

PRÁCTICA FINAL “COMPRESIÓN DE IMÁGENES BASADA EN DCT: ESTUDIO EXPERIMENTAL”

Compresión Multimedia, 4º curso, 2020/21



Alumno: José Antonio Nicolás Navarro
Email: joseantonio.nicolasn@um.es

Índice de contenidos

1. Resumen	3
2. Introducción	3
3. Metodología.....	4
3.1. Criterios para la selección del conjunto de imágenes.....	4
3.1.1. Tipo de imagen.....	4
3.1.2. Tamaño de la imagen	4
3.1.3. Otros elementos deseables en una imagen.....	4
3.1.4. Entropía, frecuencia y variabilidad	5
3.1.4.1. Idea general del concepto de entropía	5
3.1.4.2. Entropía calculada entre la cuantización y la codificación.....	5
3.1.4.4. Variabilidad	6
3.1.4.5. Criterios concluidos.....	6
3.2. Selección del conjunto de imágenes	7
3.3. Selección de valores del factor de calidad	13
3.4. Hipótesis generales y cuestiones a tratar durante la experimentación.....	14
3.5. Procedimiento desarrollado durante la investigación.....	15
4. Resultados experimentales.....	17
4.1. Imagen “loros”	17
4.2. Imagen “F-16”	23
4.3. Imagen “faro”	29
4.4. Imagen “despegue”	33
4.5. Imagen “mandril”	38
4.6. Imagen “casa”	44
4.7. Sobre la influencia de la entropía de la imagen	49
5. Parte opcional I: Análisis de los tiempos de ejecución	50
6. Parte opcional II: Comparativa usando software alternativo.....	54
7. Conclusiones.....	59
8. Bibliografía.....	60
9. Anexos.....	61
9.1. Anexo I – Extensión de conceptos técnicos.....	61
9.1.1. Relación de compresión	61
9.1.2. Error cuadrático medio (MSE)	61
9.1.3. Matrices DCT e imágenes base	61
9.1.4. Cuantización	62

9.1.5. Codificación en JPEG	63
9.2. Anexo II – Implementación de software para facilitar el análisis.....	64

1. Resumen

En este documento se realizará una evaluación experimental de las implementaciones para compresión y descompresión de imágenes JPEG desarrollados en prácticas anteriores. Estos compresores y descompresores serán aplicados a un conjunto reducido, de unas 6 imágenes y 7 factores de calidad. Los resultados que arroje el estudio serán evaluados de forma cualitativa y cuantitativa, teniendo en cuenta ciertos parámetros para comparar (relación de compresión, error cuadrático medio, tiempos de ejecución...), posible relación de los resultados con el funcionamiento teórico del algoritmo JPEG y la percepción visual sobre la recuperación de las imágenes comprimidas con respecto a las originales.

2. Introducción

La compresión de datos es el proceso de convertir un flujo de datos de entrada en otro que tenga un tamaño más pequeño. La compresión (y descompresión) de imágenes es objeto del estudio experimental que se va a llevar a cabo.

La idea detrás de comprimir una imagen es conseguir aprovechar lo máximo los dispositivos de almacenamiento y aumentar la velocidad de procesamiento y transmisión de estas imágenes (también a través de canales de comunicación) debido a su tamaño reducido. Existen muchos algoritmos de compresión, pero este documento se centra en la técnica JPEG.

El algoritmo JPEG se basa en la compresión con pérdidas, puesto que durante el proceso de compresión se pierde información y la imagen podrá ser recuperada de forma que quede como la original, aunque esta pérdida de información consigue que los datos puedan comprimirse mucho afectando levemente a su calidad perceptual. Este algoritmo tiene una manera de codificar los datos de una imagen por defecto, fija según el estándar, aunque en prácticas se desarrolló, además de esta versión, otra usando una codificación “custom” o ajustada a las necesidades de cada imagen. El proceso general que sigue una imagen para ser comprimida usando este algoritmo puede ser consultado a través del [Anexo I](#).

La implementación de los compresores y descompresores desarrollados en prácticas anteriores serán puestos a prueba y analizados sobre un conjunto de imágenes en los siguientes puntos para analizar su comportamiento y cómo éste afecta a las imágenes, así como la extracción de conclusiones sobre los mismos y sobre JPEG en general.

3. Metodología

3.1. Criterios para la selección del conjunto de imágenes

El punto de partida para realizar la selección de imágenes que se analizarán en esta evaluación experimental de los compresores y descompresores es que estas imágenes deben ser **imágenes sin pérdida**, preferiblemente en formato **BMP** porque 2 imágenes con las mismas dimensiones ocupan el mismo tamaño independientemente de su contenido y esto facilitará el estudio más adelante. Además del formato, debe verse claro que las imágenes no contengan defectos como el pixelado o el contorneado (para no comprimir imágenes que ya han pasado por un proceso de compresión previamente). Teniendo esto en cuenta, se pueden empezar a establecer los criterios de selección.

3.1.1. Tipo de imagen

JPEG funciona muy bien desde el punto de vista de la compresión de imágenes cuando trata con imágenes fotográficas y pinturas de escenas realistas con suaves variaciones de tono y color ^[1].

Es por esta razón que el primer criterio para la selección de estas imágenes parece razonable que sea que estas imágenes no hayan sido diseñadas artificialmente ni sean artísticas.

3.1.2. Tamaño de la imagen

Para facilitar el análisis de forma que se puedan comparar las medidas de los parámetros utilizados entre imágenes es conveniente que todas las imágenes que vayan a ser analizadas sean del mismo tamaño para que el estudio sea independiente de esta variable y sea más sencillo extraer conclusiones. Este tamaño no puede ser muy grande, primero, debido al tiempo que tarda MATLAB comprimiendo y descomprimiendo imágenes y segundo, por el tamaño final que pueda alcanzar este documento.

Las imágenes no pueden ser muy pequeñas tampoco, pues si ya de entrada se perciben con una calidad pésima debido a su resolución, no tiene sentido su estudio con JPEG.

El segundo criterio establecido, por tanto, para la selección de imágenes, es su tamaño, que se establecerá en 512x512 píxeles en la medida de lo posible y para la mayoría de imágenes, ya que este tamaño me parece razonable y válido para aplicar JPE, puesto que se obtienen tiempos de ejecución lo suficientemente bajos para no desesperarse con la ineficiencia de MATLAB.

3.1.3. Otros elementos deseables en una imagen

Un criterio más para la selección de imágenes es la inclusión de ciertos elementos dentro de las mismas que pueden resultar interesantes ver cómo el compresor y descompresor las maneja.

Es conocido que las imágenes con cambios de color brusco o con texto no se recuperan tan bien como las demás como será explicado en el siguiente punto y que servirá como justificación de este y de otros criterios. Así que, para el estudio, incluiré alguna imagen con bordes bastante pronunciados o con texto (con bastantes bordes implícitos) para estudiar cómo de mala es la recuperación de la imagen original.

Otro fenómeno conocido que se da en imágenes con regiones de píxeles parecidas es el de “contouring” o efecto de contorneado. Dondequiera que haya cambios suaves en los colores en la imagen original, se producen ahora cambios bruscos. Este efecto se hace especialmente evidente en zonas donde el cambio gradual de colores se ha transformado en regiones “manchadas” de valores constantes. El efecto se debe a que se está cuantizando una gama de valores diferentes a la misma etiqueta, y en la recuperación, al descuantizar, quedan como el mismo valor. ^[3]

3.1.4. Entropía, frecuencia y variabilidad

Para finalizar con la lista de criterios para seleccionar una imagen, es necesario establecer ciertas definiciones de conceptos e ideas que serán usados más adelante también.

La explicación y los razonamientos sobre el funcionamiento de JPEG serán explicados de la forma más breve y sencilla posible que me permita avanzar sobre la justificación de los criterios. Soy consciente de que no se puede pasar por estas explicaciones sin que surjan preguntas. Es por ello que los siguientes puntos serán extendidos en el [Anexo I](#), así como una explicación sobre la definición de relación de compresión y error cuadrático medio, que son dos de los parámetros que mediremos sobre la imagen.

3.1.4.1. Idea general del concepto de entropía

El primero de estos conceptos importantes a los que haré referencia en el estudio es la entropía. Desde un punto de vista puramente computacional nos interesa la siguiente interpretación de la entropía: es el número de cifras (en promedio) en base b que se necesitan para codificar cada uno de los símbolos del alfabeto de una fuente. Este valor es máximo, si la distribución de probabilidad de los símbolos de la fuente es uniforme. ^[4]

La entropía, de forma general, **está ligada a la frecuencia** de aparición de sus símbolos o valores. Cuando todos estos símbolos o valores aparecen con la misma frecuencia, entonces se generan códigos uniformes (códigos donde las palabras código tienen la misma longitud), y cuando unos símbolos tienen una frecuencia de aparición más alta que otros, entonces se generan códigos de longitud diferente.

Idealmente, lo más deseable es que la fuente no tenga símbolos con la misma frecuencia de aparición, ya que si esto ocurriera se estaría afectando a la longitud de los códigos de forma negativa, haciéndolos más largos debido a la mayor entropía.

3.1.4.2. Entropía calculada entre la cuantización y la codificación

En el caso que concierne a la práctica con JPEG, los símbolos de los que se hablaba en el apartado anterior, en el caso de las imágenes, la codificación no se hace sobre la frecuencia de aparición de los valores de los canales RGB sino sobre de la frecuencia de aparición de los coeficientes transformados y cuantizados.

La entropía que yo consideraré es la entropía de los datos originales. Esta entropía no tiene nada que ver con la posterior codificación de la imagen, la cual la interpretaré como una fuente que emite símbolos de tres valores (RGB). La entropía considerada no influye en la longitud de los códigos, pero el objetivo es determinar si está relacionada de alguna manera con los datos que arroja la compresión de imágenes (aunque adelanto que esto no va a ocurrir puesto que no existen resultados matemáticos a favor de esta idea).

3.1.4.4. Variabilidad

La variabilidad es un concepto que en términos de transformadas discretas del coseno indica que, si una imagen presenta una alta variabilidad, o lo que es lo mismo, **una continuidad de cambios bruscos de valores**, serán necesarias funciones coseno o exponenciales complejas de altas frecuencias para recuperar una señal. Por el contrario, una baja variabilidad indica que una señal puede ser recuperada con funciones coseno o exponenciales complejas de bajas frecuencias.

En términos de JPEG, esto implica que una alta variabilidad signifique usar muchas de las 64 imágenes base. Una baja variabilidad implica que los valores en uno de esos bloques 8x8 no oscilan y se usarán pocas bases. Cuantos más coeficientes sea necesario codificar una vez cuantizado el bloque, es esperable que se podrá comprimir menos una imagen porque más valores necesitarán ser codificados. Es por esto que JPEG funciona muy bien cuando las imágenes presentan suaves variaciones de tono y color, ya que más coeficientes se convertirán en 0 al ser cuantizados y menos datos serán codificados.

La pérdida de calidad en los bordes con cambios bruscos de color (incluida la aparición de texto) está relacionada con el hecho de que en los bloques 8x8 que se encuentran en esos bordes requieren, por lo general, de altas frecuencias en términos de transformadas discretas del coseno, y estas altas frecuencias, al ser la matriz DCT cuantizada, se van a perder, de forma que se pierde el detalle también debido a la pérdida de nitidez, convirtiéndose estos bloques en ‘manchurroneos’ de baja variabilidad. El caso contrario se da en bloques donde hay gradientes de color o colores muy parecidos dentro del bloque, donde las frecuencias bajas son suficientes para recuperar los datos, pero conforme aumenta el factor de calidad este gradiente empieza a desaparecer y a tomar un solo valor o forma capas de bloques de tonalidad similar.

3.1.4.5. Criterios concluidos

Se pueden añadir, por tanto, los siguientes criterios para seleccionar una imagen que permita estudiar los fenómenos antes descritos:

- Tomaré ciertas imágenes que presenten cambios bruscos de color, con una cierta rapidez en la alternancia sus datos, de forma que sea esperable que se necesiten más bases para codificarla.
- Tomaré imágenes con el caso contrario, donde los cambios de color sean suaves siendo esperable esta vez que se necesiten menos bases para codificar la imagen, pero que en algún caso puedan generar efecto de contorneado.
- Tomaré imágenes donde ambos fenómenos se den en cierta proporción.
- Añadiré también alguna imagen que tenga detalles pero que éstos no impliquen un cambio de color brusco, para provocar que el error presentado no sea (hablando de forma hipotética) tan alto pero que sea igualmente perceptible.
- Sería interesante tomar alguna imagen donde los colores no fueran nada parecidos entre ellos, siendo vecinos, o que en pequeñas regiones de la imagen los bloques sean muy diferentes entre ellos. En la naturaleza esto no es tan común y uno de los motivos por los que JPEG funciona muy bien con imágenes fotográficas es este.

3.2. Selección del conjunto de imágenes

Aplicando los criterios establecidos en los apartados que preceden a este, voy a presentar las imágenes seleccionadas.

Cabe destacar antes de nada que todas ellas tienen un tamaño de 512x512 píxeles y que son BMP originales o TIFF transformados a BMP, de forma que no ha habido un proceso previo de compresión para ninguna de las imágenes. Además, todas ellas son imágenes de fotografía y en ningún caso han sido diseñadas artificialmente.

Las fuentes de estas imágenes pueden ser consultadas haciendo clic en las referencias proporcionadas y son mencionadas en el apartado de [Bibliografía](#).

La *Figura 1* muestra la primera imagen seleccionada. El motivo de esta elección es la suavidad de los cambios de color en el fondo (con probable efecto de contorneado en la experimentación) en la que no parece haber ningún gradiente, por lo que la cuantización de los bloques en esa zona no debería pixelarla mucho pero sí que se generarán bloques de exactamente el mismo color. Su entropía es de 7.374, que es, como ya veremos, una de las más bajas debido a la frecuencia de los colores, alta para unos pocos valores y baja para otros muchos.

Tiene relevancia además por los detalles del plumaje de los loros, que al ser muy finos es probable que sea una zona donde antes se empieza a percibir la pérdida de calidad y el aumento del error. Lo mismo ocurre con la piel de los loros en la cabeza, cerca del pico. Es esperable en estas regiones que se pierda detalle conforme aumente el factor de calidad, aunque quizás no se llegue a percibir del todo porque es algo en lo que el ojo humano es malo percibiendo.

También es interesante porque el cambio de color del fondo a los loros es bastante brusco, y es esperable que en los bloques 8x8 donde se encuentran esos bordes se recuperen peor por las ideas explicadas en el apartado de **Variabilidad**. Sobre todo, en la parte de los picos.

El error cuadrático medio no debería ser muy alto porque los píxeles dentro de los bloques de alta variabilidad van a ser bastante parecidos. Es decir, los detalles no están en zonas de cambios de color bruscos en los bloques implicados.



Figura 1. Imagen “loros” [6]

La *Figura 2* es interesante por el hecho de que presenta un color blanco prácticamente igual en muchas regiones del fondo de la imagen donde se encuentra una mezcla entre nieve y nubes, por lo que es esperable que se comprima bastante debido a que muchos de los coeficientes de los bloques ubicados ahí no serán necesarios, como con el fondo de la imagen de los loros.

Además, presenta sombras en las laderas de las montañas, con bastante detalle en las rocas, y el efecto producido haciendo zoom podría tener bastante interés, aunque desde cierta distancia visual puede que no se note la compresión producida, al igual que con la imagen de los loros.

También, aunque de forma limitada, se muestra texto en el caza, y servirá para poner a prueba la teoría de que el texto va a ser una de las primeras características donde se perciba pérdida de calidad en la compresión.

En cuanto al error producido al ser recuperada, no debería ser muy alto, pero la introducción de logos y texto con bordes muy pronunciados, sumado a la zona montañosa, que presenta más variabilidad con respecto al resto de zonas y píxeles bastante más diversos entre ellos dentro de cada bloque hará que el error sea ligeramente superior al de la imagen de los loros.

La entropía de la imagen es de 6.718, notablemente más baja que la anterior, ya que en la imagen predomina el color blanco y su probabilidad de aparición es muy alta con respecto al resto de colores.



Figura 2. Imagen “F-16”^[5]

La justificación de la imagen mostrada en la *Figura 3* es parecida a la de la *Figura 1*, aplicándolos esta vez a los detalles de los ladrillos del faro, las paredes de las casetas y posiblemente la cuerda amarilla. Se puede sumar a esto el hecho de que la barandilla del faro y el mirador provocan cambios de color bastante bruscos también, y probablemente, aunque de forma general se recupere bien la imagen, esas partes mostrarán una peor calidad visual, y probablemente sean las primeras cosas en las que se empiece a notar la pérdida de calidad.

El cielo, además, no está despejado y hay una buena alternancia entre nubes y cielo, que probablemente no afecte a la compresión porque la variabilidad de los bloques 8x8 parece

baja, pero es posible que propicien la aparición de gradientes y que se convierta en una de las primeras zonas afectadas por el aumento del tamaño del intervalo.

Siguiendo con la pared de la caseta creo que, ya que los ladrillos difícilmente se distinguen, la cuantización que se producirá en esa zona no va a ser realmente apreciable porque los valores ya se parecen mucho. Los tejados hacen mucho contraste con el cielo y será una de las zonas con más pérdida de calidad cualitativa.

Estimo que en lo referente al MSE, será más bajo que en la imagen del caza, pero ligeramente más alto que en la imagen de los loros, que era la más suave en su totalidad. El motivo es que en general, el número de zonas con cambios más bruscos de color es menor que en la imagen del caza y la mayor parte de los detalles, como pueden ser los ladrillos o las paredes, su color tienen un valor parecido y la cuantización va a mantenerlos en medida de lo posible.

La entropía de la imagen es de 7.506, más alta que en la de los loros, ya que la frecuencia de aparición de los colores parece ser más uniforme que en las dos imágenes anteriores.



Figura 3. Imagen “faro” [6]

En la *Figura 4* es interesante el hecho de que el cielo no es exactamente del mismo color y presenta diferentes gradientes de azul (a priori) en toda la región que ocupa, por eso es otro candidato perfecto para mostrar el efecto de contorneado generado con un factor de calidad alto.

Además, en la plataforma de despegue hay bastantes detalles y cambios bruscos de color entre hierros, luces y el cielo. En las formas que toma el humo hay mucho detalle también, y tomará importancia cuando se vea cómo le afecta el escalón generado por factor de calidad, porque como se ha explicado antes, los bloques que se encuentren en la aparición de los píxeles partícipes en estos detalles finos van a hacer que se pierda la forma hasta cierto punto y se distorsione.

Aunque existen gradientes de color en el cielo, hay partes dentro del mismo donde la variación es muy suave y mi hipótesis es que facilitará la compresión, ya en principio esto implicaría una baja variabilidad o, lo que es lo mismo, un número menos elevado de coeficientes no nulos para recuperar los bloques. Lo mismo ocurriría con la estructura de la nave espacial despegando, así como las luces fruto de la combustión que produce el despegue.

El error no creo que aumente mucho con respecto a la imagen anterior, ya que son parecidas en los aspectos que he tratado, y tengo la sensación de que la plataforma se va a ver mal con cualquier factor de calidad igual o por encima de 100, porque la cantidad de detalle en esa zona es bastante alta.

Su entropía es de 7.663, bastante parecida a la del faro.



Figura 4. Imagen “despegue” [7]

La siguiente imagen, mostrada en la *Figura 5*, es un clásico en el ámbito del procesamiento de imágenes. He escogido esta imagen porque el pelaje del mandril presenta una alta variabilidad y es un indicador, a priori, de que la imagen se va comprimir menos que el resto, puesto ocupa muchas de las regiones de la imagen. Lo mismo con los bigotes, que probablemente se entremezclen, debido nuevamente a que estos detalles es esperable que desaparezcan y se difuminen mucho conforme aumente el factor de calidad.

El resto de la cara tiene variaciones suaves de color excepto los bordes de la nariz y los ojos, donde es esperable que se pierda calidad, nuevamente, por los motivos explicados en el apartado de **Variabilidad**.

La entropía de esta imagen de un mandril es de 7.707, la más alta de toda la selección. Este valor hace bastante contraste con el obtenido con la imagen del caza, por ejemplo. Esta alta entropía se debe a la frecuencia de aparición de sus valores, sobre todo por el pelaje, donde parecen repartirse de forma bastante uniforme.

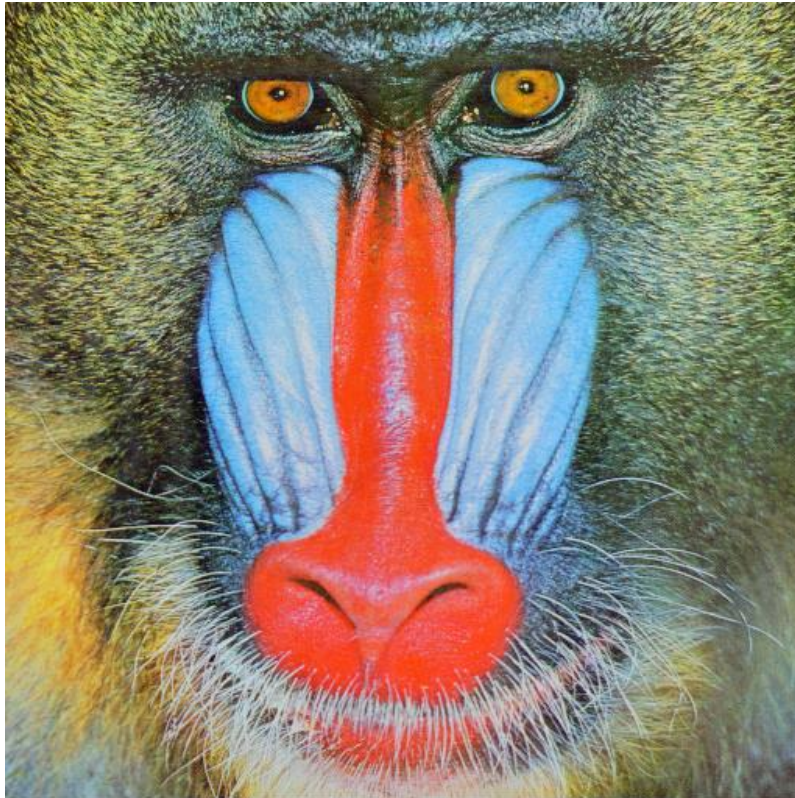


Figura 5. Imagen original “mandril”^[5]

Llegados a este punto, ya he cubierto imágenes con gradientes de color, imágenes con variaciones suaves de color, cambios bruscos, imágenes con más detalles, imágenes con menos detalles, y una imagen que incluye texto y sombras también. Esta última imagen, mostrada en la *Figura 6*, recoge todas las variaciones anteriores de alguna manera, incluyendo un poco de cada; alta variabilidad en las hojas de los árboles, cambios bruscos

de color en la casa y el coche (sombras incluidas), variaciones suaves en el cielo y en la hierba con posible aparición de gradientes, reflejos de luz, detalles en el tejado...

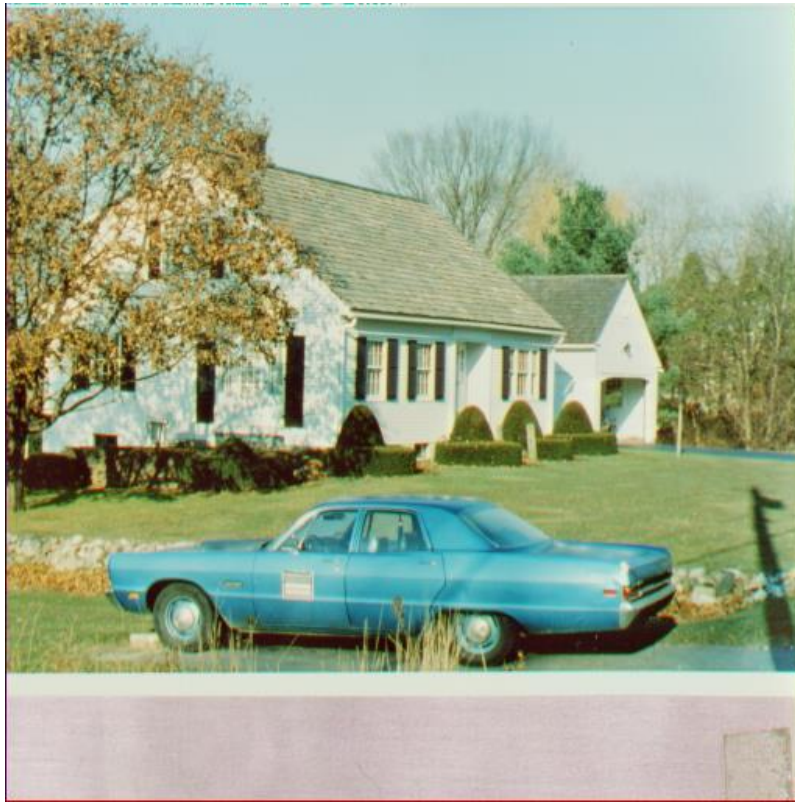


Figura 6. Imagen original “casa” [5]

La cantidad de elementos diferentes que forman la imagen limita un poco la especulación en este caso con respecto al ratio de compresión y al error, pero podría decirse que el error va a ser menor que en la imagen del mandril, pero mayor que en la del faro por poner un ejemplo y se van a lograr altas relaciones de compresión para todos los factores de calidad porque las zonas de más variabilidad son reducidas. Quizás se comprima menos para el factor de calidad más bajo, donde más coeficientes de altas frecuencias se recuperarán. Su entropía es de 7.416, un valor intermedio de todas las entropías del resto de imágenes.

3.3. Selección de valores del factor de calidad

Para seleccionar los factores de calidad que se utilizan a la hora de comprimir una imagen, se debe tener en cuenta lo siguiente:

- El valor mínimo para este factor de calidad es 1, y con este valor la imagen se comprime muy poco, debido a que los intervalos para los que se cuantizan los datos son pequeños.
- El valor recomendado por el estándar de JPEG es 100.

El objetivo es comprobar cómo afecta este valor a la relación de compresión, al error cuadrático medio y a la calidad perceptual final de la imagen. Como se está hablando de compresión y lo que interesa es ver su comportamiento, voy a descartar los valores que no comprimen o comprimen muy poco. Lo interesante es experimentar con valores que generen una relación de compresión alta, y ver hasta qué punto podría comprimirse una imagen sin pérdidas observables de calidad.

Un factor de calidad bajo haría que la matriz de cuantización de cada bloque 8x8 generado tenga más coeficientes no nulos, obteniendo bastante menos relación de compresión. Un factor de calidad demasiado grande aumentaría el ancho del intervalo a la hora de cuantizar, y más coeficientes se convertirían en nulos, consiguiendo una gran relación de compresión, pero aumentando mucho el error cuadrático medio y afectando bastante a la calidad de la imagen recuperada.

Por esta razón, el rango de valores podría empezar en mitad de lo recomendado por el estándar y el mínimo, es decir, en 50. Para este valor es esperable que la compresión de las imágenes comience a ser notable.

En cuanto al límite superior, se podría establecer en 2 o 3 veces el valor recomendado por el estándar. Mi hipótesis es que en algún punto alrededor de un factor de calidad de 200 o 300 el error comience a dispararse, el intervalo de cuantización sería excesivamente grande (lo que significa más coeficientes no nulos y más pérdida de información) y no merezca la pena seguir aumentando el valor del factor de calidad, puesto que difícilmente se va poder reducir más el tamaño del fichero comprimido. Por si esta hipótesis fuera errónea, incluiré el valor de 500 también, donde el error sí que debería dispararse de forma más evidente.

Además, los valores de los factores de calidad escogidos deben estar lo suficientemente separados entre ellos para poder observar diferencias notables entre las distintas imágenes. Sería interesante ver también valores cercanos a los del estándar, para ver si es justificable de alguna manera que el valor del estándar sea 100 y no otro.

Por todas estas razones, los factores de calidad que voy a escoger son los siguientes: 50, 75, 100, 150, 200, 300 y 500, aunque este último tiene interés para el análisis de los datos, más que con el relacionado con la percepción de la imagen.

3.4. Hipótesis generales y cuestiones a tratar durante la experimentación

En este punto se tratará de reunir todas aquellas hipótesis lanzadas en puntos anteriores y unas cuestiones que puedan ser resueltas durante la experimentación. Se definen entonces las siguientes hipótesis con respecto a la compresión y recuperación de las imágenes:

- Las imágenes que contengan texto o bordes pronunciados, se recuperarán peor y es esperable que se compriman menos. Probablemente no ocurra esto último porque los bloques donde se encuentre el contorno del texto serán muy pocos en comparación con el resto.

- Las imágenes con regiones donde el cambio de color sea gradual o muy parecido, es esperable que, con un factor de calidad relativamente alto, se produzca “contouring”.
- Las imágenes con más cambios de color bruscos es esperable que se compriman menos.
- Las que sus cambios sean más suaves se comprimirán más, pero serán en las que el error cualitativo se perciba con más anterioridad.
- Las imágenes que tengan bloques parecidos entre ellos y dentro de los mismos se comprimirán más y al revés.
- Las imágenes recuperadas que presenten un error mayor no se verán necesariamente peor. También es posible que imágenes recuperadas que tengan menos error con respecto a la original se vean peor.
- Un factor de calidad por encima de 200 aproximadamente hará que se perciba una reducción de calidad perceptual y cuantitativa (en formar de mayor error cuadrático medio).
- Un factor de calidad por debajo de 200 aproximadamente hará que no se perciba una pérdida de calidad sustancial de forma cualitativa, aunque el error sea igualmente alto.
- Las imágenes originales y las recuperadas se verán prácticamente idénticas siempre y cuando el factor de calidad no sea muy alto y no existan muchos cambios bruscos de color y/o bordes pronunciados.
- Las imágenes con alta variabilidad (y mayor número de coeficientes no nulos) se comprimirán menos y al revés.
- Es esperable que las imágenes comprimidas usando el compresor personalizado se compriman ligeramente más, puesto que se generan códigos adaptados a las frecuencias de los coeficientes transformados y cuantizados extraídos de la imagen.

Para finalizar, se pueden añadir también las siguientes preguntas, que sería interesante tratar de resolver en la experimentación:

- ¿Qué ayuda a comprimir más? ¿la variabilidad (coeficientes necesarios para recuperar la imagen), la longitud de los códigos, o ambos por igual?
- ¿En general, qué compresor sería mejor utilizar en función de sus resultados, el compresor ‘custom’ o ‘default’?

3.5. Procedimiento desarrollado durante la investigación

La estrategia seguida durante el desarrollo de la experimentación pasa, primeramente, por la aplicación de ambos compresores a todas las imágenes para todos los factores de calidad seleccionados, de forma que son recogidos todos sus valores de razón de compresión, error cuadrático medio, coeficientes no nulos, entropía y tiempo de ejecución.

Este proceso será automatizado con la ayuda de un script descrito en el [Anexo II](#). Los resultados obtenidos serán ubicados en forma de tablas, una por compresor para tener los datos separados y no crear tablas gigantes que no se puedan interpretar fácilmente.

Adicionalmente se mostrará una gráfica comparativa relacionando el error cuadrático medio con los ratios de compresión para todos los factores de calidad elegidos, para investigar cuánto distan las diferencias entre estos ratios de compresión unos de otros según el compresor.

El análisis se dividirá en un apartado por imagen y, una vez analizados los resultados arrojados para cada una, se realizará un análisis cualitativo, relacionado en la medida de lo posible con estos resultados. Para ayudarme en el análisis, mostraré ciertas regiones de las imágenes de forma ampliada, donde se puedan ver prácticamente los bloques cuantizados en las zonas más relevante e interesantes.

Los resultados serán discutidos individualmente por figura, o de forma conjunta con figuras obtenidas de la misma imagen que se esté comentando o con imágenes analizadas previamente, estableciendo paralelismos y hechos diferenciables según el contenido de las mismas.

4. Resultados experimentales

En este apartado, se realizará una discusión comparativa cuantitativa y cualitativa de los resultados obtenidos, examinando las similitudes y las diferencias sobre los valores de error cuadrático medio (MSE), relación de compresión (RC) en función de su factor de calidad. Estos valores serán generados para ambos compresores (y descompresores) también.

Cada imagen tiene un punto separado donde se comenta de forma individual, y conforme se realice el análisis de más imágenes, serán analizadas de forma conjunta entre ellas, explicando el porqué de los resultados que hayan sido obtenidos. Los tiempos de ejecución serán analizados en un apartado separado, junto con otras imágenes fuera de la selección para poder comparar por tamaño.

Finalmente, cabe añadir que primero se efectuará el análisis cuantitativo y después el cualitativo, de forma que, una vez se disponga de todos los datos sobre los parámetros establecidos, se puedan comparar con la percepción visual de las imágenes recuperadas y ver si cumplen lo esperado con respecto al análisis cuantitativo o no y por qué.

4.1. Imagen “loros”

Para la imagen de la [Figura 1](#), los resultados que fueron arrojados usando el compresor por defecto se muestran en la siguiente tabla:

Tabla de resultados usando compresión por defecto					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
loros.bmp	7.374	50	95.60	11.5108	49158
		75	96.56	16.0969	38953
		100	97.05	20.4269	33475
		150	97.66	29.3573	26682
		200	97.97	37.9326	22843
		300	98.32	55.5487	18639
		500	98.62	97.4884	14432

Hay dos resultados que destacan a simple vista. El primero de ellos, que obviaré en el resto del análisis puesto que va a ser común en todos los casos es que, aun usando el factor de calidad más bajo escogido, se logra una relación de compresión considerablemente significativa, en este caso del **95.6 %**, cumpliéndose así la hipótesis realizada sobre las imágenes con zonas de variación de color suave.

El segundo es que el error aumenta de forma casi lineal, como se observa en la *Figura 7* y, además, este error empieza siendo bastante pequeño. Esto es un indicador de que los coeficientes asociados a imágenes base de altas frecuencias son muy poco relevantes en esta imagen y, por tanto, sus bloques presentan una baja variabilidad de forma conjunta. Este hecho también explicaría las altas relaciones de compresión que se dan para todos los factores de calidad, lo quiere decir que la imagen, en general, muestra cambios suaves de color y el detalle de las plumas y la piel de los loros no parece ser suficiente para afectar a estas relaciones. Es esperable que, para un factor de calidad de 50, 75 o 100 no se aprecien diferencias notables entre imagen original y recuperada.

Otro hecho a destacar es que el error se dispara con el factor de calidad 300 y 500. El intervalo de cuantización en este caso sería demasiado grande y comienzan a perderse coeficientes asociados a imágenes base menos descartables, además de que cuanto más grande sea el intervalo, más alejados pueden estar los valores originales del valor medio de recuperación.

Es por esto que, pese a descartarse más coeficientes durante el proceso de cuantización y aumentar bastante el error, el número de coeficientes no nulos no parece haberse reducido de forma masiva con respecto al número de coeficientes no nulos con factor de calidad 150, ya que los saltos deben ser cada vez más grandes para abarcar más valores y cuantificarlos.

En cuanto a los valores de relación de compresión obtenidos, es razonable afirmar que se estancan alrededor del 97%, consiguiendo mejoras muy leves conforme aumenta el factor de calidad a partir de, curiosamente, el factor de calidad 100 recomendado por el estándar. ¿Por qué? En primera instancia, hay que dejar claro que una imagen no puede ser comprimida al 100%, si esto ocurriera, no existiría ni siquiera la imagen recuperada. En segunda instancia, aunque se usa el compresor por defecto (y por tanto no se guarda en la imagen comprimida información necesaria para su descompresión), la codificación de la imagen requiere de un tamaño en bytes, que tiene que asumirse y que siempre estará ahí si se quiere recuperar una imagen.

Teniendo esto en mente, se puede afirmar que existe un límite teórico en la relación de compresión obtenida, y que alcanzar ese límite es cada vez más complicado puesto que los intervalos creados con un cierto factor de calidad tendrían que ser cada vez más grandes para que todas las etiquetas resultantes de la cuantización tuvieran el mismo valor. Es por esto que, en algún punto, los coeficientes menos “energéticos” o importantes dejarán de perderse y se necesitan intervalos más agresivos para poder eliminar los coeficientes más relevantes, que son pocos, pero esenciales para poder lograr una recuperación aproximada de la imagen.

Esto explicaría la poca ganancia en ese porcentaje, en este caso, a partir del factor de calidad 100. Es esperable que este fenómeno se dé en el resto de imágenes y no insistiré sobre esta idea más adelante para no ser redundante, excepto si es realmente necesario.

Los resultados que han sido obtenidos usando el compresor con codificación ‘custom’ se muestran en la siguiente tabla:

Tabla de resultados usando compresión 'custom'					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
loros.bmp	7.374	50	95.60	11.5108	49158
		75	96.66	16.0969	38953
		100	97.21	20.4269	33475
		150	97.83	29.3573	26682
		200	98.22	37.9326	22843
		300	98.65	55.5487	18639
		500	99.04	97.4884	14432

Llama la atención el hecho de que el error cuadrático medio y el número de coeficientes no nulos para ambos compresores es el mismo, aparte de que los ratios de compresión son ligeramente más altos, como era de esperar puesto que la codificación se adapta a la imagen y no existe una codificación por defecto. Que el error y el número de coeficientes no nulos tiene que ver con que en ambos compresores se transforman los valores y se cuantizan los valores transformados de la misma manera, la diferencia entre ambos compresores se va a dar en las razones de compresión solamente, pues codifican de manera distinta, pero los seguiré mostrando en las siguientes imágenes también para apoyarme en ellos de cara al análisis.

Las diferencias entre los resultados producidos por ambos compresores para los 7 factores de calidad elegidos se muestran en la siguiente gráfica, la *Figura 7*.

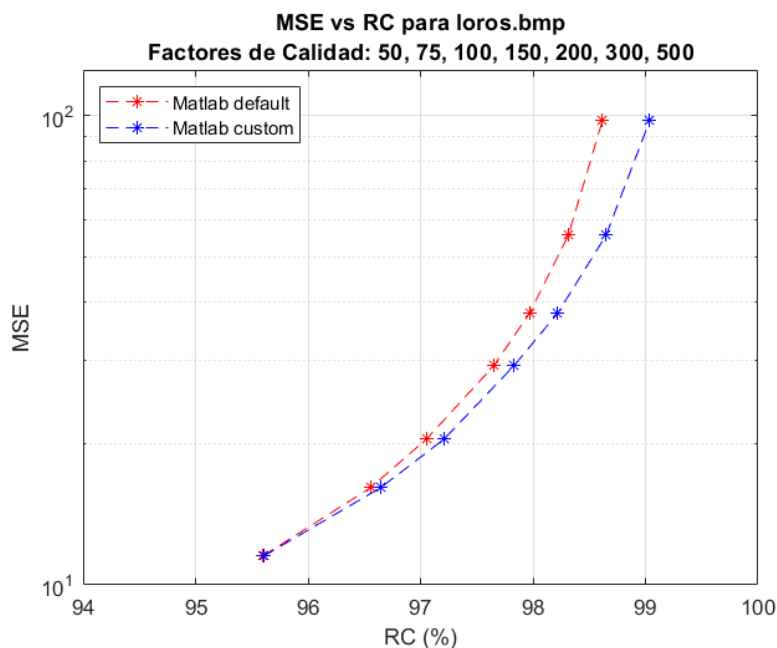


Figura 7. Comparación entre MSE y RC en imagen "loros.bmp"

Lo más destacable a comentar en esta gráfica es que se ve claramente cómo el compresor con codificación ‘custom’ obtiene unos ratios de compresión ligeramente más altos como se ha visto en su tabla y que conforme aumenta el factor de calidad, el compresor ‘custom’ mejora la relación de compresión con respecto al compresor ‘default’.

Este es un resultado muy interesante, que se da debido a que el compresor con codificación personalizada funciona mejor conforme la variedad de coeficientes no nulos va aumentando. Es decir, se sabe que la codificación por defecto, sobre todo de los valores AC, es tan larga como se establezca en la categoría en la que cae cada valor, independientemente de su número de apariciones en el bloque cuantizado.

En otras palabras, el número de etiquetas y categorías con valor 0 va a aumentar y ambos codificadores van a funcionar bien, pero es en los valores diferentes de 0 donde el codificador por defecto va a resultar ser más ineficiente, puesto que al no tener en cuenta las frecuencias de las apariciones de cada uno y codificar en función de su categoría únicamente, va a generar códigos más largos con el consecuente aumento de tamaño en la compresión, siendo el Huffman personalizado más óptimo en estos casos.

Para comenzar el análisis cualitativo con esta imagen, empezaré con la introducción de la imagen usando un factor de calidad de 100, como se muestra en la siguiente figura:



Figura 8. Imagen “loros” comprimida con factor de calidad 100

Por motivo nuevamente de redundancia, evitaré utilizar la imagen original otra vez. En su defecto, es posible hacerle clic para que te lleve a esta si se quieren observar las

diferencias, ya que se aprecian mejor cuando la imagen toma un cierto tamaño en el documento.

Para este factor de calidad 100, perceptualmente, no se aprecian diferencias significativas con respecto a la original. De hecho, sin hacer ningún tipo de zoom parecería ser la misma. La cuantización realizada con este primer factor de calidad afecta primeramente al fondo de la imagen, donde los colores marrón y verdes son muy parecidos entre ellos y se aprecia cierto pixelado. Quizá, fijándonos y siendo agudos, podríamos llegar a percibir cierta distorsión entre las cabezas de los loros y el fondo de la imagen. Este resultado es sorprendente ya que se logra un RC de 97.21%.

La pérdida de calidad comienza a ser notable cuando se aumenta el factor de calidad a un valor de 200, donde se da el efecto de contorneado en el fondo como fue teorizado en el apartado de **Hipótesis**, puesto que el fondo es suave y el intervalo de cuantización aumenta y más píxeles se recuperan como el mismo:



Figura 9. Imagen “loros” comprimida con factor de calidad 200

En el cuerpo de los loros y sus picos no parecen haber perdido tanta calidad como en el fondo, aunque el cambio brusco de color empieza a notarse, pero no de una manera exagerada.

El error empieza a ser importante en este factor de calidad, así que ampliaré ciertas zonas para tratar de mostrar más claramente la cuantización producida:



Figura 10. Comparación entre ampliación de la imagen original y comprimida con factor de calidad 200



Figura 11. Comparación entre ampliación de la imagen original y comprimida con factor de calidad 200

De cerca es más evidente la pérdida de calidad. El contorno del pico se ha difuminado con respecto a la imagen original y ciertos detalles dentro del mismo han desaparecido. El plumaje se ve bastante afectado, ya que son zonas de mayor detalle.

Lo mismo le ocurre al loro de la izquierda:



Figura 12. Comparación entre ampliación de imagen comprimida con factor de calidad 100 y comprimida con factor de calidad 200

En este caso, se muestra la diferencia de esta región entre la imagen recuperada para un factor de 100 y otra de 200. Se puede apreciar cómo en el fondo aparecen más bloques del mismo color en la imagen de la derecha y cómo la pérdida de detalle al utilizar menos coeficientes de las transformadas de altas frecuencias en cada bloque para recuperarla distorsiona el borde del pico y sus regiones más internas. En las marcas de la cara se pierde nitidez también, pero no es observable y además genera poco error, pues la diferencia entre píxeles es muy parecida después de la recuperación.

4.2. Imagen “F-16”

Para la imagen de la [Figura 2](#), los resultados que fueron obtenidos usando el compresor por defecto se muestran en la siguiente tabla:

Tabla de resultados usando compresión por defecto					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
f-16.bmp	6.718	50	94.39	18.8513	62359
		75	95.62	25.7020	49194
		100	96.28	32.2456	41773
		150	97.07	45.2831	32155
		200	97.50	57.8438	26322
		300	97.99	81.7435	19500
		500	98.41	125.8019	13538

Lo primero que se puede comentar de esta tabla es que las razones de compresión son significativamente menores que en la imagen anterior para todos los factores de calidad. Si nos fijamos, el número de coeficientes no nulos para el primer factor de calidad empieza siendo más alto también.

Esto es otro indicador de que esta segunda imagen contiene bloques que presentan una variabilidad más alta con respecto a la primera, ya que el hecho de que “sobrevivan” más de estos coeficientes quiere decir que más imágenes base son requeridas para recuperar los bloques. Este suceso cobra más importancia cuando se observa el error producido en la imagen recuperada, que para todos los factores es más grande que en la imagen anterior.

El error, al igual que en la imagen anterior, escala de forma casi lineal hasta que se dispara en 300 de nuevo, por los mismos motivos comentados.

Tabla de resultados usando compresión ‘custom’					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
f-16.bmp	6.718	50	94.32	18.8513	62359
		75	95.67	25.7020	49194
		100	96.41	32.2456	41773
		150	97.29	45.2831	32155
		200	97.69	57.8438	26322
		300	98.29	81.7435	19500
		500	98.82	125.8019	13538

En cuanto a los resultados mostrados con el compresor ‘custom’, es relevante comentar el hecho de que para un factor de calidad 50, el compresor con codificación por defecto consigue una mejora mayor, aunque muy pequeña.

Esto puede llegar a ocurrir porque existe un ‘overhead’ o carga adicional añadida al fichero comprimido por almacenar las tablas Huffman y demás información necesaria para la recuperación de la imagen. Descarto una codificación más óptima por parte del compresor por defecto (quizás cambie de idea más adelante en el análisis si los datos así lo indican) porque la codificación Huffman se adapta en función del número de apariciones de las categorías, y por tanto no puede ser peor que la codificación por defecto.

La gráfica comparativa entre ambos compresores es similar a la obtenida para la imagen de los loros, excepto por el hecho de que para el factor de calidad 50 es mejor el compresor por defecto.

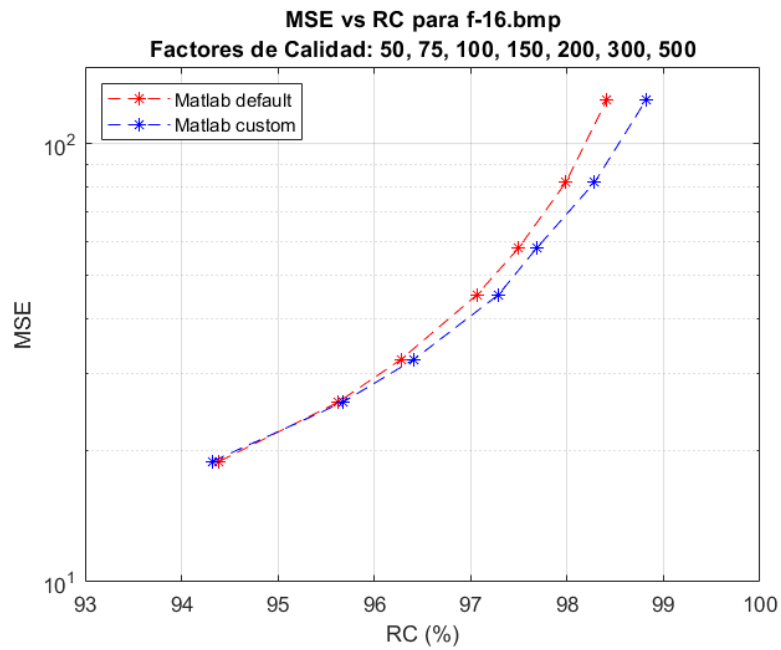


Figura 13. Comparación entre MSE y RC en imagen “f-16.bmp”

Más allá de los datos obtenidos y su explicación, quizás lo más importante es comprobar cómo percibe la imagen el ojo humano, que es el fin último de la compresión de imágenes.



Figura 14. Imagen “f-16” comprimida con factor de calidad 100

Usando el factor de calidad recomendado por el estándar, como en la imagen anterior difícilmente pueden observarse diferencias con respecto a la imagen original y la compresión se encuentra por encima del 96%, debido al hecho de que los coeficientes relativos a los valores de las crominancias son los primeros que se pierden, pero no son percibidos, ya que nuestro ojo es más sensible a la capa de luminancia. Forzaré un poco más la pérdida de calidad usando un factor de 300.



Figura 15. Imagen “f-16” comprimida con factor de calidad 300

En este caso, la pérdida de detalle es muy evidente. El logo de “U.S. AIR FORCE” sigue siendo legible, pero en los bordes se puede observar una distorsión con la chapa blanca del caza. Lo mismo ocurre en el cambio brusco de color que existe entre las alas rojas y el fondo blanco.

Este fondo también se ha pixelado bastante, aunque no se ha llegado a producir ‘contouring’. Lo sorprendente es que la cordillera y las sombras no parecen verse mal realmente.

El fenómeno que más clama al cielo es la pérdida de color en toda la imagen, encontrándose más apagada. Esto tiene que ver con lo comentado en párrafos anteriores sobre cómo los valores de crominancia van perdiéndose antes en la cuantización que los de luminancia.

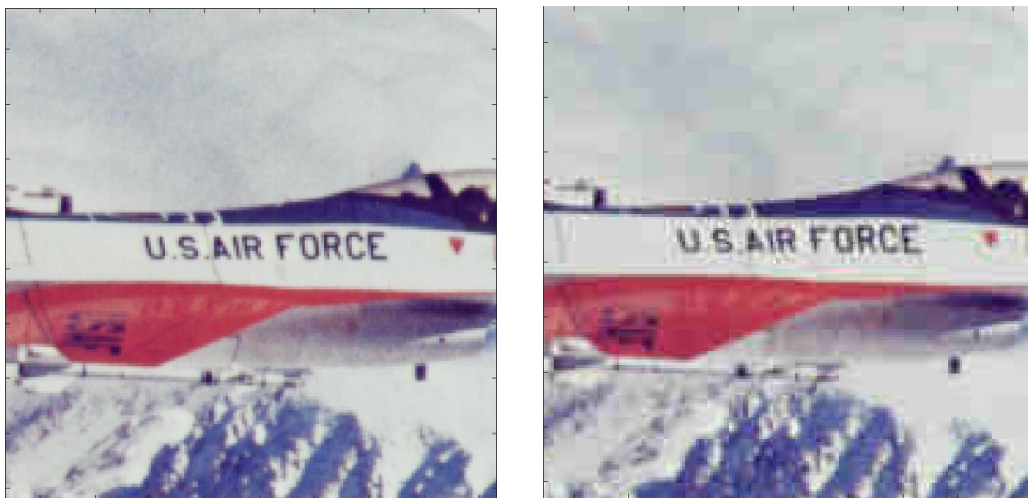


Figura 16. Comparación entre ampliación de la imagen original y comprimida con factor de calidad 200

Haciendo zoom sobre el logo del caza para un factor de calidad de 200 puede verse mejor la difuminación en los bordes, pero sigue sin notarse excesivamente, probablemente porque los bloques implicados en las letras ocupen gran parte del contorno y la variabilidad en esa zona siga siendo más alta de lo esperable, puesto que presenta colores muy diferentes. Observemos este fenómeno de cerca:

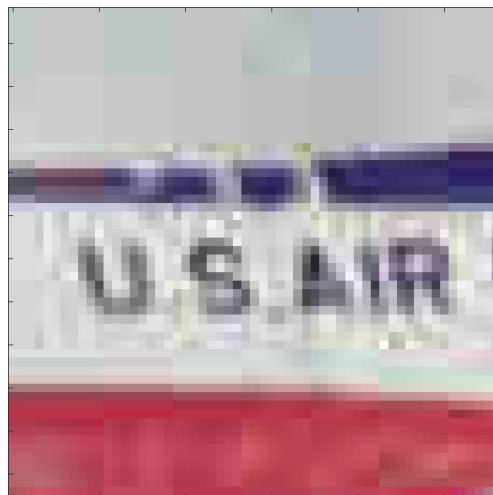


Figura 17. Ampliación de la imagen “f-16” sobre rótulos comprimidos con factor de calidad 200

Efectivamente, los bloques en esta región ocupan gran parte del contorno y, aunque se ha perdido mucho el detalle, un número importante de coeficientes de altas frecuencias se han mantenido.

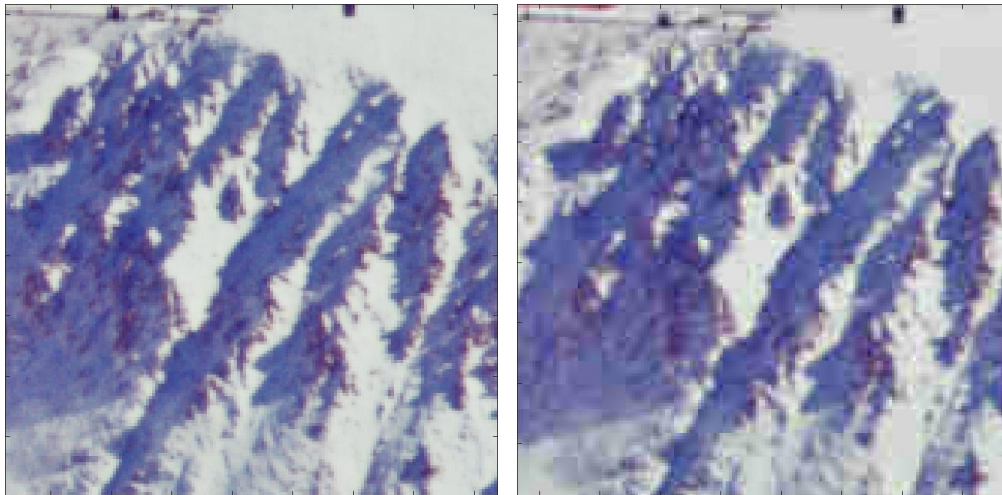


Figura 18. Comparativa entre montañas de “f-16” con imagen original y comprimida con factor de calidad 200

Para las laderas de las montañas sí que se nota bastante más el zoom, a cierta distancia el ojo humano (al menos el mío) no percibe la falta de detalle y de pixelación que se genera, que es bastante contundente. Esta es una región con bloques de alta variabilidad que producen parte del error de la imagen recuperada, pero este error no implica que se vea mucho peor la imagen recuperada.

Haciendo la diferencia entre píxeles de la imagen original y recuperada se puede observar perfectamente:



Figura 19. Diferencia absoluta de imágenes original y recuperada con factor de calidad 200 para “f-16”

4.3. Imagen “faro”

Para la imagen de la [Figura 3](#), los resultados que producidos usando el compresor por defecto se muestran, como viene siendo habitual, en la siguiente tabla:

Tabla de resultados usando compresión por defecto					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
faro.bmp	7.506	50	94.39	20.4035	62978
		75	95.60	29.1063	49807
		100	96.25	37.3254	42471
		150	97.02	53.8708	33388
		200	97.44	68.6850	28128
		300	97.92	98.5308	21881
		500	98.36	157.9859	15526

Las relaciones de compresión son muy similares a las observadas en el experimento anterior, así como el número de coeficientes no nulos para todos los factores de calidad propuestos, debido a que el detalle que presenta la imagen (y por consiguiente la alta variabilidad de muchos de sus bloques) es bastante alto.

La diferencia más notable es que para el factor de calidad de 300, el error se dispara un poco más con respecto a la imagen recuperada anterior. Esto se debe a que, en líneas generales, las diferencias entre píxeles recuperados y originales son un poco más grandes, en parte porque esta imagen tiene bordes más marcados y los detalles se encuentran en zonas (y bloques) de coloración más variada.

En la tabla con los resultados para el compresor personalizado no se arrojan resultados interesantes que no se hayan comentado anteriormente, excepto el hecho de que para el primer factor de calidad ocurre lo mismo que con la imagen del caza, donde se da el fenómeno que provoca que el compresor por defecto obtenga un mayor ratio de compresión.

Tabla de resultados usando compresión ‘custom’					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
faro.bmp	7.506	50	94.34	20.4035	62978
		75	95.69	29.1063	49807
		100	96.41	37.3254	42471
		150	97.30	53.8708	33388

		200	97.76	68.6850	28128
		300	98.25	98.5308	21881
		500	98.78	157.9859	15526

Este fenómeno se observa claramente en la siguiente gráfica, donde la diferencia entre relaciones de compresión para un MSE dado sigue siendo lineal, pero ya no escala tan rápidamente como en las dos imágenes anteriores.

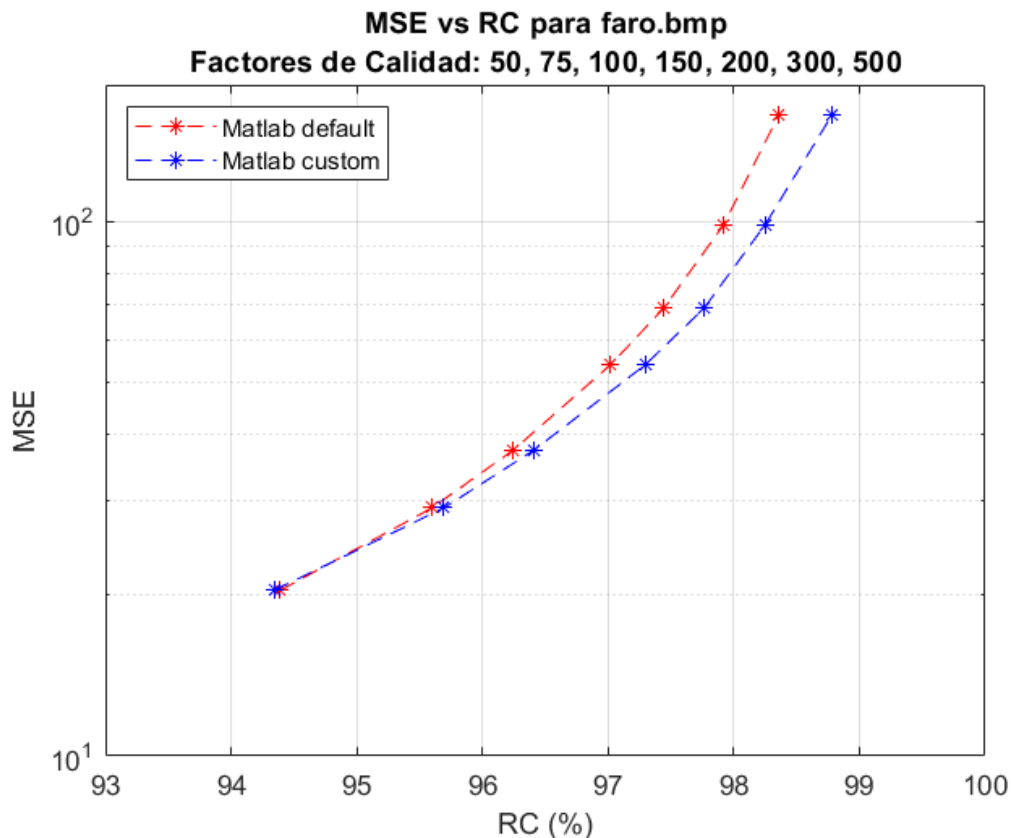


Figura 20. Comparación entre MSE y RC en imagen “faro.bmp”

Esto es debido a que el número de categorías codificadas distintas de 0, sobre todo para los valores AC, deben ser un mayor número o ser un mismo número, pero siendo sus frecuencias de aparición más parecidas a lo que la codificación por defecto espera. De esta manera, la codificación usando el primer compresor es bastante óptima con respecto a la codificación personalizada y no se queda tan atrás en ratios de compresión.

Para el análisis cualitativo comenzaré mostrando la imagen recuperada comprimida con un factor de calidad de 300 directamente, ya que no tiene mucho interés un factor de calidad más bajo donde todo se aprecie de manera parecida al resto de imágenes.

Queremos observar si realmente los detalles generan cambios de color menos bruscos y cómo se perciben visualmente.



Figura 21. Imagen “faro” comprimida con factor de calidad 300

Las zonas más afectadas en la recuperación son claramente la que involucra al faro, la pared de la caseta y el cielo nublado.

Este último elemento se ve significativamente mal y representa otro claro ejemplo de efecto de contorneado; donde había cambios suaves en los colores de la imagen original, se producen ahora cambios bruscos, debido a que en la región donde se encuentra el cielo existían cambios graduales de colores y estos cambios se han transformado en zonas “manchadas” con valores constantes.

La cuerda amarilla no termina de difuminarse porque en realidad la zona que ocupa es muy reducida y su resolución es baja.

Esta imagen contrasta con la del caza en el hecho de que presenta mayor error para este factor de calidad, y se ve notablemente peor también. La pared de la caseta ha perdido prácticamente cualquier detalle y los bordes entre el mirador, junto con las barandillas del faro y el cielo están más afectados que en cualquier de las imágenes anteriores, en primer lugar, porque el contorno en el caso de los hierros del faro es muy fino.

En la siguiente comparativa se aprecia mejor:

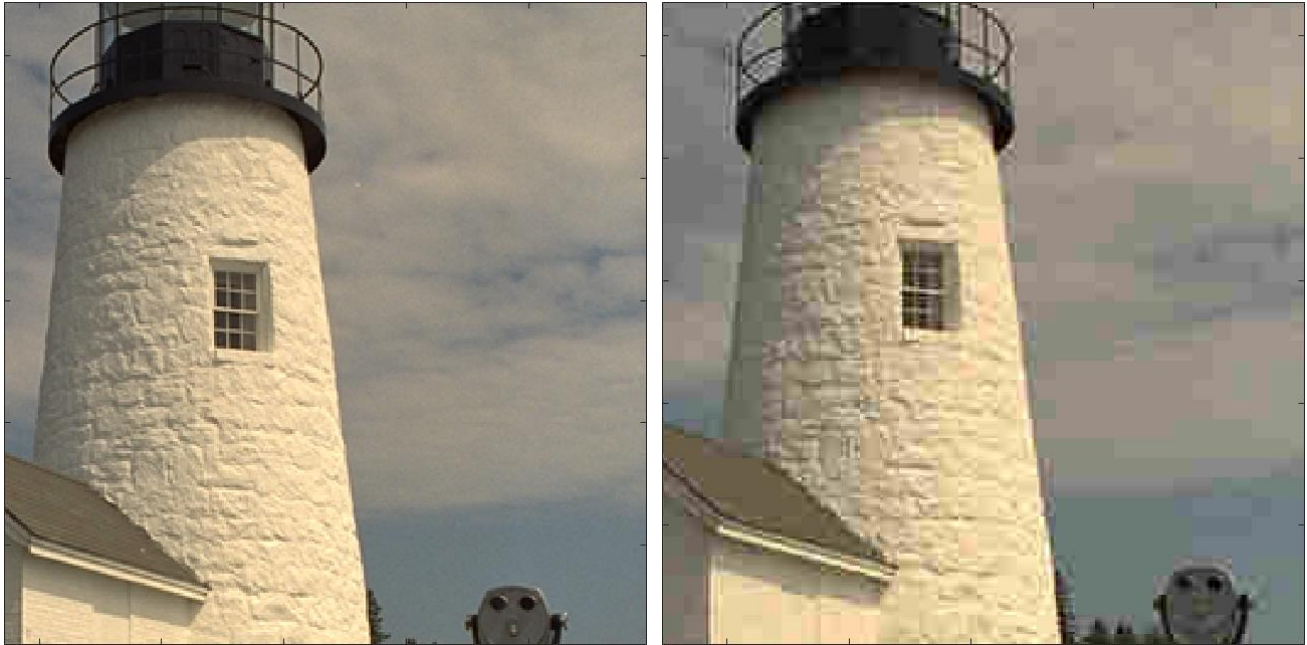


Figura 22. Comparativa entre faro con imagen original y comprimida con factor de calidad 300

Las zonas mencionadas junto con los ladrillos, están formadas por bloques de una variabilidad muy alta y nuevamente, al descartarse primero los coeficientes relativos a las imágenes de altas frecuencias, el detalle fino se pierde, y en este caso también provoca una deformación de la imagen. De hecho, produce una sensación extraña, como si el faro original estuviese “girado”. Este efecto se da con más fuerza en la ventana, donde directamente parece ser de otro faro distinto y las proporciones originales parecen haberse perdido.

Volviendo a la caseta, se vuelve a dar el efecto de contorneado, y se puede observar, comparándola con la imagen recuperada con un factor de calidad 100, que se ha perdido cualquier detalle incluso en los bordes, que se difuminan haciendo casi desaparecer el cambio...



Figura 23. Comparativa entre caseta comprimida con factor de calidad 100 y 300

... aunque esta diferencia parece ser mínima perceptualmente con el factor de calidad 100 observando la imagen original:



Figura 24. Caseta de la imagen “faro” original

En definitiva, el error mostrado en el análisis cuantitativo es peor que el obtenido para el F-16, y esta diferencia en el error sí que afecta a la percepción de su calidad, existiendo más cambios de color, y más bruscos. Además, las barandillas del faro son un elemento que parece afectar al algoritmo de forma general y es uno de los puntos donde primero se notan los efectos de la compresión.

4.4. Imagen “despegue”

Para la imagen de la [Figura 4](#), los resultados que obtenidos haciendo uso del compresor con codificación por defecto pueden ser observados en la siguiente tabla:

Tabla de resultados usando compresión por defecto					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
despegue.bmp	7.663	50	95.01	26.7945	57693
		75	96.19	34.9431	44649
		100	96.79	41.9086	37888
		150	97.47	55.0147	29719
		200	97.84	66.6993	25108
		300	98.25	89.2081	19704
		500	98.61	135.3451	14713

Existen dos resultados relevantes en esta tabla. El primero es que el error es alto para todos los factores de calidad y es el segundo más alto para el factor de calidad 500 que se ha obtenido hasta este momento. El segundo es que el número de coeficientes no nulos es alto, pero menos que en la imagen del faro o el caza. Esto indica que, en general, la imagen presenta más zonas con variaciones de color suave que las anteriores, pero tiene más cambios bruscos debido a detalles que la de los loros. El porqué del error tan grande con un factor de calidad 300 y 500 se verá más adelante.

Tabla de resultados usando compresión 'custom'					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
despegue.bmp	7.663	50	94.96	26.7945	57693
		75	96.29	34.9431	44649
		100	96.96	41.9086	37888
		150	97.64	55.0147	29719
		200	98.10	66.6993	25108
		300	98.60	89.2081	19704
		500	99.04	135.3451	14713

Con el codificador 'custom' vuelve a ser levemente peor para el factor de calidad 50 y las relaciones de compresión para un MSE dado aumentan de forma lineal con cada factor de calidad, como se muestra en la siguiente figura, hasta que empieza a dispararse con los factores de calidad 300 y 500.

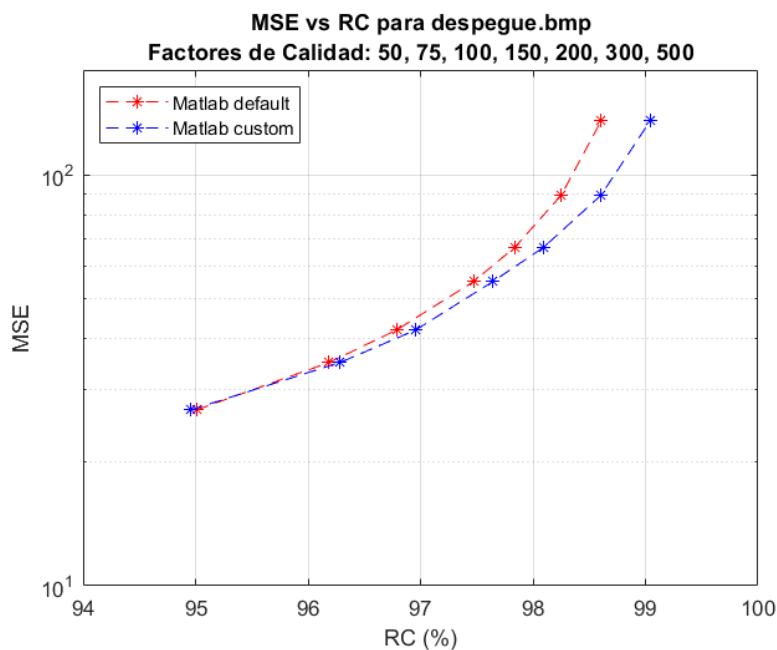


Figura 25. Comparación entre MSE y RC en imagen "despegue.bmp"

El análisis cuantitativo de la gráfica en este caso no va a añadir mucha nueva información a lo que ya se ha estudiado para otras imágenes. Veamos cuál es el comportamiento de los compresores con los elementos de la imagen.



Figura 26. Imagen “despegue” comprimida con un factor de calidad de 150

Para variar un poco con respecto a los ejemplos estudiados con anterioridad, he utilizado un factor de calidad de 150 esta vez. La calidad percibida es bastante buena, pero las nubes finas esparcidas por el cielo generan bastante pixelado y se crean pequeñas capas del mismo color provocadas por los gradientes que existen.

Las luces que se producen en la combustión del despegue no se ven afectadas de ninguna manera porque forman zonas con un color sólido, y la cuantización de los bloques que forman parte de esa región no le afecta. Es muy probable que el color de esta luz haya perdido fuerza, pero no es importante.

Si así es como se percibe con un factor de calidad de 150, con un factor de calidad de 100 no existirán graves diferencias. Es interesante puntualizar cómo el valor idóneo, desde el punto de vista de la percepción, en todos los casos parece ser el establecido por el estándar.

A continuación, mostraré la imagen con un factor de calidad de 300, para ver si se llegan a perder las formas del humo y las vigas de la plataforma se distorsionan como ha ocurrido con la imagen del faro.



Figura 27. Imagen “despegue” comprimida con factor de calidad 300

Se observa una pérdida general en la calidad del humo, pero no muy pronunciada. Esto tiene que ver con que el humo (así como podrían ser unas nubes espesas), por el hecho de ser humo, forma zonas ya difuminadas y los bloques que se encuentran ahí requieren de imágenes de bastante poca frecuencia para ser recuperados.

La pérdida es más notable en el cielo, donde parecía haber menos detalle. Se empieza a dar un patrón común en las imágenes, donde la pérdida de nitidez no es tan evidente y las regiones con gradientes de color son las más castigadas, junto con los cambios bruscos.

¿Por qué el error se disparaba con el factor 300 y 500 de calidad? En primer lugar, esto es explicable por el hecho de que en prácticamente toda la imagen se perciben diferencias notables; donde parecía no haber detalle, se tienen varios gradientes de color, y el resto son zonas de alta variabilidad, donde cualquier alteración va a producir diferencias con respecto a la imagen original.

En segundo lugar, las regiones con más detalles como pueden ser las vigas de la plataforma son muy pequeñas y va a existir una concentración de píxeles muy alterados, tanto en la capa de crominancias como luminancia.

Se puede ver esta situación haciendo zoom sobre las zonas comentadas, donde el contorno está prácticamente deformado y difuminado:



Figura 28. Comparativa entre plataforma de la imagen original y comprimida con factor de calidad 300

El error producido por la plataforma de despegue se ve más claramente haciendo la diferencia absoluta de los píxeles entre la imagen original y la recuperada:



Figura 29. Diferencia absoluta de píxeles entre imagen original y comprimida con factor de calidad 300

4.5. Imagen “mandril”

La siguiente imagen da un poco de frescura trayendo nuevos fenómenos diferenciales y únicos que analizar. Esta imagen se encuentra en la [Figura 5](#), y los resultados obtenidos para el compresor ‘default’ han sido los siguientes:

Tabla de resultados usando compresión por defecto					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
mandril.bmp	7.707	50	88.10	129.8602	138369
		75	91.03	167.4438	107072
		100	92.64	197.3341	88860
		150	94.57	245.7043	66331
		200	95.59	283.5412	53512
		300	96.72	348.0861	38835
		500	97.70	455.0125	25076

Como fue predicho en el apartado de [Selección de del conjunto de imágenes](#), la imagen de este mandril presenta la mayor variabilidad observada hasta el momento, debida en su mayor parte al pelaje de la piel y el bigote. Esta variabilidad se traduce en un número inusualmente alto de coeficientes no nulos con respecto al resto de imágenes para el factor de calidad más bajo seleccionado especialmente, pero también para el resto.

Anteriormente, para un factor de calidad de 100, este número de coeficientes eran del orden de 40000 de forma aproximada, y en este caso son más del doble. Es bueno recordar que este análisis tan sencillo es posible realizarlo debido a que todas las imágenes son del mismo tamaño y se puede realizar comparaciones muy fácilmente.

Se puede comprobar también que los ratios de compresión son notablemente más bajos en general, lográndose, para un factor de calidad de 100, un 92.64% de relación de compresión. Es la primera imagen que no pasa la barrera del 90% para el factor de calidad más pequeño.

El error es demasiado grande en comparación nuevamente con lo que se estaba analizando de forma habitual. Hagamos una comparativa de estos resultados con la segunda imagen con más error cuadrático medio observado, el faro en este caso, con la tabla a continuación.

Se puede ver que el MSE es entre 4 y 6 veces más alto para todos los factores de calidad y el número de coeficientes no nulos es más del doble para los factores de calidad más conservadores, (recordemos, teniendo el mismo tamaño).

Tabla de resultados comparativa entre “mandril” y “faro”				
	Mandrill		Faro	
Factor de calidad	Error cuadrático medio	Coefficientes no nulos	Error cuadrático medio	Coefficientes no nulos
50	129.8602	138369	20.4035	62978
75	167.4438	107072	29.1063	49807
100	197.3341	88860	37.3254	42471
150	245.7043	66331	53.8708	33388
200	283.5412	53512	68.6850	28128
300	348.0861	38835	98.5308	21881
500	455.0125	25076	157.9859	15526

También destaca el hecho de que el error cuadrático medio no termina de dispararse de forma exponencial como ocurría en las imágenes que preceden a esta, y se verá de forma más gráfica en la Figura 35.

¿A qué se debe este MSE? La [Figura 5](#), como se ha comentado, presenta muchísimos detalles en el pelaje, por lo que la concentración de píxeles diferentes unos de otros en cada bloque ubicado en estas zonas es altísima, como puede verse en las siguientes figuras comparativas:



Figura 30. Comparativa ampliada entre “mandril” original y comprimida con factor de calidad 200

Existen muchos bloques que requerirán de coeficientes de altas frecuencias para recuperarse completamente. Estas imágenes que he mostrado están en el espacio de color RGB, pero a nivel de capas en el espacio YCbCr en el que han sido transformados los valores, estos han sido cuantizados en las 3, y en conjunto, como en las transformadas los

coeficientes de altas frecuencias eran más importantes de lo normal el error es muy alto, puesto que se han descartado muchos de ellos durante el proceso de cuantización, ensanchando las diferencias entre el valor medio recuperado tras la descuantización, que además, parece estar más lejos de los valores reales que con el resto de imágenes.

Es por esto que en la imagen recuperada no parece apreciarse bien el error producido, en primer lugar. Además, como se explica en el [Anexo I](#), los primeros valores que se pierden se encuentran en las capas de crominancia. Esto está basado en el hecho de que el ojo humano es muy malo percibiendo detalles en estas capas y los coeficientes de altas frecuencias que se encuentran en ellas pueden ser descartados al cuantizar sin afectar a la calidad percibida. JPEG se aprovecha de este punto débil en nuestras capacidades.

En la siguiente imagen se pueden apreciar mejor los puntos que he comentado donde el error es más significativo:

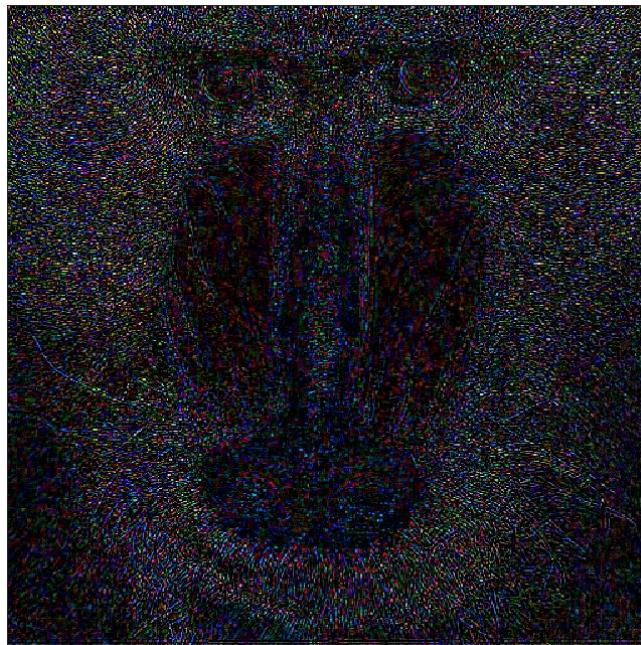


Figura 31. Diferencia absoluta entre imagen original y comprimida con factor de calidad 300

Que la imagen se comprima peor tiene que ver con lo explicado en párrafos anteriores también. Si se tienen más coeficientes no nulos en cada bloque (y diferentes entre ellos), se generan códigos de mayor longitud, o simplemente una mayor cantidad, aunque su longitud sea corta.

Mostraré en la siguiente figura la imagen recuperada con una compresión de un factor de calidad de 100, para ver si este MSE se traduce en una pérdida de calidad excesiva con respecto a lo que se ha visto anteriormente, pero nos llevamos una sorpresa:

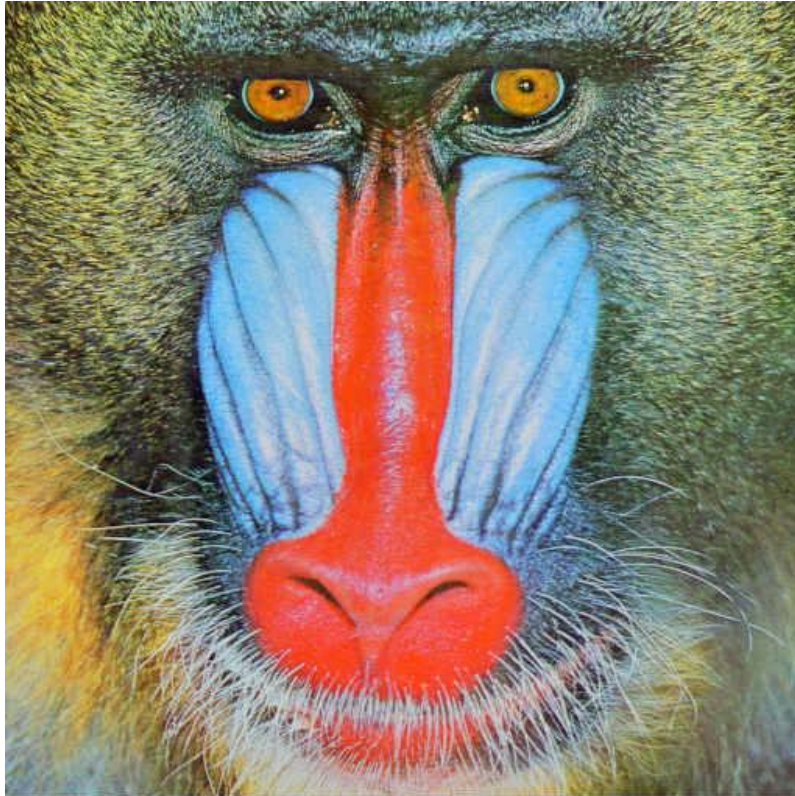


Figura 32. Imagen “mandril” comprimida con factor de calidad 100

El error producido no solamente no se percibe, sino que ambas imágenes, original y recuperada, son prácticamente indiferenciables una de otra. ¿Ocurre lo mismo con un factor de calidad mayor? Comprobémoslo con el máximo seleccionado, de 300:

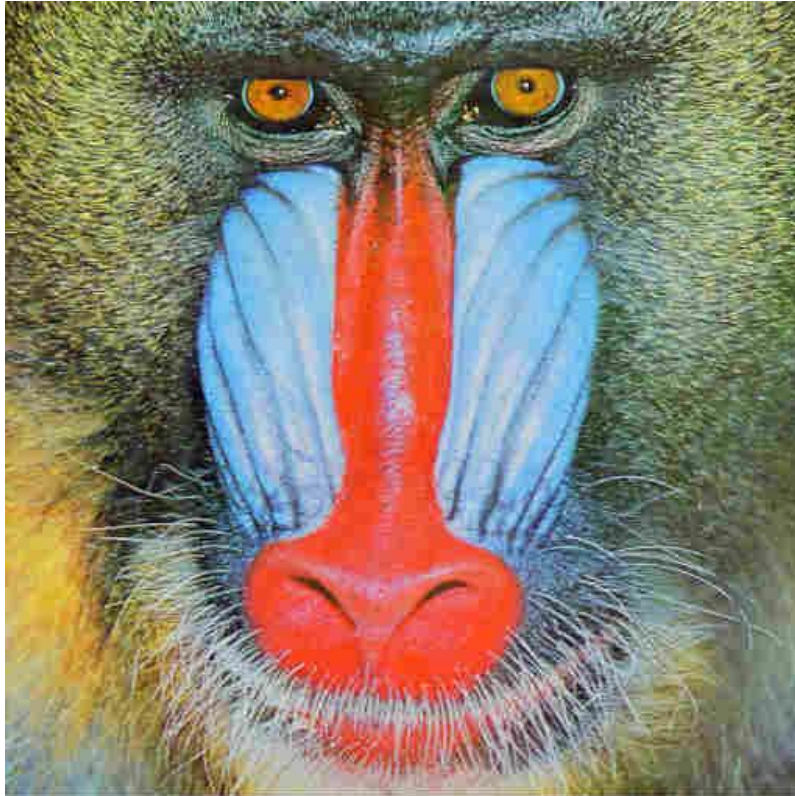


Figura 33. Imagen “mandril” comprimida con factor de calidad 300

En este caso, la pérdida de calidad se hace más evidente, pero poco tiene que ver con las deformaciones que se producían con el resto de imágenes. Lo más perceptible es la pérdida en la intensidad del color, la imagen se encuentra “apagada”, pero los detalles no se han perdido tanto como era esperable.

Este último fenómeno es debido a que el número de coeficientes no nulos es muy alto y se mantienen gran parte de los coeficientes de altas frecuencias que se produjeron en el proceso de transformación de los datos y los coeficientes de altas frecuencias de las crominancias son los que han desaparecido antes.

Quizás haciendo zoom de nuevo se pueda observar la pérdida de calidad cuantitativa sugerida en la tabla de resultados:



Figura 34. Comparativa ampliada entre imagen original y comprimida con factor de calidad de 200

Si nos fijamos en los bigotes, con un factor de calidad de 200, se aprecia claramente el cambio, pero desde la distancia, viendo toda la imagen a la vez, el cambio es imperceptible. En esto reside la “magia” del algoritmo JPEG.

Los resultados de todos estos experimentos dan a entender que JPEG funciona muy bien con aquellos detalles que no se encuentran sobre un fondo de variaciones suaves de tonalidad y color, funcionando mal en estos casos ya que los fallos en la recuperación de los bordes se hacen más evidentes.

Aún quedan por comentar los datos obtenidos para el compresor con codificación ‘custom’. Veamos la tabla:

Tabla de resultados usando compresión ‘custom’					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
mandril.bmp	7.707	50	87.76	129.8602	138369
		75	90.87	167.4438	107072
		100	92.57	197.3341	88860
		150	94.65	245.7043	66331
		200	95.67	283.5412	53512
		300	97.01	348.0861	38835
		500	98.08	455.0125	25076

La observación más relevante que puede ponerse en valor es que los ratios de compresión empeoran hasta el factor de calidad 150 con respecto al compresor con codificación por defecto, como puede comprobarse en la gráfica:

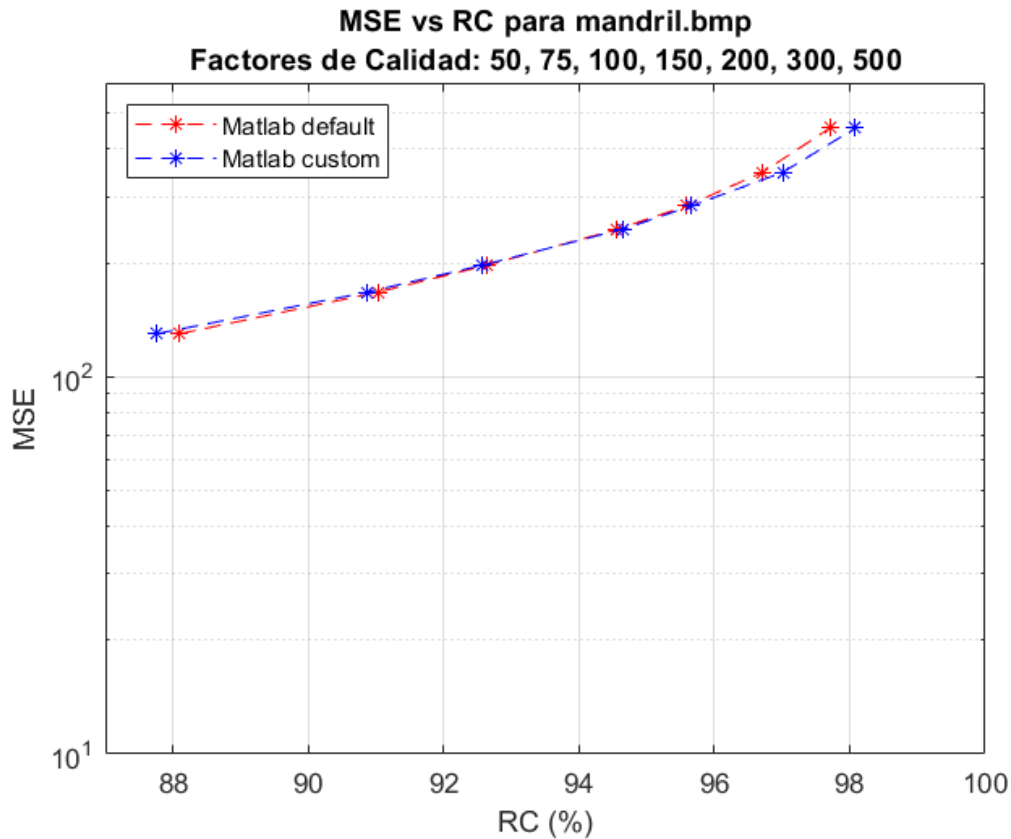


Figura 35. Comparación entre MSE y RC en imagen “mandril.bmp”

Este efecto puede ser explicado con el ‘overhead’ mencionado en imágenes anteriores.

Además, como se ha comentado al comienzo del análisis, el error cuadrático medio no se termina de disparar con un factor de calidad de 300, en parte porque ya es bastante grande de por sí. Va a ser necesario un descarte de coeficientes durante el proceso de cuantización más agresivo que el que se obtiene en estas circunstancias, lo que parece indicar que, para que en la recuperación de una imagen se dispare el error de forma “temprana” según el factor de calidad, los bloques de ésta no deben tener que necesitar coeficientes de muy altas frecuencias para poder ser recuperados.

4.6. Imagen “casa”

Llegamos a la última imagen seleccionada para su análisis. Debe ser recordado que esta era la imagen más ‘híbrida’, donde muchos de los problemas que debían hacerse cargo los compresores van a darse de forma casi equitativa.

Los primeros resultados obtenidos para esta imagen con el compresor por defecto son los siguientes:

Tabla de resultados usando compresión por defecto					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
casa.bmp	7.416	50	92.65	34.7731	83231
		75	94.26	46.5482	66591
		100	95.17	57.4348	56779
		150	96.28	77.9352	44096
		200	96.89	95.4299	36662
		300	97.59	130.5832	27530
		500	98.21	198.3331	18896

El MSE es más alto de lo normal con respecto al resto de resultados, pero bastante inferior a lo que se encontraba con la foto del mandril, y el número de coeficientes no nulos es también alto con respecto a la media. Esto indica una variabilidad de los bloques de la imagen relativamente alta.

Es por ello que los ratios de compresión para los primeros factores de calidad son más reducidos, pero se recuperan rápidamente conforme se van descartando coeficientes. Esto podría explicarse porque las zonas más ‘conflictivas’ en términos de variabilidad pierden rápidamente los coeficientes de altas frecuencias porque sus valores se encuentran muy cerca y pasan a formar parte del mismo intervalo.

Tabla de resultados usando compresión ‘custom’					
Imagen	Entropía de la imagen	Factor de calidad	Relación de compresión (%)	Error cuadrático medio	Coefficientes no nulos
casa.bmp	7.416	50	92.47	34.7731	83231
		75	94.23	46.5482	66591
		100	95.21	57.4348	56779
		150	96.47	77.9352	44096
		200	97.15	95.4299	36662
		300	97.87	130.5832	27530
		500	98.62	198.3331	18896

Destaca otra vez que el compresor ‘custom’ empeora para los primeros factores de calidad. Dentro del análisis cuantitativo no se encuentran más resultados interesantes, ya que el resto de fenómenos han sido explicados previamente.

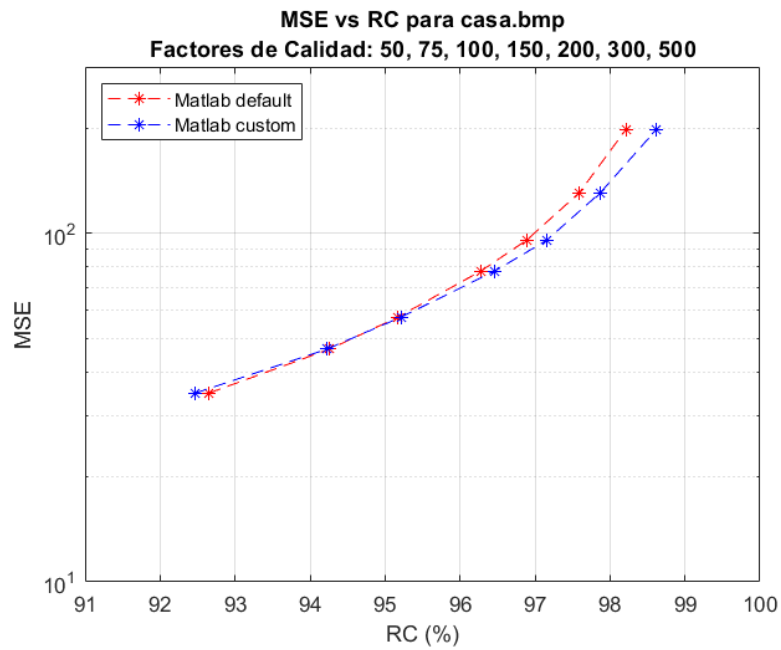


Figura 36. Comparación entre MSE y RC en imagen “casa.bmp”

Veamos cómo se recupera la imagen para un factor de calidad de 100, como se muestra en la siguiente figura:

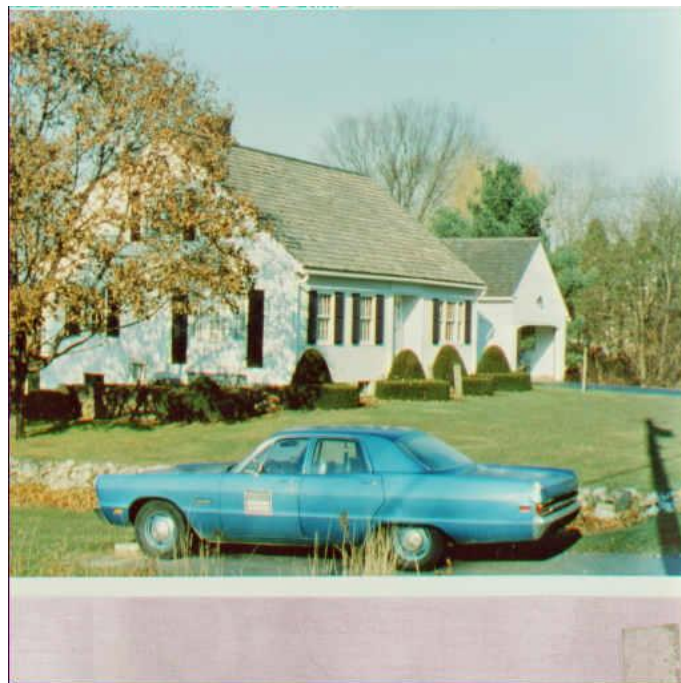


Figura 37. Imagen “casa” comprimida con factor de calidad 100

Viene siendo una constante que para un factor de calidad 100 la imagen, por mucho problemas que pueda presentar una imagen, se recupera muy decentemente visualmente manteniendo unas relaciones de compresión por encima del 95%. Está claro que el factor de calidad del estándar está bien estudiado y funciona genial.

Las primeras zonas donde debería haberse percibido errores en la recuperación, como pueden ser las hojas del árbol, las ramas del árbol del fondo o los bordes creados por el tejado no existen imperfecciones detectables por el ojo.

Veamos cómo cambia la imagen recuperada cuando se le aplica un factor de calidad de 300.

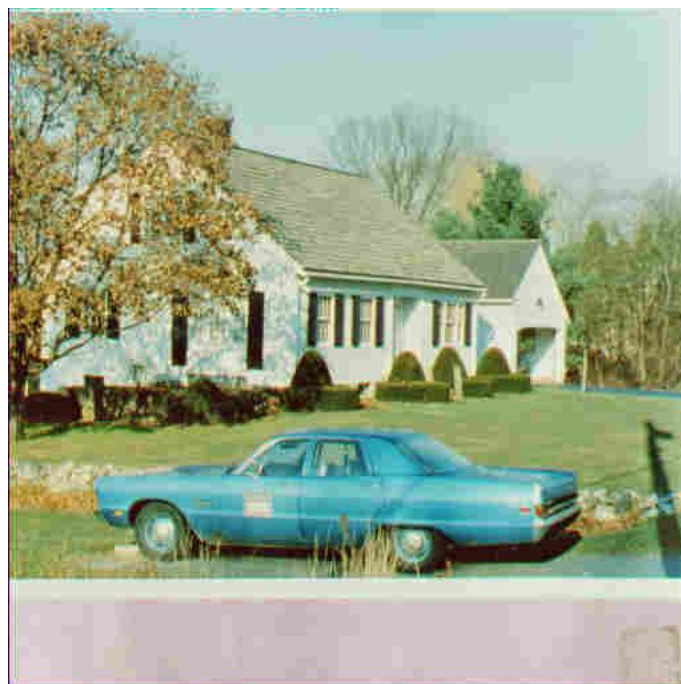


Figura 38. Imagen “casa” comprimida con factor de calidad 300

Los árboles del fondo se han difuminado muchísimo y los colores de los bloques que se encuentran en esa zona se han entremezclado y, junto con las hojas del árbol que está al lado de la casa, son los elementos causantes de la mayor parte del MSE obtenido.

Se da además un poco de efecto de contorneado en la parte de debajo de la fotografía escaneada debido a los cambios suaves en el color casi formando un gradiente.

Las sombras, sin embargo, se han recuperado bastante mejor de lo esperado, aunque generan cambios bruscos, sobre todo la del coche y la del poste. Haré zoom sobre estas zonas conflictivas para tratar de explicar el comportamiento con estas sombras.



Figura 39. Comparativa ampliada entre imagen “casa” original y comprimida con factor de calidad 300

Los bloques cuantizados recuperados se pueden diferenciar claramente en la imagen de la derecha. Como en las regiones de alta variabilidad que se han analizado en puntos anteriores, el detalle que se podía recuperar manteniendo los coeficientes de altas frecuencias ha desaparecido en su mayor parte durante el proceso de cuantización.

En el tejado también se aprecia bastante bien la pérdida de calidad, no debida a cambios bruscos sino a todo lo contrario. Los detalles de las tejas son pocos, pero suficientes para percibir que ya no se ven igual que antes y se entremezclan bastante. En el cielo, sin embargo, al no existir ningún gradiente y ser de un color prácticamente sólido, la pérdida de calidad visual es inexistente desde cierta distancia, ya que sus píxeles se recuperan en su mayoría como el mismo color.



Figura 40. Comparativa ampliada entre imagen original y comprimida con factor de calidad 300

En la zona inferior derecha de la imagen, las sombras se recuperan bastante bien porque los bloques que contienen el contorneado del borde de estas son de muy baja variabilidad si nos fijamos bien, debido a que la sombra cae de manera casi perpendicular en los bloques (lo que quiere decir que se requieren pocas imágenes base de bajas frecuencias para recuperarlos) en el caso del poste, y horizontal en el caso del coche. Además, la zona donde se proyecta es el jardín, que ya de por sí contiene poco detalle, al contrario de lo que ocurriría con la imagen del caza con las montañas.

4.7. Sobre la influencia de la entropía de la imagen

Para confirmar que de verdad la entropía de la imagen establecida en este documento no tiene relación con la longitud de los códigos con los que se comprime, he realizado la siguiente tabla. Es algo evidente, pero quería hacer énfasis en ello, ya que no debe confundirse esta entropía con la entropía sobre los coeficientes cuantizados.

Razones de compresión para un factor de calidad de 100 y entropía de la imagen		
Imagen	RC (%)	Entropía
loros.bmp	97.05	7.374
f-16.bmp	96.41	6.718
faro.bmp	96.25	7.506
despegue.bmp	96.79	7.663
mandril.bmp	92.64	7.707
casa.bmp	95.17	7.416

Se puede comprobar que las imágenes con entropía más baja no se comprimen más que otras con la entropía más alta y viceversa, como puede ser entre la imagen de los loros y el caza F-16.

5. Parte opcional I: Análisis de los tiempos de ejecución

Como parámetros adicionales para analizar, he incluido este apartado relativo a los tiempos de ejecución de la compresión y descompresión de cada imagen. Cabe resaltar que en la fase de análisis de resultados se incluyó la medida de coeficientes no nulos totales dentro de las matrices de cuantización resultantes en cada bloque y su integración fue parte del análisis básico de la práctica.

La forma de proceder para este análisis será parecida a lo que se ha realizado hasta el momento, y pasa primeramente por mostrar los datos obtenidos sobre los tiempos de ejecución para cada imagen, codificación y factor de calidad.

	loros.bmp		f-16.bmp		faro.bmp	
Factor de calidad	Tiempo ejecución (s) dflt	Tiempo ejecución (s) cstm	Tiempo ejecución (s) dflt	Tiempo ejecución (s) cstm	Tiempo ejecución (s) dflt	Tiempo ejecución (s) cstm
50	7.08	6.23	9.33	6.89	6.67	6.72
75	6.98	5.36	7.89	6.44	6.33	6.08
100	5.69	5.33	5.81	6.03	5.80	5.70
150	5.17	4.92	5.52	5.36	5.38	5.45
200	5.28	4.91	5.75	5.53	5.30	5.22
300	4.67	4.91	5.39	4.89	5.33	5.17
500	4.84	5.05	4.94	4.78	4.80	4.69

	despegue.bmp		mandril.bmp		casa.bmp	
Factor de calidad	Tiempo ejecución (s) dflt	Tiempo ejecución (s) cstm	Tiempo ejecución (s) dflt	Tiempo ejecución (s) cstm	Tiempo ejecución (s) dflt	Tiempo ejecución (s) cstm
50	6.95	6.91	10.30	11.31	7.67	7.70
75	5.94	6.53	8.89	9.23	6.98	7.05
100	5.52	6.38	7.86	8.11	6.56	6.58
150	5.28	5.72	7.13	6.81	6.08	5.95
200	5.06	5.33	6.91	6.33	5.95	5.63
300	4.92	5.22	6.08	5.91	5.63	5.25
500	4.91	4.95	5.44	5.28	5.28	5.23

El hecho que más destaca de un vistazo es que los tiempos de ejecución para el compresor 'custom' son algo más bajos que los tiempos en el compresor por defecto. Además, los tiempos para ambos se reducen conforme el factor de calidad va subiendo (debido a que los saltos de cuantización son más grandes y se codifican menos datos cada vez), existiendo una diferencia de hasta unos 6 segundos para los tiempos de ejecución

‘custom’ en la imagen del mandril, aunque para el factor de calidad 100 los tiempos son muy parecidos.

La diferencia de estos tiempos parece agudizarse cuando el factor de calidad es más pequeño.

Vamos a ver, apoyándome en un par de gráficas (no tiene mucho interés ver las 6, sino las más representativas), cómo y por qué se comportan estos valores de forma más visual. Estos valores de tiempo de ejecución serán relacionados con las razones de compresión obtenidas y mostradas durante el análisis experimental básico, ya que entenderé que una imagen va a tardar más en comprimirse cuantos más valores se deban codificar, y si se requieren codificar muchos valores es esperable que la compresión obtenida sea menor que en el caso contrario.

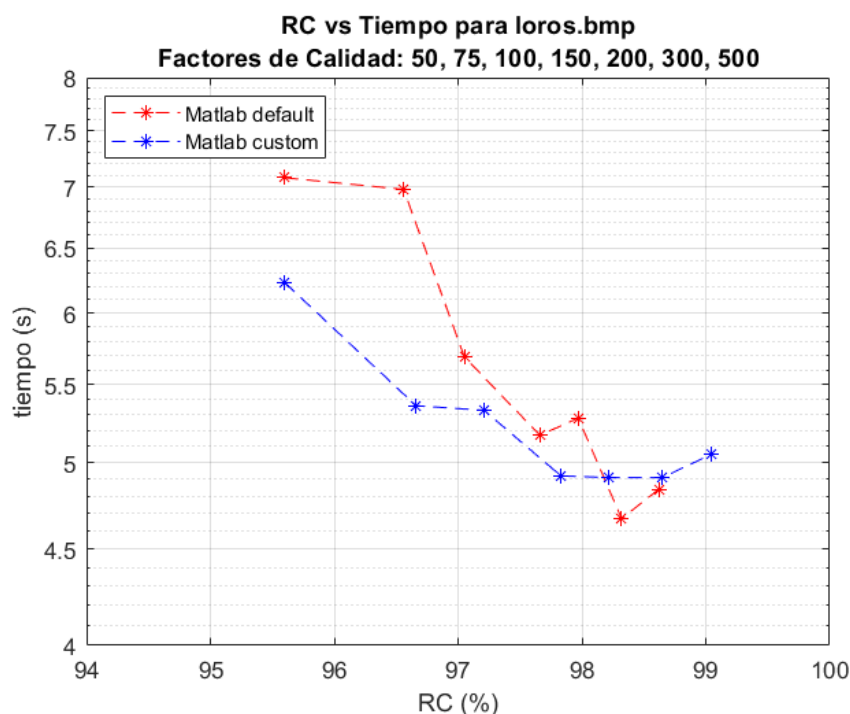


Figura 41. Comparación de tiempos de ejecución y RC en “loros” para todos los factores de calidad seleccionados

La elección de estas gráficas está basada en los ratios de compresión obtenidos para las imágenes. Los más altos pertenecen a la imagen de los loros y los más bajos a la del mandril, así que usaré estas mismas imágenes para ilustrar los datos de las tablas, ya que en el resto pasan fenómenos similares, en distinta proporción. En otras palabras, he tomado los datos extremos para resaltar los fenómenos interesantes.

Lo primero que destaca en la primera gráfica es el hecho de que el compresor ‘custom’ tarda menos tiempo en comprimir y descomprimir que el ‘default’. Esto podría llegar a sorprender, ya que los códigos en el compresor ‘default’ vienen por defecto y en el ‘custom’ deben construirse. ¿Qué está ocurriendo entonces?

Esta diferencia de tiempos puede ser explicada de diferentes maneras y es muy difícil atribuírsela a una de ellas en particular:

- Es posible que la construcción de códigos Huffman no sea tan lenta cuando el número de etiquetas distintas de 0 a codificar no es muy alto e iguale prácticamente en tiempos a la codificación por defecto (esto mejoraría si además estas etiquetas son iguales).
- Las tablas HUFF y BITS no deben ser reconstruidas en el compresor 'custom' debido a que ya se escriben en el fichero comprimido. En el compresor 'default' tampoco, pero deben pasar por un proceso de conversión previo durante la codificación y la decodificación.
- La función 'hex2vec' que transforma matrices de números en hexadecimal a enteros parece añadir tiempo extra a todo lo rápida que podría ser la codificación y decodificación.
- Podría darse la (remota) casualidad de que las tablas RLE en 'default' fueran más grandes que las construidas en 'custom' si en las matrices de los bloques existe una alternancia entre ceros y valores iguales (siendo el total de valores distintos un número bajo) duradera, tardando más su codificación.
- Tiempos de ejecución poco fiables debido a que no se están midiendo en un sistema totalmente aislado y es probable que Windows dé más prioridad a otros procesos corriendo durante alguna de las ejecuciones y se tengan unos datos sesgados. He intentado mitigar esto último lanzando cada imagen más de una vez seguida para cachear los bloques en memoria.

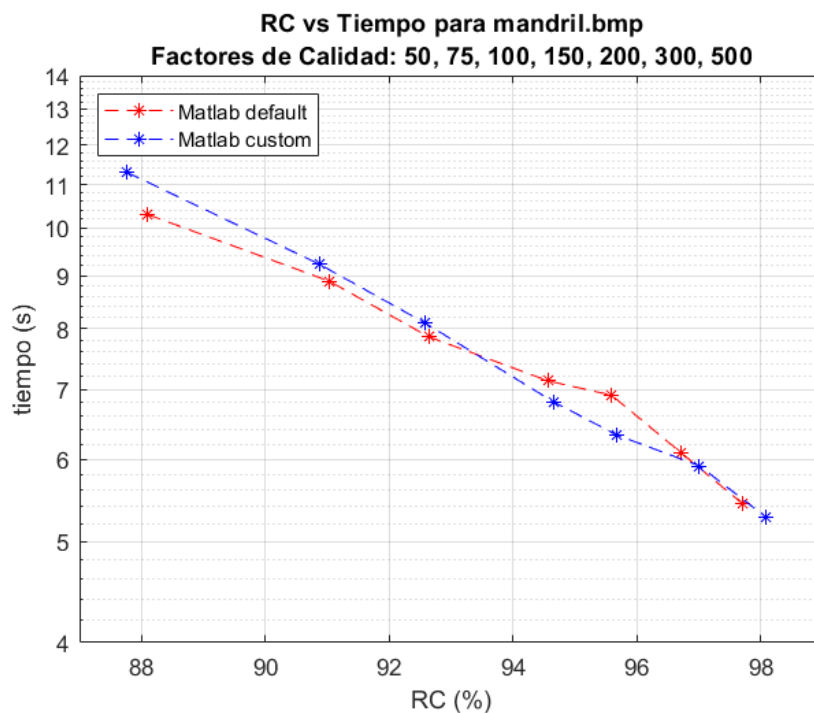


Figura 42. Comparación de tiempos de ejecución y RC en “mandril” para todos los factores de calidad seleccionados

Para la imagen del mandril, que era la imagen que presentaba una mayor variabilidad en sus bloques, puede observarse que la diferencia de tiempos no es tan grande y que, de hecho, se “entrelazan” en ciertos puntos, siendo más rápido a veces un compresor y a veces otro.

Se puede concluir, por tanto, que los tiempos de ejecución en esta implementación no sólo dependen del método de codificación, sino de los propios datos y de su distribución, que podrían inclinar la balanza sobre cuál es el compresor más rápido. Además, la influencia del sistema operativo en la prioridad de procesos puede afectar en esta medición y los tiempos no tienen por qué ser precisos en sistemas no aislados.

6. Parte opcional II: Comparativa usando software alternativo

Para la segunda parte opcional escogida, será realizado un análisis comparativo entre los resultados relativos a la compresión que se obtienen con el uso de la implementación de JPEG en MATLAB y con otro software libre, “Caesium”^[8], que también comprime imágenes BMP a formato JPEG.

Se ha elegido este software de compresión primero porque es gratuito y puede ser instalado de manera rápida, y segundo porque es muy sencillo de usar y dispone de un puntero deslizante donde puede escogerse un factor de calidad para la compresión de imágenes.

Este factor de calidad no funciona de la misma manera que en la implementación de MATLAB, sino que sus valores oscilan entre 1 y 100, lográndose mayor compresión conforme se reduce este valor, y menor compresión en caso contrario. Para poder realizar el análisis, será necesario establecer una relación entre ambos factores de calidad, de forma que los valores escogidos en “Caesium” sean equivalentes a los de MATLAB.

Para ello, y tras muchas pruebas a mano con distintos valores de calidad en las diferentes imágenes seleccionadas, he llegado a la siguiente tabla de equivalencia, donde las razones de compresión para ambos valores son muy similares entre ellas para todas las imágenes o, en otras palabras, la diferencia de RC entre las imágenes era mínima para todas las imágenes (aunque existían distancias insalvables y las equivalencias no son 100% reales, sino una buena aproximación válida para el estudio del software).

Factor de calidad MATLAB	Factor de calidad Caesium
50	80
75	70
100	60
150	40
200	33
300	19
500	11

Los resultados obtenidos en la compresión de las imágenes junto con el error producido con respecto a las imágenes originales se encuentran separados en dos tablas diferentes, y cada fila de las columnas representadas indican los valores para un factor de calidad en concreto, equivalente al de la implementación en MATLAB.

	loros.bmp		f-16.bmp		faro.bmp	
Factor de calidad equivalente	MSE	RC (%)	MSE	RC (%)	MSE	RC (%)
50	12.3246	95.52	30.3730	94.35	16.5127	93.83
75	16.1414	96.52	35.7492	95.51	23.3829	95.11
100	20.0869	97.08	40.8113	96.24	29.6275	95.87
150	27.9167	97.76	55.2649	97.12	43.0349	96.84
200	32.1382	97.98	60.8843	97.41	50.3715	97.15
300	51.6718	98.50	86.0670	98.11	80.5919	97.93
500	86.7159	98.86	125.3576	98.61	131.4134	98.48

Los datos no nos dicen nada inusual a simple vista, más que los valores arrojados son coherentes y válidos según los datos obtenidos en el estudio básico realizado. Quedarán por tanto como fuentes de las gráficas comparativas que serán mostradas a continuación.

Todas ellas muestran la comparación de MSE y RC obtenidas para los 3 compresores diferentes y para los 7 factores de calidad establecidos para una imagen en específico, haciendo el color verde referencia al nuevo software como se indica en la leyenda.

	despegue.bmp		mandril.bmp		casa.bmp	
Factor de calidad equivalente	MSE	RC (%)	MSE	RC (%)	MSE	RC (%)
50	27.1185	94.78	139.1289	88.69	51.3953	93.02
75	33.4225	95.99	167.5541	91.11	61.6551	94.42
100	39.2172	96.68	191.6440	92.58	69.9560	95.27
150	50.4637	97.49	235.7946	94.44	88.6315	96.39
200	56.3505	97.73	256.0855	95.09	97.1287	96.77
300	79.6382	98.37	328.3689	96.64	132.7474	97.68
500	117.9243	98.79	425.3782	97.71	189.6532	98.36

La primera de ellas representa la evolución de los parámetros con la imagen de los loros. Puede observarse que, para los factores de calidad más bajos, el MSE es más alto en el nuevo software, pero a partir del factor de calidad 100, este MSE es el más bajo de los 3. Además, se puede ver que está ligeramente más optimizado que el compresor 'default', pero el compresor 'custom' sigue comprimiendo más que el nuevo software, a costa de un mayor MSE claro está.

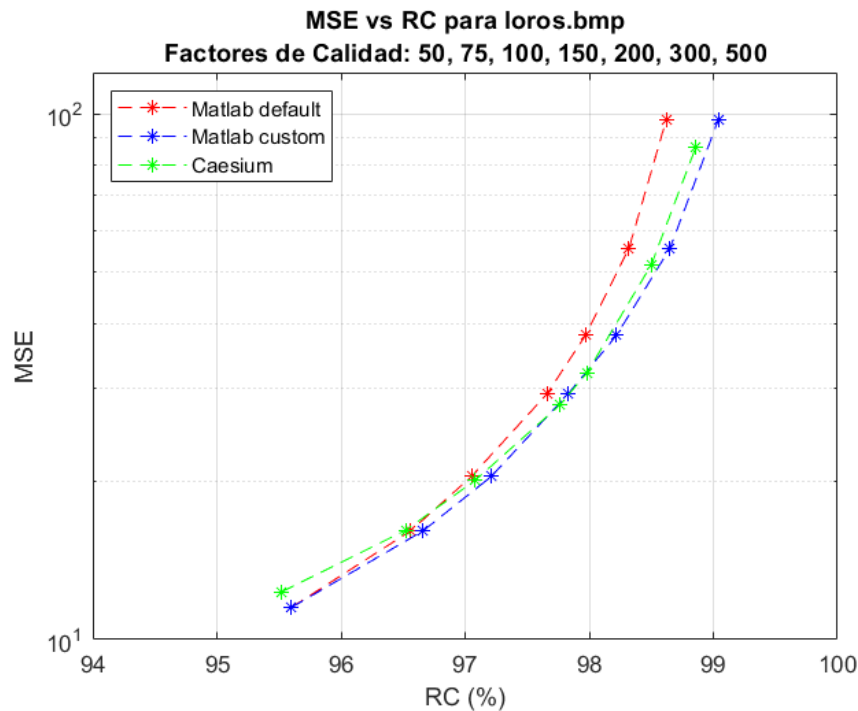


Figura 43. Gráfica comparativa del MSE y RC obtenidos con el software “Caesium” para “loros”

Esta diferencia en el error puede residir en que los redondeos de MATLAB no sean del todo exactos y el hecho de utilizarlo ya va a provocar una pérdida irre recuperable en las imágenes comprimidas.

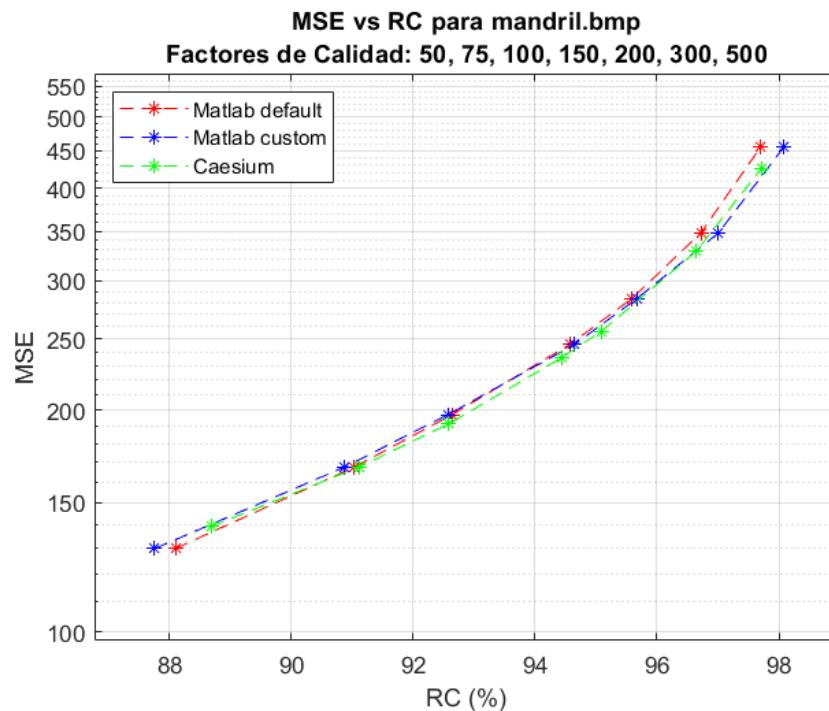


Figura 44. Gráfica comparativa del MSE y RC obtenidos con el software “Caesium” para “mandril”

Es importante decir que la compresión con codificación ‘default’ implementada en MATLAB está basada en las tablas de cuantización recomendadas por el estándar, y es probable que este nuevo compresor implemente tablas similares, pero no las mismas, y de ahí podría venir también la diferencia en el error, mayor para factores de calidad pequeños y viceversa. También que la equivalencia entre factores de calidad no es del todo exacta para todas las imágenes y los puntos son aproximativos.

Para la gráfica referente a la imagen del mandril, se puede observar que este fenómeno se vuelve a dar, aunque le cuesta superar los ratios de compresión obtenidos por el compresor ‘default’ (que no son muy diferentes). Esto refuerza la idea de que las tablas de cuantización recomendadas por el estándar han sido alteradas, ya que los intervalos de cuantización para los mismos factores de calidad no afectan de la misma manera a los bloques de la imagen y se producen RC y MSE distintos. También podría ser posible que la codificación se realice de manera diferente e incluso el particionado del espacio de color no sea YCbCr.

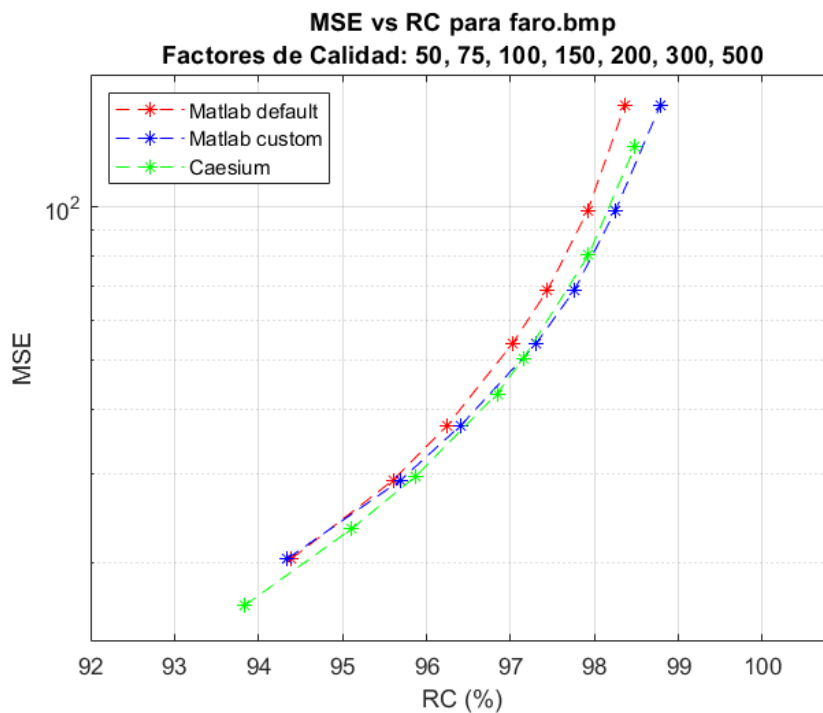


Figura 45. Gráfica comparativa del MSE y RC obtenidos con el software “Caesium” para “faro”

Para la imagen del faro, la gráfica mostrada es un poco diferente, porque el software de compresión se queda notablemente atrás en los resultados de RC obtenidos hasta el factor de calidad 300, aunque manteniendo siempre un MSE menor. Esto podría indicar un pequeño posible desajuste entre los factores de calidad equivalentes más bajos (como he comentado antes, la equivalencia es aproximada) y no creo que sea representativo del comportamiento general del nuevo compresor.

Las gráficas obtenidas para el resto de imágenes son similares a las comentadas y no serán incluidas para evitar redundancia. Para el análisis cualitativo, tomaré una imagen de ambos compresores y las comentaré, teniendo siempre en cuenta que no sé cómo está implementado el software por dentro.

La imagen escogida es la del faro, ya que como se vio en el análisis básico, tiene bastantes elementos distintos.



Figura 46. Comparativa entre compresor MATLAB y Caesium con factor de calidad 100

Para un mismo factor de calidad equivalente, ambas imágenes se comprimen de la misma manera y la diferencia en calidad es casi imperceptible. Ambas presentan imperfecciones en los bordes con cambios de color brusco y podrían pasar por la imagen original si no se amplían hasta cierta distancia.

¿Qué compresor es mejor? La elección en este caso está bastante clara: “Caesium”. El motivo es que puede que presente pequeñas diferencias en las relaciones de compresión, pero la velocidad a la que realiza la compresión es muy superior a MATLAB, ya que comprimía todas las imágenes de seguido en apenas medio segundo y, al final, el bien máspreciado para el usuario es su tiempo, no unos 500 bytes de ganancia tras el proceso de compresión.

7. Conclusiones

Como conclusión al análisis desarrollado, se puede decir que JPEG, como algoritmo de compresión con pérdidas, funciona muy bien con imágenes que tienen variaciones suaves de tonalidad y color, y es en éstas donde menos se percibe la pérdida de calidad producida (independientemente del error producido por la diferencia de imágenes) por el proceso de compresión, siempre y cuando no existan gradientes de color u otros fenómenos que provoquen la aparición de zonas “cuadradas” o bloques “manchados” en la imagen comprimida.

A este algoritmo le afecta particularmente las imágenes con bordes pronunciados y otras figuras como texto, donde la pérdida de coeficientes de alta variabilidad producidos durante el proceso de cuantización de los datos de la imagen transformados normalmente implicaban una pérdida de calidad perceptual y un aumento del error cuadrático medio.

En cuanto a las razones de compresión, las imágenes que presentaban una alta variabilidad (en forma de mayor número de coeficientes no nulos tras la cuantización) se comprimían peor, ya que se tenían que codificar más datos, y las imágenes con variabilidad más baja lograban una mayor compresión gracias a que sobrevivían menos coeficientes, aunque esto no afectaba apenas, en general, a la calidad percibida. Destacar que un error cuadrático medio alto no implicaba una peor calidad percibida siempre.

La codificación ‘custom’ se adaptaba mejor a los datos de las imágenes y para factores de calidad por encima de 100 o 150 lograba mejores resultados.

Como nota final al trabajo, JPEG es una excelente aportación tecnológica en el ámbito de la compresión que revolucionó la transmisión y el almacenamiento de imágenes y lo seguirá haciendo en el futuro, sin duda, por los buenos resultados que ofrece.

8. Bibliografía

- [1] JPEG, Wikipedia angloparlante, <https://en.wikipedia.org/wiki/JPEG>
- [2] David Salomon & Giovanni Motta, Handbook of Data Compression, 5th Edition, pág. 451
- [3] Khalid Sayood, Introduction to Data Compression (4th Edition), pág. 261
- [4] Tema 3. Entropía y codificación. Interpretación de la Entropía III, pág 17, Luis Daniel Hernández Molinero, basado en el trabajo de Roque Marín.
- [5] Imagen “mandril”, “f-16” y “casa”, repositorio de la Universidad de Carolina del Sur, <http://sipi.usc.edu/database/database.php?volume=misc>, 20:36 - 22/12/2020
- [6] Imágenes “loros” y “faro”, Instituto Informático Max-Planck, <https://gvv.mpi-inf.mpg.de/projects/ImageEnhancement/>, 20:42 - 22/12/2020
- [7] Imagen “despegue”, W. Randolph Franklin (WRF), Rensselaer Polytechnic Institute (RPI), Troy NY USA, <https://wrf.ecse.rpi.edu/wiki/ComputerGraphicsFall2013/guha/NewCode/Chapter20/Tutorials/>, 20:49 - 22/12/2020
- [8] Caesium Image Compressor website, desarrollado por Matteo Paonessa <https://saerasoft.com/caesium/>

9. Anexos

9.1. Anexo I – Extensión de conceptos técnicos

En este anexo se explicarán y se extenderán ciertas ideas comentadas de manera muy rápida en el punto referente metodología y selección de criterios. La idea no es repetir los conceptos que se explican en la asignatura sino darle una visión más completa a los lectores más curiosos y explicar el funcionamiento del compresor sin entrar en el código (ya que está comentado en los scripts).

9.1.1. Relación de compresión

La relación es una medida que indica cuánto comprime un algoritmo de compresión. Existen varias definiciones sobre esta medida, pero en este caso, yo uso la definición de “tasa de compresión”, que lo que expresa es el porcentaje de espacio que se ha podido reducir con respecto al tamaño original. Por ejemplo, si un archivo logra una tasa de compresión del 66% significa que el tamaño del fichero original se ha reducido en 2/3. Esta definición de relación de compresión viene dada por la siguiente expresión:

$$RC = (TO - TC / TO) \times 100$$

Siendo TO el tamaño del fichero original en bytes y TC el tamaño del fichero comprimido.

9.1.2. Error cuadrático medio (MSE)

El error cuadrático medio es una medida que indica qué tan cerca se encuentra un dato reconstruido del original, en nuestro caso píxeles. Es la media de cada una de las diferencias elevadas al cuadrado y viene dado por la siguiente expresión:

$$\sigma_d^2 = \frac{1}{N} \sum_{n=1}^N (x_n - y_n)^2$$

Donde N es el número de elementos, ‘x’ el valor original e ‘y’ el valor recuperado.

9.1.3. Matrices DCT e imágenes base

Los datos de una imagen, antes de ser cuantizados y codificados, requieren de una transformación en la que se cambien estos datos por unos coeficientes, de forma que se deja el bloque de 8x8 píxeles que sufre la transformación como una combinación lineal de estos coeficientes, que indican “cuánto” de cada imagen base contiene el bloque. Las imágenes base son las siguientes:

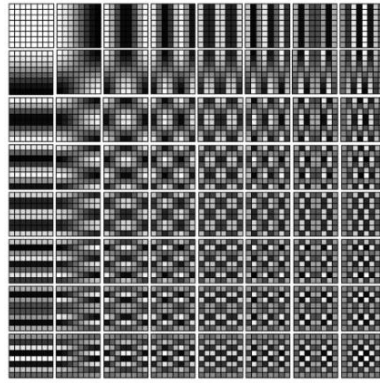


Figura 46. Imágenes base

Como se puede observar en la imagen, conforme se avanza hacia abajo y hacia la derecha por la imágenes base, la variabilidad u “oscilación” aumenta (aunque esta matriz no es simétrica). Alrededor del primer valor de estos bloques, el coeficiente DC, se van a encontrar los coeficientes más “energéticos” o importantes, de forma que cuando se cuantice el bloque, los coeficientes menos importantes (los llamados coeficientes AC) serán anulados debido a estar su valor próximo a 0.

A la matriz transformada del bloque se le llama matriz DCT, porque la técnica usada para transformar los datos se llama Transformada Discreta del Coseno.

¿Por qué los bloques son de tamaño 8x8? Esta idea se basa en que los bloques de las imágenes son bastante parecidos entre ellos para facilitar la codificación como se explicará más adelante. Si el bloque es muy pequeño, todos serán prácticamente iguales, pero aumentaría el costo computacional y no se lograrían buenos resultados en la compresión, por apenas se perdería detalle. Si el bloque fuera muy grande, estos dejarían de parecerse entre ellos y los efectos producidos por la compresión en las imágenes recuperadas serán mucho más perceptibles y no merecería la pena comprimirlas.

Cabe mencionar que esta transformación se aplica a las 3 capas, de forma independiente, pertenecientes al espacio de color YCbCr en el que se convierte la imagen (cuyo espacio de color es RGB) previamente a la aplicación de las transformadas.

9.1.4. Cuantización

Tras transformar los datos, se realiza un proceso de cuantización. De lo que se trata es de hacer intervalos de una cierta longitud configurable, de forma que los datos que caigan dentro de un intervalo pasen a valer lo mismo. El resultado de esta cuantización es que dentro de cada bloque 8x8, los datos van a convertirse en la etiqueta que indica el intervalo donde han caído al ser cuantizados.

Existen tres matrices de cuantización en nuestra implementación basada en el estándar de JPEG, una para cada capa del espacio de color transformado y se aplica nuevamente bloque a bloque. Cada matriz tiene un tamaño de intervalo predefinido según la posición

del coeficiente, de forma que los coeficientes más alejados del DC tienen, por lo general, intervalos de un tamaño mayor inicialmente. Estos intervalos son más grandes en las capas de crominancia que en la de luminancia, debido a que el ojo humano no es tan bueno percibiendo cambios en estas capas y el descarte temprano de estos valores permite compresión con pérdidas cuantitativas pero pocas cualitativas.

Gracias a este proceso, los datos transformados anteriormente que son menos importantes se convierten en 0 (cuanto más grande sea el factor de calidad, más grande es el intervalo y más valores entrarán en el mismo intervalo), puesto que se transforman en el mismo valor. En este punto ya se ha perdido información sobre la imagen original. Es importante destacar que los valores cuantizados no son más que ‘etiquetas’ para identificar el intervalo donde han caído los valores.

En el caso de los compresores utilizados, estos intervalos son fijos y se ven alterados por el factor de calidad comentado. Este factor se divide entre 100 y se multiplica por los valores en la matriz de cuantización prefijada, de forma que, si el factor es 100, la aplicación de la cuantización es la estándar. Si está por debajo de 100, se logra menos compresión y los intervalos son más pequeños con respecto al estándar y viceversa.

9.1.5. Codificación en JPEG

Los bloques 8x8 (que, por cierto, podrían no tener este tamaño), una vez cuantizados sufren un preprocesado en el que las matrices cuantizadas resultantes son escaneadas en forma de zigzag para que los valores que no hayan sido cuantizados a 0 se ubiquen ahora en los valores iniciales. Esto funciona porque si recordamos, los coeficientes más “energéticos” resultantes de la aplicación de la DCT se concentran alrededor del primer valor. Además, esto también se hace para que los ceros queden lo más lejos posible de los valores no nulos y se puedan codificar fácilmente todos juntos de forma que su recuperación sea sencilla al decodificar.

Una vez preprocesadas las matrices cuantizadas, la codificación de los valores DC y de los valores AC se realiza de forma separada. Para los coeficientes DC, en lugar de codificar su valor tal cual, se calcula la diferencia con el bloque anterior (que es esperable que sea pequeña porque como he comentado esto se basa en que los bloques son muy parecidos entre ellos) y esa diferencia se corresponde con una categoría establecida por el estándar JPEG. Esta categoría junto con la posición que ocupa dentro de ella es lo que se codifica finalmente para cada uno de los coeficientes DC.

Esta idea se basa en el Principio de Compresión de Imágenes ^[2], que lo que viene a decir es que, si se selecciona un bloque de la imagen de forma aleatoria, hay una buena probabilidad de que sus bloques vecinos sean iguales o bastante similares, ya que existe una correlación entre ellos.

Para los valores AC el proceso es distinto. Se utiliza la codificación “Run-Length”, donde primero se obtienen el número de ceros previos a cada etiqueta no nula (hasta 16, que se codificarían como un solo bloque de 16 ceros consecutivos), después la categoría que se

le asigna a la etiqueta según JPEG y finalmente la posición de la etiqueta en la categoría asignada para poder identificar completamente a la etiqueta durante el proceso de descompresión. Estos datos se almacenan en tablas que son posteriormente codificadas. Cabe señalar que esta idea proviene del hecho de que las matrices han sido recorridas previamente en zigzag y, por tanto, esta codificación no es tan pesada como podría serlo sin este recorrido.

9.2. Anexo II – Implementación de software para facilitar el análisis

Para la generación de gráficas que permitan comparar el MSE con la relación de compresión para cada factor de calidad, realicé un pequeño script que, para una imagen, la comprimiera y descomprimiera una vez por factor de calidad, devolviendo así un vector con las relaciones de compresión y con el MSE de cada uno, que pudieran ser dibujados en una gráfica usando comandos de MATLAB.

Este script tiene el nombre de “jcomdes_vectors”, por si se quisiera consultar su código dentro de la carpeta “source”.