

**UNIVERSIDAD RICARDO PALMA**

**FACULTAD DE INGENIERIA ELECTRÓNICA**



**CODIFICACIÓN DE IMÁGENES SATELITALES  
UTILIZANDO TÉCNICAS DE COMPRESIÓN CON  
PÉRDIDAS Y SIN PÉRDIDAS**

**TESIS PARA OPTAR POR EL TÍTULO PROFESIONAL DE  
INGENIERÍA ELECTRÓNICA**

**CARLOS ALBERTO, FLORES GOYCOCHEA**

**LIMA – PERÚ**

**2010**

*Agradecimiento eterno a Dios, a mis padres que hicieron denodados esfuerzos para poder convertirme en un Profesional.*

# ABSTRACT

In this work of thesis there appears the development of a technology of codification to achieve the compression of images satelitales, facilitating hereby his transmission on having used minor bandwidth and minor time of transmission, as well as also facilitating his storage on having used devices with minor capacity in bytes.

This project is characterized by the implementation of computational algorithms based on the software Matlab, where one has developed two technologies of compression with the purpose of reaching a high degree of compression without altering too much the information contained in the image. They are two technologies combined that are in use in this thesis: the codification with losses, and without losses.

The codification with losses is based on the use of transformed discreet of the cosine in 2D (two dimensions), as the JPEG standard uses it, and the codification without losses based on the use of the codification Huffman, where it is achieved to assign the minor quantity of bits for the codification without losing any information.

This combination of technologies obtains valuable results especially in images satelitales that are obtained by very poor resolutions in comparison with the conventional photographies

# PALABRAS CLAVES

- Imágenes Digitales Satelitales
- Imagen Pancromática y Multiespectral
- Transformada Discreta del Coseno
- Código de Huffman

# PROBLEMA DE LA INVESTIGACIÓN

Con el transcurrir del tiempo las imágenes digitales han evolucionando en cuanto a resolución, calidad, colores, y entre otras características, lo cual hace que sea necesario utilizar medios de almacenamiento o anchos de bandas mayores, para lograr guardar o transmitir tal información, respectivamente.

Así tenemos el caso de imágenes médicas, educativas y hasta satelitales. Por lo tanto, existe claramente la necesidad de comprimir o reducir la elevada tasa de bits o bytes que se utilizan para almacenar o transmitir todo tipo de imágenes digitales.

Uno de los principales problemas encontrados en el trabajo de investigación es la necesidad de utilizar sofisticados algoritmos matemáticos, como es el caso de la transformada discreta del coseno 2D. Esta técnica requiere alto esfuerzo computacional que incrementa más aún cuanto mayor sea el tamaño de la imagen satelital a tratar.

# OBJETIVOS

General:

- Desarrollar una técnica de codificación para lograr la compresión de imágenes satelitales, facilitando de esta forma su transmisión y/o almacenamiento.

Específicos:

- Implementar una técnica de compresión con pérdida, basado en conceptos matemáticos de transformaciones a frecuencia.
- Implementar una técnica de compresión sin pérdida, basado en conceptos de la teoría de la información.
- Fusionar la técnica de compresión con pérdida y sin pérdida.

# HIPOTESIS

General:

- Utilizar una combinación de dos técnicas de compresión: una con pérdida y otra sin pérdida. De esta forma se alcanza alta tasa de compresión con reducida pérdida en la compresión.

Específicos:

- Utilizar la transformada discreta del coseno en dos dimensiones a partir de una transformación a frecuencia, de esta manera se alcanza una alta tasa de compresión.
- Utilizar la codificación de Huffman basado en la teoría de la información, de esta manera se alcanza una baja tasa de compresión pero sin pérdida alguna.
- Complementar ambas técnicas de compresión, iniciando con la transformada discreta del coseno en dos dimensiones, complementado con la codificación de Huffman.

# METODOLOGIA

La metodología empleada en este trabajo de investigación está basado en un mecanismo para comprimir o reducir la tasa de bits, pero sin deteriorar la imagen original, tanto así que el ojo humano no perciba tales alteraciones que podrían perjudicar o en todo caso empeorar la información que se está almacenando o transmitiendo. Este mecanismo de compresión es posible conseguirlo utilizando herramientas matemáticas que logran altas tasas de compresión, pero presentando reducida pérdida en la recuperación de la información; o, como también aquellos que alcanzan bajas tasas de compresión, pero presentando cero pérdidas en la recuperación de la información.

Pues, gracias a muchos investigadores han surgido herramientas matemáticas que hoy en día logran solucionar los problemas de almacenamiento y transmisión de volúmenes de imágenes digitales. Así tenemos, por nombrar solo algunas de ellas, la transformada discreta de wavelets, las redes neuronales artificiales, los fractales, la transformada discreta del coseno, codificación por entropía, entre otras. Técnicas que combinadas adecuadamente son posibles de alcanzar elevadas tasas de compresión.

Consecuentemente, se plantea en este trabajo de tesis, el uso mixto de estas técnicas de compresión para lograr una disminución considerable del número de bits por pixel por un lado, y por el otro conservar la información en la imagen digital. Para ello, se ha optado por realizar un trabajo similar al que el Joint Photographic Experts Group (JPEG) realizar, pero con la salvedad que la técnica a desarrollar sólo será aplicado sobre un tipo especial de imágenes, que son las satelitales.

# TABLA DE CONTENIDOS

INTRODUCCIÓN.....	11
CAPÍTULO 1 FUNDAMENTOS DE IMÁGENES SATELITALES .....	17
1.1 – Definición de satélite .....	17
1.2 – Clasificación de Satélites Artificiales.....	18
1.3 – Los Satélites Meteorológicos.....	19
1.4 – Los Satélites METEOSAT .....	21
1.5 – Imágenes Digitales Satelitales .....	22
1.6 – Beneficios de los Satélites Meteorológicos .....	24
CAPÍTULO 2 TÉCNICA DE COMPRESIÓN CON PÉRDIDA .....	28
2.1 – Introducción.....	28
2.2 – Transformada Discreta del Coseno Unidimensional (DCT-1D) .....	29
2.3 – Transformada Discreta del Coseno Bidimensional (DCT-2D) .....	32
2.4 – Ejemplos Prácticos de la DCT-2D e IDCT-2D .....	34
2.5 – Aplicaciones de la DCT.....	41
2.5.1 – Codificación de Imagen.....	41
2.5.2 – Transmisión Progresiva de Imágenes .....	42
CAPÍTULO 3 TÉCNICA DE COMPRESIÓN SIN PÉRDIDA .....	46
3.1    Introducción.....	46
3.2    Codificación de Huffman .....	47
3.3    Ejemplo Práctico .....	49
CAPÍTULO 4 TÉCNICA DE CODIFICACIÓN DESARROLLADA.....	59
4.1    Introducción.....	59
4.2    Pre Procesamiento de la imagen .....	60
4.2.1    Filtrado y operación de substracción de la imagen.....	61
4.2.2    Obtención de la DCT-2D.....	64
4.2.3    Normalización y Cuantificación .....	68
4.2.4    Ordenamiento de los Coeficientes DCT .....	69
4.3    Codificación con Huffman de la Imagen Diferencia.....	73
4.4    Técnica de Decodificación .....	74
4.5    Ejemplo de Codificación y Decodificación.....	75

CAPÍTULO 5 PRUEBAS, RESULTADOS Y SIMULACIONES.....	78
5.1    Imágenes satelitales utilizadas.....	78
5.2    Error numérico y visual .....	79
CONCLUSIONES.....	83
WEBGRAFÍA .....	85
BIBLIOGRAFÍA.....	86

# INTRODUCCIÓN

Con el paso del tiempo las imágenes digitales han venido evolucionando en cuanto a resolución, calidad, colores, y entre otras características, lo cual hace que sea necesario utilizar medios de almacenamiento o anchos de banda mayores, para lograr guardar o transmitir tal información. Por ejemplo, si tuviéramos una imagen digital a colores perteneciente a una tomografía axial computarizada (TAC) cuya resolución es de 1024x512 pixels, y codificada a 16 bits por pixel, se utilizaría un total de  $1024 \times 512 \times 16 \times 3$  bits, lo que hace un total de 25165824 bits, o su equivalente a tres megabytes (3 MBytes). Por lo tanto, si esto corresponde solo a un paciente que fue atendido en una mañana en un centro hospitalario público, surge la siguiente pregunta ¿cuántos megabytes se utilizarían para representar digitalmente las TAC's de 30 pacientes que fueron atendidos un único día?, y si a eso lo multiplicamos por 26 días útiles de un mes, daría un total de 2340 MBytes, solo para un consultorio en uno de los muchos centros hospitalarios que tenemos sólo en la capital de nuestro país. Tener en cuenta que la cantidad de información es variada, y no solo se limita a imágenes de TAC's, sino también a imágenes de Mamografía, Resonancia Magnética, Gammagrafía, entre otras. Entonces surge la necesidad de almacenar abundante información, aun conociendo la existencia de dispositivos de almacenamiento por el orden de los TeraBytes (TBytes) que no se abastecerían en el caso de almacenar miles de imágenes de años atrás con la finalidad de alimentar una base de datos que sirva para enriquecer los conocimientos de los futuros médicos. Asimismo, si la necesidad surge ahora por el lado de la transmisión, también se tendría que contar con un mayor ancho de banda, para poder transmitir abundante data en el menor tiempo posible. Entonces, a continuación se muestra un ejemplo del tiempo empleado para transmitir, serialmente, la cantidad anteriormente calculada, a través del puerto USB (Universal Serial BUS versión 2.0) de una PC convencional.

Este ejemplo resultaría pieza clave para tener una referencia en el caso que la transmisión se realice a través de este puerto, la tasa de bits es igual a 480 Mbits por segundo<sup>1</sup>, por lo tanto, haciendo un cálculo simple de regla de tres, se tiene un resultado

---

<sup>1</sup> [http://www.es.wikipedia.org/wiki/Universal\\_Serial\\_Bus](http://www.es.wikipedia.org/wiki/Universal_Serial_Bus)

de un total de 39 segundos para transmitir completamente los 2340 MBytes. Sin contar que esta cantidad de datos nunca viaja sola como una trama de bits, sino también va acompañado de ciertos bits adicionales utilizados para la parte de detección y corrección de errores, tema que escapa al contenido de esta tesis porque la idea de este trabajo solo se limita al planteamiento y desarrollo de un algoritmo de compresión.

Entonces, existe claramente la necesidad de comprimir o reducir la elevada tasa de bits o bytes que se utilizan para almacenar o transmitir imágenes médicas, pertenecientes sólo a un tipo en particular y en un solo consultorio de uno de los centros médicos de nuestra ciudad.

De manera similar se presenta esta gran limitación en otros tipos de imágenes digitales y esta vez señalizamos como ejemplo el caso de las imágenes satelitales. Como es de conocimiento son muchos los satélites que se encuentran orbitando y captando información en intervalos de tiempo diferentes, así tenemos información relacionada a la temperatura superficial del mar, la presencia de clorofila, de salinidad, seguridad, entre otros diferentes tipos de datos. Por lo tanto, si realizamos un cálculo similar para cada imagen satelital, notaremos que el número de megabytes también resulta elevado según la resolución y área que se desea cubrir, sin dejar a lado el tema de captura de información sea del tipo pancromática o multiespectral. Esta cantidad de megabytes en incremento será mostrado a partir de un ejemplo en las siguientes líneas. Es de saber, que la resolución de imágenes satelitales está presentada por el tamaño mínimo que puede ser observado desde el espacio; por ejemplo, si un satélite artificial en particular tiene una resolución de 60 centímetros<sup>2</sup>, 11 bits por pixel, y una captura de imagen correspondiente a un área de 10x10 Km, entonces la imagen digital satelital llegará a tener un aproximado de: 380 MBytes lo cual se obtiene realizando el siguiente cálculo para la unidad de bits:  $11 \times (10 \times 10 \times 10^6) / (60 \times 60 \times 10^{-4})$  bits. Este cálculo es obtenido del principio básico de resolución de imágenes satelitales, es decir, cuando se señala que la resolución es de 60 centímetros, entonces equivale a asociar 1 pixel de la imagen visualizada desde el espacio a un área de 60x60 centímetros. Por lo tanto, un objeto en la tierra de tamaño real e igual a 1.2x1.2 metros corresponderá a cuatro pixeles de la imagen satelital capturada desde el espacio a la resolución de 60 centímetros.

---

<sup>2</sup> Satellite Technology Principles and Applications. Anil K. Maini. 2010.

Es necesario resaltar que el tamaño de las imágenes digitales satelitales depende mucho del área geográfica que se desea capturar y de la resolución. Es decir, no es lo mismo capturar un área de 1.2x1.2 Km a una resolución de 60 centímetros, frente a otra captura de un área geográfica de 5x5 Km. a una resolución de 2 metros. Se aclara nuevamente que cuando se señala la resolución, sea de 60 centímetros o de 2 metros, se hace referencia al tamaño de un pixel:

Área de 60x60 centímetros equivale a 1 pixel de la imagen digital satelital.

Área de 2x2 metros equivale a 1 pixel de la imagen digital satelital.

Seguidamente, la figura 1.1 muestra un ejemplo de imágenes satelitales de distintos lugares del mundo pero con resoluciones diferentes. La imagen de la izquierda presenta una resolución de 15 metros, es decir que cada pixel tendrá un tamaño de 15x15 metros. Por otro lado, la imagen satelital de la derecha corresponde a una resolución de 82 centímetros, es decir que cada pixel tendrá un tamaño de 82x82 centímetros.



Figura 1. Comparación de imágenes satelitales. Izquierda: Imagen satelital de la cuenca amazónica Perú. Derecha: Imagen satelital de Jinhua CHINA.<sup>3</sup>

Es así, que las imágenes digitales se encuentran presentes en diversos sectores, tales como el médico y satelital, que ya fueron comentados, también las encontramos en el sector militar, educacional, climatológico, entre muchos otros sectores más.

---

<sup>3</sup> <http://www.satimagingcorp.es/gallery.html>

De esta forma, se hace totalmente indispensable utilizar un mecanismo para comprimir o reducir la tasa de bits, pero sin deteriorar la imagen original, tanto así que el ojo humano no perciba tales alteraciones que podrían perjudicar o en todo caso empeorar la información que se está almacenando o transmitiendo. Este mecanismo de compresión es posible conseguirlo utilizando herramientas matemáticas que logran altas tasas de compresión, pero presentando reducida pérdida en la recuperación de la información; o, como también aquellos que alcanzan bajas tasas de compresión, pero presentando cero pérdidas en la recuperación de la información.

Pues, gracias a muchos investigadores han surgido herramientas matemáticas que hoy en día logran solucionar los problemas de almacenamiento y transmisión de volúmenes de imágenes digitales. Así tenemos, por nombrar solo algunas de ellas, la transformada discreta de wavelets, las redes neuronales artificiales, los fractales, la transformada discreta del coseno, codificación por entropía, entre otras. Técnicas que combinadas adecuadamente son posibles de alcanzar elevadas tasas de compresión.

Consecuentemente, se plantea en este trabajo de tesis, el uso mixto de estas técnicas de compresión para lograr una disminución considerable del número de bits por pixel por un lado, y por el otro conservar la información en la imagen digital. Para ello, se ha optado por realizar un trabajo similar al que el Joint Photographic Experts Group (JPEG) realizar, pero con la salvedad que la técnica a desarrollar sólo será aplicado sobre un tipo especial de imágenes, que son las satelitales.

## **OBJETIVOS**

General:

- Desarrollar una técnica de codificación para lograr la compresión de imágenes satelitales, facilitando de esta forma su transmisión y/o almacenamiento.

Específicos:

- Implementar una técnica de compresión con pérdida, basado en conceptos matemáticos de transformaciones a frecuencia.
- Implementar una técnica de compresión sin pérdida, basado en conceptos de la teoría de la información.

- Fusionar la técnica de compresión con pérdida y sin pérdida.

## **IMPORTANCIA**

La importancia del presente trabajo de investigación radica en que las imágenes satelitales representan informaciones claves para centros militares, meteorológicos, de investigación, entre otros; por tal motivo día tras día el número de imágenes satelitales incrementa, por lo cual el tiempo de transmisión de estas, o como también, el espacio de almacenamiento va incrementando exponencialmente. Por esta razón, urge en desarrollar algoritmos de codificación que alcancen altas tasas de compresión con la finalidad de mantener y no deteriorar la información presente en tales imágenes. Con este propósito e importancia, nace este tema para ser desarrollado en la búsqueda de una combinación de algoritmos de compresión con pérdidas y sin pérdidas.

## **RESUMEN DE CADA CAPÍTULO**

El desarrollo de este trabajo de tesis se inicia con los fundamentos de imágenes satelitales, partiendo de conceptos relacionados a los satélites, sus órbitas, generalidades, tipos, características y diversidades de ejemplares de imágenes satelitales que hoy en día son utilizadas con diferentes fines: militares, climatológico, defensa civil, entre otras.

En el segundo capítulo, se trata de los conceptos de la técnica de compresión con pérdida a utilizar en esta tesis. Es decir, la transformada discreta del coseno (DCT). Su aplicación en el caso unidimensional y bidimensional, otorgando algunos ejemplos prácticos con ayuda del software de simulación matemática Matlab.

La técnica de compresión sin pérdida, basado en la Codificación Huffman y en particular en el árbol de Huffman, será tratado en el capítulo 3, en donde se darán definiciones de este tipo de compresión, así como también algunos ejemplos prácticos que serán simulados con ayuda del software Matlab.

Luego, en el capítulo 4, se desarrollará la técnica de compresión planteada realizando una fusión entre la técnica de compresión con pérdida y sin pérdida. Esto llevará a la realización de un algoritmo en el software Matlab para simular y obtener los resultados numéricos y gráficos, lo cual servirá como base para más adelante hacer las respectivas

comparaciones y obtener los resultados desde el punto de vista numérico haciendo uso de la relación señal ruido pico, y desde el punto de vista visual haciendo comparaciones directas y así como también sustracciones entre imagen original y codificada. Esto se abordará en el último capítulo.

Finalmente, se darán las conclusiones de este trabajo desarrollado haciendo hincapié en la complejidad computacional en comparación con otros métodos de compresión.

# CAPÍTULO 1

## FUNDAMENTOS DE IMÁGENES SATELITALES

Este primer capítulo de este trabajo de tesis cubre los fundamentos básicos de imágenes satelitales; es decir, se comienza con una teoría básica acerca del fundamento de imágenes satelitales. Luego, se continúa con las variedades de imágenes satelitales que son posibles de obtener desde los satélites, y sobre todo se comenta sobre la influencia que tiene este tipo de información en el desarrollo y bienestar de nuestra población mundial.

### **1.1 – Definición de satélite**

Existen diversos conceptos relacionados a satélites artificiales, pero el que mostraremos a continuación resulta el más apropiado para esta tesis. “Un satélite es cualquier objeto que orbita alrededor de otro, que se denomina principal. Los satélites artificiales son naves espaciales fabricadas en la Tierra y enviadas en un vehículo de lanzamiento, un tipo de cohete que envía una carga útil al espacio exterior. Los satélites artificiales pueden orbitar alrededor de lunas, cometas, asteroides, planetas, estrellas o incluso galaxias. Tras su vida útil, los satélites artificiales pueden quedar orbitando como basura espacial”.<sup>4</sup>

También se encuentran definiciones tales como la de: Son ingenios lanzados por un cohete, que pasan a describir una órbita alrededor de la Tierra o de otro astro”<sup>5</sup>.

A continuación la figura 1.1 muestra una fotografía como ejemplo de uno de los primeros satélites que fue lanzado desde la tierra.

---

<sup>4</sup> [http://es.wikipedia.org/wiki/Sat%C3%A9lite\\_artificial](http://es.wikipedia.org/wiki/Sat%C3%A9lite_artificial)

<sup>5</sup>

[http://www.inegi.gob.mx/inegi/contenidos/espanol/ciberhabitat/medios/satelites/artificiales/que\\_es.htm](http://www.inegi.gob.mx/inegi/contenidos/espanol/ciberhabitat/medios/satelites/artificiales/que_es.htm)

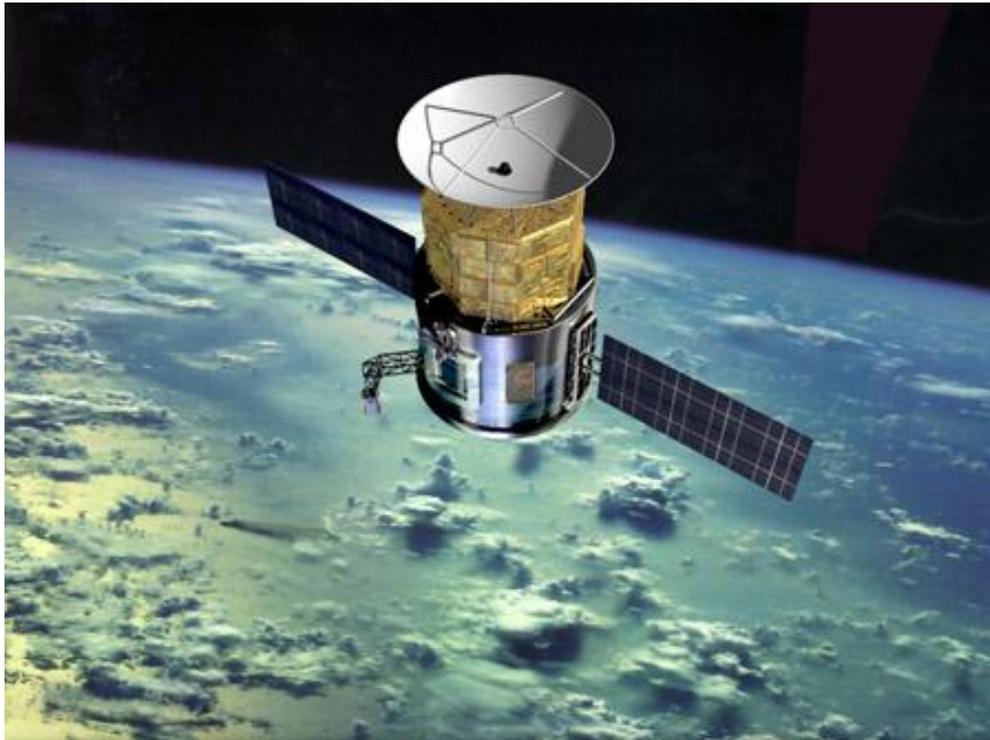


Figura 1.1 Fotografía de satélite artificial.<sup>6</sup>

## 1.2 – Clasificación de Satélites Artificiales

Los satélites artificiales se clasifican de acuerdo a su tamaño que van desde los micro satélites con pesos menores a 50 Kg. a satélites grandes de varias toneladas tal como aquellos que corresponden a estaciones espaciales. También existe otra forma de clasificarlos, y esta vez en función de su órbita es decir como giran alrededor de nuestro planeta. Asimismo, también se pueden clasificar según el uso que tienen. A continuación se muestra la clasificación de los mismos:<sup>7</sup>

1. Satélites de Comunicación.
2. Satélites de Navegación.
3. Satélites Meteorológicos
4. Satélites de estudios de recursos terrestres y marítimos.
5. Satélites militares y de espionaje.
6. Satélites científicos.

---

<sup>6</sup> [http://www.proyectosalohogar.com/Enciclopedia/Astronautica/NE\\_Astronautica.htm](http://www.proyectosalohogar.com/Enciclopedia/Astronautica/NE_Astronautica.htm)

<sup>7</sup> [http://www.astronomos.org/articulistas/Polaris/2004/43-SATELITES\\_ARTIFICIALES.pdf](http://www.astronomos.org/articulistas/Polaris/2004/43-SATELITES_ARTIFICIALES.pdf)

Por ejemplo, los satélites de comunicación sirven de enlace para las comunicaciones telefónicas, las emisiones de señal de televisión, de internet y hasta para tener contacto de radio permanente entre buques, trenes y aviones.

Asimismo, los satélites de navegación, permiten la localización precisa de cualquier punto sobre nuestro planeta Tierra. Basados en métodos de triangulación. Aquí encontramos los sistemas de posicionamiento GPS (sistema de posicionamiento global).

Y de igual forma, los satélites meteorológicos, son encargados de recoger datos de la atmósfera y/o de la superficie terrestre. Esto sirve como un satélite de observación que muchas veces también es utilizado en la oceanografía, los estudios ambientales o también la cartografía.

En este trabajo de tesis se tratará el tema de un solo tipo de satélite denominado satélite meteorológico, el cual tiene importante desempeño hoy en día y cumple con dotarnos de información propia para el pronóstico del clima en diferentes partes de nuestro país.

### **1.3 – Los Satélites Meteorológicos**

Los satélites meteorológicos constituyen el Subsistema Espacial del Sistema Mundial de Observación, y cuyo principal objetivo es completar la información facilitada por el subsistema de superficie de una forma económicamente viable<sup>8</sup>. En la figura 1.3 se observan los principales satélites meteorológicos situados alrededor de nuestro planeta. Esta figura se encuentra referenciada por el documento hallado en la página de internet del Senamhi. “Estos satélites pueden clasificarse en dos grandes grupos. Un primer grupo denominado de órbita polar o heliosincrónicos, es decir están sincronizados con el Sol, tal como su nombre lo indica orbitan nuestro planeta de polo a polo y lo componen principalmente la serie TIROS de la agencia NOAA (National Oceanic and Atmospheric Administration) de origen americano (NOAA-14, NOAA-15, etc), y los METEOR (METEOR-2, METEOR 3-5, etc.) de origen ruso”<sup>8</sup>. Ver las figuras 1.3 y 1.4.

---

<sup>8</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos]

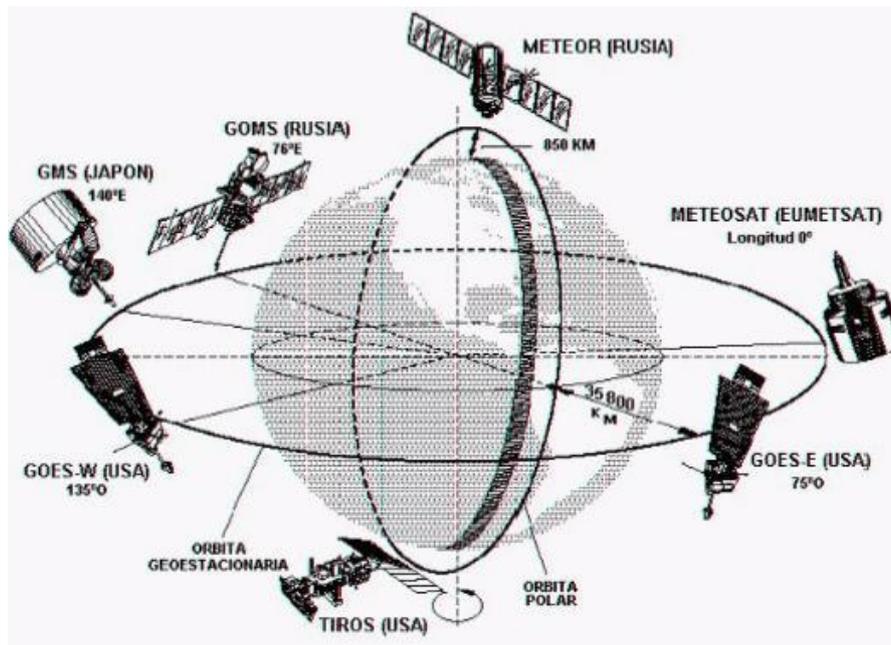


Figura 1.2 Principales Satélites Meteorológicos<sup>9</sup>



Figura 1.3 Satélite NOAA de origen americano. <sup>10</sup>

<sup>9</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos, página 1]

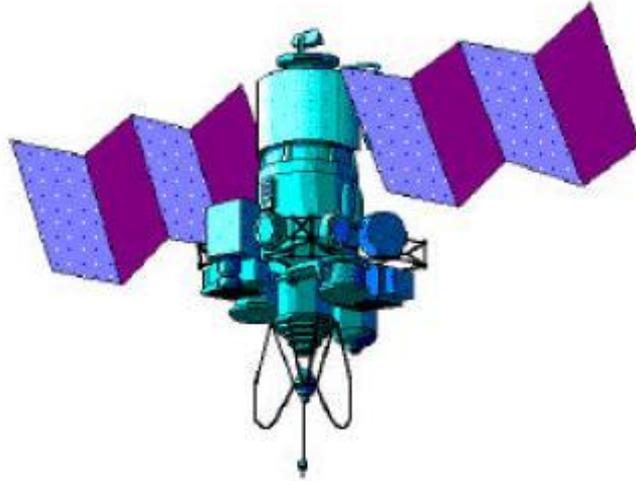


Figura 1.4 Satélite METEOR-3 de origen ruso. <sup>10</sup>

Los satélites del segundo grupo lo componen todos aquellos denominados geoestacionarios o geosincrónicos, esto quiere decir que se encuentran sincronizados con el movimiento de rotación de nuestro planeta. Este grupo de satélites orbitan a mayor altura y se encuentran sobre o muy próximos a la línea ecuatorial<sup>11</sup>.

Una de las principales utilidades de estos satélites es la de poder visualizar el conjunto Tierra-Atmósfera, y extraer la máxima información posible a través de distintas técnicas y procesos para obtener los productos; y cuyo objetivo se basa en el análisis cualitativo y cuantitativo del grupo de imágenes obtenidas<sup>11</sup>. Tales imágenes capturadas se utilizan para principalmente para la visualización de nubes, clasificación observación del vapor de agua presente en la alta y media atmósfera, temperaturas de la superficie terrestre y temperatura superficial del mar.<sup>11</sup>

## 1.4 – Los Satélites METEOSAT

Son satélites geoestacionarios europeos. La altitud de estos satélites se encuentra alrededor de 35800 Km. y se encuentran equipados con un radiómetro que explora la

---

<sup>10</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos, página 4]

<sup>11</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos, página 3]

tierra línea por línea; cada línea consiste en una serie de elementos de imagen o de píxeles. Para cada píxel el radiómetro mide la energía radiada de las diversas gamas espectrales. Esta medida digital se cifra y se transmite a la estación de tierra para procesarla antes de ser entregada a la comunidad para su utilización. Este radiómetro es un instrumento de tres canales: el canal visible es 0.45 a 1.0  $\mu\text{m}$ , el canal infrarrojo es 10.5 a 12.5  $\mu\text{m}$  y el canal vapor de agua es 5.7 a 7.1  $\mu\text{m}$ .<sup>12</sup>

Las imágenes capturadas por estos satélites se toman cada 30 minutos y el canal visible explora 5000 líneas, donde cada línea consiste de 5000 píxeles; por su parte, los canales infrarrojos exploran 2500 líneas y donde cada línea consiste de 2500 píxeles. Por lo tanto, esto equivale a una resolución de 2.5 y 5 Km respectivamente.<sup>12</sup> Debido a la curvatura de nuestro planeta, dicha resolución disminuye ligeramente hacia los bordes externos de la imagen cuando se intenta capturar áreas muy grandes de kilómetros cuadrados. Entonces, una imagen capturada con la resolución anteriormente citada, daría:  $5000 \times 5000 \times 11$  bits lo que es aproximadamente: 33 MBytes, para el caso de considerarse 11 bits por píxel.

## **1.5 – Imágenes Digitales Satelitales**

Las imágenes digitales capturadas de los satélites pertenecen a dos grupos. Por un lado se tiene aquellas imágenes denominadas pancromáticas, y por el otro imágenes multi espectrales.

Entonces, a continuación se muestra una definición breve sobre cada tipo de imagen satelital existente.

“Cabe preguntarse qué ventajas tiene el utilizar imágenes de satélites cuando existen muchas otras fuentes de datos geográficos, como fotografías aéreas, estudios sobre el terreno y mapas sobre papel”<sup>13</sup>

- a) Imagen pancromática.

---

<sup>12</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos, página 8]

<sup>13</sup> [http://www.srgis.cl/pdf/guia\\_basica\\_imagenes\\_satelitales.pdf](http://www.srgis.cl/pdf/guia_basica_imagenes_satelitales.pdf)

“Se captan mediante un sensor digital que mide la reflectancia de energía en una amplia parte del espectro electromagnético (con frecuencia, tales porciones del espectro reciben el nombre de bandas). Para los sensores pancromáticos más modernos, esta única banda suele abarcar la parte visible y de infrarrojo cercano del espectro. Los datos pancromáticos se representan por medio de imágenes en blanco y negro”.<sup>13</sup>. Ver la figura 1.5.

b) Imagen multiespectral.

“Se captan mediante un sensor digital que mide la reflectancia en muchas bandas. Por ejemplo, un conjunto de detectores puede medir energía roja reflejada dentro de la parte visible del espectro mientras que otro conjunto mide la energía del infrarrojo cercano. Es posible incluso que dos series de detectores midan la energía en dos partes diferentes de la misma longitud de onda. Estos distintos valores de reflectancia se combinan para crear imágenes de color. Los satélites de teledetección multiespectrales de hoy en día miden la reflectancia simultáneamente en un número de bandas distintas que pueden ir de tres a catorce”<sup>13</sup>. Ver la figura 1.6.



Figura 1.5 Ejemplo de imagen pancromática.<sup>14</sup>

---

<sup>14</sup> [http://www.srgis.cl/pdf/guia\\_basica\\_imagenes\\_satelitales.pdf](http://www.srgis.cl/pdf/guia_basica_imagenes_satelitales.pdf)



Figura 1.6 Ejemplo de imagen multiespectral <sup>14</sup>

## 1.6 – Beneficios de los Satélites Meteorológicos

Son muchos los beneficios que obtenemos al contar con satélites meteorológicos. Seguidamente indican los beneficios más resaltantes<sup>15</sup>.

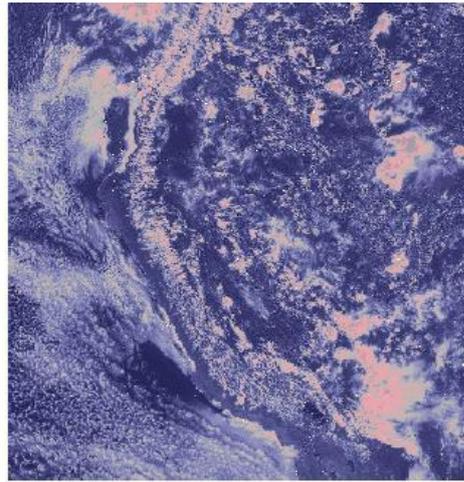
- a) Imágenes de alta y baja resolución. Las imágenes de menor resolución y de datos, sólo son útiles a los fines observacionales y en meteorología sinóptica, para el análisis de nubes, formas, frentes, estimaciones globales, entre otros. Por otro lado, las imágenes cuantitativas, que contienen gran cantidad de información porque son de alta resolución y las componentes de la imagen están sumamente procesadas, pueden utilizarse para la observación, análisis y estudio más exacto de la atmósfera y el suelo<sup>15</sup>. Ver la figura 1.7. Estas imágenes son utilizadas en el desarrollo de este trabajo como parte del conjunto de prueba.

---

<sup>15</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos, página 13]



PERU GVAR - SENAMHI (Alta resolución)  
3-SET-2001 VIS 17:15Z (Escala gris)



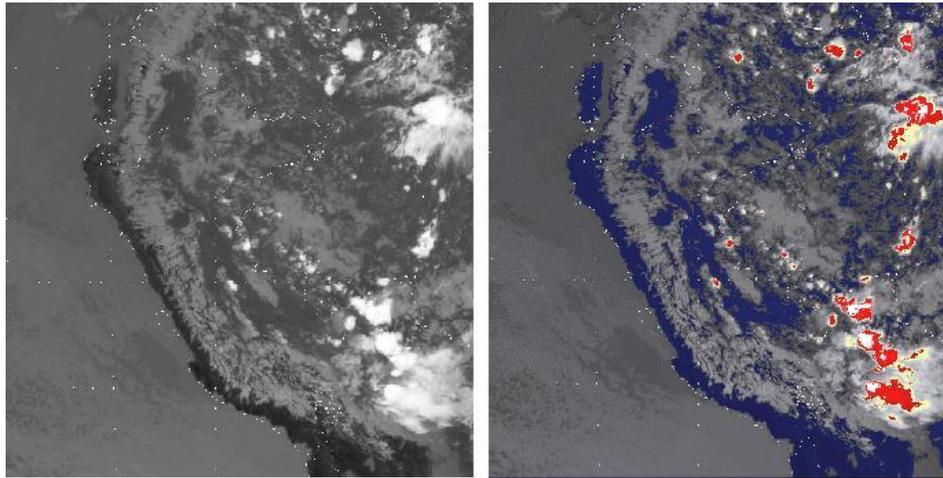
PERU GVAR - SENAMHI (Alta resolución)  
3-SET-2001 VIS 17:15Z (ENH reforzado)

Figura 1.7 Ejemplo de imagen de alta resolución.<sup>15</sup>

b) Imágenes en el espectro Infrarrojo (IR). La Tierra y la atmósfera emiten radiación térmica confinada dentro del intervalo espectral 3 a 100  $\mu\text{m}$ , donde se encuentra la banda infrarroja media (3 a 30  $\mu\text{m}$ ). En estas longitudes de onda la reflectividad es virtualmente nula y la radiación solar despreciable, por eso se considera como radiación de cuerpo negro y se relaciona con la temperatura, medida en grados Kelvin. Las imágenes en infrarrojo, se utilizan principalmente para la observación de las estructuras cuando no existe radiación solar, es decir de noche y en donde los puntos cálidos aparecen oscuros y los fríos de color blanco<sup>16</sup>. Vea la figura 1.8. De igual forma que en el caso de las imágenes de alta resolución, se tomará en cuenta este grupo de imágenes para el desarrollo de este trabajo de tesis.

---

<sup>16</sup> <http://www.senamhi.gob.pe/?p=1003>

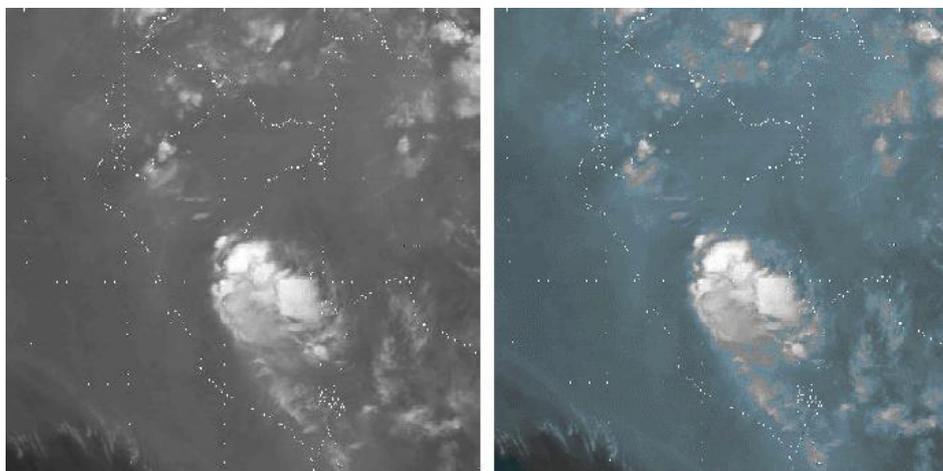


PERU GVAR - SENAMHI (Alta resolución)  
3-SET-2001 IR 17:15Z (Escala gris)

PERU GVAR - SENAMHI (Alta resolución)  
3-SET-2001 IR 17:15Z (ENH reforzado)

Figura 1.8 Ejemplo de imagen en el espectro infrarrojo<sup>16</sup>.

c) Imágenes de vapor de agua (WV). Las imágenes en visible e infrarrojo térmico, utilizan las bandas del espectro electromagnético donde la absorción por los gases atmosféricos es pequeña, sin embargo son de interés también los intervalos espectrales donde la radiación infrarroja emitida por la Tierra es absorbida por el vapor de agua de la atmósfera (figura 1.9)<sup>17</sup>.



PERU GVAR - SENAMHI (Alta resolución)  
10-SET-2001 WV 14:15Z (Escala gris)

PERU GVAR - SENAMHI (Alta resolución)  
10-SET-2001 WV 14:15Z (ENH reforzado)

Figura 1.9. Ejemplo de imagen de vapor de agua<sup>17</sup>

<sup>17</sup> <http://www.senamhi.gob.pe/?p=1003> [CAPÍTULO 12 : Los satélites meteorológicos, página 14]

Otros tipos de imágenes. Asimismo, existe un grupo de imágenes tales como: temperatura superficial del mar, del suelo, índices de vegetación, etc, que son posibles de obtener a través de algoritmos y técnicas de comparación de bandas, filtrado, combinaciones de datos de satélites y radares, entre otros.

## CAPÍTULO 2

# TÉCNICA DE COMPRESIÓN CON PÉRDIDA

Este segundo capítulo trata de la técnica de compresión con pérdida elegido para el desarrollo de la tesis planteada. Para ello, se prefirió por la Transformada Discreta del Coseno (DCT) por ser una herramienta ampliamente utilizada en compresión de imágenes, pues posee una excelente propiedad cuando es aplicada a señales de voz y video, haciendo que haya una mayor concentración de energía en pocos coeficientes. Se revisarán conceptos matemáticos relacionados a la DCT unidimensional para luego pasar a la DCT bidimensional, así como también se enumerarán las principales aplicaciones de la DCT y algunas simulaciones sobre imágenes satelitales, haciendo uso del Matlab.

### **2.1 – Introducción**

La Transformada Discreta del Coseno (DCT) o simplemente, transformada del coseno, viene siendo muy utilizada en la actualidad para efectos de compresión de datos. Tal como lo podemos encontrar en los estándares Joint Photographic Experts Group (JPEG) y Motion Picture Experts Group (MPEG), en donde se valen de esta herramienta matemática para alcanzar altas tasas de compresión con pérdidas imperceptibles al ojo humano. Esta transformada, al igual que la Transformada Karhunen-Loève (KLT), que produce coeficientes incorrelados, logra descorrelacionar sus coeficientes logrando de esta manera tratarlos de forma independiente sin afectar el grado de compresión. Otro aspecto importante de la DCT, es la capacidad de cuantificar los coeficientes utilizando valores de cuantificación que se encuentran más sujetos o dependientes del sistema visual humano.

## 2.2 – Transformada Discreta del Coseno Unidimensional (DCT-1D)

La Transformada Discreta del Coseno Directa (DCT) es un tipo de transformada real y ortogonal, encargada de compactar la energía para datos altamente correlacionados.

Sea una matriz cuadrada ortogonal  $C$ , entonces la inversa de dicha matriz será completamente igual a la transpuesta de la misma. O bien, si multiplicamos la matriz ortogonal  $C$  con su respectiva transpuesta, el resultado es la matriz identidad<sup>18</sup>.

$$\begin{aligned} C^{-1} &= C^T \\ C * C^T &= I \end{aligned} \tag{2.1}$$

De esta manera, es posible transformar un vector columna  $X$  de dimensión  $N$ , en otro vector columna  $Y$  de la misma dimensión, utilizando una matriz ortogonal  $C$  de dimensión  $N \times N$ <sup>18</sup>.

$$Y_{N \times 1} = C_{N \times N} * X_{N \times 1} \tag{2.2}$$

Donde el vector  $Y$  será denominado como la transformación unidimensional. De igual manera, existe la respectiva transformación inversa si despejamos  $X$  de la ecuación 2.2:

$$\begin{aligned} C_{N \times N}^T * Y_{N \times 1} &= C_{N \times N}^T * C_{N \times N} * X_{N \times 1} \\ C_{N \times N}^T * Y_{N \times 1} &= I_{N \times N} * X_{N \times 1} \\ X_{N \times 1} &= C_{N \times N}^T * Y_{N \times 1} \end{aligned} \tag{2.3}$$

También, es viable demostrar que por la propiedad de ortogonalidad, la suma de los cuadrados de todos los elementos de los vectores  $X$  e  $Y$ , se conservan<sup>19</sup>:

$$\sum_{i,j} X_{i,j}^2 = \sum_{i,j} Y_{i,j}^2 \tag{2.4}$$

---

<sup>18</sup> Material didáctico: Transformadas de Imágenes por Alba y Martín 2006.

Donde la matriz C de ortogonalidad, se define como<sup>19</sup>:

$$C(i, j) = \begin{cases} \sqrt{\frac{1}{N}} \cos\left(\frac{\pi(j+1)i}{2N}\right) & i = 0, \quad j = 0, 1, \dots, N-1 \\ \sqrt{\frac{2}{N}} \cos\left(\frac{\pi(j+1)i}{2N}\right) & i = 1, 2, \dots, N-1, \quad j = 0, 1, \dots, N-1 \end{cases} \quad (2.5)$$

y es la que contiene a los vectores o funciones base de la DCT.

Por esta razón, se entiende que la DCT es como un proceso de descomposición de un conjunto de muestras discretas, en un nuevo conjunto ponderado de funciones base cosinusoidales. Y por el contrario, al método de reconstruir el conjunto de muestras a partir del nuevo conjunto ponderado de funciones base cosinusoidales, se le conoce como Transformada Discreta del Coseno Inversa (IDCT)<sup>19</sup>.

El conjunto ponderado de funciones base cosinusoidales son representados mediante formas de onda con una propiedad de ortogonalidad, es decir si multiplicamos dos funciones base cosinusoidales cualesquiera, el resultado será nulo. De igual forma, se multiplicamos una función base cosinusoidal por sí misma, el resultado del producto dará una constante.

La figura 2.1 muestra un ejemplo de 08 funciones base cosinusoidales, que corresponden a 08 secuencias discretas que van incrementando en frecuencia, y que fueron obtenidos a partir de la ecuación 2.5 para N=8. Claramente se observa, que el análisis de la DCT en K=0, da como respuesta una constante.

---

<sup>19</sup> Material didáctico: Transformadas de Imágenes por Alba y Martín. 2006

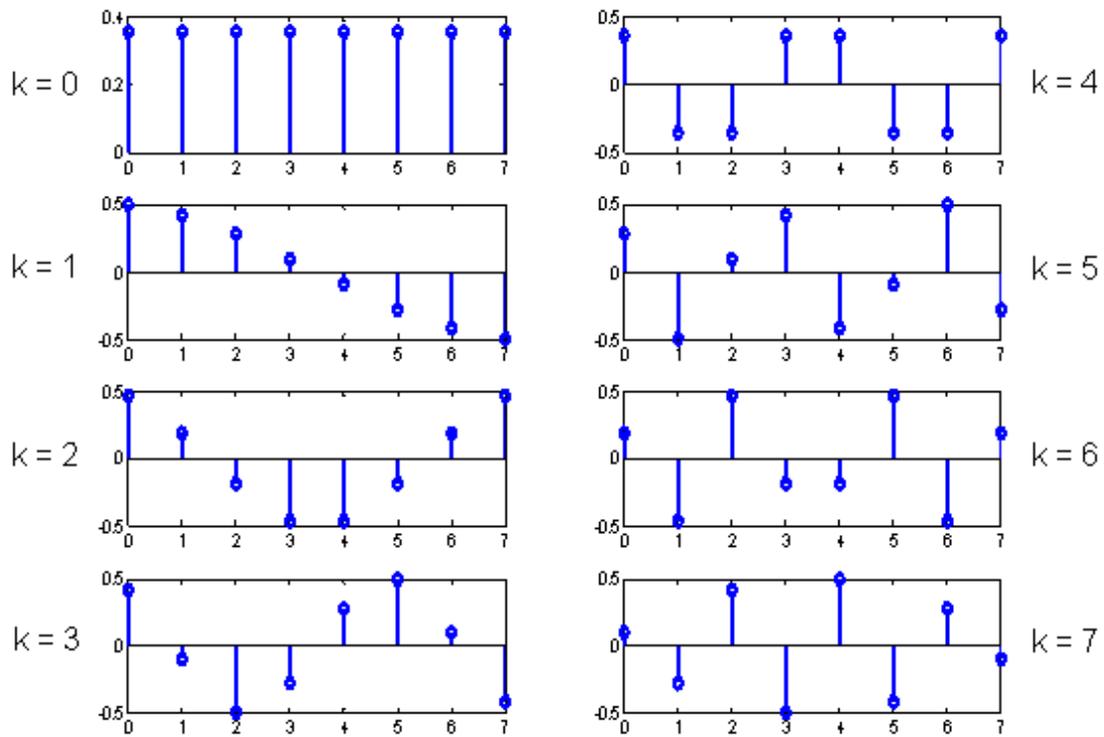


Figura 2.1 Ejemplo de funciones base Cosinusoidales 1D<sup>20</sup>

Así que, las expresiones matemáticas correspondientes a la DCT e IDCT son las presentadas a continuación en (2.6) y (2.7)<sup>21</sup>.

DCT – 1D:

$$F(k) = C(k) \sum_{n=0}^{N-1} f[n] * \cos\left(\frac{\pi(0.5+n)k}{2N}\right) \quad (2.6)$$

Para  $k = 0, 1, 2, 3, \dots, N - 1$

IDCT – 1D:

$$f[n] = \sum_{k=0}^{N-1} C(k) * F(k) * \cos\left(\frac{\pi(0.5+n)k}{2N}\right) \quad (2.7)$$

Para  $k = 0, 1, 2, 3, \dots, N - 1$

<sup>20</sup>. Tratamiento digital de imágenes. Rafael Gonzalez. 1996

<sup>21</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

Donde:

$f[n]$ : representa la secuencia de muestras discretas.

$F(k)$ : representa los coeficientes de la DCT de  $f[n]$

$N$ : número de coeficientes de la DCT

$$C(k) = \begin{cases} \sqrt{\frac{1}{N}}, & \text{para } k = 0 \\ \sqrt{\frac{2}{N}}, & \text{para } 1 \leq k \leq N - 1 \end{cases} .$$

### 2.3 – Transformada Discreta del Coseno Bidimensional (DCT-2D)

Por consiguiente, la Transformada Discreta del Coseno Bidimensional o DCT-2D se obtiene a partir de una extensión de la DCT-1D. Es decir, es aplicado sobre imágenes digitales que no vienen hacer otra cosa que matrices representando un conjunto de pixels. Por esta razón es necesario aplicar la DCT-1 consecutivamente, primero a cada una de las filas, y luego a cada una de las columnas.

La demostración es similar al caso de la DCT-1D, y se muestra a través de las siguientes expresiones matemáticas<sup>22</sup>.

$$Y_{N \times N} = C_{N \times N} I^T$$

$$X_{N \times N} = C_{N \times N} * Y^T = C_{N \times N} * I * C_{N \times N}^T \quad (2.8)$$

Asimismo, para el caso del cálculo de la DCT-2D de imágenes digitales, se obtiene a partir de un conjunto de imágenes bases ortogonales, las cuales son presentadas en la figura 2.2 para un caso de  $N=8$ .

---

<sup>22</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

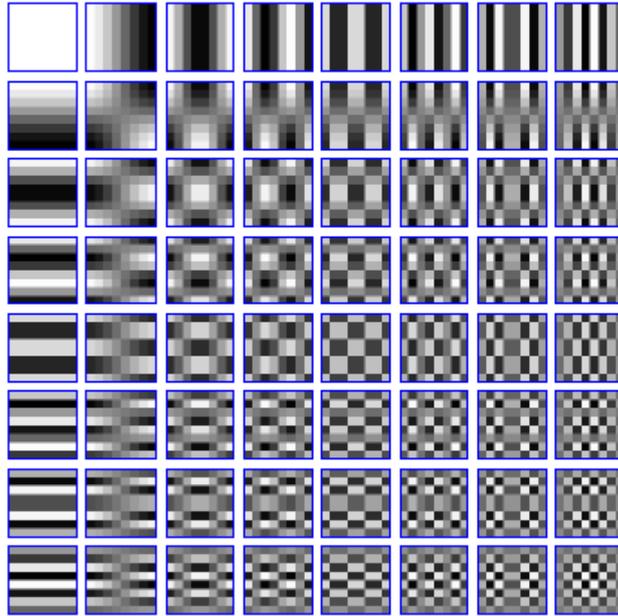


Figura 2.2 Ejemplo de funciones base Cosinusoidales 2D<sup>22</sup>

De la misma manera, existen expresiones matemáticas definidas para el cálculo de esta transformada en 2D. A continuación se muestran en las ecuaciones 2.9 y 2.10, la transformada directa y su respectiva transformada inversa<sup>23</sup>.

DCT – 2D :

$$F(u, v) = C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} f(x, y) * \cos\left(\frac{\pi(2x+1)u}{2N}\right) * \cos\left(\frac{\pi(2y+1)v}{2M}\right) \quad (2.9)$$

Para  $u, v = 0, 1, 2, 3, \dots, N - 1$

IDCT – 2D :

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{M-1} C(u)C(v) * F(u, v) * \cos\left(\frac{\pi(2x+1)u}{2N}\right) * \cos\left(\frac{\pi(2y+1)v}{2M}\right) \quad (2.10)$$

Para  $x, y = 0, 1, 2, 3, \dots, N - 1$

Donde:

$f(x,y)$ : representa la imagen compuesta de muestras discretas.

<sup>23</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

$F(u,v)$ : representa los coeficientes de la DCT-2D de  $f(x,y)$

$N, M$ : número de coeficientes de la DCT-2D

$$C(u), C(v) = \begin{cases} \sqrt{\frac{1}{N}}, & \text{para } u, v = 0 \\ \sqrt{\frac{2}{N}}, & \text{para } u, v = 1, 2, \dots, N-1 \end{cases}.$$

## 2.4 – Ejemplos Prácticos de la DCT-2D e IDCT-2D

Después de comprender el concepto básico de la transformada discreta del coseno bidimensional (DCT-2D) y su respectiva transformada inversa (IDCT-2D), se expone en la figura 2.3 el código fuente de dicha transformada y su respectiva inversa, los cuales han sido implementados en dos archivos basado en las expresiones matemáticas (2.9) y (2.10). Estos archivos llevan por nombre “dctdct2.m” e “idctidct2.m”, y serán utilizados bajo el entorno del software Matlab, con el propósito de ser empleados más adelante sobre algunas imágenes satelitales, como medida de ejemplo.



```

Editor - C:\idctidct2.m
File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + ÷ 1.1 x %>% %>% %>%
1 function iI = idctidct2(IT)
2
3 % Calcula la DCT-2D inversa a partir de la DCT-1D
4 % inversa. Es decir, primero se calcula sobre cada
5 % columna de la imagen transformada, y luego sobre
6 % cada fila pero en transpuesta.
7
8 % iI = idctidct2(IT)
9
10 % Carlos Flores Goycochea. Octubre 2009
11
12 [F,C] = size(IT);
13 N = F;
14 for i=1:C %PARA COLUMNAS
15     for n=0:N-1
16         A = IT(0+1,i)*cos(pi*(n+0.5)*0/N)*sqrt(1/N);
17         for k = 1:N-1
18             A = IT(k+1,i)*cos(pi*(n+0.5)*k/N)*sqrt(2/N) + A;
19         end
20         iI1(n+1,i) = A;
21     end
22 end
23 iI1 = iI1';
24 N = C;
25 for i=1:F %PARA FILAS
26     for n=0:N-1
27         A = iI1(0+1,i)*cos(pi*(n+0.5)*0/N)*sqrt(1/N);
28         for k = 1:N-1
29             A = iI1(k+1,i)*cos(pi*(n+0.5)*k/N)*sqrt(2/N) + A;
30         end
31         iI(n+1,i) = A;
32     end
33 end
34 iI = iI';

```

(b)

Figura 2.3 Algoritmo implementado de la (a) DCT-2D (b) IDCT-2D

Es necesario resaltar, que el propio Matlab cuenta con dos comandos o funciones denominados “dct2” e “idct2”, los cuales también realizan la misma operación del cálculo de la DCT-2D e IDCT-2D, respectivamente. Estas funciones pueden ser invocadas y observadas desde la ventana de comandos del Matlab de la siguiente forma:

>> help dct2

>> help idct2

Vea la figura 2.4 a) y b).

```
>> help dct2
DCT2 2-D discrete cosine transform.
  B = DCT2(A) returns the discrete cosine transform of A.
  The matrix B is the same size as A and contains the
  discrete cosine transform coefficients.

  B = DCT2(A,[M N]) or B = DCT2(A,M,N) pads the matrix A with
  zeros to size M-by-N before transforming. If M or N is
  smaller than the corresponding dimension of A, DCT2 truncates
  A.

  This transform can be inverted using IDCT2.

Class Support
-----
A can be numeric or logical. The returned matrix B is of
class double.

Example
-----
  RGB = imread('autumn.tif');
  I = rgb2gray(RGB);
  J = dct2(I);
  imshow(log(abs(J)),[], colormap(jet), colorbar)

The commands below set values less than magnitude 10 in the
DCT matrix to zero, then reconstruct the image using the
inverse DCT function IDCT2.

  J(abs(J)<10) = 0;
  K = idct2(J);
  figure, imshow(I)
  figure, imshow(K,[0 255])

See also fft2, idct2, ifft2.

Reference page in Help browser
doc dct2
```

(a)

```

>> help idct2
IDCT2 2-D inverse discrete cosine transform.
  B = IDCT2(A) returns the two-dimensional inverse discrete
  cosine transform of A.

  B = IDCT2(A,[M N]) or B = IDCT2(A,M,N) pads A with zeros (or
  truncates A) to create a matrix of size M-by-N before
  transforming.

For any A, IDCT2(DCT2(A)) equals A to within roundoff error.

The discrete cosine transform is often used for image
compression applications.

Class Support
-----
The input matrix A can be of class double or of any
numeric class. The output matrix B is of class double.

Example
-----
  RGB = imread('autumn.tif');
  I = rgb2gray(RGB);
  J = dct2(I);
  imshow(log(abs(J)),[], colormap(jet), colorbar

The commands below set values less than magnitude 10 in the
DCT matrix to zero, then reconstruct the image using the
inverse DCT function IDCT2.

  J(abs(J)<10) = 0;
  K = idct2(J);
  figure, imshow(I)
  figure, imshow(K,[0 255])

See also dct2, dctmtx, fft2, ifft2.

Reference page in Help browser
doc idct2

```

(b)

Figura 2.4 a) Ayuda de la función “DCT2” b) Ayuda de la función “IDCT2”. Ambos pertenecientes al software Matlab

Si bien es cierto que el uso del software Matlab facilita la utilización de diversos algoritmos porque ya se encuentran implementados en funciones, siempre existe la inquietud por conocer como se implementan estos a nivel de programación básica utilizando sentencias conocidas por nosotros, tales como el IF y el FOR. Por tal razón, como propósito de independizarnos más adelante del Matlab y cuando se desee trabajar a nivel de hardware (que no es el propósito de esta tesis pero sí podría ser el de otra más adelante), se ha planteado el algoritmo real para el cálculo de la DCT-D que podría ser de utilidad para algún tesista en el futuro.

Como ejemplo, en la figura 2.5 se muestra el resultado de aplicar la DCT-2D sobre una imagen satelital de 900 x 852 pixels, en tonos de gris y codificado con 08 bits por pixel. El cálculo de la DCT-2D está basado en el algoritmo implementado que se presentó en la figura 2.3. A continuación se describe el código empleado para tal aplicación, desde la ventana de comando del Matlab.

```
>> II = imread('vaporagua.bmp'); % lectura de imagen

>> I = rgb2gray(II); % transformación de color a gris

>> I = double( I ); % transformación a variable double

>> dct_I = dctdct2( I ); % transformada DCT-2D

>> figure(1), colormap(gray(256)) % configuración a 256 niveles de gris

>> subplot( 1 , 2 , 1 ), image( I ) % gráfica imagen original

>> subplot( 1 , 2 , 2 ), image( dct_I ) % gráfica imagen transformada
```

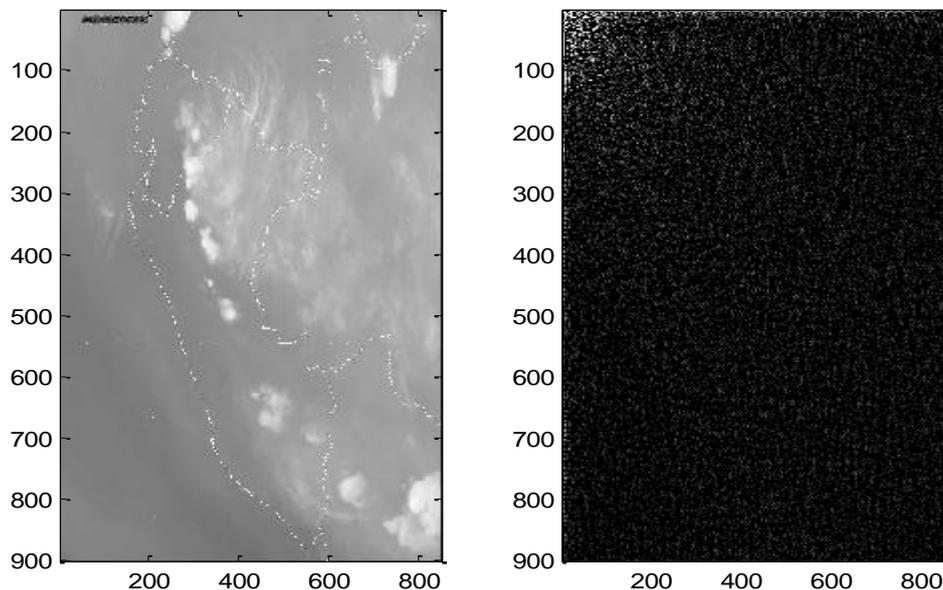


Figura 2.5 Izquierda: Imagen Satelital original de Vapor de Agua. Derecha: Resultado de la DCT-2D de la Imagen Satelital.

Obtenida la DCT-2D de una imagen satelital de nuestro litoral mostrando la presencia del vapor de agua, se observa una aparición pequeña de pixeles claros en la parte

superior izquierda de la imagen transformada, y asimismo, una gran cantidad de píxeles oscuros en el resto de la imagen. Esto simboliza la gran demanda de coeficientes de baja energía y por el contrario poca demanda de coeficientes de alta energía, lo cual indica que la DCT-2D ha logrado concentrar la energía en un número reducido de coeficientes, a los cuales se les conoce como coeficientes DC, y son estos los que darán fruto a la operación de compresión. Se conoce como coeficientes DC, a todos aquellos coeficientes, resultantes de la transformada discreta del coseno, que almacenan la mayor concentración de energía y son de mayor importancia porque de ellos dependerá el grado de compresión a alcanzar como también la calidad de la imagen a recuperar después de aplicar la operación inversa: la transformada discreta del coseno inversa.

Ahora, se procede a aplicar la IDCT-2D sobre la imagen transformada y haciendo uso de la función “idctidct2.m” que fue creada y mostrada en la figura 2.3. A continuación se describe el código empleado para tal operación, desde la ventana de comando del Matlab.

```
>> rec_I = idctidct2( dct_I );           % transformada IDCT-2D

>> figure(2), colormap(gray(256))      % configuración a 256 niveles de gris

>> subplot( 1 , 2 , 1 ), image( I )     % grafica la imagen original

>> subplot( 1 , 2 , 2 ), image( rec_I ) % grafica la imagen recuperada
```

Y, la figura 2.6, muestra una comparación entre la imagen original y la imagen recuperada después de haber aplicado la IDCT-2D. Visiblemente se observa una similitud total entre ambas imágenes, lo cual representa una recuperación completa de la imagen una vez que se le ha aplicado la transformación inversa. Esto sucede siempre y cuando la imagen transformada no sea cuantificada y mucho menos codificada. Por esta razón la pérdida es íntegramente nula; sin embargo, en el capítulo posterior se realizarán las operaciones de cuantificación y codificación con la finalidad de comprimir una imagen de interés con relativa tasa de compresión, para mantener la calidad e información en el grupo de imágenes satelitales que serán utilizadas como parte de este trabajo de investigación.

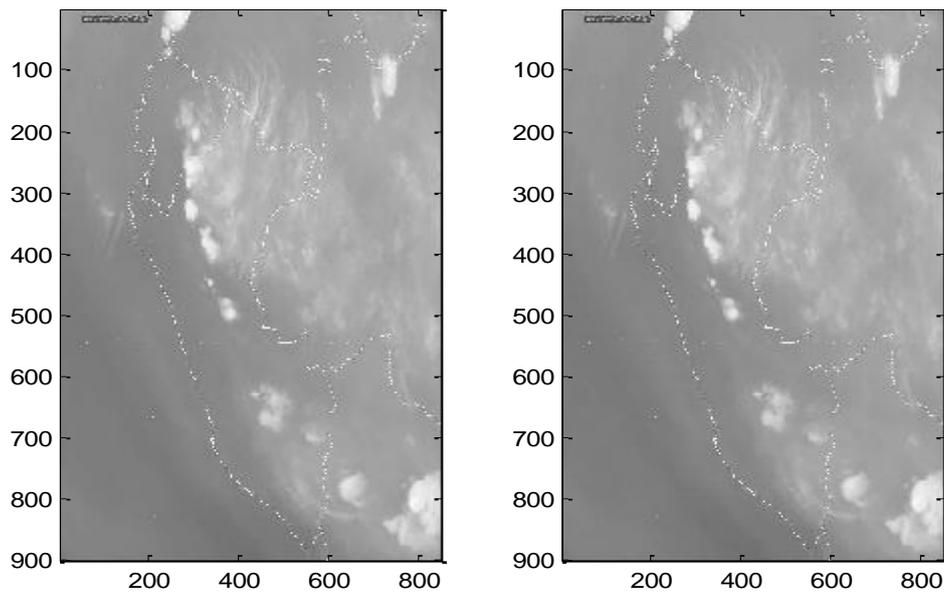


Figura 2.6 Izquierda: Imagen Satelital original de Vapor de Agua. Derecha: Resultado de la IDCT-2D de la imagen transformada.

## 2.5 – Aplicaciones de la DCT

Al consultar diversas bibliografías y sitios de Internet, se llega a la conclusión que la Transformada Discreta del Coseno, después de la Transformada Discreta de Fourier, es la más ampliamente utilizada. Esto se debe a las sorprendentes características que posee la DCT de compactación de energía, aproximándose al desempeño que otorga la Transformada de Karhunen-Loève (KLT) cuando logra descorrelacionar una señal estadísticamente.

Por lo tanto, se tiene a la DCT como una de las principales herramientas de compresión de datos, sea para aplicaciones en videofonos, teleconferencias, transmisión progresiva de imágenes, HDTV, entre otras. Inmediatamente se relata algunas de esas aplicaciones.

### 2.5.1 – Codificación de Imagen

La DCT, al igual que otras transformadas discretas, es aplicada a la codificación de imágenes, desde un punto de vista de reducción del ancho de banda o compresión de datos. Esto señala un objetivo primordial, que es lograr que una imagen, en el dominio espacial, o una secuencia de imágenes, en el dominio espacial-temporal, se traslade a un

dominio transformado de tal forma que se comprima el ancho de banda para la transmisión o para el almacenamiento; de tal forma que la recuperación de la imagen o secuencia de imágenes mediante la transformada inversa, no presente distorsión alguna y perceptible. A pesar de existir muchas medidas cuantitativas que hacen referencia a la calidad de la imagen codificada (error cuadrático medio, relación señal a ruido, etc), existe también la manera subjetiva o visual para evaluar el desempeño de la transformada DCT. A esto se le debe sumar algunos otros factores tales como la complejidad de implementación y su realización en hardware.

Más allá del resultado o complejidad que ofrece una DCT, se encuentra la forma de aplicación de esta transformada sobre una imagen o secuencia de imágenes. Para ello, es usual aplicar la DCT-2D sobre sub-bloques de la imagen a codificar. Estos sub-bloques, por lo general, corresponden a pequeñas matrices de 8x8 píxeles, y que es justamente la recomendación que hace el estándar JPEG para la aplicación de la DCT sobre imágenes estáticas<sup>24</sup>. Una vez que se haya aplicado la DCT-2D sobre cada sub-bloque de 8x8 píxeles, se procede a descartar algunos coeficientes del grupo de baja energía según la importancia que tienen en la obtención de la IDCT-2D. Este procedimiento realizado es seguido de una cuantificación y codificación de los coeficientes de alta energía, según el grado de compresión y calidad de imagen a alcanzar.

Este procedimiento es prácticamente la parte fundamental de este tema de tesis, por lo tanto será tocado con más detalles, tanto gráficos, numéricos y expresiones matemáticas, en un capítulo posterior. Mientras tanto, se visualiza a manera de diagrama de bloques, la forma de realización de la codificación de una imagen digital señalando tanto los bloques que conforman la etapa de transmisión, como los bloques que conformarían la etapa de recepción. Esto es posible observarlo de manera clara en la figura 2.7, que es mostrada seguidamente.

## **2.5.2 – Transmisión Progresiva de Imágenes**

Otra de las aplicaciones de la DCT es la transmisión progresiva de imágenes sobre canales con un ancho de banda reducido, tal como se puede encontrar en una línea telefónica. Es decir, si una imagen se transmite sobre canales de bandas reducidas, el tiempo de envío de toda la imagen llevaría un tiempo muy prolongado. Por lo tanto,

---

<sup>24</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

existen formas de agilizar tales transmisiones utilizando métodos y/o algoritmos que hagan que la calidad de una imagen se vaya creando jerárquicamente. Entre las principales aplicaciones de la transmisión progresiva de imágenes, encontramos: tele-conferencia, tele-diagnóstico médico, servicios de seguridad, tele-presencia, etc.

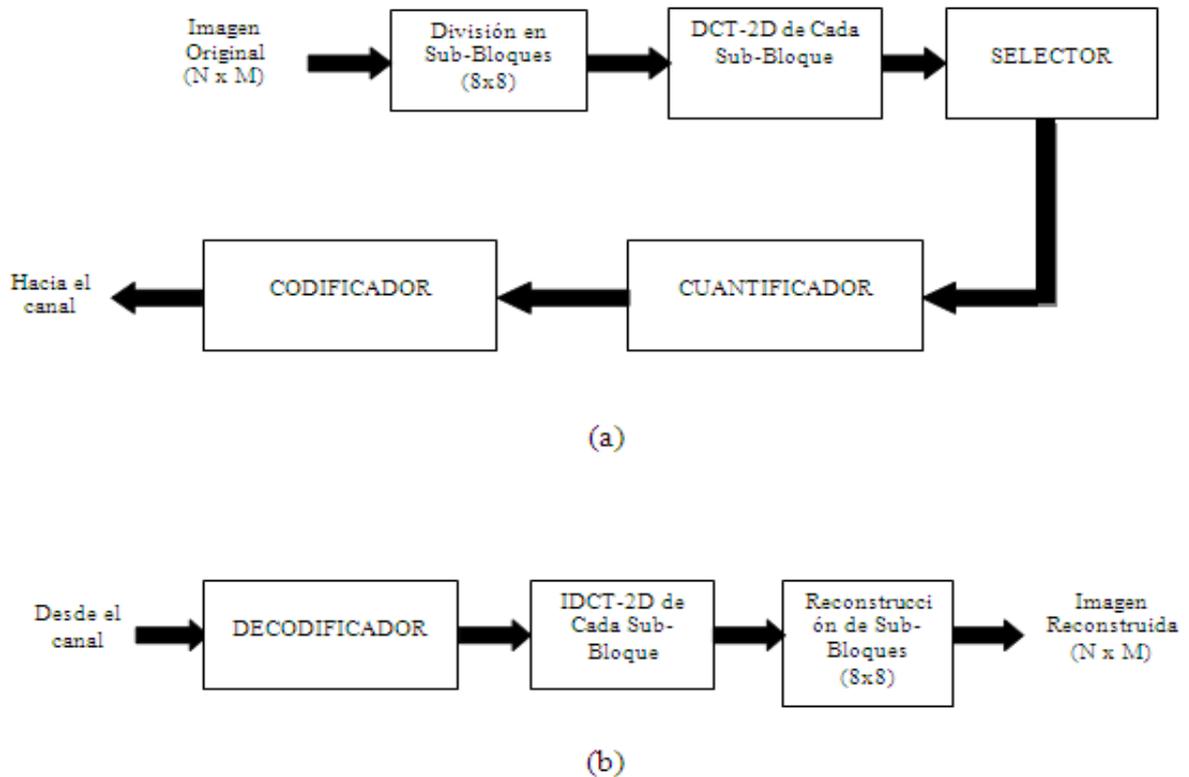


Figura 2.7 Diagrama de bloques de la Codificación de imágenes<sup>25</sup>. a) Transmisor. b) Receptor.

Seguidamente se describe de manera general y resumida, como una imagen de  $N \times M$  pixels puede ser transmitida de forma progresiva. Como primer paso la imagen es dividida en bloques menores de  $P \times Q$  pixels acomodados de una forma particular, para luego calcular la DCT-2D sobre cada sub-bloque de  $P \times Q$  pixels. De esta forma, después de cuantificar y elegir adecuadamente los coeficientes de alta energía de cada sub-bloque transformado, se transmiten en forma grupal o secuencial tales coeficientes. Por el otro extremo, el receptor recibirá cada grupo de coeficientes y pasará a calcular la IDCT-2D, para luego re-acomodar los bloques y tener una reconstrucción de la imagen

<sup>25</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

al tamaño original. Este procedimiento posee variantes en el momento de la transmisión de los coeficientes, los cuales puede realizarse de forma adaptativa o basado en la estadística de los mismos. Lo descrito anteriormente, puede ser observado en la figura 2.8 a través de un diagrama de bloques.

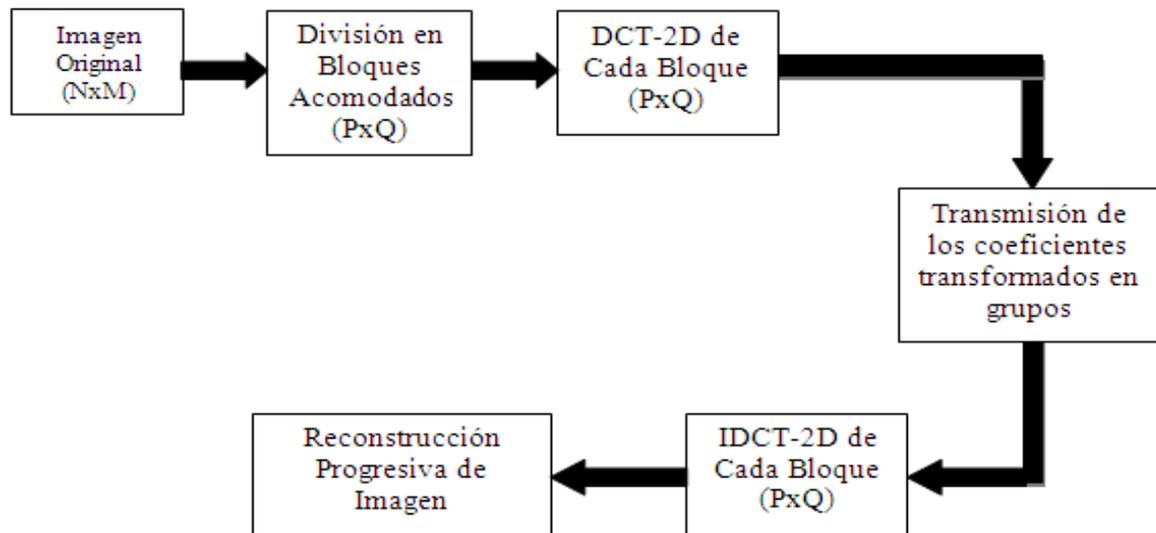


Figura 2.8 Diagrama de bloques de la División/Reconstrucción Progresiva<sup>26</sup>

Del mismo modo, la figura 2.9 representa el procedimiento descrito, líneas arriba, sobre una imagen de 350x350 pixels, que es dividida en 4 bloques y en donde se utiliza solo el 20% de los coeficientes de alta energía en la transmisión. Esta imagen corresponde a una captura realizada por el satélite meteorológico GOES 10 (satélite geoestacionario operacional ambiental)<sup>26</sup>.

<sup>26</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

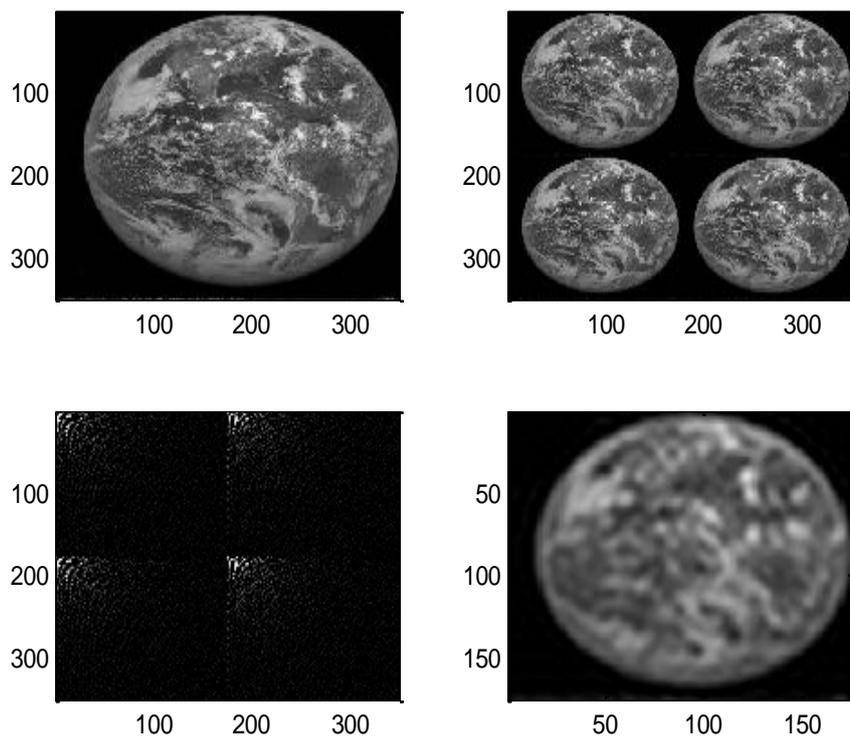


Figura 2.9 Ejemplo de Transmisión Progresiva en una Imagen del GOES 10

# CAPÍTULO 3

## TÉCNICA DE COMPRESIÓN SIN PÉRDIDA

Este tercer capítulo trata de la técnica de compresión sin pérdidas utilizado en el desarrollo de este trabajo de tesis. Para ello se inicia con una introducción seguido del concepto de la técnica utilizada; luego, se muestra un ejemplo práctico de cómo es utilizada dando las pautas necesarias para su total entendimiento, y finalmente se citan algunas aplicaciones muy comunes donde se utilizan algoritmos de compresión sin pérdidas.

### **3.1 Introducción**

Son muchas las imágenes que necesitan ser comprimidas, es decir disminuir su representación de bits por pixel, de una manera que no sufra modificación alguna. Tal es el ejemplo de muchas imágenes médicas en donde es crucial alterar alguna intensidad o intensidades de gris en la imagen original. Pues la alteración de un pixel en una imagen de este tipo, podría traer como consecuencia falsos diagnósticos en el paciente, lo que podría dar origen a un tratamiento equivocado para la persona equivocada y con ello hasta una consecuencia nefasta para la salud.

Por esta razón en el mundo real no todas las imágenes pueden someterse a una técnica de codificación con pérdidas, sino a veces es necesario complementarlo con una técnica que codifique pero que no origine pérdidas durante la decodificación. Esto es una gran ventaja que caracteriza a este grupo de técnicas, pero como desventaja se tiene la poca disminución del número de bits por pixel. Esto quiere decir que una técnica de compresión sin pérdidas puede lograr una compresión de 50% hasta 20%, según el tipo de imagen que se codifique.

En la actualidad son muchas las técnicas de compresión sin pérdidas que existen, desde las más utilizadas para imágenes binarias o imágenes codificadas con un bit, hasta aquellas más sofisticadas que son utilizadas sobre imágenes con 8 o más bits por pixel.

Así tenemos, la codificación de Huffman, la codificación aritmética, el código de longitud de cadenas, entre otros<sup>27</sup>.

Para el desarrollo de este trabajo, se eligió la codificación Huffman por tratarse de una técnica de compresión utilizada muy frecuentemente y con muy buenos resultados de tasa de bits. Además, resulta ser una técnica que el JPEG también utiliza en la compresión de imágenes estáticas.

## 3.2 Codificación de Huffman

Antes de definir la codificación Huffman es necesario comentar acerca de la teoría de la información. Esta teoría proporciona la herramienta matemática para determinar la mínima cantidad de datos que resulte ser suficiente para describir completamente la imagen sin pérdida de información. Por ello, un evento con poca probabilidad de ocurrencia se dice que contiene más información que un evento que es más probable de ocurrir. Por ejemplo, si consideramos las tres afirmaciones:

Los peces mueren si permanecen mucho tiempo fuera del agua.

Mañana lloverá

En 30 los robots podrán sentir emociones.

Nos preguntaríamos, cuál de esas tres afirmaciones contiene mayor información. Analizando cada una determinamos lo siguiente. La primera afirmación es un evento conocido por lo tanto tiene poca información y le corresponde una probabilidad de ocurrencia igual al 100%. La segunda afirmación tiene más información que la primera, por lo tanto le corresponde una probabilidad menor al 100%, en cuanto que la tercera afirmación contiene mucho más información por tratarse de un evento que muy poco probable.

De esta manera, la información es medible a partir de la siguiente expresión<sup>28</sup>.

---

<sup>27</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

<sup>28</sup> Tratamiento digital de imágenes. Rafael Gonzalez. 1996.

$$I(E) = \log_r \left( \frac{1}{P(E)} \right) = -\log_r P(E) \quad (3.1)$$

Donde, un evento aleatorio E que ocurre con probabilidad P(E) contiene I(E) unidades de información. La base “r” del logaritmo determina la unidad usada para medir la información<sup>28</sup>. Para el caso binario, se emplea “r” igual a 2.

Seguidamente, para determinar la cantidad media de información o también llamada entropía se utiliza la siguiente expresión<sup>28</sup>:

$$H(z) = -\sum_{j=1}^L P(a_j) \log_r P(a_j) \quad (3.2)$$

La cual posee la unidad de bits en el caso de considerarse “r” igual a 2. Asimismo, L denota la cantidad de pixels cuando se trata de una imagen.

Por ejemplo, sea la imagen “I” de 3x3 pixels y codificada con 4 bpp (bits por pixel). La probabilidad de ocurrencia de cada pixel será:

$$I = \begin{vmatrix} 15 & 12 \\ 12 & 10 \\ 15 & 15 \end{vmatrix}$$

$$P(15) = 3 / 6$$

$$P(12) = 2 / 6$$

$$P(10) = 1 / 6$$

Por lo tanto, la entropía será:

$$H(z) = - ( (3/6) * \log_2(3/6) * 3 + (2/6) * \log_2(2/6) * 2 + (1/6) * \log_2(1/6) )$$

$$H_0(z) = 2.9875 \text{ bits.}$$

Esto indica que se necesitarán por lo menos 2.9875 bits por pixel para codificar esta imagen “I” sin perder información alguna.

La codificación de Huffman es un tipo de código de longitud mínima. Esto significa que dada la distribución estadística de los niveles de gris, es decir el histograma, el algoritmo de Huffman genera un código lo más próximo posible al límite mínimo, es decir a la entropía.<sup>29</sup>

Este tipo de método genera un código de longitud variable donde las palabras de código tienen longitudes desiguales. Para imágenes complejas se consigue una reducción entre el 10% y 50%, pero esta razón de compresión puede aumentarse si se pre procesa la imagen para eliminar la información no relevante<sup>30</sup>.

A continuación se describe los pasos a seguir para aplicar el algoritmo de Huffman sobre una imagen de tonos de gris:

Obtener el histograma de la imagen e inmediatamente obtener las probabilidades de los niveles de gris.

Ordenar las probabilidades de menor a mayor

Combinar por medio de una suma las dos menores probabilidades.

Retornar al paso 2 hasta que queden solo 2 probabilidades.

Retornar de la última columna hacia la primera, generando códigos y alternando las asignaciones de “0” y “1”,

### **3.3 Ejemplo Práctico**

Una vez entendido lo que es la codificación Huffman, consecutivamente se plantea un ejemplo que servirá para entender de forma clara y precisa la forma de cómo se codifica utilizando esta técnica de compresión sin pérdida.

Suponiendo que se desea transmitir la siguiente oración conformada por 34 caracteres ASCII, donde cada carácter es representado por 8 bits.

---

<sup>29</sup> Tratamiento digital de imágenes. Rafael Gonzalez. 1996.

<sup>30</sup> Tratamiento digital de imágenes. Rafael Gonzalez. 1996.

“LOLO PONE POMADA A LA ROPA DE LOLA”

Esto indica que serán necesarios  $34 \times 8 = 272$  bits para representar indicada oración. Sin embargo, haciendo uso del árbol de Huffman, será posible utilizar un número menor de bits para representar la información y sobre todo para conseguir transmitirla en un menor tiempo posible sin tener pérdida alguna en la codificación.

Entonces, como punto de partida se procede a ordenar cada carácter de mayor a menor (columna 1ra) según el grado de frecuencia de aparición (probabilidad) que tienen en la frase anteriormente propuesta. Vea la Tabla N° 3.1.

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	" "	7 / 34								
2	O	6 / 34								
3	A	6 / 34								
4	L	5 / 34								
5	P	3 / 34								
6	E	2 / 34								
7	D	2 / 34								
8	M	1 / 34								
9	N	1 / 34								
10	R	1 / 34								

Tabla N° 3.1. Ordenamiento de probabilidades

El siguiente paso consiste en sumar las dos menores probabilidades y reubicar dicho resultado más las probabilidades anteriores, en la columna denotada por “2da”.

Realizar el mismo procedimiento anterior pero esta vez sobre la columna “2da”, y el nuevo resultado será ingresado en la columna “3ra”. Esta operación deberá de realizarse en forma consecutiva hasta quedarnos solo con dos probabilidades, que al sumarlas darán el valor de 34. Este último resultado se ubica en la columna denotada por “9na”.

Tener presente que aquellas probabilidades que no intervinieron en la suma, mantienen su valor y se reubican en la columna siguiente en función al ordenamiento de mayor a menor.

Para un mejor entendimiento de lo anteriormente narrado, observe la Tabla N° 3.2.

	CARACT.	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”	7/34	7/34	7/34	7/34	7/34	9/34	12/34	13/34	21/34
2	O	6/34	6/34	6/34	6/34	6/34	7/34	9/34	12/34	13/34
3	A	6/34	6/34	6/34	6/34	6/34	6/34	7/34	9/34	
4	L	5/34	5/34	5/34	5/34	6/34	6/34	6/34		
5	P	3/34	3/34	3/34	4/34	5/34	6/34			
6	E	2/34	2/34	3/34	3/34	4/34				
7	D	2/34	2/34	2/34	3/34					
8	M	1/34	2/34	2/34						
9	N	1/34	1/34							
10	R	1/34								

Tabla N° 3.2. Resultado del ordenamiento de probabilidades

Finalizada esta operación se continúa con la codificación, es decir la asignación de bits desde la última columna hacia la primera. Tal asignación de bits se inicia otorgando el “0” a la primera posición, y un “1” a la segunda.

De igual manera para la columna anterior, pero teniendo presente que en el caso de aquellas probabilidades que se sumaron y dieron como resultado una probabilidad en la siguiente columna, se debe anteceder un código binario. Este código binario corresponde a la probabilidad resultante de la suma de dos probabilidades previas. Por ejemplo:

El desarrollo paso a paso para la asignación de bits, es mostrado de manera clara y metódica, columna por columna, en las tablas siguientes: Tabla N° 3.3, Tabla N° 3.4, Tabla N° 3.5, Tabla N° 3.6, Tabla N° 3.7, Tabla N° 3.8, Tabla N° 3.9, Tabla N° 3.10, y Tabla N° 3.11.

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”									0
2	O									1
3	A									
4	L									
5	P									
6	E									
7	D									
8	M									
9	N									
10	R									

Tabla N° 3.3. Asignación de bits para la columna “9na”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”								1	0
2	O								00	1
3	A								01	
4	L									
5	P									
6	E									
7	D									
8	M									
9	N									
10	R									

Tabla N° 3.4. Asignación de bits para la columna “8va”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”							00	1	0
2	O							01	00	1
3	A							10	01	
4	L							11		
5	P									
6	E									
7	D									
8	M									
9	N									
10	R									

Tabla N° 3.5. Asignación de bits para la columna “7ma”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”						01	00	1	0
2	O						10	01	00	1
3	A						11	10	01	
4	L						000	11		
5	P						001			
6	E									
7	D									
8	M									
9	N									
10	R									

Tabla N° 3.6. Asignación de bits para la columna “6ta”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”					10	01	00	1	0
2	O					11	10	01	00	1
3	A					000	11	10	01	
4	L					001	000	11		
5	P					010	001			
6	E					011				
7	D									
8	M									
9	N									
10	R									

Tabla N° 3.7. Asignación de bits para la columna “5ta”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”				10	10	01	00	1	0
2	O				11	11	10	01	00	1
3	A				000	000	11	10	01	
4	L				010	001	000	11		
5	P				011	010	001			
6	E				0010	011				
7	D				0011					
8	M									
9	N									
10	R									

Tabla N° 3.8. Asignación de bits para la columna “4ta”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”			10	10	10	01	00	1	0
2	O			11	11	11	10	01	00	1
3	A			000	000	000	11	10	01	
4	L			010	010	001	000	11		
5	P			0010	011	010	001			
6	E			0011	0010	011				
7	D			0110	0011					
8	M			0111						
9	N									
10	R									

Tabla N° 3.9. Asignación de bits para la columna “3ra”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”		10	10	10	10	01	00	1	0
2	O		11	11	11	11	10	01	00	1
3	A		000	000	000	000	11	10	01	
4	L		010	010	010	001	000	11		
5	P		0010	0010	011	010	001			
6	E		0110	0011	0010	011				
7	D		0111	0110	0011					
8	M		00110	0111						
9	N		00111							
10	R									

Tabla N° 3.10. Asignación de bits para la columna “2da”

	CARACTER	1ra	2da	3ra	4ta	5ta	6ta	7ma	8va	9na
1	“ ”	10	10	10	10	10	01	00	1	0
2	O	11	11	11	11	11	10	01	00	1
3	A	000	000	000	000	000	11	10	01	
4	L	010	010	010	010	001	000	11		
5	P	0010	0010	0010	011	010	001			
6	E	0110	0110	0011	0010	011				
7	D	0111	0111	0110	0011					
8	M	00111	00110	0111						
9	N	001100	00111							
10	R	001101								

Tabla N° 3.11. Asignación de bits para la columna “1ra”

Después de alcanzar la primera columna, se ha logrado realizar la codificación final de cada carácter. En la Tabla N° 3.12 se muestra el código correspondiente a cada carácter, indicando la probabilidad y el número de bits utilizado.

	CARACT.	Probabilidad	Codificación	Nº Bits
1	“ ”	7/34	10	2
2	O	6/34	11	2
3	A	6/34	000	3
4	L	5/34	010	3
5	P	3/34	0010	4
6	E	2/34	0110	4
7	D	2/34	0111	4
8	M	1/34	00111	5
9	N	1/34	001100	6
10	R	1/34	001101	6

Tabla N° 3.12. Asignación de bits a cada carácter

Luego, en la Tabla N° 3.13 se muestra la comparación del número de bits utilizados cuando se codifica directamente cada carácter con un número fijo de bits, con la cantidad de bits utilizados utilizando la codificación de Huffman. Se puede observar que el total de bits sin codificación es 272, mientras que para el caso codificado es de 104 bits. Por lo tanto, el grado de compresión en este ejemplo es de  $272/104 = 2.62$ . Este factor señala una compresión muy pobre pero garantiza la recuperación total de la información sin perder ni un solo bit en el mensaje transmitido.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
FRASE	L	O	L	O		P	O	N	E		P	O	M	A	D	A	
SIN CODIFICACION	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
CON CODIFICACION	3	2	3	2	2	4	2	6	4	2	4	2	5	3	4	3	2

18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	TOTAL
A		L	A		R	O	P	A		D	E		L	O	L	A	
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	272 bits
3	2	3	3	2	6	2	4	3	2	4	4	2	3	2	3	3	104 bits

Tabla N° 3.13. Codificación de la frase utilizada. Total de bits utilizados

A continuación se presenta la cadena de bits que codifica el mensaje anteriormente señalado, esto como forma de comparación entre la codificación con ASCII y la Huffman.

## **Codificación ASCII**

“LOLO PONE POMADA A LA ROPA DE LOLA”

01001100. 01001111 01001100. 01001111.00100000.01010000.  
01001111.01001110. 01000101.00100000.01010000.01001111. 01001101  
01000001 01000100.01000001.00100000.01000001.  
00100000.01001100.01000001.00100000.01010010.01001111.  
01010000.01000001.00100000.01000100.01000101.00100000.  
01001100.01001111.01001100.01000001

## **Codificación Huffman**

“LOLO PONE POMADA A LA ROPA DE LOLA”

010.11.010.11.10.0010.11.001100.0110.10.0010.11.00111.000.0111.000.10.000.10.010  
.000.10.001101.11.0010.000.10.0111.0110.10.010.11.010.000

**Se puede notar a simple vista que al momento de decodificar la cadena de bits, nunca se creará confusión con los símbolos o mensajes transmitidos debido a que la codificación Huffman asigna un único y exclusivo conjunto de bits a cada carácter separadamente.**

# CAPÍTULO 4

## TÉCNICA DE CODIFICACIÓN DESARROLLADA

Este capítulo se encarga de describir la técnica de codificación desarrollada y planteada para este trabajo de tesis. Esta técnica está comprendida por la combinación de la técnica de compresión con pérdida más la técnica de compresión sin pérdida aplicada sobre las imágenes satelitales y digitalizadas en niveles de gris. En este capítulo se toca una breve introducción de la técnica de codificación desarrollada, luego la etapa de pre procesamiento que a su vez comprende el filtrado en 2D, el cálculo de la transformada discreta del coseno (DCT), ordenamiento y normalización de los coeficientes de dicha transformada, para culminar con la etapa de codificación o asignación de bits por medio de la codificación Huffman. Se finaliza mostrando el algoritmo de codificación y el respectivo ejemplo con ayuda del software Matlab.

### **4.1 Introducción**

Existen muchas alternativas para codificar una imagen digital, encontrándose una gran diferencia entre ellas en el tiempo de procesamiento utilizado, el grado de compresión y la relación señal a ruido que refleja el nivel de calidad en la imagen comprimida. Por tal motivo se ha elegido una combinación de dos técnicas diferentes. Por un lado aquella que ofrece pérdida, y por el otro lado aquella que no lo ofrece. De esta forma se controla el error alcanzado en el proceso de cuantificación, después de aplicar la Transformada Discreta del Coseno en 2D (DCT-2D).

El estándar JPEG utiliza de manera similar la combinación de ambas técnicas planteadas en esta tesis, codificando los coeficientes ordenados de la DCT-2D según una tabla previamente construida con la finalidad de asignar los bits respectivamente. Es por ello que en este trabajo de investigación se procede de manera diferente planteando una alternativa simple con un ligero sacrificio del grado de compresión; pero así mismo, ganar una considerable calidad en la imagen comprimida.

El punto de partida es el filtrado de la imagen a través de un filtro de suavizado. Una vez filtrada la imagen, se procede a realizar una operación de sustracción entre la imagen original y la imagen filtrada, lo cual da origen a una imagen de error. De esta forma, la técnica de codificación con pérdida se aplica sobre la imagen filtrada, la cual hace posible ordenar los coeficientes de la DCT-2D en un conjunto de cantidades con alto grado de números ceros. Mientras que a la imagen diferencia o imagen de error se le codifica con Huffman, debido a que este código se basa en asignar más bits a aquello que menos aparece, y menor cantidad de bits a aquellos que son más repetitivos.

## 4.2 Pre Procesamiento de la imagen

Antes de iniciar con el cálculo de la DCT-2D sobre la imagen a codificar, se aplica una operación de suavizado haciendo uso de un filtro digital pasabajo en 2D. Esto tiene como finalidad reducir el ruido y otros efectos no deseados que pueden estar presentes en una imagen digital como resultado del muestreo, cuantización y transmisión, o bien por perturbaciones en el sistema tales como partículas de polvo en el sistema óptico<sup>31</sup>.

Después de un análisis entre los diferentes tipos de filtros pasabajos en 2D, se eligió uno correspondiente a la máscara Smooth<sup>32</sup>. Esta elección se hizo debido a la respuesta en frecuencia que posee dicho filtro y a la facilidad que posee en suavizar una imagen digital. A continuación, en la ecuación 4.1, se muestra la máscara Smooth utilizada  $H_s$ <sup>32</sup>:

$$H_s[m, n] = \frac{1}{16} \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix} \quad (4.1)$$

Haciendo uso de los comandos del Matlab entre ellos la transformada rápida de Fourier, se llegó a graficar el módulo de la transformada de Fourier en 2D de la ventana señalada.

```
>> h = [ 1 2 1 ; 2 4 2 ; 1 2 1 ] / 16;
```

<sup>31</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

<sup>32</sup> Tratamiento digital de imágenes. Rafael Gonzalez. 1996.

```

>> H1 = fft( h , 256 );

>> H2 = fft( H1' , 256 );

>> H2 = fftshift( H2 );

>> H12 = abs( H2 );

>> mesh( H12 )

>> colormap( jet(256) )

```

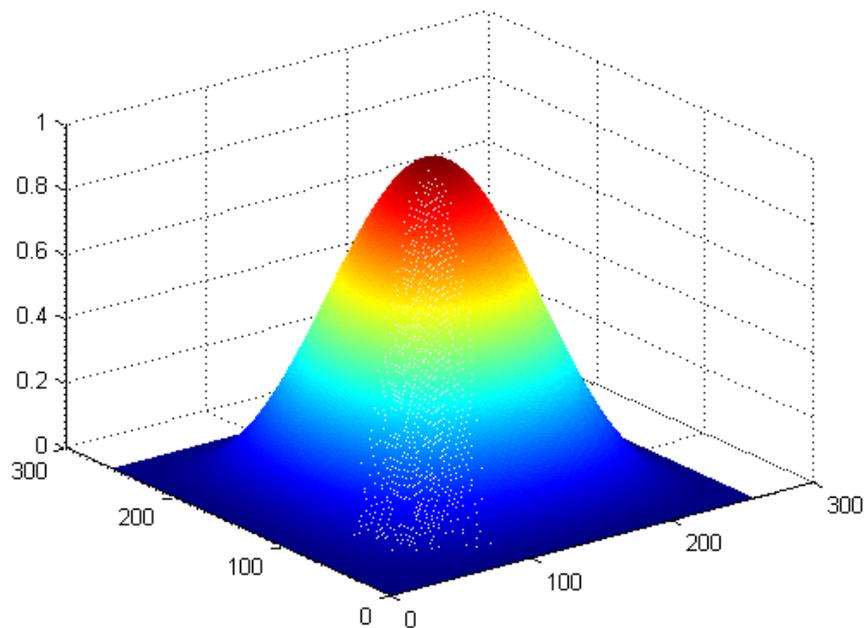


Figura 4.1. Respuesta en frecuencia en 2D de la máscara Smooth

#### 4.2.1 Filtrado y operación de substracción de la imagen

La operación de filtrado en 2D es equivalente a una operación de convolución en 2D o también denominado filtrado espacial. Pues la máscara recorre toda la imagen por filas y columnas realizando la convolución sobre zonas conformadas por grupos de 3x3 pixels. Seguidamente se muestra la expresión matemática correspondiente a la

operación de filtrado espacial en 2D, la cual fue realizada con ayuda del comando CONV2 del toolbox de Signal Processing del Matlab<sup>33</sup>.

$$I_f [m, n] = I [m, n] \otimes h [m, n] = \sum_{j=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} I [j, k] h [m-j, n-k] \quad (4.2)$$

Donde:

$I_f [m, n]$ : corresponde a la imagen filtrada o convolucionada

$I [m, n]$ : imagen inicial a la cual se le aplicará la ventana 2D

$h [m, n]$ : ventana Smooth.

```
>> load ImSat1
```

```
>> figure( 1 )
```

```
>> colormap( gray(256) )
```

```
>> subplot( 1 , 2 , 1 ), image( ImSat1 )
```

```
>> [ m , n ] = size( ImSat1 )
```

```
m =
```

```
896
```

```
n =
```

```
848
```

```
>>ImSat1_Z=[zeros(1,n+2);zeros(m,1) ImSat1 zeros(m,1);zeros(1,n+2)];
```

```
>> h = [ 1 2 1 ; 2 4 2 ; 1 2 1 ] / 16;
```

```
>> A = conv2( h , ImSat1_Z , 'full' );
```

```
>> [ m , n ] = size( A )
```

---

<sup>33</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

m =

900

n =

852

```
>> %Retirando las filas y columnas adicionales de la imagen resultante
```

```
>> ImSat1_F = A( 2:m-1 , 2:n-1 );
```

```
>> figure(1), colormap(gray(256))
```

```
>> subplot( 1,2,2 ) , image( ImSat1_F )
```

La figura 4.2 muestra el resultado de aplicar la convolución en 2D.

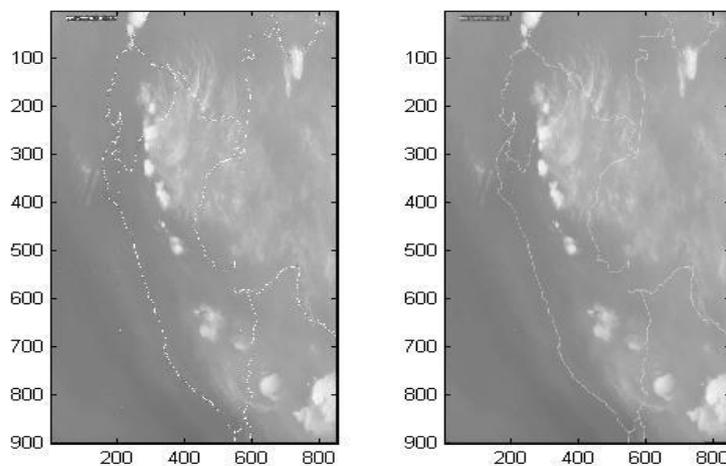


Figura 4.2. a) Imagen original. b) Imagen filtrada con la máscara Smooth.

Posteriormente, se continúa con la operación de sustracción entre la imagen original y la imagen filtrada. Esta diferencia da origen a la denominada imagen error o imagen diferencia. De la figura 4.3 es posible observar a la imagen diferencia “ImSat1\_E” como una imagen con gran cantidad de pixels alrededor de cero, por lo tanto se procede a desplazar su intensidad de gris añadiéndole un nivel de gris igual a 64.

```
>> figure(2), colormap(gray(256))  
  
>> ImSat1_E = ImSat1 - ImSat1_F;  
  
>> subplot( 1,2,1 ) , image( ImSat1_E )  
  
>> subplot( 1,2,2 ) , image( ImSat1_E + 64 )
```

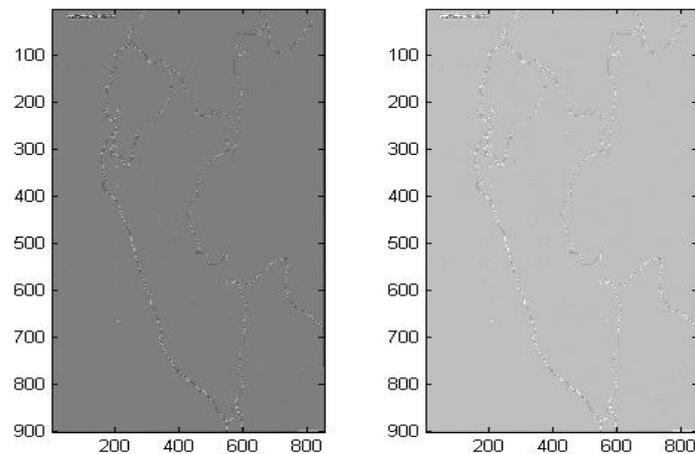


Figura 4.3. a) Imagen diferencia. b) Imagen diferencia con un nivel de gris añadido

De esta forma, las imágenes se encuentran listas para realizar los procesos de codificación con pérdida y sin pérdida. El diagrama de bloques presentado en la figura 4.4 muestra de manera general la manera de cómo se ejecuta las dos técnicas de codificación, para posteriormente aplicar la decodificación.

#### 4.2.2 Obtención de la DCT-2D

Con ayuda del toolbox de Signal Processing del Matlab, o con el código de la DCT-2D implementado y mostrado en el capítulo 2, se procede a obtener dicha transformada sobre la imagen filtrada.

Aquí es necesario aclarar que la DCT-2D no es aplicada directamente sobre toda la imagen completa, sino más bien se aplica sobre bloques de 8x8 pixels de la imagen

filtrada. La forma como es aplicada dicha transformada tiene como finalidad disminuir el esfuerzo computacional que produce la DCT-2D. Por lo tanto, es necesario aplicar un algoritmo basado en la sentencia FOR para aplicar la DCT-2D solamente sobre tal tamaño de imagen.

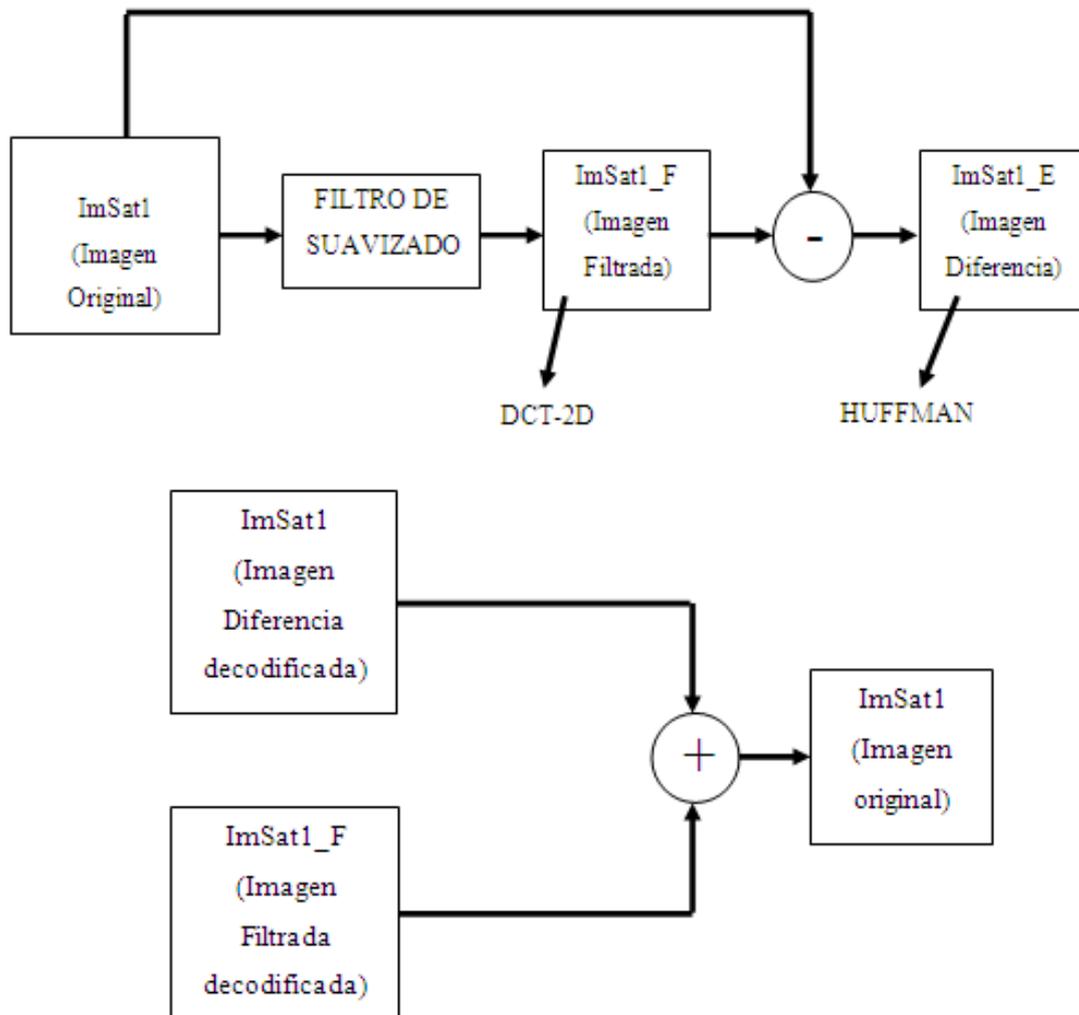


Figura 4.4. a) Diagrama de bloques para la codificación y b) decodificación.

A continuación se muestra el código utilizado para el cálculo de la DCT-2D sobre los bloques de 8x8 pixels. Ver la figura 4.5

```

C:\DCT8x8.m
File Edit View Text Debug Breakpoints Web Window Help
[Icons]
1 - [ m , n ] = size( ImSat1_F_D );
2 - ImSat1_F_DCT = zeros( m , n );
3
4 - for i=1:8:m-7
5 -     for j=1:8:n-7
6 -         B = zeros(8,8);
7 -         A = dct2( ImSat1_F_D(i:i+7,j:j+7) );
8 -         B(1:4,1:4) = A(1:4,1:4);
9 -         A = B;
10 -        ImSat1_F_DCT(i:i+7,j:j+7) = A;
11 -    end
12 - end
13 - figure(1), colormap(gray(256))
14 - subplot( 1,2,1 ),image( ImSat1_F_D )
15 - subplot( 1,2,2 ),image( ImSat1_F_DCT )
16

```

Figura 4.5 Algoritmo de obtención de la DCT-2D sobre bloques de 8x8 pixels

Asimismo, al aplicar el algoritmo de la figura 4.5 sobre la imagen ImSat1\_F\_D que viene utilizándose, se obtiene el resultado en la figura 4.6.

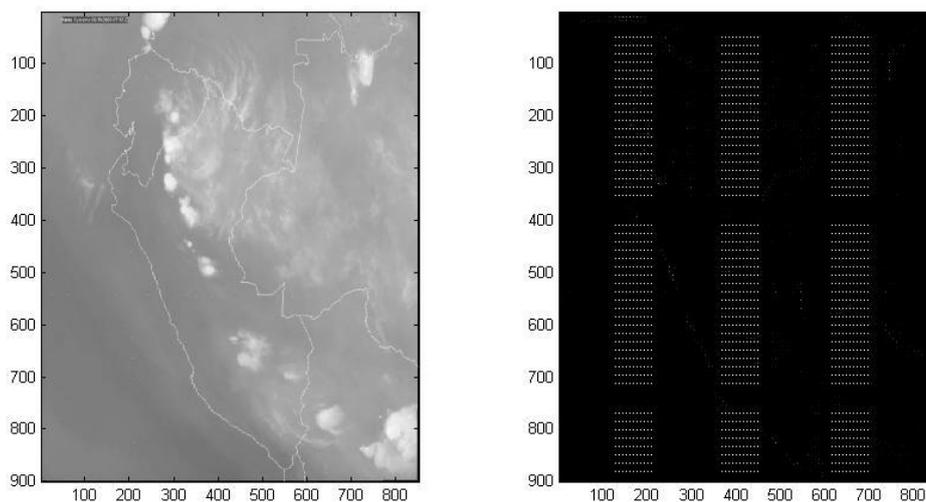


Figura 4.6 a) Imagen original. b) DCT-2D de la imagen original

De la parte b) de la figura 4.6 se observa gran cantidad de zonas negras debido a la abundancia de coeficientes de baja energía, los cuales se sitúan numéricamente cercano

y alrededor de cero. Eso quiere decir que tales coeficientes pueden ser despreciados o codificados adecuadamente utilizando el código de Huffman. Recordando que dicho código codifica con mayor cantidad de bits a aquello que aparece con menor frecuencia, en cuanto codifica con menor cantidad de bits a aquello que se presenta con mayor frecuencia.

Antes de aplicar la DCT-2D sobre la imagen filtrada, “ImSat1\_F”, se realiza un desplazamiento de los niveles de gris. Este desplazamiento equivale a una operación de substracción de cada pixel de la imagen por un factor igual a “ $2^{n-1}$ ”, donde “n” representa al número de bits utilizados en la codificación de la imagen original. Para nuestro caso “n” es igual a 7. Vea la figura 4.7.

```

MATLAB
File Edit View Web Window Help
Current Directory: C:\
>> load ImSat1_F
>> ImSat1_F(1:8,1:8)
ans =
19.1250 25.5000 25.5000 25.5000 25.5000 25.5000 25.5000 25.5000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000
25.5000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0000 34.0625

>> ImSat1_F_D = ImSat1_F - 128;
>> ImSat1_F_D(1:8 , 1:8)
ans =
-108.8750 -102.5000 -102.5000 -102.5000 -102.5000 -102.5000 -102.5000 -102.5000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000
-102.5000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -94.0000 -93.9375

>> |

```

Figura 4.7 Substracción de cada pixel de la imagen filtrada (ImSat1\_F\_D)

Esta operación de substracción tiene como objetivo hacer que aparezca gran cantidad de coeficientes iguales a “0” una vez que se obtenga la DCT-2D. Tal resultado es apreciado en la Figura 4.8

```

>> ImSat1_F_DCT(1:8,1:8)

ans =

-768.7266  -11.4322  -10.7485  -9.6918     0     0     0     0
-11.4322    0.5261    0.4672    0.4460     0     0     0     0
-10.7485    0.4672    0.4668    0.3961     0     0     0     0
-9.6918     0.4460    0.3961    0.3781     0     0     0     0
     0         0         0         0         0     0     0     0
     0         0         0         0         0     0     0     0
     0         0         0         0         0     0     0     0
     0         0         0         0         0     0     0     0
>>

```

Figura 4.8 Coeficientes de la DCT-2D de un bloque de 8x8 de la imagen filtrada y desplazada

### 4.2.3 Normalización y Cuantificación

Obtenido los coeficientes DCT-2D en cada bloque de 8x8 pixels de la Imagen Filtrada y Desplazada, se procede a cuantificar y normalizar el resultado de tal transformada. Esto ayuda a aumentar el número de coeficientes iguales a cero, así como también facilita la codificación o asignación de bits de manera aislada en cada coeficiente obtenido.

Para ello, se procede a dividir cada bloque de 8x8 de los coeficientes de la DCT-2D por el bloque  $Q(u,v)$  [03], el cual representa un estándar en la compresión utilizada por el JPEG. Dicha operación de división simboliza una normalización de cada bloque de 8x8 de la imagen tratada. Una vez realizada la normalización, se continúa con la cuantización o redondeo de cada coeficiente. Esta cuantificación ayuda a la codificación o asignación de bits de cada coeficiente presentado en un formato entero. Por lo tanto, se aplicó la siguiente expresión matemática sobre la imagen filtrada y desplazada<sup>34</sup>.

$$I_q(m,n) = \left\lfloor \frac{I(m,n)}{Q(m,n)} \right\rfloor \quad (4.3)$$

Donde:

$I(m,n)$  : representa la imagen a normalizar. Para nuestro caso es la imagen a la que se ha filtrado, desplazado un nivel de gris y obtenido la DCT-2D.

<sup>34</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

$Q(m,n)$ : representa la matriz de normalización

Barras “|” representan una operación de redondeo de cada coeficiente obtenido después de aplicar la operación de división.

Seguidamente se muestra la matriz de normalización utilizada en el desarrollo de esta tesis, la cual fue obtenida de la misma referencia bibliográfica por recomendación del estándar JPEG<sup>34</sup>.

$$Q(m, n) = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Para continuar con el ejemplo que se viene presentando, se muestra a continuación el resultado de aplicar la ecuación 4.3 sobre la imagen resultante del filtrado, desplazamiento y obtención de la DCT-2D y o, que es lo mismo, la variable  $ImSat1\_F\_DCT$ . Ver la figura 4.9, la cual también muestra el resultado de la normalización y cuantificación sobre el primer bloque de 8x8. Es decir, las primeras 8 filas y primeras 8 columnas.

#### 4.2.4 Ordenamiento de los Coeficientes DCT

Completada la operación de normalización y cuantificación, se procede a ordenar los coeficientes partiendo del hecho de la abundancia de coeficientes igual a cero después de obtener la DCT-2D. Tal ordenamiento de coeficientes está basado en el modelo mostrado en la matriz “ $M_{ord}$ ”, que también es utilizado por el algoritmo de codificación del JPEG. Dicho modelo da origen a una vector de dimensión  $8 \times 8 = 64$  elementos, que siguen una correspondencia en zig-zag<sup>35</sup>.

Seguidamente se muestra la matriz “ $M_{ord}$ ” utilizada en el desarrollo de esta tesis.

---

<sup>35</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

$$M_{ord} = \begin{bmatrix} 0 & 1 & 5 & 6 & 14 & 15 & 27 & 28 \\ 2 & 4 & 7 & 13 & 16 & 26 & 29 & 42 \\ 3 & 8 & 12 & 17 & 25 & 30 & 41 & 43 \\ 9 & 11 & 18 & 24 & 31 & 40 & 44 & 53 \\ 10 & 19 & 23 & 32 & 39 & 45 & 52 & 54 \\ 20 & 22 & 33 & 38 & 46 & 51 & 55 & 60 \\ 21 & 34 & 37 & 47 & 50 & 56 & 59 & 61 \\ 35 & 36 & 48 & 49 & 57 & 58 & 62 & 63 \end{bmatrix}$$

De esta manera, cada bloque de 8x8 de la imagen ImSat1\_F\_DCT se convierte en un vector de 64 elementos cuyos elementos se ordenan en base a la matriz “M<sub>ord</sub>”. El código de Matlab utilizado para tal ordenamiento se muestra seguidamente en la figura 4.10.

```

MATLAB
File Edit View Web Window Help
Current Directory: C:\

>> [ m , n ] = size( ImSat1_F_DCT );
ImSat1_F_DCT_N = zeros( m , n );

Q=[ 16 11 10 16 24 40 51 61 ;
    12 12 14 19 26 58 60 55 ;
    14 13 16 24 40 57 69 56 ;
    14 17 22 29 51 87 80 62 ;
    18 22 37 56 68 109 103 77 ;
    24 35 55 64 81 104 113 92 ;
    49 64 78 87 103 121 120 101 ;
    72 92 95 98 112 100 103 99 ];

for i=1:8:m-7
    for j=1:8:n-7
        B = zeros(8,8);
        A = ImSat1_F_DCT(i:i+7,j:j+7);
        B = round( A ./ Q );
        ImSat1_F_DCT_N(i:i+7,j:j+7) = B;
    end
end
ImSat1_F_DCT(1:8,1:8)
ImSat1_F_DCT_N(1:8,1:8)

ans =

-768.7266  -11.4322  -10.7485  -9.6918     0     0     0     0
-11.4322    0.5261    0.4672    0.4460     0     0     0     0
-10.7485    0.4672    0.4668    0.3961     0     0     0     0
-9.6918     0.4460    0.3961    0.3781     0     0     0     0
     0         0         0         0     0     0     0     0
     0         0         0         0     0     0     0     0
     0         0         0         0     0     0     0     0
     0         0         0         0     0     0     0     0

ans =

-48  -1  -1  -1  0  0  0  0
-1  0  0  0  0  0  0  0
-1  0  0  0  0  0  0  0
-1  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0
 0  0  0  0  0  0  0  0

>>

```

Figura 4.9 Normalización y Cuantización de los coeficientes de la DCT-2D.

```

C:\ordenamiento.m
File Edit View Text Debug Breakpoints Web Window Help
Stack: Base
1 - [ m , n ] = size( ImSat1_F_DCT );
2 - ImSat1_F_DCT_N = zeros( m , n );
3 - F = m*n/64;
4 - ImSat1_F_DCT_N_0 = zeros(64,F);
5
6 - M=[ 0 1 5 6 14 15 27 28 ;
7     2 4 7 13 16 26 29 42 ;
8     3 8 12 17 25 30 41 43 ;
9     9 11 18 24 31 40 44 53 ;
10    10 19 23 32 39 45 52 54 ;
11    20 22 33 38 46 51 55 60 ;
12    21 34 37 47 50 56 59 61 ;
13    35 36 48 49 57 58 62 63 ];
14
15 - k = 1;
16 - for i=1:8:m-7
17 -     for j=1:8:n-7
18 -         A = ImSat1_F_DCT(i:i+7,j:j+7);
19 -         B = [ A(1,1) A(1,2) A(2,1) A(3,1) A(2,2) A(1,3) A(1,4) A(2,3) ;
20             A(3,2) A(4,1) A(5,1) A(4,2) A(3,3) A(2,4) A(1,5) A(1,6) ;
21             A(2,5) A(3,4) A(4,3) A(5,2) A(6,1) A(7,1) A(6,2) A(5,3) ;
22             A(4,4) A(3,5) A(2,6) A(1,7) A(1,8) A(2,7) A(3,6) A(4,5) ;
23             A(5,4) A(6,3) A(7,2) A(8,1) A(8,2) A(7,3) A(6,4) A(5,5) ;
24             A(4,6) A(3,7) A(2,8) A(3,8) A(4,7) A(5,6) A(6,5) A(7,4) ;
25             A(8,3) A(8,4) A(7,5) A(6,6) A(5,7) A(4,8) A(5,8) A(6,7) ;
26             A(7,6) A(8,5) A(8,6) A(7,7) A(6,8) A(7,8) A(8,7) A(8,8) ];
27 -         C = [ B(1,:) B(2,:) B(3,:) B(4,:) B(5,:) B(6,:) B(7,:) B(8,:) ];
28 -         ImSat1_F_DCT_N_0(k,1:64) = C;
29 -         k = k + 1;
30 -     end
31 - end
32 - ImSat1_F_DCT_N_0(1:8,1:8)
33

```

Figura 4.10 Ordenamiento de coeficientes de la DCT-2D

Obtenido el ordenamiento de los coeficientes de la variable `ImSat1_F_DCT_N`, se procede a ignorar todos aquellos coeficientes de la posición 10 hasta la 63 de la matriz de ordenamiento “`Mord`”. Este conjunto representa el 84.375% del total de los 64 coeficientes resultantes. El 15.625% de coeficientes restantes, son codificados de forma directa asignando siete (07) bits es decir una cantidad mayor de bits al coeficiente de la posición 0 de la matriz de ordenamiento “`Mord`”. Luego, a los otros nueve coeficientes se le asigna tres (03) bits. Tener presente que para dichas asignaciones se ha verificado cada uno de los bloques de 8x8 obtenidos. Asimismo, tales asignaciones de bits tienen incluido el bit de signo correspondiente a “0” cuando se trata de una cantidad positiva, mientras que un bit igual a “1” cuando se trata de una cantidad negativa. Por ejemplo, para el grupo de coeficientes obtenidos en las primeras 8 filas y primeras 8 columnas de la variable `ImSat1_F_DCT_N` correspondiente a la figura 4.9, se procede a codificar de forma directa teniendo presente lo anteriormente descrito.

-1 → 11

0 → 00

Por lo tanto, el número de bits utilizados para codificar esos primeros 10 coeficientes se ha utilizado un total de:

$$7*1 + 3*9 = 34 \text{ bits}$$

### 4.3 Codificación con Huffman de la Imagen Diferencia

Con lo anteriormente descrito se logra la codificación de la imagen filtrada, la cual a su vez fue normalizada y cuantificada. Ahora se procede a codificar la imagen diferencia. Esta imagen diferencia se caracteriza por presentar gran cantidad de coeficientes alrededor de cero. Para ello, se ha procedido a obtener el histograma de la imagen diferencia para observar la concentración de los niveles de gris existentes alrededor del cero. La figura 4.11 muestra el resultado del histograma y en donde 0.5 corresponde al nivel de gris 127, 0 corresponde al color negro y el valor de 1 representa el color blanco. Entonces se puede observar gran concentración en el color negro.

```
>> load ImSat1_E
```

```
>> colormap(gray(256))
```

```
>> image( ImSat1_E )
```

```
>> imhist( ImSat1_E/255 )
```

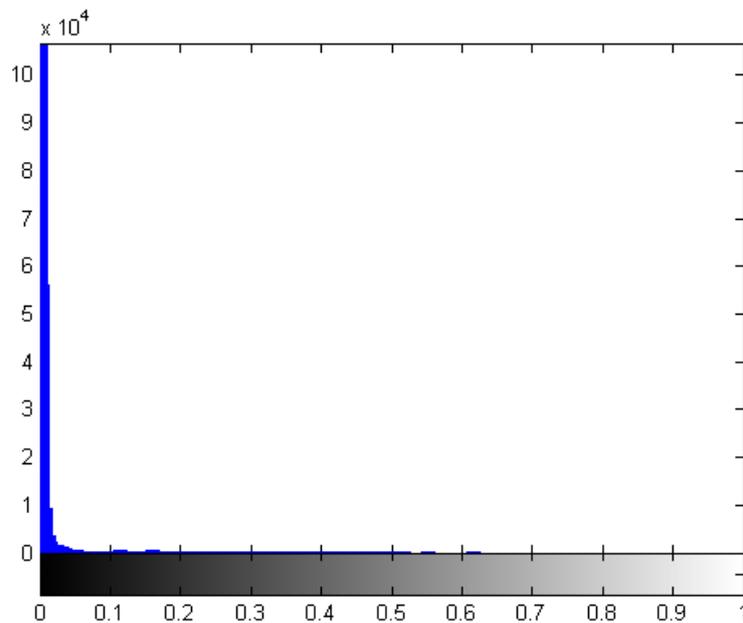


Figura 4.11 Histograma de la Imagen Diferencia ImSat1\_E

Para la codificación por Huffman se procede a utilizar la metodología empleada en el ejemplo práctico del ítem 3.3, que anteriormente fue explicado.

#### 4.4 Técnica de Decodificación

Realizada la codificación, se procede a la etapa de recuperación o decodificación para traer de retorno la imagen codificada. Para la realización de la decodificación, se procede primero a decodificar la imagen filtrada, cuantificada y normalizada basada en la DCT-2D; mientras que para decodificar la imagen diferencia, se procede a la asignación directa de los códigos basados en el árbol de Huffman.

Obtenida la decodificación de los coeficientes de la imagen diferencia ImSat1\_E y de la variable ImSat1\_F\_DCT\_N\_O, se procede a sumar ambos resultados dando origen a la imagen recuperada después de la codificación.

## 4.5 Ejemplo de Codificación y Decodificación

Este capítulo finaliza mostrando un ejemplo completo de la aplicación de la técnica de codificación planteada para esta tesis. El ejemplo es presentado con la imagen patrón satelital que ha venido utilizándose.

El tema de codificación es presentado a través de seis pasos, los cuales son explicados detalladamente uno tras otro.

Primer paso.- Se procede a utilizar un filtro Smooth sobre la imagen satelital digital que se desea codificar. Previamente, la imagen original es aumentada en tamaño utilizando coeficientes iguales a cero.

```
>>ImSat1_Z=[zeros(1,n+2);zeros(m,1) ImSat1 zeros(m,1);zeros(1,n+2)];
```

```
>> h = [ 1 2 1 ; 2 4 2 ; 1 2 1 ] / 16;
```

```
>> A = conv2( h , ImSat1_Z , 'full' );
```

```
>> ImSat1_F = A( 2:m-1 , 2:n-1 );
```

Segundo paso.- Se procede a substraer de la imagen original la imagen filtrada. Esto da como resultado una nueva imagen de error denominada ImSat1\_D.

```
>> ImSat1_E = ImSat1 - ImSat1_F;
```

Tercer paso.- Se procede a codificar la imagen filtrada, la cual previamente es desplazada en un nivel de intensidad de gris, luego normalizada, seguido de una cuantificación, luego por la extracción de la DCT-2D, el ordenamiento de sus coeficientes, y el truncado y asignación de bits a 1/8 de los coeficientes de cada bloque de 8x8.

```
>> ImSat1_F = ImSat1_F - 128;
```

```
>> load Q
```

```
>> ImSat1_F_N = round( ImSat1_F ./ Q );
```

```
>> ImSat1_F_N_DCT = tdc2( ImSat1_F_N );    % función que corresponde
```

```
% a la DCT-2D
```

```
>> ImSat1_F_N_DCT = ordena( ImSat1_F_N_DCT );    % ordena los coeficientes
```

```
>> ImSat1_F_N_DCT = trunca( ImSat1_F_N_DCT );
```

```
>> ImSat1_F_N_DCT_COD = codificaIF( ImSat1_F_N_DCT );
```

Cuarto paso. Se procede a codificar la imagen diferencia a partir del código de Huffman que fue tratado en el capítulo anterior.

```
>> ImSat1_E_COD = codificaDI( ImSat1_E );
```

Quinto paso.- Se procede a decodificar la imagen filtrada aplicando la DCT-2D inversa seguido de la desnormalización y culminando con un tamaño de imagen original.

```
>> ImSat1_F_N_DCT_DEC = decodificaIF( ImSat1_F_N_DCT_COD );
```

```
>> ImSat1_F_DCT_DEN = denormaliza( ImSat1_F_N_DCT_DEC );
```

```
>> ImSat1_F_IDCT = tdcinversa( ImSat1_F_DCT_DEN );
```

Sexto paso.- Se procede a decodificar la imagen diferencia a partir del árbol de Huffman, el cual permite retornar cada uno de los coeficientes de dicha imagen. Finalmente, se procede a sumar las imágenes decodificadas pertenecientes a la imagen

filtrada y la imagen diferencia. Dicha suma dará como origen la recuperación de la imagen inicial.

```
>> ImSat1_E_DEC = decodificaDI( ImSat1_E_COD );
```

```
>> ImSat1_NUEVO = ImSat1_F_IDCT + 128;
```

# CAPÍTULO 5

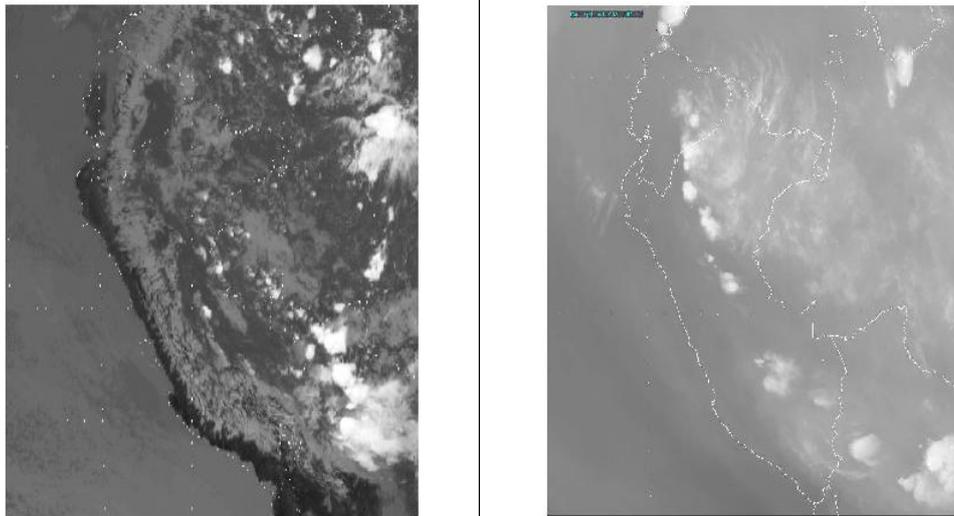
## PRUEBAS, RESULTADOS Y SIMULACIONES

Finalmente, después de haber desarrollado la técnica propuesta en esta tesis; se procede a la realización de las pruebas y la obtención de los resultados al trabajar con un grupo de imágenes satelitales obtenidas de algunos sites de Internet. De esta manera, en este capítulo se muestra el resultado de las simulaciones basado en algoritmos implementados bajo el entorno del software Matlab.

### **5.1 Imágenes satelitales utilizadas**

Se utilizó un grupo reducido de imágenes satelitales para la realización de las pruebas de codificación. Ese número fue igual a seis imágenes y a continuación se muestran dos de ellas en donde es posible apreciar la abundancia de pixeles con niveles de gris prácticamente en 128. Con fines de extracción de información o aclaración de zonas, es probable manipular el histograma de tales imágenes y conseguir mejores vistas que faciliten analizar y posteriormente otorgar conclusiones, sobre todo cuando se trata de predicciones climáticas o fenómenos ambientales.

La figura 5.1 muestra dos de las imágenes utilizadas en la codificación por la técnica de codificación planteada. Tales imágenes fueron codificadas con 8 bpp (bits por pixel) desde el punto de vista de PCM (modulación por código de pulsos), y con una resolución de 896 x 848 pixels. Ya con la técnica planteada se logra codificar cada pixel por un número menor a un bit, hablando del punto de vista global en la imagen.



(b)

Figura 5.1 Imágenes utilizadas en la codificación con la técnica planteada

## 5.2 Error numérico y visual

La codificación es el proceso que permite convertir una matriz con pixels representados con 8 bits en un conjunto de coeficientes listos para ser almacenados y/o transmitidos. Por tal razón, la codificación deja de ser PCM (modulación por codificación de pulso, es decir 8 bits por pixel) y pasa a ser 1 bpp. Que fue el grado de compresión alcanzado en este trabajo de tesis.

El grado o razón de compresión es un tema que depende mucho de la calidad de la información que se desea codificar. Por tratarse de imágenes satelitales, la cantidad de pixels es elevada, pero así mismo la información contenida es de alta importancia porque otorga diferentes tipos de ayudas sobre todo en la predicción del clima. Es por tal motivo que no se puede elevar el grado de compresión sacrificando calidad en los pixels debido a una posible alteración en la información contenida en la imagen. De la misma forma y en un caso más delicado, sucede con las imágenes médicas en donde el grado de compresión es reducido porque cabe la posibilidad de alterar grupos importantes o característicos de pixels que podrían ser pieza clave para el diagnóstico de diferentes enfermedades existentes en la actualidad.

Para comprobar el error numérico existente entre la imagen original y la reconstruida después de aplicar la técnica de codificación y decodificación planteada, se ha utilizado la relación señal-ruido de pico (PSNR) la cual es definida en la ecuación 5.1<sup>36</sup>

$$\text{PSNR} = 10 * \log_{10} \left[ \frac{255^2}{N * N \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (I_{ij} - I^R_{ij})^2} \right] \quad 5.1$$

Donde:

255 : es el valor pico de la señal (nivel mayor de intensidad de gris)

I : la imagen original a comprimir

I<sup>R</sup> : la imagen reconstruida o decodificada. Ambas imágenes con tamaño de N x N pixels.

Por ejemplo, para el caso de la imagen (a) y (b) de la figura 5.1, el error numérico encontrado después de utilizar la relación señal ruido pico fue de:

Imagen (a) : 34.2 dB

Imagen (b) : 34.5 dB

Para comprobar el error visual existente, se procedió a realizar una operación de substracción directa entre la imagen original y la imagen reconstruida o decodificada. Pero, debido a que la diferencia es mínima entre ambas imágenes, se procedió a sumar un nivel de gris igual a 128 para que de esa forma se pueda apreciar mejor el error existente entre ambas imágenes con la misma resolución y cantidad de bits por pixel.

A continuación, las figuras 5.2 y 5.3 muestran una comparación entre las imágenes originales y las decodificadas, así como también la imagen error producto de la diferencia entre estas dos imágenes citadas. Sin olvidar que existe una manipulación del nivel de gris de la imagen error con fines de poder apreciar mejor los errores encontrados.

---

<sup>36</sup> Visión por Computador Imágenes Digitales y Aplicaciones. Gonzalo Pajares. 2001.

Se puede observar que el error es mínimo y ello se debe a la naturaleza de estas imágenes satelitales, pues cuentan con abundante grupo de pixels con intensidades de gris similares. Para corroborar dicha afirmación, se procede a mostrar en la figura 5.4 y 5.5 los histogramas correspondientes a dichas imágenes, con lo cual se corrobora lo anteriormente comentado.

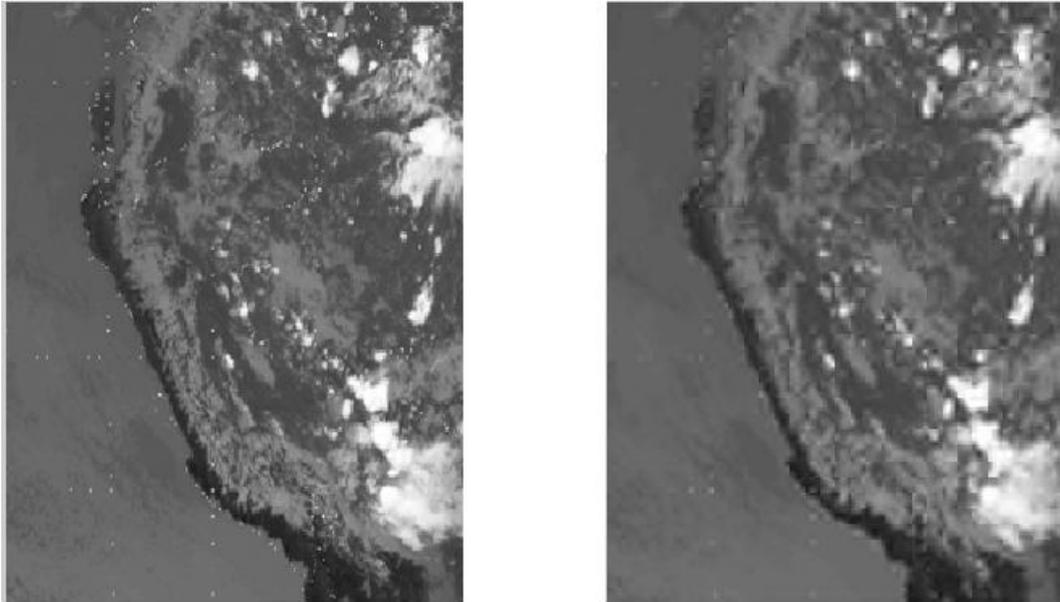


Figura 5.2 Imagen Original e imagen decodificada correspondiente a la imagen (a) de la figura 5.1

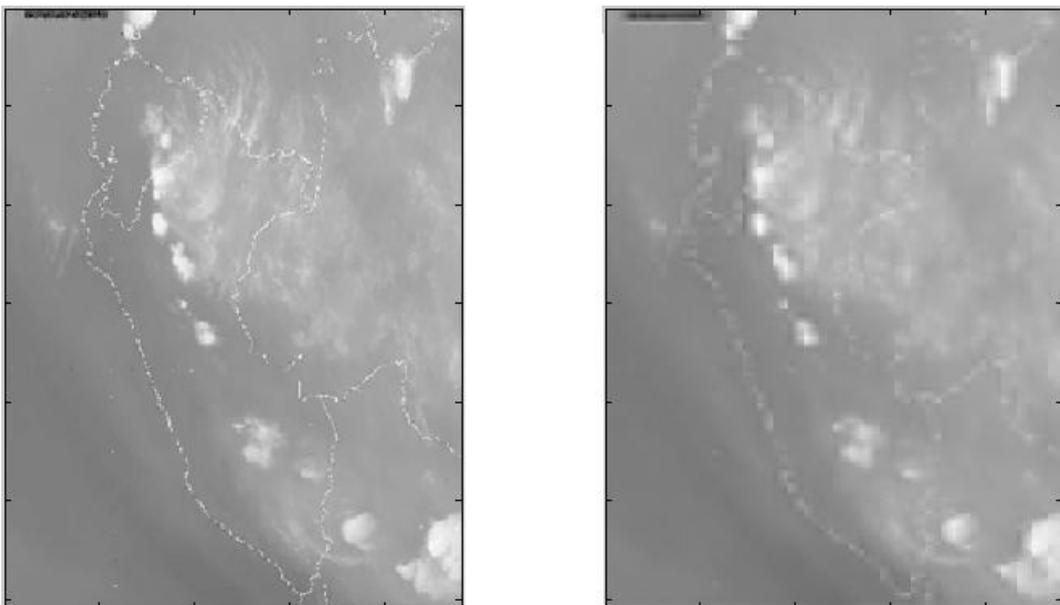


Figura 5.3 Imagen Original e imagen decodificada correspondiente a la imagen (b) de la figura 5.1

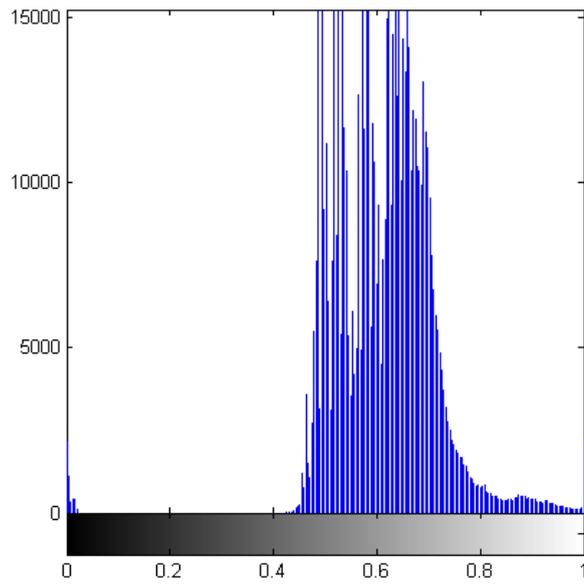


Figura 5.4 Histograma de la imagen (a) de la figura 5.1

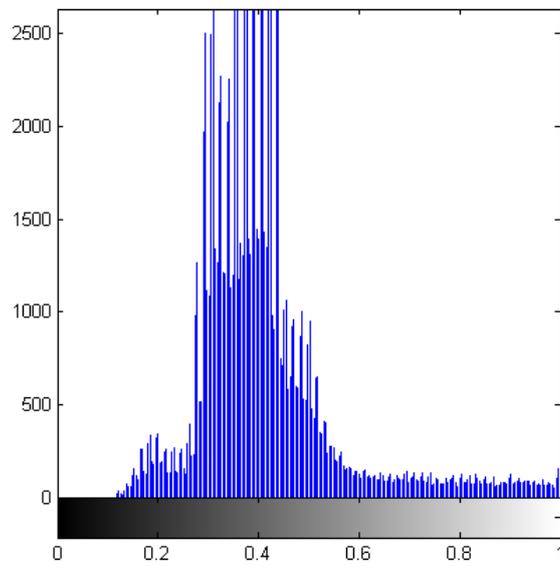


Figura 5.5 Histograma de la imagen (a) de la figura 5.2

# CONCLUSIONES

Las conclusiones relacionadas al desarrollo de este trabajo de tesis, son presentadas consecutivamente de forma enumerada.

1. Las imágenes satelitales, en su mayoría, están representadas por matrices con abundancia de píxeles en intensidades de gris. Esto facilita alcanzar altos grados de compresión.
2. Tal distribución casi uniforme de intensidades de gris facilita alcanzar altas tasas de compresión, y estas se diferencian notablemente de a las imágenes con información de alta frecuencia. Como se sabe, la característica de este conjunto de imagen aumenta la información en baja frecuencia, lo que posibilita conseguir una mayor compresión.

El pre procesamiento de la imagen a codificar ayuda a incrementar la cantidad de coeficientes iguales o cercanos a cero. Ello se debe a que la imagen es suavizada con el filtro de Smooth. Tal filtro tiene la propiedad de suavizar la imagen, es decir disminuir algunos picos en cuanto a intensidades de gris existentes en la imagen.

Esto claramente se comprobó obteniendo el histograma de dicha imagen. El histograma mostró como está representada la distribución de los diferentes niveles o intensidades de gris en la imagen a codificar.

Aprovechando esta gran concentración de intensidades de gris cercanos al cero, pueden nacer nuevas técnicas encargadas de realizar manipulaciones del histograma con la finalidad de llegar a comprimir en mayor grado tales imágenes diferencia.

3. Los grados de compresión alcanzados dependerá mucho de la diversidad de intensidades de gris existentes en la imagen satelital a codificar. Esto quiere decir que si existe alta frecuencia, la imagen diferencia (ImSat1\_E) contendrá mayor información y por lo tanto su codificación por Huffman alcanzará bajas tasas de compresión.

En resumen, una codificación con pérdidas que es la que fue desarrollada por la Transformada Discreta del Coseno en 2D, logra mayor compresión en la imagen sacrificando calidad en la misma por la aparición de errores casi imperceptibles.

Por otro lado, una codificación sin pérdidas tal como la desarrollada por Huffman, logra menores o bajas tasas de compresión pero con la ventaja de recuperar al 100% la información codificada.

Es así que en este trabajo de tesis, se combina ambas técnicas para equilibrar el grado de compresión con la calidad de imagen codificada.

4. Si bien es cierto que el estándar JPEG alcanza mayores tasas de compresión, la técnica propuesta se aproxima a tal tasa de compresión pero con la ventaja de mantener una buena calidad en la imagen codificada. Ello se debe principalmente a la separación que se realiza de la imagen en dos grupos de pixeles diferentes. Uno correspondiente a una imagen con abundante información de pasa bajos, mientras que otra con abundante información de pasa altos.

# WEBGRAFÍA

- [1] [http://www.es.wikipedia.org/wiki/Universal\\_Serial\\_Bus](http://www.es.wikipedia.org/wiki/Universal_Serial_Bus)
- [2] <http://www.satimagingcorp.es/gallery.html>
- [3] [http://es.wikipedia.org/wiki/Sat%C3%A9lite\\_artificial](http://es.wikipedia.org/wiki/Sat%C3%A9lite_artificial)
- [4] [http://www.inegi.gob.mx/inegi/contenidos/espanol/ciberhabitat/medios/satelites/Artificiales/que\\_es.htm](http://www.inegi.gob.mx/inegi/contenidos/espanol/ciberhabitat/medios/satelites/Artificiales/que_es.htm)
- [5] [http://www.proyectosalohogar.com/Enciclopedia/Astronautica/NE\\_Astronautica.htm](http://www.proyectosalohogar.com/Enciclopedia/Astronautica/NE_Astronautica.htm)
- [6] [http://www.astronomos.org/articulas/Polaris/2004/43-SATELITES\\_ARTIFICIALES.pdf](http://www.astronomos.org/articulas/Polaris/2004/43-SATELITES_ARTIFICIALES.pdf)
- [7] <http://www.senamhi.gob.pe/?p=1003>
- [8] [http://www.srgis.cl/pdf/guia\\_basica\\_imagenes\\_satelitales.pdf](http://www.srgis.cl/pdf/guia_basica_imagenes_satelitales.pdf)

# BIBLIOGRAFÍA

- [01] ANIL K. Maini & VARSHA Agrawal. “Satellite Techonology Principles and Applications”. Editorial Wiley. Segunda edición. U.S.A. 2010. Páginas 704.
- [02] PAJARES Gonzalo y DE LA CRUZ Jesús. “Visión por Computador. Imágenes digitales y aplicaciones”. Editorial Alfaomega. Segunda edición. U.S.A. 2008. Páginas 740.
- [03] GONZALEZ Rafael, WOODS Richard. “Tratamiento digital de imágenes”. Editorial Addison-Wesley Iberoamericana S.A. Primera edición. U.S.A. 1996. Página 757.
- [04] ALBA José y MARTIN Fernando. “Material didáctico: Transformadas de Imágenes“. Universidad Carlos III de Madrid. Publicación año

