el sistema visual humano lo percibe. La capacidad limitada por canal permite solamente bajas velocidades de transmisión. Si se requieren aceptables niveles de calidad de imagen descomprimida, el impedimento de la baja velocidad no lo dejará al usar las técnicas de codificación inter campo/cuadro solamente, que reduce las redundancias espaciales.

Los sistemas de muy alta compresión en estas aplicaciones deben además reducir las redundancias temporales utilizando la codificación inter-campo/trama con compensación de movimiento. Algunas aplicaciones de sistemas de alta compresión mantienen un buen nivel de calidad de imagen a través de un número limitado de imágenes generadas. Un ejemplo de este sistema es el sistema de compresión para la *transmisión de señales digitales de componentes 4:2:2* a 34, y 45 Mbps de estudio a estudio y de ciudad a ciudad (un estándar conocido como la Recomendación 723 del CCIR). La señal decodificada, sin embargo, no es lo suficientemente robusta para editarse en multigeneración extendida o manipulación digital requerida para la creación de efectos especiales. Un factor importante en el distintivo entre una técnica de compresión y otra para transmisión de señales de televisión y para la producción de video, es el efecto de *diferentes algoritmos de compresión en cascada*. Los sistemas de compresión de alta calidad diseñados para producción y post-producción han demostrado afortunadamente un desempeño completamente *transparente* en el manejo de material previamente codificado, en canales de transmisión de 34, y 45 Mbps. Los actuales sistemas comerciales para aplicaciones de edición de acceso-aleatorio, utilizan sistemas de compresión inter-campo/trama basados en técnicas TDC o de *oleaje*. En estas aplicaciones, el objetivo del sistema es el de producir el más alto grado

de compresión posible, al variar niveles de calidad de imagen, llegando a tener una reducción significativa de costos en equipo de almacenamiento basados en disco o en memorias de acceso aleatorio (RAM). Estos sistemas típicamente muestran factores de compresión en el rango de 8:1 a 30:1, dependiendo de la capacidad de almacenamiento del sistema y la máxima duración de la secuencia del video. La necesidad de búsqueda y edición de tramas y campos, los sistemas precisos que fuerzan el acceso aleatorio como la codificación JPEG, usan solo técnicas de compresión inter trama/campo.

La calidad de la imagen de estos sistemas muestran algún grado de artefactos de codificación, tales como "cuadrage" en áreas llanas de la imagen u oscilación de los detalles espaciales. Por ejemplo una señal digital 4:2:2 produce aproximadamente 168Mbps en el área activa de la imagen-lo cual traducido a la terminología de los sistemas no lineales, significan como 700Kbytes por trama de video. Si la señal de video se la comprime, a digamos 20Kbytes por trama de video para el almacenamiento, entonces se requiere un factor de 35:1. Un sistema de codificación inter trama/campo con este alto factor de compresión necesitaría como 8Gbytes de almacenamiento para una hora de material de video comprimido-con una calidad de imagen suficiente para bajas aplicaciones de edición.

El material de video que ha sido comprimido usando técnicas de codificación intra imagen, mantiene algo de su robustez original solamente a bajos rangos de compresión-de 4:1 a 8:1 para señales de una fuente 4:2:2. Esto lo hace más trabajable en cuestiones de edición de video producción: sin embargo se requieren bajos rangos de compresión. Por otro lado se incrementan los costos de almacenamiento: Un sistema de compresión de 10:1, por ejemplo, necesita de casi 8Gbytes de almacenamiento para soportar una hora de material procesado.

3.1.3.1 Perfil de una aplicación profesional:

Los algoritmos de compresión de alta calidad para una grabación de componente digital 4:2:2 ya se ha diseñado para soportar todas las características operacionales de los modernos VCR's y para soportar muchos niveles altos de calidad de imágenes en el trabajo de producción y post-producción de televisión. Las técnicas de codificación inter-campo TDC (Transformada Discreta Coseno) con tasas de compresión de 2:1 han probado su valor en VCR's del sistema Profesional Betacam Digital, y son capaces de soportar ediciones de campos precisos, modos de lanzamiento de recuadros, reproducción en velocidad variable, técnicas efectivas de encubrimiento de errores, y múltiples generaciones sin pérdidas discernibles en la calidad de las imágenes.

3.2 LA COMPRESION EN EL CAMPO DE LA RADIODIFUSION TELEVISIVA

Las 525 líneas por trama, 60 campos por segundo y relación de barrido de líneas en el sistema de televisión NTSC, han sido utilizados en los Estados Unidos por más de 50 años. La efectividad de este sistema de televisión ha sido mejorada significativamente cada año. Las mejoras más significantes y más simples fueron la adición de color. Los ingenieros fueron muy hábiles en añadir la información de color a la señal en blanco y negro sin incrementar el ancho de banda de transmisión. Para llegar a esto, la información de luminancia fue disminuida y fue introducida una sub-portadora, conteniendo la información del color. El resultado para los receptores en blanco y negro fue que bajó su resolución y que apareció una estructura de puntos, que se consideró aceptable. [30,31]

Otras mejoras han tomado muchas formas y surgen de la expansión constante de la tecnología. Los dispositivos captadores y los dispositivos de monitoreo han mejorado dramáticamente. Los circuitos de estado sólido ahora permiten funciones complejas que no fueron posibles cuando el sistema

fue diseñado. La tecnología actual permitirá otro mejoramiento significativo, la televisión de alta definición (HDTV). Por lo pronto un sistema de televisión de HDTV ya a sido diseñado con 1125 líneas por trama, 60 campos por segundo, y una relación de barrido de 2:1. Los estándares se fortifican en el CCIR (Comité Consultivo Internacional de Radio) y se concentra en un estándar de estudio de TV como una primera prioridad. El desarrollo de la tecnología en lo que respecta a transmisión por fibra óptica y circuitos VLSI (Sistema de Muy Alta Integración) hacen posible la red integrada de servicios digitales de banda ancha (B-ISDN)[21].

El gran ancho de banda disponible en las redes basadas en fibra óptica podrá soportar una multiplicidad de datos a altas velocidades y servicios de comunicación visual incluyendo los servicios de radiodifusión de televisión de calidad.

3.2.1 TRANSMISION DE VIDEO EN MOVIMIENTO:

Los trabajos en la codificación digital de señales de televisión se han desarrollado desde hace muchos años, inicialmente para las aplicaciones de transmisión y más recientemente para el procesamiento de señales. Existe ahora un renovado interés en la transmisión de señales de TV debido a la emergencia de B-ISDN y a la proliferación de técnicas de procesamiento digital de video en el estudio de televisión.

En Marzo de 1982 el CCIR unánimemente adoptó la *Recomendación 601*(Anexo 3): el cual estableció un acuerdo sobre un código digital que fuese compatible con los estándares de 625 y 525 líneas. Esta recomendación es conocida como codificación digital de las señales componentes basadas en el muestreo de la señal de luminancia a 13.5MHz y el muestreo de las dos señales diferencia de color:[20]

$$Cr=0.713(R-Y) \quad \text{y} \quad Cb=0.564(B-Y)$$

a 6.75MHz. Cada muestra es codificada linealmente en una palabra de 8 bits. El estándar de estudio es conocido como

código digital 4:2:2. La nomenclatura "4:2:2" fue originalmente seleccionada como una referencia al patrón de muestreo y a las características de resolución del nivel profesional del estudio. Para cada una de las señales diferencia de color se acordó primero que estén representadas por la mitad del número de las muestras correspondientes a la señal de luminancia.

3.2.2 JERARQUIAS DE LOS SERVICIOS B-ISDN:

Tradicionalmente, un interfáz ISDN contenía a un número de canales posicionados dentro de un interfáz de base, a fin de soportar a una variedad de aplicaciones sobre otro sencillo interfáz para usuarios de red. Un canal posicionado consiste de intervalos de tiempo específicos, asignados periódicamente, desde donde se lleva la información (flujo digital) hasta un servicio específico según la duración de la llamada. En B-ISDN, esta aproximación a los canales de servicio de comunicación se la conoce como Modo de Transferencia Sincrónica(MTS). Una segunda aproximación es el Modo de Transferencia Asincrónica(MTA), En consideración a la variación sustancial en las tasas de transferencia requeridas para diferentes calidades de video, una familia de TRES canales de banda ancha(MTS) se propuso para un mayor estudio por el grupo de trabajo en ISDN de Banda Ancha del grupo de estudio 18 del CCITT:[21]

- (a) Alto ritmo de transmisión(Nivel de acceso H₁).
- (b) Mediano ritmo de transmisión(Nivel de acceso H₃).
- (c) Bajo ritmo de transmisión(Nivel de acceso H₂).

Se mencionan las *Actividades de Contribución* y las *Actividades de Distribución* las que se refieren al trabajo sobre el material de video realizado en el estudio de grabación solamente o cuando se lo transmite (de estudio a estudio o dirigido a otros usuarios), respectivamente.

3.2.2.1 Alto ritmo de transmisión(H_4 ; 120 - 140 Mbps):

El nivel más alto en la jerarquía propuesta, H_4 , corresponde a una capacidad tal que es necesaria para las aplicaciones tanto de contribución como las de distribución de video de alta calidad. Para el caso particular de la transmisión de señales de HDTV, el ritmo de codificación de la fuente está por los 1.0Gbits/seg.

El uso de las técnicas de compresión es por lo tanto necesaria para la contribución y para la distribución. Se sospechaba que la calidad alcanzada dentro de la capacidad del canal H_4 será necesaria solo para los servicios de distribución. Cuando se trate de sistemas relacionados con el modo 4:2:2, las técnicas de codificación utilizadas para este ritmo de transmisión podrían proveer una adecuada ejecución combinada con una simplicidad de implementación.

3.2.2.2 Mediano ritmo de transmisión(H_3 ; 60 - 70 Mbps):

Los sistemas en este nivel se los podría utilizar tanto para aplicaciones de contribución como para las de distribución. Los requerimientos para calidad de imagen y procesamiento de flujo de datos son los mismos que para las tasa de H_4 , pero con un incremento de la complejidad en la implementación física de los codecs.

3.2.2.3 Bajo ritmo de transmisión(H_2 ; 34 - 44 Mbps):

Este sistema es adecuado para las aplicaciones de distribución y para algunas aplicaciones de contribución. Se espera una degradación imperceptible (del 45% respecto a una destrucción total de la imagen) con respecto a la señal original de 4:2:2 luego de una transmisión del codec.

3.2.3 CODIFICACION DIGITAL A UN RITMO PRIMARIO - H_1 :

Durante la primera fase del B-ISDN, el uso de canales de alto ritmo de transmisión en una red de larga distancia resultaba ser muy caro. Para los servicios como las de video-telefonía es posible hacer uso de bajos ritmos de transmisión.

como los de H_1 ($H_{11}=1536\text{Mbits/seg}$)-Norte América y $H_2=1920\text{-Mbits/seg}$ -Europa). La transmisión de video a esas tasas es un servicio ISDN, pero la tecnología ha ido evolucionando rápidamente y los algoritmos utilizados son de interés significativo. Existe un número de productos comercialmente disponibles para teleconferencias a estos ritmos de transmisión; su relativamente alto costo es más una consecuencia del mercado (ISDN ya no es disponible) que de la tecnología.

La actual generación de estos codecs está basada en el MICD con compensación en el movimiento o en la Codificación inter/intra-trama por Transformada.

La clase de algoritmo investigado es la codificación híbrida: remueve la redundancia temporal en la señal a través de la predicción inter-trama en el dominio de los pixeles y remueve la redundancia espacial utilizando la codificación por transformada (bloque de transformación de 8×8). A fin de llegar a tener una calidad suficientemente alta, la compensación en el movimiento en la predicción se mantiene aún cuando ésta sea opcional. Para una gran extensión sin embargo, lo que utiliza es la predicción con compensación en el movimiento que permite una buena calidad a bajos ritmos binarios. Un filtro en el lazo de predicción inter-trama se utiliza para mejorar la predicción así como para reducir el efecto visual del ruido de cuantificación.

3.2.4 TRANSMISION DE VIDEO EN H_2 :

Para obtener una calidad satisfactoria a ritmos H_2 (34-44Mbps) es necesario utilizar predicción adaptiva tri-dimensional con compensación en el movimiento, adaptación del bloque mismo y codificación de longitud de palabra variable. En el sistema 3:1:1 la predicción seleccionada se la apunta al receptor por información lateral y se basa en un bloque uni-dimensional que consiste de 9 muestras de luminancia y 3 muestras de cada señal de diferencia de color. Las técnicas de codificación basadas en la Transformada Discreta

(TDC) se las ha reconocido como las teóricamente más eficientes, en términos de una reducción de ritmo binario, en relación a los métodos de codificación diferencial. Más aún esta eficiencia de codificación se ha verificado en aplicaciones de video conferencias y video telefonía. Una de las principales razones por las que hasta ahora no se implemente el uso de TDC para la televisión digital, ha sido la excesiva complejidad del proceso de transformación coseno para aplicaciones en tiempo real. Este obstáculo está siendo superado en vista del progreso alcanzado en el desarrollo de sencillos dispositivos VLSI de codificación por transformada. Esta técnica (TDC) no utiliza sub-muestreo así que se preservan las bandas completas de las señales componentes.

El sistema ha sido diseñado para aceptar señales de componente 4:2:2 o señales compuestas. Una TDC bi-dimensional se aplica a bloques de 8x8 pixeles y así se provee el almacenamiento de 8 líneas consecutivas de un campo (codificación intra-campo). Se traslada el intervalo de blanqueo (blanking) horizontal para utilizarlo de forma que el ritmo binario se reduce a 166Mbits/seg. El tamaño de la transformada es manipulable y así se obtienen imágenes de excelente calidad, con una complejidad aceptable de equipo (hard-ware) para el procesador TDC y para la memoria de líneas.

Uno de los dos procesadores TDC opera sobre las muestras de la señal de luminancia (Y) mientras que el otro se lo utiliza sobre las muestras de las señales diferencia de color, Cr y Cb. La precisión aritmética interna del procesador es de 16 bits más tres bits para desbordamiento, mientras que 8 bits se utilizan para los coeficientes TDC. Los pixeles transformados, o coeficientes de salida, se escalan por un parámetro variable el cual depende del límite del búffer y de la energía media de los coeficientes mismos. La codificación por longitud de palabra variable de los coeficientes no-nulos y la codificación longitudinal de los coeficientes de ceros consecutivos se aplican a los coeficientes escalados. El

escalamiento adaptivo de los datos transformados por TDC da un valor promedio de entropía (bits/pixel) de cerca de 1.6-bits/pixel para la luminancia y 1.2bits/pixel para las señales diferencia de color, Cr y Cb.

3.2.5 EL SISTEMA M.U.S.E.:

MUSE son las siglas de Codificación por Múltiple Muestreo Sub-Nyquist y es un esquema de compresión de ancho de banda propuesto por NHK (Corporación de Radiodifusión de Japón) para mono-canal, en la radiodifusión directa por satélite de señales de HDTV 1125/60/2:1 en la banda de los 12GHz.

Se desarrollaron dos tipos de sistemas MUSE[23]: El primer sistema se previó para radiodifusión satelital directa para los usuarios y reducía el ancho de banda total de la señal HDTV a 8.1MHz. Este esquema recibe el nombre de MUSE-E (Emisión). Para aplicaciones de producción de campos y/o inter-estudio, un MUSE de mayor ancho de banda (16.2MHz) se ha propuesto recientemente. Este sistema se llama MUSE-T (Transmisión). Los principios básicos de estos sistemas son los mismos y consiste de un patrón de muestreo sub-nyquist, con interpolación inter-trama y compensación en el movimiento.

Una representación vectorial del movimiento de la escena se calcula para cada campo en el codificador. Esta señal se multiplexa en el intervalo de blanqueo vertical y así es transmitida al receptor. En el decodificador y para zonas móviles de la escena, las direcciones lectoras de pixeles de los campos previos son desplazadas de acuerdo a la información proporcionada por el vector de movimiento a fin de que el dato sea procesado de una forma aparentemente "estática". Aunque el sistema MUSE tendía a la transmisión analógica el uso de un sencillo muestreo nyquist a 16.2MHz y 8bits MIC, resulta en una señal digital que se puede transportar hasta con ritmos de H_4 . Se ha reportado también que la aplicación de las técnicas MICD pueden reducir el ritmo de

transmisión de la señal MUSE a 90Mbits/seg. Es de mérito notar la existencia de otros sistemas analógicos de compresión para DBS (Sistemas de Radiodifusión Digital) y para transmisión vía cable de señales de HDTV[20]: El sistema Europeo HD-MAC y el sistema norteamericano HD-MAC-60 de la Corporación Philips. Estos sistemas producen señales cuyos anchos de banda permanecen en el rango de 9.5 a 12MHz y por lo tanto son adecuados para un codificación MIC o MICD para la transmisión digital.

3.2.6 PERFIL DE UNA APLICACION COMERCIAL:

A continuación se va a describir un CODEC (CODificador-DECodificador) inter-campo como un ejemplo para aplicación comercial. Este dispositivo codifica a la señal NTSC de TV a color compuesta a un ritmo de transmisión de 30Mbits/seg. Se escoge este ritmo de transmisión con el propósito de llevar por radiodifusión dos programas de TV simultáneamente en un solo transpondedor del satélite INTELSAT IV.

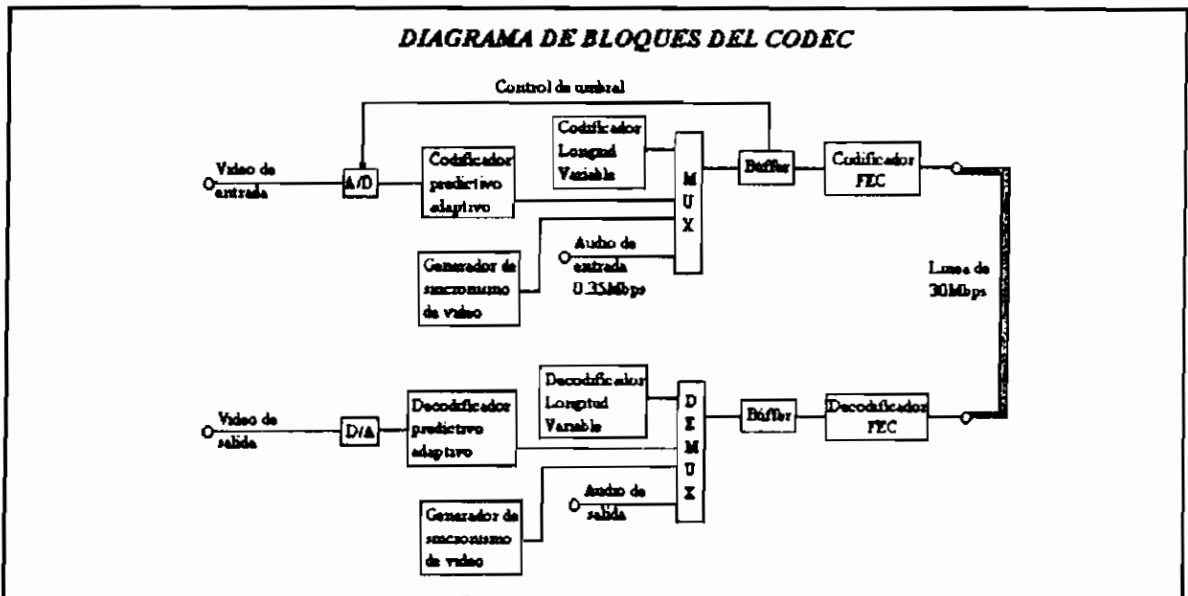


Figura 3.3 Diagrama de bloques de CODEC.

El transpondedor permite un flujo de 60Mbits/seg. Así, la tasa de transmisión asignada a un canal es de 30Mbits/seg, el cual incluiría a uno o dos canales de voz así como también bits extras para una corrección de errores tipo "hacia adelante" (FEC).

Así, la tasa neta de transmisión permitida para un canal de video debe ser menor a los 28Mbits/seg. Para tal ritmo de transmisión, es muy dificultoso conseguir una calidad suficiente de imagen para difundir programas de TV por los convencionales métodos de codificación intra-campo (v.g., MICD o codificación por transformada). En el presente codec se ha adoptado una codificación adaptiva utilizando dos tipos de predictores de señales compuestas de color; uno es un predictor inter-campo y el otro es un predictor intra-campo. En este esquema de codificación adaptivo, los errores de predicción de estos dos predictores son comparados a nivel de bloque, con cada 8 pixeles considerados como un bloque, y uno de los predictores, los cuales dan el error más pequeño, se escoge como el actual predictor de los 8 pixeles en el mismo bloque. Un diagrama de bloques del codec se muestra en la Figura 3.3. Y los parámetros principales del sistema se muestran a continuación[17]:

Señal de TV de entrada	NTSC, color, $1V_p/75\Omega$
Tasa de transmisión	30 Mbits/seg 27.65Mbps para video 0.35Mbps para audio 2.00Mbps para FEC
Frecuencia de muestreo Precisión Nivel de carga A/D	$3f_{sp}$ (=10.7MHz) 9 bits/muestra -3dB de rango dinámico
Algoritmo de codificación	Predicción adaptiva/ predictiva.
Capacidad del Búffer	512Kbits
FEC	BCH(239/255) código de corrección doble

Una precisión de digitalización de 8bits/pixel es usualmente popular para las señales de imagen. En el presente codec, sin embargo, se adopta una precisión de 9bits/pixel. Esto está basado en la siguiente consideración. El valor pico-pico nominal de una señal de TV se especifica como $1V$ el cual consiste de las señales de sincronismo de $0.3V_{pp}$ y la

señales de la imagen de $0.7V_{p-p}$ (Véase Anexo 4). Sin embargo, no es inusual que las señales de TV excedan el 1V, dependiendo del contenido de las imágenes transmitidas o de la condición del medio de transmisión. Al tomar estos hechos en consideración, el convertidor análogo a digital del presente codec ha sido diseñado para cubrir un rango de $1.4V_{p-p}$ ($-3dB$ de carga). En este caso, una precisión de 8bits/pixel podría ser insuficiente para las señales de las imágenes.

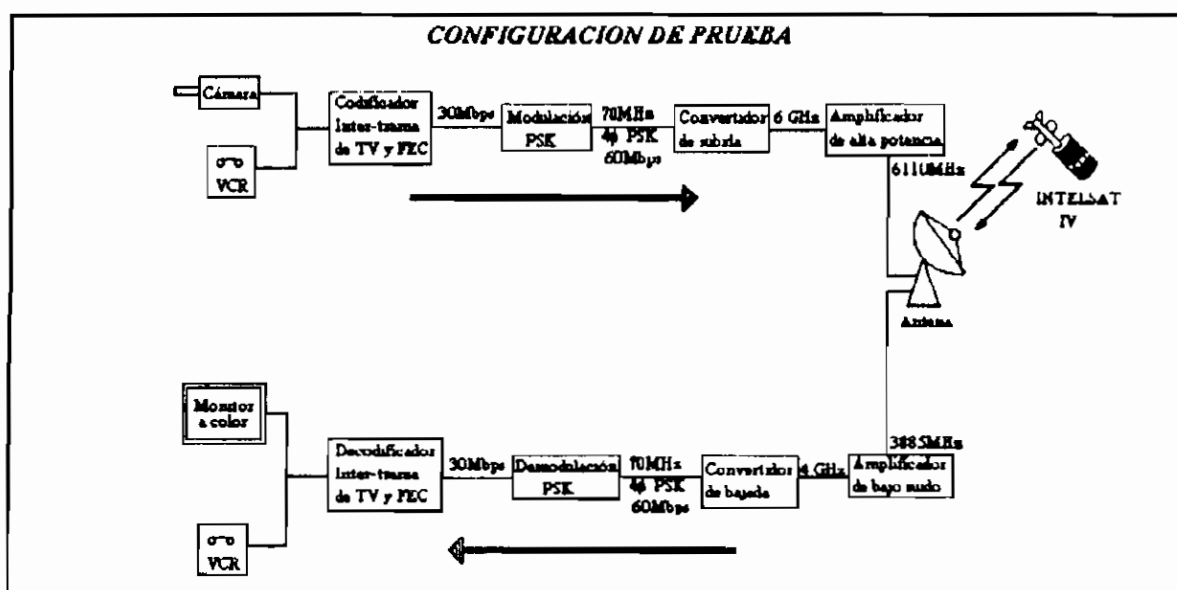


Figura 3.4 Configuración de prueba.

La precisión de 9-bit adoptada aquí, implica el asegurar una precisión de prácticamente 8-bit para el rango nominal pico-a-pico ($0.7V$) de las señales de TV. Hasta ahora, la calidad de la imagen ha sido discutida solo para una tasa de transmisión de 30Mbits/seg, donde la tasa neta para señales de TV es de 27.7Mbits/seg. La calidad de imagen al ritmo neto de cerca de 30Mbit/seg ha sido examinada al *remover* la función FEC de este codec. El desempeño logrado para un ritmo de transmisión bajo, de 10Mbits/seg se examinó al forzar en extremo el ritmo de transmisión en el presente codec con los otros parámetros conservándose intactos. De acuerdo con este experimento, la calidad de la imagen se deteriora gradualmente según se decremente el ritmo de transmisión, sin embargo los defectos del movimiento y/o las molestias de trama-trama no son perceptibles aún a 10Mbits/seg. En abril de 1979, una

prueba de campo de transmisión digital de la señal de televisión NTSC a color a través del Océano Pacífico por medio del satélite INTELSAT (IS-IV, F-8, transpondedor número 5) fue llevada a cabo en la estación terrena de Ibaraki, localizada en una área central de Japón, utilizando un codec de 30Mbits/seg. La configuración de prueba se muestra en la Figura 3.4[17].

Todo el equipamiento para esta prueba fue instalado en la estación terrena. La sección de RF es la estación comercial (Ibaraki #3), y sus características van con el estándar A del INTELSAT. El MODEM (MODulador-DEModulador) de 60Mbits/seg es un modem DPSK de cuatro fases y sigue la especificación descrita en el documento BG-1-18 del INTELSAT. Como se muestra en la Figura 3.4, el flujo digital de salida a 30Mbits/seg del codec es alimentado dentro del modem de 60Mbits/seg junto con otro flujo de señales de 30Mbits/seg (señales de televisión repetidas ; una secuencia pseudo randómica). Esta secuencia randómica fue empleada para medir una tasa de bits errados(BER) en este enlace digital de satélite. Los experimentos de transmisión fueron ejecutados para varios valores de relación portadora/ruido(C/N) en la estación terrena.

Las ventajas de la codificación adaptiva predictiva inter-campo intra-campo para la señal de TV compuesta NTSC en color puede inferirse de los resultados que se obtengan en la aplicación práctica y real de un codec como el presentado a 30Mbits/seg. Esta predicción adaptiva da una eficiencia de predicción buena y estable aún para imágenes con movimientos violentos. El método de predicción adaptivo inter-campo intra-campo es también un relativamente simple y efectivo método para alcanzar imágenes de mucha mejor calidad para la transmisión digital de programas de difusión televisiva a altos ritmos de transmisión, tales como, 32, 34, o 44-Mbits/seg, mejor que las técnicas de predicción convencional intra-campo. Un estudio cuidadoso de la calidad subjetiva

para varios ritmos de transmisión basados en esta predicción adaptiva está ahora bajo estudio. Con respecto a la implementación, este esquema de predicción intra/inter-campo, se tiene las siguientes ventajas sobre la predicción inter-trama solamente[17]:

- (1) La capacidad de memoria necesaria para la predicción se reduce a la mitad.
- (2) El proceso de control asociado es notablemente simplificado, debido a que el incremento del error de predicción para las escenas en movimiento es mucho menor para el predictor inter-campo que para el predictor inter-trama.

En el presente codec a 30Mbits/seg, la frecuencia de muestreo es exactamente tres veces la frecuencia de la sub-portadora de color, sin embargo los predictores pueden trabajar a una frecuencia de muestreo arbitraria mientras esta sea un múltiplo de la mitad de la frecuencia de la línea (60/50Hz para USA/Europa). Así, una frecuencia de muestreo relativamente baja se puede aplicar a este esquema de codificación, si se requiere un ritmo de transmisión muy bajo.

3.3 ESTANDARES DE COMPRESION

El video codificado digitalmente está en plena vigencia. Así como los discos compactos de audio han revolucionado a la industria disquera, así la nueva tecnología promete dar un gran bagaje de aplicaciones de video; entre éstas, el disco láser digital, la cámara electrónica, el videófono y sistemas de tele-conferencia, herramientas de imagen y video interactivo en computadores personales y estaciones de trabajo, desarrollo de programas en cable y satélite, y televisión de alta definición (HDTV). A diferencia de la tecnología de audio digital de los '80s, muchas aplicaciones del video digital rotan alrededor del uso de la compresión de datos. El ancho de banda del audio, después de todo está por los 20KHz, lo cual se traduce en un ritmo

digital de datos de cerca de 1.4Mbits/seg para un sonido estéreo de alta calidad. Las señales de video fuente muestreadas, por otro lado, requieren de mucho mayores ritmos, rangos de 10Mbits/seg para video de calidad difusiva hasta más de 100Mbits/seg para señales de HDTV. Mientras tanto, para facilitar el crecimiento de la industria, se han desarrollado tres estándares para imágenes estáticas (Fotografías), en movimiento (Películas, Programas de TV), y para video conferencia. Ya existen arreglos de chips para los tres propósitos algunos de ellos son diseños particulares y otros en armonía con los estándares propuestos.

Tres estándares de video digital se han propuesto:

- (1) El Grupo Conjunto de Expertos en Fotografía (JPEG) para compresión de imágenes estáticas.
- (2) La recomendación H.261 del CCITT para teleconferencias de video.
- (3) El Grupo de Expertos en Imágenes Móviles (MPEG) para compresión de total movimiento en medios de almacenamiento digital (MAD).

3.3.1 EL ESTANDAR DE JPEG:

Este es un algoritmo de codificación de imágenes estacionarias desarrollado por un equipo de investigación bajo el auspicio de la Organización Internacional de Estándares (ISO). El equipo convino en 1987 y el algoritmo es actualmente un *bosquejo* de recomendación del comité ISO (recomendación 10918).

El algoritmo base está en la categoría de la codificación de imágenes basados en transformada. Una imagen en color puede representarse en sistemas diferentes de color. Estos, incluyen al RGB (los tres colores primarios R=Rojo, G=Verde, B=Azul) en la industria de la computación; YUV (Y=Luminancia o Brillo, U y V=Señales de Diferencia de Color, Y-R y Y-B respectivamente) en la industria de la televisión; y CMYK (C=Cyan, M=Magenta, Y=Amarillo, K=Negro) en la indu. -

tria impresora. Dentro de cada sistema de color, las partes constituyentes se llaman componentes. Así, existen tres componentes de color en el sistema RGB, y cuatro en el sistema CMYK.

Cada componente de la imagen fuente en el codificador y decodificador JPEG se divide en bloques de 8x8 pixeles (Figura 3.5[3] y 3.6[3]). Cada bloque se transforma entonces utilizando la transformada discreta coseno bi-dimensional (TDC) con un kernel (semilla) de 8x8. Los 64 coeficientes resultantes, computados como un arreglo numérico bi-dimensional de 8x8, representan los contenidos de frecuencia del bloque dado.

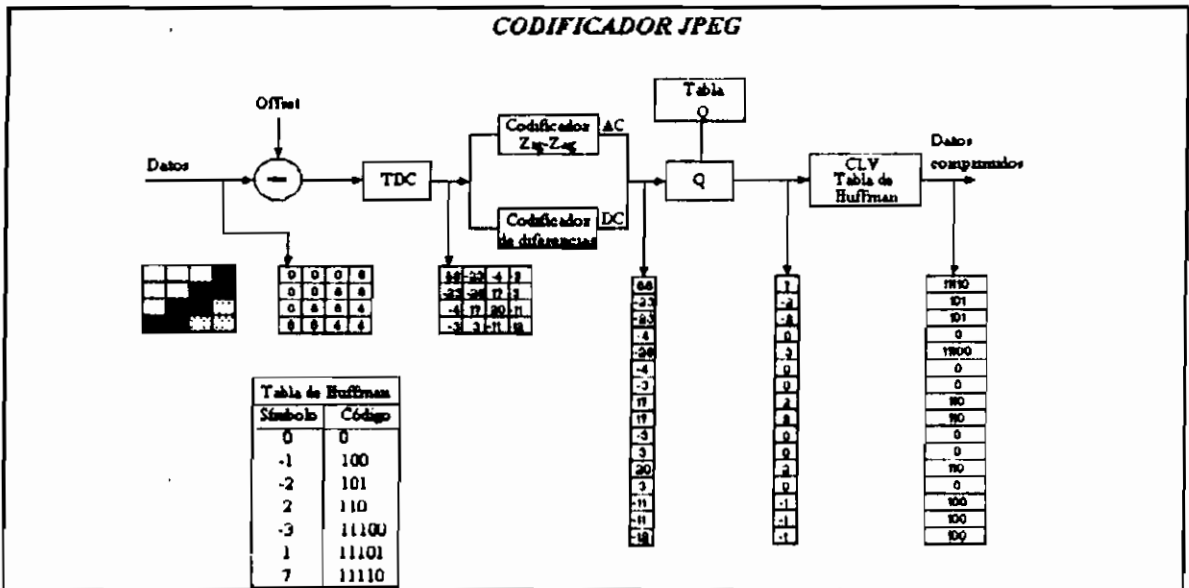


Figura 3.5 Codificador JPEG.

El coeficiente TDC en la parte superior izquierda del arreglo bi-dimensional es una medida de la energía de la frecuencia cero o el término DC (Direct Current). Por ejemplo, si la imagen original de 8x8 tiene un valor constante, entonces el término DC en el espacio transformado es no-nulo.

Las otras 63 entradas son los coeficientes de AC o "Alternos"; ellos dan la relativa fortaleza de los términos de la señal con un incremento horizontal de la frecuencia de izquierda a derecha y para los términos que incrementan la

frecuencia verticalmente de arriba hacia abajo. Lo siguiente es, la cuantificación de los coeficientes TDC. El tamaño de paso en la cuantificación varía con la frecuencia. La dependencia en la frecuencia se refleja en el hecho de que los coeficientes de alta frecuencia son subjetivamente menos importantes que los coeficientes de baja frecuencia y pueden por lo tanto cuantificarse con un paso muy grande (esto es, más toscamente). En adición, una componente individual puede tener su propia tabla de cuantificación. Siguiendo a la cuantificación, los coeficientes son reordenados en un arreglo unidimensional al examinar las entradas del arreglo bidimensional a lo largo de una ruta en Zig-Zag.

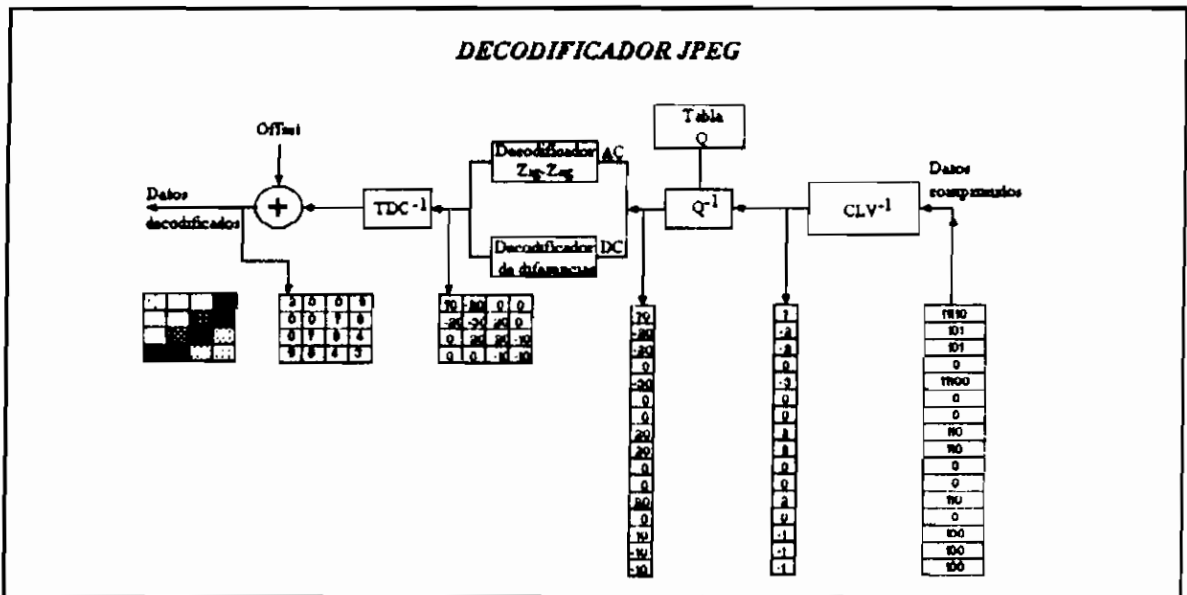


Figura 3.6 Decodificador JPEG.

De esta forma, los coeficientes cuantificados son "aproximadamente" acomodados en un orden ascendente de frecuencia. A continuación, los coeficientes AC y DC son codificados sin pérdida alguna utilizando para ambos una codificación al estilo de Huffman pero con diferentes parámetros. La codificación de Huffman es un medio muy bien conocido de reducción del número de bits necesarios para representar un arreglo de datos sin pérdida de ninguna información. Los coeficientes DC son codificados diferencialmente a fin de que los coeficientes DC de un bloque previo de 8x8 del mismo componente se utilice para predecir el coeficiente DC del

bloque actual de 8x8 y la diferencia entre estos dos términos DC se codifica. La tabla del código Huffman para el término DC se basa en las diferencias entre los valores. Los coeficientes AC codificados en zig-zag son primero codificados con codificación longitudinal. Este proceso reduce cada uno de los bloques de 8x8 de los coeficientes TDC a un número de eventos. Cada evento representa a un coeficiente no-nulo y el número de coeficientes nulos precedentes. Como los coeficientes de alta frecuencia son nulos más frecuentemente, la codificación de Huffman de estos eventos hace que se pueda alcanzar compresiones eficientes.

El algoritmo básico JPEG permite hasta dos tablas para la codificación de los coeficientes DC y AC. En la decodificación JPEG, el algoritmo de codificación se lo recorre en forma inversa, y es así que comúnmente se lo describe con un algoritmo simétrico.

3.3.2 EL ESTANDAR DEL CCITT:

Para la video-telefonía, la Recomendación H.261 del CCITT especifica un método de comunicación para la telefonía visual. Este a veces se lo conoce como el *Estándar p*64* debido a que la tasa de datos sobre el canal de comunicación es p veces 64Kbits/seg, donde p es un entero positivo menor o igual a 32. Para $p=1$, entonces, una señal de video de baja calidad puede transmitirse sobre una línea de 64Kbits/seg. Si $p=32$, una señal de video de alta calidad se puede transmitir para teleconferencia sobre una línea de 2Mbits/seg.

El estándar especifica la organización y la interpretación de los bits transmitidos a fin de que dos codificadores-decodificadores (codecs) de diferente manufactura puedan satisfactoriamente establecer una sesión de videoconferencia. Un codificador CCITT es más complicado que el codificador JPEG, aunque se puede distinguir bloques de funcionamiento familiar tales como el de la TDC y el Cuantificador.

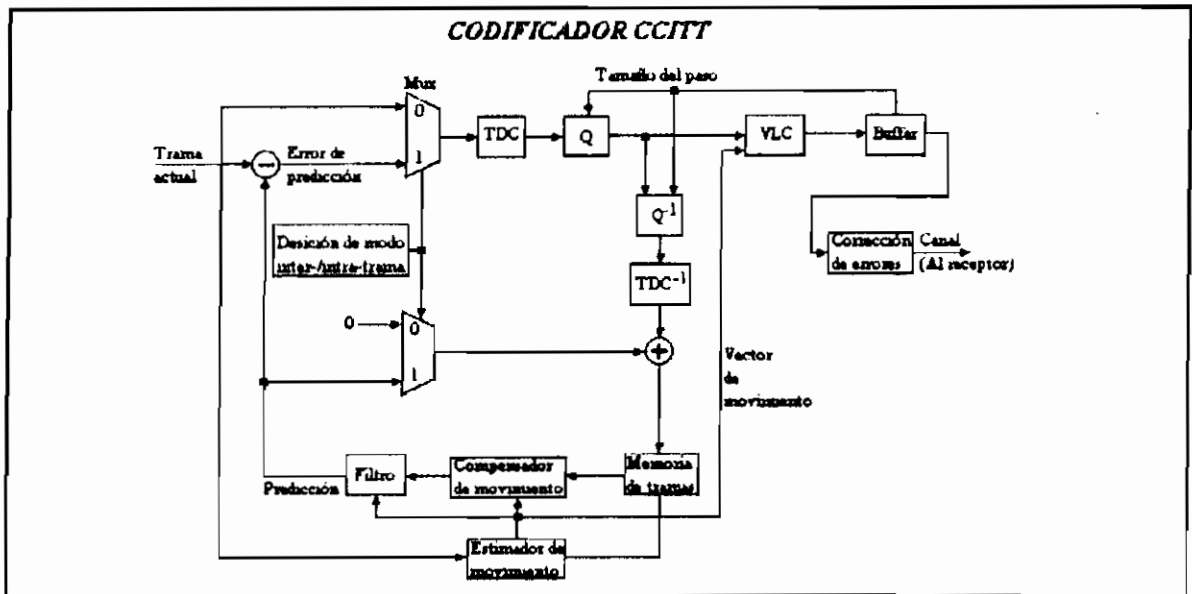


Figura 3.7 Codificador CCITT.

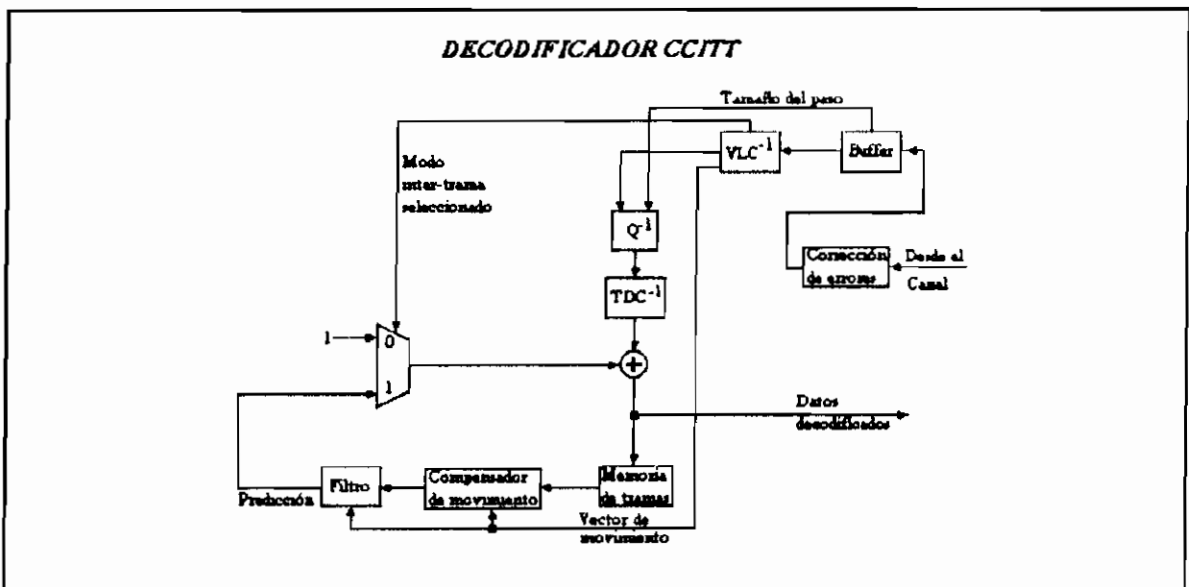


Figura 3.8 Decodificador CCITT.

El decodificador, sin embargo, es menos complejo. Una detallada descripción del algoritmo básico va más allá de este estudio. En lugar de este, se dará una visión de las técnicas utilizadas. El codificador CCITT es llamado híbrido porque combina una codificación por transformada (basada también en la TDC) con una codificación predictiva, en la cual un bloque en la trama actual se lo predice de un bloque en la trama previa utilizando un lazo de realimentación. (En contraste, el algoritmo JPEG básicamente opera en un lazo

abierto y se reinicia al término de cada imagen). Esto es, su predicción inter-trama resulta en una tasa muy alta de compresión. También en lugar de la codificación predictiva basada puramente en la diferencia entre la trama actual y la imagen reconstruida en la trama de memoria, el estándar H.261 del CCITT tiene una especificación opcional para la compensación en el movimiento. El codificador y decodificador CCITT se muestra a continuación en las Figuras 3.7[3] y 3.8[3] respectivamente.

3.3.3 EL ESTANDAR PROPUESTO PARA TOTAL MOVIMIENTO:

Así como H.261, el estándar MPEG propuesto es un algoritmo de compresión de imágenes en total movimiento con los dos modos intra-trama e inter-trama. Al contrario de H.261, el ritmo no puede exceder de 1.5Mbits/seg en los medios de almacenamiento digital actuales. Los presentes medios incluyen controladores de discos compactos, de cintas de audio digital (DAT), y discos duros magnéticos.

El comité de estandares en relación a MPEG no ha completado su trabajo y las especificaciones de audio y del sistema permanecen en discusión. Para un alto grado, se aplica la funcionabilidad del diagrama de bloques del codificador H.261. Sin embargo, las especificaciones de cuantificación y estimación/compensación del movimiento son diferentes. Una segunda fase de estandarización es el MPEG-2, su mira está orientada a una codificación eficiente hasta 10Mbits/seg con un objetivo posterior de resultados de mas alta calidad.

MPEG(Grupo de Expertos en Imágenes en Movimiento), el tercer estándar de video digital, puede aplicarse a medios de almacenamiento como ROM de Disco Compacto (CD-ROM), cintas de audio digital, discos Winchester (Discos Duros), y discos ópticos de escritura, así como en canales de comunicación como la ISDN y las redes de área local (LAN's). MPEG comprime las señales de video a cerca de 1.5Mbits/seg y para señales de audio digital a ritmos de 64, 128, y 192Kbits/seg por

canal. Esto también involucra a la sincronización y a la multiplexación de audio comprimido en formas múltiples y flujos binarios de video. El algoritmo MPEG se realiza en tres etapas:

- **PREPROCESAMIENTO:** Se escoge la resolución en función de la tasa de datos deseada, y se realiza un sub-muestreo de la señal de crominancia.
- **COMPRESION CON PERDIDAS:** Se remueve las redundancias temporales y espaciales, se utiliza una codificación híbrida TDC/MICD.
- **COMPRESION SIN PERDIDAS:** Codificación del flujo resultante de bits.

3.3.3.1 El estándar MPEG-1:

Utilizado en codificadores de video que podrían manejar ritmos de hasta 1.856Mbps/seg. (Canal de telecomunicación: $31 \times 64 \text{Kbps} = 1.984 \text{Mbps}$). Como ejemplo, el CD-ROM maneja un ritmo de 1.2Mbps.

El desarrollo de MPEG-1 enfocado a las técnicas de reducción de la velocidad binaria de imágenes móviles progresivamente barridas que requieren una calidad de imagen correspondiente, en forma aproximada al video VHS, a un ritmo de 1.5Mbps. Para lograr esto, los elementos algorítmicos del estándar MPEG-1 se optimizaron para la compresión de imágenes definidas por los así llamados parámetros "SIF" (Source Input Format de MPEG-1): Barrido progresivo a 30/25 tramas por segundo de imágenes con una resolución de 360 pixeles (352 procesados) por 240/288 líneas. Sin embargo las técnicas de compresión MPEG-1 han sido aplicadas a imágenes de mayor resolución y velocidades binarias con buenos resultados. El siguiente desarrollo es el llamado MPEG-2, centrado alrededor de específicas técnicas de compresión genéricas para uso en un amplio rango de aplicaciones, velocidades binarias, resoluciones, calidades, y servicios.

Los esquemas de compresión MPEG-2 son orientados a los medios de almacenamiento digitales, radiodifusión, transmisión HDTV, y la distribución de empaquetamientos de video a través de canales de telecomunicaciones, a más de otras aplicaciones -en rangos de 4Mbps a 80Mbps. Pero por el contrario un conjunto de herramientas de compresión puede ser utilizado en conjunto con reglas exactas de una sintaxis de compresión. Estas reglas exactas facilitan el intercambio de flujos de bit a más aplicaciones distintas.

3.3.3.2 El estándar MPEG-2:

Utilizado bajo ritmos de 4 a 5Mbps con calidad PAL según el CCIR601, 7 a 9Mbps con calidad de componente según el CCIR601 y de 20 a 40Mbps para HDTV. Este sistema de compresión incorpora todas las técnicas desarrolladas por los sistemas MPEG-1 a más de la adición importante de nuevas herramientas de compresión que optimizan el proceso de señales de video entrelazadas.

Por primera vez, MPEG-2 especifica técnicas que requieren las aplicaciones como los *múltiples arreglos de reducción de datos*. En MPEG-2, el flujo de bits comprimido está hecho de sujetos incluidos en la compresión de datos. Esta organización de datos dentro de sujetos, permite la reconstrucción de imágenes a fin de hallar varios estándares de calidad y resolución en las imágenes. Las aplicaciones que requieren el uso de estas características especiales del MPEG-2 se las llama *sistemas escalables*. En estos sistemas, la calidad de la imagen de cada subarreglo decodificado es directamente proporcional a la cantidad de bits de información que el subarreglo transporta. El mínimo subarreglo que puede ser decodificado se llama la capa base; los grandes subarreglos se los conoce como bases ampliadas. La estructura de las herramientas de codificación eficiente ya se discutieron, tales como predicción por compensación de movimiento inter trama/campo y la Transformada Discreta Coseno de los errores de predicción.

MPEG emplea una nueva técnica de compresión conocida como *Predicción Bidireccional*. MPEG se ayuda de la compresión de información temporal y espacial al separar la señal de video entrante en tres clases de tramas: la Trama I, Trama P, y Trama B. Todas las tres clases de tramas, codificadas juntas en una estructura secuencial, da lugar al *Grupo de Imágenes* (GOP). La Figura 3.9[39] muestra la estructura de codificación del GOP usada en los sistemas MPEG.

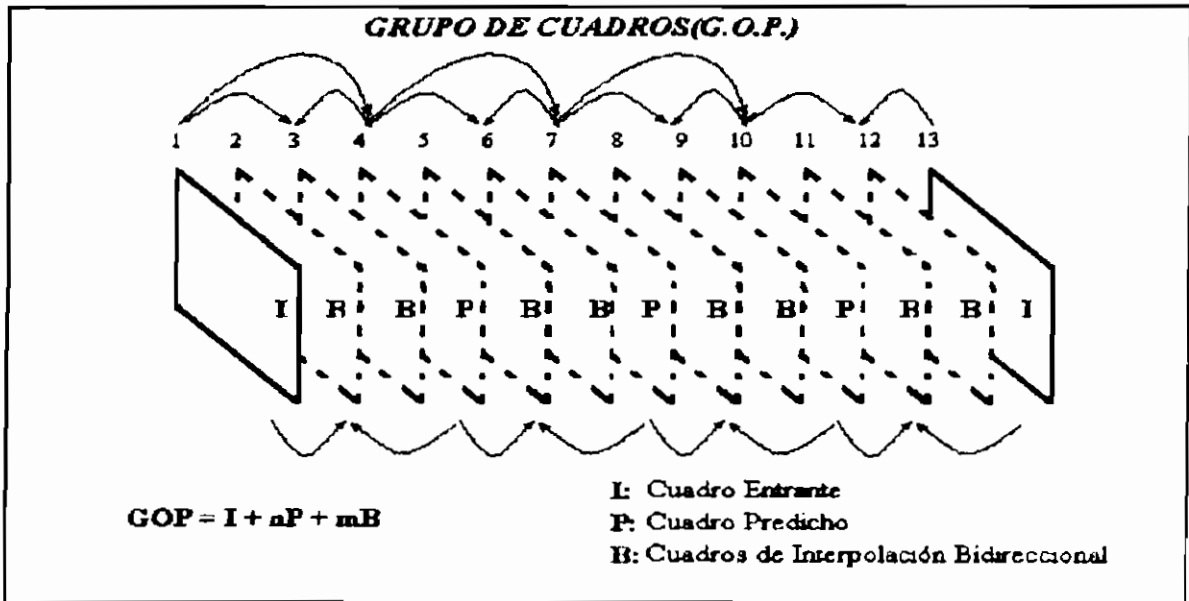


Figura 3.9 Estructura de codificación GOP.

IMAGENES-I: Son la clave de los elementos conformadores de la señal de video. Estas imágenes, las que pueden ser campos de video o tramas de video, se comprimen usando solo información intra-imagen. Cada imagen I entrante se divide en bloques de 8x8 píxeles para compresión usando técnica de la transformada discreta coseno (TDC). Las imágenes I, son la plataforma, o el punto de referencia, para la compresión de futuras imágenes en la secuencia de video.

IMAGENES-P: Son predichas de previas imágenes I o imágenes P usando técnicas de compensación de movimiento. Los valores de error de predicción son luego reducidos por: TDC, cuantificación y codificación de entropía.

En los sistemas de compresión MPEG las imágenes P predichas de las imágenes I, son llamadas de predicción delantera.

IMAGENES-B: Son también comprimidas a través de la predicción sin embargo puede ser derivado de tramas de referencia que ocurren tempranamente (predicción delantera) o más tarde (-predicción posterior) en la secuencia.

Las predicciones delanteras y posteriores pueden promediarse también como valores de predicción de imágenes B. Como en la codificación de imágenes P, el error de predicción (la diferencia entre la imagen predicha y la imagen actual de entrada) es comprimida aún más por la codificación TDC.

El MPEG divide la señal de video entrante en una sucesión de grupos de imágenes. Las imágenes I (tramas entrantes) sirven como plataforma o imágenes de referencia para la compresión de futuras imágenes. Las imágenes P (tramas predichas) son predichas de imágenes I o imágenes P usando técnicas de compensación de movimiento. Las imágenes B (tramas bidireccionales) son predichas de tempranas imágenes I e imágenes P (lo cual se llama "Predicción Delantera") o luego imágenes I o imágenes P (llamada "Predicción Trasera"). Por qué tres clases de imágenes? Así cada clase de imagen puede codificarse a diferentes velocidades binarias y diferentes niveles de calidad de imagen. Las imágenes I requieren de los mayores rangos de velocidad binaria, porque estas sirven como referencia para la codificación de otras imágenes y deben ser comprimidas de maneras que preserven la alta calidad de la imagen para evitar la propagación de artefactos de codificación. Para la video edición y las aplicaciones en los modos de disparo de imágenes, las imágenes I debe ser también capaces de la descompresión sin referencia a otras imágenes. Las imágenes P codificadas pueden alcanzarse con 2 o 3 veces menos datos que la codificación de imágenes I-sin embargo las imágenes P deben servir como referencia para la compresión de

mayores tramas, que requieren todavia de mayores calidades de imagen descomprimida.

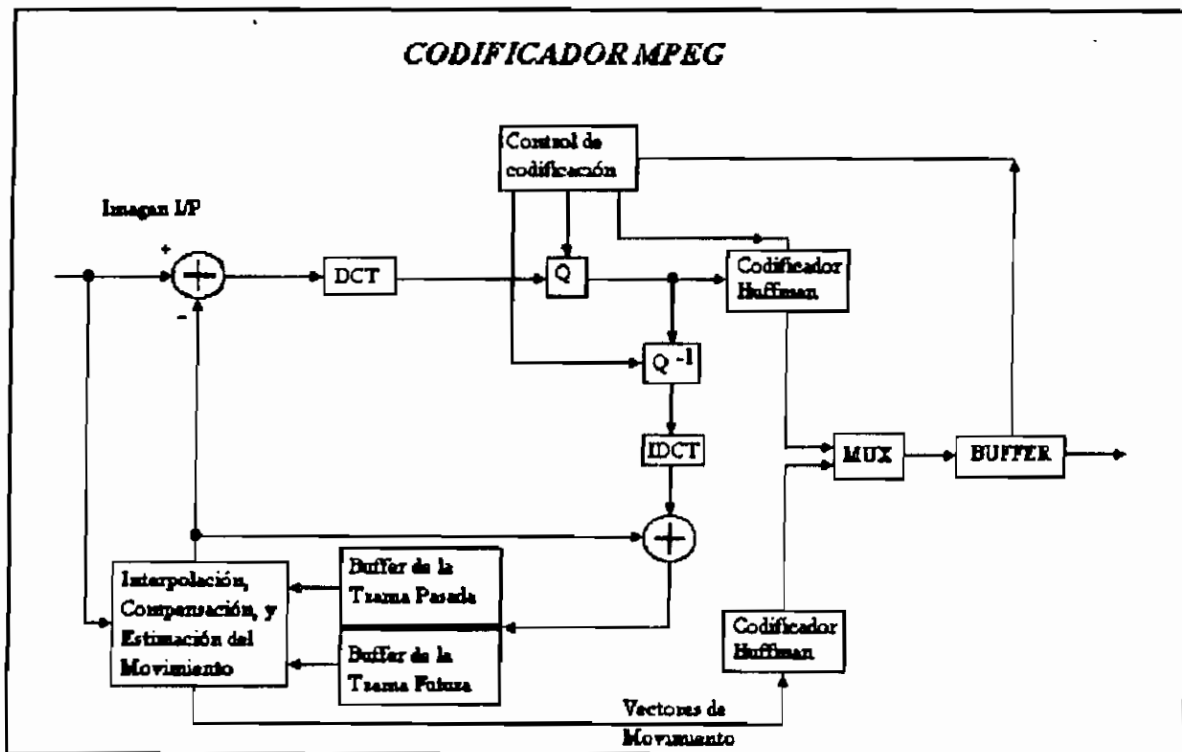


Figura 3.10 Codificador MPEG.

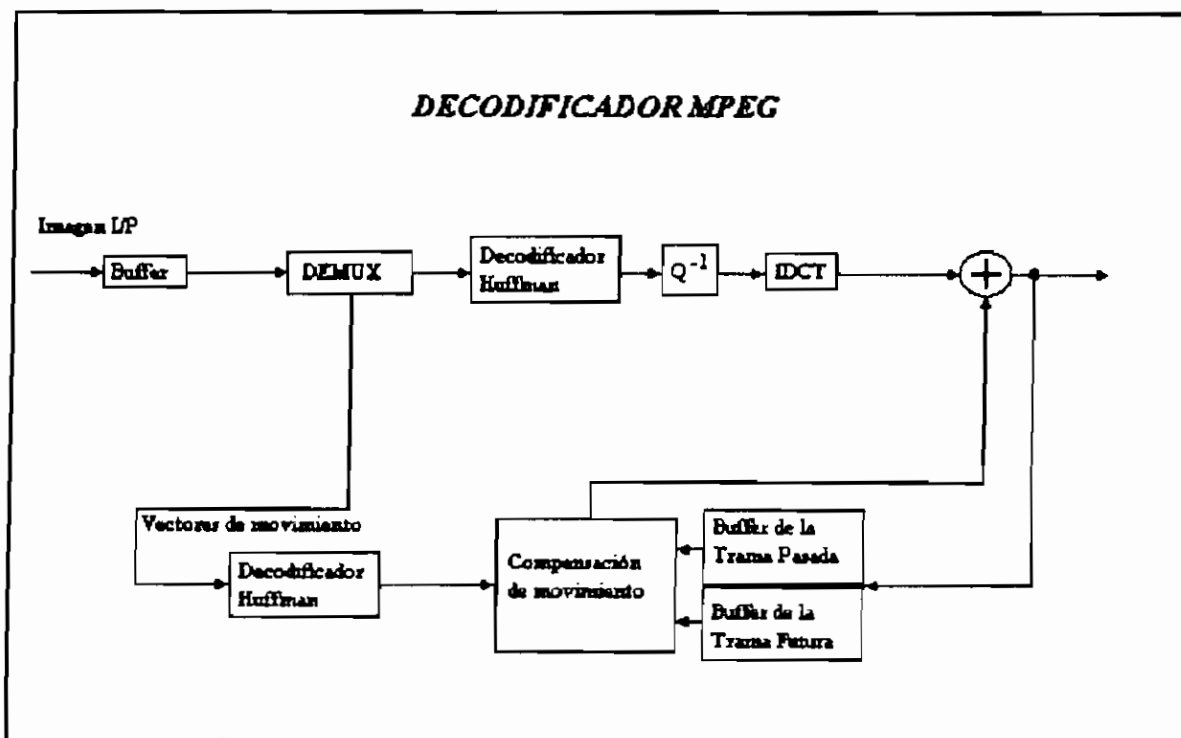


Figura 3.11 Decodificador MPEG.

Las imágenes B requieren de los menores rangos de velocidades de transmisión de todos; su codificación típicamente permite variar las cantidades de artefactos de codificación debido a sus grandes rangos de compresión. Los cuadros B no se usan para referencia de compresión para otras imágenes, así que no requieren gran cuidado. Sin embargo el uso de las imágenes B involucra mas equipo complejo y caro, debido a que muchas imágenes deben ser almacenados en memorias por tramas para permitir la necesaria predicción bidireccional. La Figura 3.10[39] y la Figura 3.11[39] respectivamente muestran los diagramas simplificados de un codificador y decodificador MPEG-2. Una de las principales ventajas del estándar MPEG-2 sobre el MPEG-1 es que el MPEG-2 es capaz de codificar señales de video entrelazadas. Es posible seleccionar entre la predicción de trama o campo, así como un predictor especial para aplicaciones de codificación/decodificación que no hacen uso de imágenes B. En el MPEG-2, la TDC puede también presentarse en bloques de pixeles tendientes a campos o a tramas como se muestra en la Figura 3.12[39].

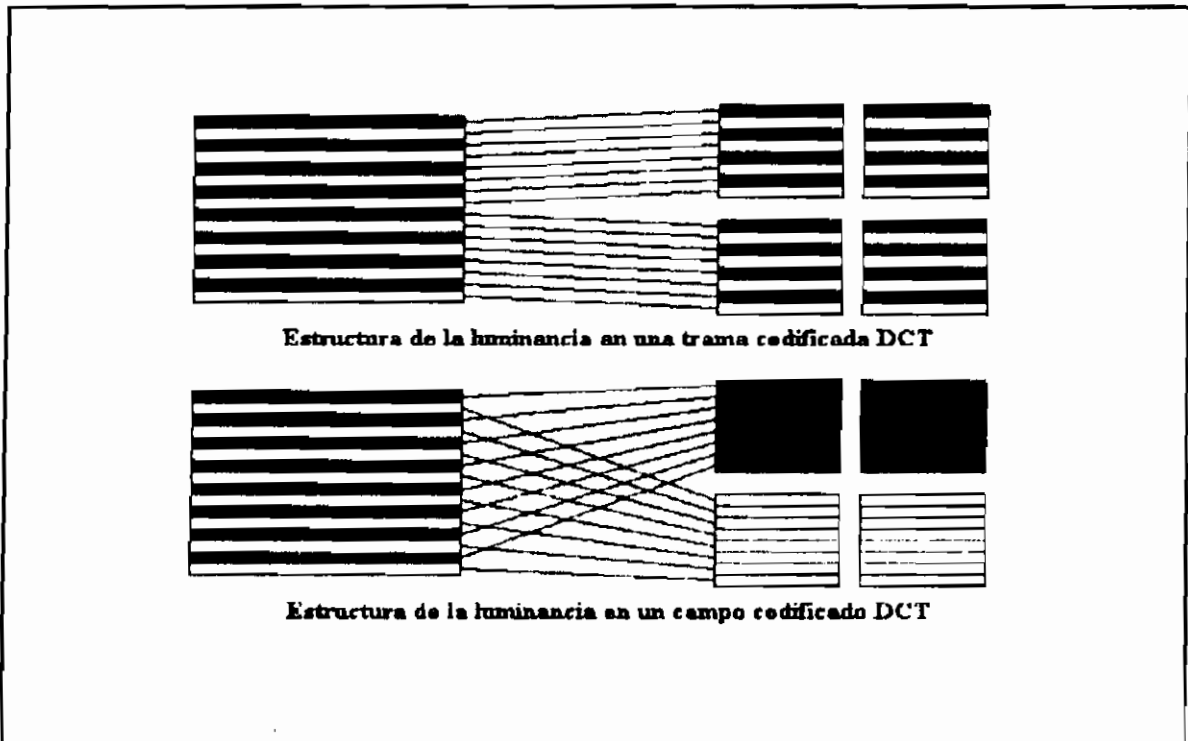


Figura 3.12 La TDC en acción sobre campos o tramas.

NOTA: Algunos sectores industriales usan la terminología MPEG 1++ o MPEG 1.5. Estas referencias usualmente tienen la indicación de MPEG 1 y se la usa para convertir la información referida a una imagen entrelazada de televisión.

El MPEG-2 permite la codificación TDC de campo o trama. Los bloques de pixeles basados en trama dan buenos resultados aún con imagenes estáticas; los bloques de pixeles basados en campo reducen la codificación de artefactos con imagenes móviles. A continuación se muestran sistemas de compresión con rangos de compresión, y de calidad de imagen.[39]

Sistema de compresión	Rango de compresión	Calidad de imagen
JPEG	2:1	Excelente
JPEG	11:1 - 8:1	NTSC
JPEG	22:1	< NTSC
JPEG	64:1	Pobre
JPEG	160:1	Muy pobre
Señal fuente = Componente de video 4:2:2 (720x486x30x16 = 168Mbps)		

Sistema de compresión	Rango de compresión	Calidad de imagen
MPEG-1* (1.5Mbps)	21:1	VHS
MPEG-2 (5.0Mbps)	25:1	>NTSC
MPEG-2 (10Mbps)	13:1	<D1
MPEG-2 (Imagen-I)	12:1	NTSC
Señal fuente = Componente de video 4:2:0 (720x486x30x12 = 126Mbps) *Señal fuente = 360x243x30x12 = 31.5Mbps (720x486x30x16 = 168Mbps)		

Sistema de compresión	Rango de compresión	Calidad de imagen
CCIR 723 ₁	5.3:1 - 4:1	Contribución
Betacam Digital ₂	2.37:1	Componentes de estudio
Oleajes	15:1	Buena
₁ Señal fuente = Componente de video 4:2:2 a 8 bits=168Mbps ₂ Señal fuente = Componente de video 4:2:2 a 10bits=210Mbps ₃ Señal fuente = Componente de video 4:2:0 a 8 bits=126Mbps		

A continuación se hallan las tendencias actuales sobre los esquemas MPEG[34], conocidas como las Fases de Estandarización de MPEG debido a que según estas el MPEG evolucionaría hasta hoy:

- **MPEG-1:** Codificación genérica de imágenes en movimiento con audio asociado a un ritmo de hasta 1.5-Mbits/seg. Contiene un formato SIF (Source Input Format) y actualmente se halla en sus etapas finales: Estándar ISO 11172. Sus video-codificadores podrían manejar tasas de hasta 1.856Mbps
- **MPEG-2(Altos ritmos binarios):** Ritmo de hasta 10-Mbits/seg según un formato del CCIR 601. El formato de HDTV se puede combinar con MPEG-2. Un esquema de comité se alcanzó en Noviembre de 1992. Para CCIR 601 en calidad PAL se obtiene de 4Mbps a 5Mbps, con calidad de componente CCIR 601, de 7Mbps a 9Mbps; y, para HDTV de 20Mbps a 40Mbps.
- **MPEG-4(Codificación de muy bajo ritmo):** Ritmo de 64Kbits/seg y menor. Codificación basada en modelos, y basada en objetos. Utiliza un algoritmo H.261 avanzado.

Capítulo
4

*Sistema de
Transmisión-Recepción
de Imágenes*

CAPITULO IV

Sistema de Transmisión-Recepción de Imágenes

4.1 ESTUDIO DEL SISTEMA

En el Capítulo I se mostró la secuencia a seguir en lo que respecta al procesamiento digital de imágenes: su captura y digitalización. A fin de transmitir este flujo digital de datos se explicaron los criterios científicos de una transmisión digital (transmisión de pulsos, IIS, sincronismo, ancho de banda, distorsión de amplitud y fase). Un sistema inicial de transmisión de imágenes, se muestra en la Figura 4.1 en la que se visualizan las etapas de una comunicación digital: El terminal transmisor, el canal de transmisión y el terminal receptor.

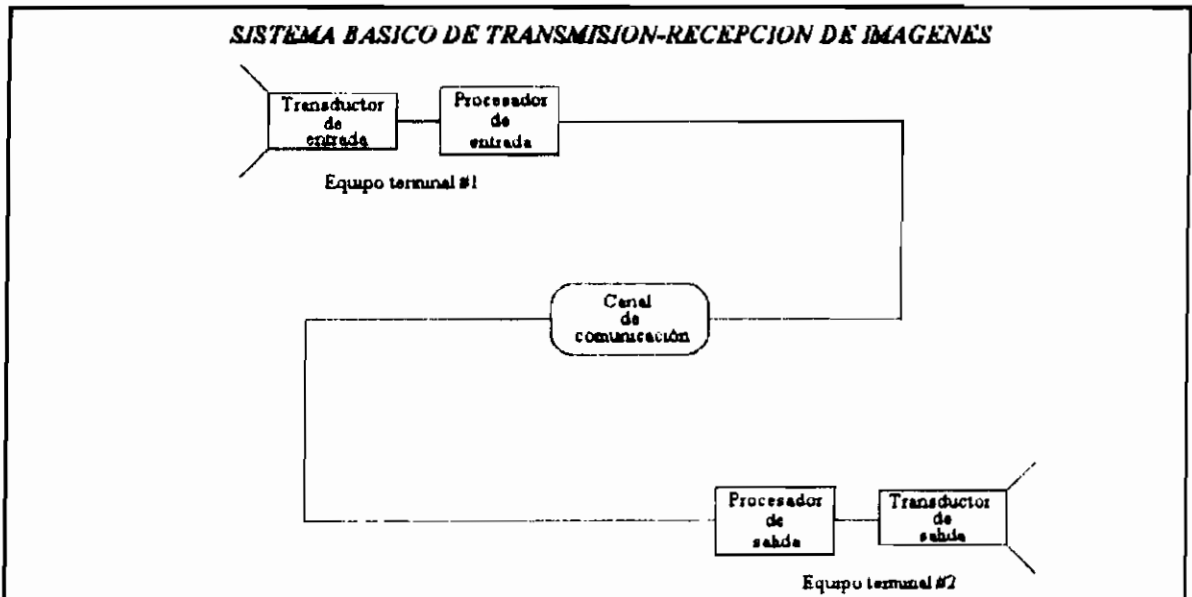


Figura 4.1 Sistema básico de Tx-Rx de imágenes.

El interés de este capítulo es presentar un esquema global de un sistema de comunicación de imágenes, el que consta de una etapa transmisora, una receptora, y entre las dos, el canal de transmisión. Se ha hablado de la comunicación digital como la más moderna forma de intercambio de información, existiendo tres maneras de enviar y reconocer mensajes, es decir de establecer una comunicación: [29]

- Simplex.
- Semi-Duplex.
- Duplex.

4.1.1 SISTEMA SIMPLEX:

Según este sistema, la comunicación entre dos terminales se la realiza en forma *UNIDIRECCIONAL*, es decir la información se traslada de una estación (terminal #1) a un receptor (terminal #2), y solamente, en ese sentido. El medio de transmisión está implícito, y un ejemplo de este sistema es la radiodifusión de señales de televisión.

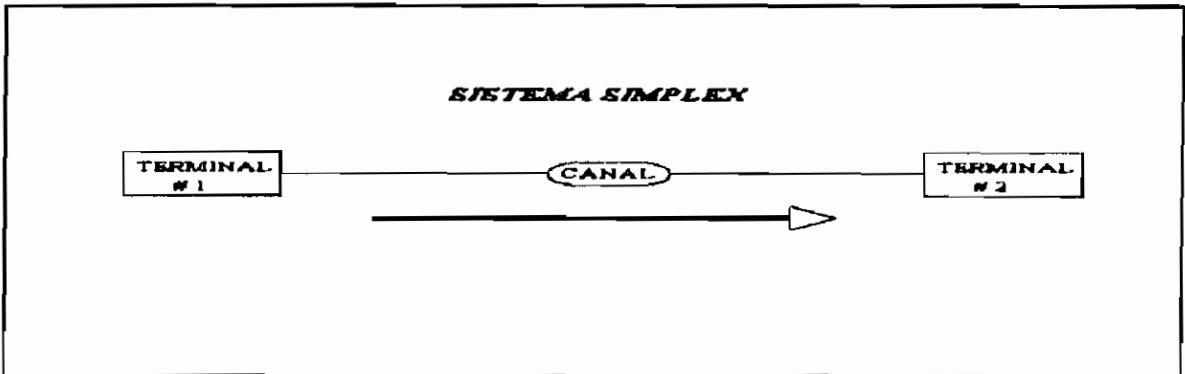


Figura 4.2 Sistema Simplex.

4.1.2 SISTEMA SEMI-DUPLEX:

Según este sistema, la comunicación entre dos terminales se la realiza en forma *BIDIRECCIONAL*, es decir, la información se traslada de una estación (terminal #1) a otra estación (terminal #2), y recíprocamente, solo que para que se establezca la comunicación un terminal transmite a la vez, pues los dos pueden recibir información así como también la pueden transmitir; así mismo el canal de transmisión queda implícito. Ejemplo de este sistema fue el utilizado en las misiones Apolo donde se esperaba la transmisión terrestre para luego transmitir las lunares y viceversa.

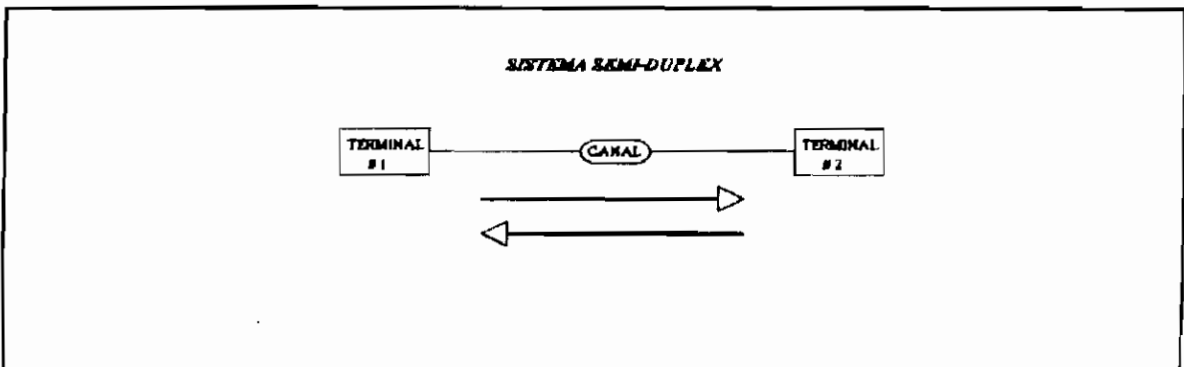


Figura 4.3 Sistema Semi-Duplex.

4.1.3 SISTEMA DUPLEX:

En este sistema, la comunicación entre dos terminales se la realiza en forma *BIDIRECCIONAL*, es decir, la información se traslada de una estación (terminal #1) a otra estación (terminal #2), y reciprocamente, esta vez la comunicación de un terminal se puede hacer aún si el segundo terminal está transmitiendo, sin que se produzca una mezcla inentendible de información (aunque esto a veces ocurre) sino que la comunicación es más dinámica si ambos terminales pueden establecerla como y cuando quieran. Así mismo el canal de transmisión queda implícito. Ejemplo de este sistema se lo encuentra en la telefonía en general pues lo que se busca en ella es lograr una comunicación lo más cercana a la comunicación personal.

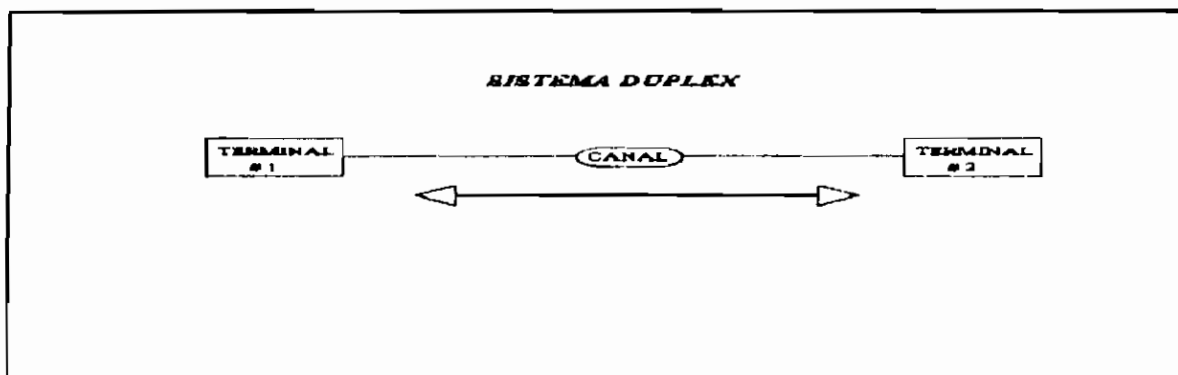


Figura 4.4 Sistema Duplex.

En la Figura 4.1 se muestra un canal de comunicación sin interferencias, sin embargo el mundo real no es así, la incidencia de ruido y perturbaciones dentro del canal de transmisión puede ser muy severa. Este efecto se lo puede visualizar mejor en la Figura 4.5[34]. Se debe notar que solo se indica la incidencia del ruido sobre el canal de comunicación, esto no significa que no exista ruido en las otras etapas, pues este ruido está tan implícito y es tan grave como el que se introduce en el canal o el que este (el canal) genera.

Los siguientes párrafos explicarán a las tres etapas que se muestran en la Figura 4.1, a pesar de que en esta no se muestra, lo que dentro de ellas existe se desarrollará.

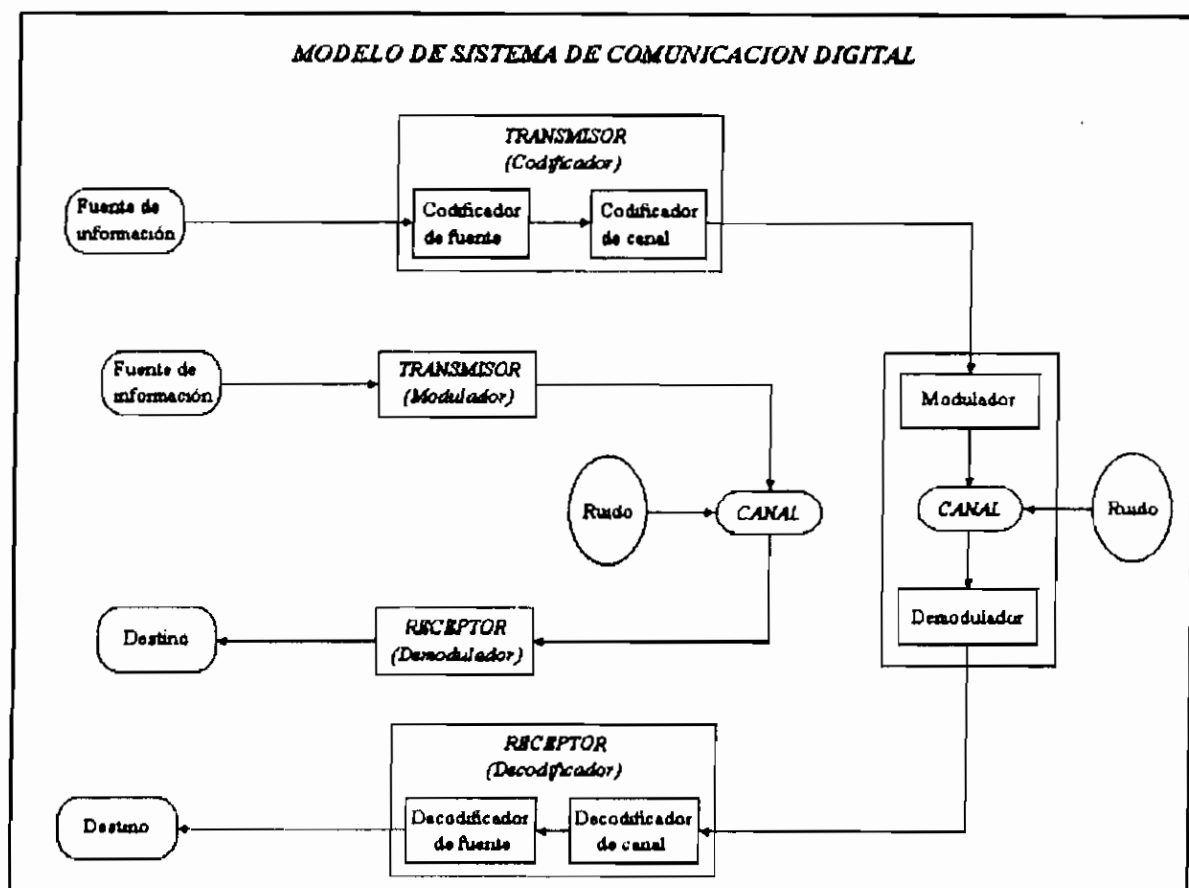


Figura 4.5 Modelo de Sistema de Comunicación Digital.

4.2 ETAPA DE TRANSMISION-COMPRESION

En principio se analizará una comunicación del tipo Simplex, pues resulta muy explicativo y general. Dentro del equipo terminal #1 de la Figura 4.1 se encuentra el transductor de entrada y el procesador de entrada.

4.2.1 TRANSDUCTOR DE ENTRADA:

Este dispositivo es el encargado de cambiar el dominio "palpable" de la imagen, al dominio eléctrico, con el que se trabaja desde este punto en adelante. El término palpable se refiere al efecto que produce una imagen al sistema visual humano, es decir la imagen misma, que puede estar en forma de texto, como una fotografía o como una secuencia de imágenes formando una película. En la Figura 1.6 en el Capítulo I se ilustra el sitio de actuación del transductor. El transductor puede ser un dispositivo sensible a la luz visible el cual produce una señal eléctrica proporcional

al nivel de energía censada o un rastreador digitalizador ("scanner") que convierte la señal de un sensor a la forma digital. Un esquema ilustrativo es el de la Figura 1.1 (Capítulo I).

4.2.2 PROCESADOR DE ENTRADA:

Luego de obtener la interpretación eléctrica de la imagen, se produce todo un procesamiento de esta señal el cual involucra algoritmos que incluyen funciones bidimensionales. La mayoría de las funciones del procesamiento digital de imágenes se las puede implementar por Soft-ware. Las únicas razones por las cuales se necesita Hard-ware especializado es la necesidad de aumentar la velocidad en algunas aplicaciones o para superar limitaciones computacionales.

El procesamiento convierte la imagen a un formato tal que se pueda adaptar al canal de transmisión. El muestreo de la señal se produce previa a la digitalización (todo este proceso se lo cubrió en el Capítulo I), luego de la cual los datos digitales (flujo de información) pasan a un proceso de codificación cuyas técnicas ya se desarrollaron en el capítulo II a fin de optimizar dicho flujo de datos y así optimizar el desempeño del equipo terminal #1. Esta optimización no estaría completa si no se incluyera a algunas de las diversas técnicas de protección contra errores y esquemas de modulación.

4.2.2.1 Protección contra errores:

El problema del control de errores es el de diseñar e implementar sistemas codificadores y decodificadores que trabajen en las siguientes condiciones:[41]

- La información debe ser transmitida o grabada en un ambiente ruidoso de la manera más rápida posible, conservando de esta forma la velocidad de transmisión de los mensajes.
- Se debe recuperar la información a la salida del canal decodificador de una manera confiable.

- Los costos del codificador y decodificador deben estar dentro de límites aceptables y deben ser fácilmente implementados.
- Debe disminuir la probabilidad de error efectiva del canal.

El control de errores es aplicable a todo tipo de transmisión tal como la línea telefónica, enlaces de radio, enlaces telemétricos, enlaces satelitales. Cuando se almacena información, muchas veces el tiempo de almacenamiento, puede ser causa de que el momento en que se quiera recuperar la información, esta haya sufrido variaciones mientras estuvo almacenada en memorias semiconductoras, magnéticas y ópticas, o discos flexibles.

Los tipos de errores que más comúnmente se pueden hallar se clasifican en dos tipos:[41]

(a) **Errores Aleatorios:** Son los que se producen al azar en una secuencia de dígitos.

(b) **Errores Ráfaga:** Son aquellos en los cuales la ocurrencia de un error afecta también a secuencias anteriores y posteriores de secuencias de dígitos.

La manera en la cual se pretenda eliminar el efecto que producen los errores en la transmisión mucho va a depender del tipo de sistema de comunicación con el cual se está trabajando. Por lo general en sistemas en los cuales la transmisión de datos se da en una sola dirección, es conveniente implementar sistemas de corrección y detección automática de errores debido a la imposibilidad de pedir retransmisión del mensaje. En sistemas en los cuales la transmisión de mensajes se da en los dos sentidos, la utilización de códigos detectores y correctores de errores dependerá de la rapidez con que se quiera obtener la información libre de error. de lo contrario es suficiente con pedir retransmisión del mensaje utilizando para ello un sencillo sistema de detección de errores.

4.2.2.2 ESQUEMAS DE MODULACION:

Luego de una adecuada codificación (compresión y protección contra errores), la optimización de la transmisión no puede estar completa sin una adecuada adaptación del flujo digital de datos al canal de comunicación, pues como se explicó en capítulos anteriores, no es posible transmitir con adecuada eficiencia datos digitales por los tradicionales medios de comunicación, es por eso que se adecúa dicho flujo de información de tal forma que se lo pueda enviar tanto por medios tradicionales como no tradicionales de comunicación.

Por ejemplo, es posible enviar información binaria (ceros y unos) por medio de cables metálicos sin mucha degradación de la información para longitudes cortas de conductor; pero si lo que se quiere es enviar esta información por cables tan largos como líneas telefónicas la información sufre una gran descomposición; por esto lo que se hace es modular esos pulsos binarios con una portadora de frecuencia telefónica (300 a 3400 Hz) y así la degradación de la información digital no es tan grave pues está protegida por la característica propia de la modulación.

Las señales binarias no pueden ser transmitidas directamente sobre largas distancias, ellas deben ser primeramente codificadas de tal forma que garantice los siguientes parámetros:[41]

- Un nivel constante a lo largo del camino de transmisión.
- Una apropiada forma del espectro de energía.
- El esquema de codificación debe ser transparente a todas las señales binarias de información, puesto que no se pone ninguna restricción a la información a ser transmitida.
- La señal codificada debe posibilitar la decodificación de manera única para reproducir la señal binaria original.

- Una componente DC no significativa.
- Un número importante de cruces por cero que permita la extracción de una señal de reloj.
- Mínima multiplicación de errores.
- Adecuada eficiencia de codificación.

Algunas estructuras de codificación ofrecen adicionalmente características deseables como detección de errores; una reducción del ancho de banda y otras. En cuestiones de transmisión digital los esquemas de modulación más utilizados son la modulación AM, FM, PM, y todas sus modalidades (QPSK, BFM, etc.) y las técnicas complejas QAM (Quadrature Amplitude Modulation). Por medio de la investigación se ha logrado verificar que los esquemas de modulación en fase resultan ser los más adecuados en lo que respecta al SNR y al BER (Figura 4.6)[41].

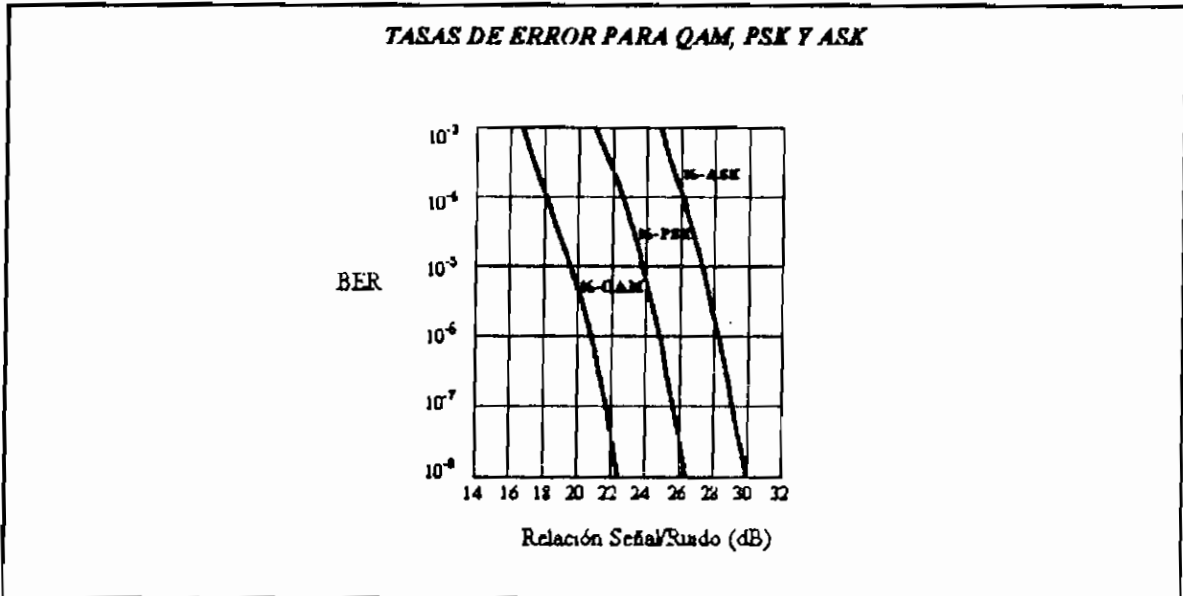


Figura 4.6 Tasas de error para QAM, FSK y PSK.

4.2.2.2.1 Esquemas de modulación de fase:

Adicionalmente al uso ineficiente de potencia de señal, las señales moduladas en amplitud tienen otra desventaja. Por definición, una señal modulada en amplitud utiliza múltiples niveles de señal. Así AM es muy vulnerable a la saturación de la señal debido a la estrecha distancia entre los niveles de amplitud.

En sistemas de radio generalmente la saturación ocurre a la salida del amplificador de potencia del transmisor. En algunos casos los amplificadores de salida operan a potencias menores que la máxima posible para evitar la saturación y otras no-linealidades así pueden acomodar señales de amplitud modulada. En los sistemas con modulación angular, como la modulación de frecuencia(FM) o la modulación de fase(PM) utilizan señales de amplitud constante que no se ven afectadas adversamente por las saturaciones de la señal. Así con FM y PM se pueden transmitir niveles más altos de potencia que en los sistemas AM.

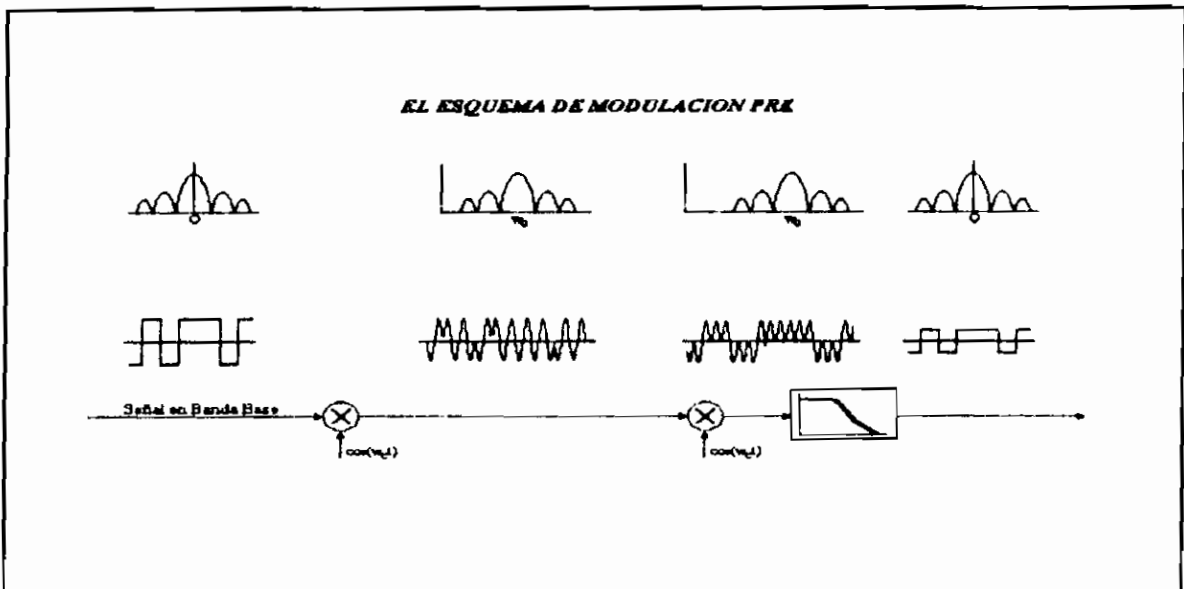


Figura 4.7 El esquema de modulación PSK.

Un esquema bifase de PSK(2-PSK) denominado PRK (Phase Reversal Keying), se muestra en la Figura 4.7[29], e indica que a cada intervalo de señal, se le asigna una de dos fases cada una separada 180° . Las señales con modulación digital en fase, generalmente se denominan PSK (Phase Shift Keying). Son posibles PSK's de múltiples fases. 4-PSK (también llamada QPSK) y 8-PSK son los más comunes ejemplos de PSK de fase múltiple. La PSK ha sido la más popular técnica de modulación para densidades medias de información en aplicaciones de alta eficiencia. Su popularidad se debe primeramente a su desarrollo constante, insensibilidad a las variaciones de nivel, y buen desempeño para con los errores.

Una expresión general para un n-PSK se muestra en la ecuación (4.1)[29]. Esta expresión asume que se emplea un 100% de modulación. Esto es, el desplazamiento de fase de un intervalo a otro puede estar dentro de -180° a $+180^\circ$.

$$x(t) = \cos\left(\omega_c + \frac{m_n(t) \cdot \Delta\phi}{2}\right) \quad (4.1)$$

donde $\Delta\phi=2\pi/n$ es la separación entre las fases adyacentes de la señal, y $m_n(t)$ es una señal simétrica sin retorno a cero de n-ésimo nivel, siendo estos: $\pm 1, \pm 3, \pm 5, \dots$

4.2.2.2.2 Representaciones en cuadratura de señales:

Las representaciones en cuadratura involucran formas de onda sinusoidales de fase arbitraria como una combinación lineal de ondas coseno y seno con inicios nulos de fase. La derivación de esta representación se muestra en la siguiente ecuación.[29]

$$\cos(\omega_c t + \phi) = \cos\phi \cdot \cos(\omega_c t) - \text{sen}\phi \cdot \text{sen}(\omega_c t) \quad (4.2)$$

Nótese que $\cos\phi$ y $\text{sen}\phi$ son constantes para un intervalo de señalización, y así representan coeficientes para expresar $\cos(\omega_c t + \phi)$ como una combinación lineal de las señales $\cos\omega_c t$ y $\text{sen}\omega_c t$. Como $\cos\omega_c t$ y $\text{sen}\omega_c t$ están desfasados 90° uno con respecto al otro, son ortogonales en un diagrama fasorial y así se dice que están "en cuadratura". En esencia, $\cos\omega_c t$ y $\text{sen}\omega_c t$ representan a vectores básicos en un diagrama fasorial bi-dimensional. A la señal coseno se la refiere usualmente como la señal I; y, a la seno, como la señal Q. La siguiente tabla muestra a los coeficientes de señal para una modulación 4-PSK.

Valor de datos	Coeficientes en cuadratura		Señal compuesta
	$\cos(\omega_c t)$	$\text{sen}(\omega_c t)$	
01	0.707	-0.707	$\cos(\omega_c t + \pi/4)$
00	-0.707	-0.707	$\cos(\omega_c t + 3\pi/4)$
10	-0.707	0.707	$\cos(\omega_c t - 3\pi/4)$
11	0.707	0.707	$\cos(\omega_c t - \pi/4)$

El uso de los moduladores en cuadratura viene siendo una continuación de los esquemas de modulación de frecuencia análoga. Los moduladores en cuadratura pueden generar cualquier señal modulada en amplitud, frecuencia o fase.

4.3 CANAL DE TRANSMISION, INCIDENCIA DE RUIDO Y PERTURBACION

Luego de establecer las fases de adquisición, y procesamiento de imágenes, éstas (su flujo digital) se pueden transmitir a través de canales en los que se halla la incidencia de ruido y perturbaciones tales como el corrimiento de fase, IIS (interferencia inter-símbolo), distorsión de amplitud y fase.

4.3.1 TIPOS DE CANALES:

Un canal de comunicación puede ser un medio físico como un par telefónico, y se puede modelar en un medio lógico como el Soft-ware informático o en un modelo estadístico dentro del cual la información se puede procesar. Dentro de este último, se pueden distinguir dos tipos de canales:[35]

(a) **Canal con memoria:** Es aquel en el cual el resultado obtenido en el receptor depende de señales previamente transmitidas.

(b) **Canal sin memoria:** Es aquel en el cual la salida del canal solo depende de la señal transmitida en ese instante.

Hasta aquí se ha asumido que el canal entre el codificador y el decodificador es un canal sin ruido. Para que no sea tan severa la acción perturbadora en canales con ruido, lo que se hace es añadir redundancia a los datos al añadir bits de corrección de errores. Así un compromiso adecuado entre codificación de fuente (con redundancia removida) y codificación de canal (con redundancia inyectada) se tiene que alcanzar en el diseño de sistemas de compresión de datos. A menudo los códigos de corrección de errores se diseñan para reducir la probabilidad de los errores de bits.

y por simplicidad, una igual protección se la provee a todas las muestras. Para los algoritmos de compresión de imágenes, esto no minimiza la totalidad de los errores. Lo que se busca es hallar métodos de codificación que minimicen la totalidad del error cuadrático medio. Las imágenes mostradas en la Figura 4.9 dan una idea de como actúa el canal ruidoso en una transmisión real de una imagen si esta se comprime con alguna técnica basada en transformada

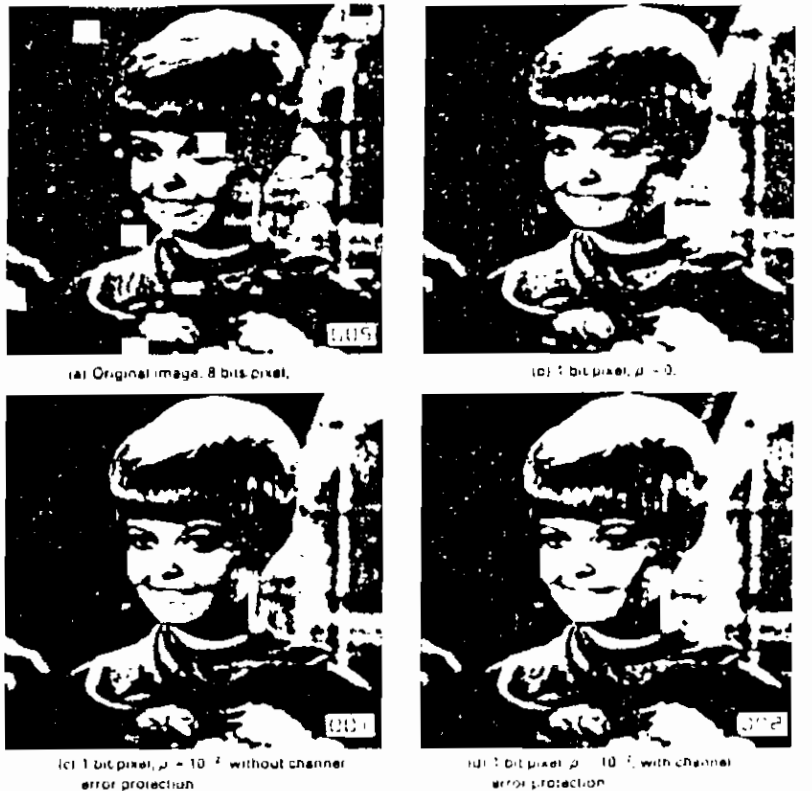


Figura 4.8: Codificación por transformada en presencia de ruido de canal.

Así la imagen (a) muestra los errores que ingresan a la imagen si se codifica con 8bits/pixel; la imagen (b) con una codificación de 1bit/pixel y un BER=0; la imagen (c) a 1bit/pixel, con BER=10⁻² sin protección contra errores de canal; y la imagen (d) a 1bit/pixel, BER=10⁻² y con protección contra errores. Para esta última, se consigue una mejora de 10dB en el SNR sobre la imagen (c) haciendo de la protección contra errores muy necesaria si lo que se requiere es una alta eficiencia de transmisión.

4.4 ETAPA DE RECEPCION-DESCOMPRESION

En lo que respecta a la recepción de imágenes se supone que éstas (sus datos digitales representativos) han atravesado el canal de comunicación y por tanto dicho flujo de información indiscutiblemente se ha contaminado. El proceso de demodulación simplemente sigue una secuencia inversa a la modulación, devolviendo los datos digitales (tren de pulsos) a su forma analógica.

Luego de la demodulación, se procede a la decodificación de los datos digitales, y dentro de esta se procede a corregir los errores introducidos, descomprimir el flujo digital, convertir estos datos a su formato analógico mediante el post-procesamiento de su información, que puede ser un convertidor de digital a análogo, esto es, permitir que el transductor de salida reciba la información.

4.4.1 CORRECCION DE ERRORES:

Así como los métodos de modulación en fase son tan populares en los esquemas prácticos de transmisión de imágenes, así mismo la popularidad en los esquemas de control de errores se orientan a los códigos convolucionales y a los códigos bloque, uno de estos esquemas es la *Corrección Directa de Errores* o *FEC*. La FEC utiliza ingeniosos procedimientos matemáticos para disminuir considerablemente los efectos causados por el ruido introducido en el canal, en la transmisión de datos.[35] La teoría de los códigos bloque es mucho más antigua y rica que la de los códigos convolucionales y la mayor parte del tiempo se ha consumido en tratar de encontrar códigos óptimos y en desarrollar técnicas que permitan una fácil implementación de estos. Como resultado de esta investigación han aparecido los códigos cíclicos muy fáciles de implementar, los códigos denominados *BCH* que se caracterizan principalmente por su capacidad para la corrección de múltiples errores y por sus eficientes algoritmos de decodificación.

4.4.2 ETAPA DE DESCOMPRESION:

En el proceso de descompresión la implementación de algoritmos y esquemas ya se han indicado en el Capítulo II, según el método de compresión utilizado en el terminal #1 (terminal transmisor), se implementa su contraparte descompresiva.

Debido a la naturaleza de la imagen, y al sistema visual humano, es posible tener límites aceptables de error en los datos enviados por el canal aún mayores que los límites establecidos para la recepción de datos no-imágenes. Así por ejemplo para una transmisión de datos bancarios un BER de 10^{-7} resulta ser un límite aceptable más no alcanzable, en cambio que para una transmisión de televisión digital digamos NTSC digital, un BER de hasta 10^{-2} es bastante bueno, aunque para esquemas de video de alta definición como monitoreo médico requeriría al menos de 10^{-4} en su tasa de bits erróneos.

Por estas razones, la sofisticación alcanzada en sistemas correctores de errores de transmisión no es que se la menosprecie, es solo que se requiere de menos esfuerzo por parte de los algoritmos desarrollados en la corrección de errores, así el costo en equipo de recepción baja y por tanto la eficiencia se vuelve mayor.

4.4.3 TRANSMISION DE IMAGENES EN DOS TONALIDADES:

Una desventaja para la eficiente codificación de fuente es el incremento de su sensibilidad a los canales erróneos. Por ejemplo, en la codificación longitudinal un sencillo bit erróneo arruinará al resto de la línea barrida. (Se asume que las palabras de sincronismo de línea se utiliza para evitar que el error se propage a las siguientes líneas). Este incremento de la sensibilidad a los canales ruidosos se compensa parcialmente por el hecho de que después de la compresión se tiene pocos bits por imagen y así pocas oportunidades de tener un daño por canal ruidoso.

Por ejemplo, a resoluciones de 100ppp (puntos por pulgada), una imagen de dos tonos de $8\frac{1}{2}'' \times 10''$ contiene aproximadamente 10^6 bits (sin compresión). Si se transmite esta imagen con un convencional esquema MIC a través de un canal con una probabilidad de error de 10^{-6} , se tendrá un promedio de 1 error por imagen. Si se utiliza un esquema más eficiente de codificación con una compresión de datos de 10:1, entonces se tendrá 10^5 bits por imagen, y por lo tanto el promedio de error será de 1 en 10 imágenes (10 imágenes contendrán un error).

Para dar alguna idea del efecto de los errores en un canal en imágenes codificadas, se presenta un ejemplo concreto de una sección de 7.7'' de ancho por 1.7'' de alto de una carta que fue muestreada a aproximadamente 100ppp. Una computadora experimental[12] codificó este párrafo de prueba con codificación longitudinal (Factor de Compresión=4:1) y por la convencional MIC (punto por punto) a través de canales con ruido del tipo ráfaga. El ruido se lo obtuvo de una medición de un canal telefónico común con intensidades que se incrementan obteniéndose así los párrafos (a), (b), (c), (d), y (e). Se utilizó una palabra de sincronismo de línea de 30-bits. Los resultados se muestran en las Figuras 4.9[12], 4.10[12] y 4.11[12].

La Figura 4.9[12] muestra imágenes reconstruidas con el convencional decodificador MIC. Nótese que a medida que aumenta la tasa de error, la claridad e inteligibilidad del párrafo se pierde aún así el texto podría leerse.

La Figura 4.10[12] muestra a las imágenes reconstruidas según la codificación longitudinal con detección de errores y utilizando una repetición de líneas. Es interesante la diferencia entre el párrafo (d) y (e) en los que a pesar de que la tasa de error es mayor (párrafo e) el texto puede aún leerse al contrario del párrafo d.

La Figura 4.11[12] muestra a las imágenes reconstruidas para la codificación longitudinal. Con este método se halla una mayor compresión, pero los resultados del desempeño logrado se muestran no muy favorables. Es así que a partir del párrafo (b) el texto se vuelve menos confiable y mientras más aumenta la tasa de error, el texto se contamina con defectos (como líneas y desfase de puntos). En el terminal receptor, una línea barrida se considera con error si el decodificador longitudinal no da el número correcto de píxeles para esa línea. Entonces la línea precedente se repite (Este esquema requiere un almacenamiento de dos líneas barridas en el terminal receptor). Se nota que la detección de errores y el método de repetición de líneas no es muy efectivo cuando los errores se presentan en varias líneas barridas.

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

a

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

b

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

c

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

d

Fig. 13. Reconstructed images, point by point transmission.
(a) error rate 10^{-4} , (b) 6.5×10^{-4} , (c) 10^{-3} , (d) 1.7×10^{-3} .

Figura 4.9 Recepción en transmisión punto a punto.

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

a

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

b

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

c

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

d

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could come.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

e

Fig. 14. Reconstructed images, runlength coding (a) error rate 10^{-4} , (b) 6.5×10^{-4} , (c) 10^{-3} , (d) 1.7×10^{-3} , (e) 4.4×10^{-3}

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could comm.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

a

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could comm.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

b

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could comm.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

c

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could comm.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

d

As you may recall, I did mentioned that we should invite you visiting and talking of your research at Sherbrooke this fall. So I am writing now these lines to let you know that we wish you could comm.

As for the academic task in this period, I shall teach a course on the organization and management of information. Besides it, we are also interested in the compression of Data Problem where I have two graduate students to work on it.

e

Fig. 15. Reconstructed images, runlength coding with error detection and line repeating. (a) error rate 10^{-4} , (b) 6.5×10^{-4} , (c) 10^{-3} , (d) 1.7×10^{-3} , (e) 4.4×10^{-3} .

Figura 4.11 Recepción con CL y corrección de errores.

Capítulo
5

*Aplicación de los
Métodos de Compresión
a una Imagen de Prueba*

CAPITULO V

Aplicación de los Métodos de Compresión a una Imagen de Prueba

5.1 ESTUDIO DE LA IMAGEN A COMPRIMIR

A fin de utilizar una imagen real para ejecutar la aplicación de los métodos de compresión se va a utilizar una imagen tipo de 256 tonalidades de gris que es típica en los anales de la IEEE para procesamiento de imágenes, LENA.



Figura 5.1 LENA

Como se puede notar, la imagen tiene ciertas características importantes: bordes (que significan altas frecuencias), pequeños detalles en donde podría perderse mas información, grandes zonas de una tonalidad definida en donde algún error de decodificación se visualizaría como pequeñas manchas, y ciertas zonas de gran intensidad luminosa, lo que significa palabras binarias muy grandes.

Aunque esta imagen originalmente tiene dimensiones de 256x256 pixeles, para la simulación solo se requerirá de 256x240 pixeles, esto para que se ajuste a las medidas de la pantalla de trabajo del programa. Pese a que existen monitores de muy alta resolución y de varias tonalidades de un solo color, los tonos grisáceos de la imagen se re-escalarán a fin de poder visualizar más cómodamente, sin embargo todos los cálculos que se realizan se los hace en base a la imagen original, es decir con sus 256 tonalidades de gris.

5.2 DESCRIPCION Y ANALISIS DE LOS METODOS DE COMPRESION A UTILIZARSE

Para esta aplicación, se podría utilizar cualquiera de los métodos del capítulo II, sin embargo la idea es realizar una comparación de dos métodos en particular, un método innovador, la codificación por truncamiento de bloque (CTB), y un método comercialmente muy utilizado, la codificación por medio de la transformada discreta coseno (TDC).

Se realiza también la codificación por medio del MIC Diferencial, una técnica también utilizada a nivel comercial pero que tiene un gran defecto y este se muestra en la simulación que por sí sola es muy vulnerable en presencia de errores en la codificación, esto es, si un solo error ocurriese en alguna parte del proceso (en la compresión por ejemplo), la estructura se iría abajo por completo, pues una muestra codificada depende siempre de la anterior y si esta está errada todas las futuras muestras se verán afectadas. De otro lado la codificación sin errores es un caso bastante eficaz de compresión teniéndose que en efecto la entropía se reduce hasta niveles de 1bit/pixel, y el error cuadrático medio es prácticamente cero.

5.2.1 CODIFICACION POR TRUNCAMIENTO DE BLOQUE:

Este es un método muy interesante de codificación que se lo desplegó en la sección 2.1.4.5.1, ya se han visualizado sus ecuaciones y su comportamiento en las secciones 2.1.4.5.1 y 2.2.6 respectivamente, siendo notorias sus ventajas en la codificación de imágenes tanto estáticas como en movimiento, así como para imágenes monocromáticas.

El algoritmo que se va a utilizar en esta aplicación se puede visualizar mejor si se presenta como diagrama de flujo, pues así será como se lo aplicará en el programa computacional que realizará la codificación.

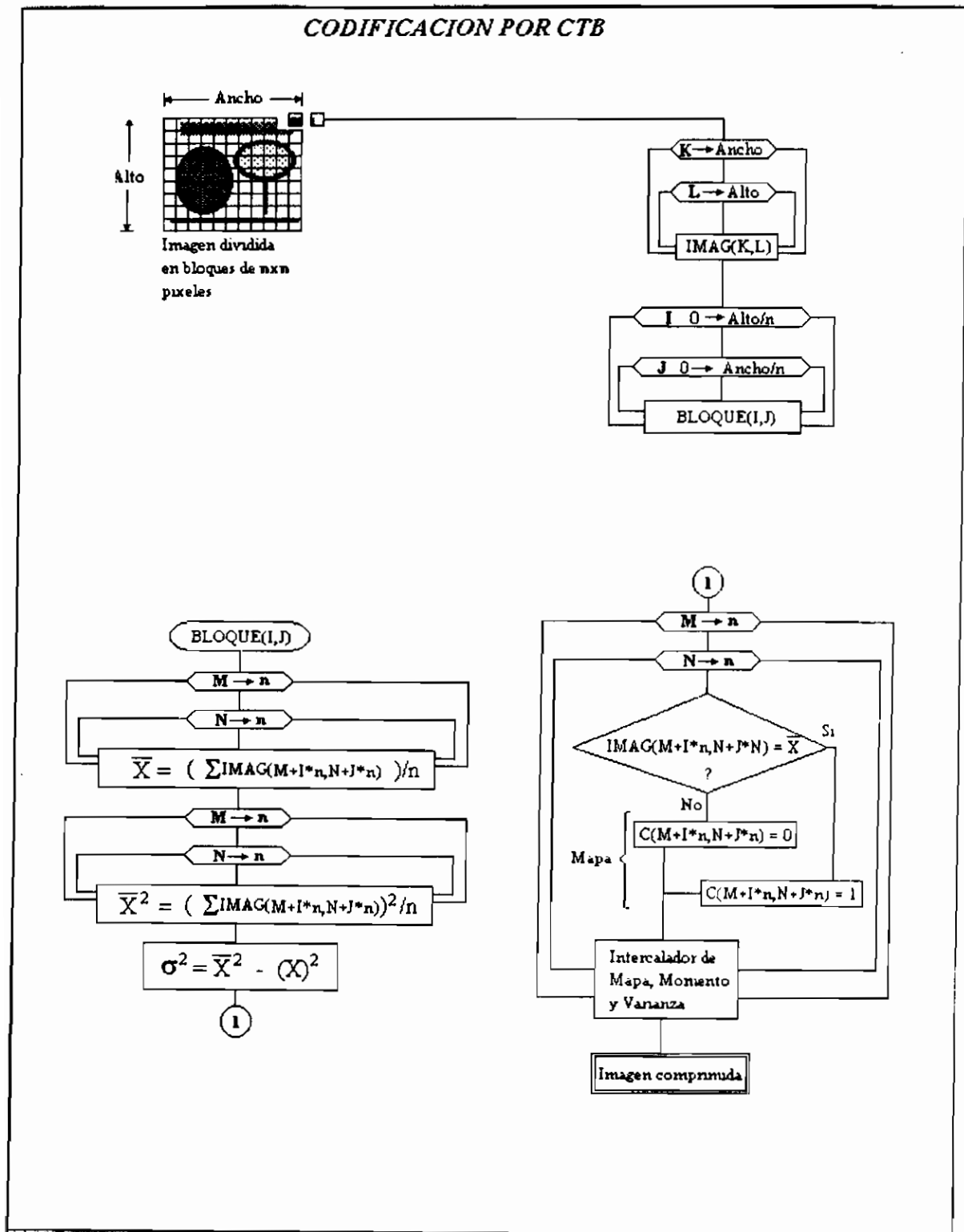


Figura 5.2 Diagrama de bloques del compresor CTB.

5.2.2 CODIFICACION POR MEDIO DE LA TRANSFORMADA COSENO:

Este es un método muy práctico de codificación que se lo desplegó en la sección 2.1.3.5, conjuntamente con las otras transformadas; ya se ha visualizado sus ecuaciones y su comportamiento en la sección 2.1.3, notando sus bondades en la codificación de imágenes tanto estáticas como en movimiento, así como para imágenes monocromáticas y a color. Las ecuaciones básicas que se utilizarán en la codificación por TDC, debido a que se requiere establecer una forma de evaluación del tipo lazo y por bloques en la implementación de la TDC, una notación de la transformación directa es:

$$C(u, v) = \alpha(u) \alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \frac{(2x+1) u \pi}{2N} \cos \frac{(2y+1) v \pi}{2N} \quad (5.1)$$

para $u=0, 1, 2, \dots, N-1$;

$v=0, 1, 2, \dots, N-1$;

$f(x, y)$ es el valor del pixel de la fila x , columna y .

$C(x, y)$ es el coeficiente transformado.

La transformación inversa es:

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u) \alpha(v) C(u, v) \cos \frac{(2x+1) u \pi}{2N} \cos \frac{(2y+1) v \pi}{2N} \quad (5.2)$$

para $x=0, 1, 2, \dots, N-1$;

$y=0, 1, 2, \dots, N-1$;

con α definiéndose como:

$$\alpha(i) = \begin{cases} \sqrt{\frac{1}{N}} & \text{si } i=0 \\ \sqrt{\frac{2}{N}} & \text{si } i=1, 2, \dots, N-1 \end{cases} \quad (5.3)$$

El procedimiento algorítmico a utilizar en esta aplicación se puede visualizar mejor si se lo presenta como diagrama de bloques, que representa como se lo aplicará en el programa computacional que realizará la codificación.

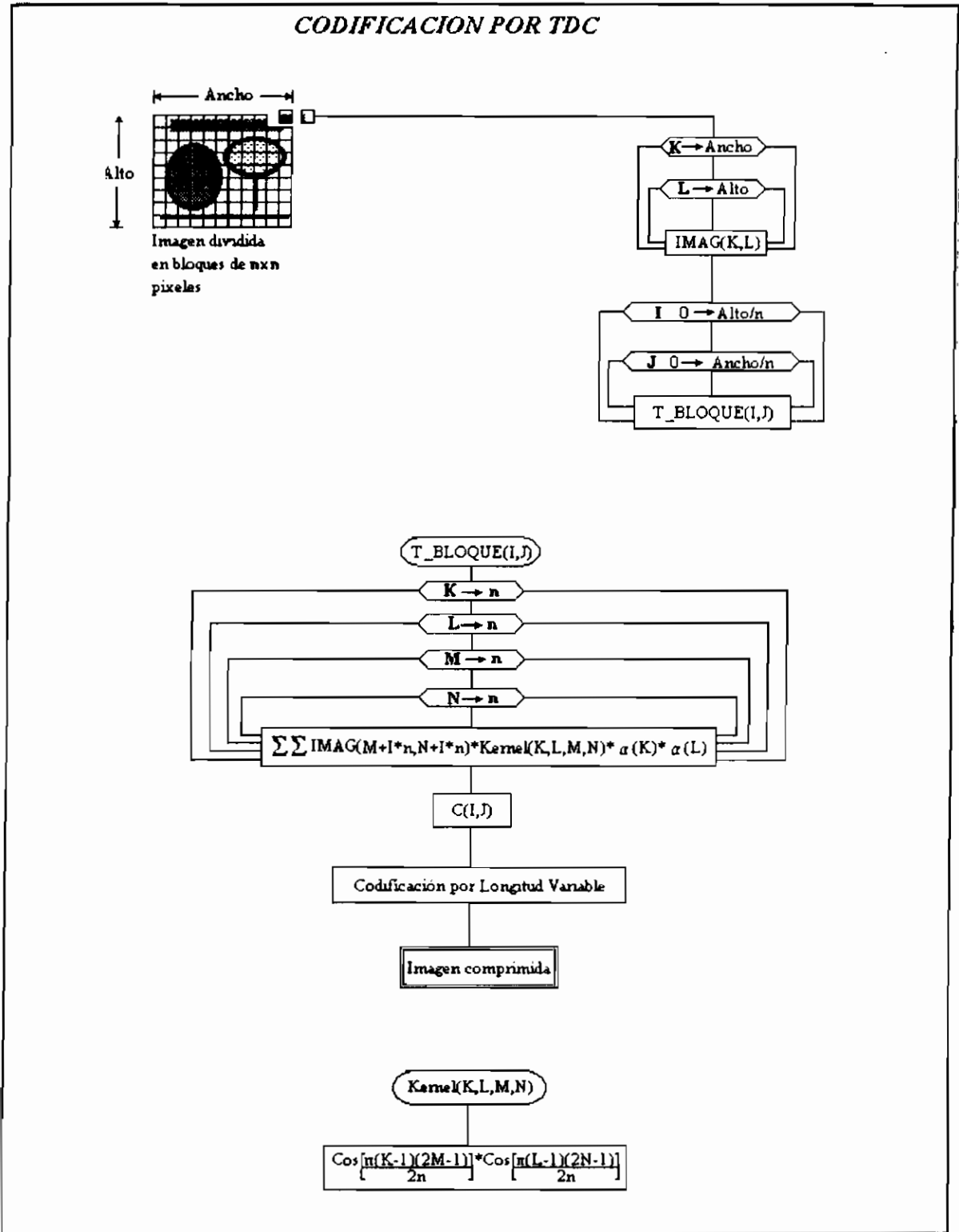


Figura 5.3 Diagrama de bloques del compresor TDC.

5.2.3 CODIFICACION POR MEDIO DE LA MICD:

Este método fue explicado en la sección 2.1.2.1 y su implementación algorítmica puede verse a continuación:

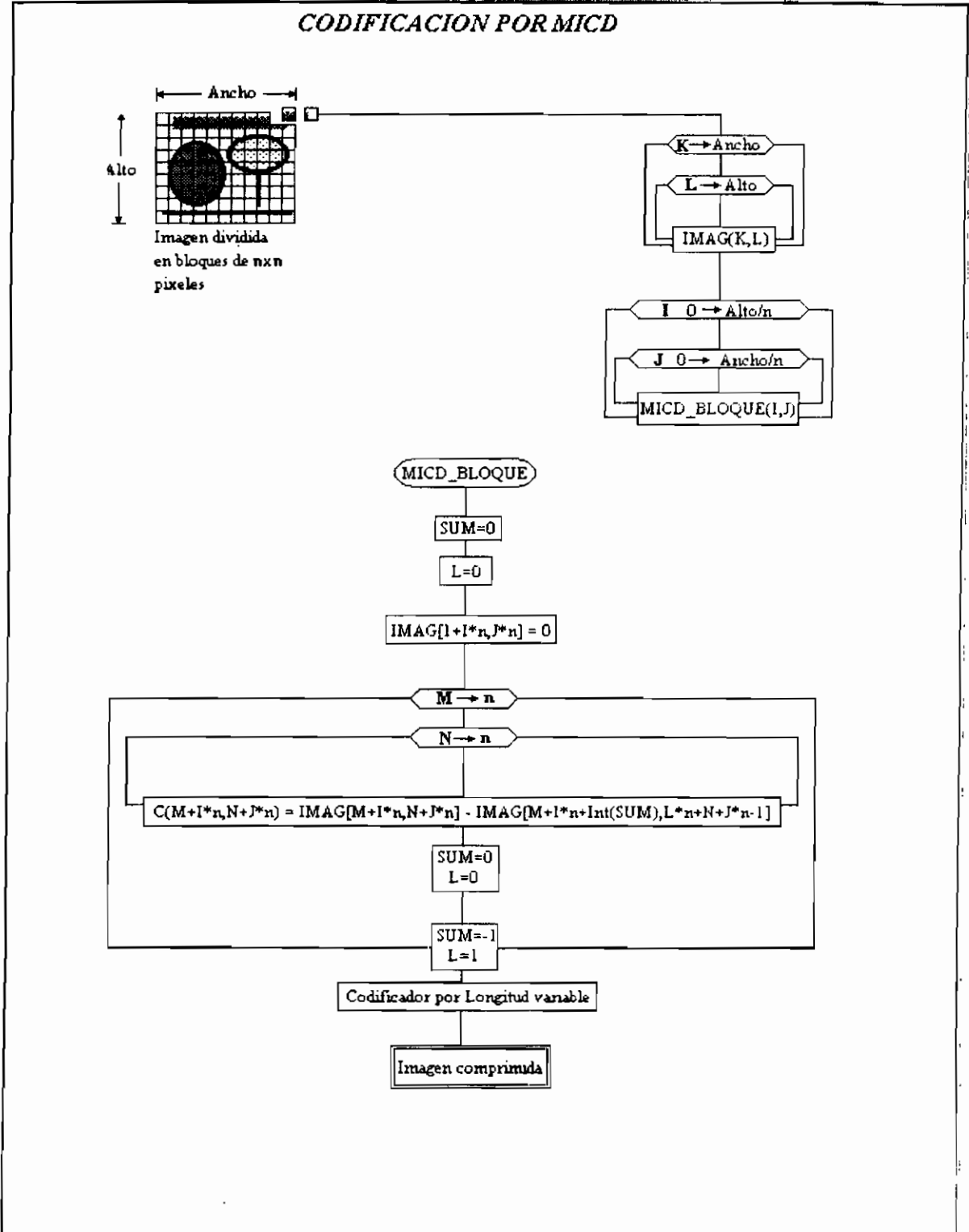


Figura 5.4 Diagrama de bloques del compresor MICD.

5.3 DESCRIPCION DEL PROGRAMA COMPUTACIONAL UTILIZADO

Debido a su capacidad de trabajo en forma de "Procedimientos", depurador incorporado; y, por tener una característica de lenguaje "Estructurado", se ha escogido al Lenguaje de Programación PASCAL versión 7.0 como herramienta informática de implementación de los algoritmos de compresión a simularse. El compilador PASCAL permite la ejecución de un programa principal localizado en el segmento de código de la memoria y puede acceder a otros segmentos como segmentos de datos y así realizar una compilación en bloques llamados las **UNITS**. El programa se estableció con cinco bloques auxiliares y de esta forma la ejecución total de la simulación consta del programa principal:

COMPIMAG.PAS

y las seis unidades:

**VARIOS.TPU
VARIABLE.TPU
INICIAL.TPU
PROCESO.TPU
CUADROS.TPU
ANALISIS.TPU**

El programa trabaja bajo entorno DOS(Disk Operating System), las partes constitutivas del programa son:

(a) Pantalla de presentación.

(b) Parámetros de la imagen.

- Nombre del archivo en el disco de la imagen.
- Dimensión horizontal: Ancho de la imagen.
- Dimensión vertical: Altura de la imagen.
- Dimensión n del bloque compresor de nxn pixeles.
- Procesamiento completo o presentación de resultados.
- Proceso con incidencia de ruido.

(c) Tipos de compresión.

- Compresión utilizando TDC.
- Compresión utilizando CTB.
- Compresión utilizando MICD.

(d) Tipos de evaluación.

- Visualización de la imagen original.
- Visualización de la imagen comprimida.
- Visualización de la imagen descomprimida.
- Visualización de la imagen diferencia.

(e) Resultados de la compresión.

- Características de la imagen original.
 - Nombre del archivo.
 - Tipo de imagen(color/monocromática).
 - Altura de la imagen(pixeles).
 - Ancho de la imagen(pixeles).
 - Número de pixeles conformadores de la imagen.
 - Bits asignados por pixel.
 - Tamaño del archivo-imagen(en bytes).
 - Varianza de la imagen.

- Características de la imagen comprimida por TDC, CTB, y MICD.
 - Nombre del archivo.
 - Altura de la imagen(pixeles).
 - Ancho de la imagen(pixeles).
 - Número de pixeles conformadores de la imagen.
 - Bits asignados por pixel(nivel de compresión).
 - Tamaño del archivo-imagen comprimida(bytes).
 - Tamaño del archivo-imagen descomprimida(bytes).
 - Error cuadrático medio.
 - Relación Señal/Ruido(SNR en dB).
 - BER promedio.

(f) Conclusión.

5.3.1 PROGRAMA PRINCIPAL: COMPIMAG.PAS:

En este programa, se realiza el algoritmo total de compresión, se llama a los diferentes procedimientos por su nombre y se ejecutan secuencialmente, para luego tener la opción de simular nuevamente otra compresión o para terminar la sesión de trabajo y devolver el control al sistema operativo.

Un flujograma explicativo se muestra en la Figura 5.5:

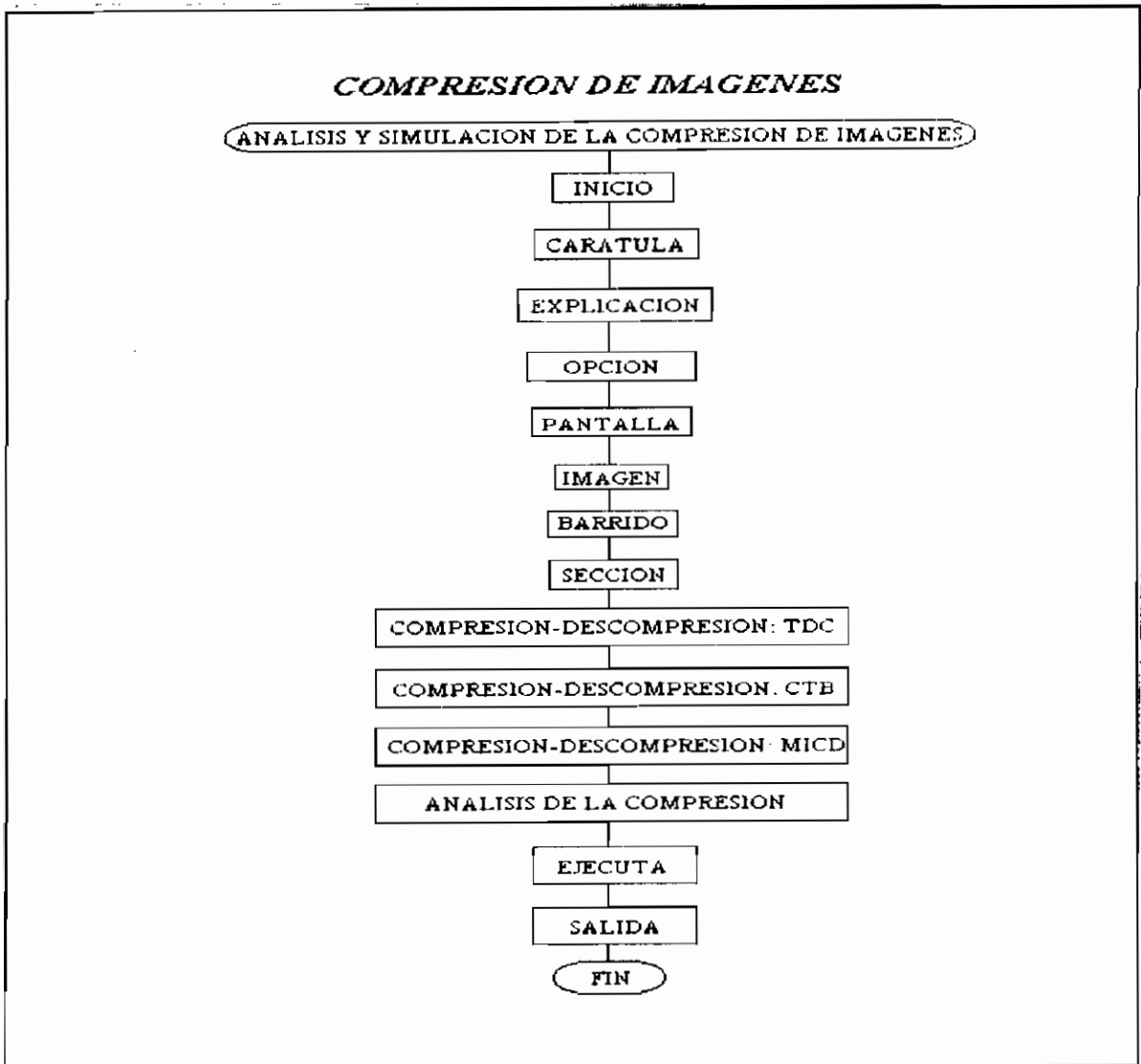


Figura 5.5 Diagrama de Bloques general del programa.

5.3.2 UNIDAD VARIOS.TPU:

Esta es la unidad de presentación, aquí se encuentran los procedimientos:

SON: Genera una señal audible de 200ms de duración.

SIGA: Espera a que una tecla sea presionada.

BORDE: Dibuja dos rectángulos circunscritos como ventana.

PREGUNTA: Realiza la petición de opciones de trabajo al usuario al pedir la ubicación de archivos, las dimensiones de la imagen, las dimensiones del bloque, el tipo de procesamiento, si es una compresión completa o solo una visualización, y la opción de si la simulación se la realiza con incidencia de ruido.

CUADRO: Abre una ventana en coordenadas específicas.

BORRA: Cierra una ventana en coordenadas específicas con un determinado color.

BORRAS: Cierra una ventana en coordenadas específicas con el mismo color del fondo.

GTEXTO: Localiza en la pantalla la forma 1 de texto.

TTEXTO: Localiza en la pantalla la forma 2 de texto.

RTEXTO: Localiza en la pantalla la forma 3 de texto.

MENS_I: Localiza en la ventana 1 un diálogo.

MENS_II: Localiza en la ventana 2 un diálogo.

ESTADO: Muestra el estado del proceso.

MARCO: Un segundo tipo de ventana.

PEDESTAL: Un fondo adecuado para los diálogos.

LOZA: Un fondo adecuado para las imágenes.

EXISTE: Verifica si el archivo de la imagen existe en el disco si no existe, repite la petición.

5.3.3 UNIDAD VARIABLE.TPU:

En esta unidad se dimensionan y nombran las variables a utilizar tanto en el programa principal como en las otras unidades. Las variables son del tipo entero (INTEGER), real (REAL), real doble (BOUBLE), caracter (CHAR), cadena (STRING), arreglo (ARRAY), y booleana (BOOLEAN).

5.3.4 UNIDAD INICIAL.TPU:

En esta unidad se hallan los procesos de inicializar el modo gráfico de la pantalla, la pantalla de presentación inicial, la explicatoria y la final:

INICIO: Inicializa a la pantalla en modo gráfico.

CARATULA: Dibuja la pantalla de presentación.

PANTALLA: Dibuja la pantalla de trabajo.

EXPLICACION: Dibuja cierto texto de explicación al usuario.

SALIDA: Dibuja la pantalla final de la simulación.

PUNTO: Dibuja un punto en la pantalla pero de tonalidad cambiada.

5.3.5 UNIDAD CUADROS.TPU:

La imagen como elemento de trabajo se la toma como un arreglo matricial de 256x240 elementos enteros de valores entre 0 y 255 correspondientes a los valores de color que un pixel pudiera tomar.

IMAGEN1: Se rescata del disco el archivo de la imagen de prueba. La imagen está en el formato BitMap(BMP) que es muy popular en las aplicaciones *Windows* existentes en la actualidad. A fin de obtener la imagen, el procedimiento *IMAGEN* realiza la conversión del formato BMP a un formato "crudo" (estado de la imagen sin procesamiento) de la imagen es decir en el arreglo de 256x240.

IMAGEN2: Si el proceso de cálculo ha terminado por completo, se generan 15 archivos de datos estos se visualizarán en la ejecución del programa y uno de estos es el que se muestra aquí, la imagen de trabajo.

IMAGEN3: Se visualiza a la imagen sin compresión a 256 tonalidades de gris como LENA (LENA.RAW).

IMAGEN4: Se visualiza a la imagen de 256 tonalidades sin compresión.

RUIDO: Según el nombre de la imagen-archivo de entrada, genera un ruido que contamina a esta imagen, este ruido da un aspecto de haber sido "espolvoreado" sobre la imagen, al igual que el ruido "spike" o "sal y pimienta".

5.3.6 UNIDAD PROCESO.TPU:

Es quizá la unidad más importante, pues aquí se hallan todos los procedimientos de cálculo de compresión por los tres métodos señalados:

BARRIDO: Barre los 256x240 pixeles de la imagen.

SECCION: Divide a la imagen en bloques de $n \times n$ pixeles.

T_BLOQUE: Subproceso que aplica la TDC a un bloque.

COMP_TRAN: Proceso de aplicación de la TDC por bloques.

BLOQUE: Subproceso que aplica la CTB a un bloque.

COMP_TB: Proceso de aplicación de la CTB por bloques.

COMP_MICD: Proceso aplicación del MICD a la imagen.

MUESTRA_IMAG_COMP_TR_COS: Dibuja la compresión por TDC.

MUESTRA_IMAG_COMP_TB: Dibuja la compresión por CTB.

MUESTRA_IMAG_COMP_MICD: Dibuja la compresión por MICD.

T_DESBLOQUE: Subproceso que descomprime un bloque por TDC.

DESCOMP_TRAN: Proceso de descompresión por bloques(TDC).

DESBLOQUE: Subproceso que descomprime un bloque por CTB.

CUANTIFICA: Subproceso de cuantificación de parámetros.

DESCOMP_TB: Proceso de descompresión por bloques(CTB).

DESCOMP_MICD: Proceso de descompresión por MICD.

MUESTRA_IMAG_DESCOMP_TR_COS: Dibuja la imagen descomprimida.

MUESTRA_IMAG_DESCOMP_TB: Dibuja la imagen descomprimida.

MUESTRA_IMAG_DESCOMP_MICD: Dibuja la imagen descomprimida.

DIFERENCIA_TDC: Dibuja la diferencia de imágenes.

DIFERENCIA_CTB: Dibuja la diferencia de imágenes.

DIFERENCIA_MICD: Dibuja la diferencia de imágenes.

PORCENTAJE: Dibuja un medidor de procesamiento que monitorea la actividad del programa en un rango de 0 a 100% de ejecución.

LIMPIA: Cada vez que termina un proceso, encera el medidor.

5.3.7 UNIDAD ANALISIS.TPU:

En esta se muestran los resultados de la compresión alcanzados por los tres métodos, y además se calculan magnitudes importantes como el error cuadrático medio, la tasa de entropía, el SNR y el BER.

EJECUTA: Comienza el proceso de análisis.

FORMA: Subproceso que muestra a una imagen y sus parámetros.

BUCLE: Lazo repetitivo en donde se muestran los resultados.

DIB_DAT: Dibuja la imagen original.

DIB_TDC: Dibuja la imagen descomprimida por TDC.

DIB_CTB: Dibuja la imagen descomprimida por CTB.

DIB_MICD: Dibuja la imagen descomprimida por MICD.

ECM: Proceso de cálculo del error cuadrático medio.

VAR_IMA: Proceso de cálculo de la varianza y el SNR.

Por último, si la simulación se inicia con la imagen de nombre **IMAGEN.BMP** o **IMAGEN.RAW** que es con la que se trabaja, luego de la ejecución completa de la simulación se crean los siguientes archivos:

IMAGEN.DAT: Es la imagen "cruda" original sin ningún tipo de codificación.

IMAGEN.BLO: Aquí se hallan ciertos resultados de la ejecución como el tamaño del bloque, las dimensiones de la imagen, y todos los valores de varianza, primer y segundo momentos, todos estos datos con el fin de utilización del programa durante su ejecución y que el usuario pudiera acceder para alguna cuestión particular.

IMAGEN.TCC: La imagen comprimida por medio de la TDC.

IMAGEN.TCK: La imagen comprimida(TDC) y cuantificada.

IMAGEN.TCD: La imagen descomprimida por medio de la TDC.

IMAGEN.DFC: La imagen diferencia según la TDC.

IMAGEN.TBC: La imagen comprimida por medio de la CTB.

IMAGEN.TEK: La imagen comprimida(CTB) y cuantificada.

IMAGEN.TBD: La imagen descomprimida por medio de la CTB.
IMAGEN.DFB: La imagen diferencia según la CTE.
IMAGEN.MDC: La imagen comprimida por medio de MICD.
IMAGEN.MDK: La imagen comprimida(MICD) y cuantificada.
IMAGEN.MDD: La imagen descomprimida por medio de MICD.
IMAGEN.DFM: La imagen diferencia según MICD.
IMAGEN.DAR: La imagen a la que se inyecta ruido.

5.4 VALORACION CUANTITATIVA Y CUALITATIVA

Utilizando las opciones del programa se realiza una comparación de las ejecuciones de los tres métodos de compresión, cuando se realiza una codificación sin ruido y cuando éste está presente.

Se ingresa la siguiente imagen de prueba:



IMAGEN de PRUEBA

LENA .RAW

Ancho: 256 pixeles

Alto: 240 pixeles

Tamaño del bloque: 8x8 pixeles

Tamaño del archivo: 61440 bytes

Varianza: 8.5759

Media: 7.2627

Valor cuadrático medio: 61.323

5.4.1 COMPRESION SIN RUIDO:

Se seleccionan los siguientes parámetros:

Nombre de la imagen: LENA.RAW
Altura de la imagen: 256 pixeles
Ancho de la imagen: 240 pixeles
Dimensión del bloque: 8 pixeles x 8 pixeles
Proceso completo: Sí
Incidencia de ruido: No

Imagen comprimida por medio de la TDC:

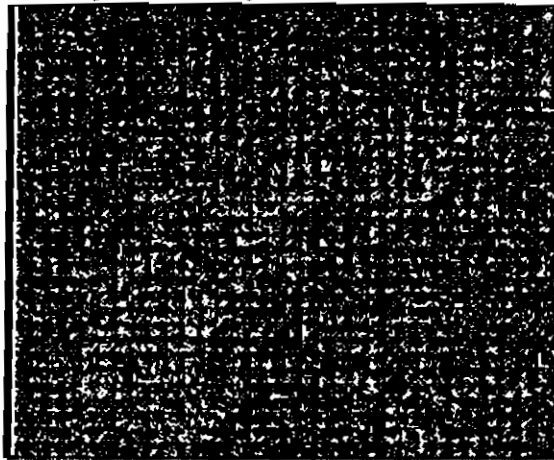


IMAGEN COMPRIMIDA - TDC

Imagen comprimida por medio de la CTB:

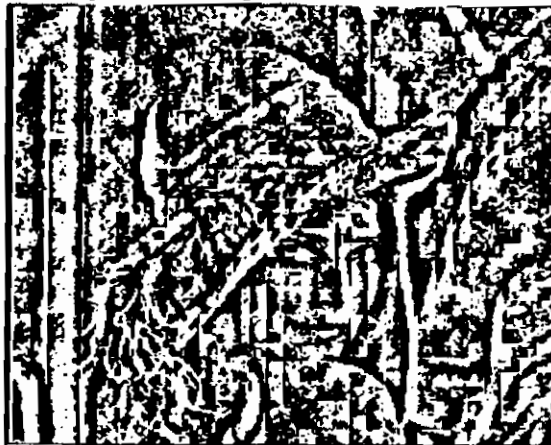


IMAGEN COMPRIMIDA - CTB

Imagen comprimida por medio de MICD:



IMAGEN COMPRIMIDA - MICD

Si se comparan estas tres imagenes con la imagen original, se nota a simple vista el efecto compresor, los niveles de brillo han disminuido notablemente, esto significa que el número de bits requerido por pixel es menor al que originalmente se necesitaba, esto es ya un modo de compresión.

El número de bytes N de las imágenes comprimidas son:

$N_{TDC} = 55345$

$N_{CTB} = 9600$

$N_{MICD} = 46731$

Existe entonces una tasa mayor de compresión según el método CTB, una relación de (Imagen original: 61440bytes):

$$\frac{N_{original}}{N_{MICD}} = 1.315 \quad \frac{N_{original}}{N_{CTB}} = 6.4 \quad \frac{N_{original}}{N_{TDC}} = 1.12$$

Es decir las expectativas hasta ahora son favorables para la codificación CTB por su mayor tasa de compresión. Y a continuación se muestra a la imagen descomprimida por los tres métodos:

Imagen TDC descomprimida:



IMAGEN DESCOMPRIMIDA - TDC

Imagen CTB descomprimida:



IMAGEN DESCOMPRIMIDA - CTB

Imagen MICD descomprimida:



IMAGEN DESCOMPRIMIDA - MICD

Si se comparan con la imagen original, a simple vista se puede notar una pérdida de información muy fuerte en las regiones donde existen bordes como las plumas del sombrero o las grandes diferencias de tono de gris muy claro a gris muy oscuro. En la codificación por MICD como puede notarse en la imagen descomprimida no se presenta diferencia alguna. Se pueden presentar los resultados obtenidos con las tres compresiones:

Tamaño del archivo comprimido:

TDC: 55345 bytes

CTB: 9600 bytes

MICD: 46731 bytes

Tamaño del archivo descomprimido:

TDC: 61440 bytes

CTB: 61440 bytes

MICD: 61440 bytes

Bits/pixel necesarios para presentar a la imagen comprimida:

TDC: 2.553

CTB: 7.3325

MICD: 2.87

Error cuadrático medio:

TDC: 0.536

CTB: 0.5388

MICD: 0.000001 (Valor mínimo en el programa)

Relación Señal/Ruido(SNR en dB):

TDC: 16.18

CTB: 12.037

MICD: 69.33

Tasa de bits erróneos(BER):

TDC: 0.256

CTB: 0.325

MICD: 0.000

Varianza de la imagen descomprimida:

TDC: 9.342

CTB: 8.553

MICD: 8.576

Media de la imagen descomprimida:

TDC: 7.510

CTB: 7.259

MICD: 7.263

Valor cuadrático medio de la imagen descomprimida:

TDC: 65.745

CTB: 61.244

MICD: 61.322

Se nota una clara ventaja de desempeño de MICD en contra de CTB y TDC, esto es debido a que la codificación no se la realiza con ruido y debido a la poderosa y sencilla forma de codificación que presenta MICD.

Tal vez la magnitud de los archivos comprimidos no representen la gran compresión que se logra con los tres métodos, esta sin embargo se la visualiza mejor por medio de los Bits/pixel que en el peor de los casos(TDC) es de 1.083 bits/pixel.

Otra característica a notarse es que las relaciones señal/ruido apuntan como mejor a la TDC en contra de la CTB, y por supuesto que es menor que la de MICD debido a que no existe error alguno en la decodificación de esta última. Ya entrando en un caso más cercano a lo real la siguiente es la compresión con cierta cantidad de ruido y en donde la MICD no se presenta tan ideal como en el caso anterior.

5.4.2 COMPRESION CON RUIDO:

Se seleccionan los siguientes parámetros:

Nombre de la imagen: LENA.RAW
Altura de la imagen: 256 pixeles
Ancho de la imagen: 240 pixeles
Dimensión del bloque: 8 pixeles x 8 pixeles
Proceso completo: Sí
Incidencia de ruido: Sí
Porcentaje de ruido: 20%

Imagen comprimida por medio de la TDC:

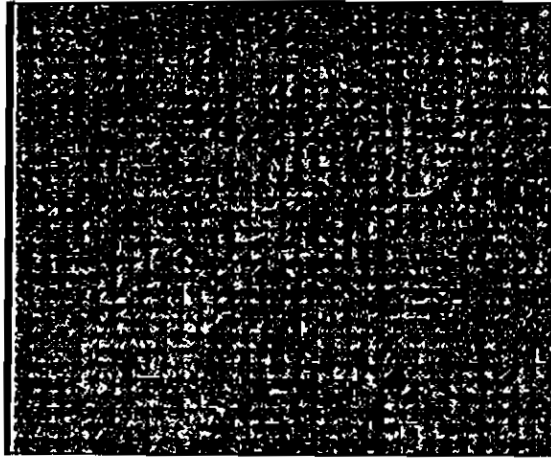


IMAGEN COMPRIMIDA - TDC

Imagen comprimida por medio de la CTB:

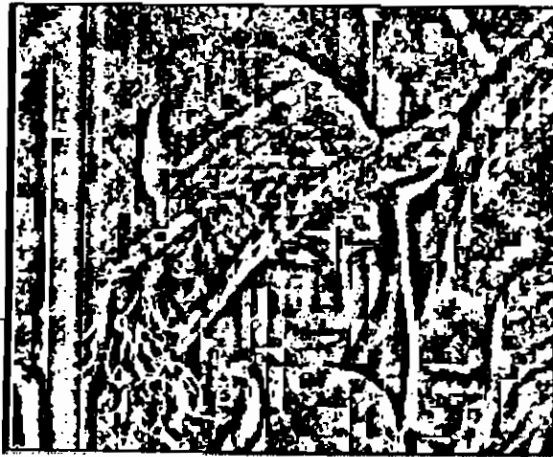


IMAGEN COMPRIMIDA - CTB

Imagen comprimida por medio de MICD:



IMAGEN COMPRIMIDA - MICD

Aquí también se nota el efecto compresor, al observar que los niveles de brillo han disminuido notablemente.

El número de bytes N de las imágenes comprimidas son:

$$N_{TDC} = 52900$$

$$N_{CTB} = 9600$$

$$N_{MICD} = 46739$$

Existe entonces una tasa mayor de compresión según el método de la MICD, una relación de (Imagen original: 61440 bytes):

$$\frac{N_{Original}}{N_{MICD}} = 1.315 \quad \frac{N_{Original}}{N_{CTB}} = 6.4 \quad \frac{N_{Original}}{N_{TDC}} = 1.16$$

Aquí también las expectativas son favorables para la codificación por CTB. A continuación se muestran a la imagen descomprimida por los tres métodos:

Imagen TDC descomprimida:



IMAGEN DESCOMPRIMIDA - TDC

Imagen CTB descomprimida:



IMAGEN DESCOMPRIMIDA - CTB

Imagen MICD descomprimida:



IMAGEN DESCOMPRIMIDA - MICD

Si se comparan con la imagen original, a simple vista se puede notar una pérdida de información muy fuerte en las regiones de alta frecuencia como los bordes, en regiones de grandes extensiones de una sola tonalidad de color, y debido a que la codificación se la realiza con ruido, este es muy notable en los tres esquemas mostrados, y ahora si se puede apreciar el efecto de este sobre la codificación: MICD que en la sección 5.4.1 pasó limpiamente, y aquí se ve que es muy frágil y muy sensible a medios de almacenamiento o transmisión ruidosos. Se pueden presentar los resultados obtenidos con las tres compresiones:

Tamaño del archivo comprimido:

TDC: 52900 bytes

CTB: 9600 bytes

MICD: 46739 bytes

Tamaño del archivo descomprimido:

TDC: 61440 bytes

CTB: 61440 bytes

MICD: 61440 bytes

Bits/pixel necesarios para presentar a la imagen comprimida:

TDC: 2.315

CTB: 7.333

MICD: 3.876

Error cuadrático medio:

TDC: 0.2325

CTB: 0.5388

MICD: 0.1518

Relación Señal/Ruido(SNR en dB):

TDC: 15.667

CTB: 12.018

MICD: 12.180

Tasa de bits erróneos(BER):

TDC: 0.3015

CTB: 0.3300

MICD: 0.4196

Varianza de la imagen descomprimida:

TDC: 8.991

CTB: 8.572

MICD: 8.578

Media de la imagen descomprimida:

TDC: 7.519

CTB: 7.256

MICD: 6.346

Valor cuadrático medio de la imagen descomprimida:

TDC: 65.071

CTB: 61.222

MICD: 61.322

En este caso de codificación con ruido las expectativas cambian de dirección al ver como se comportaría una transmisión o grabación de una imagen en un ambiente más real (con presencia de ruido).

- La tasa bits/pixel es mejor para la CTB.
- La potencia del error (Error cuadrático medio) es menor según la TDC.
- La calidad de la imagen es mejor con MICD (SNR=3.255dB).
- La cantidad de datos erróneos (BER) es muy baja con MICD.
- Existe un desbalance de energía de pixeles mayor según la MICD (Varianza=164.885>8.57 de la imagen original) y el que lo mantiene es la CTB (Varianza=8.57). Un elevado desbalance produciría una sobresaturación de los circuitos de recepción al recibir dicha imagen.
- Casi no hay variaciones en el promedio de la potencia de los pixeles (Las Medias están alrededor de la media de la imagen original).

De estos dos últimos puntos se puede ver que la energía total (Valor cuadrático medio) con la que la imagen se descomprimirá apunta a un gran desbalance en el caso de la MICD y un equilibrio en el caso de CTB y TDC.

5.4.3 ANALISIS GRAFICO:

A fin de establecer una comparación más didáctica a continuación se realizarán algunas *compresiones-descompresiones* de una imagen de prueba variando el tamaño del bloque de cálculo y así establecer gráficas muy representativas de como se desenvuelve un método de compresión en relación a otro al variar el tamaño del bloque (B). Se presentarán las gráficas siguientes:

- Figura 5.7: B VS. SNR.
- Figura 5.8: B VS. Tamaño del archivo comprimido.
- Figura 5.9: B VS. Bits/pixel.
- Figura 5.10: B VS. BER.
- Figura 5.11: B VS. Error Cuadrático Medio.
- Figura 5.12: B VS. Varianza.
- Figura 5.13: B VS. Valor Cuadrático Medio.
- Figura 5.14: B VS. Media.

La imagen de prueba es la siguiente(TURBO.BMP):

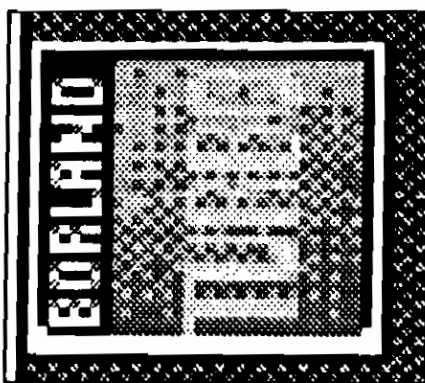


Figura 5.6 TURBO.BMP

Ancho:	32 pixeles
Alto:	32 pixeles
Tamaño del archivo:	1024 bytes
Varianza:	20.039
Media:	7.231
Valor cuadrático medio:	72.330

La ejecución se realiza variando el tamaño del bloque de cálculo desde 2x2 pixeles hasta 32x32 pixeles, una incidencia de ruido del 5% y un orden de cuantificación de 1. Los resultados a obtener son:

Tamaño del archivo comprimido: Así se mide el nivel de compresión de una manera aproximada.

Varianza: Un equivalente AC de la potencia de la imagen.

Media: Un equivalente DC de la potencia de la imagen.

Valor cuadrático medio: Un equivalente de la potencia total de la imagen.

Bits/píxel: El nivel de compresión.

Error cuadrático medio: Potencia del error de codificación.

Relación señal/ruido(dB): Medida de la calidad "analógica" de la imagen.

BER promedio: Medida de la calidad "digital" de la imagen.

De estos resultados se puede concluir que al incrementar el tamaño del bloque ocurren los siguientes efectos:

- El tamaño del archivo comprimido se reduce.
- La potencia AC y la DC se reducen debido a que los valores de píxeles conformadores de cada bloque de la imagen se cuantifican a un solo valor: la media de cada bloque.
- La tasa bits/píxel se reduce bordeando el 1bit/píxel.
- La potencia del error en la codificación aumenta.
- La relación Señal/Ruido disminuye debido a que cada vez el bloque es más grande y existe menos precisión de cálculo.
- El BER promedio aumenta.

Y lo más notable, la calidad visual de la imagen decrece.

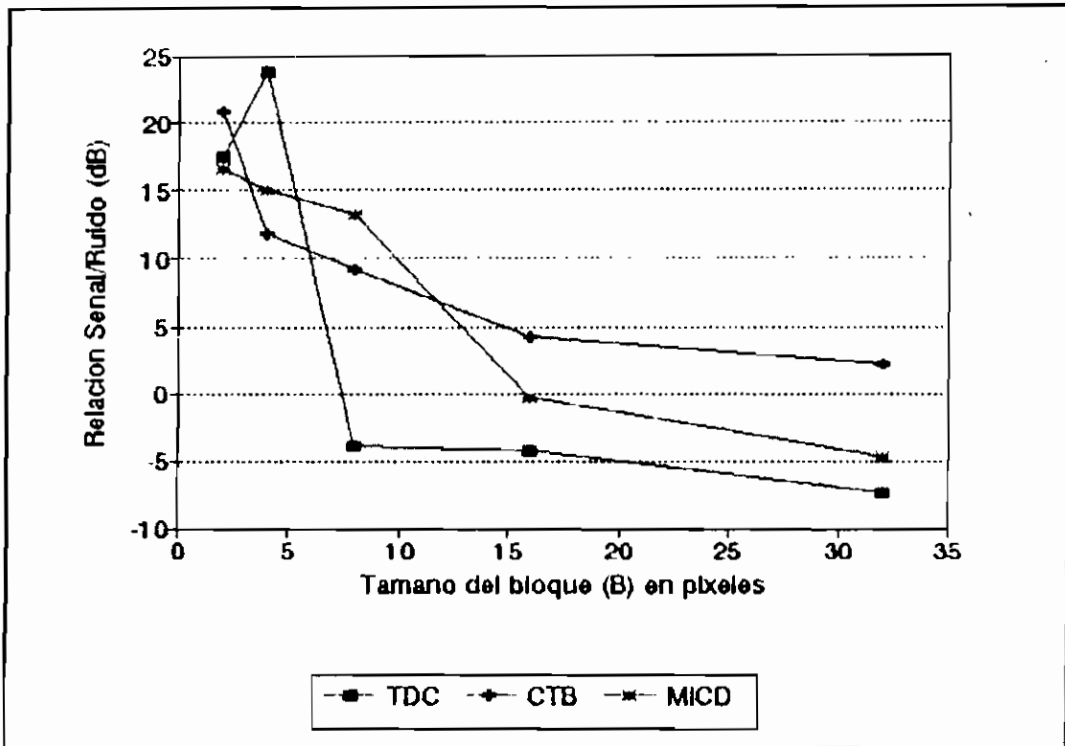


Figura 5.7 Tamaño del bloque vs. SNR.

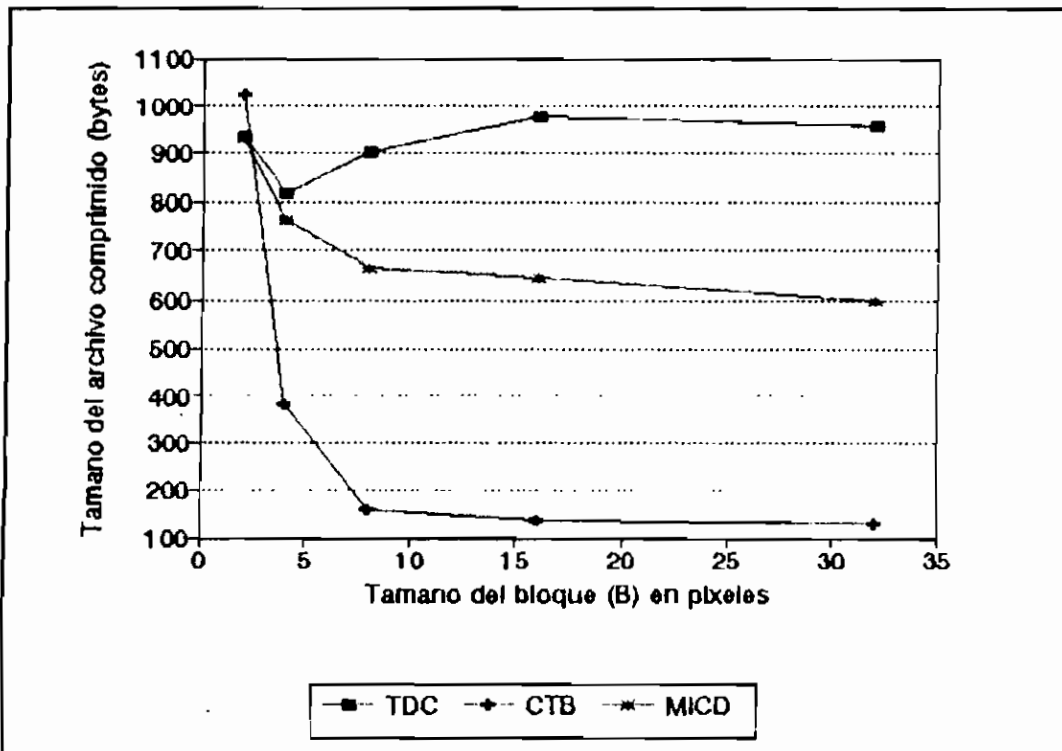


Figura 5.8 Tamaño del bloque vs. Tamaño del archivo.

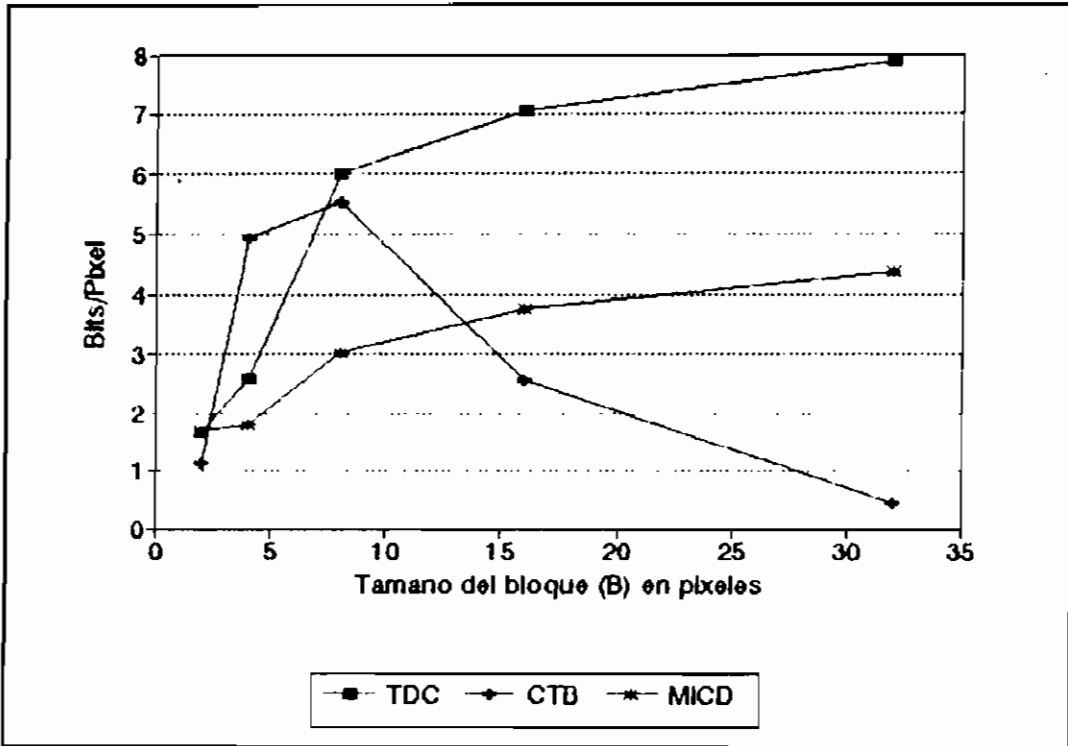


Figura 5.9 Tamaño del bloque vs. bits/pixel.

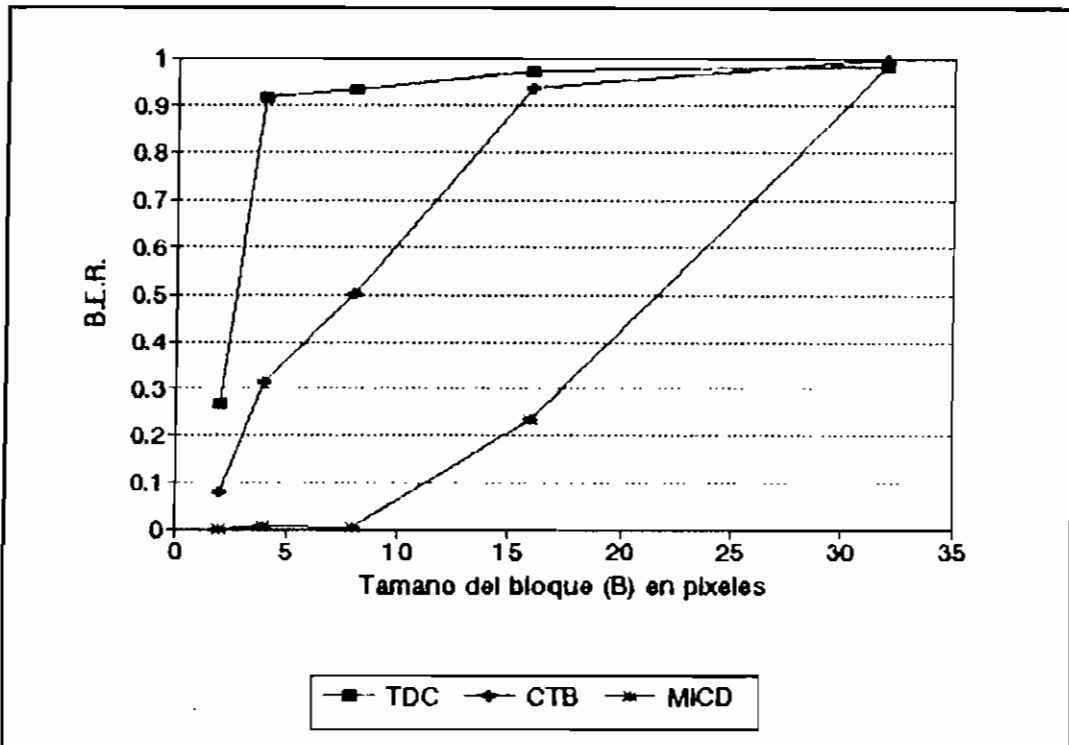


Figura 5.10 Tamaño del bloque vs. B.I.R.

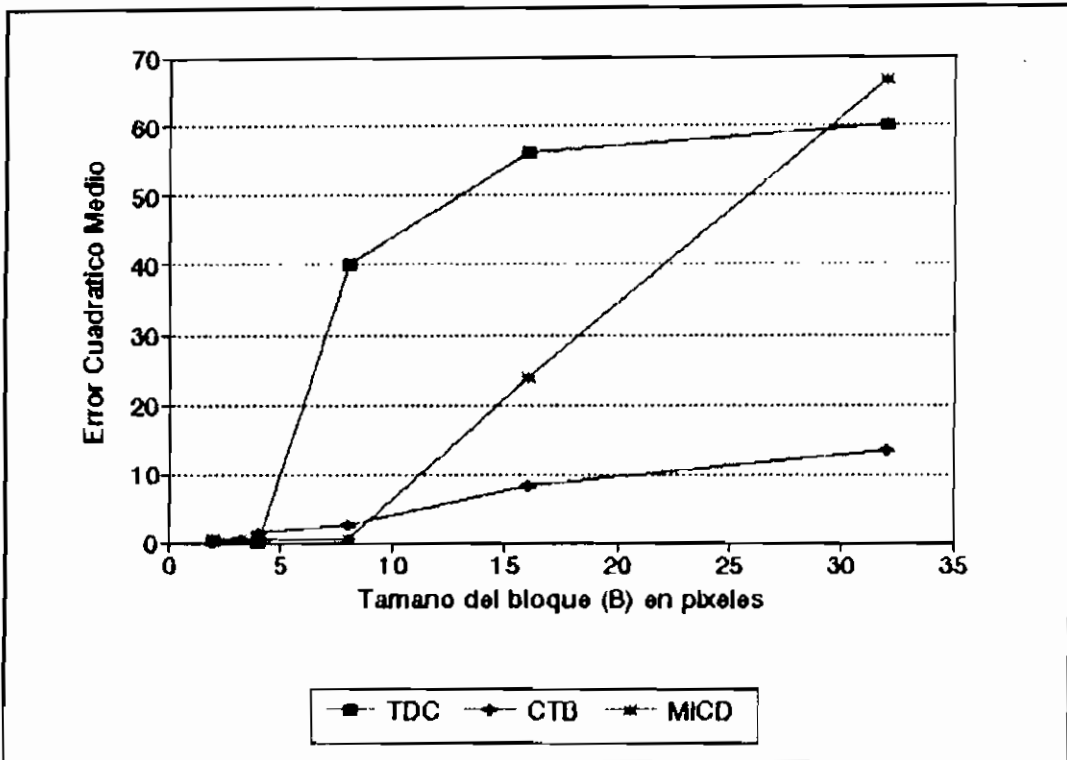


Figura 5.11 Tamaño del bloque vs. ECM.

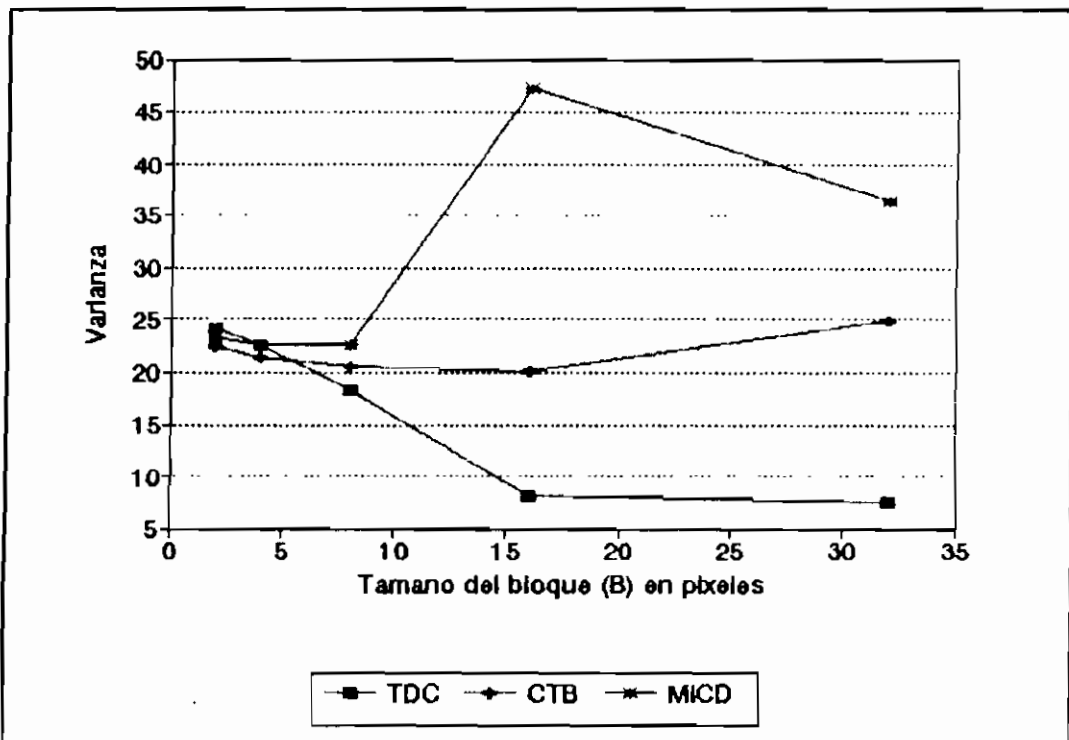


Figura 5.12 Tamaño del bloque vs. Varianza.

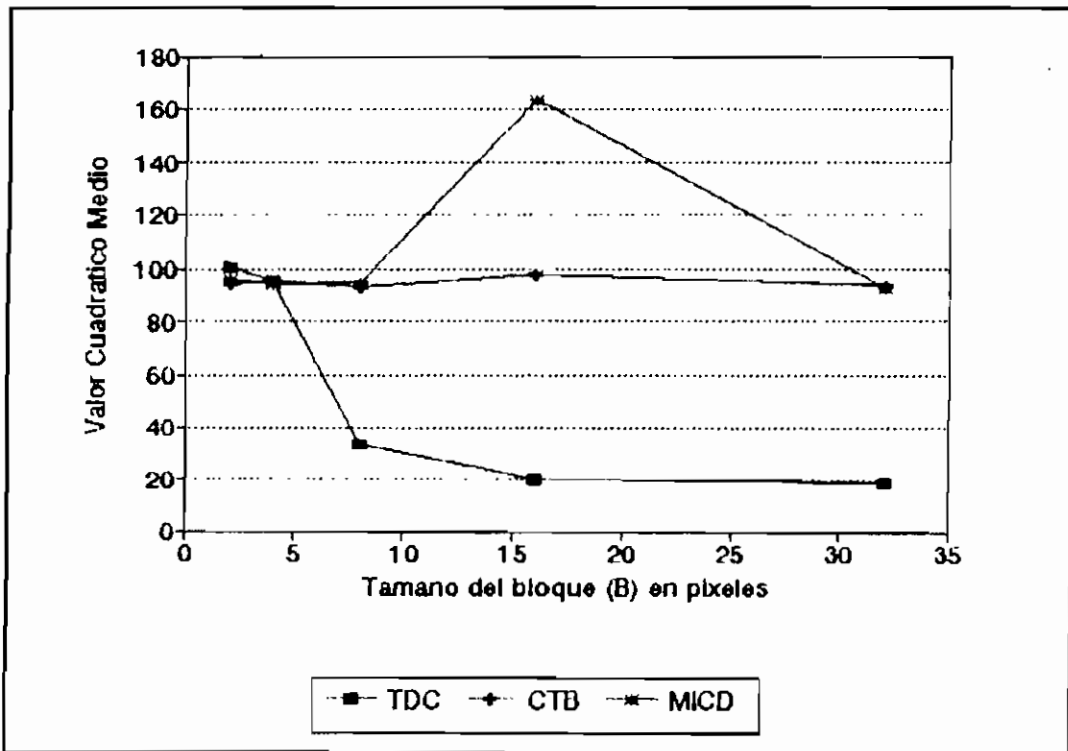


Figura 5.13 Tamaño del bloque vs. VCM.

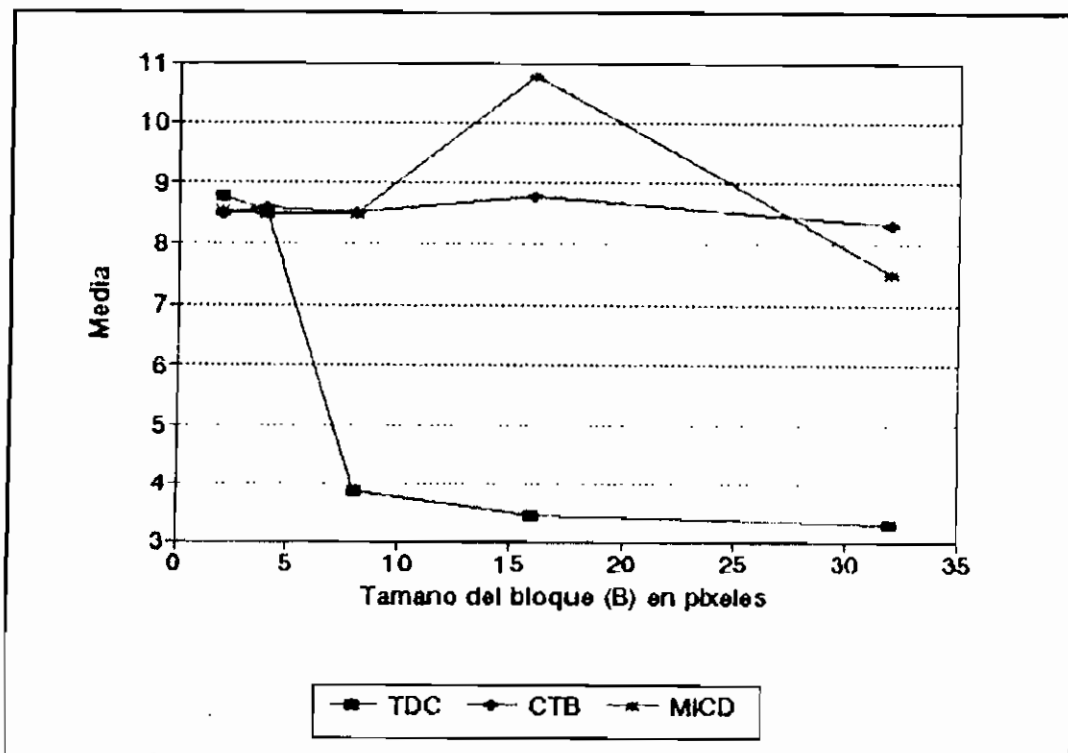
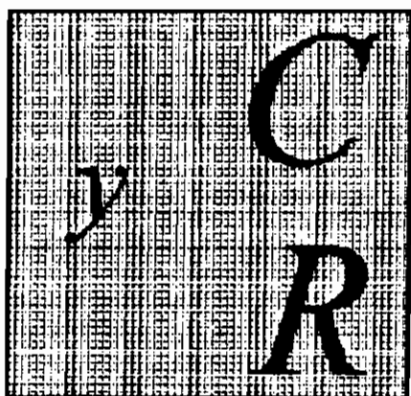


Figura 5.14 Tamaño del bloque vs. Media.



Conclusiones

Recomendaciones

CONCLUSIONES Y RECOMENDACIONES

(*) En este estudio, se han descrito algunas técnicas de reducción de la velocidad binaria (bps) de datos, así como sus principales y más actuales aplicaciones. Se ha analizado que la mejor forma de llegar a un entendimiento entre la velocidad binaria y la calidad de la imagen se logrará solo al seleccionar con mucho cuidado los parámetros de compresión según los requerimientos de calidad de imagen y de la aplicación particular.

(*) Se han dividido los sistemas de compresión de video digital en dos clases: los que requieren rangos relativamente elevados de compresión para satisfacer la capacidad digital limitada de los canales de transmisión y lograr un almacenamiento no costoso y eficiente; y, los que requieren de bajos rangos de compresión para preservar la alta calidad de una imagen para la producción y post-producción de video.

(*) Los métodos de compresión se basan tanto en la redundancia de los datos de la imagen digitalizada, como en la no linealidad de la visión humana. Explotan la correlación en el espacio para imágenes estáticas y la correlación en el espacio y el tiempo para las imágenes en movimiento.

(*) Se ha dividido los sistemas de reducción de redundancia en tres clases: (1) Los que optimizan la longitud de las palabras código (como el Código Huffman), (2) Los que aprovechan el alto grado de correlación entre muestras codificadas (como M1CD), y (3) Los que toman ventaja de la poca sensibilidad visual del ojo humano (como TDC).

(*) La compresión en el **ESPACIO** se conoce como una compresión **INTRA-TRAMA**, mientras que la compresión en el **TIEMPO** se la llama compresión **INTER-TRAMA**.

(*) Generalmente, los métodos que alcanzan altas tasas de compresión (de 10:1 a 50:1 para imágenes estáticas y de 50:1 a 100:1 para video). Son con pérdidas, debido a que los datos reconstruidos no son idénticos al original. Los métodos que

no producen errores existen, pero sus tasas de compresión son muy bajas, quizá no mayores que 3:1. Tales técnicas son utilizadas solo en aplicaciones de alta sensibilidad, como imágenes médicas o archivos legales.

(*) Los algoritmos con pérdidas generalmente explotan los aspectos débiles del sistema visual humano. Por ejemplo, el ojo es mucho más perceptivo a finos detalles en la señal de luminancia (o brillo) que a las señales de crominancia (o color). Consecuentemente, la señal de luminancia se muestrea usualmente a altas resoluciones espaciales (Por ejemplo, en la difusión de televisión de calidad, la resolución digital de la señal de luminancia muestreada es de 720x480 pixeles, mientras que para las señales de color esta puede ser de solamente 360x240 pixeles). Generalmente a la representación codificada (o comprimida) de la señal de luminancia se le asigna más bits (un rango dinámico mayor) que a las señales de crominancia.

(*) La predicción lineal bidimensional remueve mucha de la redundancia de las señales monocromáticas bidimensionales. La predicción lineal de imágenes se puede aplicar satisfactoriamente a la codificación predictiva adaptiva de imágenes. Al utilizar un esquema MCD Adaptivo de la misma complejidad que el utilizado en la codificación predictiva de voz, se pueden obtener imágenes monocromáticas reconstruidas de alta fidelidad a una tasa aproximada de 1bit/pixel.

(*) Hay que ser precavidos en lo que concierne a la valoración de los desempeños de las distintas técnicas de codificación de imágenes a color. Como mínimo, tal valoración requiere de la medida de la tasa de transmisión binario para una calidad dada de imagen. Se debe tener en cuenta que la calidad de la imagen es variable, esta depende de algunos factores, tales como condiciones de iluminación y ajustes de monitor, de la cantidad y tipo de movimiento contenido en la

escena, así como de la experiencia y las aspiraciones de los observadores.

(*) Los variados esquemas de codificación de imágenes de dos tonalidades discutidos han sido estudiados por distintos investigadores en épocas distintas. Estos, no han sido aplicados a un mismo arreglo de imágenes. Como consecuencia, es muy difícil comparar sus eficiencias de una manera definitiva. Afortunadamente, es mucho menos importante comparar a los varios esquemas de codificación en detalle que tener una visión general de ellos.

(*) De todas las discusiones y estudios en el campo de la codificación de imágenes a dos tonalidades, se podría dar una visión general como la siguiente: Para una baja resolución (alrededor de 100ppp), los esquemas uni-dimensionales de codificación son usualmente preferidos (debido a su facilidad de implementación y porque ellos arrojan factores de compresión comparables a los esquemas bi-dimensionales), a altas resoluciones (cerca de 200ppp); los esquemas bi-dimensionales de codificación pueden dar factores considerablemente altos de compresión y se los prefiere a pesar de su complejidad de implementación.

(*) La ventajas de una codificación predictiva adaptiva inter-campo intra-campo se mostró como aplicación práctica en la transmisión-recepción de señal NTSC a color a 30Mbps, con un codec basado en este principio. Esta predicción adaptiva da una buena y estable eficiencia de predicción para imágenes desde las estáticas a las de movimientos violentos.

(*) Aunque la TDCC y la TDH superan a la TDF, la TDC supera consistentemente a la TDCC y la TDH. Para imágenes que exhiben una gran cantidad de áreas de intensidades constantes de pixeles, los coeficientes TDC se mejoran si se modelan con una distribución gamma, mientras que para imágenes caracteri-

zadas con muchos bordes y texturas irregulares, los coeficientes se modelan mejor con una distribución Laplaciana.

(*) La codificación híbrida es una combinación de la sencillez de construcción (Hardware) de la técnica MICD y la robustez y alta eficiencia de la codificación por transformada. Se adapta fácilmente a las imágenes ruidosas y a los cambios estadísticos de las imágenes. La codificación híbrida es menos sensible a los errores de canal que la MICD pero no es tan eficiente como la codificación por transformada.

(*) La técnica CTB es una sencilla técnica que se ha descrito y que se puede modificar para mejorar su desempeño (CTB híbrida, CTB con un mayor número de momentos). Las imágenes reconstruidas resultantes tienen artefactos muy distintos de los producidos por otras técnicas, pero las imágenes resultantes son comparables y de alguna forma superiores a aquellas producidas por técnicas más sofisticadas. Este método produce una imagen codificada que es más robusta en la presencia de errores en el canal y así se requiere muy poca protección contra errores.

(*) La codificación por truncamiento de bloque se ha mostrado como una técnica simple y efectiva para la compresión de video digital. El cuantificador a 1bit inherente tiende a realzar los bordes por medio de la preservación de las variaciones de intensidad dominante a expensas de cambios secundarios que son imperceptibles para el sistema visual humano.

(*) El compromiso entre reducción de datos y reducción de ancho de banda se establece cuando la imagen procesada pierde sus características visuales útiles (cuando una imagen se vuelve menos distinguible) y cuando la capacidad del canal ya está limitada.

(*) La gran cantidad de programas informáticos de gráficos existentes en la actualidad, hacen necesaria una estandarización de sus formatos de forma que un archivo generado en un paquete computacional pueda ser utilizado en otro. De la gran cantidad de formatos para archivos gráficos existentes, los más populares son, posiblemente, PCX, TIFF y BMP.

(*) La estructura de un archivo PCX es fija, en el sentido que las características de la imagen están ya definidas. Esto es una ventaja para escribir programas de manejo de archivos PCX, pero es también un inconveniente ya que no se puede acomodar la estructura a los nuevos tipos de tarjetas gráficas existentes. Un ejemplo se da en los colores; el formato original fue diseñado para 16 colores que se incluyen en la cabecera, pero cuando apareció la tarjeta VGA con 256 colores fue necesario aumentar una nueva paleta al final del archivo para recoger la información correspondiente a estos 256 colores.

(*) En cuestiones de programación para el campo de la simulación o la implementación circuital, es recomendable el trabajar con los procesos de cálculo en forma de bloques. Es decir que si lo que se quiere es construir un circuito impreso que codifique y decodifique imágenes por las técnicas estudiadas (o algunas otras más), se acelera el proceso si el tratamiento se lo hace por secciones de imagen (bloques), mucho mejor si estos son cuadrados ($n \times n$ pixeles).

(*) Con respecto a la ejecución del programa de compresión de imágenes con las tres técnicas (TDC, CTB y MICD) es necesario notar que el tiempo de procesamiento es dependiente de los siguientes factores:

- El tamaño del bloque.
- Los niveles de color de la imagen.
- La incidencia de ruido.
- La técnica a utilizar.

Con estos factores se concluye que una imagen se comprimirá más rápidamente (que es lo que se busca en aplicaciones prácticas) si el tamaño del bloque es pequeño al usar TDC, aunque el número de bits finales aumente (mayor precisión) y por tanto baja la eficiencia; o si el tamaño del bloque es grande al utilizar CTB y en este caso se producen mayores pérdidas aunque la eficiencia de la compresión es buena. Por otro lado si el tamaño del bloque es pequeño y se utiliza CTB, la ejecución es lenta, el número de bits significativos es demasiado grande (mayor precisión) y la eficiencia de compresión es por tanto muy baja aunque la imagen esté libre de pérdidas. Si el tamaño del bloque es grande y se utiliza TDC la ejecución es bastante lenta, se alcanzan altos niveles de compresión, y la eficiencia por tanto es muy buena a pesar de que la imagen tiene mayores pérdidas al realizar su descompresión. Con respecto a MICD su ejecución es prácticamente independiente del tamaño del bloque al no existir ruido, pero la incidencia de este en la codificación es un demérito muy grande de esta técnica y así es mejor tratar a la compresión en bloques muy pequeños minimizando la incidencia de perturbaciones en razón directa a la dimensión de aquellos. Se concluye entonces que las mejores técnicas (en cuanto a altas tasas de compresión, bajo tiempo de ejecución y buena calidad visual en la imagen descomprimida) son en orden decreciente:

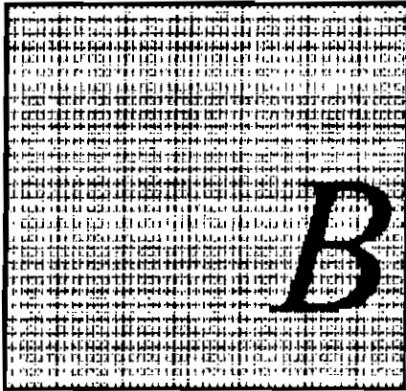
- Sin incidencia de ruido:
 - Bloques pequeños: MICD, TDC, CTB
 - Bloques grandes: MICD, CTB, TDC
- Con incidencia de ruido:
 - Bloques pequeños: MICD, TDC, CTB
 - Bloques grandes: TDC, CTB, MICD

(*) El programa computacional está diseñado para ejecutarse con imágenes estáticas como fotografías y recuadros gráficos. Una implementación mayor para aplicar los mismos algoritmos a imágenes móviles puede realizarse sencillamente si se lograra reducir el tiempo de procesamiento de por ejemplo los "cose-

nos" de TDC. Debido a que el programa está implementado a nivel de bloques de trabajo y subprocesos, la implementación a imágenes de video NTSC por ejemplo necesitaría que un cuadro de imagen se "procese" 30 veces por segundo, lo cual es posible si las rutinas matemáticas se canalizan a un coprocesador matemático como el 80287.

El llamado "proceso" antes indicado constaría de las siguientes partes:

- Barrido de la imagen.
- Seccionamiento en bloques.
- Cálculos (por ejemplo la TDC por cada bloque).
- Grabación o Transmisión.



Bibliografia

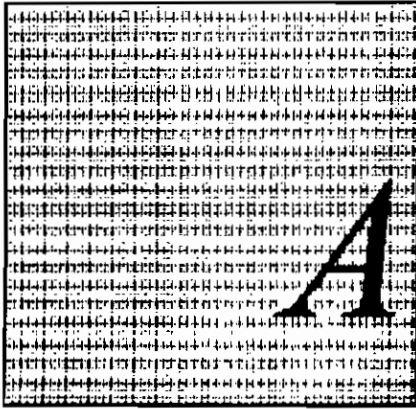
BIBLIOGRAFIA

- [1] COMPUTER CRAFT, "*Image File Formats*", Abril 1993.
- [2] AUBREY HARRIS, "*The New World of HDTV*", Electronics Now, Mayo 1993.
- [3] ANG H. PENG, RUETZ PETER A., & AULD DAVID, LSI LOGIC CORP., "*Video compression makes big gains*", IEEE Spectrum, Octubre 1991.
- [4] ESPECIAL REPORT/CONSUMER ELECTRONICS, "*An abundance of video formats*", IEEE Spectrum, Marzo 1992.
- [5] GRISWOLD N.C., HALVERSON D.R., & WISE G.L., "*A Note on Adaptive Block Truncation Coding for Image Processing*", IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP.35, N°8, Agosto 1987.
- [6] DELP EDWARD J., & MITCHELL O. ROBERT, "*Image Compression Using Block Truncation Coding*", IEEE Transactions on Communications, Vol. COM-27, N°9, Septiembre 1979.
- [7] HEALY DONALD J., & MITCHELL O. ROBERT, "*Digital Video Bandwidth Compression Using Block Truncation Coding*", IEEE Transactions on Communications, Vol. COM-29, N°12, Diciembre 1981.
- [8] PERKINGS MICHAEL G., "*A Comparison of the Hartley, Cos-Cas, Fourier, and Discrete Cosine Transforms for Image Coding*", IEEE Transactions on Communications, Vol.36, N°6, Junio 1988.
- [9] MARAGOS PETROS A., SHAFER RONALD W., MERSKREAU RUSSEL M., "*Two-Dimensional Linear Prediction and Its Application to Adaptive Predictive Coding of Images*", IEEE Transactions on Acoustics and Signal Processing, Vol. ASSP-32, N°6, Diciembre 1984.
- [10] WERNESS SUSAN A., "*Statistical Evaluation of Predictive Data Compression Systems*", IEEE Transactions on Acoustics and Signal Processing, Vol. ASSP-35, N°8, Agosto 1987.
- [11] LIMB JOHN O., RUBINSTEIN CHARLES B., & THOMPSON JOHN E., "*Digital Coding of Color Video Signals-A Review*", IEEE Transactions on Communications, Vol. COM-25, N°11, Noviembre 1977.

- [12] HUANG THOMAS S., "Coding of Two-Tone Images", IEEE Transactions on Communications, Vol. COM-25, N°11, Noviembre 1977.
- [13] PRABHU K.A., "A Predictor Switching Scheme for DPCM Coding of Video Signals", IEEE Transactions on Communications, Vol.COM-33, N°4, Abril 1985.
- [14] KANEFSKY MORTON & FONG CHUNG-BIN, "Predictive Source Coding Techniques Using Maximum Likelihood Prediction for Compression of Digitized Images", IEEE Transactions on Information Theory, Vol. IT-30, N°5, Septiembre 1984.
- [15] GOLDBERG MORRIS, BOUCHER PAUL R., & SHLIEN SEYMOUR, "Image Compression Using Adaptive Vector Quantization", IEEE Transactions on Communications, Vol. COM-34, N°2, Febrero 1986.
- [16] BERTOCCI GUIDO, SCHOENHERR BRIAN W., & MESSERSCHMITT DAVID G., "An Approach to the Implementation of a Discrete Cosine Transform", IEEE Transactions on Communications, Vol. COM-30, N°4, Abril 1982.
- [17] YAMAMOTO HIDEO, HATORI YOSHINORI, & MURAKAMI HITOMI, "30Mbit/s Codec for the NTSC Color TV Signal Using an Interfield-Intrafield Adaptive Prediction", IEEE Transactions on Communications, Vol. COM-29, N°12, Diciembre 1981.
- [18] MURAKAMI HITOMI, HATORI YOSHINORI, & YAMAMOTO HIDEO, "Comparison Between DPCM and Hadamard Transform Coding in the Composite Coding of the NTSC Color TV Signal", IEEE Transactions on Communications, Vol. COM-30, N°3, Marzo 1982.
- [19] CHOUDHARY ALOCK & RANKA SANJAY, "Parallel Processing for Computer Vision and Image Understanding", IEEE COMPUTER SOCIETY, "COMPUTER", Febrero 1991.
- [20] HOPKINS ROBERT, "Advanced Television Systems", IEEE Transactions on Consumer Electronics, Vol. 34, N°1, Febrero 1988.
- [21] CAGGIONI HUGO & LE GALL DIER. Bell Communications Research, "Digital Video Transmission and Coding for the Broadband ISDN", IEEE Transactions on Consumer Electronics, Vol. 34, N°1, Febrero 1988.

- [22] YAMAMITSU C., IDE A., & JURI T. Central Research Laboratory, "An Experimental Digital VTR Capable of 12-Hour Recording", IEEE Transactions on Consumer Electronics, Vol. CE-33, N°3, Agosto 1987.
- [23] SONODA SHOJI, IMAHORI YUTAKA. Engineering Headquarters, NHK(Japan Broadcasting Corporation), "An Overview of Current DBS Situation in Japan", IEEE Transactions on Consumer Electronics, Vol. CE-33, N°3, Agosto 1987.
- [24] HABIBI ALI, "Special Issue on Image Bandwidth Compression", IEEE Transactions on Communications, Vol. COM-25, N°11, Noviembre 1977.
- [25] SAKRISON DAVID J., "On the Role of the Observer and a Distorsion Measure in Image Transmission", IEEE Transactions on Communications, Vol. COM-25, N°11, Noviembre 1977.
- [26] AHMED N., NATARAJAN T., & RAO K., "Discrete Cosine Transform", IEEE Transactions on Computers, Vol. C-23, Enero 1974.
- [27] ANIL JAIN K., "Fundamentals of Digital Image Processing", Prentice-Hall, 1989.
- [28] HOSSAIN ZAHID, "Digital Image Processing: Practical Applications of Parallel Processing Techniques", Ellis Horwood, 1991.
- [29] BELLAMY JOHN, "Digital Telephony", John Wiley&Sons, 1982.
- [30] HERNÁNDEZ J., GUTIERREZ J., "Video Avanzado 1", Sony Corp., 1988.
- [31] GUTIERREZ J., SÁENZ J., "Video Avanzado 3, Cámaras VIDEO-8", Sony Corp., 1991.
- [32] BROW JOHN, "Curso Básico de T.V.", Glem S.A., 1979.
- [33] ALVEAR ALFREDO, "Introducción al procesamiento digital de imágenes", Escuela Politécnica del Ejército, TELIS de Ingeniería Electrónica, Marzo 1995.
- [34] SAKANE TOSHINORI Ph.D., "Codificación digital de señales de voz y video", Escuela Politécnica del Ejército, Vicerrectorado de Investigación y Desarrollo, Enero 1995.

- [35] EGAS CARLOS ING., Folleto de "*Teoría de la Información y la Codificación*", Escuela Politécnica Nacional.
- [36] TERRADILLOS G. FERNANDO, "*El arte de reducir*", PC-WORLD Ecuador, Año XV, N° 124, Ed. ECUASISTEM, Abril 1993.
- [37] GATICA M. JORGE, "*Procesamiento Computacional de Audio y Video*", PC-WORLD Ecuador, Año XV, N° 124, Ed. ECUASISTEM, Abril 1993.
- [38] CCIR, Recomendación 601.
- [39] CAGGIONI HUGO, "*Video Compression*", SONY Corporation of America, 1994.
- [40] GONZALEZ C. R., WOODS R., "*Digital Image Processing*", Addison-Wesley Publishing Co., 1992.
- [41] ING. PABLO HIDALGO, Folleto de "*Comunicación Digital*", Escuela Politécnica Nacional.
- [42] MURRAY SPIEGEL, "*Probabilidad y Estadística*", Colección de libros Schaum, Mgraw-Hill, 1981.



Anexos

ANEXO-2

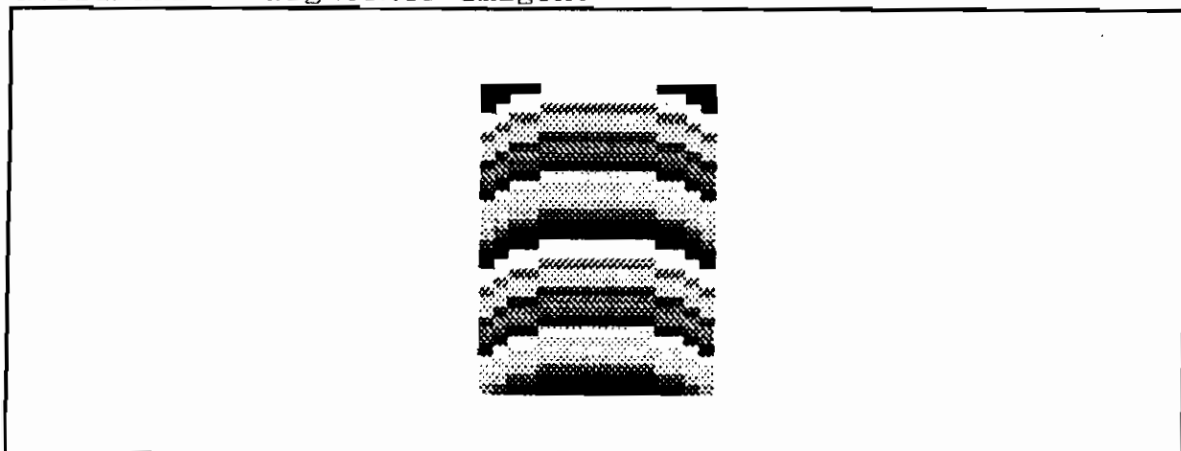
EJEMPLOS DE APLICACION

- (a) MIC Ley A.
- (b) Huffman.
- (c) MIC-D.
- (d) MD.
- (e) Transformada de KL.
- (f) Transformada de Fourier.
- (g) Transformada de Walsh-Hadamard.
- (h) Transformada de Haar.
- (i) Transformada Discreta Coseno.
- (j) OBB Upidimensional.
- (k) Truncamiento de Bloque.
- (l) Formatos PCX, y BMP.

ANEXO II--(a)

CODIFICACION MIC Ley A

Para la ejecución de este tipo de codificación se utilizará la siguiente imagen:

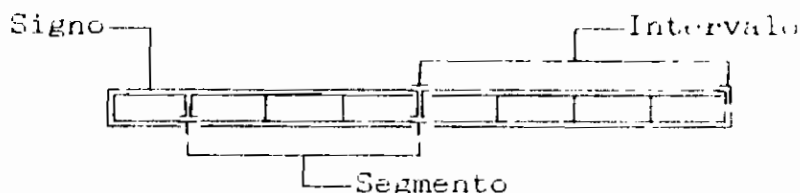


Cuya representación numérica es:

0	0	0	0	15	15	15	15	15	15	15	15	0	0	0	0
0	0	15	15	14	14	14	14	14	14	14	14	15	15	0	0
0	15	14	14	7	7	7	7	7	7	7	7	14	14	15	0
15	14	7	7	11	11	11	11	11	11	11	11	7	7	14	15
14	7	11	11	10	10	10	10	10	10	10	10	11	11	7	14
7	11	10	10	3	3	3	3	3	3	3	3	10	10	11	7
11	10	3	3	8	8	8	8	8	8	8	8	3	3	10	11
10	3	8	8	5	5	5	5	5	5	5	5	8	8	3	10
3	8	5	5	2	2	2	2	2	2	2	2	5	5	8	3
8	5	2	2	13	13	13	13	13	13	13	13	2	2	5	8
5	2	13	13	12	12	12	12	12	12	12	12	13	13	2	5
2	13	12	12	6	6	6	6	6	6	6	6	12	12	13	2
13	12	6	6	9	9	9	9	9	9	9	9	6	6	12	13
12	6	9	9	4	4	4	4	4	4	4	4	9	9	6	12
6	9	4	4	1	1	1	1	1	1	1	1	4	4	9	6
9	4	1	1	0	0	0	0	0	0	0	0	1	1	4	9
4	1	0	0	15	15	15	15	15	15	15	15	0	0	1	4
1	0	15	15	14	14	14	14	14	14	14	14	15	15	0	1
0	15	14	14	7	7	7	7	7	7	7	7	14	14	15	0
15	14	7	7	11	11	11	11	11	11	11	11	7	7	14	15
14	7	11	11	10	10	10	10	10	10	10	10	11	11	7	14
7	11	10	10	3	3	3	3	3	3	3	3	10	10	11	7
11	10	3	3	8	8	8	8	8	8	8	8	3	3	10	11
10	3	8	8	5	5	5	5	5	5	5	5	8	8	3	10
3	8	5	5	2	2	2	2	2	2	2	2	5	5	8	3
8	5	2	2	13	13	13	13	13	13	13	13	2	2	5	8
5	2	13	13	12	12	12	12	12	12	12	12	13	13	2	5
2	13	12	12	6	6	6	6	6	6	6	6	12	12	13	2
13	12	6	6	9	9	9	9	9	9	9	9	6	6	12	13
12	6	9	9	4	4	4	4	4	4	4	4	9	9	6	12
6	9	4	4	1	1	1	1	1	1	1	1	4	4	9	6
9	4	1	1	0	0	0	0	0	0	0	0	1	1	4	9
4	1	0	0	15	15	15	15	15	15	15	15	0	0	1	4
1	0	15	15	14	14	14	14	14	14	14	14	15	15	0	1
0	15	14	14	7	7	7	7	7	7	7	7	14	14	15	0
15	14	7	7	11	11	11	11	11	11	11	11	7	7	14	15
14	7	11	11	10	10	10	10	10	10	10	10	11	11	7	14
7	11	10	10	3	3	3	3	3	3	3	3	10	10	11	7
11	10	3	3	8	8	8	8	8	8	8	8	3	3	10	11
10	3	8	8	5	5	5	5	5	5	5	5	8	8	3	10
3	8	5	5	2	2	2	2	2	2	2	2	5	5	8	3
8	5	2	2	13	13	13	13	13	13	13	13	2	2	5	8
5	2	13	13	12	12	12	12	12	12	12	12	13	13	2	5
2	13	12	12	6	6	6	6	6	6	6	6	12	12	13	2
13	12	6	6	9	9	9	9	9	9	9	9	6	6	12	13
12	6	9	9	4	4	4	4	4	4	4	4	9	9	6	12
6	9	4	4	1	1	1	1	1	1	1	1	4	4	9	6
9	4	1	1	0	0	0	0	0	0	0	0	1	1	4	9

El objetivo es pasar de la muestra digitalizada a 8 bits (que se obtiene de algún conversor A/D o que representan los valores de los pixeles de la imagen), a una palabra código especialmente codificada (Ley A).

Como ya se vió en el Capítulo I el formato MIC de una muestra digital debe tener la forma:



Donde: Signo=1/0 para signo -/+ de la muestra.

Segmento=de 000 a 111, 8 segmentos.

Intervalo=de 0000 a 1111, 16 intervalos.

Y la ley A de codificación se expresa según las ecuaciones (1.8) y (1.9) de compresión:

$$F_A = \operatorname{sgn}(x) \left(\frac{A|x|}{1+\ln(A)} \right) \quad 0 \leq |x| \leq \frac{1}{A} \quad (1.8)$$

$$F_A = \operatorname{sgn}(x) \left(\frac{1+\ln|Ax|}{1+\ln(A)} \right) \quad \frac{1}{A} \leq |x| \leq 1 \quad (1.9)$$

Y la característica inversa o de expansión (1.10) y (1.11), con $y = F_A(x)$:

$$F_A^{-1} = \operatorname{sgn}(y) \frac{|y|[1+\ln A]}{A}; \quad 0 \leq |y| \leq \frac{1}{1+\ln(A)} \quad (1.10)$$

$$F_A^{-1} = \operatorname{sgn}(y) \frac{(e^{|y|(1+\ln A)} - 1)}{A}; \quad \frac{1}{1+\ln(A)} \leq |y| \leq 1 \quad (1.11)$$

Se trabaja con muestras positivas (Signo=0), pues la imagen se representa con 16 niveles de gris (0=negro a 15=blanco). Como inicio se toman las 16 últimas muestras para indicar como trabaja el método, luego representaremos a la imagen total codificada por MIC.

Valor máximo luego de un rastreo: 0Fh = 15d

Muestra relativa: Muestra/Valor máximo

A = 37.6

1/A = 0.01142

Valor de salida máximo: 80h=12ed

Valor MIC: y*128

Muestra	Muestra relativa	Valor MIC	Palabra MIC
9	0.6000	116.05	01110011b = 73h
4	0.2667	97.09	01100000b = 60h
1	0.0666	64.67	01000000b = 40h
1	0.0666	64.67	01000000b = 40h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
0	0.0000	0.00	00000000b = 00h
1	0.0666	64.67	01000000b = 40h
1	0.0666	64.67	01000000b = 40h
4	0.2667	97.09	01100000b = 60h
9	0.6000	116.05	01110011b = 73h

Tal vez con este ejemplo no se visualiza la cantidad de compresión alcanzada, esto es porque el valor máximo de la muestra es de 15 (baja resolución), en el caso de ser 256 como en las imágenes de alta resolución los primeros seis valores se mostrarían como:

Muestra	Muestra relativa	Valor MIC	Palabra MIC
9	0.0352	49.73	00110001b = 31h
4	0.0156	30.69	00011110b = 1Eh
1	0.0039	7.99	00000111b = 07h
1	0.0039	7.99	00000111b = 07h
0	0.0000	0.00	00000000b = 00h

La compresión se presenta pues si para representar a la muestra de valor 01d se requerían 8bits originalmente:

$$01d = 0000\ 0001b$$

luego de la aplicación de la ley A efectivamente solo se necesitarían 7bits más la polaridad que se considera positiva, por lo que esto deja un total de 7bits netos:

$$01d = 000\ 0111b(MIC)$$

como el valor máximo es 256 y su representación sería:

$$256d = 111\ 1111b(MIC)$$

lo que más se necesitarían son 7bits netos de información, así por medio de MIC se reduce la tasa de bits necesarios para codificación y con un menor uso de potencia pues la cuantificación es no-uniforme.

ANEXO II-(b)

CODIFICACION HUFFMAN

Para la ejecución de este tipo de codificación se utilizará la misma imagen del Anexo II-(a) de la cual ya se tienen sus valores numéricos de los que se deduce:

Número de apariciones de 15	=	36
Número de apariciones de 14	=	32
Número de apariciones de 13	=	32
Número de apariciones de 12	=	32
Número de apariciones de 11	=	32
Número de apariciones de 10	=	32
Número de apariciones de 9	=	32
Número de apariciones de 8	=	32
Número de apariciones de 7	=	32
Número de apariciones de 6	=	32
Número de apariciones de 5	=	32
Número de apariciones de 4	=	30
Número de apariciones de 3	=	32
Número de apariciones de 2	=	32
Número de apariciones de 1	=	28
Número de apariciones de 0	=	34

Número total de muestras = 512

Lo que indica que cada muestra (símbolo) tiene una probabilidad de ocurrencia (que se obtiene al dividir el número de apariciones de la muestra para el total de muestras) de:

$$P(15)=0.07031$$

$$P(14)=P(13)=P(12)=P(11)=P(10)=P(9)=P(8)=P(7)=P(6)=P(5)=0.0625$$

$$P(4)=0.05859$$

$$P(3)=P(2)=0.0625$$

$$P(1)=0.05469$$

$$P(0)=0.06641$$

Se ordenan de mayor a menor sus probabilidades:

$$P(15) = 0.07031$$

$$P(0) = 0.06641$$

$$P(14) = 0.0625$$

$$P(13) = 0.0625$$

$$P(12) = 0.0625$$

$$P(11) = 0.0625$$

$$P(10) = 0.0625$$

$$P(9) = 0.0625$$

$$P(8) = 0.0625$$

$$P(7) = 0.0625$$

$$P(6) = 0.0625$$

$$P(5) = 0.0625$$

$$P(3) = 0.0625$$

$$P(2) = 0.0625$$

$$P(4) = 0.05859$$

$$P(1) = 0.05469$$

Entropía(Ecuación 2.2):

$$H = 1.203588 \text{ Ui/símbolo}$$

donde, U_i = unidad de información

Se procede a sumar las dos últimas probabilidades:

$$\begin{aligned} P(4) &= 0.05859 \\ P(1) &= 0.05469 \end{aligned}$$

$$P(41) = 0.11328$$

Se reordena:

$$\begin{aligned} P(41) &= 0.11328 \\ P(15) &= 0.07031 \\ P(0) &= 0.06641 \\ P(14) &= 0.0625 \\ P(13) &= 0.0625 \\ P(12) &= 0.0625 \\ P(11) &= 0.0625 \\ P(10) &= 0.0625 \\ P(9) &= 0.0625 \\ P(8) &= 0.0625 \\ P(7) &= 0.0625 \\ P(6) &= 0.0625 \\ P(5) &= 0.0625 \\ P(3) &= 0.0625 \\ P(2) &= 0.0625 \end{aligned}$$

Se procede a sumar las dos últimas probabilidades:

$$\begin{aligned} P(3) &= 0.0625 \\ P(2) &= 0.0625 \end{aligned}$$

$$P(32) = 0.1250$$

Se reordena:

$$\begin{aligned} P(32) &= 0.1250 \\ P(41) &= 0.11328 \\ P(15) &= 0.07031 \\ P(0) &= 0.06641 \\ P(14) &= 0.0625 \\ P(13) &= 0.0625 \\ P(12) &= 0.0625 \\ P(11) &= 0.0625 \\ P(10) &= 0.0625 \\ P(9) &= 0.0625 \\ P(8) &= 0.0625 \\ P(7) &= 0.0625 \\ P(6) &= 0.0625 \\ P(5) &= 0.0625 \end{aligned}$$

Se procede a sumar las dos últimas probabilidades:

$$\begin{aligned} P(6) &= 0.0625 \\ P(5) &= 0.0625 \end{aligned}$$

$$P(65) = 0.1250$$

Se reordena:

P(32) = 0.1250
P(65) = 0.1250
P(41) = 0.11328
P(15) = 0.07031
P(0) = 0.06641
P(14) = 0.0625
P(13) = 0.0625
P(12) = 0.0625
P(11) = 0.0625
P(10) = 0.0625
P(9) = 0.0625
P(8) = 0.0625
P(7) = 0.0625

Se procede a sumar las dos últimas probabilidades:

P(8) = 0.0625
P(7) = 0.0625

P(87) = 0.1250

Se reordena:

P(32) = 0.1250
P(65) = 0.1250
P(87) = 0.1250
P(41) = 0.11328
P(15) = 0.07031
P(0) = 0.06641
P(14) = 0.0625
P(13) = 0.0625
P(12) = 0.0625
P(11) = 0.0625
P(10) = 0.0625
P(9) = 0.0625

Se procede a sumar las dos últimas probabilidades:

P(10) = 0.0625
P(9) = 0.0625

P(109) = 0.1250

Se reordena:

P(32) = 0.1250
P(65) = 0.1250
P(87) = 0.1250
P(109) = 0.1250
P(41) = 0.11328
P(15) = 0.07031
P(0) = 0.06641
P(14) = 0.0625
P(13) = 0.0625
P(12) = 0.0625
P(11) = 0.0625

Se procede a sumar las dos últimas probabilidades:

$$P(12) = 0.0625$$

$$P(11) = 0.0625$$

$$P(1211) = 0.1250$$

Se reordena:

$$P(32) = 0.1250$$

$$P(65) = 0.1250$$

$$P(87) = 0.1250$$

$$P(109) = 0.1250$$

$$P(1211) = 0.1250$$

$$P(41) = 0.11328$$

$$P(15) = 0.07031$$

$$P(0) = 0.06641$$

$$P(14) = 0.0625$$

$$P(13) = 0.0625$$

Se procede a sumar las dos últimas probabilidades:

$$P(14) = 0.0625$$

$$P(13) = 0.0625$$

$$P(1413) = 0.1250$$

Se reordena:

$$P(32) = 0.1250$$

$$P(65) = 0.1250$$

$$P(87) = 0.1250$$

$$P(109) = 0.1250$$

$$P(1211) = 0.1250$$

$$P(1413) = 0.1250$$

$$P(41) = 0.11328$$

$$P(15) = 0.07031$$

$$P(0) = 0.06641$$

Se procede a sumar las dos últimas probabilidades:

$$P(15) = 0.07031$$

$$P(0) = 0.06641$$

$$P(150) = 0.13672$$

Se reordena:

$$P(150) = 0.13672$$

$$P(32) = 0.1250$$

$$P(65) = 0.1250$$

$$P(87) = 0.1250$$

$$P(109) = 0.1250$$

$$P(1211) = 0.1250$$

$$P(1413) = 0.1250$$

$$P(41) = 0.11328$$

Se procede a sumar las dos últimas probabilidades:

$$\begin{array}{r} P(1413) = 0.1250 \\ P(41) = 0.11328 \\ \hline \end{array}$$

$$P(141341) = 0.23828$$

Se reordena:

$$\begin{array}{r} P(141341) = 0.23828 \\ P(150) = 0.13672 \\ P(32) = 0.1250 \\ P(65) = 0.1250 \\ P(87) = 0.1250 \\ P(109) = 0.1250 \\ P(1211) = 0.1250 \end{array}$$

Se procede a sumar las dos últimas probabilidades:

$$\begin{array}{r} P(109) = 0.1250 \\ P(1211) = 0.1250 \\ \hline \end{array}$$

$$P(1091211) = 0.25$$

Se reordena:

$$\begin{array}{r} P(1091211) = 0.25 \\ P(141341) = 0.23828 \\ P(150) = 0.13672 \\ P(32) = 0.1250 \\ P(65) = 0.1250 \\ P(87) = 0.1250 \end{array}$$

Se procede a sumar las dos últimas probabilidades:

$$\begin{array}{r} P(65) = 0.1250 \\ P(87) = 0.1250 \\ \hline \end{array}$$

$$P(6587) = 0.25$$

Se reordena:

$$\begin{array}{r} P(1091211) = 0.25 \\ P(6587) = 0.25 \\ P(141341) = 0.23828 \\ P(150) = 0.13672 \\ P(32) = 0.1250 \end{array}$$

Se procede a sumar las dos últimas probabilidades:

$$\begin{array}{r} P(150) = 0.13672 \\ P(32) = 0.1250 \\ \hline \end{array}$$

$$P(15032) = 0.26172$$

Se reordena:

P(15032) = 0.26172
P(1091211) = 0.25
P(6587) = 0.25
P(141341) = 0.23828

Se procede a sumar las dos últimas probabilidades:

P(6587) = 0.25
P(141341) = 0.23828

P(6587141341) = 0.48828

Se reordena:

P(6587141341) = 0.48828
P(15032) = 0.26172
P(1091211) = 0.25

Se procede a sumar las dos últimas probabilidades:

P(15032) = 0.26172
P(1091211) = 0.25

P(150321091211) = 0.51172

Se reordena:

P(150321091211) = 0.51172
P(6587141341) = 0.48828

Se asigna valores a las dos últimas probabilidades:

P(150321091211) = 0.51172 0
P(6587141341) = 0.48828 1

Y según estos se vuelve con las anteriores:

P(6587141341) = 0.48828 1
P(15032) = 0.26172 00
P(1091211) = 0.25 01

P(15032) = 0.26172 00
P(1091211) = 0.25 01
P(6587) = 0.25 10
P(141341) = 0.23828 11

P(1091211) = 0.25 01
P(6587) = 0.25 10
P(141341) = 0.23828 11
P(150) = 0.13672 000
P(32) = 0.1250 001

P(1091211)= 0.25 01
P(141341) = 0.23828 11
P(150) = 0.13672 000
P(32) = 0.1250 001
P(65) = 0.1250 100
P(87) = 0.1250 101

P(141341)= 0.23828 11
P(150) = 0.13672 000
P(32) = 0.1250 001
P(65) = 0.1250 100
P(87) = 0.1250 101
P(109) = 0.1250 010
P(1211) = 0.1250 011

P(150) = 0.13672 000
P(32) = 0.1250 001
P(65) = 0.1250 100
P(87) = 0.1250 101
P(109) = 0.1250 010
P(1211)= 0.1250 011
P(1413)= 0.1250 110
P(41) = 0.11328 111

P(32) = 0.1250 001
P(65) = 0.1250 100
P(87) = 0.1250 101
P(109)= 0.1250 010
P(41) = 0.11328 111
P(15) = 0.07031 0000
P(0) = 0.06641 0001
P(14) = 0.0625 1100
P(13) = 0.0625 1101
P(12) = 0.0625 0110
P(11) = 0.0625 0111

P(32) = 0.1250 001
P(65) = 0.1250 100
P(87) = 0.1250 101
P(41) = 0.11328 111
P(15) = 0.07031 0000
P(0) = 0.06641 0001
P(14) = 0.0625 1100
P(13) = 0.0625 1101
P(12) = 0.0625 0110
P(11) = 0.0625 0111
P(10) = 0.0625 0100
P(9) = 0.0625 0101

P(32)	=	0.1250	001
P(65)	=	0.1250	100
P(41)	=	0.11328	111
P(15)	=	0.07031	0000
P(0)	=	0.06641	0001
P(14)	=	0.0625	1100
P(13)	=	0.0625	1101
P(12)	=	0.0625	0110
P(11)	=	0.0625	0111
P(10)	=	0.0625	0100
P(9)	=	0.0625	0101
P(8)	=	0.0625	1010
P(7)	=	0.0625	1011

P(32)	=	0.1250	001
P(41)	=	0.11328	111
P(15)	=	0.07031	0000
P(0)	=	0.06641	0001
P(14)	=	0.0625	1100
P(13)	=	0.0625	1101
P(12)	=	0.0625	0110
P(11)	=	0.0625	0111
P(10)	=	0.0625	0100
P(9)	=	0.0625	0101
P(8)	=	0.0625	1010
P(7)	=	0.0625	1011
P(6)	=	0.0625	1000
P(5)	=	0.0625	1001

P(41)	=	0.11328	111
P(15)	=	0.07031	0000
P(0)	=	0.06641	0001
P(14)	=	0.0625	1100
P(13)	=	0.0625	1101
P(12)	=	0.0625	0110
P(11)	=	0.0625	0111
P(10)	=	0.0625	0100
P(9)	=	0.0625	0101
P(8)	=	0.0625	1010
P(7)	=	0.0625	1011
P(6)	=	0.0625	1000
P(5)	=	0.0625	1001
P(3)	=	0.0625	0010
P(2)	=	0.0625	0011

Por último las muestras se codifican así:

P(15)	= 0.07031	0000
P(0)	= 0.06641	0001
P(14)	= 0.0625	1100
P(13)	= 0.0625	1101
P(12)	= 0.0625	0110
P(11)	= 0.0625	0111
P(10)	= 0.0625	0100
P(9)	= 0.0625	0101
P(8)	= 0.0625	1010
P(7)	= 0.0625	1011
P(6)	= 0.0625	1000
P(5)	= 0.0625	1001
P(3)	= 0.0625	0010
P(2)	= 0.0625	0011
P(4)	= 0.05859	1110
P(1)	= 0.05469	1111

Se puede apreciar como se redujo la cantidad de bits para representar a las muestras de imagen (de 8 bits a 4 bits) esto a groso modo representa una tasa de compresión de 2:1 aunque hacen falta además los valores iniciales como el número de píxeles, de todas formas esta es una técnica bastante buena para acompañar a codificaciones como la de TDC en donde se mezcla la eficacia de un esquema con la flexibilidad y alta tasa de compresión de la otra. Esta técnica tiene algo de "Adaptiva" pues en su ejecución optimiza la codificación al asignar el número de bits necesario y suficiente a muestras con particularidades propias. En efecto si la anterior imagen hubiese tenido mucha cantidad de muestras de un solo tipo de color, digamos 300 muestras de valor 15, esto, al codificarse se llegaría a la palabra código 0, pues su probabilidad de ocurrencia será de 0.58594, que es la mayor de entre todas las otras probabilidades incluso mayor que la suma de las otras 212 probabilidades.

ANEXO II-(c)

CODIFICACION MICD

Para la ejecución de este tipo de codificación se utilizará la misma imagen del Anexo II-(a) de la cual ya se tienen sus valores numéricos. Utilizando el esquema de codificación de la Figura 2.2 la labor del codificador MICD es la siguiente (Valor inicial=0):

Secuencia de entrada	Secuencia de salida
15	15 - 0 = 15
0	0 - 15 = -15
0	0 - 0 = 0
0	0 - 0 = 0
15	15 - 0 = 15
15	15 - 15 = 0
15	15 - 15 = 0
15	15 - 15 = 0
15	15 - 15 = 0
15	15 - 15 = 0

.....

Esto para la primera línea de imagen, para la imagen completa se obtienen los siguientes resultados:

15-15	0	0	15	0	0	0	0	0	15	15-15	0	0	15	
-15	0	15	0	-1	0	0	0	0	0	0	1	0	-15	15
-15	15	-1	0	-7	0	0	0	0	0	0	7	0	1	1
0	-1	-7	0	4	0	0	0	0	0	0	-4	0	7	1
-1	-7	4	0	-1	0	0	0	0	0	0	1	0	-4	7
-7	4	-1	0	-7	0	0	0	0	0	0	7	0	1	-3
4	-1	-7	0	5	0	0	0	0	0	0	-5	0	7	1
-1	-7	5	0	-3	0	0	0	0	0	0	3	0	5	7
-7	5	-3	0	-3	0	0	0	0	0	0	3	0	3	-5
5	-3	-3	0	11	0	0	0	0	0	0	-11	0	3	3
-3	-3	11	0	-1	0	0	0	0	0	0	1	0	-11	3
-3	11	-1	0	-6	0	0	0	0	0	0	6	6	1	-11
-1	-6	3	0	-5	0	0	0	0	0	0	5	0	-3	6
-6	3	-5	0	-3	0	0	0	0	0	0	3	0	5	-3
3	-5	-3	0	-1	0	0	0	0	0	0	1	0	-3	5
-1	-3	-1	0	15	0	0	0	0	0	0	-15	0	1	3
-3	-1	15	0	-1	0	0	0	0	0	0	1	0	-15	1
-1	15	-1	0	-7	0	0	0	0	0	0	7	0	1	-15
15	-1	-7	0	4	0	0	0	0	0	0	-4	0	7	1
-1	-7	4	0	-1	0	0	0	0	0	0	1	0	-4	7
-1	-7	4	0	-1	0	0	0	0	0	0	1	0	-4	7
-7	4	-1	0	-7	0	0	0	0	0	0	7	0	1	-3
4	-1	-7	0	5	0	0	0	0	0	0	-5	0	7	1
-1	-7	5	0	-3	0	0	0	0	0	0	3	0	-5	7
-7	5	-3	0	-3	0	0	0	0	0	0	3	0	3	-5
5	-3	-3	0	11	0	0	0	0	0	0	-11	0	3	3
-3	-3	11	0	-1	0	0	0	0	0	0	1	0	-11	3
-3	11	-1	0	-6	0	0	0	0	0	0	6	6	1	-11
-1	-6	3	0	-5	0	0	0	0	0	0	5	0	-3	6
-6	3	-5	0	-3	0	0	0	0	0	0	3	0	5	-3
3	-5	3	0	-1	0	0	0	0	0	0	1	0	-3	5

La MICD es una excelente técnica de reducción de redundancia tal como se evidencia en este ejemplo. Nótese la optimización de carga binaria cuando se codifica con 0's a los elementos repetidos, además su acción codificadora reduce drásticamente la componente continua del flujo de datos original (todos son no negativos). La acción del predictor presente en la codificación es muy importante y es la pieza clave de la decodificación pues si la acción del predictor falla tan solo una vez, la decodificación fallará por completo.

Con respecto al nivel de compresión alcanzado, se lo puede visualizar desde el punto de vista de la reducción del número de bits representativos de la imagen, esto, como si la codificación fuese por longitud variable:

Los valores	15	codificados con	01111
Los valores	-15	codificados con	11111
Los valores	0	codificados con	0
Los valores	1	codificados con	01
Los valores	-1	codificados con	11
Los valores	3	codificados con	011
Los valores	-3	codificados con	111

Y así sucesivamente....Entonces el nivel de compresión será claramente visualizado debido a que en la codificación se registra una gran cantidad de valores 0 y como lo que se codifica son las diferencias entre pixeles adyacentes (en imágenes comunes esta diferencia no es muy grande), esta codificación requiere de muy poca cantidad de bits representativos por muestra, en efecto en la imagen codificada se necesitan (sumando el número de bits requeridos): 1032 bits, y como existen 16x32 pixeles, la tasa binaria alcanzada será de:

$$(1032 \text{ bits}) / (16 \times 32 \text{ pixeles}) = 2.016 \text{ bits/pixel}$$

En cambio que en la imagen original se necesitaban:
 $(8 \text{ bits/pixel})(16 \times 32 \text{ pixeles}) = 4096 \text{ bits/imagen}$; y,
8bits/pixel

ANEXO II-(d)

CODIFICACION MD

Para la ejecución de este tipo de codificación se utilizará la misma imagen del Anexo II-(a) de la cual ya se tienen sus valores numéricos. Utilizando el esquema de codificación de la Figura 2.6 la labor del codificador MICD es la siguiente (Valor inicial=0):

Secuencia de entrada	Secuencia de salida
15	15 - 0 = 15 > 0 1
0	0 - 15 = -15 < 0 0
0	0 - 0 = 0 -
0	0 - 0 = 0 -
15	15 - 0 = 15 > 0 1
15	15 - 15 = 0 -
15	15 - 15 = 0 -
15	15 - 15 = 0 -
15	15 - 15 = 0 -
15	15 - 15 = 0 -

.....

Esto para la primera línea de imagen, para la imagen completa se obtienen los siguientes resultados:

1	0	-	-	1	-	-	-	-	-	-	1	0	-	-	1
0	-	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	1	0	-	-	-	-	-	-	1	-	-	-	
-	0	0	-	1	-	-	-	-	-	-	-	0	1	0	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	-	0	0	
1	0	0	-	1	-	-	-	-	-	-	0	-	1	0	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	-	1	0	
1	0	0	-	1	-	-	-	-	-	-	0	-	1	1	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	1	1	0	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	-	1	0	
1	0	0	-	1	-	-	-	-	-	-	0	-	1	1	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	1	1	0	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	-	1	0	
1	0	0	-	1	-	-	-	-	-	-	0	-	1	1	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	1	1	0	
0	0	1	-	0	-	-	-	-	-	-	1	-	0	1	
0	1	0	-	0	-	-	-	-	-	-	1	-	1	0	
1	0	0	-	0	-	-	-	-	-	-	1	-	0	1	

La técnica MD ya se ha mostrado y es interesante la gran reducción tanto de entropía como de potencia de señal, ya no son necesarios 8bits para representar un pixel, tan solo con 1bit se puede discriminar a los datos, sin embargo su robustez es mucho menor que MICD o cualquier otra técnica de codificación, pues un sencillo cambio en uno de sus bits arruinaría la armonía de codificación mientras que en alguna otra técnica, se podría prescindir de alguno de sus bits sin perder mucha información.

Con respecto al esquema de imagen comprimida presentada, se debe indicar que un 1 representa que la futura muestra será mayor que la actual, se toman las medidas del caso y se procede a decodificar esta muestra, reciprocamente se cumple con el valor de muestra 0. El signo - indica que la muestra futura permanece al mismo nivel de la actual, es decir no ocurre cambio de nivel.

La tasa de compresión se puede estimar del número de bits/pixel que la imagen original necesitaba (8 bits/pixel) y la imagen codificada:

- Sea el estado 1 post-codificado con 11
- Sea el estado 0 post-codificado con 00
- Sea el estado - post-codificado con 0
- Sea el valor inicial post-codificado con 8bits

Entonces el número total de bits necesario para que el receptor la pueda decodificar es:

$$2(\# \text{ de "0"}) + 2(\# \text{ de "1"}) + \# \text{ de "-" } + 8\text{bits}$$
$$117 + 98 + 297 + 8 = 520 \text{ bits}$$

Que para la imagen de 16x32 pixeles se obtiene:

$$(520 \text{ bits}) / (16 \times 32 \text{ pixeles}) = 1.016 \text{ bits/pixel}$$

ANEXO II - (e)

TRANSFORMADA DE KL

Como un ejemplo sencillo, se considera la imagen conformada de dos pixeles:

$$f(x_1)f(x_2)$$

la cual se puede expresar como el vector:

$$[f(z)] = [f(x_1)f(x_2)]$$

A fin de computar la matriz de covarianza, debe asumirse que esta imagen de dos pixeles se la ha tomado de una matriz mayor de 4x4 pixeles que tiene los siguientes valores:

6	4	5	5
5	4	5	4
5	6	5	6
4	6	5	5

El pixel de valor medio es de 5 y la matriz covarianza de la imagen original de dos pixeles es entonces:

$$C_r = \begin{bmatrix} E[(f_1-5)^2] & E[(f_1-5)(f_2-5)] \\ E[(f_2-5)(f_1-5)] & E[(f_2-5)^2] \end{bmatrix}$$

donde $E[.]$ es el operador esperanza. A fin de simplificar el cómputo de los elementos C_r se subtrae el valor medio (=5) de cada elemento en la matriz de 4x4, obteniéndose:

1	-1	0	0
0	-1	0	-1
0	1	0	1
-1	1	0	0

Luego se computan los elementos de C_r promediándolos en cada caso sobre los 12 posibles casos de subimágenes de 2 pixeles en la imagen de 4x4:

$$(f_1, f_2) : (1, -1); (0, -1); (0, 1); (-1, 1); (-1, 0); (-1, 0); (1, 0); (1, 0); (0, 0); (0, -1); (0, 1); (0, 0)$$

Luego:

$$E[(f_1-5)^2] = [1+0+0+(-1)^2+(-1)^2+(-1)^2+1+1+0+0+0+0] / 12 = 1/2$$

$$E[(f_2-5)^2] = [(-1)^2+(-1)^2+1+1+0+0+0+0+0+(-1)+1+0] / 12 = 1/2$$

$$E[(f_1 - 5)(f_2 - 5)] = E[(f_2 - 5)(f_1 - 5)] = -2 / 12 = -1/6$$

Así,

$$C_2 = \begin{bmatrix} 1/2 & -1/6 \\ -1/6 & 1/2 \end{bmatrix} = \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix}$$

Resolviendo la ecuación característica,

$$\det[C_2 - \alpha I] = 0$$

$$\det \begin{bmatrix} (3-\alpha) & -1 \\ -1 & (3-\alpha) \end{bmatrix} = 0$$

$$\begin{aligned} (3 - \alpha)^2 - 1 &= 0 \\ (\alpha - 2)(\alpha - 4) &= 0 \end{aligned}$$

Así, los valores propios son:

$$\alpha_1 = 4 \text{ y } \alpha_2 = 2$$

Para encontrar los vectores propios, se sustituyen los valores propios en $[C_2 - \alpha I]\phi = 0$.

para $\alpha_1 = 4$:

$$\begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = 0$$

$$\circ \quad \phi_1 = -\phi_2 \implies \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

para $\alpha_2 = 2$:

$$\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = 0$$

$$\circ \quad \phi_1 = \phi_2 \implies \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

La transformación es entonces:

$$[A] = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}$$

Si se computa la matriz covarianza de los pixeles transformados, se conseguirá la matriz diagonal esperada:

$$G = [A]^t C_x [A] = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 & -1 \\ -1 & 3 \end{bmatrix} \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 8 & 0 \\ 0 & 4 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$$

ANEXO II - (F)

TRANSFORMADA DE FOURIER

Dada la secuencia $X = 1, 2, 3, 4$

Hallar la TDF y la TDF Inversa.

Aplicando la ecuación (2.41) con $N=4$, la TDF de esta secuencia (bloque imagen de 4×4 pixeles) está dada por:

$$\begin{bmatrix} C(0) \\ C(1) \\ C(2) \\ C(3) \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

$$\begin{aligned} C(0) &= 5/2 &= 2.5 \\ C(1) &= -1/2 + j/2 &= -0.5 + j0.5 \\ C(2) &= -1/2 &= -2.5 \\ C(3) &= -1/2 - j/2 &= -0.5 - j0.5 \end{aligned}$$

La TDF inversa estará dada por(2.46):

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 5/2 \\ -1/2(1-j) \\ -1/2 \\ -1/2(1+j) \end{bmatrix}$$

$$\begin{aligned} X(0) &= 1 \\ X(1) &= 2 \\ X(2) &= 3 \\ X(3) &= 4 \end{aligned}$$

Como puede verse, la transformación logra preservar a la información pues se obtienen los datos originales libres de error.

ANEXO II-(h)

TRANSFORMADA DE HAAR

Dada la secuencia $X = 1, 2, 3, 4, 5, 6, 7, 8$

Hallar la Transformación de Haar y su inversa.

Aplicando la matriz de transformación (2.50) con $N=8$, la transformación de esta secuencia (bloque imagen de 8×8 pixeles) está dada por:

$$\begin{array}{l}
 H(0) \\
 H(1) \\
 H(2) \\
 H(3) \\
 H(4) \\
 H(5) \\
 H(6) \\
 H(7)
 \end{array}
 = \frac{1}{8}
 \begin{bmatrix}
 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\
 \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\
 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2
 \end{bmatrix}
 \begin{array}{l}
 1 \\
 2 \\
 3 \\
 4 \\
 5 \\
 6 \\
 7 \\
 8
 \end{array}$$

Así,

$$\begin{array}{l}
 H(0)=9/2, \quad H(1)=-2, \quad H(2)=-\sqrt{2}/2, \quad H(3)=-\sqrt{2}/2 \\
 H(4)=-1/4, \quad H(5)=-1/4, \quad H(6)=-1/4, \quad H(7)=-1/4
 \end{array}$$

Y la transformación inversa es:

$$\begin{array}{l}
 X(0) \\
 X(1) \\
 X(2) \\
 X(3) \\
 X(4) \\
 X(5) \\
 X(6) \\
 X(7)
 \end{array}
 =
 \begin{bmatrix}
 1 & 1 & \sqrt{2} & 0 & 2 & 0 & 0 & 0 \\
 1 & 1 & \sqrt{2} & 0 & -2 & 0 & 0 & 0 \\
 1 & 1 & -\sqrt{2} & 0 & 0 & 2 & 0 & 0 \\
 1 & 1 & -\sqrt{2} & 0 & 0 & -2 & 0 & 0 \\
 1 & -1 & 0 & \sqrt{2} & 0 & 0 & 2 & 0 \\
 1 & -1 & 0 & \sqrt{2} & 0 & 0 & -2 & 0 \\
 1 & -1 & 0 & -\sqrt{2} & 0 & 0 & 0 & 2 \\
 1 & -1 & 0 & -\sqrt{2} & 0 & 0 & 0 & -2
 \end{bmatrix}
 \begin{array}{l}
 9/2 \\
 -2 \\
 -\sqrt{2}/2 \\
 -\sqrt{2}/2 \\
 -1/4 \\
 -1/4 \\
 -1/4 \\
 -1/4
 \end{array}$$

Así,

$$\begin{array}{l}
 X(0)=1, \quad X(1)=2, \quad X(2)=3, \quad X(3)=4 \\
 X(4)=5, \quad X(5)=6, \quad X(6)=7, \quad X(7)=8
 \end{array}$$

ANEXO II-(i)
TRANSFORMADA DISCRETA COSENO

Al igual que las otras transformadas, se procede a representar a una imagen como una matriz, en este caso de 16x16:

Imagen de entrada:

15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15
0	0	15	0	15	15	15	15	15	15	15	0	15	15	0	15
0	0	0	15	15	7	7	7	7	15	0	15	0	15	0	15
0	15	0	15	7	7	7	7	7	15	0	0	15	0	0	15
0	0	15	7	7	7	7	7	15	0	15	0	0	15	0	15
0	15	15	7	7	7	7	7	15	15	0	15	0	0	0	15
0	15	15	7	7	7	7	15	0	15	15	0	15	0	0	15
0	0	15	7	7	7	7	15	15	0	15	15	0	15	0	15
0	15	15	7	7	7	15	0	0	15	0	15	15	0	0	15
0	0	15	15	7	7	15	15	0	0	15	0	15	15	0	15
0	0	0	15	15	15	15	0	15	0	0	15	0	15	0	15
0	15	0	0	15	0	15	15	0	15	0	0	15	0	0	15
0	0	15	0	0	15	0	15	15	0	15	0	0	15	0	15
0	15	0	15	0	0	15	0	15	15	0	15	0	0	0	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15

Imagen codificada por la TDC:

$$C(u, v) = \alpha(u) \alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}$$

para $u=0,1,2,\dots,N-1$;

$v=0,1,2,\dots,N-1$

donde $N=16$ y $f(x,y)$ es el pixel en la fila x columna y ;

116	-7	-11	-17	2	-18	-1	-19	6	-18	14	-16	8	-12	10	-3
16	0	-1	1	-2	-1	0	8	0	0	0	0	2	6	0	0
-9	-5	2	-1	8	3	4	10	2	2	0	3	-3	-2	-1	0
16	-1	0	-4	-9	12	0	-2	0	5	2	-1	2	4	1	4
0	1	7	12	3	-4	1	-10	-4	1	-5	6	1	5	-1	1
10	5	0	3	8	-1	-1	0	-2	-1	-8	-1	2	1	-3	1
0	3	9	-3	-5	1	0	1	2	4	3	6	-5	-7	-3	2
12	1	2	6	0	0	-3	-2	-1	0	-1	2	0	-2	-4	-3
5	0	0	-2	-2	3	4	6	7	2	-2	-5	-11	0	-1	-5
21	3	-3	2	-5	-3	-2	0	0	3	2	3	-4	-5	-1	-8
10	1	0	2	0	1	0	0	-2	-5	-1	-2	-5	10	8	-2
20	-1	-9	-1	-3	0	0	5	2	4	-6	13	3	-18	14	2
13	-6	-5	-5	-1	4	-8	2	-8	12	0	-8	36	10	-6	11
15	6	-2	6	0	2	0	-3	-5	-11	11	-13	-22	12	-8	4
8	2	-2	0	0	0	0	2	-2	-10	-3	8	4	0	-2	0
7	0	-4	2	4	1	-5	-2	3	0	0	-4	-4	?	-1	0

Imagen decodificada por TDC inversa:

$$f(x, y) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \alpha(u) \alpha(v) C(u, v) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}$$

para $x=0,1,2,\dots,N-1$;
 $y=0,1,2,\dots,N-1$

con α definiéndose como:

$$\alpha(i) = \begin{cases} \sqrt{\frac{1}{N}} & \text{si } i=0 \\ \sqrt{\frac{2}{N}} & \text{si } i=1,2,\dots,N-1 \end{cases} \quad (5.9)$$

14	14	14	14	14	14	14	14	14	14	14	15	14	15	13	14
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	15	1	14	14	14	14	14	14	14	0	14	14	0
14	0	0	0	14	14	7	8	7	7	14	0	14	0	13	0
15	0	14	0	14	7	7	6	6	6	14	0	0	14	0	0
15	1	0	14	7	6	7	7	6	13	0	14	0	0	14	0
14	0	14	14	7	7	6	6	6	14	14	0	15	0	1	0
15	0	15	14	7	7	7	6	14	0	14	14	0	14	0	0
14	0	0	13	6	6	7	7	13	14	0	14	15	0	14	0
14	0	14	14	7	7	6	13	0	1	14	0	14	15	0	0
13	0	0	15	14	6	7	13	15	0	0	15	0	14	15	0
14	0	1	1	15	14	14	15	0	14	1	0	14	0	13	0
13	0	14	0	1	14	0	14	14	0	14	0	0	14	0	0
14	0	0	15	0	0	14	0	14	13	0	14	0	0	14	0
15	0	15	0	14	1	0	14	0	13	15	0	15	0	1	0
14	14	0	1	0	0	0	0	0	0	1	0	0	0	0	0

Nótese que a diferencia de las anteriores transformadas, la TDC no reproduce fielmente a los valores de pixeles originales con su transformación inversa, existen notables pérdidas de información que en algunos casos y dentro de ciertos rangos de la percepción visual humana, estos errores son aceptables. Estos casos se dan en el procesamiento por bloques de una imagen de video en movimiento, en este tipo de imagen, la correlación entre tramas consecutivas puede hacer que las imperfecciones de la TDC, no sean tan drásticas. Para esta codificación se generó un error cuadrático medio e^2_{cm} (Ecuación 2.80 con $N=16$) de:

$$e^2_{cm} = 71.84, \text{ demasiado alto}$$

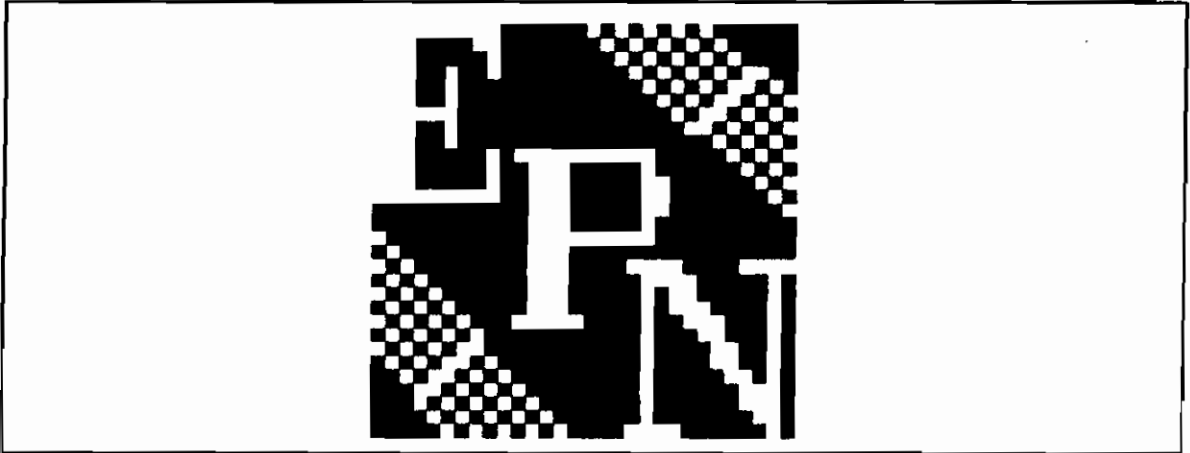
Y un SNR(dB) (Ecuación 2.81, con $\sigma^2_1=49.19$) de:

$$\text{SNR(dB)} = -1.64\text{dB}, \text{ bajo.}$$

ANEXO II - (j)

CODIFICACION OBB UNIDIMENSIONAL

Para la ejecución de este tipo de codificación se utilizará la siguiente imagen a dos tonalidades:



Cuya representación numérica es:

```
1111111111111111111111111111111111
11111111110000001010101010000001
11110000110000000101010101000001
11110000010000000010101010100001
11110010010000000001010101011001
11110010000000000000101010110101
111100100000000000000010101101011
11111110000000000000001011010101
111100100000000000000000110101011
11110010000000000000000001010101
11110000010111111111100000101011
11110000010011100000100000010101
11110000010011100000110000001011
11111111110011100000110000000101
10000000000011100000110000000011
10000000000011100000100000000001
110000000000111111111100000000001
10100000000011100000000000000001
11010000000011100001111000011111
10101000000011100000111100000101
11010100000011100000101100000101
10101010000011100000101110000101
11010101000111110000100111000101
10101010100000000000100011000101
11010101100000000000100001100101
10101011010000000000100001100101
10010110101000000000100000110101
10001101010100000000100000011101
10001010101010000000100000011101
10000101010101000000100000001101
10000010101010100001111000000101
1111111111111111111111111111111111
```

La codificación se realiza de la siguiente forma:

Sea N=8 (tamaño del bloque en pixeles, Ec. 2.67),

1=Blanco, 0=Negro,

Primera línea: 4 bloques de 8 pixeles

Primer bloque : 11111111 =>1
Segundo bloque: 11111111 =>1
Tercer bloque : 11111111 =>1
Cuarto bloque : 11111111 =>1

Segunda línea: 4 bloques de 8 pixeles

Primer bloque : 11111111 =>1
Segundo bloque: 11000000 =>011000000
Tercer bloque : 10101010 =>010101010
Cuarto bloque : 10000001 =>010000001

Tercera línea: 4 bloques de 8 pixeles

Primer bloque : 11110000 =>011110000
Segundo bloque: 11000000 =>011000000
Tercer bloque : 01010101 =>001010101
Cuarto bloque : 01000001 =>001000001

Cuarta línea : 4 bloques de 8 pixeles

Primer bloque : 11110000 =>011110000
Segundo bloque: 01000000 =>001000000
Tercer bloque : 00101010 =>000101010
Cuarto bloque : 10100001 =>010100001

Quinta línea : 4 bloques de 8 pixeles

Primer bloque : 11110010 =>011110010
Segundo bloque: 01000000 =>001000000
Tercer bloque : 00010101 =>000010101
Cuarto bloque : 01011001 =>001011001

Sexta línea : 4 bloques de 8 pixeles

Primer bloque : 11110010 =>011110010
Segundo bloque: 00000000 =>000000000
Tercer bloque : 00001010 =>000001010
Cuarto bloque : 10110101 =>010110101

Séptima línea: 4 bloques de 8 pixeles

Primer bloque : 11110010 =>011110010
Segundo bloque: 00000000 =>000000000
Tercer bloque : 00000101 =>000000101
Cuarto bloque : 01101011 =>001101011

Y así sucesivamente hasta completar las 16 líneas, con lo que se obtiene un grupo de palabras código a transmitir. En el decodificador lo que recibe son estas palabras código y además debería conocerse el tamaño del bloque y el de la imagen, por lo general se toman 100ppp o 200 ppp.

ANEXO II - (k)

CODIFICACION POR TRUNCAMIENTO DE BLOQUE

Al igual que las otras codificaciones, se procede a representar a una imagen como una matriz, en este caso de 16x16:

Imagen de entrada:

15 15 15 15	15 15 15 15	15 15 15 15	15 15 15 15
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 15
0 0 15 0	15 15 15 15	15 15 15 0	15 15 0 15
0 0 0 15	15 7 7 7	7 15 0 15	0 15 0 15
0 15 0 15	7 7 7 7	7 15 0 0	15 0 0 15
0 0 15 7	7 7 7 7	15 0 15 0	0 15 0 15
0 15 15 7	7 7 7 7	15 15 0 15	0 0 0 15
0 15 15 7	7 7 7 15	0 15 15 0	15 0 0 15
0 0 15 7	7 7 7 15	15 0 15 15	0 15 0 15
0 15 15 7	7 7 15 0	0 15 0 15	15 0 0 15
0 0 15 15	7 7 15 15	0 0 15 0	15 15 0 15
0 0 0 15	15 15 15 0	15 0 0 15	0 15 0 15
0 15 0 0	15 0 15 15	0 15 0 0	15 0 0 15
0 0 15 0	0 15 0 15	15 0 15 0	0 15 0 15
0 15 0 15	0 0 15 0	15 15 0 15	0 0 0 15
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 15

La técnica CTB puede aplicarse a toda esta imagen como si fuese un bloque de 16x16 pixeles, pero experiencias prácticas con este tipo de codificación demuestran que no es lo más recomendable, pues se producen demasiadas pérdidas en los datos por la particularidad propia del método.

En este caso se procede a dividir a la imagen en bloques de 4x4 pixeles y a cada uno se le aplicará la técnica CTB que implica los siguientes pasos:

Para cada bloque de 4x4 se calcula:

- Primer momento = \bar{X} ec.(2.69)
- Segundo momento = \bar{X}^2 ec.(2.70)
- Varianza = σ^2 ec.(2.71)

Con estos datos se procede a realizar una matriz mapa:

Si $f(i,j) > \bar{X}$ entonces $m(i,j) = 1$
 Si $f(i,j) < \bar{X}$ entonces $m(i,j) = 0$

donde $f(i,j)$ es el pixel de fila i y columna j del bloque.

Y lo que se transmite es el primer momento, la varianza y la matriz mapa.

Para el bloque 1:

$$\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array} \quad \bar{X} = 5.625$$

$$\sigma = 8.874$$

Para el bloque 2:

$$\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{array} \quad \bar{X} = 9.750$$

$$\sigma = 6.379$$

Para el bloque 3:

$$\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{array} \quad \bar{X} = 8.875$$

$$\sigma = 7.132$$

Para el bloque 4:

$$\begin{array}{cccc} 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{array} \quad \bar{X} = 9.375$$

$$\sigma = 7.262$$

Para el bloque 5:

$$\begin{array}{cccc} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{array} \quad \bar{X} = 7.875$$

$$\sigma = 6.753$$

Para el bloque 6:

$$\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array} \quad \bar{X} = 7.500$$

$$\sigma = 1.936$$

Para el bloque 7:

$$\begin{array}{cccc} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{array} \quad \bar{X} = 7.938$$

$$\sigma = 7.250$$

Para el bloque 8:

$$\begin{array}{cccc} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{array} \quad \bar{X} = 6.563$$

$$\sigma = 7.441$$

Para el bloque 9:

$$\begin{array}{cccc} 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{array} \quad \bar{X} = 6.500$$

$$\sigma = 6.946$$

Para el bloque 10:

$$\begin{array}{cccc} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{array} \quad \bar{X} = 9.750$$

$$\sigma = 5.219$$

Para el bloque 11:

$$\begin{array}{cccc} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{array} \quad \bar{X} = 7.500$$

$$\sigma = 7.500$$

Para el bloque 12:

$$\begin{array}{cccc} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{array} \quad \bar{X} = 8.438$$

$$\sigma = 7.441$$

Para el bloque 13:

0	1	0	0
0	0	1	0
0	1	0	1
0	0	0	0

$$\bar{X} = 3.750$$

$$\sigma = 6.495$$

Para el bloque 14:

1	0	1	1
0	1	0	1
0	0	1	0
0	0	0	0

$$\bar{X} = 5.625$$

$$\sigma = 7.262$$

Para el bloque 15:

0	1	0	0
1	0	1	0
1	1	0	1
0	0	0	0

$$\bar{X} = 5.625$$

$$\sigma = 7.262$$

Para el bloque 16:

1	0	0	1
0	1	0	1
0	0	0	1
0	0	0	1

$$\bar{X} = 5.625$$

$$\sigma = 7.262$$

Es evidente la disminución de bits para representar a una muestra que en definitiva se representa con un bit de la matriz mapa más los bits necesarios de la varianza y momento, que por lo general son 8, así la cantidad de bits por bloque serían:

$$4 \times 4 + 8 + 8 = 32 \text{ bits/bloque}$$

Como originalmente existían 16 pixeles (representados por 8 bits en el caso de una imagen de alta resolución) por bloque, esto nos daba una tasa de:

$$16 \times 8 = 128 \text{ bits/bloque}$$

Se ha reducido esta tasa por un factor de 4. La imagen reconstruida se la obtiene al calcular a y b según las fórmulas (2.73) y (2.74) teniendo en cuenta que $m=4 \times 4=16$, y q es el número de 1's en la matriz mapa, todo este cálculo lo hace el receptor además de obtener los valores de pixeles según:

$$\text{Si } m(i,j) = 1 \text{ entonces } f(i,j) = a$$

$$\text{Si } m(i,j) = 0 \text{ entonces } f(i,j) = b$$

Y de esta forma se obtienen los valores de pixeles de la imagen en el receptor:

14 14 14 14	15 15 15 15	15 15 15 15	14 14 14 14
0 0 0 0	2 2 2 2	0 0 0 0	0 0 0 14
0 0 14 0	15 15 15 15	15 15 15 0	14 14 0 14
0 0 0 14	15 2 2 2	0 15 0 15	0 14 0 14
1 15 1 15	7 7 7 7	0 15 0 0	15 0 0 15
1 1 15 1	7 7 7 7	15 0 15 0	0 15 0 15
1 15 15 1	7 7 7 7	15 15 0 15	0 0 0 15
1 15 15 1	7 7 7 14	0 15 15 0	15 0 0 15
0 0 13 13	5 5 5 15	15 0 15 15	0 15 0 15
0 13 13 13	5 5 15 5	0 15 0 15	15 0 0 15
0 0 13 13	5 5 15 15	0 0 15 0	15 15 0 15
0 0 0 13	15 15 15 5	15 0 0 15	0 15 0 15
0 15 0 0	14 0 14 14	0 14 0 0	14 0 0 14
0 0 15 0	0 14 0 14	14 0 14 0	0 14 0 14
0 15 0 15	0 0 14 0	14 14 0 14	0 0 0 14
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 14

Las diferencias existen, pero no son muy notables. Para esta codificación se generó un error cuadrático medio ϵ_{α} (Ecuación 2.80 con $N=16$) de:

$\epsilon_{\alpha} = 2.00$, valor que puede considerarse bastante bueno. Y un **SNR(dB)** (Ec. 2.81, con $\sigma_r=49.19$) de:

$$\text{SNR(dB)} = 13.91\text{dB}, \text{ que es alto.}$$

Al comparar estos resultados con los del Anexo II-(i) se pueden establecer dos límites muy distantes tanto del ϵ_{α} como del **SNR(dB)** sin embargo es admirable el hecho de que las imágenes descomprimidas por los dos métodos se "observan" por separado con ligeras diferencias con respecto al original, esto es debido claro a la característica capacidad visual humana.

ANEXO II-(1)

FORMATO GRAFICO PCX

Para este tipo de codificación se utilizará la misma imagen del Anexo II-(a), con su respectiva representación numérica. Se debe recordar como se realizaba la codificación de los diferentes tipos de formatos gráficos(Sección 2.1.4.5.2) en los que existían: una cabecera, y bytes adicionales de información sobre el tipo de paleta a utilizar, el número de colores de la imagen, las dimensiones de la imagen, y dependiendo del tipo de formato, el esquema de codificación utilizado(CL, Huffman, OBB, etc.). Para el formato PCX, la imagen de prueba se presentaría de la siguiente forma(archivo de 466bytes):

Disk Editor
6.01
Wednesday, May 24, 1995 10:33 am

C:\WIN95\prueba.pcx

Cluster 3,950, Sector 61,532

00000000:	6A 03 01 01 00 00 00 00	- 0F 00 1F 00 86 02 E0 01
00000010:	FF 00 4A 6C 96 51 27 01	- 2C 00 7F 11 05 00 FF FF	..Ji.0'...0....
00000020:	58 0v 50 6F 84 70 10 6C	- 0E 0F 4F 13 52 6C 5F 13	λ.Po.p.i..0.R1_
00000030:	00 00 0C 00 84 1A 50 6F	- 84 70 86 14 97 04 7E 03Po.p....?
00000040:	00 04 02 00 00 00 00 00	- 04 06 E6 6B 00 00 7F 03k..0
00000050:	00 7F 00 00 EA 6B 04 15	- 67 08 50 6F FB 6B 3D 09	.0...k..g.Fo.k=.
00000060:	00 00 7E 03 50 6F 04 00	- 67 08 13 6C FC 0E CF 13	..?.Po.g..l....
00000070:	F9 25 0C 00 00 00 8E 12	- D7 00 2D 22 DB 24 E0 19	.5.....-'.5..
00000080:	0F C1 F0 0F C1 F0 0F C1	- F0 0F C1 F0 30 0C 3F C10.?
00000090:	FC 3F C1 FC 3F C1 FC 40	- 02 70 0E 70 0E 7F C1 FE	.?..?.e.p.p.0..
000000A0:	8F C1 F1 C1 CF C1 F3 C1	- C0 03 C2 FF 30 0C BF C10...
000000B0:	FD 80 01 C2 FF 4F C1 F2	- 7F C1 FE C2 00 C1 F0 0F0..0.....
000000C0:	DF C1 FD C2 FF 0F C1 F0	- C1 C0 03 7F C1 FE C1 F00.....
000000D0:	0F 3F C1 FC 80 01 C1 F0	- 0F C1 CF C1 F3 70 0E C2	.?.....p..
000000E0:	00 C1 CF C1 F3 80 0D C1	- 2F C1 F3 0F C1 F0 80 0D0.....
000000F0:	40 C2 BF C1 FD 3F C1 FC	- 40 C2 30 01 7F C1 FE 7F	é...?.é...0.0
00000100:	C1 FE 8F C1 F1 C2 00 C1	- F0 0F C2 FF 30 0C C2 000.....
00000110:	C1 CF C1 F3 C1 F0 0F 4F	- C1 F2 C2 00 80 00 C1 C0e...0.....
00000120:	03 80 0D C2 00 40 02 80	- 01 4F C1 F2 0F C1 F0 8F?.?.?.?
00000130:	C1 F1 0F C1 F0 80 0D 3F	- C1 FC 3F C1 FC 3F C1 FCe.p.p.0.....
00000140:	40 02 70 0E 70 0E 7F C1	- FE 8F C1 F1 C1 CF C1 F30.....0.
00000150:	C1 00 03 C2 FF 30 0C 6F	- C1 FD 80 01 C2 FF 4F C10.....0.
00000160:	F2 7F C1 FE C2 00 C1 F0	- 0F 8F C1 F0 C2 FF 0F C1	.0.....0.....
00000170:	F0 C1 C0 03 7F C1 FE C1	- F0 0F 3F C1 FC 80 01 C10.....?
00000180:	F0 0F C1 CF C1 F3 70 0E	- C2 00 C1 CF C1 F3 50 0Dp.....
00000190:	C1 CF C1 F3 0F C1 F0 80	- 00 40 02 BF C1 FD 3F C1e...?.
000001A0:	FC 40 02 30 01 7F C1 FE	- 7F C1 FE 8F C1 F1 C2 00	.é...0.0.....
000001B0:	C1 F0 0F C2 FF 30 0C C2	- 00 C1 CF C1 F3 C1 F0 0F0.....
000001C0:	4F C1 F2 C1 00 80 0D C1	- C0 03 80 0D C2 00 40 02	0.....é.....
000001D0:	80 01		..

Al inspeccionar al archivo desde la posición 0000-0000 se puede observar lo siguiente:

00000000: 0A Como es PCX, siempre tendrá este valor(10)
00000001: 03 Indica la versión del codificador PCX(3)
00000002: 01 Esquema de codificación(1=Cod. Longitudinal)
00000003: 01 Número de bits/pixel(1)
00000004: 00 00 Coord. X superior izquierda de la imagen(0)
00000006: 00 00 Coord. Y superior izquierda de la imagen(0)
00000008: 0F 00 Coord. X inferior derecha de la imagen(15)
0000000A: 1F 00 Coord. Y inferior derecha de la imagen(31)
0000000C: 80 02 Resolución horizontal de la pantalla(640pels)
0000000E: E0 01 Resolución vertical de la pantalla(480pels)

Paleta de 16 colores, 48 bytes(3 bytes por color):

00000010: FF 00 4A 6C 96 51 27 01 - 2C 00 7F 11 05 00 FF FF
00000020: 58 00 50 6F 84 70 10 6C - 0E 0F 4F 13 52 6C 5F 13
00000030: 00 00 0C 00 B4 1A 50 6F - 84 70 B6 14 97 04 7E 03

00000040: 00 Reservado(0)
00000041: 04 Planos de color(4)
00000042: 02 00 Bytes por línea barrida(2)
00000044: 00 00 Tipo de paleta(0)

Bytes de relleno(58 bytes):

00000046: 00 00 - 04 00 E6 6B 00 00 7F 03
00000050: 00 7F 00 00 EA 6B 04 15 - 67 08 50 6F FB 6B 3D 09
00000060: 00 00 7E 03 50 6F 04 00 - 67 08 13 6C FC 0E CF 13
00000070: F9 35 0C 00 00 00 AE 12 - D7 00 2D 22 DB 24 E0 19

Imagen misma(338 bytes):

00000080: 0F C1 F0 0F C1 F0 0F C1 - F0 0F C1 F0 30 0C 3F C1
00000090: FC 3F C1 FC 3F C1 FC 40 - 02 70 0E 70 0E 7F C1 FE
000000A0: 8F C1 F1 C1 CF C1 F3 C1 - C0 03 C2 FF 30 0C BF C1
000000B0: FD 80 01 C2 FF 4F C1 F2 - 7F C1 FE C2 00 C1 F0 0F
000000C0: BF C1 FD C2 FF 0F C1 F0 - C1 C0 03 7F C1 FE C1 F0
000000D0: 0F 3F C1 FC 80 01 C1 F0 - 0F C1 CF C1 F3 70 0E C2
000000E0: 00 C1 CF C1 F3 B0 0D C1 - CF C1 F3 0F C1 F0 B0 0D
000000F0: 40 02 BF C1 FD 3F C1 FC - 40 02 80 01 7F C1 FE 7F
00000100: C1 FE 8F C1 F1 C2 00 C1 - F0 0F C2 FF 30 0C C2 00
00000110: C1 CF C1 F3 C1 F0 0F 4F - C1 F2 C2 00 B0 0D C1 C0
00000120: 03 B0 0D C2 00 40 02 80 - 01 4F C1 F2 0F C1 F0 8F
00000130: C1 F1 0F C1 F0 B0 0D 3F - C1 FC 3F C1 FC 3F C1 FC
00000140: 40 02 70 0E 70 0E 7F C1 - FE 8F C1 F1 C1 CF C1 F3
00000150: C1 C0 03 C2 FF 30 0C BF - C1 FD 80 01 C2 FF 4F C1
00000160: F2 7F C1 FE C2 00 C1 F0 - 0F BF C1 FD C2 FF 0F C1
00000170: F0 C1 C0 03 7F C1 FE C1 - F0 0F 3F C1 FC 80 01 C1
00000180: F0 0F C1 CF C1 F3 70 0E - C2 00 C1 CF C1 F3 B0 0D
00000190: C1 CF C1 F3 0F C1 F0 B0 - 0D 40 02 BF C1 FD 3F C1
000001A0: FC 40 02 80 01 7F C1 FE - 7F C1 FE 8F C1 F1 C2 00
000001B0: C1 F0 0F C2 FF 30 0C C2 - 00 C1 CF C1 F3 C1 F0 0F
000001C0: 4F C1 F2 C2 00 B0 0D C1 - C0 03 B0 0D C2 00 40 02
000001D0: 80 01

Ya teniendo los datos de la cabecera como las dimensiones de la imagen, tipo de resolución, bits/pixel, bytes/línea y planos de color, la decodificación se la realiza de la siguiente forma a partir del byte 00000080:

Para cada BYTE hacer:

*Si (BYTE & C0h) <> 0 entonces hacer:
 CONTADOR=(BYTE & 3Fh)
 PIXEL=BYTE siguiente
Fin Si*

*Caso contrario hacer:
 CONTADOR=1
 PIXEL=BYTE
Fin Caso*

PONER_PIXEL_PANTALLA

Fin Para

En el proceso PONER_PIXEL_PANTALLA lo que se hace es dibujar el pixel de color PIXEL en la pantalla gráfica de la resolución dada, un número de veces igual al valor de CONTADOR, es decir se realiza una decodificación longitudinal (de-CL).

Existe entonces un proceso de compresión de los datos representativos de la imagen, pues a nivel de archivo en el disco, si la imagen hubiese sido grabada en forma "cruda", ocuparía:

$$16 \times 32 = 512 \text{ bytes}$$

Pero con la codificación en formato PCX, lo que se obtiene es un volumen neto (cabecera + imagen) de 466 bytes.

ANEXO II--(m)

FORMATO GRAFICO BMP

Para este tipo de formato gráfico se utilizará la misma imagen del Anexo II-(a). Se mostrará a continuación algunas variantes de este formato BMP:

BMP monocromático.

BMP 16 colores.

BMP 256 colores.

BMP 24 colores.

Archivo BMP monocromático(190 bytes):

Dist Editor
6.01
Wednesday, May 24, 1995 11:49 am

C:\WINDOWS\Syspneu.bmp

Cluster 7,752, Sector 60,364

00000000:	42 4B BE 00 20 00 00 00	- 00 00 2E 00 00 00 29 00	EM.....?
00000010:	00 00 10 00 00 00 20 00	- 00 00 01 00 01 00 00 00
00000020:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000030:	00 00 00 00 00 00 00 00	- 00 00 FF FF FF 00 00 00
00000040:	15 00 00 00 15 00 00 00	- 00 00 80 01 80 00 40 02?
00000050:	54 00 30 00 00 00 0F F0	- AA 00 00 00 00 00 80 01	T.O.....
00000060:	00 00 C0 03 00 00 F0 0F	- 00 00 FF FF 00 00 FF FF
00000070:	00 00 7F FE 00 00 3F FC	- 00 00 0F F0 00 00 00 00	..Δ...?
00000080:	00 00 00 00 00 00 00 00	- 00 00 80 01 00 00 40 C2è.
00000090:	00 00 30 00 00 00 0F F0	- 00 00 00 00 00 00 80 01	..0.....
000000A0:	00 00 C0 03 00 00 F0 0F	- 00 00 FF FF 00 00 FF FF
000000B0:	00 00 7F FE 00 00 3F FC	- 00 00 0F F0 00 00 00 00	..Δ...?

Archivo BMP 16 colores(374 bytes):

Disk Editor

5.01

Wednesday, May 24, 1995 11:59 am

XXXXXXXXXXXXXXXXXXXX
C:\WINDOWS\p16c.bmp
XXXXXXXXXXXXXXXXXXXX

Cluster 3,758, Sector 60,460

00000000:	42 4D 7c 01 00 00 00 00	- 00 00 76 00 00 00 28 00	BMy.....v....(.
00000010:	00 00 10 60 00 00 20 00	- 00 00 01 00 04 00 00 00
00000020:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000030:	00 00 00 00 00 00 00 00	- 00 00 00 00 80 00 00 80f.....
00000040:	00 00 00 80 80 00 80 00	- 00 00 80 00 80 00 80 80
00000050:	00 00 80 80 80 00 C0 C0	- C0 00 00 00 FF 00 00 FF
00000060:	00 00 00 FF FF 00 FF 00	- 00 00 FF 00 FF 00 FF FF
00000070:	00 00 FF FF FF 00 C1 44	- 00 00 00 00 44 1C 9C 11D....D...
00000080:	44 44 44 44 11 C9 99 CC	- 11 11 11 11 CC 99 09 99	DDDD.....
00000090:	CC CC CC CC 99 9D 2D 99	- 99 99 99 99 99 D2 52 D0-.....F.
000000A0:	99 99 99 99 00 25 75 22	- DD DD DD DD 22 57 67 55%u"....*wgb
000000B0:	22 22 22 22 55 76 A6 77	- 55 55 55 55 77 6A EA 66	""""Uv.wUUUW).f
000000C0:	77 77 77 77 66 AE BE AA	- 66 66 66 66 AA EB BB EE	wwwwf...ffff....
000000D0:	AA AA AA AA EE 86 FB 88	- EE EE EE EE 88 BF 0F 88
000000E0:	88 88 88 88 88 F0 40 FF	- 88 88 88 88 FF 04 14 00e.....
000000F0:	FF FF FF FF 00 41 C1 44	- 00 00 00 00 44 1C 9C 11A.D....D...
00000100:	44 44 44 44 11 C9 99 CC	- 11 11 11 11 CC 99 09 99	DDDD.....
00000110:	CC CC CC CC 99 9D 2D 99	- 99 99 99 99 99 D2 52 D0-.....F.
00000120:	99 99 99 99 00 25 75 22	- DD DD DD DD 22 57 67 55%u"....*wgb
00000130:	22 22 22 22 55 76 A6 77	- 55 55 55 55 77 6A EA 66	""""Uv.wUUUW).f
00000140:	77 77 77 77 66 AE BE AA	- 66 66 66 66 AA EB BB EE	wwwwf...ffff....
00000150:	AA AA AA AA EE 86 FB 88	- EE EE EE EE 88 BF 0F 88
00000160:	88 88 88 88 88 F0 00 FF	- 88 88 88 88 FF 00 00 00
00000170:	FF FF FF FF 00 00 40 19	- 6A 18 5D 16 4E 2D 00 00e.).].N-..

Archivo BMP 24 bits(374 bytes):

Dist Editor
6.01
Wednesday, May 24, 1995 12:00 pm

C:\WINDOWS\p24b.bmp
#####

Cluster 3,767, Sector 50,504

00000000:	42 4D 7E 01 00 00 00 00	- 00 00 7b 00 00 00 2B 00	BMP.....v....
00000010:	00 00 10 00 00 00 20 00	- 00 00 01 00 04 00 00 00
00000020:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000030:	00 00 00 00 00 00 00 00	- 00 00 00 00 80 00 00 80
00000040:	00 00 00 80 80 00 80 00	- 00 00 80 00 80 00 80 80
00000050:	00 00 80 80 80 00 C0 C0	- C0 00 00 00 FF 00 00 FF
00000060:	00 00 00 FF FF 00 FF 00	- 00 00 FF 00 FF 00 FF FF
00000070:	00 00 FF FF FF 00 C1 44	- 00 00 00 00 44 1C 9C 11B....D...
00000080:	44 44 44 44 11 C9 99 CC	- 11 11 11 11 CC 99 09 99	GGGG.....
00000090:	CC CC CC CC 99 9D 2D 99	- 99 99 99 99 99 D2 52 DD-.....h.
000000A0:	99 99 99 99 DD 25 75 22	- DD DD DD DD 22 57 67 55%u"...."Wgu
000000B0:	22 22 22 22 55 76 A6 77	- 55 55 55 55 77 6A EA 66	"*"Uv.WUUULw;.t
000000C0:	77 77 77 77 66 AE BE AA	- 66 66 66 66 AA EB 6B EE	wwwwf...ffff...
000000D0:	AA AA AA AA EE 8B FB 8B	- EE EE EE EE 8B BF 0F 8B
000000E0:	8B 8B 8B 8B 8B F0 40 FF	- 8B 8B 8B 8B FF 04 14 00@.....
000000F0:	FF FF FF FF 00 41 C1 44	- 00 00 00 00 44 1C 9C 11A.D....E...
00000100:	44 44 44 44 11 C9 99 CC	- 11 11 11 11 CC 99 09 99	GGGG.....
00000110:	CC CC CC CC 99 9D 2D 99	- 99 99 99 99 99 D2 52 DD-.....h.
00000120:	99 99 99 99 DD 25 75 22	- DD DD DD DD 22 57 67 55%u"...."Wgu
00000130:	22 22 22 22 55 76 A6 77	- 55 55 55 55 77 6A EA 66	"*"Uv.WUUULw;.t
00000140:	77 77 77 77 66 AE BE AA	- 66 66 66 66 AA EB 6B EE	wwwwf...ffff...
00000150:	AA AA AA AA EE 8B FB 8B	- EE EE EE EE 8B BF 0F 8B
00000160:	8B 8B 8B 8B 8B F0 00 FF	- 8B 8B 8B 8B FF 00 00 00
00000170:	FF FF FF FF 00 00 40 19	- 6A 1B 5D 16 4E 2D 00 00@.j.j.h...

Archivo BMP 256 colores(1590 bytes):

Disk Editor
6.01
Wednesday, May 24, 1995 12:00 pm

C:\WINDOWS\p256c.bmp
#####

Cluster 3,761, Sector 60,508

00000000:	42 4B 36 06 00 00 00 00	- 00 00 36 04 00 00 28 00	BMP.....6...(.
00000010:	00 00 10 00 00 00 20 00	- 00 00 01 00 08 00 00 00
00000020:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000030:	00 00 00 00 00 00 00 00	- 00 00 00 00 80 00 00 80
00000040:	00 00 00 80 80 00 80 00	- 00 00 80 00 80 00 80 80
00000050:	00 00 80 80 80 00 C0 C0	- C0 00 00 00 FF 00 00 FF
00000060:	00 00 00 FF FF 00 FF 00	- 00 00 FF 00 FF 00 FF FF
00000070:	00 00 FF FF FF 00 00 00	- 00 00 00 00 00 00 00 00
00000080:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000090:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000000A0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000000B0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000000C0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000000D0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000000E0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000000F0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000100:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000110:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000120:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000130:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000140:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000150:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000160:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000170:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000180:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000190:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000001A0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000001B0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000001C0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000001D0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000001E0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000001F0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00

Cluster 3,761, Sector 60,509

00000200:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000210:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000220:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000230:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000240:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000250:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000260:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000270:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000280:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
00000290:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000002A0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000002B0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000002C0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00
000002D0:	00 00 00 00 00 00 00 00	- 00 00 00 00 00 00 00 00

```

000002E0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000002F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000300: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000310: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000320: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000330: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000340: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000350: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000360: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000370: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000380: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000390: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000003A0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000003B0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000003C0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000003D0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000003E0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
000003F0: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....

```

Cluster 3,761, Sector 60,510

```

00000400: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000410: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000420: 00 00 00 00 00 00 00 00 - 00 00 00 00 00 00 00 00 .....
00000430: 00 00 00 00 00 00 00 01 - 04 04 00 00 00 00 00 00 .....
00000440: 00 00 04 04 01 00 09 00 - 01 01 04 04 04 04 04 04 .....
00000450: 04 04 01 01 00 09 09 09 - 00 00 01 01 01 01 01 01 .....
00000460: 01 01 00 00 09 09 00 09 - 09 09 00 00 00 00 00 00 .....
00000470: 00 00 09 09 09 00 02 00 - 09 09 09 09 09 09 09 09 .....
00000480: 09 09 09 09 00 02 05 02 - 00 00 09 09 09 09 09 09 .....
00000490: 09 09 00 00 02 05 07 05 - 02 02 00 00 00 00 00 00 .....
000004A0: 00 00 02 02 05 07 06 07 - 05 05 02 02 02 02 02 02 .....
000004B0: 02 02 05 05 07 06 0A 06 - 07 07 05 05 05 05 05 05 .....
000004C0: 05 05 07 07 06 0A 0E 0A - 06 06 07 07 07 07 07 07 .....
000004D0: 07 07 06 06 0A 0E 0B 0E - 0A 0A 06 06 06 06 06 06 .....
000004E0: 06 06 0A 0A 0E 0B 0B 0B - 0E 0E 0A 0A 0A 0A 0A 0A .....
000004F0: 0A 0A 0E 0E 0B 0B 0F 0B - 0B 0B 0E 0E 0E 0E 0E 0E .....
00000500: 0E 0E 0B 0B 0B 0F 00 0F - 0B 0B 0B 0B 0B 0B 0B 0B .....
00000510: 0B 0B 0B 0B 0F 00 04 00 - 0F 0F 0B 0B 0B 0B 0B 0B .....
00000520: 0B 0B 0F 0F 00 04 01 04 - 00 00 0F 0F 0F 0F 0F 0F .....
00000530: 0F 0F 00 00 04 01 00 01 - 04 04 00 00 00 00 00 00 .....
00000540: 00 00 04 04 01 00 09 00 - 01 01 04 04 04 04 04 04 .....
00000550: 04 04 01 01 00 09 09 09 - 00 00 01 01 01 01 01 01 .....
00000560: 01 01 00 00 09 09 00 09 - 09 09 00 00 00 00 00 00 .....
00000570: 00 00 09 09 09 00 02 00 - 09 09 09 09 09 09 09 09 .....
00000580: 09 09 09 09 00 02 05 02 - 00 00 09 09 09 09 09 09 .....
00000590: 09 09 00 00 02 05 07 05 - 02 02 00 00 00 00 00 00 .....
000005A0: 00 00 02 02 05 07 06 07 - 05 05 02 02 02 02 02 02 .....
000005B0: 02 02 05 05 07 06 0A 06 - 07 07 05 05 05 05 05 05 .....
000005C0: 05 05 07 07 06 0A 0E 0A - 06 06 07 07 07 07 07 07 .....
000005D0: 07 07 06 06 0A 0E 0B 0E - 0A 0A 06 06 06 06 06 06 .....
000005E0: 06 06 0A 0A 0E 0B 0B 0B - 0E 0E 0A 0A 0A 0A 0A 0A .....
000005F0: 0A 0A 0E 0E 0B 0B 0F 0B - 0B 0B 0E 0E 0E 0E 0E 0E .....

```

Cluster 3,761, Sector 60,511

```

00000600: 0E 0E 0B 0B 0B 0F 00 0F - 0E 0E 0B 0B 0B 0B 0B 0B .....
00000610: 0B 0B 0B 0B 0F 00 00 00 - 0F 0F 0B 0B 0B 0B 0B 0B .....
00000620: 0B 0B 0F 0F 00 00 00 00 - 00 00 0F 0F 0F 0F 0F 0F .....
00000630: 0F 0F 00 00 00 00 00 01 - 04 04 00 00 00 00 00 00 .....

```

SECCIÓN 11F: MÉTODOS DIGITALES PARA TRANSMITIR LA INFORMACIÓN DE TELEVISIÓN

RECOMENDACIÓN 601-2

PARÁMETROS DE CODIFICACIÓN DE TELEVISIÓN DIGITAL PARA ESTUDIOS*

(Cuestión 25/11, Programas de Estudios 25G/11, 25H/11)

(1982-1986-1990)

El CCIR,

CONSIDERANDO

- a) que existen claras ventajas para los organismos de radiodifusión y productores de programas de televisión en la existencia de normas de televisión digital para los estudios, de forma que tengan el mayor número de valores de parámetros significativos comunes a los sistemas de 525 y 625 líneas;
- b) que un método digital compatible en todo el mundo posibilitará el desarrollo de equipos con muchas características comunes, permitirá economías de explotación y facilitará el intercambio internacional de programas;
- c) que conviene disponer de una familia ampliable de normas compatibles de codificación digital; los niveles de dicha familia podrían corresponder a diferentes niveles de calidad, facilitar el tratamiento adicional requerido por las actuales técnicas de producción y tener en cuenta las necesidades futuras;
- d) que un sistema basado en la codificación de las componentes puede satisfacer algunos de estos objetivos deseables, o su totalidad;
- e) que la coincidencia en el espacio de las muestras que representan las señales de luminancia y de diferencia de color (o, en su caso, las señales de rojo, verde y azul) facilita el tratamiento de las señales componentes digitales que requieren las actuales técnicas de producción,

RECOMIENDA, POR UNANIMIDAD:

Que se empleen los siguientes parámetros como base para las normas de codificación digital para estudios de televisión en aquellas zonas del mundo que utilizan tanto sistemas de 525 líneas como de 625 líneas:

1. Codificación de las componentes

La codificación digital debe basarse en el empleo de una señal de luminancia y de dos señales de diferencia de color (o, en su caso, señales de rojo, verde y azul).

Deben controlarse las características espectrales de las señales para evitar el solape de los diferentes espectros, conservando al mismo tiempo la respuesta en la banda de paso. Cuando se utiliza una señal de luminancia y dos de diferencia de color como se define en el cuadro I, puede lograrse este objetivo empleando filtros apropiados, como los que se definen en las figs. 1 y 2, del anexo III. Cuando se utilizan señales E'_R , E'_G , E'_B o señales de luminancia y de diferencia de color según se define en el cuadro II del anexo I, la característica adecuada es la definida en la fig. 1 del anexo III.

Los principales términos sobre televisión digital utilizados en esta Recomendación están definidos en el Informe 629.

2. Familia ampliable de normas compatibles de codificación digital

La codificación digital debe permitir el establecimiento y la evolución de una familia ampliable de normas compatibles de codificación digital.

Debe permitir el interfaz simple entre dos niveles cualesquiera de la familia.

El nivel de la familia que ha de utilizarse para el interfaz digital normalizado entre el equipo digital principal de estudio y para el intercambio internacional de programas (es decir, para el interfaz con los magnetoscopios y para el interfaz con los equipos de transmisión) debe ser aquel en que las frecuencias de muestreo de las señales de luminancia y de diferencia de color tienen una relación de 4 : 2 : 2.

En un posible nivel más alto de la familia, las frecuencias de muestreo de las señales de luminancia y de diferencia de color (o, en su caso, las señales de rojo, verde y azul) podrían tener una relación de 4 : 4 : 4. En el anexo I figuran especificaciones provisionales para el nivel con una relación de 4 : 4 : 4 (véase la nota).

Nota — Se ruega a las administraciones que continúen con toda urgencia los estudios para especificar los parámetros de las normas digitales para otros niveles de la familia. Debe darse prioridad a los niveles de la familia por debajo de 4 : 2 : 2. El número de normas adicionales especificadas debe mantenerse en el mínimo.

3. Especificaciones aplicables a cualquier nivel de la familia

3.1 Las estructuras de muestreo deben ser estáticas en el espacio. Este es el caso, por ejemplo, de la estructura de muestreo ortogonal especificada en el punto 4 a continuación de la presente Recomendación para el nivel de la familia de relación 4 : 2 : 2.

3.2 Si las muestras representan la señal de luminancia y dos señales simultáneas de diferencia de color, cada par de muestras de diferencia de color deben tener igual ubicación en el espacio. Si se utilizan muestras que representan las señales de rojo, verde y azul, deben tener igual ubicación en el espacio.

3.3 La norma digital adoptada para cada nivel de la familia debe hacer posible la aceptación y aplicación a nivel mundial en la explotación; una condición para lograr este objetivo es que, en cada nivel de la familia, el número especificado de muestras por línea para los sistemas de 525 y 625 líneas sea compatible (De ser posible, conviene que haya un número igual de muestras por línea.)

4. Valores de los parámetros de codificación para el nivel de la familia de relación 4 : 2 : 2

La siguiente especificación (cuadro I) es aplicable a nivel de la familia de relación 4 : 2 : 2, que ha de utilizarse para el interfaz digital normalizado entre los equipos digitales principales de estudio y para el intercambio internacional de programas.

CUADRO I - Valores de los parámetros de codificación para el nivel de la familia de relación 4 : 2 : 2

Parámetros	Sistemas de 525 líneas ⁽¹⁾ 60 tramas/s	Sistemas de 625 líneas ⁽¹⁾ 50 tramas/s
1. Señales codificadas: Y, C_R, C_B	Estas señales se obtienen a partir de señales con precorrección gamma es decir: $E'_Y, E'_R - E'_Y, E'_B - E'_Y$ (véase el anexo II, § 2)	
2. Número de muestras por línea completa: - Señal de luminancia (Y) - Cada señal de diferencia de color (C_R, C_B)	858 429	864 432
3. Estructura de muestreo	Ortogonal, repetitiva en cada línea, en cada trama y en cada imagen. Las muestras de las señales C_R y C_B coinciden en el espacio con las muestras impares (1.ª, 3.ª, 5.ª, etc.) de la señal Y en cada línea	
4. Frecuencia de muestreo: - Señal de luminancia - Cada señal de diferencia de color	13,5 MHz ⁽²⁾ 6,75 MHz ⁽²⁾ La tolerancia para las frecuencias de muestreo debe coincidir con la tolerancia para las frecuencias de línea de la norma de televisión en color que corresponda	
5. Forma de codificación	MIC con cuantificación uniforme, 8 bits por muestra, tanto para la señal de luminancia como para las señales de diferencia de color	
6. Número de muestras por línea activa digital: - Señal de luminancia - Cada señal de diferencia de color	720 360	
7. Relación de temporización horizontal analógico-digital: - Desde el final de la línea activa digital hasta 0_H	16 periodos del reloj de luminancia	12 periodos del reloj de luminancia
8. Correspondencia entre los niveles de la señal de video y los niveles de las señales de cuantificación: - Escala - Señal de luminancia - Cada señal de diferencia de color	0 a 255 220 niveles de cuantificación; el nivel de negro corresponde al nivel 16 y el nivel de blanco de cresta corresponde al nivel 235. El nivel de la señal puede ocasionalmente sobrepasar el nivel 235 225 niveles de cuantificación en la parte central de la escala de cuantificación; la señal cero corresponde al nivel 128	
9. Uso de palabras de código	Las palabras de código que corresponden a los niveles de cuantificación 0 y 255 se utilizan exclusivamente para sincronización. Los niveles 1 a 254 están disponibles para video	

⁽¹⁾ Véase el cuadro I del Informe 624.

⁽²⁾ Las frecuencias de muestreo a 13,5 MHz (luminancia) y a 6,75 MHz (diferencia de color) son múltiplos enteros de la frecuencia 2,25 MHz, mínimo común múltiplo de las frecuencias de línea en los sistemas de 525/60 y 625/50 líneas, lo que da lugar a un diagrama de muestreo ortogonal estático para ambas.

ANEXO I

ESPECIFICACIÓN PROVISIONAL PARA EL NIVEL DE LA FAMILIA DE RELACIÓN 4:4:4

En el presente anexo figura una especificación provisional para el nivel de la familia de normas de codificación digital de relación 4:4:4.

La siguiente especificación podría aplicarse al nivel de la familia de relación 4:4:4 adecuado para equipos de fuente de señales de televisión y aplicaciones de tratamiento de señales de video de alta calidad.

CUADRO II – Especificación provisional para el nivel de la familia de relación 4:4:4

Parámetros	Sistemas de 525 líneas 60 tramas/s	Sistemas de 625 líneas 50 tramas/s
1. Señales codificadas: Y, C_R, C_B o R, G, B	Estas señales se obtienen a partir de señales con corrección gamma, es decir: $E'_Y, E'_R - E'_Y, E'_B - E'_Y$ o E'_R, E'_G, E'_B	
2. Número de muestras por línea completa para cada señal	858	864
3. Estructura de muestreo	Ortogonal, repetitiva en cada línea, en cada trama y en cada imagen. Las tres estructuras de muestreo deben ser coincidentes tanto entre sí como con la estructura de muestreo de luminancia del nivel de relación 4:2:2	
4. Frecuencia de muestreo para cada señal	13,5 MHz	
5. Forma de codificación	MIC con cuantificación uniforme: al menos 8 bits por muestra	
6. Duración de la línea activa digital expresada en número de muestras	Al menos 720	
7. Correspondencia entre los niveles de la señal de video y los 8 bits más significativos del nivel de cuantificación de cada muestra: – Escala – Señales R, G, B o señal de luminancia ⁽¹⁾ – Cada señal de diferencia de color ⁽¹⁾	<p>0 a 255</p> <p>220 niveles de cuantificación, el nivel de negro corresponde al nivel 16 y el nivel de blanco de cresta corresponde al nivel 235. El nivel de la señal puede ocasionalmente sobrepasar el nivel 235</p> <p>225 niveles de cuantificación en la parte central de la escala de cuantificación; la señal cero corresponde al nivel 128</p>	

⁽¹⁾ Si se utiliza.

ANEXO II

DEFINICIÓN DE LAS SEÑALES UTILIZADAS EN LAS NORMAS DE CODIFICACIÓN DIGITAL

Relación entre la línea activa digital y la referencia analógica de sincronismo

Se muestra seguidamente la relación entre las 720 muestras de luminancia de la línea activa digital y la referencia analógica de sincronismo para sistemas de 625 y 525 líneas.

CUADRO III

Sistemas de 525 líneas 60 tramas/s	122 T	720 T	16 T
0_H (flanco anterior de los sincronismos de línea, referencia amplitud mitad)		Periodo de línea activa digital	Línea siguiente 0_H
Sistemas de 625 líneas 50 tramas/s	132 T	720 T	12 T

T : Periodo del reloj de muestreo de luminancia (74 ns nominales).

Los números respectivos de muestras de diferencia de color se pueden obtener dividiendo el número de muestras de luminancia por 2. Los números (12, 132) y (16, 122) se escogieron para disponer simétricamente la línea activa digital con respecto a las variaciones permitidas. No forman parte de la especificación de la línea digital y sólo se refieren a interfaces analógicos.

Definición de las señales digitales Y , C_R , C_B , partiendo de las señales (analógicas) primarias E'_R , E'_G y E'_B

Para definir las señales Y , C_R , C_B , se describen aquí las reglas de construcción de esas señales partiendo de las señales analógicas primarias E'_R , E'_G , E'_B . Esta construcción encadena las tres etapas descritas más adelante en los § 2.1, 2.2 y 2.3, y se da a título de ejemplo. En la realización práctica hay otros métodos de construcción que, aplicados a esas señales primarias o a otras señales analógicas o digitales, pueden conducir a resultados idénticos. En el § 2.4 se da un ejemplo.

1 Construcción de las señales de luminancia (E'_Y) y de diferencia de color ($E'_R - E'_Y$) y ($E'_B - E'_Y$)

La construcción de las señales de luminancia y de diferencia de color es la siguiente:

$$E'_Y = 0,299E'_R + 0,587E'_G + 0,114E'_B \quad (\text{Véase la nota})$$

onde:

$$\begin{aligned} (E'_R - E'_Y) &= E'_R - 0,299E'_R - 0,587E'_G - 0,114E'_B \\ &= 0,701E'_R - 0,587E'_G - 0,114E'_B \end{aligned}$$

$$\begin{aligned} (E'_B - E'_Y) &= E'_B - 0,299E'_R - 0,587E'_G - 0,114E'_B \\ &= -0,299E'_R - 0,587E'_G + 0,886E'_B \end{aligned}$$

ota — Véase el Informe 624, cuadro II.

Suponiendo que los valores de las señales están normalizados a la unidad (ejemplo, niveles máximos de 1,0 V), los valores que se obtienen para los colores blanco, negro, los colores primarios saturados y sus complementarios son los siguientes:

CUADRO IV

Condición	E'_R	E'_G	E'_B	E'_Y	$E'_R - E'_Y$	$E'_B - E'_Y$
Blanco	1,0	1,0	1,0	1,0	0	0
Negro	0	0	0	0	0	0
Rojo	1,0	0	0	0,299	0,701	-0,299
Verde	0	1,0	0	0,587	-0,587	-0,587
Azul	0	0	1,0	0,114	-0,114	0,886
Amarillo	1,0	1,0	0	0,886	0,114	-0,886
Cian	0	1,0	1,0	0,701	-0,701	0,299
Magenta	1,0	0	1,0	0,413	0,587	0,587

2.2 Construcción de las señales de diferencia de color renormalizadas (E'_{C_R} y E'_{C_B})

Aunque los valores de E'_Y están ya comprendidos en la gama de 1,0 a 0, los de $(E'_R - E'_Y)$ se sitúan en la gama de +0,701 a -0,701, y los de $(E'_B - E'_Y)$ en la gama de +0,886 a -0,886. Para renormalizar respecto a la unidad la gama de las señales de diferencia de color (es decir, de +0,5 a -0,5), se pueden calcular los coeficientes siguientes:

$$K_R = \frac{0,5}{0,701} = 0,713; K_B = \frac{0,5}{0,886} = 0,564$$

Entonces:

$$E'_{C_R} = 0,713 (E'_R - E'_Y) = 0,500E'_R - 0,419E'_G - 0,081E'_B$$

y:

$$E'_{C_B} = 0,564 (E'_B - E'_Y) = -0,169E'_R - 0,331E'_G + 0,500E'_B$$

donde E'_{C_R} y E'_{C_B} son las señales renormalizadas de diferencia de color para rojo y azul, respectivamente (véanse las notas 1 y 2).

Nota 1 - Los símbolos (E'_{C_R} y E'_{C_B}) se reservarán para designar las señales de diferencia de color «renormalizadas», es decir, las que tengan una amplitud nominal cresta cresta idéntica a la de la señal de luminancia E'_Y , elegida así como referencia de amplitud.

Nota 2 - En el caso en que las señales de las componentes no estén normalizadas en la gama de 1 a 0 (por ejemplo, cuando se efectúa la conversión a partir de señales componentes analógicas, con amplitudes de luminancia y de diferencia de color desiguales), se necesitará un factor de ganancia adicional y deberán modificarse en consecuencia los factores de ganancia K_R , K_B .

2.3 Cuantificación

En el caso de una codificación binaria de 8 bits con cuantificación uniforme, se especifican 2^8 , es decir, 256 niveles de cuantificación equidistantes, con lo que la gama de números binarios disponibles va de 0000 0000 a 1111 1111 (00 a FF en la notación hexadecimal) siendo los números decimales equivalentes 0 a 255.

En el caso del sistema 4 : 2 : 2 descrito en esta Recomendación, los niveles 0 y 255 se reservan para datos de sincronización, en tanto que los niveles 1 a 254 se utilizan para video.

Dado que la señal de luminancia sólo va a ocupar 220 niveles, para proporcionar márgenes de trabajo y que el negro se encuentre en el nivel 16, el valor decimal de la señal de luminancia, \bar{Y} , antes de la cuantificación es:

$$\bar{Y} = 219 (E'_Y) + 16,$$

y el número de nivel correspondiente, después de la cuantificación, es el número entero más próximo.

Análogamente, como las señales de diferencias de color van a ocupar 225 niveles y como el nivel 0 va a ser el nivel 128, los valores decimales de las señales de diferencia de color, \bar{C}_R y \bar{C}_B , antes de la cuantificación son:

$$\bar{C}_R = 224 [0,713 (E'_R - E'_Y)] + 128$$

y:

$$\bar{C}_B = 224 [0,564 (E'_B - E'_Y)] + 128$$

que simplificando se convierte en:

$$\bar{C}_R = 160 (E'_R - E'_Y) + 128$$

y:

$$\bar{C}_B = 126 (E'_B - E'_Y) + 128$$

y el número de nivel correspondiente, después de la cuantificación, es el número entero más próximo.

Los equivalentes digitales se denominan Y , C_R y C_B .

2.4 Construcción de Y , C_R , C_B mediante la cuantificación de E'_R , E'_G , E'_B

Cuando las componentes se obtienen directamente de las señales componentes con corrección gamma E'_R , E'_G , y E'_B o se generan directamente en forma digital, la cuantificación y la codificación serán equivalentes a:

$$E'_{R_d} \text{ (en forma digital)} = \text{Parte entera de } (219 E'_R) + 16$$

$$E'_{G_d} \text{ (en forma digital)} = \text{Parte entera de } (219 E'_G) + 16$$

$$E'_{B_d} \text{ (en forma digital)} = \text{Parte entera de } (219 E'_B) + 16$$

Por consiguiente:

$$Y = \frac{77}{256} E'_{R_d} + \frac{150}{256} E'_{G_d} + \frac{29}{256} E'_{B_d}$$

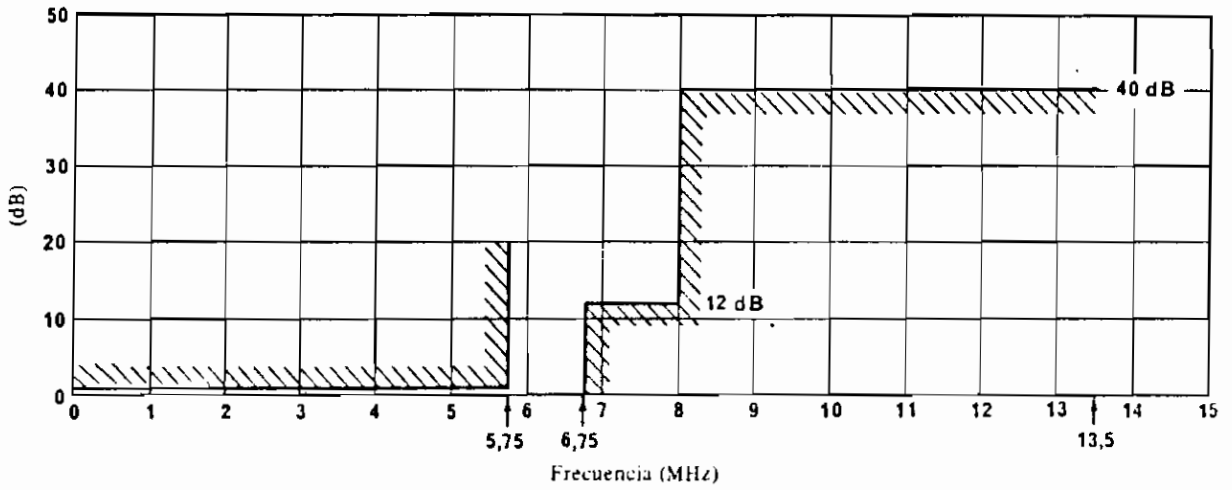
$$C_R = \frac{131}{256} E'_{R_d} - \frac{110}{256} E'_{G_d} - \frac{21}{256} E'_{B_d} + 128$$

$$C_B = -\frac{44}{256} E'_{R_d} - \frac{87}{256} E'_{G_d} + \frac{131}{256} E'_{B_d} + 128$$

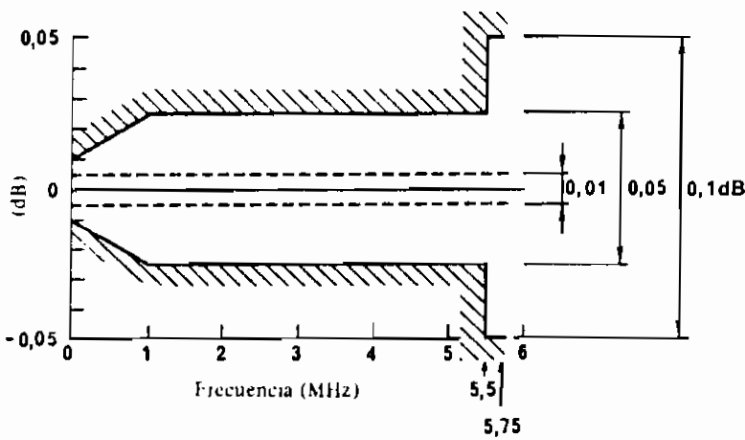
tomando los coeficientes enteros más próximos en base 256. Para obtener las componentes 4 : 2 : 2 Y , C_R , C_B , debe efectuarse el filtrado de paso bajo y el submuestreo en las señales 4 : 4 : 4 C_R , C_B anteriormente descritas. Conviene tomar nota de que podría haber ligeras diferencias entre las componentes C_R , C_B , obtenidas de este modo y las obtenidas por filtrado analógico antes del muestreo.

ANEXO III

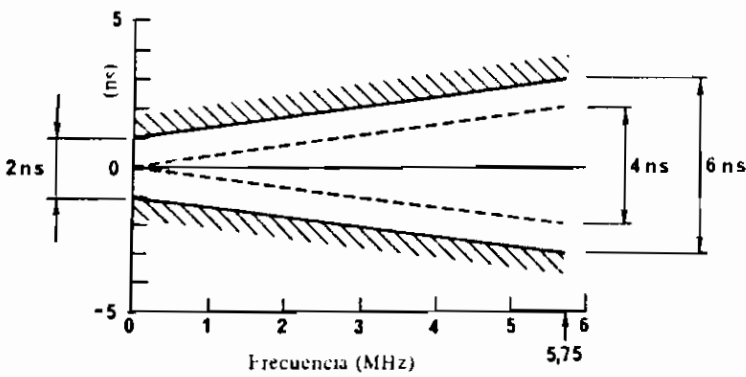
CARACTERÍSTICAS DE FILTRADO



a) Máscara de la característica de pérdida de inserción en función de la frecuencia



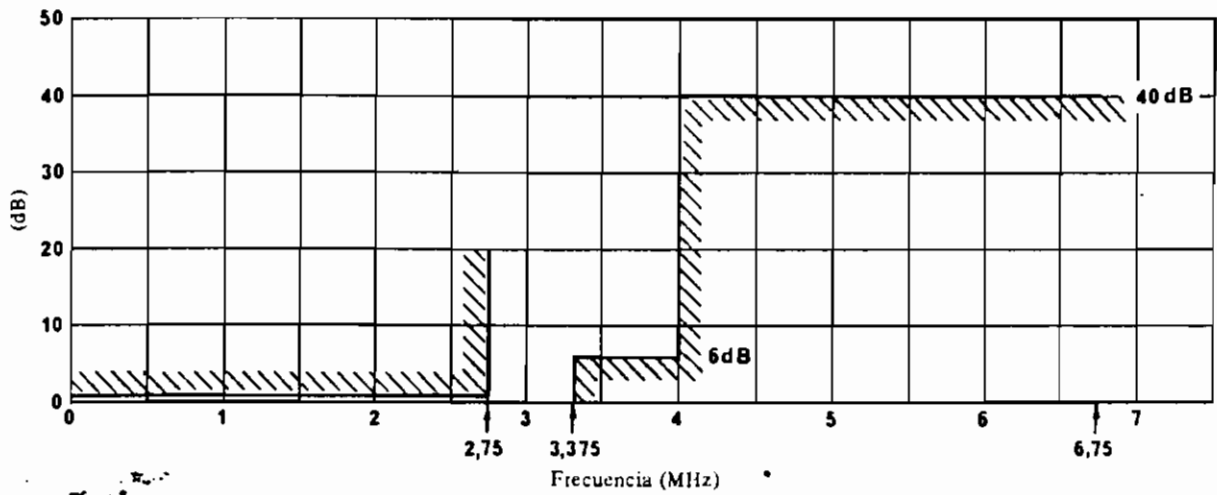
b) Tolerancia para el rizado en la banda de paso



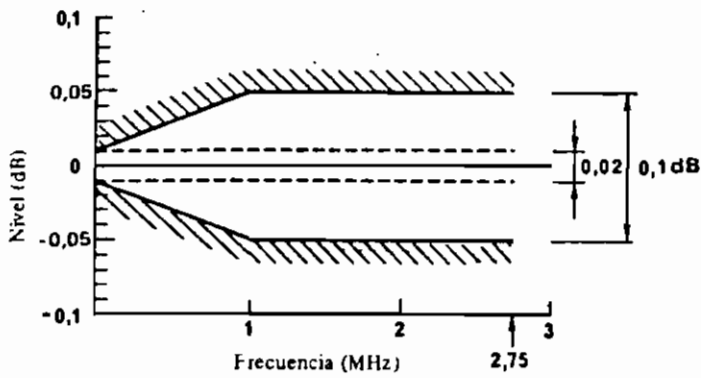
c) Tolerancia para el retardo de grupo en la banda de paso

FIGURA 1 – Especificación de un filtro para las señales de luminancia o RGB utilizado cuando se muestrea a 13,5 MHz

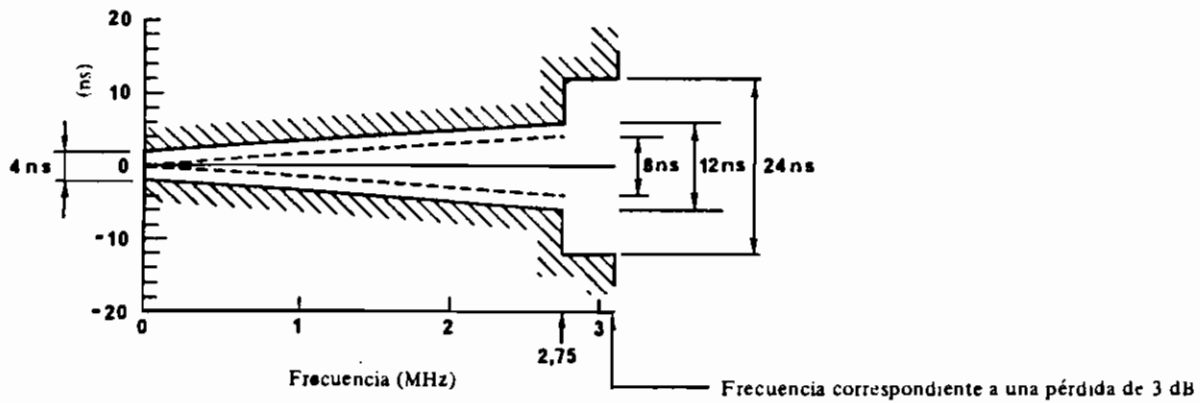
Nota – Los valores más bajos de la escala de ordenadas en las figs. 1b) y 1c) corresponden a 1 kHz (en vez de 0 MHz).



a) Máscara de la característica de pérdida de inserción en función de la frecuencia



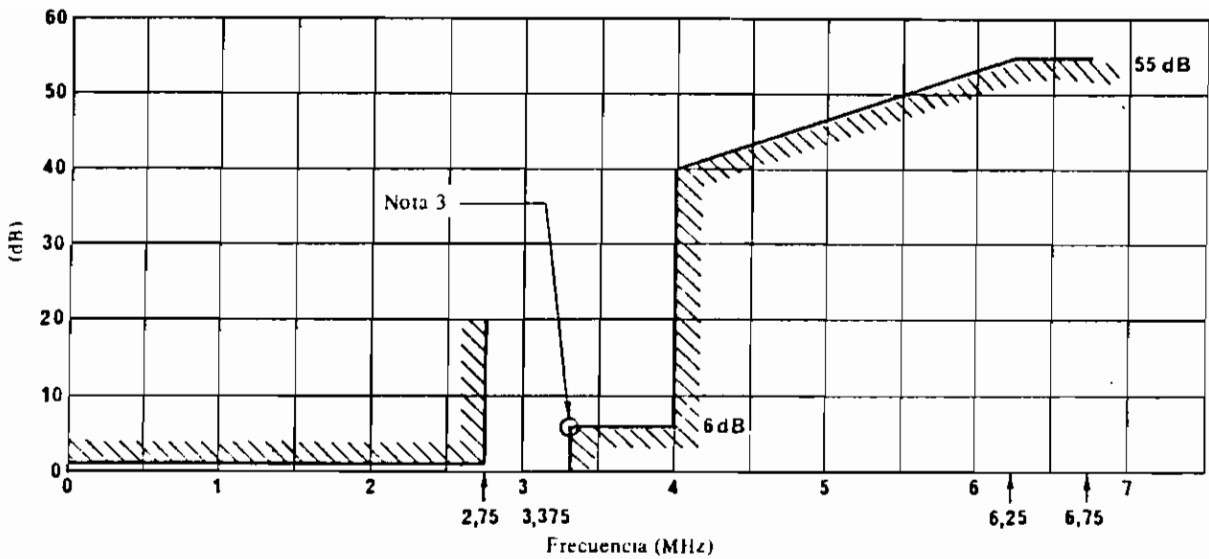
b) Tolerancia para el rizado en la banda de paso



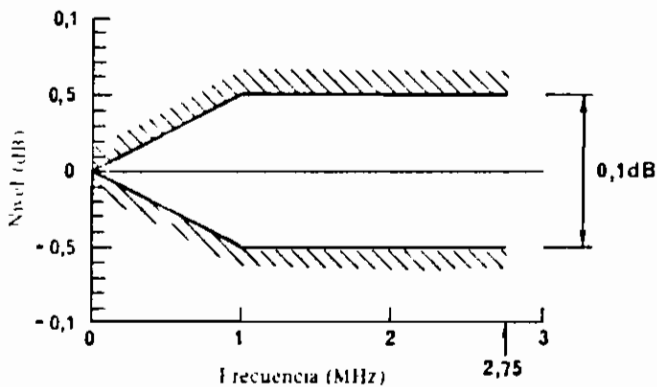
c) Tolerancia para el retardo de grupo en la banda de paso

FIGURA 2 - Especificación de un filtro para la señal de diferencia de color, utilizado cuando se muestrea a 6,75 MHz

Nota - Los valores más bajos de la escala de ordenadas en las figs. 2b) y 2c) corresponden a 1 kHz (en vez de 0 MHz).



a) Máscara de la característica de pérdida de inserción en función de la frecuencia



b) Tolerancia para el rizado en la banda de paso

FIGURA 3 - Especificación de un filtro digital para la conversión de frecuencia de muestreo de señales de diferencia de color de relación 4:4:4 a señales de relación 4:2:2

Notas relativas a las figs. 1, 2 y 3 anteriores:

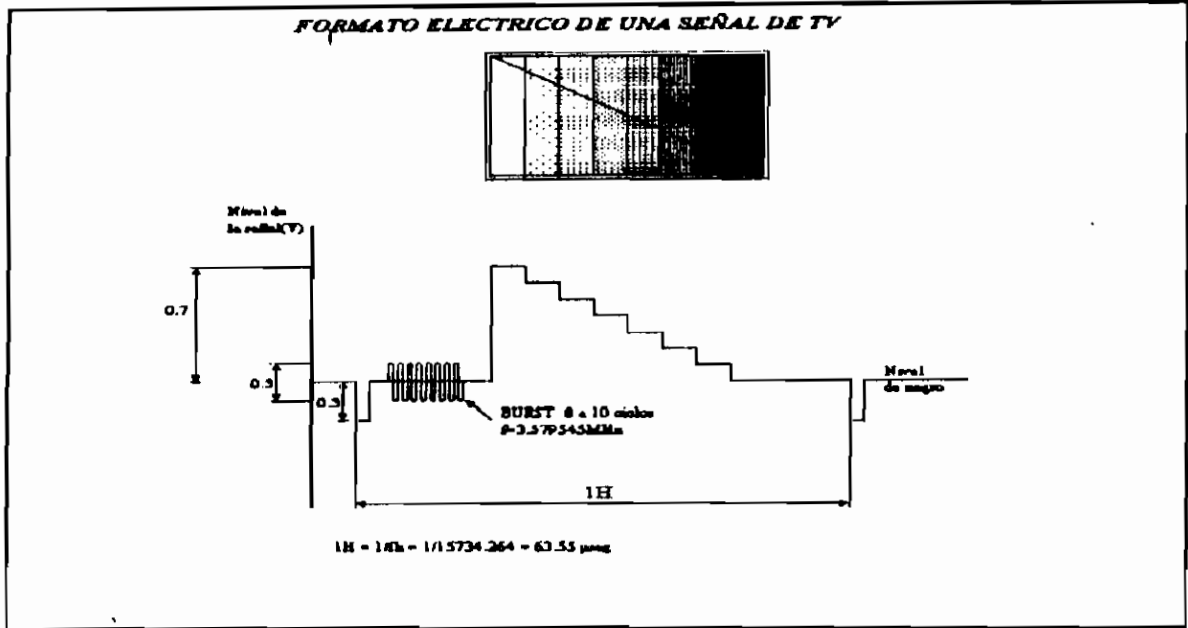
Nota 1 - El rizado y el retardo de grupo se especifican respecto de sus valores a 1 MHz. Las líneas continuas representan los límites prácticos y las líneas de trazo discontinuo indican los límites sugeridos para el diseño teórico.

Nota 2 - En el filtro digital, los límites prácticos y de diseño son los mismos. La distorsión de retardo es cero, por diseño.

Nota 3 - En el filtro digital (fig. 3), la característica de amplitud en función de la frecuencia (en escalas lineales) debe ser asimétrica respecto del punto de amplitud mitad, que se indica en la figura.

Nota 4 - En las propuestas para los filtros utilizados en los procesos de codificación, y decodificación, se ha supuesto que se introduce la corrección de la característica (sen x/x) del circuito de muestreo y retención en los filtros que van a continuación de la conversión digital-analógica.

FORMATO ELECTRICO DE UNA SEÑAL DE TV



La figura representa a una línea barrida. En general el sistema NTSC presenta las siguientes características:

Número de líneas barridas por cuadro: 525

Número de cuadros por segundo: 30

Número de campos por cuadro: 2

Frecuencia de pulsos de sincronismo horizontal: 15734.264Hz

Frecuencia de pulsos de sincronismo vertical: 59.94Hz

Frecuencia de la subportadora de color: 3.579545MHz

Señal total de video: $1V_{p-p}$

Señal de luminancia: Modulada en amplitud (Banda vestigial)

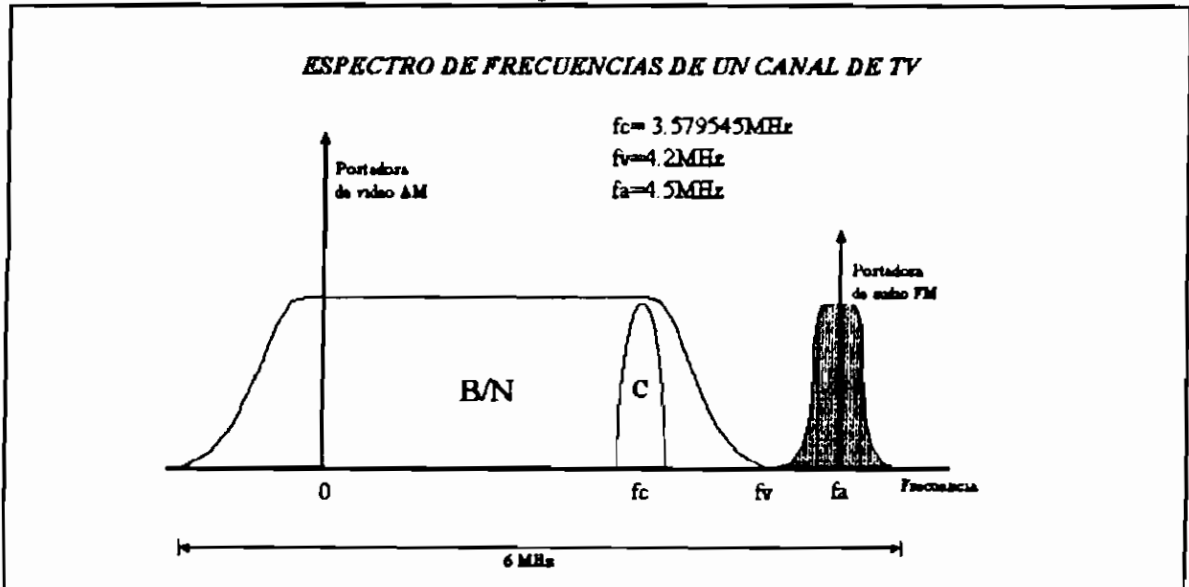
Señal de audio: Modulada en frecuencia.

Ancho de banda total: 6MHz (sistema M de TV B/N)

Ancho de banda efectivo de luminancia: 4.2MHz

Audio: 4.5MHz arriba de la portadora de luminancia

ESPECTRO DE FRECUENCIAS DE UN CANAL DE TV



ADAPTIVO: Criterio aplicado a sistemas de procesamiento de señales según el cual el sistema se debe "adaptar" a los requerimientos de las señales con las que se trabaje, o a los parámetros de estas. Todo esto con el fin de lograr la mayor eficiencia posible del proceso.

AUTO REGRESIVE(AR): Un tipo de proceso de estimación de señales, utilizado en formas de filtro AR o predictor AR.

AUTO REGRESIVE MOVING AVERAGE(ARMA): Una combinación de dos procesos de estimación de señales, del AR y del MA.

BIT ERROR RATE(BER): Tasa de bits erróneos, número de bits errados de un total.

BIT RATE: Tasa de bits de los bits por pixel o bits por segundo, según si la aplicación es la medida de entropía o velocidad binaria.

BLOCK TRUNCATION CODING(BTC): Codificación por truncamiento de bloque(CTB).

C(Croma): Señal eléctrica que representa a la característica colorida de una imagen de televisión.

CIF(Common Intermediate Format): Formato Intermedio Común.

COMB FILTER: Estructura electrónica que consiste de un puente de 4 resistencias y 3 cristales alineados, con el propósito de separar la señal de crominancia(C) de las señales de interferencia y de inducción magnética (CrossTalk) en el caso de la reproducción y se encarga de separar la señal de luminancia(Y) de la señal de color(C) en grabación. Esto en el proceso del video en cintas magnéticas convencionales de un VCR (VHS, β max, etc.).

CONSULTIVE COMITEE INTERNATIONAL of TELEFONY and TELEGRAFY-(CCITT): Comité consultivo internacional de telefonía y telegrafía.

CONSULTIVE COMITEE INTERNATIONAL of RADIO(CCIR): Comité consultivo internacional de radio.

COMPANDING: La acción de realizar un proceso de compresión y luego uno de descompresión.

DISCRETE HARTLEY TRANSFORM(DHT): Transformada discreta de Hartley.

DISCRETE COSINE TRANSFORM(DCT): Transformada discreta coseno.

DISCRETE CAS-CAS TRANSFORM(DCCT): Transformada discreta Cas-Cas.

DISCRETE FOURIER TRANSFORM(DFT): Transformada discreta de Fourier.

DISCRETE FAST FOURIER TRANSFORM(DFFT): Transformada discreta rápida de Fourier.

DIFERENTIAL PULSE CODE MODULATION(DPCM): MIC Diferencial.

DELTA MODULATION(DM): Modulación delta.

DIFERENTIAL PREDICTIVE QUANTIZING(DPQ): Cuantificación diferencial predictiva(CDP).

DOWN SAMPLING: Sub-muestreo, el ancho de banda de la señal muestreada se "comprime" por un determinado factor entero.

FRAME: Trama, marco, armadura. En video, una de las dos partes (trama par, trama impar) de las que está constituido un campo.

FIELD: Campo, fondo, cuadro. En video, una imagen completa que se forma a determinada frecuencia (50Hz o 60Hz según el esquema televisivo estándar).

GOP(Group Of Pictures): Trama de un determinado grupo de imágenes, las cuales pueden ser I(Entrantes), P(Precedentes), o B(Bidireccionales).

IMAGEN DATA COMPRESION: Compresión de imágenes.

JOINT PHOTOGRAPHIC EXPERTS GROUP(JPEG): Grupo conjunto de expertos fotográficos (Imágenes fijas).

KERNEL: Semilla, germen, originador.

MERGE & NEW START: Fusión y Nuevo Inicio.

MOVING AVERAGE(MA): Un tipo de proceso de estimación de señales, utilizado en formas de filtro de Media Móvil(MA) o predictor Media Móvil(MA).

MOVING PICTURES EXPERTS GROUP(MPEG): Grupo de expertos en imágenes móviles.

M.U.S.E.: Son las siglas de Codificación por Múltiple Muestreo Sub-Nyquist y es un esquema de compresión de ancho de banda propuesto por NHK (Corporación de Radiodifusión de Japón) para mono-canal, en la radiodifusión directa por satélite de señales de HDTV 1125/60/2:1 en la banda de los 12GHz.

NATIONAL TELEVISION SYSTEM COMMITTEE(NTSC): Esquema estandarizado de transmisión de señales de televisión analógica. Su formato consiste 30 cuadros por segundo, cada cuadro consta de 2 campos (impar y par), y con rastreo entrelazado de 525 líneas horizontales.

PICTURE ELEMENT(PIXEL o PEL): Unidad de información mínima luminosa en los formatos de imágenes. Parte mínima de una imagen.

PHASE ALTERNATING LINE(PAL): (Líneas de Fase Alternativa) Esquema estandarizado de transmisión de señales de televisión analógica. Su formato consiste 25 cuadros por segundo, cada cuadro consta de 2 campos (impar y par), y con rastreo entrelazado de 625 líneas horizontales. Existen además variantes de PAL, el PAL-M y el PAL-N.

PULSE CODE MODULATION(PCM): Modulación de impulsos codificados(MIC).

QCIF(Quarter CIF): Formato Cuádruple Intermedio Común.

QUANTIZATION: Cuantificación.

RAW: Apócope de señal "CRUDA" es decir en el caso de una imagen, la que no ha sido todavía procesada, es decir sus píxeles digitales están en la representación binaria como son obtenidos luego de por ejemplo un "scanning" o exploración.

RELATIVE ADDRESS CODING(RAC): Codificación de la dirección relativa(CDR).

RELATIVE ELEMENT ADDRESS DESIGNATION(READ) CODING: Codificación por designación de la dirección relativa del elemento.

RED, GREEN, BLUE(RGB): Colores primarios Rojo, Verde y Azul.

RUN LENGTH CODING(RLC): El término inglés de la Codificación Longitudinal o CL.

SIF(Source Input Format): Formato Entrante de Fuente.

SIGNAL TO NOISE RATIO(SNR): Relación señal a ruido(SQR).

SEQUENTIEL Á MEMOIRE(SECAM): Es un sistema de televisión a color de origen francés y posee las mismas bases teóricas que el NTSC pero elimina la necesidad de una elevada sensibilidad de fase.

UP SAMPLING: Super-muestreo, el ancho de banda de la señal muestreada se "expande" por un determinado factor entero.

TWO TONE IMAGES: Imágenes a dos tonalidades (Blanco/Negro).

VARIABLE LENGTH CODING: Codificación por longitud variable(CLV).

VIDEO GRAPHICS ADAPTER(VGA): Modelo de resolución de pantallas de computador cuya resolución mínima es de 640x480 píxeles.

VIDEO COMPUESTO: Es el patrón de representar a una imagen de televisión en donde la imagen eléctricamente representa a la suma de la señal de luminancia(**Y**) más el vector de croma(**C**).

WHITE BLOCK SKIPING(WBS) CODING: Codificación por omisión de bloques blancos.

Y(Luminancia): Señal eléctrica que representa a los niveles de brillo (de gris) de una señal de televisión.

ANEXO-6

ESTRUCTURA DE UN PROTOTIPO PARA LA COMPRESION-DESCOMPRESION DE IMAGENES

INTRODUCCION:

La necesidad de incluir este anexo se debe a que en la práctica, un equipo realizable es una conclusión lógica de un adecuado estudio teórico. A continuación se verán expuestos paso a paso las partes físicas que podrían conformar a un práctico "Compresor-Descompresor de Imágenes", esto, con el afán de que todos los conceptos manejados en este trabajo en este libro (y además tópicos más amplios como la electrónica aplicada a las comunicaciones) tengan un fin práctico. El estudio de este prototipo será una conclusión de todos ellos.

ESTUDIO DEL SISTEMA:

El problema se puede definir de la siguiente manera: "SE REQUIERE DE UN SISTEMA CAPAZ DE CAPTAR SENALES DE VIDEO EN FORMA DIRECTA, ADAPTAR ESTAS SENALES A UN MEDIO DE TRANSMISION DEFINIDO, TRANSMITIR ESTAS SENALES A UN PUNTO RECEPTOR DADO VIA ESTE MEDIO DE TRANSMISION, DECODIFICAR ESTE FLUJO DE INFORMACION Y PRESENTAR EN UN TERMINAL ADECUADO LAS IMAGENES CAPTADAS CON EL EQUIPO TRANSMISOR".

Se puede notar que se está hablando de una comunicación SIMPLEX (que se pudiera explotar a un sistema SEMI DUPLEX o DUPLEX a corto plazo con la implementación de dos prototipos). Es obvia la necesidad de componer al prototipo de dos equipos uno en el terminal transmisor para capturar imágenes, procesarlas y adaptarlas al medio de transmisión y otro equipo localizado en el terminal receptor que produzca un efecto contrario, obtenga del canal las imágenes transmitidas (que necesariamente estarán en un formato distinto al visual), las "reconstruya" y luego las "presente" en el formato visual en el que fueron capturadas. Se habla de UN MEDIO DE TRANSMISION DEFINIDO, que pudiese ser el par telefónico (resultando entonces en un equipo muy popular y comercial), una línea coaxial (para aplicaciones específicas), una línea de comunicación basada en fibra óptica, y por último, por el medio atmosférico, una radiodifusión electromagnética.

Pese a que la decisión del medio de transmisión a utilizar es dependiente de la aplicación a la que se orientará la comunicación, en el siguiente ejemplo se opta por el

medio atmosférico por constituir el medio más económico. En efecto, al utilizar la atmósfera como *línea de transmisión* se economiza el equipo considerablemente.

Básicamente en los capítulos 3 y 4 se han dado ya los mayores criterios de diseño, es más, la Figura 3.3 es ya una tentativa de construcción y la figura 3.4 deja satisfecho al lector acucioso cuando necesita conocer la aplicación práctica de un equipo de comunicación de imágenes. Un diagrama muy generalizado se halla en la figura 4.1 y en la figura 4.5 se tiene el esquema formal de una comunicación digital.

DESCRIPCION DEL EQUIPO:

El diseño se lo realizará por etapas a saber:

- Terminal captador de imágenes-transmisor de señal.
 - a- Captura.
 - b- Digitalización.
 - c- Compresión.
 - d- Codificación.
 - e- Modulación.
 - f- Amplificación de potencia y transmisión.

- Terminal receptor de señal-expositor de imágenes.
 - a- Recepción.
 - b- Amplificación.
 - c- Demodulación.
 - d- Decodificación.
 - e- Descompresión.
 - f- Conversión Digital-Análoga.
 - g- Presentación.

TERMINAL RECEPTOR DE IMAGENES-TRANSMISOR DE SEÑAL:

(a) **CAPTURA:** En este punto lo que se realiza es la transformación de la señal visual de una imagen, a su equivalente eléctrico analógica. Este proceso lo realiza fácilmente una cámara de video y de esta es de donde se toma la señal para comenzar el proceso. La cámara de video capta a las imágenes en su dispositivo sensor (como el CI CCD) y obtiene la señal de video como la del Anexo 4 que representa a una línea barrida. Se supondrá que se trabaja con el esquema NTSC

de televisión a color cuyas características se muestran en el Anexo 4. De este dispositivo se obtendrán las señales Y, y C (Luminancia y Crominancia), analógicas que alimentarán a las siguientes secciones, la circuitería que separa la señal Y de C de la señal de video (Comb Filter) se muestra en la HOJA #1. La señal de video pasa a través de Q1 hacia el filtro (retardo de 1H). Por el emisor de Q2 sale la señal de color y por el emisor de Q3 la señal de luminancia. Por medio de RV1 y L1 se ajusta para que la señal de luminancia en el emisor de Q3, tenga la mínima cantidad de señal de color.

(b) DIGITALIZACION: La señal de video está compuesta por la señal de luminancia ($Y=0.3R + 0.59G + 0.11B$) y la de croma(C) las que se codifican por separado o bien se codifican las tres componentes de color R, G, y B; o lo que es más adecuado, codificar según los esquemas del CCIR (Anexo 3) en el cual se codifica una señal de luminancia(Y) y dos señales diferencia de color (R-Y y B-Y). Se opta por este criterio ya que es lo más adecuado si lo que se quiere es un equipo estándar para la familia de patrón 4:2:2(Anexo 3).

Dentro de esta especificación se recomienda que el número de muestras por línea completa sea de 858 para la señal de luminancia y 429 para cada señal diferencia de color. La frecuencia del muestreo será de 13.5MHz para la señal de luminancia y de 6.75MHz para las señales diferencia de color. El circuito propuesto para la obtención de la señal de luminancia y las señales diferencia de color se muestra en la HOJA #1. Nótese como se obtiene la señal R-Y y la señal B-Y mediante un proceso de demodulación. A cada circuito demodulador se aplican dos tensiones de señal: una es la señal de referencia de 3.579545MHz (Generada por medio de la circuitería de Q4) y la otra una señal de croma desde la circuitería separadora. La fase de la señal de referencia de 3.579545MHz se desplaza una cantidad determinada con respecto a la referencia de color para cada circuito demodulador, lo que permite que cada demodulador extraiga la señal apropiada de diferencia de color de la información de croma que se le ha aplicado.

Estas señales pasan al bloque muestreador (HOJA #2), en donde se utiliza tres conversores A/D (CI ADC0805) paso decisivo en la digitalización. Los osciladores de 13.5MHz y 6.75MHz también se los puede visualizar (conformado por Q1 y Q2). Cada muestra se codifica con 8bits con los que se trabajará en la siguiente sección.

(c) COMPRESION: En esta etapa se halla el procesamiento mismo de las señales Y, R-Y y B-Y. Estas ya digitalizadas entran a la circuitería implementada con alguna técnica del Capítulo 2. Con el afán de seguir con los esquemas ya estandarizados y las regulaciones internacionales, se utilizará el estándar de codificación JPEG (Capítulo 3) cuyo algoritmo se muestra en la Figura 3.5. En el mercado existe ya circuitería encapsulada (CI's) que realizan el procesamiento según el estándar JPEG¹ en el cual a la entrada se introduce la señal a codificar (Y, R-Y o B-Y) y a la salida se obtiene el código ya comprimido y listo para transmitir, es por esto que esta parte del prototipo solo quedará mencionada, pues las indicaciones al usuario que vienen con el CI son bastante claras. Esta parte del equipo se muestra como el esquema de la HOJA #3. Nótese que la entrada es un bus de ocho líneas de datos (pues la señal que está codificada a 8bits) y la salida es otro bus con un número de líneas útiles según el tipo de codificación (por ejemplo 8). Esto es debido a que cada bloque de imagen muestreado (que de antemano se tomará de 8x8 muestras) se codifica por longitud variable y esta depende de la probabilidad de aparición de ciertos valores de muestras, de la entropía del bloque, etc.

Nótese en el esquema de la HOJA #3 que haría falta algún bloque de retención en el cual la imagen de entrada digitalizada se debería tomar en porciones de 16 pixeles pues se va a codificar en JPEG bloque a bloque, esta circuitería y otras más ya se encuentran establecidas dentro del CI princi-

(1) Fueron ofrecidas soluciones para JPEG primero por C-Cube Microsystems Inc., San José, Calif. luego por LSI Logic Corp., Milpitas, Calif. El procesador es un único CHIP C1550 anunciado ya antes en 1987. La solución "Bi-Chip" de LSI Logic combina el procesador TDC L64739 con el Cuantificador más un codificador JPEG con CLV L64745. Los procesadores de las dos compañías implementan el algoritmo JPEG, con la posibilidad de que el usuario pueda cambiar los parámetros de JPEG tales como la resolución, de la imagen, número de componentes, la cuantificación y las tablas de código Huffman utilizadas.

pal cuyas condiciones de trabajo como la dimensión de los bloques de imagen número de niveles de cuantificación dimensión de una memoria externa (CI 4016) para almacenar las tablas Huff-man, etc. se pueden variar a gusto externamente. Lo que realmente interesa es como y cuando los datos se procesan dentro de este macrobloque que realmente es un microprocesador.

(d) CODIFICACION: Esta parte del circuito se entiende es el punto en donde a los datos ya comprimidos se les refuerza para atravesar el canal de comunicación, esto es se les proporciona un código que lo protegerá en la medida de lo posible contra los errores que a su paso encuentre en la transmisión. La circuitería capaz de realizar dicho control de errores es muy complicada; afortunadamente nuevas tecnologías han reducido el volumen físico a microcircuitos.

(e) MODULACION: Ya se ha establecido una suficiente base teórica sobre los esquemas de modulación utilizados en la transmisión de imágenes digitalizadas, y la práctica apuntaba a que los esquemas de modulación en cuadratura de fase son los más adecuados. Es así que el flujo de datos obtenido luego de la codificación, es llevado al proceso de modulación que es la adaptación de la información al canal de transmisión. Al igual que ocurre con el procesador de (c) y el de (d) la tarea de modulación se puede realizar también vía micro circuito, pues ya en el mercado existen CI's capaces de modular flujo binario hasta esquemas de 32 PSK con relativa sencillez.

Para el caso particular del prototipo un esquema Q-PSK será suficiente y la circuitería implicada en éste se halla en la HOJA #4. No interesa que tipo de información llega al modulador (si es la luminancia digitalizada o las diferencias de color digitalizadas), siempre y cuando éste realice adecuadamente su tarea, así lo que se va a modular son tres trenes de bits multiplexados en tiempo que conforman un solo flujo de información a modular (y luego a transmitir). Nótese que a la salida se obtiene ya no trenes binarios de unos y ceros, sino una señal sinusoidal modulada en cuadra-

tura de fase, de amplitud constante y esta es la señal a transmitir, luego de su correspondiente amplificación.

(f) AMPLIFICACION DE POTENCIA Y TRANSMISION: En esta etapa el nivel de potencia de la señal se aumenta y por tanto está lista para conectarse a un circuito de antena que bien puede ser una antena de banda media o banda alta, o una antena parabólica si la transmisión se realiza vía satélite.

La circuitería de esta etapa se halla en la HOJA #5 en donde también se encuentra la red acopladora a la antena de transmisión. El cálculo de esta red y de otras circuiterías que se muestren en este Anexo se suponen particulares al diseño que cada aplicación requiera, pues se pretende solamente dar una visión práctica de como se propondría un sistema real de transmisión de imágenes.

TERMINAL RECEPTOR DE SEÑAL-EXPOSITOR DE IMAGENES:

(a) RECEPCION: En este punto lo que se realiza es la captación y detección de la señal de RF transmitida. Los procesos de cálculo de antenas adecuadas y los estudios de propagación son aquí establecidos aunque no se los expondrá. Se sabe de antemano que la señal entrante es una señal modulada en cuadratura de fase que debido a la propagación atmosférica su nivel ha disminuido desde que salió de la antena transmisora.

(b) AMPLIFICACION: Debido a que la señal entrante es de muy baja potencia, se procede a amplificarla a fin de obtener una señal de amplitud suficiente como para proseguir en el procesamiento posterior. Esta circuitería y la de recepción se halla en la HOJA #6. Nótese que lo único que se obtendrá aquí es una señal de RF de amplitud constante y fase cambiante.

(c) DEMODULACION: Recíprocamente al proceso de modulación en la etapa transmisora, en este punto lo que se hace es convertir la señal análoga Q-PSK a su equivalente flujo binario de unos y ceros, el circuito que realiza esta labor es el complementario al de la etapa transmisora de

características idénticas entre ellos y con sus mismas cualidades y restricciones. La circuitería de la demodulación se encuentra en la HOJA #7. A la salida de este bloque lo que se obtiene es el flujo de bits codificado.

(d) DECODIFICACION: Se sigue un proceso inverso que su contraparte codificadora, esto es, se devuelven los bits originales tal vez con errores en su estructura que no hayan podido eliminarse.

(e) DESCOMPRESION: Los procesos de compresión-descompresión son inseparables en la implementación de un circuito de transmisión de imágenes, es así que el microprocesador utilizado en el terminal transmisor, es el mismo que se utilizará en este punto, solo que la filosofía de su aplicación cambia: Dentro del mismo micro circuito se hallan los algoritmos y bloques decodificadores y re-procesadores que realizan la tarea inversa a la compresión. De este modo la tarea sigue siendo modular y sin presiones propias de algún otro sistema decodificador. La existencia de este microprocesador compresor/descompresor JPEG es la única llave de acceso a la implementación sencilla de un sistema de comunicación transmisión/recepción de imágenes. Las señales resultantes a la salida de esta etapa son los valores digitalizados de los pixeles de la imagen original que ahora están listos para su presentación.

(f) PRESENTACION: En esta etapa se vuelve a la representación visual de una imagen desde su representación eléctrica dada en los bloques anteriores. Las tonalidades de color son disminuidas porque según el CCIR solo se pueden cuantificar con un limitado número de bits. De todas maneras debido a la limitación visual humana estas "pérdidas" al igual que los "errores de transmisión" son relativamente "imperceptibles" cuando se realiza la exposición de estas muestras de imagen. El proceso de presentación sigue una secuencia inversa a la captura de la imagen y consiste en pasar del dominio digital al dominio analógico y la señal obtenida que deberá ser en formato NTSC para este equipo se pueda inyectar a un sistema de televisión común y así visua-

lizar cual fue el desempeño de la transmisión. La circuitería para esta etapa se muestra en la HOJA #8. Nótese que las señales Y, R-Y, y B-Y se vuelven a procesar para hallar la señal de video original en el televisor

OBSERVACIONES SOBRE TECNOLOGIAS ACTUALES:

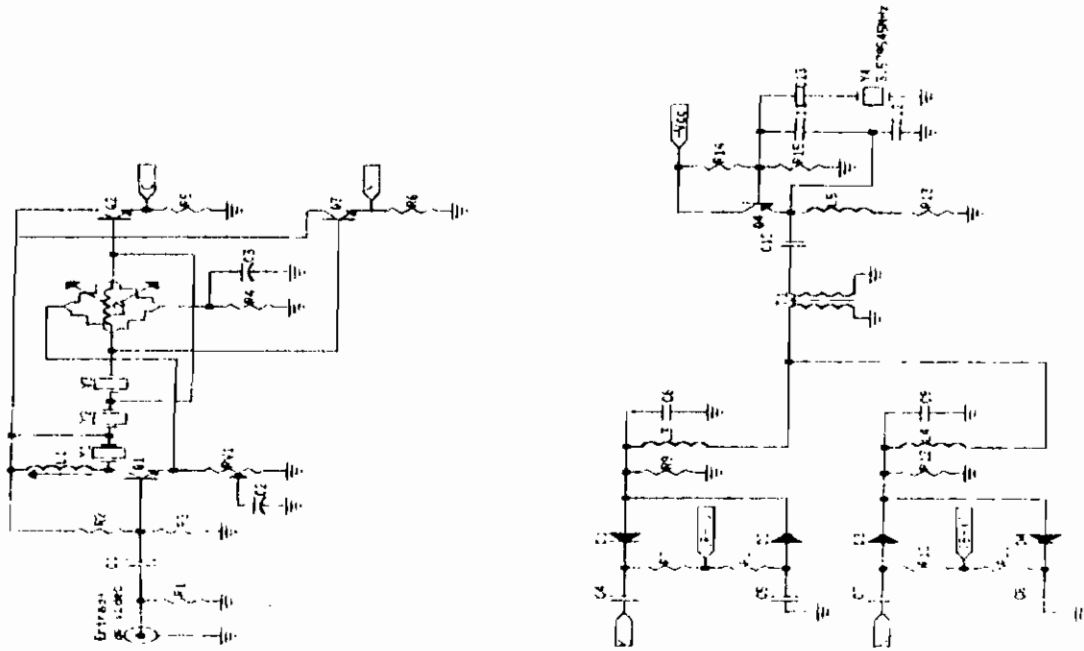
La tecnología DVI (Video Interactivo Digital) ha recorrido un largo y exitoso camino desde que en 1982 los ingenieros de RCA y General Electric comenzaron sus esfuerzos de investigación y desarrollo. INTEL anunció su aprobación en 1987. Actualmente (Abril 1993) la tecnología DVI pertenece a INTEL y está bajo el control de la división **Princeton Operation**. Los productos comerciales son una amalgama de tarjetas opcionales INTEL/IBM, software, productos de sistema, y software de terceros. El primero de los productos comerciales DVI de IBM incluía dos tarjetas Action Media 750 para máquinas de arquitectura MicroChannel. La tarjeta "liberadora" costaba 1995 dólares en EE.UU. y desplegaba archivos de datos de audio y video DVI comprimidos, desde discos compactos (CD-ROMs), discos ópticos, o unidades estándares de discos duros.

Una sola trama en colores de video NTSC requiere desde 500 KBytes a 2 MBytes para un archivo computacional de datos con imágenes. Un segundo de datos de video de movimiento completo (30 tramas/segundo en NTSC) podría ocupar unos 60 MBytes. El circuito pionero de INTEL el **i750** para procesamiento de video, comprime digitalmente y expande archivos de audio y video de movimiento completo (a veces llamados **data streams**). El circuito integra dos chips: el **82750PA Pixel Processor** (corriendo a 12.5 MIPS) y el **82750DA Output Display Processor**. El chip ODP puede desplegar video digital a resoluciones entre 256x200 pixeles y 1024x512 pixeles, a 8 y 16 colores. La tecnología puede desplegar movimiento completo, salida de imagen digital con una cantidad de efectos especiales (por ejemplo, espejo, arrugado y enrollado) dado que el microcódigo del **i750** puede ser reprogramado con rapidez de operación completa. El principio para la compresión de imágenes es simple: reducir los datos a una forma abreviada que mantenga la misma información básica.

Según INTEL, la razón de compresión y descompresión de un archivo de video móvil es de casi 160 a 1. (Utilizando tecnología de fractales, **Iterated Systems** se obtiene tasas de 2500 a 1!). Esto permite expresar información de audio/video de una sola trama móvil en 4.5 KBytes para expandirlo mientras se lo ve "al vuelo", de vuelta a su tamaño original de 720 KBytes. Esto permite que un típico disco CD-ROM almacene 72 minutos de video y audio estéreo comprimidos en DVI en vez de los 28 segundos originales sin comprimir. La solución: guardar solo los cambios y guardarlos en forma comprimida.

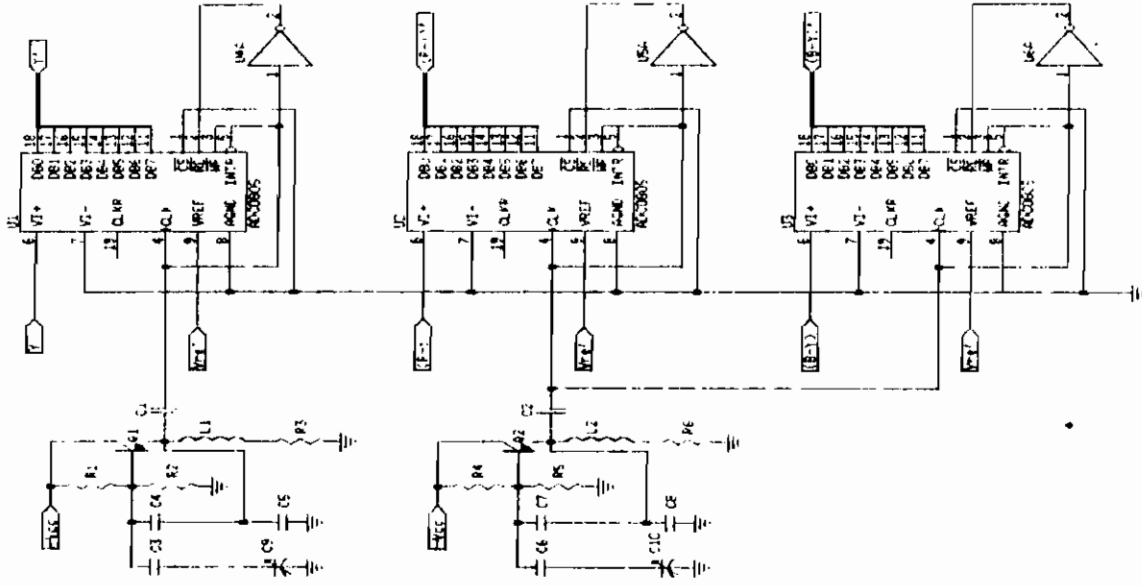
Son disponibles productos de la recomendación H.261 de *LSI Logic* y *Graphics Communications America Ltda. (CGA)*, Glen Ellyn, Ill. LSI Logic anunció en Septiembre de 1990 una familia estructurada en bloques de siete chips para codificación y decodificación de video en total movimiento (incluyendo a H.261), y CGA la siguió con una familia estructurada en bloques de doce chips un mes más tarde. Las dos son las mejores clasificadas como codecs estructurados en bloques para video de función específica. La línea de LSI Logic consiste de cuatro chips procesadores y tres codificadores y decodificadores. Los procesadores son el **L64720** para la estimación del movimiento, el **L64730** de la Transformada Discreta Coseno (TDC), el **L64740** para la cuantificación y el **L64760** para la discriminación entre inter/intra trama. Los otros tres chips son el **L64715** codec BCH (Bose-Chaudhuri-Hocquenghen) de corrección de errores, el **L64750** codificador por longitud variable (CLV) de H.261 y el **L64751** decodificador por longitud variable (DLV) de H.261. Todos los cuatro procesadores pueden ser utilizados en cualquier sistema de compresión basado en transformada. Por ejemplo, el lazo de codificación de video MPEG (sin el codificador por longitud variable) se puede implementar con los chips LSI. Desde Mayo de 1990, Intel ha estado investigando en software utilizado por sus chips **i750** para realizar la compresión y descompresión basada en algoritmos de Cuantificación Vectorial alcanzando desempeños en tiempo real para imágenes de 256 por 240 pixeles. Una característica del impacto de Intel es su "programabilidad".

UVC, Irvine, Calif. tiene un chip capaz de realizar compresiones en tiempo real de hasta 500:1, utilizando técnicas puramente intra-trama. Otra compañía, *SGS-Thompson Microelectronics*, Phoenix, Ariz., ha anunciado partes afines para el uso en sistemas de compresión. Esta tiene varios procesadores TDC (*STV3200*, *STV3208*, y *A121*) y un procesador de estimación de movimiento (*STV3220*).



6521A-0005
 13.5MHz
 A

6521A-0005
 6.75MHz
 B



6521A-0005
 13.5MHz
 A

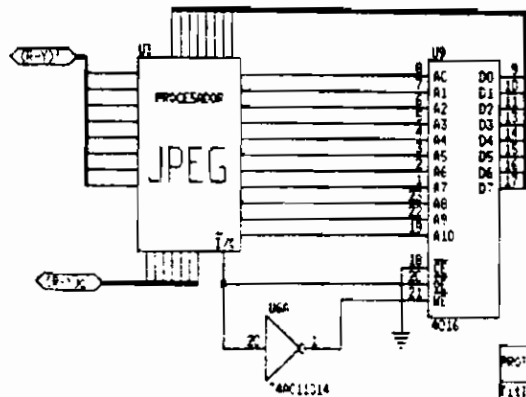
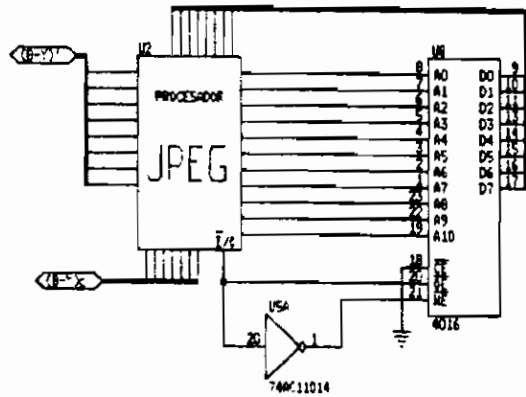
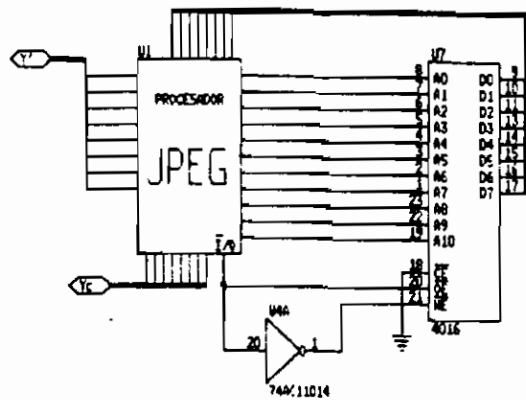
6521A-0005
 6.75MHz
 B

6521A-0005
 13.5MHz
 A

6521A-0005
 6.75MHz
 B

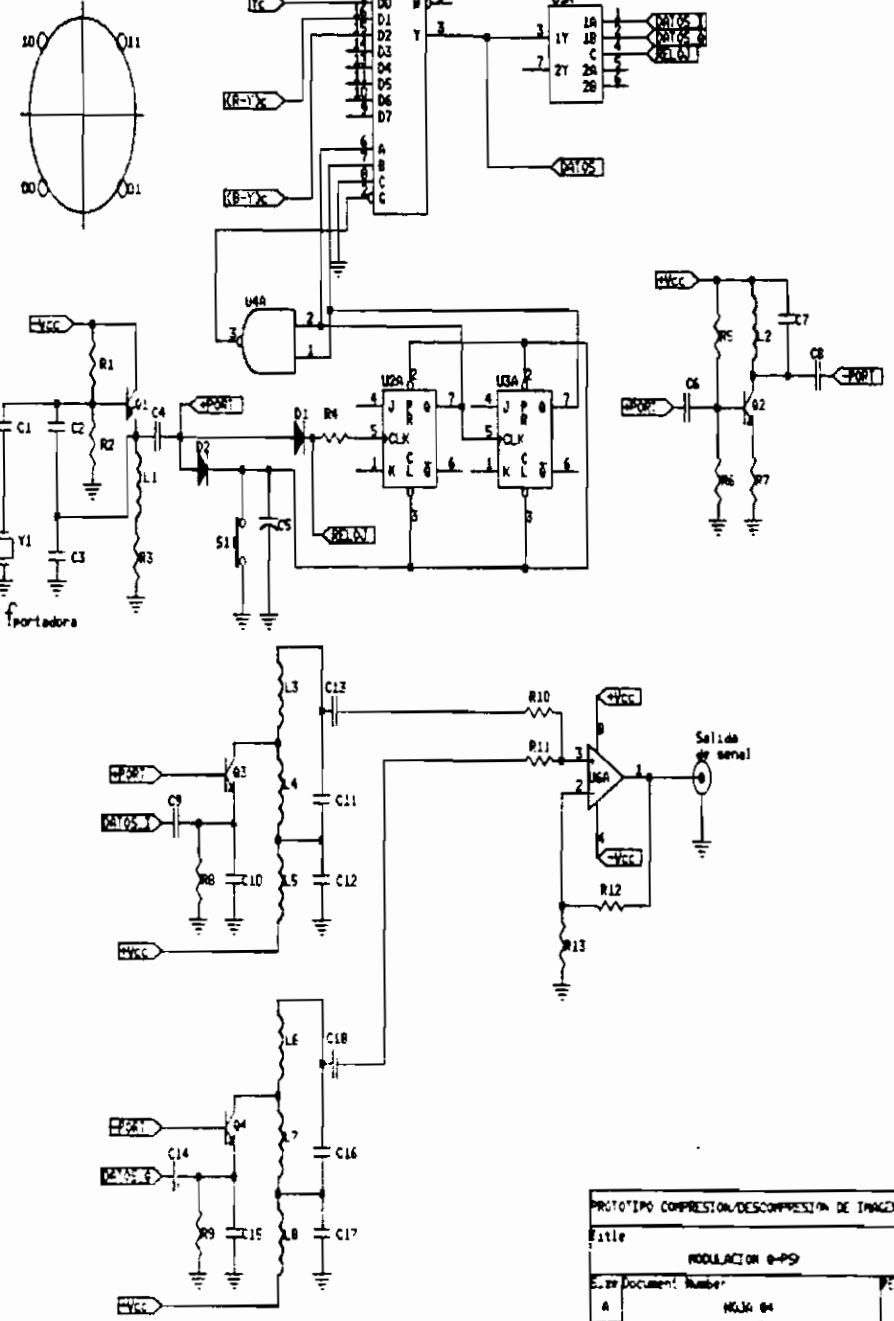
6521A-0005
 13.5MHz
 A

6521A-0005
 6.75MHz
 B

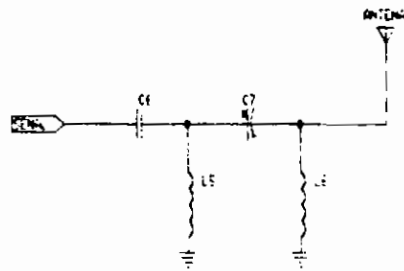
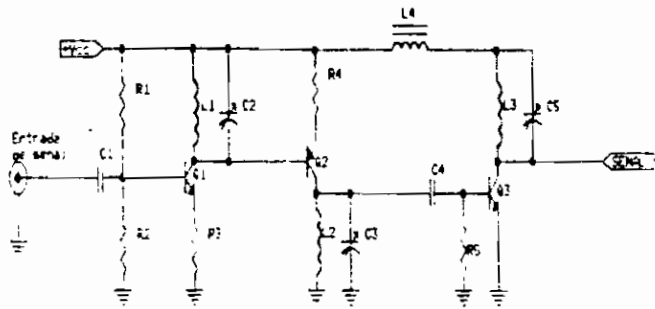


PROTOTIPO COMPRESION/DESCOMPRESION DE IMAGEN	
Title	
CODEC JPEG	
Size Document Number	REV
A	NOVA 03
Date: June 1, 1995	Sheet 3 of 8

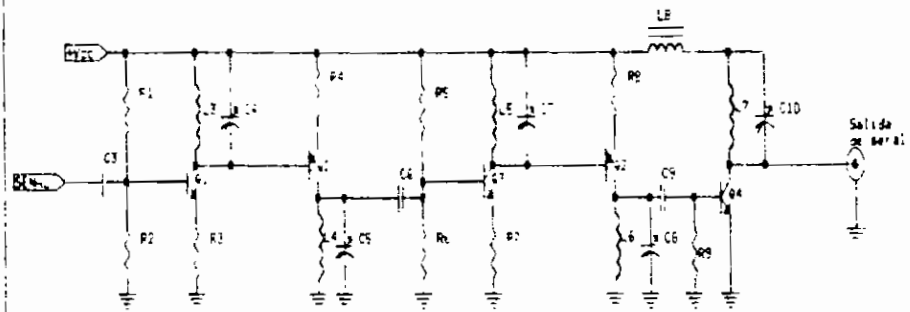
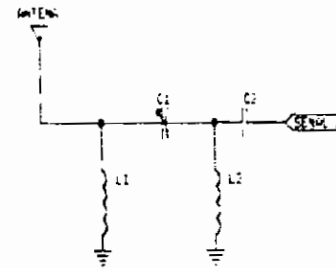
Diagrama de fase



PROTOTIPO COMPRESION/DESCOMPRESION DE IMAGEN	
Title	
MODULACION 0-PS	
Size Document Number	REV
A	NOVA 04
Date: November 11, 1995	Sheet 4 of 8



PROYECTO COMPLETO DEL APRENDIZAJE DE INGENIERIA
 TITULO
 AMPLIFICACION DE POTENCIA Y RADIAION
 Documento Number
 A HOLA DE
 Date: June 1, 1995+re: 5 of 6



PROYECTO COMPLETO DEL APRENDIZAJE DE INGENIERIA
 TITULO
 RECEPCION Y AMPLIFICACION
 Documento Number
 A HOLA DE
 Date: June 1, 1995+re: 6 of 6

