

**Universidad Central “Marta Abreu” de Las Villas**

**Facultad de Ingeniería Eléctrica**

**Centro de Estudios de Electrónica y Tecnologías de la  
Información.**

**CEETI**



## **TRABAJO DE DIPLOMA**

**Estimación de TC máxima para imágenes de TAC  
usando el códec JPEG2000.**

**Autor: Roger R. Góngora Martínez.**

**Tutor: Ing. Iroel Miranda Castañeda.**

**Santa Clara**

**2013**

**"Año 55 de la Revolución"**

**Universidad Central “Marta Abreu” de Las Villas**

**Facultad de Ingeniería Eléctrica**

**Centro de Estudios de Electrónica y Tecnologías de la  
Información.**

**CEETI**



## **TRABAJO DE DIPLOMA**

**Estimación de TC máxima para imágenes de TAC  
usando el códec JPEG2000.**

**Autor: Roger R. Góngora Martínez.**

[gongora@uclv.edu.cu](mailto:gongora@uclv.edu.cu)

**Tutor: Ing. Iroel Miranda Castañeda.**

**Santa Clara**

**2013**

**"Año 55 de la Revolución"**



Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de estudios de la especialidad de Ingeniería en Biomédica, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

---

Firma del Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

---

Firma del Tutor

---

Firma del Jefe de Departamento  
donde se defiende el trabajo

---

Firma del Responsable de  
Información Científico-Técnica

## **PENSAMIENTO**

*“No estudio por saber más, sino por ignorar menos”*

*Sor Juana Inés de la Cruz.*

## **DEDICATORIA**

A mi familia.

## **AGRADECIMIENTOS**

Agradezco todo el apoyo que me ha brindado mi familia, en especial a mis padres y hermano, a mi novia Katerina, a mi tutor el Ing. Iroel Miranda Castañeda, el cual me guió durante la realización de este trabajo.

A todos mis compañeros de aula.

Al personal del CEETI por su ayuda incondicional.

A todas aquellas personas que de una forma u otra ayudaron a la realización de este trabajo.

## TAREA TÉCNICA

- Selección de un grupo de imágenes representativo con lesiones adecuadas para realizar el experimento propuesto.
- Compresión del grupo de imágenes en un intervalo de Tasa de Compresión (TC) desde donde no se aprecien cambios significativos (poca compresión), hasta donde la calidad de la imagen se degrade significativamente perdiéndose información de detalle en las estructuras que la componen (mucha compresión).
- Selección de un grupo de medidas objetivas que sean representativas para el comportamiento de la calidad de la imagen en el intervalo seleccionado y cálculo del comportamiento de las medidas de calidad objetivas para el intervalo de TC estudiado.
- Diseño del experimento para evaluar mediante observadores humanos el desempeño de los algoritmos de segmentación en función de la TC.
- Validación de las observaciones realizadas por los observadores de acuerdo a los criterios de relación intra- e inter-observador.
- Estimación de una TC de compresión a partir de los criterios emitidos durante los experimentos objetivos y subjetivos.

---

Firma del Autor

---

Firma del Tutor

## **RESUMEN**

La Tomografía Axial Computarizada (TAC) es una potente herramienta de diagnóstico con la que se obtienen imágenes de alta resolución de cualquier parte del cuerpo. Una de las desventajas técnicas de los estudios por imagen se asocia a los grandes volúmenes de datos que se generan. Una de las respuestas de la comunidad científica a esta limitación, han sido los algoritmos de compresión con pérdidas, pero cuando se utiliza un algoritmo de compresión, la calidad de la imagen puede degradarse a un nivel tal que puede afectar el resultado final del proceso para el cual será utilizada, ya sea diagnóstico médico, segmentación o clasificación automática. Con el propósito de garantizar un desempeño adecuado del códec JPEG2000, se hace necesario estimar una tasa de compresión (TC) óptima para esta modalidad de imagen. Por este motivo se realiza esta investigación, en la cual se estima por diferentes medidas objetivas los parámetros óptimos para este tipo de estudio. Como resultado se obtuvo que la TC final aproximada está alrededor de 21:1, donde la calidad de las imágenes no se afecta y aún es útil para diagnóstico.



## TABLA DE CONTENIDOS

PENSAMIENTO .....	i
DEDICATORIA .....	ii
AGRADECIMIENTOS.....	iii
TAREA TÉCNICA.....	iv
RESUMEN .....	v
INTRODUCCIÓN.....	1
CAPÍTULO 1. REVISIÓN BIBLIOGRÁFICA .....	3
1.1 Las imágenes médicas en formato digital. ....	3
1.1.1 Las imágenes de Tomografía Axial Computarizada (TAC). ....	4
1.1.2 Los algoritmos de compresión. ....	4
1.1.2.1 Los algoritmos de compresión con pérdidas. El códec JPEG2000.....	5
1.2 Calidad de Imagen.....	7
1.2.1 Medidas de calidad objetivas bivariadas.....	7
1.2.2 Medidas de calidad subjetivas.....	12
CONCLUSIONES DEL CAPÍTULO.....	13
CAPÍTULO 2. MATERIALES Y MÉTODOS. ....	14
2.1 Selección de las imágenes de prueba. ....	14
2.2 Compresión de las imágenes de prueba. ....	15

2.3	Cálculo de las medidas objetivas. ....	15
2.4	Cálculo de las medidas subjetivas. ....	16
CONCLUSIONES DEL CAPÍTULO. ....		20
CAPÍTULO 3. RESULTADOS Y DISCUSIÓN. ....		21
3.1	Resultados de las medidas objetivas. ....	22
3.2	Resultados de los experimentos de segmentación. ....	23
3.3	Estimación aproximada de TC máxima por los observadores. ....	24
3.4	Validación de las observaciones realizadas. ....	25
CONCLUSIONES DEL CAPÍTULO. ....		27
CONCLUSIONES. ....		278
REFERENCIAS BIBLIOGRÁFICAS. ....		29
ANEXOS. ....		31

## INTRODUCCIÓN

La Tomografía Axial Computarizada (TAC) tiene sus inicios a finales de la década de los años setenta. Esta poderosa herramienta es muy útil para el estudio de extensión de los cánceres, como el cáncer de mama, cáncer de pulmón, cáncer de próstata, en especial en la zona craneana o la detección de cualquier cáncer en la zona nasal. Entre las ventajas de la TC, se encuentra que es un examen rápido de realizar y que ofrece nitidez de imágenes que todavía no se han superado con la Resonancia Magnética Nuclear (RMN), como es la visualización de ganglios, hueso, etc. Entre sus inconvenientes se cita que la mayoría de las veces es necesario el uso de contraste intravenoso y que al utilizar rayos X, se reciben dosis de radiación ionizante, que en ocasiones no son despreciables. Por ejemplo en una TC abdominal, se puede recibir la radiación de más de 500 radiografías de tórax, el equivalente de radiación natural de más de cinco años. Otra característica de este tipo de estudios es la gran cantidad de datos que genera, por lo que se dificulta su transmisión y almacenamiento, lo que lo convierte en una desventaja técnica. [9], [10]

Para resolver este problema, se realiza una serie de procedimientos, que incluyen la compresión mediante el códec JPEG2000, el cual reduce significativamente el espacio necesario para almacenar el estudio del paciente. El códec utilizado hace uso de la transformada Wavelet y se adopta por el estándar DICOM que es el protocolo que regula el intercambio de datos e imágenes en la medicina. Cuando se usa este tipo de algoritmos de compresión con pérdidas, al elevar demasiado la TC, se corre el riesgo de perder información diagnóstica. [16], [2], [12]

En la actualidad no existe un estándar a seguir para realizar un procesamiento de la imagen satisfactorio, por lo cual se sigue indagando en este códec para hacer un uso óptimo del mismo.

Altas TC pueden introducir considerables distorsiones en la imagen, particularmente en los bordes, lo que conlleva, por ejemplo, a una segmentación errónea. En este sentido es de gran importancia tener un criterio objetivo para decidir un máximo nivel de compresión, conservando la calidad adecuada de la imagen, la que permita realizar un diagnóstico eficaz, es decir, que se pueda obtener una cantidad de información de la misma que sea precisa, ya sea mediante la observación de un especialista o un algoritmo matemático. [17], [19]

Es por ello que se traza como objetivo de la investigación: estimar la máxima TC a utilizar en imágenes de TAC usando el códec JPEG2000.

Para dar cumplimiento a dicho objetivo se plantean como objetivos específicos:

- Crear una Base de Datos con las imágenes compactadas en el intervalo de TC de estudio.
- Realizar un análisis del comportamiento de las medidas de calidad de tipo objetivas para la elección del intervalo de TC en el que se realice el experimento.
- Estudiar el desempeño de los algoritmos de segmentación en función de la TC.
- Hacer un análisis del comportamiento de las medidas subjetivas para elegir la máxima TC.

## CAPÍTULO 1. REVISIÓN BIBLIOGRÁFICA

### 1.1 Las imágenes médicas en formato digital.

Una imagen digital es una representación bidimensional de una imagen a partir de una matriz numérica. Dependiendo de si la resolución de la imagen es estática o dinámica, puede tratarse de una imagen matricial (mapa de bits) o de un gráfico vectorial. El mapa de bits es el formato más utilizado en la informática.

La imagen médica se define como la representación de un sistema de órganos, tejidos o zonas del paciente extraídos mediante diferentes métodos con el fin de diagnosticar o evaluar la condición física del mismo. Los factores determinantes en este tipo de imágenes son la resolución y el contraste. [1]

Una imagen puede considerarse como una función bidimensional:

$$I=f(a,b); \tag{1}$$

donde  $I$  es el valor de intensidad en las coordenadas  $(a, b)$  de la matriz que representa la imagen. Una imagen digital es la representación de un objeto real como un conjunto finito de valores discretos que se obtiene cuando se cuantifican las coordenadas espaciales y la intensidad. A estos valores se les denomina *píxeles*, y son el menor elemento en que se descompone una imagen.

Las imágenes digitales están estrechamente relacionadas con el avance de la tecnología y las comunicaciones, poseen la gran ventaja de no perder calidad con el paso del tiempo y

pueden ser tratadas por diversos programas con el fin de mejorar su calidad o de proporcionar más información. [5], [6]

### 1.1.1 Las imágenes de Tomografía Axial Computarizada (TAC).

Para la realización de esta investigación se usa una muestra de imágenes de TAC. Las imágenes de TAC se obtienen mediante un procedimiento radiológico que mide indirectamente la atenuación o debilitamiento de los Rayos X en numerosos puntos o posiciones alrededor del paciente.

En la TAC el tubo de Rayos X rota alrededor del paciente completando una rotación de  $360^\circ$ . De esta forma se obtienen cortes transversales del paciente lo cual conforma una imagen 3D de cada corte realizado, como se muestra en la Fig.1. En la Fig.2 se muestra la parte exterior de un equipo convencional de TAC.

Se decide realizar el trabajo con este tipo de imágenes debido a la gran capacidad de memoria que ocupa un estudio completo de TAC, además de ser más práctica a la hora de evaluar a un paciente del que se desconoce su historia clínica.

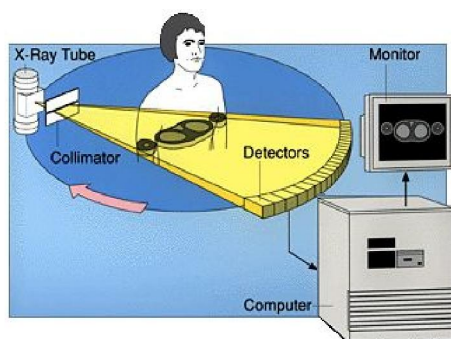


Fig.1

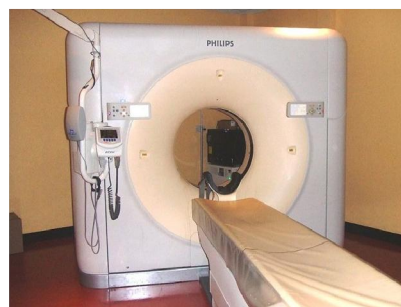


Fig.2

### 1.1.2 Los algoritmos de compresión.

Debido al gran volumen de espacio que se requiere para realizar estudios completos de TAC se hace necesario encontrar una forma de reducirlo. La tecnología va en avance y los

nuevos métodos permiten un increíble aumento de la calidad de las imágenes y la reducción de efectos secundarios que recaen sobre el paciente; de ahí que se hace indispensable la constante evolución de las diferentes técnicas de compresión que existen y la implementación entre ellas. La compresión no es más que la eliminación de información redundante en los datos y codificar la restante mediante algoritmos matemáticos.

La Tasa de Compresión (TC) puede ser calculada mediante la siguiente fórmula:

$$TC = T_{inicial} / T_{final} \quad (2)$$

donde  $T_{inicial}$  es el tamaño inicial del archivo sin compresión y  $T_{final}$  es el archivo comprimido.

Los algoritmos de compresión se pueden clasificar en dos grupos:

- Algoritmos de compresión sin pérdidas.
- Algoritmos de compresión con pérdidas.

La gran diferencia entre estos dos grupos es la TC que se puede lograr, generalmente los algoritmos de compresión con pérdidas poseen mayor TC que los algoritmos de compresión sin pérdidas. [6]

#### **1.1.2.1 Los algoritmos de compresión con pérdidas. El códec JPEG2000.**

Con el paso de los años los algoritmos de compresión con pérdidas han sido de gran utilidad para los investigadores debido a su alta TC. En la actualidad se trabaja con mucha frecuencia con una nueva versión del códec JPEG, el JPEG2000, el cual no es más que una mejora de su antecesor y es usado como estándar por *Digital Imaging Communications in*

*Medicine* (DICOM)[15],[14]. Entre las ventajas que nos ofrece este códec podemos encontrar:

- Presenta hasta un 60% mayor de compresión que su antecesor.
- Robustez frente a los errores de bits.
- Posibilidad de definir regiones de interés dentro de la imagen.
- Comprime y descomprime imágenes de distintos rangos dinámicos.
- Transmisión progresiva mediante resolución y precisión incremental.

Este códec está compuesto por tres bloques de procesamiento. En el primero ocurre la descomposición de la imagen en baldosas o bloques rectangulares, permitiendo el ahorro de memoria y la decodificación de partes específicas de la imagen. En este bloque también ocurre el traslado del nivel DC de cada bloque y la transformación de componentes para decorrelacionar la información de color.

En el segundo bloque se efectúa la transformación de *Wavelet*, la cual puede ser reversible o irreversible, se procede a la descomposición de cada bloque en distintos niveles y posteriormente ocurre la cuantificación. Para terminar este bloque finalmente se lleva a cabo la codificación de entropía, la cual es obtenida mediante un sistema que comprime símbolos binarios en un modelo de probabilidad adaptativa.

Para finalizar con la estructura del códec, cada sub-banda es dividida en bloques rectangulares que son espacialmente consistentes. Para cada uno de estos bloques se genera una cadena de bits, dicha cadena de bits comprimida de cada bloque de código, consta del cuerpo de un paquete, un conjunto de paquetes comprenden una capa. Un paquete es interpretado como un incremento de la calidad para un nivel de resolución en una localización espacial, mientras que una capa es interpretada como un incremento de la calidad para la resolución de la imagen completa.



Como característica adicional este códec cuenta con un algoritmo para controlar la TC, que es desglosado en etapas para así lograr la TC deseada, minimizando los posibles errores de la imagen reconstruida.

## **1.2 Calidad de Imagen.**

En ocasiones la calidad de la imagen puede ser estimada directamente de los valores de la función de la imagen en cada punto mediante el empleo de medidas de calidad objetiva, pero en el caso de imágenes médicas, la verdadera calidad la brinda la información que el especialista pueda adquirir de la imagen, lo cual queda en dependencia del tipo de patología que afecte al paciente. Sin embargo los criterios objetivos se continúan usando para tener un estimado de la calidad de la imagen.

Hoy día no existe un valor el cual sirva de guía para determinar si una imagen médica es buena o no, en dependencia de la magnitud que expresen las medidas objetivas; por lo que se sigue indagando, en este aspecto estas pueden ser univariadas o bivariadas. En ambos casos el resultado es un valor numérico de un proceso matemático sobre los valores de los píxeles de la muestra, con el cual puede estimarse un criterio de calidad. Como elemento adicional las medidas subjetivas utilizan un grupo de observadores los cuales exponen sus criterios sobre la propia calidad de la imagen y añaden veracidad a las anteriores medidas. [17], [19]

### **1.2.1 Medidas de calidad objetivas bivariadas.**

La calidad de la imagen está dada por parámetros como: tipo de ruido, contraste y presencia de distorsiones. Para la realización de este experimento se tomaron en cuenta una serie de medidas para evaluar de manera cuantitativa la calidad de la imagen de prueba, todas ellas comparándolas con la imagen original para así tener un patrón con el cual comparar los resultados. [9], [21]

## Error Medio Cuadrático

$$MSE = \frac{1}{m \cdot n} \sum_{i=1}^m \sum_{j=1}^n (X_{(i,j)} - Y_{(i,j)})^2 \quad (3)$$

Evaluando  $X(i, j)$  como la imagen original e  $Y(i, j)$  como la imagen reconstruida se define el error medio cuadrático ( $MSE$ ), donde  $m$  y  $n$  son el total de filas y columnas de la imagen respectivamente.

## Ganancia Relación Señal-Ruido

$$gSNR = 10 \cdot \log_{10} \left( \sigma^2 / MSE \right) \quad (4)$$

donde  $\sigma_x^2$  es la varianza de la imagen sin comprimir;  $\sigma_x^2 = \frac{1}{m \cdot n} \sum_{i=1}^m \sum_{j=1}^n (x_{i,j} - \bar{x})^2$ , con  $\bar{x} = \frac{1}{m \cdot n} \sum_{i=1}^m \sum_{j=1}^n x_{i,j}$ , y  $MSE$  es el error medio cuadrático presentado antes, mientras que  $\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$ , es el valor medio de la imagen o una región de interés.

## Distancia Espectral.

$$SD = \frac{1}{m \cdot n} \sum_{i=1}^m \sum_{j=1}^n (|\sigma(i, j)| - |\hat{\sigma}(i, j)|)^2 \quad (5)$$

donde  $\sigma(i, j)$  y  $\hat{\sigma}(i, j)$  son las partes imaginarias de las transformadas de Fourier de las imágenes sin compactar y reconstruidas.

## Ganancia Relación Contraste-Ruido.

$$gCNR_{dB} = 10 \cdot \log_{10}(CNR_x/CNR_y) \quad (6)$$

Siendo  $CNR = (i_1 - i_2)/\sigma_i$  calculada sobre las imágenes sin comprimir ( $x$ ) y reconstruida ( $y$ ), donde  $i_1$  e  $i_2$  son los valores de intensidad promedio en dos tejidos diferentes y  $\sigma_i$  es la desviación típica del ruido en la imagen  $i$ .

Índice de Similitud Estructural

$$SSIM_{(x,y)} = \frac{(2\mu_x\mu_y + C_1)(2\tau_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\tau_x^2 + \tau_y^2 + C_2)} \quad (7)$$

donde  $\mu_x$  y  $\mu_y$  son los valores medios de la luminancia, mientras que  $\tau_x$  y  $\tau_y$  son los valores estimados para el contraste en ambas imágenes. Las constantes  $C_1$  y  $C_2$  son situadas en la expresión para evitar inestabilidad y están relacionadas con el rango dinámico de los valores de los píxeles:  $C_i = (KL)^2$  donde  $L = 255$ , para imágenes de 8 bpp y con  $K_i \ll 1$ .

El valor de SSIM se promedia sobre  $M$  bloques de la imagen centrados en la posición  $x_i$  y  $y_i$  para obtener los valores promedio de la similitud estructural.

Relación Señal-Ruido promedio.

$$PSNR(dB) = 10 \cdot \log_{10} \left( \frac{MAXp^2}{MSE} \right), \quad (8)$$

Siendo  $MAXp=2^B-1$  y  $B$  es la profundidad de bits de la imagen.

Correlación.

$$Corr = \frac{\sum_{i=1}^m \sum_{j=1}^n X(i,j)^2}{\sum_{i=1}^m \sum_{j=1}^n Y(i,j)^2} \quad (9)$$

### 1.2.1.1 Distancias de Vinet como alternativa de medida de calidad objetiva.

Las mediciones espaciales se utilizan a la hora de estimar las dimensiones de una estructura y tienen particular importancia cuando se estudia la evolución de determinadas lesiones antes y después de un procedimiento médico. La degradación de la calidad en la imagen está directamente relacionada con la disminución de la resolución espacial, lo cual puede ser causa de la introducción de un error de medición espacial, tal como distancia o área realizada directamente por un observador humano o automáticamente mediante un método computacional. Utilizando un algoritmo matemático que sea capaz de detectar los bordes de una estructura y realizando el mismo experimento para imágenes con diferentes TC, la variación del resultado para las diferentes TC estaría estrechamente relacionada con la variación de la calidad de la imagen sujeta al experimento. [17]

### Algoritmos de segmentación.

La segmentación de imágenes es uno de los temas fundamentales más estudiados y útiles en el procesamiento y análisis de imágenes. El objetivo de la segmentación es dividir la imagen en regiones coherentes como paso previo al análisis de su contenido. Existen distintos métodos de segmentación, todo depende de lo que se desee hacer y del tipo de imágenes con las que se esté trabajando. Para identificar las estructuras anatómicas presentes en los estudios por imagen, es necesario hacer una segmentación previa de las imágenes, y posteriormente un etiquetado. La segmentación, se basa en la identificación de cada uno de los objetos de interés, esta es una tarea fundamental para el procesamiento y análisis de imágenes y volúmenes. El obtener una imagen segmentada correctamente sigue siendo un asunto sin resolver, ya que en general los resultados obtenidos suelen tener

sobre-segmentaciones y bordes mal definidos. Generalmente estos inconvenientes suelen recaer sobre el criterio de similitud utilizado por los algoritmos de segmentación. [4], [5], [11], [3]

### **Método *Otsu*.**

Este método es ampliamente utilizado en aplicaciones industriales. Se caracteriza por poseer buena respuesta frente a imágenes en las cuales el objeto a extraer y el fondo se encuentran claramente diferenciados, ya que su principio es la similitud de los píxeles pertenecientes a un objeto y su diferencia respecto al resto.

La mayoría de las técnicas de umbralización se basan en estadísticas sobre el histograma unidimensional. Para localizar los umbrales se utilizan métodos paramétricos y no paramétricos, pero en este caso se usan los no paramétricos, en los cuales los umbrales se obtienen de forma óptima de acuerdo a algún criterio mediante una búsqueda exhaustiva.

Existen diferentes métodos para hallar un umbral, pero al trabajar con imágenes reales es muy común encontrar ruidos y artefactos en las mismas. Según lo reportado en la literatura en estas condiciones el método más robusto es el de *Otsu*.

### **Método de los Contornos Activos.**

Se le llama contorno a la silueta de un objeto negro en un fondo blanco que puede identificarse fácilmente por la brusquedad del cambio en sus bordes. La detección local de un contorno puede encontrarse usando el gradiente  $\Delta u(x, y)$ .

Es un método complejo debido a la cantidad de ruido y artefactos siempre presentes en las imágenes digitales. Por tal motivo se hace difícil el cálculo de gradientes grandes y por ende la imprecisa definición de los contornos. No obstante el método puede mejorarse realizando un suavizado previo de la imagen, lográndose de esta manera mejores resultados.

### 1.2.2 Medidas de calidad subjetivas.

Las medidas subjetivas son aquellas que se realizan con un grupo de observadores humanos (expertos o no), los que expresan su opinión sobre las características de la imagen, como su calidad. Esta valoración queda referida a una escala de valores que se conoce como Escala Promedio de Opinión (del inglés: *Mean Opinion Score*, escala MOS).

Un ejemplo de escala subjetiva puede ser: 1- calidad muy mala, 2- calidad regular, 3- calidad promedio, 4- calidad buena y 5- calidad óptima. Uno de los criterios que pueden valorar los observadores humanos es la medición de distancias sobre la imagen, por ejemplo, entre los bordes de una lesión con el fin de estimar el tamaño (longitud, área y/o volumen) de ésta. Este criterio es una de las propiedades más importantes de la lesión y de la imagen en general, muy relacionado con el diagnóstico que se realiza a través de ella.

A causa de que este tipo de experimento se realiza con observadores humanos, el ambiente en que ellos emitirán sus criterios debe ser ideal, con el fin de minimizar errores durante la realización del experimento por ello todos los observadores realizan sus mediciones en las mismas condiciones ambientales, utilizando siempre el mismo medio de visualización con la misma calibración e iluminación. Otro de los factores importantes es la cantidad de muestras a incluir, ya que si bien es necesario tener una gran muestra con el objetivo de obtener un resultado más fiel, puede ser agotador un experimento con una muestra excesivamente grande la cual puede inducir al cansancio del observador. Otro de los requisitos importantes es garantizar la aleatoriedad de las muestras debido a que el observador pudiera memorizar los resultados. Para realizar este tipo de experimento los observadores deben conocer la herramienta de visualización por lo que reciben indicaciones previas de cómo usarla. La validez de los resultados tiene estrecha relación respecto a la experiencia que tenga el observador en este tipo de imágenes. [17]

**CONCLUSIONES DEL CAPÍTULO.**

Las imágenes de TAC ocupan una gran capacidad en memoria, por ello es necesario encontrar una forma de reducirla. Esto puede hacerse a través de la compresión, que no es más que la eliminación de información redundante en los datos y la codificación de los restantes, mediante algoritmos matemáticos. Existen dos tipos de algoritmos de compresión (sin pérdidas y con pérdidas). En la actualidad se trabaja con el códec JPEG2000 el cual presenta un 60% mayor de compresión que su antecesor, el JPEG; robustez frente a los errores de bits; posibilidad de definir regiones de interés dentro de la imagen, entre otras características. No obstante aún no se han definido parámetros por los cuales guiarse para hacer un uso óptimo de este códec, por lo cual se hace necesario utilizar medidas de calidad para determinar un estándar a partir del cual las imágenes de TAC tengan utilidad para el diagnóstico médico.

## CAPÍTULO 2. MATERIALES Y MÉTODOS.

Con el fin de determinar el valor de TC donde se reduce al mínimo el espacio necesario para almacenar la imagen y para el cual no se afecta su utilidad para el diagnóstico, se lleva a cabo una serie de experimentos sobre el conjunto de imágenes que se describe a continuación.

### 2.1 Selección de las imágenes de prueba.

El grupo de imágenes seleccionado para los experimentos consiste en una muestra de imágenes de TAC de tamaño 512 X 512 píxeles, una profundidad de bits de 8 bpp, ocupando un espacio de 257 Kb en memoria cada una y con una resolución aproximada de 0.5mm/píxel. Las imágenes se escogen por tener lesiones pequeñas con tamaños entre 3 y 18 mm, algunas presentan bajo contraste, lo que hace que sean fácilmente degradables a la hora de ser compactadas a altas TC.

A continuación se muestran algunas de las imágenes que forman parte del experimento. Fig3



Fig.3



## 2.2 Compresión de las imágenes de prueba.

La compresión de las imágenes mediante el Códec JPEG 2000 en la modalidad con pérdidas, se pone en práctica utilizando la implementación de éste en lenguaje C conocida como *JasPer* realizada y puesta a disposición de forma gratuita por la *Image Power Group* y la *British Columbia University*, Canadá. El paquete de software incluye además las funciones *jp2write.m* y *jp2read.m* para la escritura y la lectura respectivamente de las imágenes con extensiones jp2. Esta implementación utiliza el Factor de Compresión *JasPer* (FCJ) para expresar los valores de TC esperados. [2], [6], [18], [¡Error! No se encuentra el origen de la referencia.]

En el caso de las medidas objetivas bivariadas se compara la imagen original con la imagen reconstruida luego de haber sido compactada al valor de TC bajo estudio. Todas las medidas usadas en este caso dan como resultado un valor numérico no siendo necesario la intervención de factores subjetivos, como observadores humanos, para entregar su valor como ocurre en la interpretación de gráficos o histogramas.

## 2.3 Cálculo de las medidas objetivas.

Para realizar el cálculo de las medidas objetivas se utilizan *script* confeccionados con el Software Matlab, a los cuales se le cargan las imágenes del experimento y se le realizan los procesos necesarios para acondicionar la imagen.

La herramienta para cargar las imágenes en formato JPEG 2000 utiliza el *JasPer* y la función *jp2read.m*. Para hacer el cálculo de las medidas de calidad se invoca a cada una de las funciones que calcula las medidas de calidad objetivas, ya sean SNR, PSNR, SSIM, etc. La herramienta permite seleccionar una ROI para realizar el cálculo de las medidas en ella, o simplemente, realizar el cálculo en toda la imagen. Además devuelve los valores de calidad objetivos para cada una de las ROI insertadas por la herramienta de comprimir. Los resultados se inspeccionan desde la propia interfaz, se salvan en un fichero especificado por

el usuario o se exportan para el espacio de trabajo del Matlab para un post-procesamiento. [¡Error! No se encuentra el origen de la referencia.].

### **Medición de distancias de *Vinet*.**

Con el objetivo de utilizar otro método para determinar la fidelidad de los resultados obtenidos se procede a realizar el experimento de medición de distancia con la herramienta confeccionada con el software Matlab, la cual calcula las distancias desde un borde a otro de la lesión en cada una de las imágenes.

En primer lugar se procede a realizar una pre-segmentación del grupo de imágenes previamente comprimido mediante el método de Contornos Activos y mediante *Otsu*, con el fin de reforzar los bordes de las imágenes comprimidas. Luego se cargan las imágenes a la herramienta confeccionada la cual calcula las distancias de *Vinet* en cada una de las imágenes, de modo que se obtiene una curva de distancia de *Vinet* en función de la TC. Posteriormente se grafica una aproximación de los dos métodos de segmentación para llegar a un punto de acuerdo en el cual la TC no ponga en peligro la utilidad diagnóstica de la imagen.

### **2.4 Cálculo de las medidas subjetivas.**

Para realizar este experimento se utiliza una interfaz gráfica Fig.4 la cual se encarga de seleccionar el directorio donde se encuentran las imágenes para mostrarlas aleatoriamente, con el objetivo de no permitir que se memorice el resultado. Este experimento se realiza dos veces, con el fin de asegurar la fidelidad de los resultados. La interfaz elaborada mediante la herramienta GUIDE del Software Matlab permite visualizar las imágenes del experimento y de esta forma el observador clasifica en cinco clases las imágenes, las cuales van desde Muy Mala hasta Óptima, siendo guardados los datos en forma de vector en el espacio de trabajo de dicho software, pudiéndose realizar posteriores análisis.[7],[22].



Fig.4

Luego de realizar el experimento con los observadores expertos, se promedia el resultado de los observadores respecto a la imagen, y así sucesivamente con el resto de las imágenes hasta llegar a un vector en el cual se encuentra el promedio de las evaluaciones de cada una de las imágenes. Como segundo paso, se realiza un gráfico de Calidad de la imagen en función de la TC, con la cual se determina de forma aproximada el rango de TC máximo al cual se puede llegar a aspirar con este tipo de imágenes.

### **Cálculo del coeficiente de *Kappa*.**

Una forma de determinar la validez de las mediciones realizadas es mediante el coeficiente *Kappa*, que es una medida de concordancia propuesta por Cohen en 1960, que se basa en comparar la concordancia observada en un conjunto de datos, respecto a la que podría ocurrir por mero azar. Está claro que una medida simple de concordancia, es la proporción de coincidencias frente al total de sujetos, ejemplo: (acuerdos+desacuerdos /total). No obstante, aunque no exista ninguna relación entre los observadores, o entre las escalas de evaluación, puede haber algún grado de coincidencia por mero azar. Si se denomina  $Co$  a la

proporción de la concordancia observada (en tanto por uno), y  $C_a$ , a la proporción de concordancia que se espere por mero azar,  $K$  será igual a:

$$K = (C_o - C_a) / (1 - C_a) \quad (10)$$

Si  $K$  es cero, significa que la concordancia observada coincide con la que ocurriría por puro azar. Los valores positivos señalan mayor concordancia que la que se esperaría por puro azar. Si el resultado fuese 1, se trataría de una concordancia perfecta. Si  $K$  toma un valor negativo, significa existencia de discordancia, que solamente en la tabla de  $2 \times 2$ , puede llegar hasta -1, lo que señalaría una discordancia total entre las dos clasificaciones o evaluaciones.

Aunque siempre es una escala subjetiva, Landis y Koch propusieron unos límites para el grado de acuerdo estimado con el resultado del cálculo de Kappa:

Tabla.1

<i>Kappa</i>	<i>Estimación del grado de acuerdo</i>
<0	No acuerdo
0.0-0.2	Insignificante
0.2-0.4	Bajo
0.4-0.6	Moderado
0.6-0.8	Bueno
0.8-1.0	Muy Bueno

Este método se menciona mucho en la literatura médica, por su relativa facilidad de cálculo, aunque el cálculo del intervalo de confianza es más complejo. No obstante, solo debe ser aplicado en clasificaciones nominales, es decir, en las que no exista una clara diferencia en el grado de cada una de las categorías, pues no está pensado para clasificaciones en las que algunas categorías son de distinta importancia que otras, por ejemplo: clasificaciones entre banal, leve, moderado, grave, muy grave, extremadamente

grave; en las que no sería lo mismo que la falta de concordancia entre leve y banal, que entre moderado y muy grave. En estos casos, algunos estadísticos asignan un peso a cada una de las diferentes posibilidades de desacuerdo, considerando más importante un desacuerdo entre categorías más distantes, que entre las categorías próximas. No obstante, el problema es determinar qué peso asignar a cada categoría, pues el valor de la concordancia obtenido será diferente con distinta asignación de pesos, y esta siempre tendrá un cierto matiz subjetivo. [20]

**CONCLUSIONES DEL CAPÍTULO.**

En este capítulo se presentan los materiales y los métodos que se utilizan en los experimentos, con vista a determinar el valor de TC aproximado que se obtiene al reducir al mínimo el espacio necesario para almacenar o transmitir la imagen, mientras que se conserva en esta la calidad que la hace útil para el diagnóstico médico y se muestra además el conjunto de medidas de calidad.

## CAPÍTULO 3. RESULTADOS Y DISCUSIÓN

### La compresión de imágenes.

Se posee un fichero que contiene la imagen sin compactar, el cual ocupa un espacio en memoria de 257Kb, el cual se redujo a valores desde 18Kb hasta 2Kb para el intervalo de TC estudiado. El buen rendimiento del códec JPEG 2000 y la alta resolución de estas imágenes permiten mostrar las significativas reducciones de los tamaños en los ficheros.

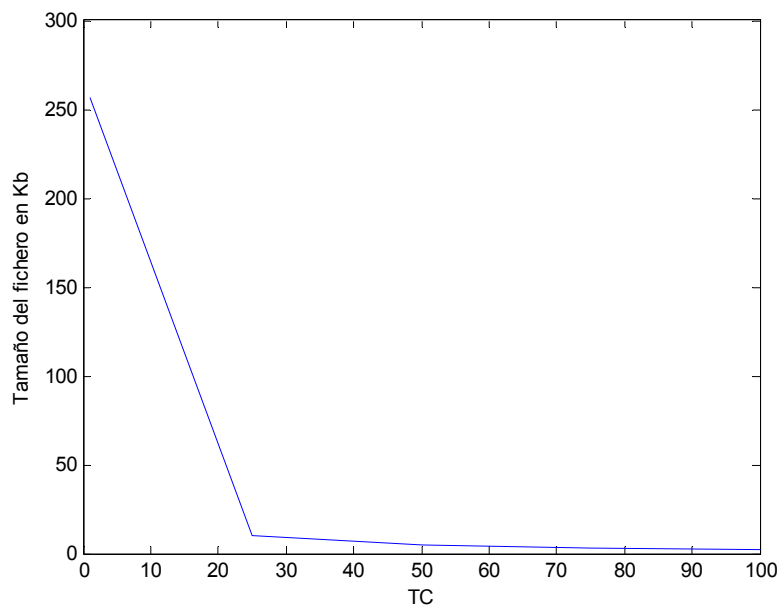


Fig.5 Tamaño del fichero en función de la TC.

### 3.1 Resultados de las medidas objetivas.

Tras compactar las imágenes se realiza el cálculo de las medidas objetivas bivariadas entre la imagen original y las compactadas en el intervalo estudiado.

Con el fin de estudiar el comportamiento de las medidas en función de la TC y extraer el intervalo de interés de TC para el experimento con los algoritmos de segmentación, se representan todas las curvas obtenidas en un mismo gráfico después de ser normalizados sus valores y expresados en por ciento (%). En el caso de las medidas objetivas que con el aumento de la TC, su valor disminuye tales como  $gSNR$  y  $PSNR$ , se hace necesario expresarlas como  $100\% - [\text{valor}]$  para facilitar la representación igualando su comportamiento gráfico al de las restantes.

En la Fig. 6 se muestra el gráfico obtenido del cálculo de las medidas objetivas sobre las imágenes a las diferentes TC. Se puede apreciar medidas tales como  $gCNR$  tienen un comportamiento más indiferente a los cambios de calidad que ocurren en la imagen y otras como  $gSNR$  refleja de manera más efectiva la distorsión presente en la imagen reconstruida. De manera general, el grupo de medidas seleccionado es apropiado para estudiar la pérdida de calidad que ocurre en las imágenes por efecto de la compresión con pérdidas.

El rectángulo delimitado en la Fig.6 representa el intervalo de TC seleccionado para realizar el experimento de medición de distancia con los algoritmos de segmentación, el cual se extiende aproximadamente desde TC 85:1 hasta 18:1 aproximadamente. Para tasas de compactación mayores, la calidad se degrada en una medida tal que la imagen pierde su utilidad para el diagnóstico por lo que sería innecesario incluirlas en el experimento. Para TC menores que el intervalo, el tamaño de la imagen no sufre cambios significativos por lo que los experimentos de segmentación no cumplen ningún objetivo.



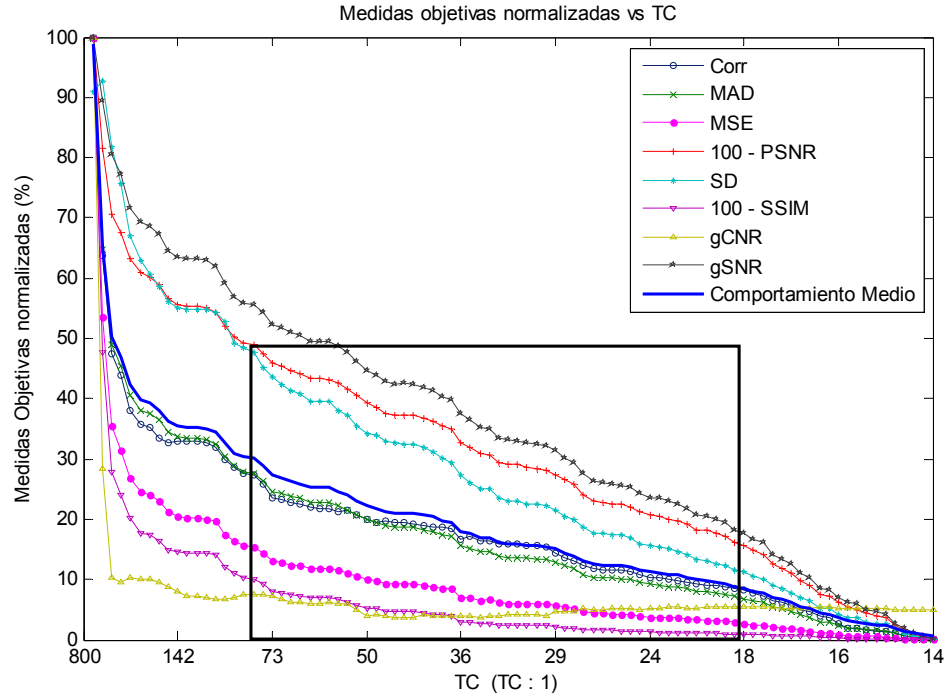


Fig. 6 Gráfico obtenido del comportamiento de las medidas objetivas en función de la TC, el rectángulo en línea discontinua representa el intervalo de TC seleccionado para realizar los experimentos con los algoritmos de segmentación.

### 3.2 Resultados de los experimentos de segmentación.

En la Fig.7 se muestra el resultado de las distancias de *Vinet* para la lesión de una imagen que ha sido compactada a diferentes TC, tal y como fue realizado para las 10 imágenes de prueba por los dos algoritmos seleccionados para el experimento, donde se evidencia claramente el aumento de la distancia entre los bordes de la imagen segmentada y la original a medida que aumenta la TC. El efecto descrito anteriormente se puede apreciar en todos los experimentos que se realizan con las imágenes de prueba. Teniendo en cuenta que los errores que introduce el tomógrafo están en el orden de los milímetros, los observadores sugieren como máximo una desviación del 10% como el máximo error permisible de distancia, por medio del cual se puede determinar la TC máxima, donde el efecto del códec de compresión con pérdidas no afecta considerablemente la calidad de la imagen. El valor aproximado de TC se encuentra alrededor de 24:1

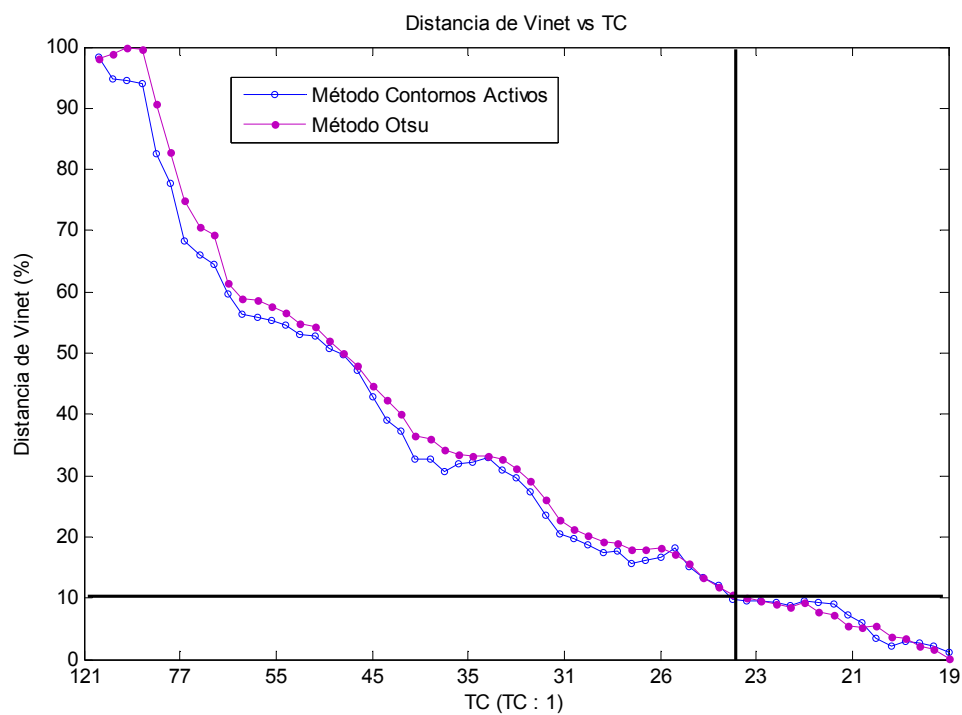


Fig. 7 Distancias de Vinet en función de la TC.

### 3.3 Estimación aproximada de TC máxima por los observadores.

En la Fig.8 se muestra un gráfico del comportamiento de la calidad en función de la TC de las mediciones realizadas por los observadores en las imágenes comprimidas en el intervalo en cuestión.

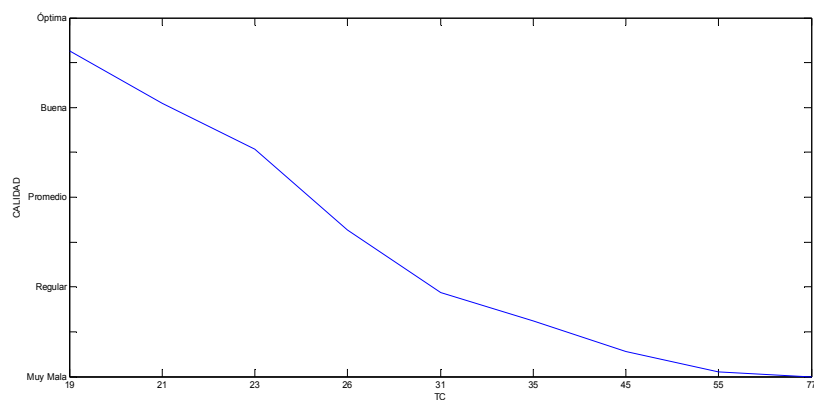


Fig.8 Comportamiento de la calidad en función de la TC de las mediciones realizadas por los observadores.

La tasa de compresión máxima aproximada que no afecta la calidad para este tipo de imágenes en este experimento, se encuentra alrededor del valor TC 21:1 donde la evaluación de los especialistas es buena.

### 3.4 Validación de las observaciones realizadas.

Con el objetivo de evaluar el grado de acuerdo entre los observadores se realiza el cálculo del coeficiente *Kappa*.

Después de calcular el índice *Kappa* entre los 6 observadores se obtienen los resultados que se muestran en la Tabla 2.

Tabla 2: Cálculo de los coeficientes de *Kappa*.

4

Observ.	1	2	3	4	5	6
1	1	0.706	0.623	0.856	0.505	0.733
2	0.706	1	0.677	0.799	0.821	0.865
3	0.623	0.677	1	0.761	0.854	0.931
4	0.856	0.799	0.761	1	0.874	0.760
5	0.505	0.821	0.854	0.874	1	0.684
6	0.733	0.865	0.931	0.760	0.684	1

Al realizar los cálculos de estos resultados se obtiene que existe una relación sustancial en cuanto al grado de acuerdo entre los observadores. El promedio de los coeficientes *Kappa* obtenido es **0.763**, lo que garantiza la validez de las observaciones.

A continuación en la Fig.9 se muestran imágenes usadas en el estudio del desempeño del códec JPEG2000 siendo A la imagen sin compactar y B la imagen compactada a TC 21:1.

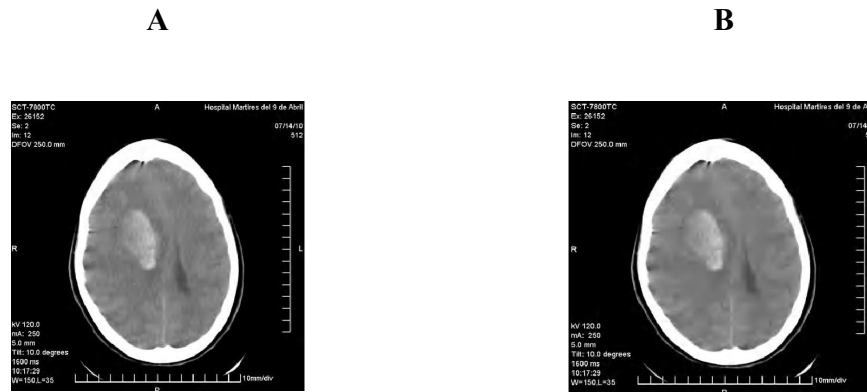


Fig.9 Imágenes de TAC.

Como se puede observar el desempeño del códec fue óptimo al comprimir este tipo de imágenes, por lo que resulta útil a la hora de ser usado para reducir al máximo posible el tamaño de las imágenes de TAC.

## CONCLUSIONES DEL CAPÍTULO

En este Capítulo se muestran los resultados de las medidas objetivas; donde se encuentra que el comportamiento de la gCNR no evidencia grandes cambios a medida que aumenta la TC, no siendo así en medidas como la PSNR y la gSNR las cuales varían su comportamiento significativamente a medida que aumenta la TC. Los experimentos de segmentación expresan que existe una dependencia lineal entre la TC y la desviación típica, es decir, que la desviación típica en las mediciones decrece a medida que la TC también lo hace. La TC máxima que aportan estas medidas es TC 24:1.

En el caso de las medidas subjetivas la tasa de compresión máxima aproximada que no afecta la calidad de las imágenes, se encuentra alrededor del valor TC 21:1 donde la evaluación de los especialistas es buena. Para validar estos resultados se calcula el coeficiente de *Kappa*, en el que se obtuvo un promedio de 0.763, lo que garantiza la validez de las observaciones.

## CONCLUSIONES

- Se confeccionó una Base de Datos (BD) con las imágenes que se utilizan para realizar el experimento, las cuales poseen buena resolución y lesiones visibles de fácil identificación.
- Se realizó un análisis de las medidas objetivas, las cuales aportan valores numéricos a tener en cuenta como referencia de calidad a la hora de comparar estos con otras imágenes de Tomografía Axial Computarizada.
- Con la implementación de algoritmos de segmentación se realizaron medidas de distancia en las imágenes, observándose que a medida que se comprime se pierde similitud con la imagen original, por lo que por consenso de los especialistas se tomó como máximo error una desviación del 10% en las medidas de distancia.
- Como medida subjetiva principal se realizó un experimento con observadores expertos, los cuales clasificaron las imágenes en categorías para su procesamiento.
- En último lugar se creó una estrategia para la obtención de forma aproximada de la máxima TC, combinándose las medidas objetivas y subjetivas; y se estimó como máxima TC 21:1, donde las imágenes no pierden calidad significativa y aún son útiles para el diagnóstico.

- 1 Acharya, T., and Ray, A.K.: ‘Image processing: principles and applications’  
(Wiley-Interscience, 2005. 2005)
- 2 Adams, M.D., and Kossentini, F.: ‘JasPer: A software-based JPEG-2000 codec  
implementation’, in Editor (Ed.)^(Eds.): ‘Book JasPer: A software-based JPEG-  
2000 codec implementation’ (IEEE, 2000), pp. 53-56 vol. 52
- 3 Becilla Peñafiel, C., and Matias Moris, J.: ‘Segmentación de Imágenes Médicas  
para Detección de detalles Tomografía y Ecografías’, 2009.
- 4 Caselles, V., and Frangi, A.: ‘La segmentación de imágenes. El método de los  
contornos activos geométricos’, 2010.
- 5 Castañeda, I.M.: ‘Efectos de la compresión con pérdidas sobre mediciones de  
distancia en imágenes de TAC’, Marta Abreu de Las Villas, 2009.
- 6 Gonzalez, R.C., and Woods, R.E.: ‘Digital Image Processing, 2001’ (Prentice-  
Hall Inc.)
- 7 Guerrero, D.O.: ‘Manual de Interfaz gráfica en Matlab’.2008
- 8 JasPer                      Software                      Reference                      Manual.  
<http://www.ece.uvic.ca/~frodo/jasper.htm>.(1999)
- 9 Lau C., J. E Cabral, D. R. Haynor, and Kim, Y.y.: ‘Telemedicine’, Handbook of  
Medical Imaging, 2000, 3, pp. 305-331
- 10 Loubele, M., Jacobs, R., Maes, F., Schutyser, F., Debaveye, D., Bogaerts, R.,  
Coudyzer, W., Vandermeulen, D., Van Cleynenbreugel, J., and Marchal, G.:  
‘Radiation dose vs. image quality for low-dose CT protocols of the head for  
maxillofacial surgery and oral implant planning’, Radiation protection  
dosimetry, 2005, 117, (1-3), pp. 211-216.

- 11 Matias, J., Becilla Peñafiel, C., and Chávez, P.: 'Segmentación de Imágenes médicas para detección de detalles', 2009.
- 12 Miano, J.: 'Compressed image file formats: jpeg, png, gif, xbm, bmp' (Addison-Wesley Professional, 1999. 1999)
- 13 Otsu, N.: 'A threshold selection method from grey level histograms', in Editor (Ed.) ^ (Eds.): 'Book A threshold selection method from grey level histograms' (1979, edn.)
- 14 Penedo M, Marco Julio., Tahoces, P.G. Souto, and M. y Vidal, J.J.: 'Effects of JPEG2000 data compression on an automated system for detecting clustered microcalcifications in digital mammograms', IEEE Trans. on Information Technology in Biomedicine, 2006, 10, (2), pp. 354-361
- 15 Pianykh, O.S.: 'Digital Imaging and Communications in Medicine (DICOM): A practical introduction and survival guide' (Springer, 2011. 2011)
- 16 Rabbani, M., and Joshi, R.: 'An overview of the JPEG 2000 still image compression standard', Signal Processing: Image Communication, 2002, 17, (1), pp. 3-48.
- 17 Rosenberger, C., Chabrier, S., Laurent, H., and Emile, B.: 'Unsupervised and Supervised Image Segmentation Evaluation', 2006, pp. 365-394.
- 18 Santa-Cruz, D., Grosbois, R., and Ebrahimi, T.: 'JPEG 2000 performance evaluation and assessment', Signal Processing: Image Communication, 2002, 17, (1), pp. 113-130.
- 19 Suetens, P.: 'Fundamentals of Medical Imaging. Cambridge University Press, Leuven', 2002.
- 20 Viera, A.J., and M., M.J.: 'Understanding Interobserver Agreement: The Kappa statistic.' May 2005.
- 21 Viera, J.E.P.: 'Evaluación de la calidad en imágenes de resonancia magnética compactas con pérdidas.', Marta Abreu de Las Villas, 2008.
- 22 www.mathworks.com: 'MATLAB The Language of Technical Computing,' (2004. 2004).



## ANEXOS

### ***Script utilizados a la hora de calcular las medidas objetivas.***

#### Cargar imágenes JP2

```
function [im, map, bits]=jp2read(filename,varargin)

% JP2READ - Reads JPEG 2000 image files from disk. Needs jasper installed

% [IM, MAP, BITS] =JP2READ ('FILENAME')

% (C) Peter Rydesäter 2002-11-08

% See also: JP2IMFORMATS, JP2WRITE, IMFORMATS, IMREAD, IMWRITE,
IMFINFO

Map = [];

try,f=fopen(filename,'r');          fclose(f);catch          er=lasterror;          if
strcmp(E.identifier,'MATLAB:interrupt'),rethrow(er);end,end

if f==-1, if length( which(filename) ), filename=which(filename); f=inf; end; end

if f==-1, error('MATLAB:io','File "%s" does not exist or is not possible to read
from.',filename); end

opts=varargin;

[jasperoptstr,opts,optexist] = parse_parameter_list('jasper',opts,'format');

if optexist.format,
```

```
    fmtstr=[' --input-format ' opts.format];
else
    idx = find(filename == '.');
    if (~isempty(idx))
        ext = filename((idx(end) + 1):end);
        fmtstr=[' --input-format ' ext];
    else
        fmtstr="";
    end
end

end

%% Try to read via 1-31 bits grayscale pgxformat
tmp=[tempname '.pgx'];
exit_code=dos(sprintf('jasper    --input    "%s"    %s    --output    "%s"    %s\n',
2,filename,fmtstr,tmp,jasperoptstr));
if exit_code==0,
    try,
        [im,map,bits]=pgxread(tmp);
    catch
        delete(tmp);
        rethrow(lasterror);
    end
else
    %% if pgx fails
    delete(tmp);
    tmp=[tempname '.bmp'];
```

---

```

    exit_code=dos(sprintf('jasper --input "%s" %s --output "%s" %s 2',filename,fmtstr,tmp,
jasperoptstr));

    st=dir(tmp);

    if isempty(st), st(1).bytes=0; end

    if st(1).bytes==0, error('MATLAB:io','The file "%s" seems not possible to read via this
interface.',filename); end

    im=imread(tmp,'bmp');

    st=imfinfo(tmp);

    bits=st.BitDepth;

end

delete(tmp);

return;

```

### Índice de Similitud Estructural. (SSIM).

```
function [mssim, ssim_map] = ssim_index(img1, img2)
```

```

%=====
=====

```

```
%SSIM Index, Version 1.0
```

```
%Copyright(c) 2003 Zhou Wang
```

```
%All Rights Reserved.
```

```
%
```

```
%The author is with Howard Hughes Medical Institute, and Laboratory
```

%for Computational Vision at Center for Neural Science and Courant

%Institute of Mathematical Sciences, New York University.

%

%-----

%Permission to use, copy, or modify this software and its documentation

%for educational and research purposes only and without fee is hereby

%granted, provided that this copyright notice and the original authors'

%names appear on all copies and supporting documentation. This program

%shall not be used, rewritten, or adapted as the basis of a commercial

%software or hardware product without first obtaining permission of the

%authors. The authors make no representations about the suitability of

%this software for any purpose. It is provided "as is" without express

%or implied warranty.

%-----

%

%This is an implementation of the algorithm for calculating the

%Structural SIMilarity (SSIM) index between two images. Please refer

%to the following paper:

%

%Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image

%quality assessment: From error visibility to structural similarity"

%IEEE Transactions on Image Processing, vol. 13, no. 4, pp.600-612,

%Apr. 2004.

%

---

```
%Kindly report any suggestions or corrections to zhouwang@ieee.org

%
%-----
%

%Input : (1) img1: the first image being compared
%      (2) img2: the second image being compared
%      (3) K: constants in the SSIM index formula (see the above
%          reference). default value: K = [0.01 0.03]
%      (4) window: local window for statistics (see the above
%          reference). default window is Gaussian given by
%          window = fspecial('gaussian', 11, 1.5);
%      (5) L: dynamic range of the images. default: L = 255
%
%Output: (1) mssim: the mean SSIM index value between 2 images.
%      If one of the images being compared is regarded as
%      perfect quality, then mssim can be considered as the
%      quality measure of the other image.
%      If img1 = img2, then mssim = 1.
%      (2) ssim_map: the SSIM index map of the test image. The map
%      has a smaller size than the input images. The actual size:
%      size(img1) - size(window) + 1.
%
%Default Usage:
% Given 2 test images img1 and img2, whose dynamic range is 0-255
```

---

```

%
% [mssim ssim_map] = ssim_index(img1, img2);
%
%Advanced Usage:
% User defined parameters. For example
%
K = [0.05 0.05];
window = ones(8);
L = 100;
% [mssim ssim_map] = ssim_index(img1, img2, K, window, L);
%
%See the results:
%
% mssim           %Gives the mssim value
% imshow(max(0, ssim_map).^4) %Shows the SSIM index map
%
%=====
=====

if (nargin < 2 | nargin > 5)
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

```

---

```
if (size(img1) ~= size(img2))
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

[M N] = size(img1);

if (nargin == 2)
    if ((M < 11) | (N < 11))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    window = fspecial('gaussian', 11, 1.5); %
    K(1) = 0.01; % default settings
    K(2) = 0.03; %
    L = 255; %
end

if (nargin == 3)
    if ((M < 11) | (N < 11))
        ssim_index = -Inf;
```

---

```
        ssim_map = -Inf;

    return

end

window = fspecial('gaussian', 11, 1.5);

L = 255;

if (length(K) == 2)

    if (K(1) < 0 | K(2) < 0)

        ssim_index = -Inf;

        ssim_map = -Inf;

        return;

    end

else

    ssim_index = -Inf;

    ssim_map = -Inf;

    return;

end

end

if (nargin == 4)

    [H W] = size(window);

    if ((H*W) < 4 | (H > M) | (W > N))

        ssim_index = -Inf;

        ssim_map = -Inf;

    return

end
```



---

```
end

L = 255;

if (length(K) == 2)
    if (K(1) < 0 | K(2) < 0)
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
else
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

end

end

if (nargin == 5)
    [H W] = size(window);
    if ((H*W) < 4 | (H > M) | (W > N))
        ssim_index = -Inf;
        ssim_map = -Inf;

        return
    end

    if (length(K) == 2)
        if (K(1) < 0 | K(2) < 0)
```

---

```
        ssim_index = -Inf;

        ssim_map = -Inf;

        return;

    end

else

    ssim_index = -Inf;

    ssim_map = -Inf;

    return;

end

end

end

C1 = (K(1)*L)^2;

C2 = (K(2)*L)^2;

window = window/sum(sum(window));

img1 = double(img1);

img2 = double(img2);


mu1 = filter2(window, img1, 'valid');

mu2 = filter2(window, img2, 'valid');

mu1_sq = mu1.*mu1;

mu2_sq = mu2.*mu2;

mu1_mu2 = mu1.*mu2;

sigma1_sq = filter2(window, img1.*img1, 'valid') - mu1_sq;

sigma2_sq = filter2(window, img2.*img2, 'valid') - mu2_sq;
```

---

```

sigma12 = filter2(window, img1.*img2, 'valid') - mu1_mu2;

if (C1 > 0 & C2 > 0)

    ssim_map = ((2*mu1_mu2 + C1).*(2*sigma12 + C2))./((mu1_sq + mu2_sq +
C1).*(sigma1_sq + sigma2_sq + C2));

else

    numerator1 = 2*mu1_mu2 + C1;

    numerator2 = 2*sigma12 + C2;

    denominator1 = mu1_sq + mu2_sq + C1;

    denominator2 = sigma1_sq + sigma2_sq + C2;

    ssim_map = ones(size(mu1));

    index = (denominator1.*denominator2 > 0);

    ssim_map(index) =
(numerator1(index).*numerator2(index))./(denominator1(index).*denominator2(index));

    index = (denominator1 ~= 0) & (denominator2 == 0);

    ssim_map(index) = numerator1(index)./denominator1(index);

end

mssim = mean2(ssim_map);

return

```

### Cálculo de medición de distancia de Vinet.

```
function [vinetv] = vinet(S1,S2,assigntype)
```

---

% The Vinet measure proposed in [1] is a measure to quantify the distance  
% between two segmentations S1 and S2 of the same image.

%

% function [vinetv] = Vinet(S1,S2,assigntype)

%

% S1, S2 : Labels matrix. The labels must be consecutives positives

% natural numbers. S1 and S2 must have the same size.

% assigntype : Type of assignment. "1" (default value) indicate

% iteratively searching the maxima and "2" by the Hungarian algorithm.

%

% vinetv : Vinet value of matching both segmentation. For convenience is

% multiplied by 100.

%

% [1] L. Cohen, L. Vinet, P. T. Sander, A. Gagalowicz, "Hierarchical

% regional based stereo matching," in Proc. Computer Vision and Pattern

% Recognition, 1989, pp. 416–421.

% M. Orozco Monteagudo. FIE, Universidad Central de Las Villas, Cuba.

% morozco@uclv.edu.cu

% 14-may-2009.

S1 = uint16(S1);

S2 = uint16(S2);

---

```
if nargin < 3
    assigntype = 1;
end

m = max(max(S1)); % Number of regions in S1
n = max(max(S2)); % Number of regions in S2
[p,q] = size(S1); % Dimension of the image
N = p*q;          % Number of pixels of the image

% Construction of matrix T.

T = zeros(m,n);
for i = 1:p,
    for j = 1:q,
        x = S1(i,j);
        y = S2(i,j);
        T(x,y) = T(x,y) + 1;
    end
end

% Vinet value from matrix T.

if assigntype == 1 % Iterative (greedy)
```

---

```

C = 0;

for i = 1:min([m,n])

    maxaux = max(max(T));

    C = C + maxaux;

    auxinds = find(T == maxaux);

    [x,y] = ind2sub([m n],auxinds(1));

    T(:,y) = 0;

    T(x,:) = 0;

end

vinetv = 1 - C/N;

else % Hungarian algorithm

    %C = 0;

    T1 = N - T;

    [assign,cost] = assignmentoptimal(T1);

    vinetv = 1.0 - (double(min([m,n]))*N - cost)/N;

end

vinetv = 100*vinetv;

```

### Distancia Media Espectral

```

function m = mad(x, y);

% Mean Absolut difference;

% m = mad(x, y);

d = abs(y - x);

```

---

```
m = sum(d(:))/(size(x,1)^2);
```

### Relación Señal-Ruido

```
function sn = snr(ori, Img);
```

```
%SNR2 Comparative Signal-to-noise Ratio (SNR): Tilo Strutz, Grundlagen der
```

```
%Bilddaten Kompression.
```

```
%
```

```
% snr = snr2(Xori, Xrec);
```

```
m = mean2(ori);
```

```
d = ori - Img;
```

```
ma = max(d(:));
```

```
mi = min(d(:));
```

```
md = mean2(d);
```

```
N = size(ori,1)*size(ori,2);
```

```
sigma_x_q = sum(sum((ori - m).^2))/N;
```

```
sigma_n_q = sum(sum((d - md).^2))/N;
```

```
sn = 10*log10( sigma_x_q/sigma_n_q);
```

### Script utilizado para la confección de la herramienta subjetiva.

```
function varargout = quality_1_5(varargin)
```

```
% QUALITY_1_5 M-file for quality_1_5.fig
```

```
% QUALITY_1_5, by itself, creates a new QUALITY_1_5 or raises the existing
```

```
% singleton*.
%
% H = QUALITY_1_5 returns the handle to a new QUALITY_1_5 or the handle to
% the existing singleton*.
%
% QUALITY_1_5('CALLBACK',hObject,eventData,handles,...) calls the local
% function named CALLBACK in QUALITY_1_5.M with the given input arguments.
%
% QUALITY_1_5('Property','Value',...) creates a new QUALITY_1_5 or raises the
% existing singleton*. Starting from the left, property value pairs are
% applied to the GUI before quality_1_5_OpeningFcn gets called. An
% unrecognized property name or invalid value makes property application
% stop. All inputs are passed to quality_1_5_OpeningFcn via varargin.
%
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
% instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help quality_1_5

% Last Modified by GUIDE v2.5 04-Jun-2013 19:53:05

% Begin initialization code - DO NOT EDIT
```



```
gui_Singleton = 1;

gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @quality_1_5_OpeningFcn, ...
                  'gui_OutputFcn', @quality_1_5_OutputFcn, ...
                  'gui_LayoutFcn', [] , ...
                  'gui_Callback', []);

if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% End initialization code - DO NOT EDIT


% --- Executes just before quality_1_5 is made visible.
function quality_1_5_OpeningFcn(hObject, eventdata, handles, varargin)

% This function has no output args, see OutputFcn.

% hObject    handle to figure

% eventdata reserved - to be defined in a future version of MATLAB
```

---

```
% handles    structure with handles and user data (see GUIDATA)

% varargin   command line arguments to quality_1_5 (see VARARGIN)


% Choose default command line output for quality_1_5

handles.output = hObject;


% Update handles structure

guidata(hObject, handles);


% UIWAIT makes quality_1_5 wait for user response (see UIRESUME)

% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.

function varargout = quality_1_5_OutputFcn(hObject, eventdata, handles)

% varargout  cell array for returning output args (see VARARGOUT);

% hObject    handle to figure

% eventdata  reserved - to be defined in a future version of MATLAB

% handles    structure with handles and user data (see GUIDATA)


% Get default command line output from handles structure

varargout{1} = handles.output;
```

---

% --- Executes on button press in pushbutton1.

function pushbutton1\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% --- Executes on selection change in popupmenu1.

function popupmenu1\_Callback(hObject, eventdata, handles)

% por up

handles.evalua(handles.pos) = get(hObject,'Value');

guidata(hObject,handles);

%siguiente

if handles.cont == handles.max\_cont

    msgbox('Experimento Terminado.!' 'Experimento Subjetivo');

    set(handles.popupmenu1,'Enable', 'off')

else

    handles.cont = handles.cont + 1;

    handles.filename = handles.filelist(handles.pos).name; %se obtiene el nombre de la imagen

---

```
guidata(hObject,handles);

handles.pos = handles.ord(handles.cont);

guidata(hObject,handles);

load_vis(hObject, eventdata, handles)

end


% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu1 contents as cell array
%      contents{get(hObject,'Value')} returns selected item from popupmenu1


% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called


% Hint: popupmenu controls usually have a white background on Windows.
%      See ISPC and COMPUTER.
if      ispc      &&      isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
```

---

end

% --- Executes on button press in pushbutton2.

function pushbutton2\_Callback(hObject, eventdata, handles)

% hObject handle to pushbutton2 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% -----

function Untitled\_1\_Callback(hObject, eventdata, handles)

% hObject handle to Untitled\_1 (see GCBO)

% eventdata reserved - to be defined in a future version of MATLAB

% handles structure with handles and user data (see GUIDATA)

% -----

function Untitled\_2\_Callback(hObject, eventdata, handles)

% cargar ruta y hacer labores iniciales

handles.thepath = uigetdir();

if handles.thepath ~= 0

handles.thepath = [handles.thepath '\'];

```
handles.filelist = dir([handles.thepath '*.jp2']);%aqui estan los nombres de cada una de las
imagenes
```

```
set(handles.popupmenu1,'Enable','on')
```

```
handles.cont = 1;
```

```
[handles.max_cont,s] = size(handles.filelist);
```

```
handles.ord = randperm(handles.max_cont);
```

```
handles.pos = handles.ord(handles.cont);
```

```
guidata(hObject,handles);
```

```
load_vis(hObject, eventdata, handles)
```

```
handles.eval = [];
```

```
guidata(hObject,handles);
```

```
end
```

```
%UTILITY FUNCTIONS
```

```
function load_vis(hObject, eventdata, handles)
```

```
%*****
```

```
***
```

```
%*****
```

```
%load curent image and visualize
```

```
handles.filename = handles.filelist(handles.pos).name; %se obtiene el nombre de la imagen
```

```
handles.img = jp2read([handles.thepath handles.filename]);% se carga la imagen
```

```
handles.img = im2double(handles.img);
```

```
guidata(hObject,handles);
```

```
colormap('gray');
```

```
% axes(handles.axes1)
```

```
imshow(handles.img,[])
```

```
%*****
```

```
guidata(hObject,handles);
```

```
% -----
```

```
function Untitled_3_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to Untitled_3 (see GCBO)
```

```
assignin('base','eval',handles.eval);
```

```
% --- Executes on key press with focus on pushbutton2 and none of its controls.
```

```
function pushbutton2_KeyPressFcn(hObject, eventdata, handles)
```

```
% hObject    handle to pushbutton2 (see GCBO)
```

```
% eventdata  structure with the following fields (see UICONTROL)
```

```
%   Key: name of the key that was pressed, in lower case
```

```
%   Character: character interpretation of the key(s) that was pressed
```

```
%   Modifier: name(s) of the modifier key(s) (i.e., control, shift) pressed
```

```
% handles    structure with handles and user data (see GUIDATA)
```

```
% --- If Enable == 'on', executes on mouse press in 5 pixel border.
```

```
% --- Otherwise, executes on mouse press in 5 pixel border or over pushbutton2.
```

---

```
function pushbutton2_ButtonDownFcn(hObject, eventdata, handles)

% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```