

Uso de la Transformada de Haar para el incremento de la eficiencia y rapidez de la Transformada Discreta de Karhunen-Loève en la compresión de imágenes con pérdidas

Mario Mastriani¹ y Juliana Gambini²

¹UNTref y ²UNGS

mmastriani@untref.edu.ar, mgambini@ungs.edu.ar

RESUMEN

La Transformada Discreta de Karhunen-Loève (TDKL) es la transformada óptima (en una estadística de segundo orden) por excelencia y resulta ser la más eficiente entre todas las existentes a la hora de descorrelacionar bandas multi e hiper-espectrales de satélites de teleobservación. Sin embargo resulta menos eficiente en algunas aplicaciones, como es el caso de la descorrelación intra-cuadros de los bloques en que se divide una imagen para su posterior compresión. En este trabajo se presenta una versión mejorada de la TDKL, que resuelve este problema. Las simulaciones demuestran que se ha alcanzado la misma eficiencia en la descorrelación intra-cuadro que en el caso inter-cuadro satelital.

Palabras clave: *Compresión de Imágenes con pérdidas, Transformada Discreta de Karhunen-Loève, y Transformada Discreta de Onditas.*

1. INTRODUCCION

Es sabido que la transformada óptima por excelencia empleada en procesamiento de señales e imágenes, es la Transformada Discreta de Karhunen-Loève (TDKL) [1], no obstante, presenta una serie de ventajas y desventajas en sus diferentes empleos, como las que se describirán en la próxima sección.

Además, es bien sabido la importancia de la compresión en general y de las imágenes en particular a la hora de transmitir y almacenar las mismas para las más diversas aplicaciones: satelitales, médicas, multimediales, etc [2, 3]. La compresión de imágenes implica en sus dos versiones con pérdidas (en inglés: lossy) y sin pérdidas (en inglés: lossless) un ahorro tanto de espacio en medios de almacenamiento como de tiempo en sistemas de transmisión analógicos y digitales [2, 3]. En otro orden de cosas, entre las desventajas de la TDKL para poder ser empleada regularmente en sistemas de compresión, se destaca su elevada complejidad computacional. Si este aspecto fuera sobrellevado (de alguna forma), el algoritmo de compresión resultante del uso de la TDKL tendría atributos excepcionales de compresión y calidad de la imagen final recuperada.

Por otra parte y como parte de un esfuerzo de la Comisión Nacional de Actividades Espaciales (CONAE) para desarrollar un algoritmo similar de compresión inter-banda (entre las bandas de una plataforma multi y/o hiper-espectral) [2, 3], e intra-banda (entre los bloques a ser dividida cada banda), se nos encomendó la creación de dicho algoritmo

de compresión para que fuera eficiente en ambos ámbitos, a los efectos de simplificar su embebido en la tecnología conocida como FPGA (del inglés, *Field Programmable Gate Array*), empleada para acelerar los tiempos de compresión-descompresión en el back-end del software a desarrollar. Este desarrollo cobraba particular relevancia dado que si el algoritmo fuera el mismo en los dos casos mencionados, el espacio ocupado en memoria de la FPGA se reduciría dramáticamente. Este era un largo anhelo nunca alcanzado por la industria de los satélites de teleobservación, en un contexto conocido como análisis multi e hiper-espectral [4, 5]. La imposibilidad residía fundamentalmente en el hecho de que cuando los mosaicos eran morfológicamente similares y además coincidentes en posición, es decir, sin desplazamiento lateral alguno entre ellos, la TDKL resultaba altamente eficiente a la hora de descorrelacionar los mosaicos, mientras que en el caso que o bien la morfología fuera diferente, o bien siendo similar existiera un desplazamiento de algún tipo, la eficiencia de descorrelación de la TDKL se degradaba drásticamente. En otras palabras, cuando existía una Similitud Morfológica Equi-Posicional (SMEP) entre mosaicos o bandas, la TDKL se comportaba eficientemente, en todo otro caso, no [2, 3].

En definitiva, la consigna consiste en generar de alguna manera artificialmente la elevación de este atributo conocido como SMEP a los efectos de poder emplear la TDKL en todos aquellos casos donde naturalmente no es aconsejable por su baja performance de descorrelación [2, 3].

El presente trabajo tiene como eje la solución del mencionado problema, para lo cual fue organizado de la siguiente manera: en la Sección 2 se desarrolla la TDKL, sus ventajas, desventajas, así como la formalización del SMEP y la transformada ondita de Haar; en la Sección 3 definimos las métricas que nos permitirán cuantificar el grado de SMEP y la calidad de la reconstrucción visual resultante del proceso de compresión/descompresión; la Sección 4 atañe a las simulaciones computacionales; y finalmente en la Sección 5 se desarrollan las conclusiones.

2. GENERACION DE SMEP

Comenzaremos describiendo las herramientas utilizadas, y que se encuentran involucradas en la generación del SMEP.

Transformada Discreta de Karhunen-Loève (TDKL)

De la Fig.1, las dimensiones de la imagen I son:

$n\mathbf{I}$ = número de filas de \mathbf{I}
 $nc\mathbf{I}$ = número de columnas de \mathbf{I}

La Fig.1 muestra el detalle de recolección y alineación de los píxeles de idéntica ubicación dentro de cada bloque. Aunque en este caso los bloques son de 4-por-4 píxeles (para simplicidad del dibujo). Dichos bloques pueden estar solapados o no. En este trabajo se trabajará con bloques no solapados.

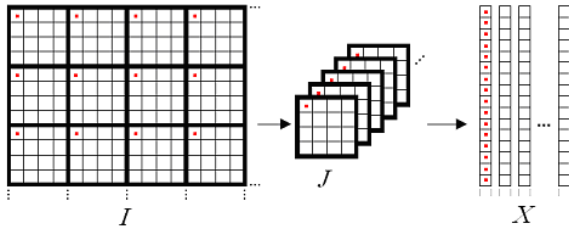


Figura 1: Recolección, alineación y vectorización del paquete de bloques

La mencionada recolección responde a algún criterio de exploración de los bloques, el cual se analiza en detalle en [2]. La etapa intermedia de alineación de los mosaicos permite construir una matriz tridimensional \mathbf{J} , en este caso de dimensiones:

nfm = número de filas de los mosaicos
 ncm = número de columnas de los mosaicos
 nm = número de mosaicos (profundidad) = $(n\mathbf{I} / nfm) \times (nc\mathbf{I} / ncm) = (512/4) \times (512/4)$

Posteriormente se exploran por filas los píxeles de cada bloque, constituyendo vectores entre valores correspondientes a posiciones idénticas, lo cual da lugar a una matriz bidimensional \mathbf{X} sobre la cual finalmente se aplicará la TDKL. La dimensión de dicha matriz para este caso es:

$n\mathbf{X}$ = número de filas de $\mathbf{X} = nm$
 $nc\mathbf{X}$ = número de columnas de $\mathbf{X} = nfm \cdot ncm$

Aplicación de la TDKL sobre \mathbf{X}

El procedimiento de aplicación de la TDKL sobre la matriz bidimensional \mathbf{X} comienza con la creación de la matriz de covarianza de la misma [1, 2, 4, 5], la cual resulta según:

$$\mathbf{C}_x = E\{(\mathbf{X}-\mathbf{m}_x)(\mathbf{X}-\mathbf{m}_x)^T\} \in \mathfrak{R}^{nm \times nm} \quad (1)$$

donde $\mathbf{X} = [x_1 \ x_2 \ \dots \ x_{nfm \cdot ncm}] \in \mathfrak{R}^{nm \times (nfm \cdot ncm)}$,

$x_k \in \mathfrak{R}^{nm \times 1}$ es su k -ésima columna,

para $k \in [1, nfm \cdot ncm]$

$E\{\bullet\}$ representa la esperanza matemática de “ \bullet ”

$\{\bullet\}^T$ representa la traspuesta de $\{\bullet\}$

$\mathbf{m}_x = E\{\mathbf{X}\}$ es el vector valor medio de \mathbf{X} y $\mathbf{m}_x \in \mathfrak{R}^{nm \times 1}$

En su apropiada forma matemática, tendremos

$$\mathbf{C}_x = \frac{1}{nfm \cdot ncm} \sum_{k=1}^{nfm \cdot ncm} (\mathbf{x}_k - \mathbf{m}_x)(\mathbf{x}_k - \mathbf{m}_x)^T \quad (2)$$

con:

$$\mathbf{m}_x = \frac{1}{nfm \cdot ncm} \sum_{k=1}^{nfm \cdot ncm} \mathbf{x}_k \quad (3)$$

Por lo tanto, la TDKL resultará

$$y_k = \mathbf{V}^T (\mathbf{x}_k - \mathbf{m}_x) \quad (4)$$

donde $\mathbf{V}^T \in \mathfrak{R}^{nm \times nm}$ y se trata de una matriz unitaria cuyas filas son los autovectores de \mathbf{C}_x y que la diagonaliza convirtiéndola en

$$\mathbf{C}_y = E\{\mathbf{Y} \mathbf{Y}^T\} \in \mathfrak{R}^{nm \times nm} \quad (5)$$

siendo:

$$\begin{aligned} \mathbf{Y} &= [y_1 \ y_2 \ \dots \ y_{nfm \cdot ncm}] \in \mathfrak{R}^{nm \times (nfm \cdot ncm)}, y_k \in \mathfrak{R}^{nm \times 1} \\ &\text{es su } k\text{-ésima columna, para } k \in [1, nfm \cdot ncm] \\ &= \mathbf{V}^T (\mathbf{X} - \mathbf{m}_x) \end{aligned} \quad (6)$$

Reemplazando (6) en (5) surge

$$\begin{aligned} \mathbf{C}_y &= E\{ \mathbf{V}^T (\mathbf{X} - \mathbf{m}_x) (\mathbf{V}^T (\mathbf{X} - \mathbf{m}_x))^T \} \\ &= E\{ \mathbf{V}^T (\mathbf{X} - \mathbf{m}_x) (\mathbf{X} - \mathbf{m}_x)^T \mathbf{V} \} \\ &= \mathbf{V}^T E\{ (\mathbf{X} - \mathbf{m}_x) (\mathbf{X} - \mathbf{m}_x)^T \} \mathbf{V} \\ &= \mathbf{V}^T E\{ (\mathbf{X} - \mathbf{m}_x) (\mathbf{X} - \mathbf{m}_x)^T \} \mathbf{V} \end{aligned} \quad (7)$$

mientras que, reemplazando (1) en (7), tenemos

$$\mathbf{C}_y = \mathbf{V}^T \mathbf{C}_x \mathbf{V} = \Lambda_y \quad (8)$$

donde $\Lambda_y \in \mathfrak{R}^{nm \times nm}$ y se trata de una matriz diagonal y unitaria que contiene los autovalores de \mathbf{C}_x en un orden decreciente, es decir, de mayor a menor y monótonamente decreciente, por lo que el primer autovalor será el mayor y el último el menor. Todos los autovalores son reales positivos y cumplirán con $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{nm} \geq 0$ [2].

Finalmente, y dado que todas las entradas de la matriz \mathbf{V} son reales, esta será entonces una matriz ortogonal, constituyendo sus columnas una base ortonormal en $\mathfrak{R}^{nm \times nm}$, al igual que sus filas. Los atributos precedentes establecen que $\mathbf{V}^{-1} = \mathbf{V}^T$, por lo que la TDKL inversa o TDKL⁻¹ será entonces

$$\mathbf{X} = \mathbf{V} \mathbf{Y} + \mathbf{m}_x \quad (9)$$

Propiedades de la TDKL [2, 3]

Ventajas

- Es óptima para decorrelacionar los coeficientes transformados
- Minimiza la representación total de la Entropía de la secuencia (la Entropía es la función de probabilidad con la que apare-

- cen los diferentes niveles de cuantización de la imagen).
- c. Permite una máxima compactación de la energía
- d. Minimiza el error de reconstrucción bajo una norma L_2 (útil para compresión).
- e. Permite reducir la dimensionalidad del proceso al permitir incluir exclusivamente a los componentes principales asociados a los autovalores más grandes y aun así minimizar el error de reconstrucción resultante bajo una norma L_2 . Este procedimiento de poda es la clave de la compresión de imágenes con pérdidas.

Desventajas

- a. Depende de la estadística de los bloques
- b. No se puede calcular eficientemente en forma distribuida para una imagen separada en bloques
- c. La matriz de la transformación no puede ser factorizada en matrices rasas.
- d. El cálculo de los autovalores y autovectores se realiza sobre la matriz C_x , y las dimensiones de ésta dependen de la cantidad de bloques, es decir, cuanto más bloques se agrupan, mayor es la dimensión de C_x y por ende la complejidad computacional asociada con este procedimiento. Como se verá en la próxima sección, cuanto más bloques se agrupan menor será el tamaño de los mismos y por ende mejor rendimiento de compactación tendrá la TDKL, es decir, la criticidad del procedimiento se da cuando mejor funciona la transformada.

Casos abiertos respecto a la TDKL

Es pertinente mencionar en este punto los casos no resueltos o considerados abiertos en relación a la TDKL, y que consisten en:

1. Mejorar el algoritmo de Jacobi (empleado en el cálculo de los autovalores y los autovectores), es decir, menor costo computacional, para obtener una TDKL rápida generalizada. Por otra parte, Anil K. Jain [1] trabajó en una versión rápida como un caso particular para imágenes cuya distribución de los valores de gris de los pixels puede aproximarse por una distribución Gaussiana.
2. Desarrollar una TDKL distribuida. Martin Vetterli [6] está trabajando en aproximaciones a un espacio lineal equivalente, para imágenes que también poseen una marcada distribución Gaussiana.
3. Crear una TDKL eficiente aplicada a una imagen aislada con mosaicos grandes (menor costo computacional) [2].

Este trabajo gira en torno del tercer problema resuelto en [2]

Finalmente, en el caso de la compresión de bandas multi e hiperespectrales en la industria aeroespacial se emplea el esquema de compresión de Tescher [4, 5] el cual incluye la TDKL para descorrelacionar dichas bandas, mientras que en lo que respecta a la compresión de cada banda en si misma se emplea el algoritmo JPEG [7, 8], aunque en la actualidad ha cobrado fuerza el uso del formato JPEG2000 [9].

Eficiencia de descorrelación de la TDKL

El problema reside en el hecho de que la TDKL es ineficiente a la hora de ser aplicada a la compresión de una imagen que es sub-dividida en mosaicos de idéntica dimensión, pero de diferente morfología. Mientras que cuando la TDKL es aplicada a bandas multi o hiperespectrales de una plataforma satelital de teleobservación la eficiencia es extremadamente alta, dado que dichas bandas representan a la misma parcela focalizada a distintas longitudes de onda [4, 5]. La clave se encuentra en la distribución de autovalores de la matriz de autocorrelación que se construye a partir de los mosaicos en los que se divide la imagen [2, 3], es decir, la diagonal de (8).

Análisis de la eficiencia de descorrelación

A modo de ejemplificación gráfica de este atributo de la TDKL aplicada a cada caso, podemos apreciar la Fig.2 para la cual, cada uno de sus mosaicos representa una porción de la imagen, no obstante son todos claramente distintos entre sí, lo cual redundaba en que si graficamos la diagonal de la matriz de (8) para este caso mediante una línea punteada en la Fig.3 esto equivale a una eficiencia de descorrelación entre mosaicos muy baja [2, 3], mientras que en la Fig.4, donde las bandas son morfológicamente similares y perfectamente coincidentes píxel-a-píxel (más allá del distinto nivel de brillo y contraste entre ellas debido a las diferentes longitudes de onda de cada una de ellas) la graficación de la diagonal de (8) para este caso la podemos apreciar en la línea continua de la Fig.3, lo que equivale a una eficiencia de descorrelación extremadamente alta [2, 3].

Atributo a ser evaluado

La clave de lo expresado reside en el grado de SMEP entre mosaicos o bandas [2, 3]. En la Sección 4 cuantificaremos este atributo mediante tres nuevas métricas, las cuales constituyen parte del aporte de este trabajo. Cuando este atributo es alto, dichas métricas están próximas a su máximo valor y la descorrelación de la TDKL es altamente eficiente. Mientras que cuando este atributo es bajo, las mencionadas métricas poseen valores moderados a bajo, siendo por ende la descorrelación de la TDKL bastante ineficiente.

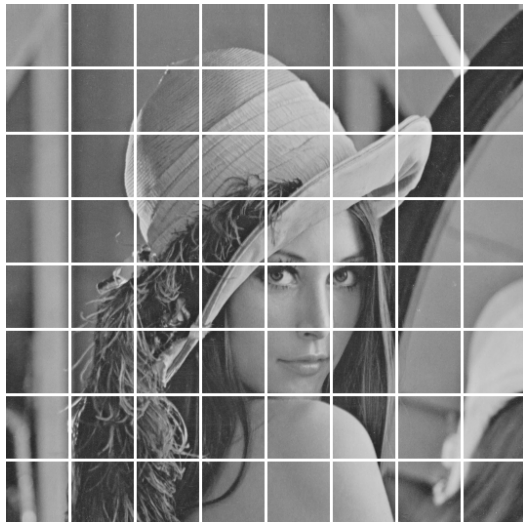


Fig.2. Imagen de Lena dividida en mosaicos de 64-por-64 píxeles.

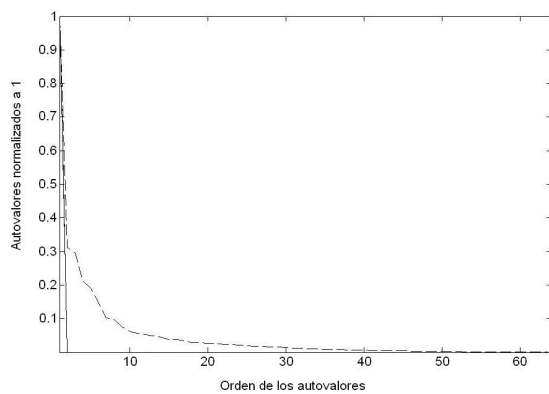


Fig.3. Espectro normalizado de autovalores. (a) Línea continua para los bloques de la Fig.2. (b) Línea de a tramos para las bandas de la Fig.4.

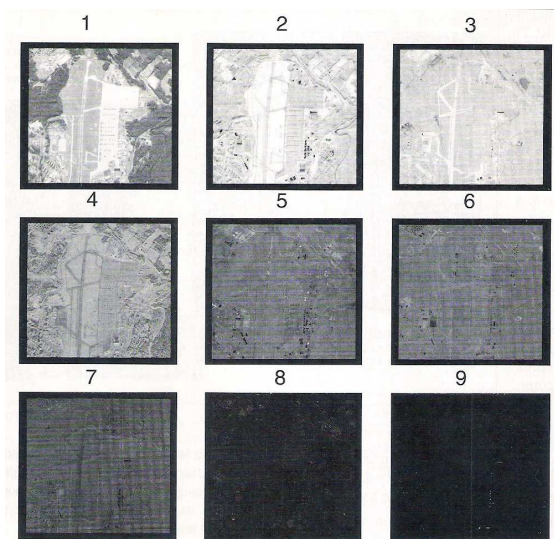


Fig.4. Bandas multispectrales

La eficiencia de descorrelación está íntimamente vinculada al espectro de autovalores, es decir, la diagonal principal de (8), resultante de aplicar la

TDKL a los mosaicos o bandas de las figuras anteriores, respectivamente [2], ver Fig.3.

En la Fig.3 se emplea el espectro normalizado, el cual consiste en dividir todos los autovalores del espectro por el primer autovalor, el cual es el de mayor tamaño de todos. La caída abrupta a cero del espectro representa una mejor eficiencia de descorrelación (línea continua) mientras que una caída a cero más suave del espectro se corresponde con una menor eficiencia de descorrelación (línea de a tramos). Una simple inspección visual y comparativa de las figuras 2 y 4 nos insinúa el atributo responsable de esta diferencia en la eficiencia de descorrelación, y tienen que ver con el contenido de SMEP de los mosaicos o bandas, es decir, cuando se da el SMEP la eficiencia de descorrelación es alta, mientras que cuando no se da es baja [2, 3]. La motivación de este trabajo reside en como provocar artificialmente SMEP en los casos donde no se da naturalmente como es el caso de imágenes monocuadro, y que es la razón por la cual no se usa hasta ahora la TDKL como parte de un esquema de compresión de las mismas [1, 3].

Cómo aumentar el SMEP artificialmente?

Del análisis del caso de la Fig.4 surge la idea de probar con alguna transformada que siendo aplicada antes de la TDKL, genere tantas bandas como bloques se desee y del tamaño requerido por estos, pero que además todos los bloques posean Similitud Morfológica Equi-Posicional, es decir, SMEP [2, 3]. Unas candidatas que parecen reunir estas dos condiciones son las onditas en general, y las onditas de Haar en particular, especialmente por su sencillez [2, 3, 10, 11].

Onditas de Haar

Representan las más simples bases de onditas para transformadas de onditas discretas [2, 3].

En Análisis Funcional, la ondita de Haar es una determinada secuencia de funciones. Se la conoce como la primera base de onditas. Esta secuencia fue propuesta en 1909 por Alfréd Haar [10]. Haar usó estas funciones para dar un ejemplo de un sistema ortonormal cuantificable para el espacio de funciones integrables cuadráticas sobre el eje real. El estudio de las onditas, fue adoptado mucho después. Haar es un caso especial de las onditas de Daubechies, y es conocida como D2.

La ondita de Haar es además la ondita más simple posible. La desventaja técnica de la ondita de Haar reside en el hecho de que no es continua, y por lo tanto no diferenciable, ver Fig.5. Esta propiedad puede, no obstante, ser una ventaja para el análisis de señales con transiciones abruptas, tales como las señales de monitoreo de fallas de herramientas en motores [10].

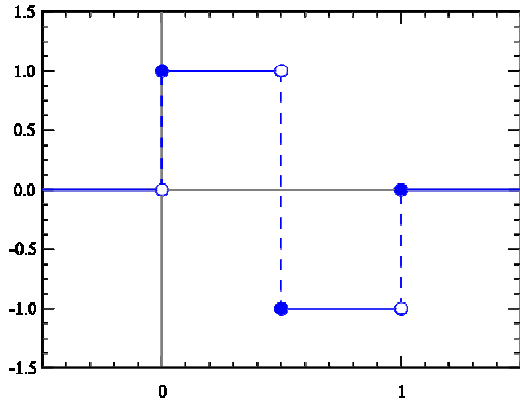


Fig.5. La ondita de Haar.

La función madre de ondita de Haar $\psi(t)$ puede ser escrita como

$$\psi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2}, \\ -1 & \frac{1}{2} \leq t < 1, \\ 0 & \text{todo otro caso.} \end{cases} \quad (10)$$

y su función de escala $\phi(t)$ puede ser escrita como:

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1, \\ 0 & \text{todo otro caso.} \end{cases} \quad (11)$$

Empleando Haar para producir SMEP

Precisamente, el empleo de la transformada de Haar [2, 10, 11] permitirá superar la desventaja (d) de la TDKL, de manera tal que el codec y decodec quedarán [2]:

CODEC:

1. Sistema generador de SMEP (el cual subdivide a la imagen en bloques).
2. TDKL
3. Evaluación y poda
4. Cuantización
5. Compresión entrópica

canal o almacenamiento

DECODEC:

1. Descompresión entrópica
2. Decuantización
3. Completar con ceros (zero-padding)
4. $TDKL^{-1}$ (TDKL inversa)
5. Sistema restaurador de SMEP (el cual reensambla la imagen a partir de los mosaicos).

El punto 1 del CODEC consiste en la aplicación recursiva de la transformada de Haar a todos los cuadrantes resultantes de su aplicación anterior y la organización de los mismos mediante el barrido de

Morton [2, 3], el cual consiste en un barrido con la forma de una Z de los bloques resultantes y una nueva aplicación con esa secuencia de la transformada de Haar. La grilla resultante de este proceso, se grafica en la Fig.6

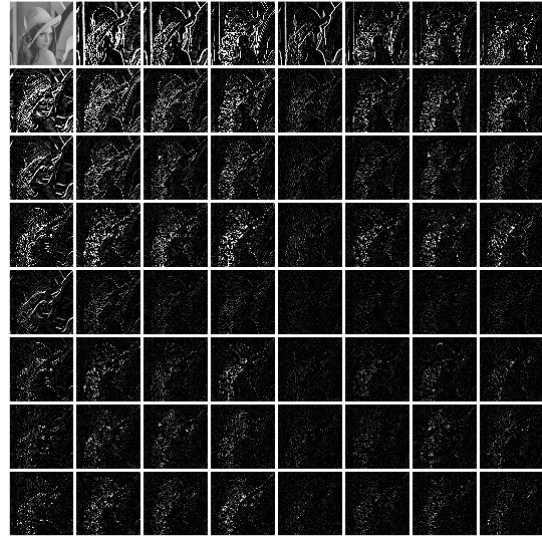


Fig.6. Grilla para bloques aplicando Haar.

De la Fig.6 se puede observar el SMEP entre los bloques, siendo el espectro de autovalores resultante, muy similar al de la Fig.4 en línea llena, donde solo el primer autovalor posee un valor relevante, mientras que el resto son prácticamente cero, lo cual es una condición imprescindible para que se puedan podar estos últimos sin impactar en la calidad final de la imagen recuperada luego del DECODEC. Por otra parte, el SMEP permite obtener los resultados de calidad de imagen que se mostrarán en la sección siguiente manteniendo grande el tamaño de los mosaicos, lo cual implica una menor cantidad de ellos que a su vez se traduce en un bajo costo computacional, dado que este último depende en la TDKL de la cantidad de mosaicos, por lo que cuando más grandes puedan ser estos, más pequeña será la matriz de autocorrelación C_x , y por ende, menor el costo computacional resultante. En definitiva es el SMEP el que hace a la TDKL rápida y eficiente [2, 3].

3. METRICAS

Definimos 2 grupos de métricas:

- a) de calidad de reconstrucción, y
- b) de contenido de SMEP,

a) De calidad de reconstrucción:

1. Mean Squared Error (MSE)

Se define para dos imágenes monocromáticas I e I_d de dimensiones $n_f l \times n_c l$, donde la primera es la imagen original, mientras que la segunda es la resultante del proceso de compresión/descompresión con pérdidas [1]

$$MSE = \frac{1}{nfI \times ncI} \sum_{nf=1}^{nfI} \sum_{nc=1}^{ncI} \left\| I(nf, nc) - I_d(nf, nc) \right\|^2 \quad (12)$$

2. Peak Signal-to-Noise Ratio (PSNR)

Es comúnmente usada como una medida de la calidad de reconstrucción en la compresión de imágenes [1]. Se define más fácilmente via el MSE

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) = 20 \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \quad (13)$$

Donde, MAX_I es el valor máximo de un pixel de la imagen.

3. Tasa de Compresión (TC)

Esta definida como la tasa entre la *Dimensión de los Datos Sin Comprimir (DDSC)* y la *Dimensión de los Datos Comprimidos (DDC)* [2]:

$$TC = \frac{DDSC \text{ [bytes]}}{DDC \text{ [bytes]}} \quad (14)$$

4. Número de bits por pixel (bpp)

Se define como

$$bpp = \frac{\text{número de bits codificados}}{\text{número de pixeles}} \quad (15)$$

No obstante, y a los efectos de cuantificar rápidamente el grado de SMEP, vamos a definir tres métricas adicionales, las cuales inequívocamente expresaran el grado de SMEP en cada caso.

b) De contenido de SMEP:

1. First Gap Percent (FGP):

Esta primera métrica se define como:

$$FGP = \left(1 - \frac{\lambda_2}{\lambda_1} \right) \times 100\% \quad (16)$$

donde λ_2 es el segundo autovalor, y λ_1 es el primer autovalor. Dado que el espectro de autovalores en monótonamente decreciente, si el SMEP es alto, esta diferencia debería ser porcentualmente elevada, como es el caso de la Fig.6, y se correspondería con la línea llena de la Fig.3, donde el primer autovalor normalizado es 1, mientras que el segundo se confunde con el eje de absisas. Mientras que en el caso de la Fig.2, al ser los mosaicos diferentes, este gap es porcentualmente bajo, por lo que λ_2 se encuentra cerca de λ_1 como muestra la línea segmentada de la Fig.3. La misma figura nos muestra que un gran

número de autovalores tienen valores considerablemente por encima de cero, lo cual no nos permitirá una compresión eficiente mediante poda, si queremos deshacernos de los mosaicos que menos aportan a la imagen final al momento de la reconstrucción [2, 3].

2. First vs Rest Percent (FRP):

Esta segunda métrica se define como:

$$FRP = \left(1 - \frac{\lambda_2 - \lambda_N}{\lambda_1} \right) \times 100\% \quad (17)$$

y nos da la noción del peso de λ_1 respecto de la diferencia entre λ_2 y λ_N . Esta métrica será crítica a la hora de evaluar la tasa de compresión en términos del porcentaje de poda de los autovalores menos significativos.

3. First Percent (FP):

Esta última métrica se define como:

$$FP = \frac{\lambda_1}{\sum_{i=1}^N \lambda_i} \times 100\% \quad (18)$$

y nos da la noción del peso de λ_1 respecto del espectro completo. Será particularmente útil a la hora de evaluar tasas de compresión extrema [2, 3].

No obstante, la pregunta que subyace es: se podrá alcanzar un SMEP artificial donde las tres últimas métricas tengan valores próximos a 100%?

3. SIMULACIONES COMPUTACIONALES

En esta sección compararemos:

1. TDKL con y sin SMEP
2. TDKL vs. JPEG y JPEG2000

en base a las métricas definidas en la sección anterior y que permiten cuantificar la calidad de reconstrucción de la imagen que atraviesa el proceso de compresión/descompresión.

En la Simulación 1 compararemos el uso de la TDKL en la que se dividirá la imagen en bloques como se indica en la Fig.2 sin la aplicación de la ondita de Haar versus el caso en el que si se aplica Haar, como en la Fig. 6 para compresión y descompresión. De esta forma, podremos apreciar el aporte al aumento del grado de SMEP introducido por el uso de dicha ondita. En la Simulación 2, en cambio, compararemos los resultados de aplicar la combinación Haar + TDKL a una imagen como la de la Fig.6, como se lo describe en el códec/decodec de la Sección 2 versus tratar su compresión con JPEG y JPEG2000 [2, 3, 7, 8, 9].

En ambas simulaciones evaluaremos el error pixel-a-pixel entre la imagen original y la recuperada.

Simulaciones 1:

En esta simulación se emplearon bloques de 64-por-64 pixeles para ambos casos, es decir, sin y con SMEP (uso de Haar antes de la TDKL).

La Fig.7 muestra el error pixel-a-pixel entre la imagen original y la recuperada contra una paleta de colores. Los colores en escala de la paleta nos dan la noción del error de recuperación. Es obvia la mejora introducida por el SMEP a la hora de reducir dicho error, dado que el error píxel-a-píxel de la Fig.7 (abajo) es muy inferior al de la Fig.7 (arriba).

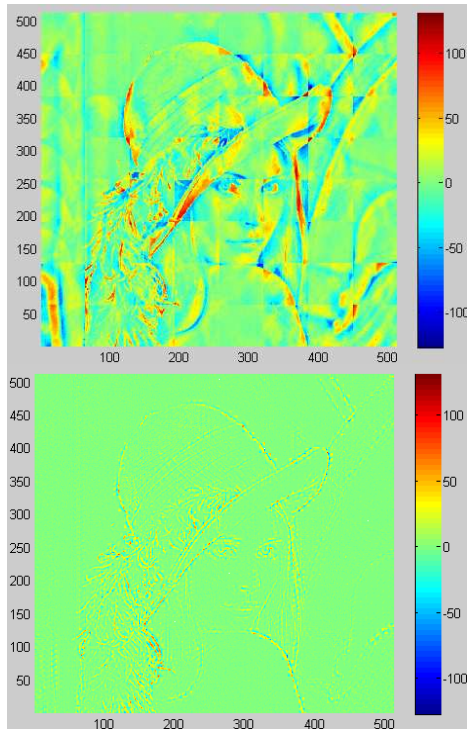


Fig.7. Arriba: Error pixel-a-pixel sin SMEP.
Abajo: Error pixel-a-pixel con SMEP.

La Tabla I cuantifica el primer grupo de métricas de la Sección 3 para ambos casos, es decir, a similar TC y bpp, la versión con SMEP tiene un MSE de casi el 10 % de la versión sin SMEP, además de un marcado aumento del PSNR de casi 10 dB.

Tabla I

SMEP	Métrica			
	TC	bpp	MSE	PSNR
sin	15.6935	0.0640	424.5221	21.8518
con	15.6935	0.0640	48.5315	31.2706

Por otra parte, la Tabla II nos proporciona los valores medidos del segundo grupo de métricas definidas en la Sección 3 para los casos de las Figuras 2 y 6. Como podemos apreciar en esta tabla, la Fig.2 evidencia una marcada ausencia del atributo, mientras que la Fig.6, todo lo contrario.

Tabla II

Figura	Métrica		
	FGP	FRP	FP
2	68.7584	68.7854	30.8915
6	94.1559	94.1601	87.8554

Esta tabla pone en evidencia claramente que la intervención de la ondita de Haar previo al empleo de la TDKL hace aumentar el grado de SMEP como se puede observar por el incremento del porcentaje en las tres métricas involucradas. Mientras que el incremento en FGP y FRP usando la ondita de Haar es del 38 %, en FP el mencionado incremento es del 190 %. Esto le permite competir de igual a igual con JPEG y JPEG2000 sin inconvenientes [2, 3].

Reducción de la complejidad computacional con el incremento del SMEP:

Para una misma calidad de reconstrucción de la imagen, es decir, con y sin SMEP, tenemos:

$$nm_s = \frac{nfl \times ncl}{nfm_s \times ncm_s} \quad (19)$$

$$nm_c = \frac{nfl \times ncl}{nfm_c \times ncm_c} \quad (20)$$

donde:

nfm_s = # de filas de los mosaicos sin SMEP

ncm_s = # de columnas de los mosaicos sin SMEP

nm_s = # de mosaicos (profundidad) sin SMEP

y

nfm_c = # de filas de los mosaicos con SMEP

ncm_c = # de columnas de los mosaicos con SMEP

nm_c = # de mosaicos (profundidad) con SMEP

Por otra parte, siendo:

$$nm_s \gg nm_c \quad (21)$$

$$L = \frac{nm_s}{nm_c} = \frac{nfm_c \times ncm_c}{nfm_s \times ncm_s} \gg 1 \quad (22)$$

Si $nfl \approx ncl \approx N$, entonces, la complejidad computacional pasa de ser $O(N^3)$, como de hecho es originalmente en la TDKL a $O(N^3/L)$, es decir, muy inferior. Además, de (21), al ser mucho menor la cantidad de mosaicos, se reduce dramáticamente el tamaño de la matriz de autocorrelación C_x , la cual depende directamente de este parámetro y por ende la rapidez del algoritmo para hallar los autovalores y autovectores de la TDKL [2, 3].

Simulaciones 2:

En esta simulación se emplearon bloques de 32-por-32 pixeles para que la versión con SMEP (uso de Haar antes de la TDKL) pueda enfrentar con eficiencia a JPEG y JPEG2000.

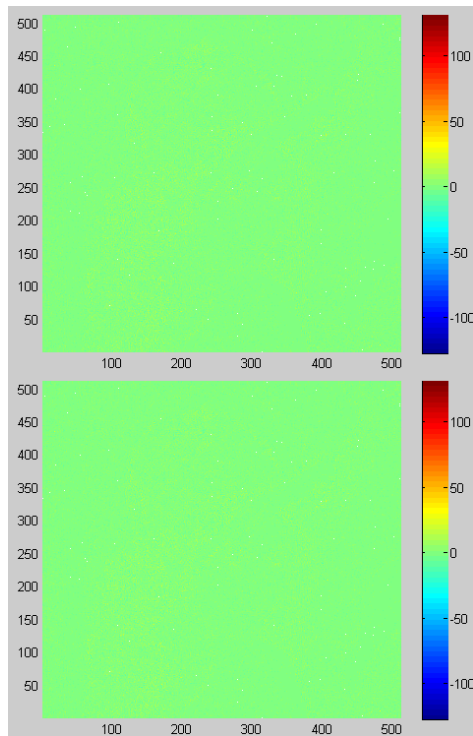


Fig.8. Arriba: Error pixel-a-pixel con SMEP. Abajo: Error pixel-a-pixel en JPEG.

La Fig.8 introduce el error píxel-a-píxel para la versión con SMEP (arriba) contra JPEG (abajo). En ambos casos el error es similarmente imperceptible contra la escala de colores de la paleta a la derecha de la figura. No obstante, se debe tener en cuenta que JPEG utiliza bloques de 8-por-8 píxeles, con el incremento en la cantidad de bloques que esto implica y con el aumento en la complejidad que esto conlleva frente a la técnica propuesta. En otras palabras, ambos producen la misma distorsión visual (extremadamente baja) pero en el caso de la técnica propuesta a un costo computacional sustancialmente inferior, con todo lo que esto representa en los sistemas de comunicaciones y almacenamiento.

Por último, la Tabla III cuantifica el primer grupo de métricas de la Sección 3 para los tres casos, a saber: la técnica propuesta con SMEP, TPEG y JPEG2000. Siendo en los tres casos similares los valores de MSE y PSNR, es notable el incremento de la técnica propuesta en lo relativo a su capacidad de compresión, lo que se evidencia por el marcado aumento de la TC y la bpp.

Tabla III

Técnica	Métrica			
	TC	bpp	MSE	PSNR
con SMEP	15.6935	0.0640	9.5315	38.2706
JPEG	8.9358	0.1124	8.7869	38.7924
JPEG2000	9.9921	0.1005	10.4207	37.9518

4. CONCLUSIONES

Las simulaciones muestran mediante las figuras el error píxel-a-píxel contra una paleta de colores donde el valor cero se corresponden con el color verde. En particular, las Simulaciones 1 muestran la

Fig.7 con el error píxel-a-píxel bastante marcado en el caso sin SMEP frente al caso con SMEP. La Tabla I evidencia la diferencia en los valores de MSE y PSNR a favor del caso con SMEP, para una misma TC y bpp. Mientras que la Tabla II nos muestra el incremento de los porcentajes de las tres métricas por la intervención de la ondita de Haar antes de la TDKL.

Por otro lado, las Simulaciones 2 muestran a partir del la Fig.8 y la Tabla III que la técnica con SMEP es la mejor, aun para bloques de 32-por-32 píxeles. Si se hubieran usado bloques de 16 u 8 como el caso de JPEG y JPEG2000 se habría obtenido una mejora notable y muy por encima de ambas técnicas, pero a costa de un mayor tiempo de cálculo. No obstante, y con bloques grandes los errores de píxeles son idénticos, lo cual la hace mucho más rápida [2, 3].

Finalmente, es muy importante señalar que comparando las columnas de MSE y PSNR de las Tablas I (bloques de 64-por-64) y III (bloques de 32-por-32) podemos de ambas simulaciones, puesto que al hacerlo se puede observar, que una disminución a la mitad en las dimensiones de los bloques considerados provoca que el MSE se reduzca a un quinto de su valor y el PSNR se incremente 7 dB. Esto pone claramente en evidencia las prestaciones de la técnica propuesta en función de este parámetro.

5. BIBLIOGRAFIA

- [1] **Jain A. K.**, *Fundamentals of Digital Image Processing*, Englewood Cliffs, New Jersey, 1989.
- [2] **Mastriani M.**, "Union is Strength in Lossy Image Compression," *International Journal of Signal Processing*, Volume 5, Number 2, pp.112-119, 2009.
- [3] **Mastriani M.**, "Decorrelación espacial rápida y de alta eficiencia para compresión de imágenes con pérdidas," Tesis de Doctorado en Ciencias de la Computación, Departamento de Computación, Facultad de Ciencias Exactas y Naturales de la Universidad de Buenos Aires, Marzo 12, 2009.
- [4] **Epstein B. R., et al.** "Multispectral KLT-wavelet data compression for landsat thematic mapper images," In *Data Compression Conference*, pp. 200-208, Snowbird, UT, March 1992.
- [5] **Lee J.**, "Optimized quadtree for Karhunen-Loève Transform in multispectral image coding," *IEEE Transactions on Image Processing*, 8(4), pp.453-461, 1999.
- [6] **Gastpar M., Dragotti P. L., and Vetterli M.**, The Distributed Karhunen-Loève Transform, *IEEE Transactions on Information Theory*, vol. 52, No. 12, December 2006
- [7] **Pennebaker W. B. y Mitchell J. L.**, *JPEG: Still Image Compression Standard*. Kluwer Academic Publishers, 1993.
- [8] **Wallace G. K.**, The JPEG Still Picture Compression Standard, *Communications of the ACM*, vol. 34, no. 4, pp. 30-44, April 1991.

- [9] **Taubman D. S. y Marcellin M. W.**, *JPEG2000: Fundamentals, Standards and Practice*. Kluwer Academic Publishers, Boston, 2002.
- [10] **Stollnitz E. J., DeRose T. D., y Salesin D. H.**, *Wavelets for Computer Graphics (Theory and Applications)*. Morgan Kaufmann Publishers, San Francisco, 1996
- [11] **Mastriani M., y Giraldez A.**, "Smoothing of coefficients in wavelet domain for speckle reduction in Synthetic Aperture Radar images," *ICGST International Journal on Graphics, Vision and Image Processing (GVIP)*, Volume 6, pp. 1-8, 2005.