

INFORME FINAL DEL PROYECTO #20051850  
DESARROLLO E INVESTIGACION DE LOS ALGORITMOS DE COMPRESION DE  
IMÁGENES DE VIDEO

**Resumen.**

En el marco del proyecto de investigación #20051850 varios algoritmos de compresión de las imágenes estáticas y dinámicas basados a la transformada wavelets fueron desarrollados. Los algoritmos desarrollados de compresión de imágenes estáticas se basan a la técnica de transformada discreta wavelet lifting y se emplean para la compresión de imágenes en marcos I en las secuencias de datos de video. Para codificar las imágenes de marcos de tipo P en las secuencias de datos de video, fueron desarrollados los algoritmos nuevos que emplean la transformada wavelet, la búsqueda completa rápida y jerárquica. Los resultados obtenidos con el codec de video desarrollado presentados en el informe actual muestran que el codec propuesto produce mejores resultados para la compresión de los datos de video comparando con el mejor codec de video actual, H.264. Los resultados de simulaciones obtenidos muestran, que el codec desarrollado se puede emplear en varias aplicaciones de compresión de datos de video, incluso el procesamiento de video en el tiempo real o casi real.

## *Métodos y materiales.*

Los valores de los pixeles producidos por el sensor son continuos sobre algún rango finito y son linealmente relacionados a la intensidad de energía radiante en cada punto muestreado. El uso del termino energía radiante (mejor dicho luz) es calculado desde el sensor sensible a las longitudes de onda fuera del rango de la visión humana. En el caso de imágenes que fueron generadas previamente, la energía radiante es el resultado de la fuente de iluminación que incide sobre la imagen y es reflejada o transmitida al sensor.

Diferentes estrategias son posibles para los puntos de muestreo, pero la más común es un espacio cuadrulado rectangular. Idénticamente, cada muestra corresponde a una región pequeña infinitesimal de la fuente, pero puesto que la naturaleza física de los sensores y ópticas asociadas es actualmente un valor integrado sobre la misma área finita. El número de puntos muestreados por unida de área define la frecuencia de muestreo del sistema y esta frecuencia de muestreo será seleccionada en base al teorema de muestreo de Nyquist, es decir la frecuencia de muestreo será menor en dos veces a la componente de frecuencia espacial máxima de la fuente. Este criterio presenta la imagen muestreada que contiene objetos aliasing generados cuando la frecuencia espacial es mayor que 1/2 de la frecuencia de muestreo. Para prevenir el fenómeno aliasing, un prefiltrado se usará en el sistema para limitar el ancho de banda de la fuente anterior al muestreo. (Note que la integración realizada por el sensor también actúa como un prefiltro). En sistemas de digitalización de imágenes, la frecuencia de muestreo está comúnmente dada en términos de la resolución de escaneo, la cual es la inversa de la frecuencia de muestreo. En general, la resolución de escaneo requerida depende de la aplicación. Por ejemplo, para una radiografía de 14"x17" que observa un radiólogo a una distancia típica de 14", la resolución de escaneo desde 70 microsegundo/píxel, de tal manera que la frecuencia espacial de la captura es mayor que a la se adquirió de 7ciclos/minuto. El escaneo de resolución más típico es el de 12 microsegundo/píxel.

Cada punto muestreado continuamente es cuantificado por uno de un número discreto de niveles en el orden para formar una imagen digital. Conoceremos que cambios uniformes en los valores de intensidad no son percibidos igualmente por el sistema visual humano. Como las imágenes son vistas para asuntos humanos en varias aplicaciones, es importante realizar la cuantificación en un dominio que este en correspondencia con el sistema de percepción de la visión humana.

El número de niveles de cuantificación requeridos para representar adecuadamente un imagen es también dependiente de la aplicación. Para documentos de texto binario, solo dos niveles (1 bit/píxel) se requieren pues cada punto muestreado será blanco o negro. Pero para escenas de la naturaleza o fotografías de tonos continuos es común usar 8 bits/píxel (256 niveles). Sin embargo, dependiendo sobre el rango dinámico de la fuente y del tipo de la salida del sensor (es decir lineal o logarítmico) puede ser necesario el uso de 10 o 12 bits/píxel.

La presente redundancia en imágenes digitales es altamente dependiente del sistema usado para formar la imagen tales como los parámetros usados para representarla. En particular, la frecuencia de muestreo, el número de niveles de cuantificación, y la presencia de la fuente y/o ruido del sensor pueden afectar toda la compresión alcanzable. A pesar de que existirán varios de imagen a imagen, la siguiente tendencia es generalmente hallar algoritmos de compresión basados en la estadística:

- Como la frecuencia de muestreo es incrementada, incrementa la correlación pixel a pixel, lo cual permite un ratio grande compresión. El ratio de compresión (CR) es definido por:

$$CR = \frac{\text{Número.de.bits.de.la.imagen.original}}{\text{Número.de.bits.de.la.imagen.comprimida}}$$

El incremento en la correlación pixel a pixel significa, que si la frecuencia de muestreo es incrementada por un factor de dos, el incremento en el número de bits requeridos para la imagen comprimida será menor que un factor de dos (a pesar de que el total el número de bits incrementa).

- Incrementando el número de niveles de cuantificación reduce la correlación píxel a píxel para algunas extensiones, esto reduce la compresión realizada.
- La presencia de algún ruido en la fuente (por ejemplo, en una película el ruido granulado en una en una fotografía) o el ruido introducido por el sensor decrecerá la correlación píxel a píxel y reducirá la cantidad de compresión realizada.

CCITT (comité consultivo de la Internacional de teléfonos y telégrafos) tiene estandarizado los algoritmos de codificación para vídeo telefonía y vídeo conferencias en un rango de frecuencia de bits que van desde los 64 a 192 Kbits/s. Desde 1988, un grupo de estandarización conocido como MPEG (Moving Picture Experts Group, grupo de expertos de imágenes en movimiento) trabajando también bajo el apoyo de ISO-IEC/JTC1/SC2/WG8 desarrollaron un estándar para telecomunicaciones y almacenamiento de imágenes en movimiento y sonido usando medios digitales de almacenamiento con una frecuencia de 1.0-1.5 Mbits/s. La meta de los estándares MPEG es una técnica de propósito general para aplicaciones tan diversa como publicaciones electrónicas, guía de viajes, (vídeo) texto, educación, juegos, entretenimiento, correos de vídeo y teleseminarios de capacitación.

El diagrama de bloques de los estándares MPEG 1 y 2 para compresión de datos de video se presenta en la Figura 1.

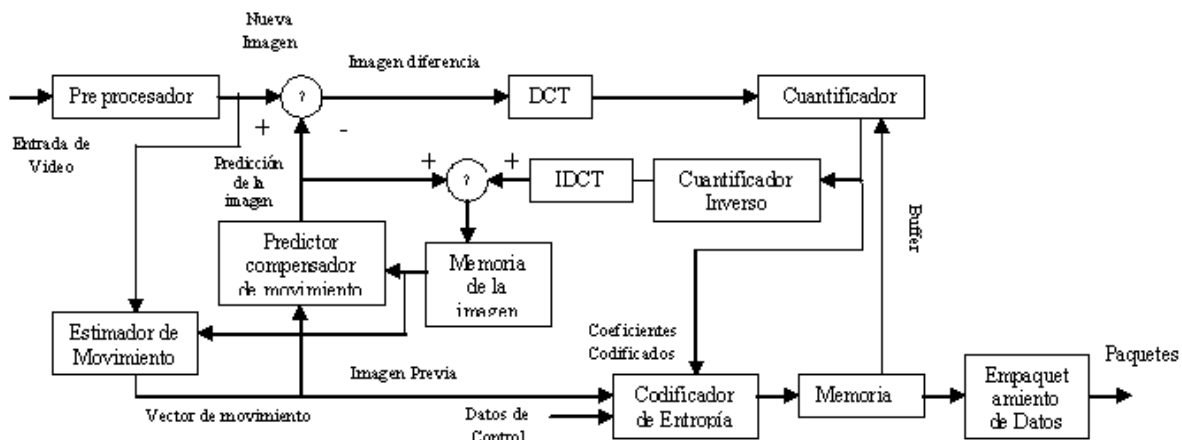


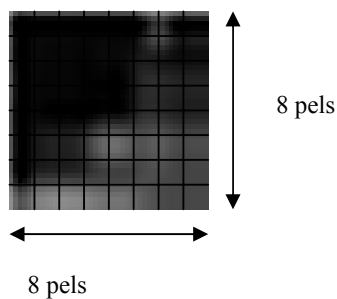
Fig. 1. El diagrama de bloques de los estándares MPEG 1 y 2 para compresión de datos de video

El estándar nuevo MPEG4-10 (ó MPEG4AVC) es basado a las técnicas de compresión de datos de video H.264 fue adoptado primero en el julio 2004 y todavía esta en el desarrollo. Este estandar permita obtener una tasa de compresión en 2 veces más alta que en el estándar MPE4 básico el cual está basado a las técnicas especificadas por H.263. En breve, en H.264 la tasa de compresión se aumenta por una gran complejidad del procesamiento, principalmente, se usan bloques de diferente tamaño para mejorar la estimación del movimiento y así, reducir el error de predicción de los marcos en la secuencia de datos de video. Además, se usa la técnica de codificación CABAC (context adaptive block arithmetic coder) que es mejor con las técnicas más simples de codificación de datos. En el mismo tiempo, se conserva la transformada discreta de coseno DCT, como procedimiento básico para la cuantificación de los datos de imágenes y en los marcos de referencia (marcos tipo I) y en los marcos predichos (marcos tipo P y B). Aunque se conocen trabajos que tratan de mejorar la DCT

usada en el H.264, así, como transformada DCT basada a números enteros, de bloques de diferente tamaño que 8x8, se sigue usando la DCT de punto flotante, por bloques de tamaño 8x8.

La DCT es una operación matemática que calcula la componente de frecuencia de una señal muestreada a una tasa de muestreo. La DCT debe ser aplicada a un número finito de muestras. La matemática se simplifica considerablemente si el número de muestras usadas es una potencia par. La DCT tiene ciertas propiedades que simplifican los modelos de codificación y hacen a la codificación eficiente en términos de las medidas de calidad perceptual.

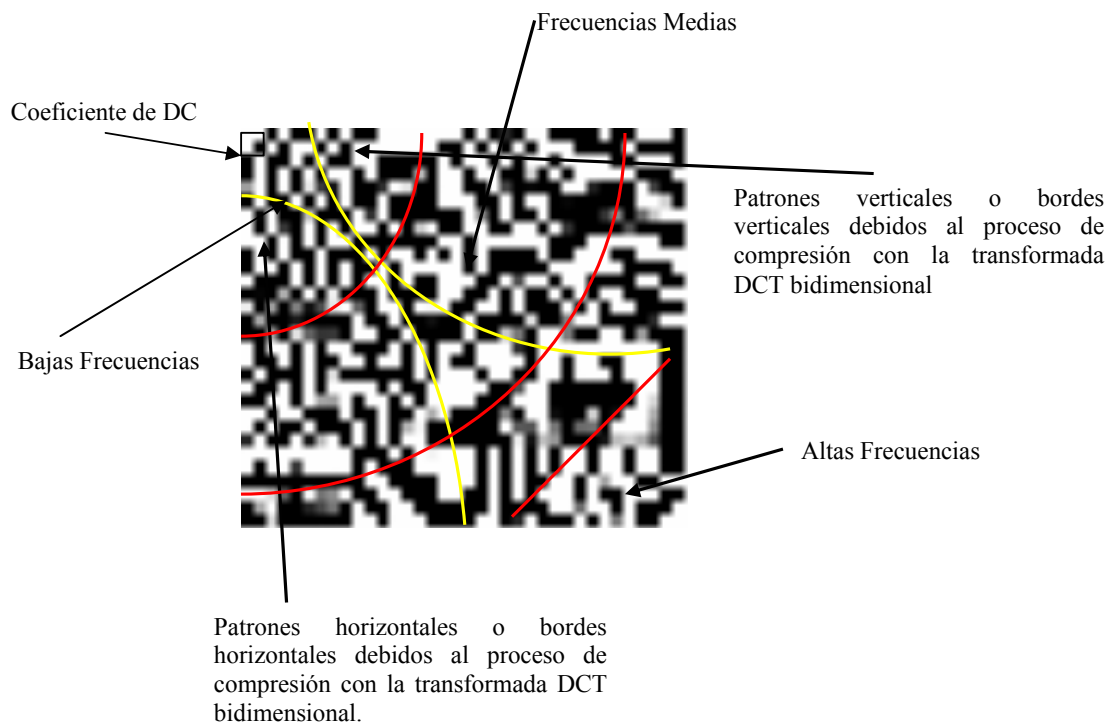
Básicamente, la DCT es un método de descomposición de un bloque de datos en una suma de pesos de frecuencias espaciales. Cada una de esos patrones de frecuencias espaciales tiene un coeficiente correspondiente, la amplitud necesaria para representar la contribución de un patrón de frecuencia espacial en un bloque de datos que serán analizados. Cada patrón de frecuencia espacial es multiplicado por sus coeficientes y el resultado son arreglos variables de 4x4, 8x8 o 16x16 que se suman, cada pixel por separado, para reconstruir el bloque de 8 x 8.



**Figura 2:** Subbloque 8x8 píxeles

Si los coeficientes de la DCT son de baja frecuencia entonces sus coeficientes no son cero, los datos en el bloque varían suavemente con la posición. Si las altas frecuencias están presentes, el bloque cambia de intensidad rápidamente desde un pixel a otro. Donde los 64 patrones bidimensionales, agrupados en arreglos de 8 x 8 el término nos da el promedio o valor de DC de los bloques de 8 x 8 de cada pixel. Hay que notar que en cada eje, la variación de la DCT es unidimensional, así en la figura tenemos que en realidad la DCT bidimensional es el producto de las oscilaciones de una dimensión que se ven a lo largo de cada eje.

La representación frecuencial por cada elemento en el arreglo resultante es función de un arreglo índice para cada elemento. El contenido de frecuencia para cada conjunto de muestras a cada frecuencia se calcula tomando un peso promedio para cada conjunto. La figura 3 la distribución de estas frecuencias.



**Figura 3** La distribución de frecuencias y la descomposición estructural de estas debida al proceso de compresión utilizando la transformada DCT bidimensional.

Por último cabe mencionar que esta transformada cuenta con una buena propiedad de compactación de energía y es una aproximación de la KLT (Karhunen-Loeve Transform), que produce coeficientes no correlacionados, con la diferencia de que los vectores base de la DCT dependen sólo del orden de la transformada seleccionado, y no de las propiedades estadísticas de los datos de entrada.

La no correlación de coeficientes es muy importante para compresión, ya que, el posterior tratamiento de cada coeficiente se puede realizar de forma independiente, sin pérdida de eficiencia de compresión. Otro aspecto importante de la DCT es la capacidad de cuantificar los coeficientes utilizando valores de cuantificación que se eligen de forma visual.

La DCT está bastante relacionada con la DFT, con la diferencia de que es una transformada real, debido a que los vectores base se componen exclusivamente de funciones coseno muestreadas. Para nuestro estudio solamente hacemos mención de la transformada de coseno discreta en dos dimensiones, dado que nuestro algoritmo se basa en este principio, el cual se menciona a continuación.

Las imágenes comprimidas por MPEG están formadas por arreglos de muestras de dos dimensiones, por lo consiguiente, MPEG usa las dos dimensiones de ocho por ocho puntos (8x8) formando la DCT. La DCT 2-D es una transformación separable, donde se quiere que este formada por DCT de una dimensión horizontales y verticales.

Supongamos que el arreglo de muestras de 2-D,  $f(x, y)$ , esta siendo transformada dentro de una DCT de 2-D. Las transformaciones DCT horizontales de una dimensión (la aplicación de la DCT de 1-D horizontal por cada fila del arreglo). La dependencia en  $x$  es transformada dentro de una suma de pesos por la función coseno de  $x$ , pero si hay una variación vertical en el arreglo, los pesos también son una función de  $y$ .

$$g(\mu, y) = \frac{C(\mu)}{2} \sum_{x=0}^7 f(x, y) \cos[(2x+1)\mu\pi/16] \quad \dots(1)$$

Las DCT's de 1-D verticales son entonces las transformaciones de las dependencias de y de cada columna del arreglo  $g(\mu, y)$  dentro de la suma de pesos de la función coseno de y con frecuencia de índice  $\nu$ .

$$F(\mu, \nu) = \frac{C(\nu)}{2} \sum_{x=0}^7 g(\mu, y) \cos[(2y+1)\nu\pi/16] \quad \dots (2)$$

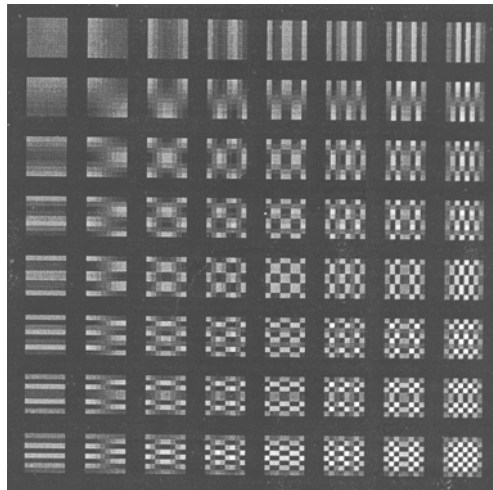
Con esto notamos que al separar los procesos no afectan los resultados. Por esto cuando juntamos estas dos secuencias, obtenemos la DCT para 2-D.

$$F(\mu, \nu) = \frac{C(\mu)}{2} \frac{C(\nu)}{2} \sum_{y=0}^7 \sum_{x=0}^7 f(x, y) \cos[(2x+1)\mu\pi/16] \cos[(2y+1)\nu\pi/16] \quad \dots (3)$$

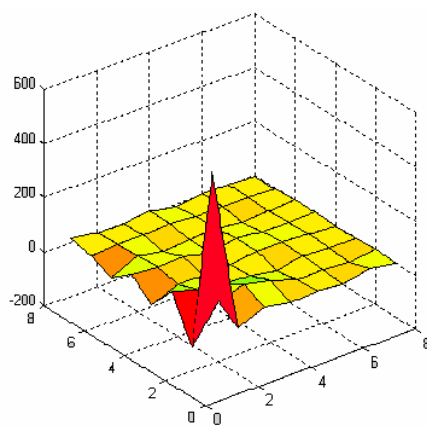
Donde  $\mu$  y  $\nu$  son índices de frecuencia horizontales y verticales respectivamente y las constantes  $C(\mu)$  y  $C(\nu)$  están dadas por.

$$\begin{aligned} C(\mu) &= 1/\sqrt{2} \quad \text{si } \mu = 0 \\ C(\mu) &= 1 \quad \text{si } \mu > 0 \end{aligned} \quad \dots (4)$$

El resultado del productos de los términos de cósenos de la función anterior se muestran como un arreglo de escala de grises en la figura 2.3.1.1. En esta figura el índice de frecuencia horizontal aumenta de izquierda a derecha y el índice de frecuencia vertical incrementa de arriba hacia abajo. La fila superior es por lo tanto el producto de una constante y la DCT de 1-D horizontal con una frecuencia  $\mu$  incrementándose de izquierda a derecha. La columna izquierda es el producto de una constante y la DCT de 1-D vertical incrementada por una frecuencia  $\nu$  de arriba hacia abajo. El bloque arriba a la izquierda, es la constante que concentra la mayor intensidad de energía, este se le conoce como coeficiente DC.



**Figura 4.** 64 valores transformados por la función DCT 2-D aplicada a un arreglo de 8x8 muestras



**Figura 5.** Representación tridimensional del subbloque de la Figura 3 después del proceso de compactación de energía con DCT, el pico más alto corresponde al coeficiente de DC, donde se concentra la mayor energía del subbloque.

Este proceso de transformación puede ser reversible, mediante la IDCT o transformada de coseno discreto inverso, recreando las muestras originales. De manera similar, esto se puede lograr mediante el cálculo de la IDCT para 1-D para cada columna del arreglo y de igual forma para cada fila del arreglo de resultados. Esto es, la IDCT de 2-D esta dada por la siguiente ecuación.

$$f(x, y) = \sum_{\mu=0}^7 \sum_{v=0}^7 \frac{C(\mu)}{2} \frac{C(v)}{2} F(\mu, v) \cos[(2x+1)\mu\pi/16] \cos[(2y+1)v\pi/16] \dots \quad (5)$$

El producto de los términos del coseno de dos dimensiones esta basado en las funciones para la DCT de 2-D, y  $F(\mu, v)$  son los coeficientes de la DCT. Los valores de  $C(\mu)$  y  $C(v)$  son los mismos que los anteriores.

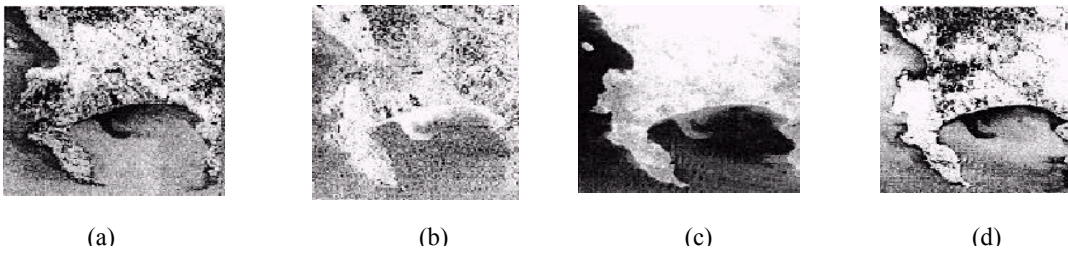
Por otra parte, hoy en día existen métodos más efectivos para el procesamiento de las imágenes, como la compresión de fractales o la transformada discreta de ondeletas (DWT, o discrete wavelet transform), los cuales dan la posibilidad de comprimir imágenes con niveles de compresión más altos y también obtener la calidad más alta en imágenes descomprimidas. Si comparamos las imágenes comprimidas con la misma tasa de compresión, podemos ver que la técnica DWT nos permite obtener una calidad más alta de la imagen comprimida en comparación con la técnica DCT.

La transformada óptima para la compresión de datos es la transformada de Karunen-Loev (KLT). Esta transformada fue introducida originalmente como un desarrollo serie para procesos aleatorios continuos, por Karhunen y Loéve. Para secuencias aleatorias, Hotelling estudió primero lo que se llamó método de las componentes principales, que es el equivalente discreto del desarrollo serie KL. En el tratamiento digital de imagen es común referirse a esta transformada como transformada de Hotelling o método de las Componentes Principales. A diferencia de las transformadas anteriormente vistas, la transformada KLT esta basada en las propiedades estadísticas de la imagen y además no cumple la propiedad de separabilidad. Las principales aplicaciones de esta transformada son la compresión y la rotación de imagen.

Como una introducción a la KLT, suponemos el caso en el que una imagen  $N \times N$  ( $f(x,y)$ ) es transmitida  $M$  veces a través de un cierto canal de comunicación. Debido a que cualquier canal fisico esta sometido a perturbaciones aleatorias (ruido), el conjunto de imágenes recibidas,  $\{f_1(x, y), f_2(x, y), \dots, f_M(x, y)\}$ , representará, en general, un conjunto estadístico cuyas propiedades están determinadas por las características del canal y la naturaleza de la perturbación. Un ejemplo de tal conjunto es una serie

de imágenes de la misma escena transmitidas desde un trasbordador espacial. En este caso, las imágenes se degradan a causa de las perturbaciones atmosféricas y el ruido eléctrico en el transmisor y el receptor. Los grupos de imágenes de la misma escena pero que difieren en las propiedades espectrales son conocidos como bandas. El conjunto de bandas se conoce como imagen multibanda o multispectral. La fuente más típica de imágenes multispectrales es un satélite de observación terrestre. Estos satélites se diseñan para enviar información detallada sobre la superficie de la tierra.

Por tanto, de la misma forma que la transformada de Fourier es un método de representar una imagen de banda única, la KLT proporciona un método nuevo pero equivalente, de representar una imagen multibanda. Si la imagen original tiene  $m$  bandas, la transformada KL de la imagen también tendrá  $m$  bandas. Sin embargo, las bandas en la imagen transformada multispectral están ordenadas de acuerdo con la información que contienen. La primera banda tiene el mayor contenido de información, la segunda el siguiente contenido más grande y así sucesivamente. Las últimas bandas pueden contener información tan pequeña que visualmente parezca ruido. El ahorro en el almacenamiento se deriva del descarte o supresión de estas bandas de contenido bajo de información., esto lo podemos apreciar mejor en la Figura 6.



**Figura 6.** Resultado del análisis de los componentes principales. (a) primer componente principal, (b) segundo componente principal, (c) tercer componente principal, (d) cuarto componente principal.

Esta claro que la mayoría de la información de las cuatro bandas esta en los dos primeros componentes principales. Por lo que el último componente principal no contiene casi nada, sólo ruido, el tercero es un poco mejor. Típicamente, con este tipo de datos, el 95% de la información está en los dos primeros componentes principales.

El análisis de las componentes principales nos permite eliminar la correlación entre las bandas de imágenes multispectrales. La técnica puede utilizarse para reducir la cantidad de datos requeridos para representar un conjunto de imágenes determinado. La eliminación de información redundante tiene el efecto práctico evidente de reducir los requerimientos para almacenar los datos. También se puede utilizar en aplicaciones de clasificación de imagen.

La transformada de Karhunen-Loeve de la forma general, es la siguiente.

$$K(k, l) = \sum_{M=0}^{N-1} \sum_{N=0}^{N-1} F(m, n) A(m, n; k, l) \quad \dots \quad (6)$$

La función base  $A(m, n; k, l)$ , satisface a la ecuación

$$\lambda(k, l) A(m, n; k, l) = \sum_{m'=0}^{N-1} \sum_{n'=0}^{N-1} C_F(m, n; m', n') A(m', n'; k, l) \quad \dots \quad (7)$$

Donde  $C_F(m, n; m', n')$  denota la función de covarianza del conjunto de datos de la imagen y  $\lambda(k, l)$  es una constante fija para cada punto. De la definición de la KLT, es evidente que las funciones base dependen de la matriz de covarianza y por tanto no pueden predeterminarse; por lo que existen



únicamente unos pocos casos en los que las soluciones analíticas están disponibles. La KLT se dice que es una transformada óptima debido a que tiene las siguientes propiedades:

- La transformada KLT proporciona una condición óptima sobre la cual valorar otras transformadas discretas.
- En el tratamiento digital de imagen decir transformada Karhunen-Loeve es lo mismo que transformada Hotelling o método de las componentes principales (que esta basado en la KLT).
- A diferencia de otras transformadas la KLT se basa en las propiedades estadísticas de la imagen.
- La función base de esta transformada depende del conjunto de datos de la imagen y únicamente puede predeterminarse para señales de Markov de primer orden estacionarias; por lo que, en general no es una transformada rápida.
- En teoría la KLT es óptima con respecto al menor error cuadrático medio y la mejor compactación de energía.
- Aunque no existe un verdadero algoritmo rápido para la KLT, si se han desarrollado aproximaciones rápidas a esta transformada.
- Las principales aplicaciones de la KLT son la rotación y la compresión de imagen.
- Descorrelaciona completamente la señal en el dominio transformado.
- Minimiza el total de la representación de entropía en la secuencia.
- Contiene la mayor varianza (energía) en el menor número de coeficientes transformados (excelente compactación de energía).
- Minimiza la representación total de la entropía de la secuencia (la entropía es la función de probabilidad con la que aparecen los diferentes niveles de cuantificación de la imagen).

A mediados de 1980, la teoría de las wavelets fue desarrollada en matemáticas aplicadas. Enseguida la codificación subbanda la cual ha sido un área de investigación muy activa tanto para la compresión de imágenes como para la compresión de video, fue identificada como un coseno discreto de wavelet. Aunque la principal característica está en la estructura de los filtros subbanda que fueron desarrollados junto con la teoría wavelet, que permite una aproximación más productiva en el diseño de filtros. Así, que hoy en día las wavelets y la codificación subbanda son términos muy usados en la literatura técnica en esta área.

Las wavelets son funciones definidas sobre un intervalo finito y con valor medio cero. La idea básica de la transformada wavelet es representar arbitrariamente una función como superposición de un conjunto de wavelets o funciones básicas. Estas wavelets se obtienen a partir de una wavelet prototipo denominada wavelet madre, mediante dilataciones, escalados y traslaciones. Se pueden aplicar a todo tipo de problemas sobre el procesado de la señal: eliminación de ruidos, sismología, música, criminología y visión, entre otras. A menudo, las wavelets se utilizan como alternativa al análisis de Fourier. Dos tipos de descomposición subbanda son usados comúnmente en la compresión de imágenes: estas son la uniforme y la piramidal. La descomposición uniforme divide una imagen en subbandas de igual tamaño. En contraste la descomposición piramidal representa una descomposición por octavo de banda, ofreciendo una representación por multiresolución de la imagen. La wavelet convencional o codificador de imagen subbanda, principalmente explota la propiedad de compactación de energía de la descomposición subbanda usando estrategias de acomodamiento de bits óptimos. Aunque este tipo de wavelet provee una mejor calidad visual al eliminar el efecto de “bloking” en comparación con codificadores en base a técnicas de bloques como JPEG, la medida de desempeño objetiva medida por la relación de la señal a ruido pico (PNSR) se incrementa moderadamente.

Una regla esencial en la codificación wavelet es que si la imagen admite un modelo estocástico como los campos aleatorios de Markov, entonces la características estadísticas de la transformada wavelet están dadas por:

- Acomodamiento espacial de las frecuencias
- Compactación de energía
- Sectorización subbanda de los coeficientes significativos
- Similitud de subbanda cruzada
- Decaimiento de las magnitudes en los coeficientes a través de las subbandas

Existen varios términos a considerar dentro de la teoría de las wavelets:

*Acomodamiento Espacial De Las Frecuencias:* Como la codificación subbanda descompone una imagen en pocas bandas de frecuencia, que no se sobreponen, cada subbanda está localizada en una frecuencia cercana que esta cercana a otra frecuencia independiente. Cada coeficiente wavelet representa información en una frecuencia cercana a un rango en una localización espacial específica.

*Compactación De Energía:* Una imagen natural esta típicamente compuesta por largas porciones homogéneas y regiones texturizadas, y pequeñas porciones de bordes que involucran limites de objetos perceptualmente importantes. Las regiones homogéneas principalmente consisten en componentes de baja frecuencia, las regiones texturizadas tienen una variación moderada y consisten de una mezcla de componentes de baja y alta frecuencia. De acuerdo a esto la transformada wavelet compacta mayor energía distribuida en las regiones homogéneas y en las regiones texturizadas en filtros paso bajo subbanda. En cada momento una banda paso bajo de fina resolución es descompuesta en cuatro subbandas a igual resolución, se aplica un muestreo crítico que permite nuevamente generar una subbanda paso bajo, para representar sólo un cuarto del tamaño original de la subbanda paso bajo original. Aplicando el proceso de descomposición repetidamente tres veces en la imagen, la energía puede compactarse efectivamente en pocos coeficientes wavelet. Después de la segunda escala de descomposición, más energía se compactará en uno de dieciséis del total de los coeficientes wavelet.

*Sectorización Subbanda De Los Coeficientes Significativos:* Un coeficiente wavelet  $c$ , se llama significativo con respecto a un umbral predeterminado  $T$  si  $|c| \geq T$ , en otro caso se dice que no es significativo. Un coeficiente no significativo también se conoce como “coeficiente cero”. La ausencia de componentes de alta frecuencia en regiones homogéneas y la presencia de altas frecuencias en regiones texturizadas y alrededor de los bordes, significa que los coeficientes significativos en subbandas paso altas usualmente aparecen en localizaciones espaciales o texturas de alta energía. En otras palabras, esto indica una prominente discontinuidad o prominente cambio.

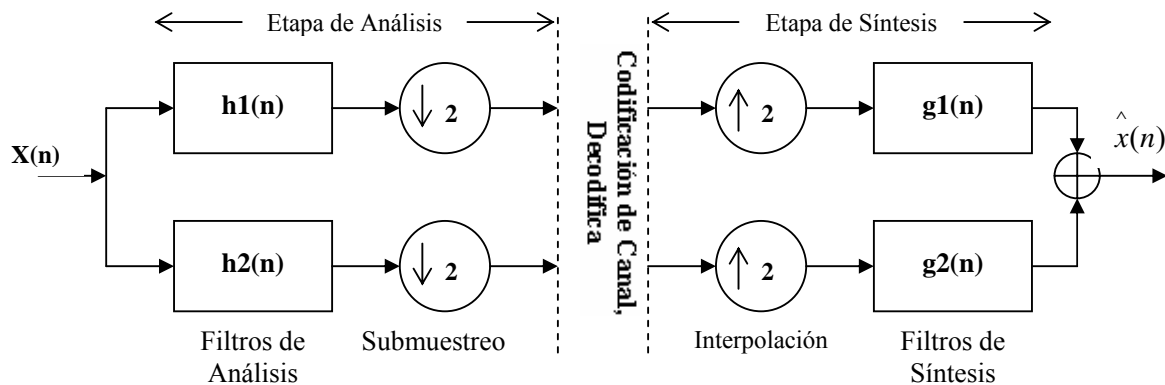
*Similitud De Subbanda Cruzada:*

Todos los coeficientes de escala fina que corresponden a la misma localización espacial se llaman “descendientes” acordadamente el coeficiente dado se llama “ancestro”. Específicamente el coeficiente de una escala anterior se llama padre y los cuatro coeficientes correspondientes a la misma localización espacial en la siguiente escala u orientación similar se llaman hijos. Aunque la correlación lineal entre los valores de padres e hijos es empírica. Experimentos muestran que la correlación de coeficientes de magnitud cuadrática de un hijo y el coeficiente de magnitud cuadrática de un padre tienen a ser de entre 0.2 y 0.6 con una fuerte concentración alrededor de 0.35.

*Decaimiento De Las Magnitudes En Los Coeficientes A Través De Las Subbandas:* Aunque es difícil de caracterizar, experimentos han mostrado que en la codificación de imágenes utilizando wavelets, las magnitudes de los coeficientes hijos generalmente son menores que las magnitudes de los coeficientes padres. Asumiendo el campo aleatorio de Markov como un modelo de una imagen, se puede probar que, estadísticamente, la magnitud de los coeficientes wavelet decae exponencialmente de un padre a un hijo, proveyendo un importante apoyo para los codificadores EZW, SPIHT y SLCCA. La codificación subbanda fue introducida por primera vez por Crochiere en 1976 en el contexto de la codificación por voz y fue aplicada a la codificación de imágenes por Woods y O’Neil en 1986. En la codificación subbanda la entrada de la imagen pasa a través de un conjunto de filtros pasabanda o

subbandas. Ya que una imagen pasabanda ha sido reducida a su ancho de banda comparada con la imagen original esta puede ser nuevamente muestreada o diezmada, a esto se le llama “análisis”.

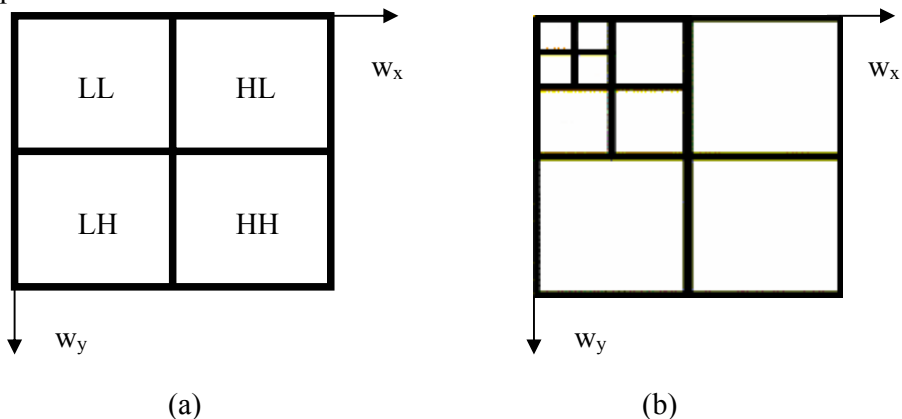
Las subbandas son cuantizadas y codificadas independientemente, al decodificar las subbandas se submuestrean (interpolan), filtrando y agregando todas las componentes para reconstruir la imagen, a esto se le conoce como la etapa de “síntesis”. La descomposición en subbandas no es un proceso de compresión en si mismo, por lo que el número total de muestras de las subbandas es igual al número total de muestras en la imagen original, esto se conoce como decimación crítica. El poder de este método reside en el hecho de que cada subbanda puede ser codificada eficientemente de acuerdo a su importancia estadística y visual. Este proceso lo podemos ver en la Figura 7.



**Figura 7.** Descomposición y reconstrucción de la señal en un codec subbanda.

Idealmente, la respuesta en frecuencia de los filtros pasa bajos y pasa altos no debe traslaparse, pero debe ser contigua y tiene ganancia unitaria sobre sus anchos de banda. En la práctica, los filtros no tienen una respuesta ideal por lo que algunas frecuencias se traslapan. El problema del traslapamiento es el “aliasing” que se introduce cuando las subbandas son submuestreadas. Un tipo de filtros que pueden resolver este problema son los filtros de espejo de cuadratura (QMF). En los QMF, los filtros son diseñados de manera que el aliasing introducido en la etapa de análisis es exactamente cancelado en la etapa de síntesis.

La descomposición en 1D puede ser ampliada a 2D de manera sencilla usando filtros por separado. En este caso, los filtros 1D pueden ampliarse primero en una dimensión y después en otra dimensión. Usando una descomposición en dos bandas para 1D en cada dirección resulta en cuatro subbandas: horizontal baja/vertical baja (LL), horizontal baja/vertical alta (LH), horizontal alta/vertical baja (HL) y horizontal alta/vertical alta (HH). Esta descomposición en cuatro bandas puede continuarse de manera repetitiva en una descomposición uniforme para todas las subbandas o sólo la subbanda LL en una descomposición no uniforme.

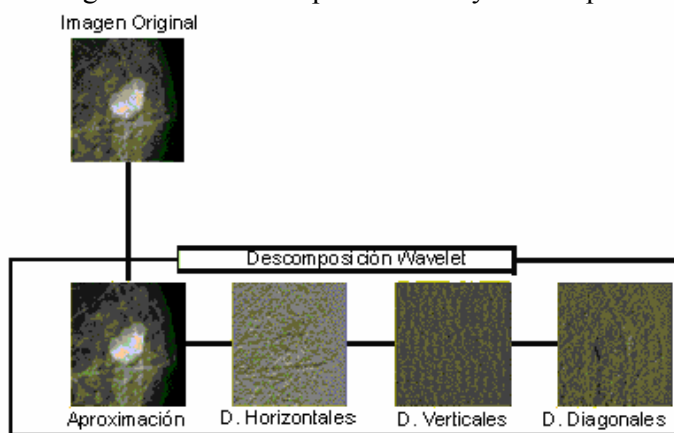


(a) Etapa de descomposición en 2-D (uniforme), (b) Etapa de descomposición 2-D (no uniforme)

La descomposición no uniforme resulta en una representación de la imagen de forma piramidal de multiresolución. Una técnica usada comúnmente para una descomposición no uniforme es la Transformada Wavelet Discreta (DWT). La DWT es una transformada que tiene la capacidad de operar a varias escalas y niveles de resolución. Al usar la DWT para descomposición, se tienen varios métodos para codificar las subbandas resultantes. Uno de los más eficientes algoritmos es el algoritmo de Shapiro de la wavelet cero-tres embebida (EZW). Este algoritmo asume que ya que es altamente probable que se tengan todos los coeficientes de la misma frecuencia espacial en la misma locación, si los coeficientes en la banda de baja frecuencia son cero, todas las altas frecuencias también serán cero, y por lo tanto descartadas. El algoritmo EZW codifica la información más importante primero y entonces codifica progresivamente las menos importantes. Por lo que se tiene un tren de bits que puede manejar un rango de tasa de transmisión por simple truncamiento.

Una ventaja de los sistemas de codificación de subbanda contra los sistemas de transformada por bloques es que estos no sufren de interferencia a bajas frecuencias. Una desventaja, sin embargo, es que a muy bajas tasas de bits, los efectos de “ringing” pueden inducir altos contrastes. Esto es debido al fenómeno de Gibbs para filtros lineales. Para evitar esto, la descomposición en subbanda se puede usar filtros no lineales. La transformada wavelet discreta (DWT) se emplea para obtener una nueva representación de la imagen, más apropiada para el proceso de compresión. La transformada wavelet consiste en comparar la señal con ciertas funciones wavelet, las cuales se obtienen a partir de las wavelet madre. La comparación permite obtener unos coeficientes que son susceptibles de interpretación y posterior manipulación. En cualquier caso, un requisito básico es la posibilidad de invertir la transformada, recuperando la señal a partir de esos coeficientes wavelet calculados.

El algoritmo clásico DWT de banco de filtros de Vetterli-Mallat [1,2,3] para la descomposición y reconstrucción de una señal en 1D de 1 etapa puede ser descrito como sigue: la señal de entrada  $x$  es filtrada y sobremuestreada siendo descompuesta en dos partes: una señal pasa-baja (LP) y una señal pasa-alta (HP). En el caso de la descomposición de una imagen, los pasos de procesamiento usando un algoritmo compuesto de un banco de filtros se representa por el esquema de descomposición subbanda y se describen como: 1) aplicar un filtro 1D sobre las señales LP y HP, submuestreando sobre las filas, 2) la parte pasa-bajas de la señal que fue filtrada sobre las filas, es pasada a través de dos filtros de 1D, un LP y un HP y submuestreada sobre las columnas. La parte pasa-alta de la imagen que fue filtrada sobre las filas, de nuevo es procesada con filtros de 1D un LP y un HP y submuestreada sobre las columnas. La Figura 8 muestra la aproximación y descomposición de una imagen.



**Figura 8.** Descomposición wavelet de primer nivel

La elección de las wavelets analizadoras juega un papel muy importante en los resultados finales. Entre las características más importantes a tener en cuenta se encuentran: soporte compacto, simetría, etc. Las wavelets bi-ortogonales son las más eficientes para un posterior proceso de compresión, y en particular, aquellas con pocos coeficientes, ya que el costo de obtención de los coeficientes se incrementa con su número.

Las ventajas del procesamiento de imágenes empleando la técnica de DWT, es que ahora se usa el estándar nuevo JPEG2000, que permite comprimir más las imágenes con capacidad de mejor calidad comparando con el estándar anterior basado en la técnica de DCT.

Por otra parte, las técnicas algorítmicas existentes para el DWT no son tan óptimos, como suponemos, es por eso, que se tratan de desarrollar las técnicas de DWT modificadas, como algoritmos adaptativos y no-lineales para mejorar la calidad de las imágenes descomprimidas, eliminando distorsiones específicas, tales, como el efecto de “ringing” que aparece en cercanía de las bordes de objetos en las imágenes.

**El algoritmo lifting 2D** En el caso de una señal discreta de 1D  $x = \{x_k\}$ ,  $k = 1, \dots, N$  y la wavelet más sencilla, Haar, puede ser descrita como sigue. La primera etapa consiste en dividir la señal en muestras pares e impares:  $\{d_j\}$  y  $\{s_j\}$ . Durante la segunda etapa, la predicción, las muestras impares son predichas usando interpolación lineal, como sigue:

$$d_j = d_j - s_j, j = 1, \dots, N/2. \quad (8)$$

Durante la tercera etapa las muestras pares son actualizadas también con el fin de preservar el valor promedio de las muestras. Para esto se usa la siguiente expresión:

$$s_j = s_j + \frac{1}{2}d_j, j = 1, \dots, N/2. \quad (9)$$

Los próximos niveles de descomposición de la DWT son obtenidos aplicando el esquema de lifting a los datos actualizados  $\{s_k\}$  de la señal original.

La DWT inversa para el esquema lifting para nuestro ejemplo es como sigue:

- actualización de datos inversa:  $s_j = s_j - \frac{1}{2}d_j, j = 1, \dots, N/2$
- predicción inversa:  $d_j = d_j + s_j, j = 1, \dots, N/2$
- composición de la señal de salida:  $x_{2i} = s_j, i = 2, \dots, N, j = 1, \dots, N/2,$   
 $x_{2i+1} = d_j, i = 1, \dots, N, j = 1, \dots, N/2.$

El ejemplo considerado presenta el esquema lifting de un nivel de descomposición y reconstrucción DWT para la bien conocida wavelet Haar, también conocida como la wavelet de Cohen-Daubechies-Feauveau de primer orden con un momento de desvanecimiento, o CDF(1,1) [7, 8]. La forma wavelet más popular de este tipo es la CDF(2,2), la cual posee dos momentos desvanecidos para ambas la wavelet primitiva y la wavelet dual. Las etapas de predicción y actualización de los datos del esquema lifting para el caso de wavelet CDF(2,2) son las siguientes:

$$d_j = d_j - \frac{1}{2}(s_j + s_{j-1}), j = 1, \dots, N/2, \quad (10)$$

$$s_j = s_j + \frac{1}{4}(d_j + d_{j-1}), j = 1, \dots, N/2. \quad (11)$$

Como fue indicado anteriormente, la ventaja principal del esquema lifting es que se requieren 2 veces menos operaciones que el algoritmo clásico de banco de filtros. Además, el esquema lifting permite redondear los valores en las etapas de predicción y actualización de datos. Esto permite operar con números enteros, lo cual es importante para las aplicaciones de compresión de datos.

**Generalización de filtrado en esquema lifting.** El proceso de la descomposición de una señal 1D empleando el esquema lifting (2,2) está ilustrando en la Figura 9.

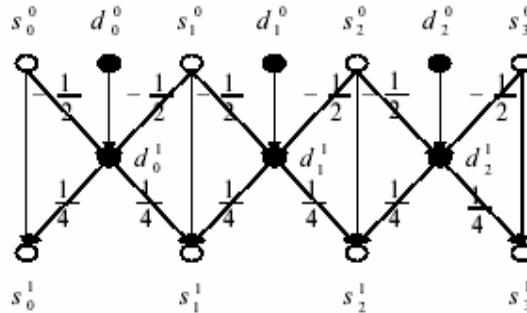


Figura 9. El diagrama de flujo de operaciones en el esquema lifting de orden (2,2).

Este proceso puede ser formalizado, como:

- división a muestras pares e impares:  $s_i = x_{2i}, d_i = x_{2i+1}$ .
- predicción:  $\hat{d}_i^{(k)} = d_i^{(k)} + p(s_{i-1}^{(k)} + s_{i+1}^{(k)})$ ,  $p = -\frac{1}{2}$
- actualización:  $\hat{s}_i^{(k)} = s_i^{(k)} + u(\hat{d}_{i-1}^{(k)} + \hat{d}_{i+1}^{(k)})$ ,  $u = \frac{1}{4}$

La función de transferencia de filtros paso altas (en la etapa de predicción) y paso bajas (en la etapa de actualización) se pueden escribir, como

$$H_p(z) = 1 + p(z + z^{-1}) \quad (12)$$

$$H_u(z) = 1 + 2up + u(z + z^{-1}) + up(z^2 + z^{-2}) = 1 + u(H_p(z) + H_p(z^{-1})) = 1 + H_p(z)u(z + z^{-1}) \quad (13)$$

Si consideramos el esquema lifting de orden mayor, por ejemplo, representado en la Figura 6 de orden (4,4), también podemos formalizar el proceso de descomposición, como

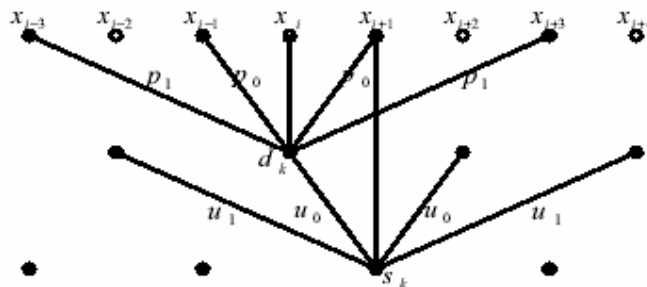


Figura 10. El diagrama generalizado de cálculos en el esquema lifting de orden (4,4).

- división a muestras pares e impares:  $s_i = x_{2i}$ ,  $d_i = x_{2i+1}$ .
- predicción:  $\hat{d}_i^{(k)} = d_i^{(k)} + p_0(s_{i-1}^{(k)} + s_{i+1}^{(k)}) + p_1(s_{i-2}^{(k)} + s_{i+2}^{(k)})$
- actualización:  $\hat{s}_i^{(k)} = s_i^{(k)} + u_0(\hat{d}_{i-1}^{(k)} + \hat{d}_{i+1}^{(k)}) + u_1(d_{i-2}^{(k)} + d_{i+2}^{(k)})$ .

Las funciones de transferencia correspondientes a formulas (18), (19) ahora se pueden escribir, como

$$H_p(z) = 1 + p_0(z + z^{-1}) + p_1(z^2 + z^{-2}) \quad (14)$$

$$H_u(z) = 1 + H_p(z)u_0(z + z^{-1}) + H_p(z)u_1(z^3 + z^{-3}) \quad (15)$$

Usando la misma técnica, llegamos al resultado importante para las funciones de transferencia en el esquema lifting de orden voluntario  $(\tilde{N}, N)$ :

$$H_p(z) = 1 + p_0(z + z^{-1}) + p_1(z^3 + z^{-3}) + \dots + p_{\tilde{N}-1}(z^{2\tilde{N}-1} + z^{-2\tilde{N}+1}) \quad (16)$$

$$H_u(z) = 1 + H_p(z)\{u_0[z + z^{-1}] + u_1[z^3 + z^{-3}] + \dots + u_{N-1}[z^{2N-1} + z^{-2N+1}]\} \quad (17)$$

La función de transferencia  $H_p(z)$  corresponde a un filtro paso altas de todo ceros. Para que este filtro sea el filtro pasa bajas, la condición necesaria es  $H_p(z) = 0$  en la frecuencia  $\omega = 0$ , que en el plano complejo  $Z$  corresponde al punto  $z = 1$ . Se puede encontrar fácilmente que esta condición  $H_p(z)|_{z=1} = 0$  cumple cuando

$$\sum_{i=0}^{\tilde{N}-1} p_i = -\frac{1}{2} \quad (18)$$

Con los valores de  $\{p_i\}$  que cumplen la condición  $H_p(z)|_{z=1} = 0$  dados por la ecuación (18), podemos encontrar  $H_p(-1) = 2$  y  $H_p(0) = 1$ , que significa, que el filtro de predicción tiene ganancia 2 en la frecuencia más alta  $\omega = \pi$  y una ganancia unitaria en la frecuencia  $\omega = \frac{\pi}{2}$ , la cual corresponde a la mitad de la banda de transición del filtro. Entonces, si se pide la respuesta en frecuencial del filtro normalizada, el factor de normalización es  $\frac{1}{2}$ .

Siguiendo la técnica del análisis considerada, los coeficientes del filtro de la actualización, que es un filtro paso bajas, se pueden encontrar. En esta manera, el filtro paso bajas  $H_u(z)$  debe tener un cero en el punto  $z = -1$  del plano complejo  $Z$  que corresponde a la frecuencia  $\omega = \pi$ . Esta condición cumple, cuando

$$\sum_{i=0}^{N-1} u_i = \frac{1}{4} \quad (19)$$

Cuando los coeficientes del filtro  $H_u(z)$  satisfacen la condición (19), entonces,  $H_u(1)|_{z=1} = 1$  que significa, que el filtro tiene ganancia unitaria en la frecuencia  $\omega = 0$ .

Usando la generalización del esquema lifting (22) (27), nosotros encontramos por simulaciones que los coeficientes de los filtros wavelet lifting de un orden arbitrario mayor que 4 se pueden encontrar usando las siguientes formulas recursivas:

$$p_0 = -\frac{128+a}{256}, \quad p_1 = -\frac{1}{2}p_0, \quad p_2 = -\frac{p_1}{c}, \quad p_3 = p_1 - p_2, \quad \dots, \quad p_{\tilde{N}} = -\frac{p_{\tilde{N}-1}}{c}, \quad p_{\tilde{N}-1} = p_{\tilde{N}-1} - p_{\tilde{N}} \quad (20)$$

$$u_0 = \frac{64+b}{256}, \quad u_1 = \frac{1}{4}u_0, \quad u_2 = \frac{u_1}{d}, \quad u_3 = u_1 - u_2, \quad \dots, \quad u_N = \frac{u_{N-1}}{d}, \quad u_{N-1} = u_{N-1} - u_{\tilde{N}} \quad (21)$$

donde los parámetros  $c, d$  controlan las características de los filtros. En esta manera, los filtros wavelet del orden arbitrario se pueden derivar. Las formulas (20), (21) usan parámetros  $a, b, c, d$  para controlar las características de los filtros lifting. Los parámetros  $a, b$  controlan el ancho de la banda de transición y los parámetros de control nuevos  $c, d$  controlan la suavidad de las bandads de paso y de supresión para prevenir la apariencia de los lóbulos laterales: con valores más grandes de  $a, b$  los valores de  $c, d$  tienden ser más grandes. El comportamiento del coeficiente  $d$  es del incremento rapido, y en la práctica el tiene los valores grandes y suficientemente altos para decir que la influencia de los terminos en  $H_u(z)$  del orden más alto que 3 se pueden negar. Por eso, en la práctica, se puede usar el filtro lifting update pasa bajoas del orden  $N = 4$  sin un ensanchamiento significativo de la banda de transición cprrespondiente: el ancho de esta banda de transición, como regla, se determina por las características frecuenciales del predictor lifting. Las figures en la sección de resultados muestran las respuestas en frecuencia de los filtros lifting de orden (10, 2) con  $a = 28, b = 8, c = 3$ .

**Método desarrollado de algoritmo adaptativo de transformada wavelet lifting para la compresión sin perdidas.** En esta sección del actual reporte representa los resultados obtenidos para la compresión de imágenes de prueba en términos de entropía normalizada para obtener valores de tasa de compresión en la cantidad de bits para representar cada píxel de la imagen, bpp. Estos resultados obtenidos muestran, que las imágenes que contienen las regiones planas, los mejores resultados se obtienen con el lifting de orden (2,2). Al contrario, la imágenes que contienen un número grande de detalles de pequeña escala y bordes de objetos abruptos, i.e., frecuencias espaciales altas, la mejor tasa de compresión se obtiene empleando el lifting de orden (4,4) y fueron encontrados los coeficientes óptimos  $a = -16$  y  $b = -8$ . Naturalmente, nosotros decidimos unir ambos filtros en un solo filtro adaptativo que puede operar con el orden (2,2) en las regiones planas y con el orden (4,4) en la vecindad de regiones de alta actividad de los datos locales.

El algoritmo adaptativo resuelto de la DWT que usa los ordenes de filtro de predicción  $\tilde{N} = 2$  y  $\tilde{N} = 4$  (o de orden superior de 4) se puede describir, como sigue:

- si la actividad local de los datos es pequeña, la decisión se toma en favor de orden  $\tilde{N} = 2$  del predictor, con  $a = 0$
- si la actividad local de los datos es alta, el predictor de orden  $\tilde{N} = 4$  y  $a = -16$  se aplica.
- las características del filtro de actualización no se cambian preservando  $N = 4$  y  $b = -8$ .
- el estimador de la actividad local de los datos es simple: si la magnitud de la diferencia de las aproximaciones  $|s_i^{(k-1)} - s_{i-1}^{(k-1)}|$  es alta, i.e., es más grande que un valor predeterminado de la umbral, el hipótesis es adoptado que la actividad local de los datos es alta y tenemos que usar el predictor del orden  $\tilde{N} = 4$ . En el caso contrario, el predictor del orden más pequeño de  $\tilde{N} = 2$  se aplica.

En esta manera, las propiedades de descomposición wavelet lifting se varían entre filtros lifting de (2,4) y (4,4).



**Método desarrollado del algoritmo adaptativo de transformada wavelet lifting para la compresión de imágenes con pérdidas.** El variante adaptativo del esquema lifting modificado se puede diseñar, como está representado en la Figura 11:

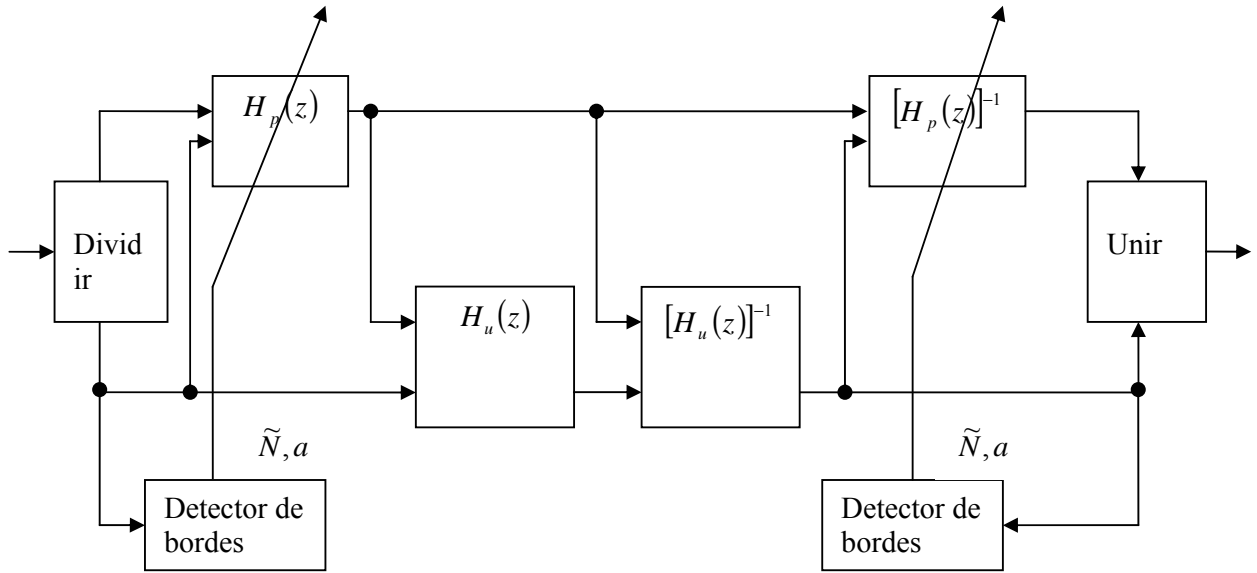


Figura 11. Técnica adaptativa de transformada wavelet lifting para la compresión de imágenes con pérdidas.

En este diagrama, valor  $\tilde{N}$  y el parámetro  $a$  del filtro predictor y del predictor inverso se cambian para cambiar sus respuestas impulsionales y sus propiedades espectrales correspondientes.

Después de varias pruebas, el detector de bordes fue implementado como sigue.

Los datos de entrada del renovador,  $s_i$  se filtran con el filtro  $H_p(z)$  modificado:

$$H_1(z) = -\frac{p_0(z + z^{-1}) + p_1(z^3 + z^{-3}) + \dots + p_{\tilde{N}-1}(z^{2\tilde{N}-1} + z^{-2\tilde{N}+1})}{0.5g_h} \quad (22)$$

Este filtro es el mismo filtro predictor, pero sin el cálculo de la suma con el valor de la muestra de la señal en la entrada del predictor. En esta manera, no se requieren los cálculos adicionales, la salida del filtro  $H_1(z)$  se puede usar para obtener la señal del filtro  $H_p(z)$ . El filtro  $H_1(z)$  en la esquema adaptativa tiene orden suficientemente grande,  $\tilde{N} = 20 \div 30$  y actúa como interpolador de alto orden para obtener una buena estima del valor de la muestra de la señal de entrada corriente del predictor,  $\hat{d}_1(i)$ .

La variación del orden  $\tilde{N}_1$  se puede implementar, usando la misma técnica, como en el caso del algoritmo desarrollado para la compresión sin pérdidas, este aspecto requiere las investigaciones adicionales para resolver los problemas conectados, la sincronización en particular.

Después, mismos datos de entrada del renovador,  $s_i$  se filtran con el filtro interpolador de primer orden:

$$H_2(z) = \frac{1}{2}(z + z^{-1}) \quad (23)$$

Las salidas de los filtros  $H_1(z)$  y  $H_2(z)$  se restan para formar el valor  $\Delta = \hat{i}_1 - \hat{i}_2$  que se usa como el parámetro de actividad local de los datos.

Finalmente, el valor absoluto de  $|\Delta|$  se compara con un umbral  $U$ , y si el valor  $|\Delta|$  es menor o igual a  $U$ , la salida del predictor es la resta  $\hat{d} = 0.5g_h d - \hat{i}_1$ ; en el caso contrario la salida del predictor se calcula con el filtro (23), pero con el orden menor de  $\tilde{N}_1$ .

Anotamos, que el procedimiento adaptativo descrito se aplica sólo para descomposiciones de las detalles verticales y horizontales, por que las detalles diagonales se calculan desde los coeficientes wavelet y ya no se cuentan con diferentes propiedades en el sentido de correlación en las regiones planas y bordes de la imagen. Además, fue determinado por las simulaciones que el procedimiento adaptativo se aplica sólo en dos primeros niveles de descomposición wavelet porque en el caso de compresión con perdidas los bordes de la imagen no siempre restauran bien y este causa errores de detección de éstas que provoca, en su cuenta, el nivel más bajo de relación señal a ruido en la imagen restaurada.

En mismo tiempo con el algoritmo adaptativo basado a la adaptación de las propiedades espectrales del filtro predictor con las propiedades de datos de la imagen, fue implementado el algoritmo adaptativo basado en la técnica conocida “update first” para adaptar las propiedades espectrales del filtro renovador, que en esta técnica filtra primero los datos, antes del predictor. En este algoritmo adaptativo, la adaptación del renovador fue implementada como sigue.

Primero, se utiliza el filtro interpolador  $H_1(z)$  de orden grande para producir la estima  $\hat{i}_1$  que es la mejor estima para regiones planas. Segundo, el interpolador de primer orden  $\hat{i}_2$  se utiliza para obtener la mejor estima en la cercanía de los bordes de la imagen. Los valores  $\hat{i}_1$  y  $\hat{i}_2$  se comparan con el valor de la entrada el renovador  $s_j$  y se elija el valor más cercano a éste,  $\hat{i}$ , para calcular la salida del renovador  $\hat{s}_j = 0.5g_h(s_j + \hat{i})$ . Como en el caso anterior, este procedimiento no se aplica para los detalles diagonales.

Los resultados de funcionamiento de esta modificación del esquema lifting adaptativo muestran, que el metodo de “update first” produce peor compactación de energía de la señal, causada por el aliasing introducido a los coeficientes wavelet que resulta en menor compresión de los datos y decremento de la relación señal a ruido.

**Método desarrollado del algoritmo adaptativo de transformada wavelet lifting IIR para la compresión de imágenes sin pérdidas.** Considerando el esquema lifting generalizado (16), (17) donde todos los ceros de los sistemas pueden ser modificados, para obtener las funciones de transferencia racional de una forma especial conteniendo ceros y polos como sigue:

$$H_p(z) = \frac{1 + p_0(z + z^{-1}) + p_1(z^3 + z^{-3}) + \dots + p_{\tilde{N}-1}(z^{2\tilde{N}-1} + z^{-2\tilde{N}+1})}{1 + a_{2p}z^{-2} + a_{4p}z^{-4} + \dots} \quad (24)$$

$$H_u(z) = \frac{1 + H_p(z) \{ u_0(z + z^{-1}) + u_1(z^3 + z^{-3}) + \dots + u_{N-1}(z^{2N-1} + z^{-2N+1}) \}}{1 + a_{2u}z^{-2} + a_{4u}z^{-4} + \dots} \quad (25)$$

En (24), (25) los denominadores contienen solo potencias impares de  $z$  antes de la salida del predictor y las fases de actualización indirectamente realizan el submuestreo de los datos y la función de transferencia que representa son expresadas en termino de la tasa de muestreo de datos de entrada.

Una condición específica para el predictor lifting es que él debe tener una ganancia fija para que cumpla la condición (19), es decir, para prevenir el sesgo en la salida del filtro actualizador en  $\omega=0$ . Esto puede ser realizado introduciendo la normalización por factor  $1-a_{2p}-a_{4p}-\dots$  en (25):

$$H_u(z) = \frac{1+H_p(z)\{u_0(z+z^{-1})+u_1(z^3+z^{-3})+\dots+u_{N-1}(z^{2N-1}+z^{-2N+1})\}}{(1-a_{2p}-a_{4p}-\dots)(1+a_{2u}z^{-2}+a_{4u}z^{-4}+\dots)} \quad (26)$$

Otro problema se hace cuando la transformada inversa se implementa con filtros IIR. Los filtros wavelets de análisis/síntesis deben proveer la restauración perfecta de los datos originales, que es especialmente importante para la compresión sin pérdida. En la tradicional técnica de descomposición/restauración diádica wavelet, el cuidado especial se toma para diseñar bancos de filtros ortonormales donde cada filtro satisface la restricción Nyquist  $|H_k(e^{j\omega})|_{\downarrow 2} = 1$ . A diferencia, el esquema lifting tiene un potencial para diseñar filtros wavelet biortogonales IIR en una manera simple: en la fase de restauración uno puede usar filtros predictores inversos y actualizadores inversos que operen con los elementos de la señal de entrada (coeficientes wavelets) reordenados a la inversa:

$$\begin{aligned} \mathbf{s}^{BT}(n) &= [s(n-N), s(n-N-1), \dots, s(0)], \\ \mathbf{d}^{BT}(n) &= [d(n-N), d(n-N-1), \dots, d(0)]. \end{aligned} \quad (27)$$

Y entonces filtrar a ellos con los filtros inversos

$$H_u(z) = \frac{1-H_p(z)\{u_0(z+z^{-1})+u_1(z^3+z^{-3})+\dots+u_{N-1}(z^{2N-1}+z^{-2N+1})\}}{(1+a_{2p}+a_{4p}+\dots)(1-a_{2u}z^{-2}-a_{4u}z^{-4}-\dots)} \quad (28)$$

$$H_p(z) = \frac{1-p_0(z+z^{-1})-p_1(z^3+z^{-3})-\dots-p_{N-1}(z^{2N-1}+z^{-2N+1})}{1-a_{2p}z^{-2}-a_{4p}z^{-4}-\dots} \quad (29)$$

para la síntesis y posteriormente hacer la reordenación de los datos filtrados:  $\{\cdot\}^B$ .

Después, queremos proceder con cálculo de números enteros cuando consideramos que esto es posible. Con este propósito, usaremos coeficientes normalizados  $a_{2p}$ ,  $a_{2u}$  como en (20):

$$a_{ip} = \frac{A_{ip}}{256}, \quad a_{iu} = \frac{A_{iu}}{256}. \quad (30)$$

Tomando en cuenta todos los resultados y restricciones antes mencionados, pueden formularse los filtros lifting IIR de enteros a enteros como sigue:

Etapa de análisis:

- predictor:

$$d_i^{(k)} = d_i^{(k-1)} + \left[ \frac{A_{2p}d_{i-2}^{(k-1)} + A_{2p}d_{i-4}^{(k-1)} + (b_p - 128)(s_{i-1}^{(k-1)} + s_{i+1}^{(k-1)}) + b_p(s_{i-3}^{(k-1)} + s_{i+3}^{(k-1)}) + \dots}{256} \right] \quad (31)$$

- actualizador:

$$s_i^{(k)} = s_i^{(k-1)} + \left[ \frac{A_{2u}s_{i-2}^{(k-1)} + A_{4u}s_{i-4}^{(k-1)}}{256} + \frac{(64-b_u)(d_{i-1}^{(k)} + d_{i+1}^{(k)}) + b_u(d_{i-3}^{(k)} + d_{i+3}^{(k)}) + \dots}{256 - A_{2p} - A_{4p}} \right] \quad (32)$$

Etapa de síntesis:

- actualizador inverso:

$$s_i^{B(k)} = s_i^{B(k-1)} \left[ \frac{A_{2l} s_{i-2}^{B(k-1)} + A_{4l} s_{i-4}^{B(k-1)} + (64-b_u)(d_{i-1}^{B(k-1)} + d_{i+1}^{B(k-1)}) + b_u(d_{i-3}^{B(k-1)} + d_{i+3}^{B(k-1)}) + \dots}{256} + \frac{(64-b_u)(d_{i-1}^{B(k-1)} + d_{i+1}^{B(k-1)}) + b_u(d_{i-3}^{B(k-1)} + d_{i+3}^{B(k-1)}) + \dots}{256 A_{2p} - A_{4p}} \right] \quad (33)$$

- predictor inverso:

$$d_i^{B(k)} = d_i^{B(k-1)} \left[ \frac{A_{2p} d_{i-2}^{B(k-1)} + A_{2p} d_{i-4}^{B(k-1)} + (b_p - 128)(s_{i-1}^{B(k)} + s_{i+1}^{B(k)}) + b_p(s_{i-3}^{B(k)} + s_{i+3}^{B(k)}) + \dots}{256} \right] \quad (34)$$

En las formulas (31)-(34),  $\lfloor \cdot \rfloor$  denota operación de redondeo, es decir tomar el valor entero más próximo.

Los coeficientes  $\{b_p\}, \{b_u\}$  son los que satisfacen a (18), (19). Adicionalmente  $\{b_p\}, \{b_u\}$ ,  $\tilde{N}, N$  y especialmente  $\{A_p\}, \{A_u\}$  son ajustados de una manera tal que los filtros (31), (32) mejoren las propiedades espectrales de los datos de la imagen, para minimizar lo que se conocerá como entropía de primer orden de los coeficientes wavelet

$$\{b_p\}, \{b_u\}, \tilde{N}, N, \{A_p\}, \{A_u\} \left\{ H(d) = -\sum_i p_i \log(p_i) \right\}, \quad (35)$$

donde  $p_i$  denotan la probabilidad de los diferentes valores de los coeficientes wavelet  $d$ . El problema de la optimización puede ser formulada ahora como el problema para minimizar los siguientes errores:

- el error cuadrático del predictor

$$\varepsilon_p = \sum_i [d_i^{(k)}]^2; \quad (36)$$

- el error cuadrático del predictor en el esquema lifting que usa actualizador seguido por el predictor

$$\varepsilon_u = \sum_i [\varepsilon_i^u]^2, \quad (37)$$

donde

$$\varepsilon_i^u = s_i^{(k-1)} \left[ \frac{A_{2l} s_{i-2}^{(k-1)} + A_{4l} s_{i-4}^{(k-1)} + (64-b_u)(\tilde{d}_{i-1}^{(k)} + \tilde{d}_{i+1}^{(k)}) + b_u(\tilde{d}_{i-3}^{(k)} + \tilde{d}_{i+3}^{(k)}) + \dots}{256} + \frac{(64-b_u)(\tilde{d}_{i-1}^{(k)} + \tilde{d}_{i+1}^{(k)}) + b_u(\tilde{d}_{i-3}^{(k)} + \tilde{d}_{i+3}^{(k)}) + \dots}{256 A_{2p} - A_{4p}} \right],$$

y

$$\tilde{d}_i^{(k)} = d_i^{(k-1)} \left[ \frac{A_{2p} d_{i-2}^{(k-1)} + A_{2p} d_{i-4}^{(k-1)} + (b_p - 128)(s_{i-1}^{(k-1)} + s_{i+1}^{(k-1)}) + b_p(s_{i-3}^{(k-1)} + s_{i+3}^{(k-1)}) + \dots}{256} \right]$$

es la salida del filtro pasa bajo de la “actualización primero”.

De esta manera, uno puede hallar la solución óptima

$$\min_{\substack{\{b_p\}, \{b_u\}, \\ \{A_p\}, \{A_u\}, \\ \{c_p\}, \{c_u\}}} \{ \varepsilon_p + \varepsilon_u \}$$

para cada nivel de descomposición. Desafortunadamente, es difícil obtener la solución analíticamente para los valores óptimos  $\{b_p\}, \{b_u\}$ ,  $\tilde{N}, N$ ,  $\{A_p\}, \{A_u\}$ , debido a la complejidad de las expresiones (36), (37).

**PRINCIPALES ESTÁNDARES DE COMPRESIÓN DE DATOS DE VIDEO.**  
**Códec de video h.261 para servicios audiovisuales a p x 64 kbit/s** describe los métodos de codificación y decodificación vídeo del componente de imagen en movimiento de los servicios audiovisuales a las velocidades de p x 64 kbit/s, donde p está comprendido entre 1 y 30. El codificador fuente actúa sobre imágenes basadas en un formato intermedio común (CIF, *common intermediate format*). Las normas de las señales de televisión de entrada y salida, que pueden, por

ejemplo, ser compuestas o de componentes analógicas o digitales, no son objeto de Recomendaciones, como tampoco lo son los métodos de realizar cualquier conversión necesaria de y hacia el formato de codificación de fuente. La Figura 12, muestra el diagrama a bloques del proceso de codificación haciendo uso del estándar H261.

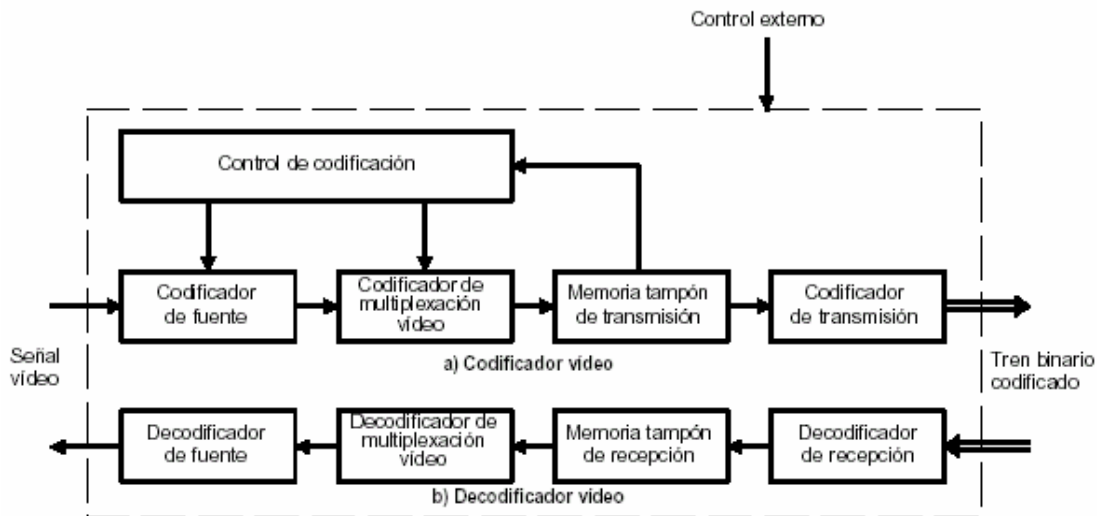


Figura 12: Diagrama de bloques resumido del codec de vídeo H.261.

El codificador de vídeo proporciona un tren binario digital autocontenido que puede combinarse con otras señales multifacilitades. El decodificador vídeo efectúa el proceso inverso. Las imágenes se muestrean a un múltiplo entero de la frecuencia de línea vídeo. Este reloj de muestreo y el reloj de red digital son asíncronos. Se adopta una combinación de predicción interimágenes para utilizar redundancia temporal y codificación de la transformada de la señal restante para reducir la redundancia espacial. El decodificador tiene la capacidad de compensación de movimiento, permitiendo la incorporación facultativa de esta técnica en el codificador. En cuanto a la velocidad binaria, esta recomendación está orientada fundamentalmente hacia la utilización de velocidades binarias vídeo entre unos 40 kbit/s y 2 Mbit/s.

El códec puede utilizarse para la comunicación visual bidireccional o unidireccional. En cuanto al tratamiento de los errores, en el tren de bits transmitido contiene un código BCH (*Bose, Chaudhuri Hocquengham*) (511/493) de corrección de errores sin canal de retorno. Su utilización en el decodificador es facultativa. A este estándar se le incluyen las características necesarias para el funcionamiento multipunto conmutado.

El codificador de fuente trabaja con imágenes no entrelazadas que aparecen 30 000/1001 (aproximadamente 29,97) veces por segundo. La tolerancia de la frecuencia de imagen es de  $\pm 50$  ppm. Las imágenes se codifican para obtener la componente de luminancia y las dos componentes diferencia de color (Y, CR y CB). Estas componentes y los códigos que representan sus valores muestreados son los que define la Recomendación 601 del CCIR.

- Negro = 16
- Blanco = 235
- Diferencia de color nula = 128
- Diferencia de color máxima = 16 y 240.

Estos valores son nominales y el algoritmo de codificación funciona con valores de entrada comprendidos entre 1 y 254. Se especifican dos formatos de exploración de imagen. En el primer formato (CIF), la estructura de muestreo de la luminancia es de 352 elementos de imagen por línea, 288 líneas por imagen, en una disposición ortogonal. El muestreo de cada una de las dos componentes de diferencia de color es de 176 elementos de imagen por línea, 144 líneas por imagen, ortogonal. Las muestras de diferencia de color se sitúan de manera que sus límites de bloque coincidan con los límites de bloque de luminancia. La zona de imagen cubierta por estos números de elementos de imagen y líneas tiene una relación de aspecto de 4:3 y corresponde a la porción activa de la entrada vídeo de norma local. El número de elementos de imagen por línea es compatible con el muestreo de las porciones activas de las señales de luminancia y diferencia de color de fuentes de 525 ó 625 líneas a 6,75 y 3,375 MHz respectivamente. Estas frecuencias tienen una relación simple con las de la Recomendación 601 del CCIR.

El segundo formato, de un cuarto de CIF (QCIF, *quarter-CIF*), tiene la mitad de elementos de imagen y de líneas que el formato anterior. Todos los codecs deben poder funcionar con QCIF. Algunos codecs pueden también funcionar con CIF. Deben preverse los medios necesarios para limitar el máximo periodo de transmisión de imagen de los codificadores, dejando de transmitir al menos 0, 1, 2 ó 3 imágenes entre las imágenes transmitidas. La selección de este número mínimo y de CIF o QCIF se hará por medios externos.

La predicción es interimágenes y puede aumentarse por compensación del movimiento y mediante un filtro espacial. La compensación del movimiento (MC, *motion compensation*) es facultativa en el codificador. El decodificador aceptará un vector por cada macrobloque. Las componentes horizontal y vertical de estos vectores de movimiento tienen valores enteros que no pasan de  $\pm 15$ . El vector se utiliza para los cuatro bloques de luminancia del macrobloque. El vector de movimiento para ambos bloques de diferencia de color se obtiene dividiendo por dos los valores de las componentes del vector de macrobloque y haciendo un truncamiento a cero de las partes de magnitud para producir componentes enteros.

Los vectores de movimiento están limitados de manera que todos los elementos de imagen por ellos referenciados estén dentro de la zona de imagen codificada. Para evitar la distorsión de cuantificación de las amplitudes de los coeficientes de transformada que cause desbordamiento aritmético en los bucles del codificador y decodificador, se insertan funciones de recorte. La función de recorte, se aplica a la imagen reconstruida que se forma sumando la predicción y el error de predicción modificados por el proceso de codificación. Este recortador actúa sobre los valores de los elementos de imagen resultantes inferiores a 0 o superiores a 255, cambiándolos a 0 y 255 respectivamente.

Pueden variarse algunos parámetros para controlar la velocidad de generación de datos del video codificado. Incluyen procesamiento antes del codificador de fuente, el cuantificador, criterio de significado de bloque y submuestreo temporal. Las proporciones de tales medidas en la estrategia de control global no son objeto de recomendación. Al ser invocado, el submuestreo temporal se realiza descartando imágenes completas. A manera de síntesis tenemos los siguientes datos referentes a este estándar:

Servicios Audiovisuales:

- $p \times 64$  kbits/s,  $p = 1, 2, \dots, 30$

Formatos de imagen:

- CIF progresivo
- 352 x 288 x 25, 4:2:0
- QCIF progresivo
- 176 x 144 x 25, 4:2:0
- 10 Hz
- Imágenes tipo I, P

- No entrelazado
- Estimación movimiento con precisión “full pixel”

**H.263. CODIFICACIÓN DE VÍDEO PARA COMUNICACIÓN A BAJA VELOCIDAD BINARIA** especifica una representación codificada, que se puede usar para comprimir el componente de imagen en movimiento de los servicios audiovisuales a velocidades binarias bajas. La configuración básica del algoritmo de codificación de fuente vídeo se basa en la Recomendación H.261, y es un híbrido de la predicción entre imágenes que utiliza la redundancia temporal y la codificación con transformada de la señal restante para reducir la redundancia espacial. En la figura 5.2.1 se muestra el diagrama a bloques de proceso de codificación con el estándar H 263.

El codificador de fuente puede funcionar con cinco formatos de fuente de vídeo normalizados: sub-cuarto de CIF, cuarto de CIF, CIF, 4 veces CIF y 16 veces CIF, y puede funcionar también utilizando una amplia gama de formatos de vídeo personalizados. El decodificador tiene capacidad de compensación de movimiento, lo que permite la incorporación opcional de esta técnica en el codificador. En la compensación de movimiento se utiliza la precisión de mitad de píxel, mientras que en la Recomendación H.261 se utiliza la precisión de píxel entero y un filtro de bucle. Se emplea la codificación de longitud variable para la transmisión de los símbolos.

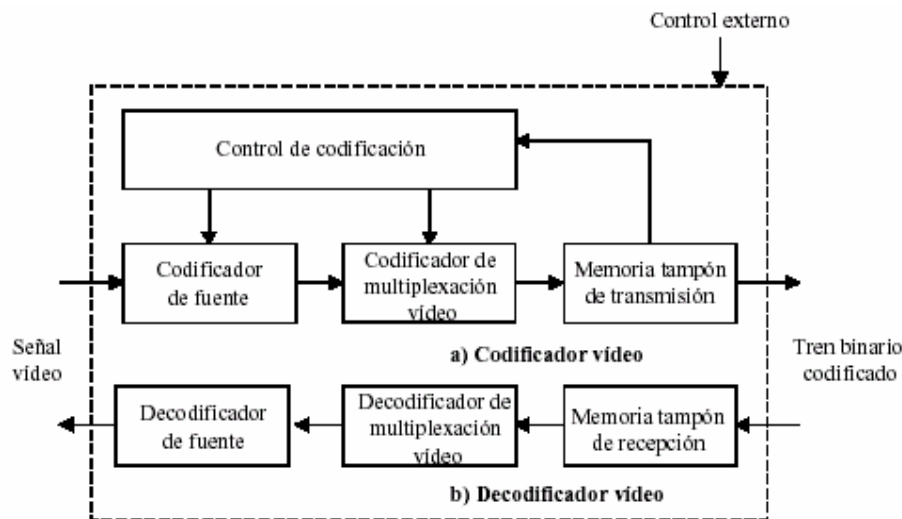


Figura 12. Diagrama de bloques resumido del códec de vídeo H.263.

Esta técnica especifica una representación codificada que se puede usar para comprimir el componente de imagen en movimiento de los servicios audiovisuales a velocidades binarias bajas. La configuración básica del algoritmo de codificación de fuente vídeo se basa en la técnica H.261. Se incluyen dieciséis opciones de codificación negociables para mejorar el funcionamiento y aumentar la funcionalidad. En este estándar existen cuatro opciones negociables para mejorar las prestaciones:

*Vectores de movimiento sin restricción:* En este modo los vectores de movimiento pueden apuntar a cualquier punto fuera de la imagen. En dicho caso, se utilizan los píxeles de la imagen como predicción de los píxeles inexistentes. Mediante este modo se consigue una ganancia significativa en aquellas secuencias de imágenes en las que exista movimiento en los bordes de la imagen. La ganancia es más significativa en los formatos de imagen más pequeños. Adicionalmente, este modo permite una extensión del margen de valores posibles de los vectores de movimiento, de forma que se pueden usar vectores más grandes (margen [-31,5 a 31,5]). Esto es especialmente útil en el caso de movimiento de la cámara o del fondo de la imagen.

Codificación aritmética basada en sintaxis: Consiste en usar codificación aritmética en lugar de codificación de longitud variable (VLC). La calidad de las imágenes decodificadas será la misma, pero la cantidad de bits necesarios para representarlas es generalmente menor (depende de la secuencia que se está codificando y de otros parámetros del codificador). En general, resulta un 5% menor.

Predicción avanzada: En algunos de los macrobloques de la imagen se usan cuatro vectores sobre bloques de 8 x 8 píxeles en vez de usar uno para el bloque de 16 x 16 píxeles, y se permite que los vectores apunten fuera de la imagen, como en el modo de vectores sin restricción descrito anteriormente. El codificador decide qué tipo de vectores usa. Utilizar cuatro vectores supone una mejor predicción, pero requiere más bits. El uso de este modo supone una mejora considerable, ya que se aprecia menos el efecto bloque en la imagen decodificada. Los vectores de movimiento correspondientes a los bloques C se obtienen a partir de los cuatro vectores de los bloques Y.

Predicción de cuadro hacia adelante o hacia atrás de forma similar a los cuadros P-E de MPEG: Un cuadro PE consiste en dos imágenes codificadas en una sola. Consiste en una imagen P predicha a partir de la última imagen P decodificada y una imagen E predicha simultáneamente a partir de la última imagen P decodificada y la imagen P que se está decodificando actualmente (figura 13). Se denomina imagen E puesto que partes de ella pueden ser predichas bidireccionalmente a partir de imágenes P pasadas y futuras. En el caso de secuencias de imágenes sencillas, este modo permite doblar la frecuencia de refresco (número de cuadros por segundo) sin incrementar demasiado la tasa de bits. Para secuencias con mucho movimiento, los cuadros PE no funcionan tan correctamente como las imágenes E de MPEG. Esto es debido a que no existen vectores bidimensionales separados en H.263, sino que se usan los vectores hacia adelante de las imágenes P escalados y sumados aun vector delta (corrección). La ventaja sobre MPEG es el uso de una información de cabecera menor para la parte de imagen E, lo cual es muy útil para las tasas de bits bajas y secuencias sencillas. En este sentido, hay que tener en cuenta que en las aplicaciones de videoconferencia las imágenes suelen ser sencillas. Cuando se usan estas opciones avanzadas se suele obtener la misma calidad que H.261 con menos de la mitad de tasa de bits. H.263 puede soportar cinco formatos de imagen, frente a los dos de H.261.

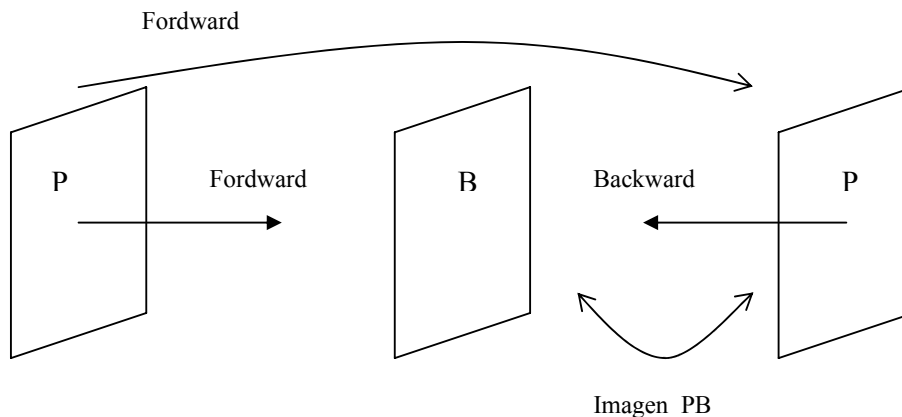


Figura 13. Predicción en modo imágenes de tipo P y B.

**H264. CODIFICACIÓN DE VIDEO AVANZADA PARA SERVICIOS AUDIOVISUALES GENÉRICOS.** El objetivo principal detrás del proyecto H.264 es desarrollar un estándar de codificación de video de alto rendimiento adoptando fundamentos básicos aproximados a un diseño simple y directo que usa bloques construidos ya conocidos. El grupo de expertos de video de la codificación de ITU-T (VCEG) inició el trabajo sobre el estándar H.26L en 1997. Hacia finales de 2001, se comprobó la superioridad de la calidad de video ofrecida por el excedente que alcanzó por la mayoría de las ofertas optimizadas basadas en MPEG-4, ISO/IEC ITU-T unido MPEG VCEG del software de H.26Lbased formando a un equipo video común (JVT) que asumió el control el proyecto



del H.26L del ITU-T. El objetivo de JVT es crear un solo estándar de video de la codificación que dé lugar simultáneamente a una nueva familia de estándares MPEG-4 y de una nueva recomendación de ITU-T (H.264). El trabajo de desarrollo del H.264 es una actividad en curso, con la primera versión del estándar esperado para ser concluido antes de 2002 y oficialmente antes de fin de año 2003. El estándar que emerge del H.264 tiene un número de características que lo distinguen de estándares existentes, mientras que al mismo tiempo, comparten características comunes con ellos.

Algunas de las características dominantes de H.264 son: ahorros de hasta un 50% en el índice binario, vídeo de alta calidad (incluyendo índices binarios bajos), adaptación a retrasos (usos de comunicaciones en tiempo real así como el almacenaje de video, aplicaciones basadas a flujo de video), herramientas de la resistencia de error para aplicaciones en pérdida de paquete en redes, y los errores de bit propensos en redes inalámbricas. Las líneas amigables de la red también está dirigido por este estándar con una separación conceptual entre una capa de codificación de video (VCL), la cual provee de la base de la representación de alta compresión del contenido de una imagen de video y una capa de la adaptación de la red (NAL), que empaqueta esa representación para la entrega sobre un tipo particular de red.

Las características dichas anteriormente se pueden traducir a un número de ventajas para diversos usos de video. El acercamiento subyacente de H.264 es similar a los estándares anteriores tales como H.263 y MPEG-4, y consiste en las cuatro etapas principales:

1. Dividiendo cada marco video en bloques de pixeles, para poder conducir el proceso del marco de video en el nivel del bloque usando la transformada DCT.
2. Explotando las redundancias espaciales que existen dentro del marco de video, cifrando algunos de los bloques originales transformados a través de la cuantización y codificación de la entropía (o codificación de longitud variable).
3. Explotando las redundancias temporales que existen entre los bloques en marcos sucesivos, de modo que solamente los cambios entre los marcos sucesivos necesiten ser codificados. Esto se logra usando la valoración y la remuneración del movimiento. Para cualquier bloque dado, una búsqueda se realiza previamente cifrando unos o más marcos para determinar los vectores de movimiento que entonces son utilizados por el codificador y el decodificador para predecir el bloque.
4. Explotando cualquier redundancia espacial restante que exista dentro del marco de video cifrando los bloques restantes, es decir, la diferencia entre los bloques de la original y la correspondiente al bloque predecida, para transformar otra vez, cuantización y codificación de la entropía. Desde el punto de vista de la codificación, las diferencias principales entre el H.264 y los otros estándares se resumen en la figura 5.3.1 a través de un diagrama de bloque del codificador. Del lado del movimiento estimación / compensación, H.264 emplea los bloques de diversos tamaños y formas, una valoración más alta del movimiento del pixel secundario de la resolución, y la selección múltiple del marco de la referencia. En el lado del transformador, H.264 utiliza un número entero basado transformada que aproxima el de la transformada DCT utilizado en estándares anteriores. En el H.264, la codificación de la entropía se puede realizar usando códigos los solos de una longitud variable universales (UVLC) tabula o con la codificación adaptante Contexto-basada de la aritmética binaria (CABAC).

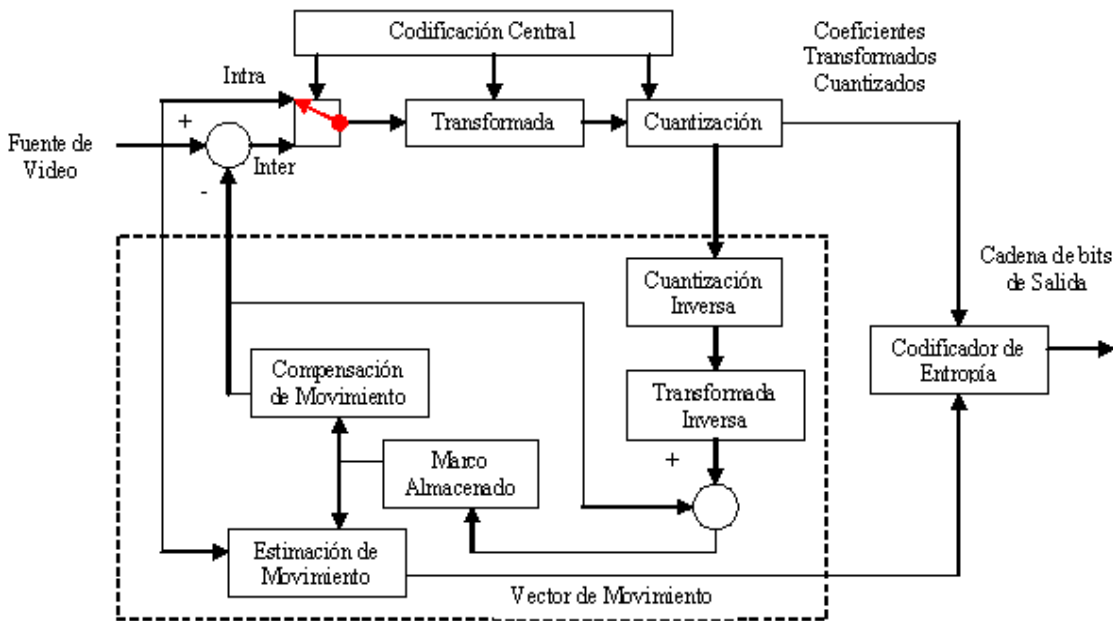


Figura 14. Diagrama de bloques resumido del codec de video H.264.

**TÉCNICA DE CODIFICACIÓN DE VIDEO PROPUESTA** usa transformada wavelet en lugar de la transformada DCT para la codificación de imágenes de marcos tipo I (marcos de referencia) y para la codificación de imágenes predichas en marcos de tipo P.

La gran importancia para cualquier técnica de codificación de video tiene el cálculo de los vectores de movimiento: más precisa es la estimación de movimiento, mayor sería la tasa de compresión de datos. Una técnica de referencia es la búsqueda exhaustiva que se realiza por buscar la mejor coincidencia del bloque de tamaño  $B_1 \times B_2$  en la vecindad de la posición de este bloque en la imagen actual en la imagen de referencia anterior (marcos tipo P) o en dos imágenes de referencia, en la anterior de la imagen actual y en la siguiente (marcos tipo B). La diferencia entre las estándares considerados para la compresión de video consiste en el tamaño de bloque y la región de la búsqueda: en H.261,  $B_1=B_2=16$  y el radio de búsqueda es  $r=15$ ; en el H.263,  $B_1=B_2=16$  ó  $B_1=B_2=8$  y el radio de búsqueda es  $r=31.5$ , ello significa que la búsqueda se realiza sobre la imagen interpolada en 2 veces, así, se minimiza el error de la predicción de movimiento, o se dice, que la búsqueda tiene una precisión de la mitad de píxeles (1/2 pixel accuracy). En el H.264 están especificados varios tamaños de bloques:  $B_1=B_2=16$ ,  $B_1=B_2=8$ ,  $B_1=B_2=4$ ,  $B_1=16$   $B_2=8$ ,  $B_1=8$   $B_2=16$ ,  $B_1=8$   $B_2=4$ ,  $B_1=4$   $B_2=8$  y la búsqueda se realiza sobre las imágenes interpoladas en 4 veces (1/4 pixel accuracy) con radio  $r$  variable.

En cuanto de la complejidad de la búsqueda, para realizar la búsqueda exhaustiva (FSA, full-search algorithm) se requiere para cada nodo  $M \cdot N \cdot r^2 B_1 B_2$  operaciones de restas y  $M \cdot N \cdot r^2 B_1 B_2$  ( $M$  y  $N$  son tamaños de imagen por filas y columnas) comparaciones de números enteros que produce una alta carga computacional. Por ejemplo, el software de referencia H.264 JM ver.10.2 (mas reciente accesible para los propósitos de investigación y evaluación en código de fuente en la dirección de Internet <http://iphome.hhi.de/suehring/tml/download/>) realiza la estimación de movimiento de un solo marco tipo P de tamaño QCIF (176x244) con el radio  $r=16$ , sin usar la estimación rápida, en

aproximadamente de 200 a 300 msec en la Pentium 4 3Ghz que es no aceptable para las aplicaciones prácticas (se requiere la tasa de codificación menos de 33ms para el video de 30 fps).

Existe una gran variedad de las técnicas de estimación de movimiento rápidas, entre ellos, se encuentran:

- algoritmos basados a conjunto reducido de candidatos de vectores de movimiento (TDL, two-dimensional logarithmic) que ponen el centro del patron de la búsqueda en el centro del mínimo encontrado en el paso anterior;
- técnicas de búsqueda completa rápida basadas a la medición reducida de la distorsión de bloque ó con el submuestreo del bloque de la búsqueda
- técnicas de búsqueda completa rápida que reducen la complejidad computacional por el costo de la calidad de predicción
- técnicas basadas al submuestreo del campo de movimientos de bloques (SDM, subsampled block-motion field)
- técnicas de búsqueda jerárquica (HSE, hierarchical search techniques) que realizan la búsqueda primero sobre el pirámide de las imágenes que se obtienen con el submuestreo de la imagen actual, se usan tres niveles de submuestreo, la búsqueda se realiza en la imagen de tamaño menor con el tamaño de bloque 4x4 y  $r=3$  y el resultado de la búsqueda se ajusta en cada nivel mayor, con bloques 8x8 y finalmente, 16x16 con el rango de búsqueda de  $r=1$ .

En el marco del presente proyecto, se propone un algoritmo de la búsqueda rápida que puede ser considerado como una combinación de las técnicas SDM y HSE. Primero, la búsqueda se realiza sobre las imágenes actual y la de referencia submuestreadas en cuarto veces, para el submuestreo se usa el filtro wavelet paso bajas conocidos como wavelet biorthogonal 9/7 (esta wavelet se usa en el estándar de compresión JPEG200). El tamaño de bloque es  $B_1=B_2=4$ , radio es  $r=16$  para la mayor exactitud de la búsqueda. En el paso siguiente, los vectores de movimiento obtenidos se tratan, como imágenes y para ellos, se aplica la interpolación de factor 4 por el sobremuestreo y el filtrado paso bajas con el mismo filtro wavelet. En el paso final, la búsqueda se realiza sobre imágenes de tamaño original, con  $B_1=B_2=4$  y es  $r=4$ . Así, se obtiene una alta calidad de predicción de movimiento y el tiempo reducido de la búsqueda comparando con las técnicas existentes.

Para codificar la secuencia de datos de video, se propone usar la DWT con pérdidas para imágenes de marco tipo I y P en parte de codificar las imágenes del error de predicción, y para codificar los vectores de movimiento que mencionamos que consideramos los como imágenes, se propone usar la transformada DWT lifting que permita la compresión sin pérdidas.

El esquema de bloques de codificador de video propuesto se presenta en la Figura 15.

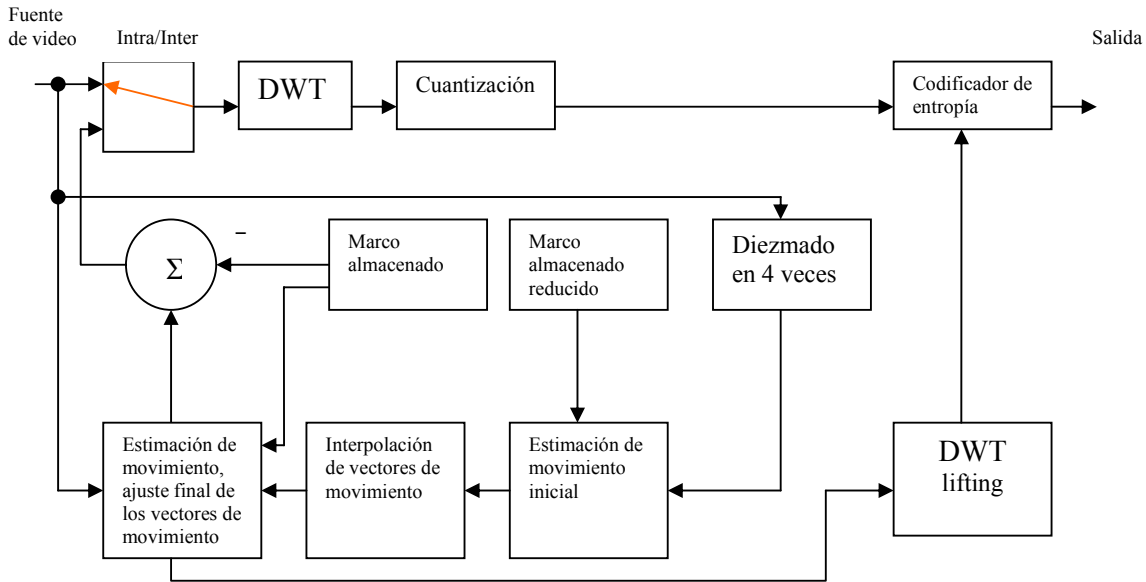


Figura 15. Diagrama de bloques del codificador de video propuesto.

De esquema presentada en la Figura 15 se puede ver, que el algoritmo propuesto de codificación de imágenes de video está basado al amplio uso de la transformada wavelets. La DWT se emplea para la compresión con pérdidas de las imágenes que forman los marcos de tipo I, además, se usa la DWT para la compresión de los errores de estimación del movimiento. Comparando con las figuras 12, 14, se puede concluir que la DWT se emplea en lugar de la técnica DCT, usada en los estándares H.263 y H.264. Además, para el tratamiento de los vectores de movimiento se puede usar la DWT lifting para realizar la compresión de dicha información sin pérdidas, en este caso, se propone el uso de los algoritmos presentados en este informe (31)-(34).

El punto crítico, del cual depende la complejidad computacional de cualquier técnica de compresión de video, es la estimación de movimiento que realiza la búsqueda de mejor coincidencia del bloque de la imagen actual con un bloque en la imagen previa en la secuencia de datos de video, realizando así la codificación de los marcos de tipo P. Para reducir la carga computacional, se propone el algoritmo de búsqueda basado al esquema jerárquico, con la distinción que se emplea la interpolación de las imágenes formados con los datos de vectores de movimiento, el nivel de interpolación se puede variar dependiendo del tamaño de bloque empleado para la búsqueda en el primer nivel y el tamaño de bloque en el nivel de tamaño de imagen de video actual. El filtro usado para la interpolación es el filtro pasa bajas wavelet. Además, la búsqueda en cada nivel se realiza empleando la técnica de búsqueda completa rápida llamada “successive elimination algorithm” (SEA). La técnica SEA se basa en la desigualdad matemática triangular dada por

$$\left| \sum_{i=1}^k a_i \right| \leq \sum_{i=1}^k |a_i|.$$

Con la norma SAD (sum of absolute differences) utilizada actualmente para realizar la búsqueda del movimiento, esta desigualdad se puede reescribir en la forma siguiente:

$$\left| \sum_{(x,y) \in \mathfrak{R}} |f_t(x,y)| - \sum_{(x,y) \in \mathfrak{R}} |f_{t-\Delta t}(x-i, y-j)| \right| \leq \sum_{(x,y) \in \mathfrak{R}} |f_t(x,y) - f_{t-\Delta t}(x-i, y-j)| = SAD(i, j),$$

donde  $f_t(x, y)$  son píxeles dentro del bloque  $\mathfrak{R}$  en la imagen actual, i.e., en el momento de tiempo  $t$  y  $f_{t-\Delta t}(x-i, y-j)$  son píxeles dentro del bloque  $\mathfrak{R}$  en la imagen anterior, i.e., en el momento de tiempo

$t - \Delta t$ , y desplazado a  $i, j$  en relación de la posición  $x, y$  en la imagen actual. En otras palabras, realizando la búsqueda del movimiento, el bloque en la posición  $i, j$  se puede omitir de la consideración completa, si

$$\left| \sum_{(x,y) \in \mathbb{K}} |f_t(x,y)| - \sum_{(x,y) \in \mathbb{K}} |f_{t-\Delta t}(x-i, y-j)| \right| \geq SAD(x,y). \quad (38)$$

El cálculo de las normas sumarias en la ecuación (38) tiene la complejidad mucho mas reducida que el cálculo de propio  $SAD(i, j)$ . Realizando el cálculo rápido de normas sumarias, se reduce significativamente el tiempo de la estimación del movimiento.

## **BIBLIOGRAFÍA**

1. S.G.Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol. 11, No. 7, p.p.674-693, 1989.
2. S.G.Mallat Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. Acoust. Speech Signal Process.*, Vol. 37, No. 12, p.p. 2091-2110, 1989.
3. Martin Vetterli, "Multi-dimensional sub-band coding: some theory and algorithms," *Signal Processing*, Vol. 6, pp.97-112, 1984.
4. W.Sweldens. The lifting scheme: A new philosophy in biorthogonal wavelet constructions. In A.F.Laine and M.Unser, editors, *Wavelet Applications in Signal and Image Processing III*, p.p. 68-79. Proc. SPIE 2569, 1995.
6. W.Sweldens. The lifting scheme: construction of second generation wavelets. *SIAM J. Math. Anal.*, Vol. 29, No. 2, p.p. 511-546, 1997.
7. A. Cohen, I.Daubechies, and J.Feauveau. Bi-orthogonal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, Vol. 45, p.p. 485-560, 1992.
8. G.Uytterhoeven and A.Bultheel. The Red-Black wavelet transform. In Proceedings of the IEEE Benelux Signal Processing Symposium, Leuven, Belgium, p.p. 191-194, March 1998.
9. R.Calderbank, I.Daubechies, W.Sweldens, and B.L.Yeo. Wavelet transforms that maps integers to integers. *Appl. Comput. Harmon. Anal.*, Vol.5, No. 3, p.p.332-369, 1996.
10. Oleksiy Pogrebnyak, Pablo Manrique Ramírez, "Fast algorithm of 2D byte-to-byte discrete wavelet transform," Proc. of the IASTED International Conference "Signal and Image Processing" (SIP2001), p.p.179-184, August 13-16, 2001, Honolulu, USA. ISBN: 0-88986-297-4. ISSN: 1482-7921.
11. Oleksiy Pogrebnyak, Juan Humberto Sossa Azuela, Pablo Manrique Ramírez "Fast algorithm of byte-to-byte wavelet transform for image compression applications", Proc. SPIE Vol.4789, Algorithms and Systems for Optical Information Processing VI, Bahram Javidi, Demetri Psaltis Chairs/Editors, 8 July - 12 Julio 2002, Seattle, USA, p.p.291-301. ISBN: 0-8194-4556-8, ISSN: 0277-786X.
12. Oleksiy Pogrebnyak, Pablo Manrique Ramírez, Juan Humberto Sossa Azuela. "Novel Wavelet Transform Algorithms for Image Coding". Proc. of the IASTED International Conference "Visualization, Imaging and Image Processing" (VIIP2002), p.p.213-218, September 9-12, 2002, Málaga, Spain. ISBN: 0-88986-354-3. ISSN: 1482-7921.
13. Proakis, J.G. y Manoakis D.G. Tratamiento digital de señales. 3ª Edición, PRENTICE HALL Int., Madrid, 1998.
14. Hoon Yoo and Jechang Jeong, A Unified Framework for Wavelet Transform Based on The Lifting Scheme, *Proc. of IEEE International Conference on Image Processing ICIP2001*, Tessaloniki, Greece, October 7-10, p.p.793-795, 2001.
15. Oleksiy Pogrebnyak, Pablo Manrique Ramírez "Adaptive wavelet transform for image compression applications" Proc. SPIE Vol.5203, Applications of Digital Image Processing XXVI, Andrew G. Tescher Chair/Editor, 5- 8 August 2003, San Diego, USA. p.p. 623-630, ISBN 0-8194-5076-6 , ISSN: 0277-786X.
16. Roger L. Claypoole Jr., Richard G. Baraniuk, Robert D. Nowak; "Adaptive wavelet transforms via lifting", Proc. ICASSP'98, pp. 1513 - 1516, May 1998.
17. G.C.K. Abhayaratne, "Spatially adaptive wavelet transforms: An optimal interpolation approach", Third International Workshop on Spectral Methods and Multirate Signal Processing (SMMSp) 2003, pp. 155-162, Barcelona, Spain, 12-13 Sept. 2003.
18. G. Piella and H. J. A. M. Heijmans, "An adaptive update lifting scheme with perfect reconstruction," *Proc. ICIP'01*, pp. 190 - 193, October 2001.
19. Mohammed E.Al-Mualla, C.Nishan Canagarajah, David R. Bull. "Video Coding for Mobile Communications". Academic Press, London, UK, 2002.

## Resultados.

### Resultados experimentales de las simulaciones de compresión de imágenes con el algoritmo desarrollado adaptativo de transformada wavelet lifting

La Tabla 1 presenta los valores de entropía de las imágenes de prueba descompuestas con los valores diferentes de los coeficientes  $a$ ,  $b$  y con diferentes valores de la umbral en el estimador de la actividad local de los datos para la toma de decisión sobre el orden del filtro predictor.

De los resultados presentados en la Tabla 1 sigue, que el algoritmo desarrollado adaptativo con la adaptación variable espacial posee la capacidad de la mejor compactación de la energía comparando con las características de la DWT lifting no adaptativa (4.23bpp y 6.25bpp para filtros lifting de orden (2,4) con  $a = 0$ ,  $b = -8$ , y 4.24 bpp y 6.28bpp para filtros lifting de orden (4,4) con  $a = -16$ ,  $b = -8$ , para imágenes de prueba “Lena” y “Mandrill”, respectivamente). Sorprendentemente, los valores óptimos del coeficiente  $b$  fueron menores que -8, el cual es óptimo en el sentido del ancho de banda de transición con la ausencia de lóbulos laterales de la respuesta en frecuencia. Esta tendencia se puede explicar por el hecho que la mitad de la banda de transición del filtro paso bajas actualizador se encuentra muy lejos desde el valor de la frecuencia ideal  $\frac{\pi}{2}$  conocido de la teoría de codificación subbanda, y, haciendo el valor  $b$  más pequeño, reducimos el aliasing causado por la no-optimalidad del filtro actualizador paso bajas.

En el mismo tiempo, si hacemos la banda de transición más estrecha con ayuda del valor menor del coeficiente  $a$ , la energía de la señal ensancha por el lóbulo lateral aparecido, que resulta en la entropía elevada de las imágenes transformadas.

El valor de la umbral en el algoritmo desarrollado de la DWT adaptativa se varia dependiendo de la naturaleza de la imagen: para las imágenes con la prevaencia de los regiones planas, este valor tiende ser pequeño que resulta en la prevaencia de filtrado de orden (2,4); par las imágenes con un numero grande de los detalles de pequeña escala y bordes abruptos, este valor se incrementa para aplicar con frecuencia el predictor del orden mayor,  $\tilde{N} = 4$  (ó  $\tilde{N} = 10$ ).

Tabla 1. Los resultados de simulaciones con el algoritmo desarrollado de la DWT adaptativa con filtros (4,4) y (2,4)

Valor de la umbral	<i>a</i>	<i>b</i>	Entropía de la imagen de prueba descompuesta "Lena", bpp	Entropía de la imagen de prueba descompuesta "Mandrill", bpp
8	-16	-8	4.211	6.273
		-16	4.208	6.265
		-32	4.203	6.249
		-48	4.198	6.237
		-64	<b>4.192</b>	6.226
	-32	-8	4.239	6.328
		-16	4.235	6.316
		-32	4.226	6.295
16	-16	-8	4.211	6.266
		-16	4.21	6.259
		-32	4.206	6.245
		-48	4.202	6.234
		-64	4.197	<b>6.225</b>
	-32	-8	4.217	6.305
		-16	4.215	6.295
		-32	4.209	6.277
32	-16	-8	4.222	6.259
		-16	4.221	6.252
		-32	4.218	6.241
		-48	4.215	6.232
		-64	4.211	<b>6.225</b>
	-32	-8	4.22	6.279
		-16	4.218	6.271
		-32	4.214	6.258

Tabla 2. Los resultados de simulaciones con el algoritmo desarrollado de la DWT adaptativa con filtros (10,4) y (2,4)

Valor de la umbral	a	b	(10,4)/(2,4) adaptive filter, c=3	
			Entropía de la imagen de prueba descompuesta "Lena", bpp	Entropía de la imagen de prueba descompuesta "Mandrill", bpp
8	16	8	4.212	6.275
		16	4.209	6.266
		32	4.204	6.251
		48	4.198	6.237
		64	<b>4.192</b>	6.225
	28	8	4.23	6.322
16	16	16	4.21	6.26
		32	4.206	6.246
		48	4.202	6.234
		64	4.197	<b>6.224</b>
		28	8	4.217
	16	16	4.211	6.292
32	16	32	4.205	6.273
		8	4.222	6.26
		16	4.221	6.253
		32	4.218	6.242
		48	4.215	6.232
	64	4.211	6.225	
28	8	4.219	6.278	
	16	4.218	6.27	
	32	4.214	6.257	

### Resultados experimentales de las simulaciones de compresión de imágenes con el el algoritmo desarrollado de transformada wavelet IIR lifting

Los algoritmos descritos en la sección anterior del presente informe fueron probados sobre conjunto de imágenes de prueba estándares de tamaño 512x512 pixeles: "Lena", "Baboon", "Barbara", "Boats", "Goldhill", "Peppers", "Bridge"

La tabla 3 presenta los valores de entropía en bits por píxel (bpp), obtenidos de estas imágenes cuando se les aplica la descomposición estándar lifting y la wavelet CDF(1,1) (wavelet Haar) con  $\tilde{N}=1$ ,  $N=1$ ,  $a=0$ ,  $b=0$ , la wavelet CDF(2,2) con  $\tilde{N}=2$ ,  $N=2$ ,  $a=16$ ,  $b=8$  (esta wavelet es usada por JPEG2000 para imágenes de compresión con pérdida), y filtros IIR lifting (31)-(34) con los mismos parámetros FIR y varios valores  $\{A_p\}, \{A_u\}$ . Los valores de  $\{A_p\}, \{A_u\}$  son aquellos que minimizan la entropía de primer orden de los coeficientes wavelet en el primer nivel de descomposición, en niveles superiores ellos fueron seleccionados como 0.

La tabla 4 presenta los valores de entropía en bits por píxel (bpp), obtenidos de las imágenes de prueba cuando se aplica la descomposición lifting generalizada (4), (6) y lifting IIR (31)-(34). Los valores de la parte FIR del esquema lifting fueron variados y los valores  $\{A_p\}, \{A_u\}$  fueron los mismos parámetros como en la simulación previa (ver Tabla 3).



Analizando la simulación de resultados presentados en la Tabla 1 y Tabla 2, uno puede concluir que la transformada lifting IIR propuesta funciona bien, proporcionando valores de entropía más bajos para todas las imágenes de prueba en comparación a los filtros lifting FIR. Mientras aumenta el orden  $\tilde{N}, N$  de los filtros lifting FIR y variando los coeficientes  $b_p, b_u, c_p, c_u$  de los filtros lifting FIR, sin el uso de los coeficientes ( $\{A_p\}=\bar{0}, \{A_u\}=\bar{0}$ ) de los filtros IIR, uno puede obtener una compresión de datos mas alta. La diferencia entre los filtros FIR y IIR en cuanto a optimización algunas veces es pequeña (por ejemplo, para la imagen Lena), pero en todos casos, la técnica de filtros IIR obtiene mejores resultados de compresión.

**Tabla 3.** Los valores de la entropía en bpp, para diferentes técnicas,  $\tilde{N}=2, N=2, b_p=16, b_u=8$  en casos de CDF(2,2) y lifting IIR con los correspondientes valores  $\{A_p\}, \{A_u\}$ .

Técnica	Imagen						
	Baboon	Lena	Barbara	Boat	Bridge	Peppers	Goldhill
CDF							
(1,1)							
lifting	6.16	4.41	5.09	5.01	3.793	4.72	4.898
CDF							
(2,2)							
lifting	6.14	4.36	4.94	4.98	3.792	4.71	4.885
IIR							
Lifting	<b>6.13</b>	<b>4.35</b>	<b>4.91</b>	<b>4.97</b>	<b>3.791</b>	<b>4.68</b>	<b>4.871</b>
$A_{2p}$	19	11	28	-11	0	-26	15
$A_{4p}$	7	-4	11	-3	0	9	16
$A_{2u}$	9	3	8	-19	3	5	-2
$A_{4u}$	-8	0	-8	-3	-5	3	-7

**Tabla 4.** Los valores de entropía en bpp para diferentes técnicas,  $\tilde{N}=8, N=8, \{A_p\}, \{A_u\}$ , están estos desde la Tabla 1,  $b_p, b_u, c_p, c_u$  fueron variados para lograr el mínimo de bpp.

Técnica	Imagen						
	Baboon	Lena	Barbara	Boat	Bridge	Peppers	Goldhill
Lifting							
genera-							
lizado	6.134	4.36	4.83	4.97	3.792	4.70	4.885
IIR							
Lifting	<b>6.125</b>	<b>4.35</b>	<b>4.82</b>	<b>4.96</b>	<b>3.79</b>	<b>4.68</b>	<b>4.869</b>
$b_p$	20	20	29	18	11	9	15
$b_u$	13	9	18	11	8	11	4
$c_p$	6	6	2	5	4	4	6
$c_u$	5	6	2	6	9	6	3

## Resultados experimentales de las simulaciones de compresión de imágenes de video con el algoritmo desarrollado.

Las simulaciones para obtener los resultados de la compresión con el codec propuesto, fueron realizados sobre el conjunto de secuencias de video de prueba “Foreman”, “Carphone”, “Container”, “Claire”, “Mother and daughter”, “News”, “Highway” de tamaño QCIF (176x144) disponibles en el Internet en el formato YUV 4:2:0. Para la evaluación comparativa, la presente sección contiene los resultados obtenidos para la compresión de los datos mencionados con el codec H.264. Con este propósito, el codec H.264 se puso en el régimen de funcionamiento adecuado al codec propuesto: solo marcos de tipo I y tipo P, la búsqueda del movimiento completa, la exactitud de la búsqueda de 1 pixel (i.e., sin usar interpolación alguna). Las simulaciones fueron realizados en la computadora PC Rentium4 3GHz. Las Tablas 5-7 presentan los resultados de simulaciones.

En la tabla 5 se puede encontrar la información sobre el comportamiento del codec de video de referencia, H.264, para comprimir diferentes secuencias de video. Este codec muestra una buena calidad de los datos descomprimidos (aproximadamente, de 35 a 37 dB), pero el tiempo de procesamiento es bastante elevado, como 115 msec por cada marco. Considerando la frecuencia de marcos de video 30 fps, empleada en el estándar NTSC, se puede concluir que para la compresión de datos de video en el tiempo casi real, se requiere una velocidad de 33 msec por marco, entonces, el codec H.264 que proporciona actualmente la mejor tasa de compresión y la calidad de datos descomprimidos, no se puede emplear para todas las aplicaciones de codificación de video en el tiempo real ó casi real.

Tabla 5. Resultados de prueba del codec H.264 en el modo de codificación sólo de marcos P, búsqueda completa (FS), rango de búsqueda 16

Secuencia de video, tamaño QCIF (176x144, 30fps)	Promedio de PSNR de componente Y, dB	Bits de componente Y per marco en promedio (taza de datos en kbps)	Tiempo medio de ME por marco, msec
Foreman	35.25	5754.56 (172.63 kbps)	113.86
Carphone	36.21	5057.22 (151.72 kbps)	114.05
Container	35.56	1925.21 (57.77 kbps)	115.94
Claire	38.97	1072.44 (32.17 kbps)	117.46
Mother and daughter	36.50	1355.84 (40.67 kbps)	118.15
News	35.88	2464.89 (73.94 kbps)	110.53
Highway	37.05	2187.32 (65.62 kbps)	117.94

La Tabla 6 presenta los resultados obtenidos con el codec de video propuesto funcionando en el modo de bloques de tamaño 16x16. Se puede observar que la calidad de los datos de video descomprimidos es casi igual a la de los resultados de codec H.264 en parte de procesamiento de los datos de video con alto grado de movimiento (secuencias “Foreman” y “Carphone”), mientras tanto la calidad de video con menor grado de movimiento es mucho más alta, aproximadamente mayor de 40dB. En mismo tiempo, la tasa de compresión de datos con el codec propuesto es mas alta en 1.5 – 5 veces comparando con el codec H.264. En la relación del tiempo de procesamiento, los valores presentados muestran, que el codec propuesto es capaz de funcionar en las aplicaciones, donde se requiere el procesamiento de los datos de video en el tiempo real o casi real (todos los valores no sobrepasan la umbral de 33 msec por marco).

Tabla 6. Resultados de prueba del codec propuesto para codificación sólo de marcos P, búsqueda completa rápida, rango de búsqueda 16, tamaño del bloque 16x16, 3 niveles de descomposición con DWT, cuantificación  $q=30$  y técnica MRWD para codificar los coeficientes wavelets

Secuencia de video, tamaño QCIF (176x144, 30fps)	Promedio de PSNR de componente Y, dB	Bits de componente Y per marco en promedio (taza de datos en kbps)	Tiempo medio de ME por marco, msec
Foreman	35.51	3853.81 (115.61 kbps)	24.91
Carphone	35.82	4541.73 (136.25 kbps)	21.64
Container	43.75	268.60 (8.06 kbps)	10.56
Claire	44.21	572.85 (17.19 kbps)	10.58
Mother and daughter	42.19	625.98 (18.78 kbps)	14.41
News	41.49	2080.43 (62.41 kbps)	8.57
Highway	36.72	1583.39 (47.50 kbps)	19.86

Cabe mencionar, que la calidad de los datos descomprimidos se depende mucho del tamaño de bloque empleado para la búsqueda del movimiento. En particular, el codec de referencia H.264 realiza la técnica de búsqueda con el tamaño del bloque que varía de 16x16 hasta 4x4. De la tabla 6 sigue, que en las situaciones de que los datos de video representan escenas con un alto grado de movimiento el codec propuesto con el tamaño del bloque 16x16 pierde unos 0.25 – 0.40 dB comparando con el H.264. Para mejorar esta característica, se puede emplear el tamaño del bloque 8x8 (especificado, en particular, por el estándar H.263). La tabla 7 representa los resultados de simulaciones de compresión de datos de video con el codec propuesto empleando el tamaño del bloque 8x8 para realizar la búsqueda del movimiento.

Tabla 7. Resultados de prueba del codec propuesto para codificación sólo de marcos P, búsqueda completa rápida, rango de búsqueda 16, tamaño del bloque 8x8, 3 niveles de descomposición con DWT, cuantificación  $q=30$  y técnica MRWD para codificar los coeficientes wavelets

Secuencia de video, tamaño QCIF (176x144, 30fps)	Promedio de PSNR de componente Y, dB	Bits de componente Y per marco en promedio (taza de datos en kbps)	Tiempo medio de ME por marco, msec
Foreman	35.92	2456.59 (73.69 kbps)	24.20
Carphone	36.24	3280.76 (98.42 kbps)	21.75
Container	43.78	250.41 (7.51 kbps)	14.77
Claire	44.26	471.84 (14.16 kbps)	14.26
Mother and daughter	42.596	380.92 (11.43 kbps)	18.55
News	41.42	1740.71 (52.22 kbps)	11.44
Highway	37.29	1097.14 (32.91 kbps)	23.48

Como se puede observar, en este régimen el codec propuesto ofrece mayor calidad de los datos descomprimidos, mayor tasa de compresión de datos y menor tiempo de procesamiento comparando con el codec de referencia H.264. En resumen, entonces, se puede concluir que reduciendo el tamaño del bloque para la búsqueda del movimiento, se mejora la calidad de los datos descomprimidos y se aumenta la tasa de compresión de información.

Pero, el empleo del tamaño del bloque 8x8 o menor se aumenta la cantidad de información requerida para describir el movimiento. Así, la Tabla 8 representa las cantidades de información necesarios para describir el movimiento.

Tabla 8. Cantidades de información del movimiento del codec propuesto y codec H.264 para codificación sólo de marcos P, búsqueda completa rápida, rango de búsqueda 16, bits per marco en promedio

Secuencia de video, tamaño QCIF (176x144, 30fps)	H.264	Codec propuesto, tamaño del bloque 16x16	Codec propuesto, tamaño del bloque 8x8	Codec propuesto, tamaño del bloque 8x8, en modo de codificación diferencial de campos de vectores de movimiento
Foreman	895.88	266.52	1336.10	1308.97
Carphone	918.43	271.65	1645.63	1820.72
Container	66.91	15.9	180.08	698.7
Claire	124.38	30.63	474.91	227.34
Mother and daughter	246.11	136.35	1231.58	1303.85
News	289.14	54.34	310.67	347.89
Highway	536.16	196.04	1315.11	1429.66

De los resultados presentados en la tabla 8 se puede concluir que la cantidad de información requerida para la descripción del movimiento se aumenta drásticamente con la reducción del tamaño de bloque de búsqueda del movimiento. Aunque con el tamaño del bloque 8x8 el codec propuesto ofrece mayor compresión de datos (bits de datos de imagen + bits de datos del movimiento), la reducción mayor del tamaño del bloque de búsqueda del movimiento, para obtener mayor calidad de datos descomprimidos, requiere las técnicas específicas para compresión de los datos generados por el proceso de la búsqueda del movimiento que puede ser el tema de la investigación posterior.

Por el último, tabla 9 presenta los resultados del empleo de la técnica jerárquica para la búsqueda del movimiento. Aunque la tasa de procesamiento de los datos con el codec propuesto es bastante alta para las aplicaciones de compresión en tiempo real o casi real, el empleo del tamaño de la imagen mayor que QCIF puede provocar las demoras inaceptables que resulta en la necesidad de reducir aun más el tiempo de procesamiento de cada marco de la secuencia de datos de video. De los resultados presentados en la tabla 9, se puede concluir que la técnica jerárquica permita reducir significativamente el tiempo requerido para la estimación del movimiento que es componente más costosa en el sentido computacional para la compresión de los datos de video. En mismo tiempo, la calidad de los datos descomprimidos no se empeora significativamente comparando con los resultados presentados en la tabla 7, y la cantidad de información requerida para la codificación no es mucho mayor comparando con los valores presentados en la tabla 7 y tabla 8.

Tabla 9. Resultados de prueba del codec propuesto para codificación sólo de marcos P, búsqueda rápida usando 2 niveles jerárquicos, rango de búsqueda en primer nivel 3, en segundo nivel 1, tamaño del bloque 8x8, 3 niveles de descomposición con DWT, cuantificación  $q=30$  y técnica MRWD para codificar los coeficientes wavelets

Secuencia de video	Promedio de PSNR, dB	Bits de componente Y per marco en promedio	Bits de información del movimiento per marco en promedio	Tiempo medio de ME por marco, msec
Foreman	35.47	3174.07	1337.46	5.21
Carphone	35.85	3890.58	1778.77	4.63
Container	43.73	298.97	259.75	5.0
Claire	44.26	503.46	990.75	4.54
Mother and daughter	42.35	435.88	1186.9	4.39
News	41.65	1888.35	365.36	2.50
Highway	36.97	1373.17	1320.57	4.85

### ***Conclusiones:***

En el marco del proyecto de investigación #20051850 varios algoritmos de compresión de las imágenes estáticas y dinámicas basados a la transformada wavelets fueron desarrollados. Los algoritmos desarrollados de compresión de imágenes estáticas se emplean para la compresión de imágenes en marcos I en las secuencias de datos de video. Para codificar las imágenes de marcos de tipo P en las secuencias de datos de video, fueron desarrollados los algoritmos nuevos que emplean la transformada wavelet, la búsqueda completa rápida y jerárquica. Los resultados obtenidos con el codec de video presentado en el informe actual muestran que el codec desarrollado produce mejores resultados para la compresión de los datos de video comparando con el mejor codec de video actual, H.264. Los resultados de simulaciones obtenidos muestran, que el codec desarrollado se puede emplear en varias aplicaciones de compresión de datos de video, incluso el procesamiento de video en el tiempo real o casi real.

En mismo tiempo, cabe mencionar que el codec de video desarrollado en el marco del proyecto no realiza toda la gama del procesamiento posible de los datos de video, así, como procesamiento de los marcos de tipo B, SP, SI, compresión de datos de tasa variante en el tiempo y fue probado sólo con las secuencias de datos de tamaño QCIF que puede ser el objetivo de la investigación en el futuro, así, como el desarrollo de la técnicas específicas de compresión de la información del movimiento y la corrección de los errores de transmisión de los datos comprimidos por un canal de transmisión de datos ruidoso.

## **Impacto.**

- En el presente trabajo se ha logrado el desarrollo de técnicas nuevas de compresión de imágenes digitales mediante la transformada discreta wavelet. Los algoritmos desarrollados se pueden usar en diferentes aplicaciones donde se requiere la compresión de imágenes estáticas o de video.
- Las técnicas desarrolladas de compresión son competitivas por el nivel de compresión logrado con los existentes.
- Entre técnicas desarrolladas se encuentran algoritmos de procesamiento de señales e imágenes digitales de punto fijo que facilita su implementación en el hardware DSP y aumenta la velocidad de procesamiento.
- El codec de video desarrollado se puede emplear en varias aplicaciones prácticas.