

INSTITUTO SUPERIOR DE
ENGENHARIA DE LISBOA
ADEETC

LICENCIATURA EM ENGENHARIA INFORMÁTICA E
MULTIMÉDIA

SEMESTRE DE VERÃO

3º TRABALHO PRÁTICO

Codificação de Sinais Multimédia

Realizado por:

João SANTOS nº 39348

Rui SANTOS nº 39286

Junho, 2017



Conteúdo

1	Introdução	3
2	Desenvolvimento	4
2.1	Compressão de imagens com perdas	4
2.2	Codificação/descodificação baseada na DCT	5
2.3	Codificação/descodificação baseada na DCT com quantificação	7
2.3.1	Resultados obtidos	9
2.4	Codificação/descodificação com perdas JPEG baseada na DCT	10
2.4.1	Codificação de Huffman dos coeficientes DC	11
2.4.2	Codificação de Huffman dos coeficientes AC	12
2.4.3	Resultados obtidos	14
3	Conclusões	18

Lista de Figuras

1	Codificação baseada na DCT	6
2	Decodificação baseada na IDCT	6
3	Imagem codificada utilizando a DCT	6
4	Imagem decodificada utilizando a IDCT	6
5	Blocos 8×8	7
6	Codificação e decodificação baseada na DCT com quantificação	8
7	Matriz de quantificação	8
8	Codificação utilizando DCT com quantificação	9
9	Decodificação utilizando IDCT com quantificação	9
10	Diagrama de blocos do codificador/descodificador JPEG . . .	10
11	Processamento em modo zigzag	13
12	Primeiro bloco 8×8 resultante da DCT	14
13	DCT e decodificação para um fator de qualidade de 25% . . .	15
14	DCT e decodificação para um fator de qualidade de 50% . . .	15
15	DCT e decodificação para um fator de qualidade de 75% . . .	15
16	Taxa de Compressão em função da SNR	17

Lista de Tabelas

1	Valores de SNR utilizando codificação DCT com quantificação	10
2	Resultados obtidos	16

1 Introdução

O terceiro trabalho prático da disciplina de Codificação de Sinais Multimédia tem como objetivo explorar os princípios básicos da norma JPEG (modo sequencial) para compressão de imagens com perdas. O JPEG é talvez dos formatos de compressão de imagens mais utilizados hoje em dia, especialmente utilizado para comprimir imagens fotográficas. É um formato de ficheiro de imagem muito popular, que permite uma grande compressão mantendo uma boa qualidade. É de realçar que as perdas de informação presentes no JPEG são proporcionais ao fator de compressão desejado.

A norma JPEG foi implementada com dois modos básicos, o método de compressão com perdas, que utiliza um algoritmo baseado na DCT (Discrete Cosine Transform) e um método de compressão sem perdas, que se baseia em métodos preditivos. Como referido anteriormente o método de compressão implementado no trabalho vai introduzir perdas na imagem, logo, baseado na DCT.

De modo a testar o algoritmo implementado neste trabalho iremos utilizar algumas imagens de teste, estas imagens vão ser codificadas e decodificadas utilizando a norma JPEG implementada. Na parte final do trabalho vão ser analisados os resultados obtidos nos valores da SNR (Signal-to-noise ratio) e as Taxas de Compressão para diferentes fatores de qualidade.

2 Desenvolvimento

O desenvolvimento do trabalho prático 3 decorreu de modo a responder às três questões principais presentes no enunciado do trabalho, sendo estas:

1. Implementação de um codificador e decodificador utilizando a DCT e a IDCT sem quantificação.
2. Implementação de um codificador e decodificador da DCT e da IDCT com quantificação.
3. Implementação de um codificador e decodificador de entropia na norma JPEG.

Como estes três pontos em consideração, iremos realizar um breve enquadramento teórico em cada um deles e apresentar os resultados obtidos em cada fase do trabalho.

2.1 Compressão de imagens com perdas

No que diz respeito à compressão de imagens com perdas, os dados tendem a possuir um grau elevado de redundância espacial, isto é, os valores dos píxeis tendem a repetir-se com muita frequência. Para além disso, os dados da imagem a comprimir destinam-se sobretudo a utilizadores humanos. Explorando-se estes dois aspetos os sistemas de compressão com perdas obtêm-se explorando dois fatores principais:

- As **redundâncias espaciais** contidas na imagem;
- As **características percetivas do sistema visual humano**, de modo a que a perda introduzida pela compressão seja impercetível ao sistema visual do ser humano.

2.2 Codificação/descodificação baseada na DCT

Como foi referido anteriormente neste trabalho prático foi utilizado o método de codificação baseado na DCT. O funcionamento de um sistema de compressão de imagem baseado na DCT assenta na transformação de um bloco de imagem, com dimensões $N \times N$, do domínio espacial para o domínio das frequências. Nas normas de compressão de imagem a dimensão mais eficiente para os blocos de píxeis corresponde a um valor de $N = 8$ píxeis, pelo que se utilizam blocos de 64 píxeis (8×8).

A DCT é uma transformada ortogonal que pode ser expressa através da seguinte expressão analítica:

$$\text{DCT: } y_{kl} = \frac{c(k) \cdot c(l)}{4} \sum_{i=0}^7 \sum_{j=0}^7 x_{ij} \cos\left(\frac{(2i+1) \cdot k\pi}{16}\right) \cos\left(\frac{(2j+1) \cdot l\pi}{16}\right)$$

onde,

$$\begin{aligned} &k, l = 0, 1, \dots, 7; \\ &c(k) \text{ e } c(l) = \frac{1}{\sqrt{2}}, \text{ se } k = 0 \text{ ou } l = 0; \\ &c(k) \text{ e } c(l) = 1, \text{ para todos os outros valores de } k \text{ e } l; \end{aligned}$$

De modo a implementar a DCT no nosso trabalho foi utilizado o método *cv2.dct()* da biblioteca OpenCV, que permite realizar o calculo representado pela fórmula descrita em cima.

Do mesmo modo que se efetuou o cálculo da DCT, a IDCT pode ser obtida através da seguinte expressão analítica:

$$\text{IDCT: } x_{ij} = \sum_{k=0}^7 \sum_{l=0}^7 y_{kl} \cdot \frac{c(k) \cdot c(l)}{4} \cos\left(\frac{(2i+1) \cdot k\pi}{16}\right) \cos\left(\frac{(2j+1) \cdot l\pi}{16}\right)$$

onde,

$$i, j = 0, 1, \dots, 7;$$

Analisando esta expressão, podemos verificar que os cálculos para a DCT e para a IDCT são muito semelhantes.

Como referido anteriormente, a primeira fase do trabalho limitou-se à implementação de um codificador e de um decodificador baseado na DCT sem quantificação. Para tal, foram desenvolvidos dois métodos que realizam as ações descritas nos diagramas de blocos das Figura 1 e 2.

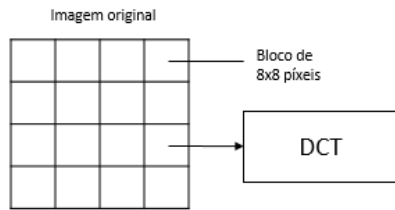


Figura 1: Codificação baseada na DCT

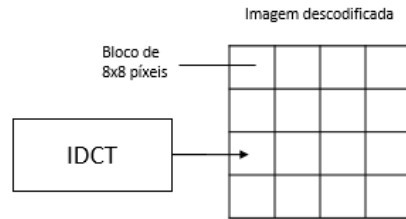


Figura 2: Decodificação baseada na IDCT

Aplicando a codificação representada na Figura 1 à imagem original obtemos os seguintes resultados:

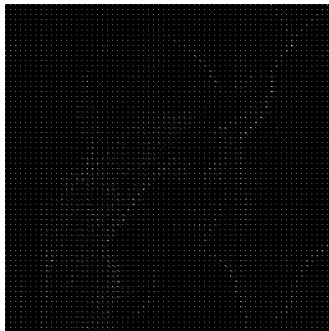


Figura 3: Imagem codificada utilizando a DCT



Figura 4: Imagem decodificada utilizando a IDCT

Na Figura 3 aplicou-se a DCT 8×8 sobre a imagem, obtendo-se a imagem transformada pela DCT. É de notar que neste ponto ainda não se obteve qualquer compressão. Deve-se notar também que, cada bloco da imagem da Figura 3 possui as maiores amplitudes concentradas em torno do elemento $[0,0]$ (canto superior esquerdo). O valor $[0,0]$ designa-se por coeficiente DC, os restantes componentes de um bloco 8×8 são designados de coeficientes AC. É possível observar mais pormenorizadamente estes blocos na Figura 5, onde estão representados 4 blocos 8×8 .

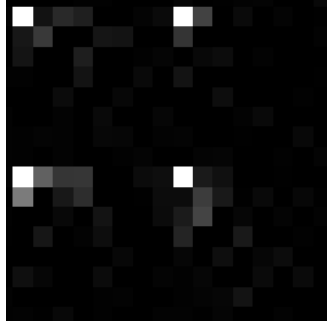


Figura 5: Blocos 8×8

2.3 Codificação/descodificação baseada na DCT com quantificação

O segundo ponto do trabalho gira em volta da quantificação, o processo de quantificação constitui a forma principal de compressão em qualquer método de compressão com perdas, uma vez que é durante a quantificação que ocorre a **perda**, ou seja, a eliminação das partes da informação irrelevantes.

De uma forma geral, para um bloco com pouca atividade, a maioria dos dados de alta frequência existirá nos coeficientes de ordem inferior, tal como se pode observar na Figura 5. Como se explicou anteriormente, é precisamente o processo de quantificação que conduz à compressão no âmbito de um sistema de codificação baseado na DCT. O processo de quantificação pode exprimir-se pela seguinte expressão:

$$z_{kl} = \text{round}\left(\frac{y_{kl}}{q_{kl}}\right)$$

onde,

$k, l = 0, 1, \dots, 7$;

q_{kl} representa os elementos da matriz de quantificação Q ;

Na segunda fase do trabalho foi implementada a quantificação, que como referido anteriormente, vai produzir a compressão na imagem. Deste modo foram desenvolvidos dois métodos de codificação e decodificação que realizam as ações descritas no diagrama de blocos da Figura 6.

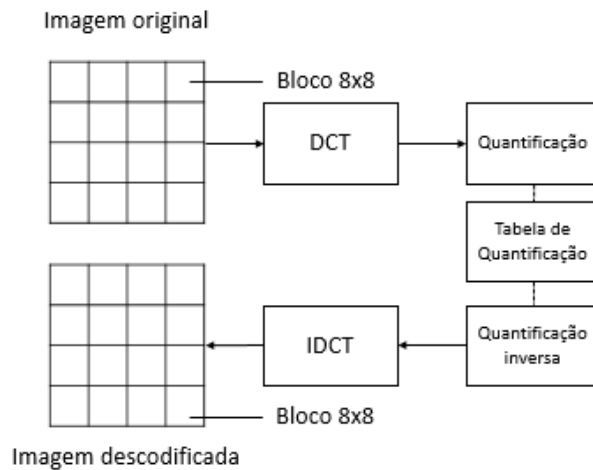


Figura 6: Codificação e decodificação baseada na DCT com quantificação

A matriz de quantificação $Q\ 8 \times 8$ foi disponibilizada juntamente com o enunciado do trabalho e pode ser observada na Figura 7.

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

Figura 7: Matriz de quantificação

2.3.1 Resultados obtidos

Implementados os métodos que realizam a codificação e a decodificação, utilizando quantificação, foram obtidos os seguintes resultados, para um fator de qualidade de 75%.

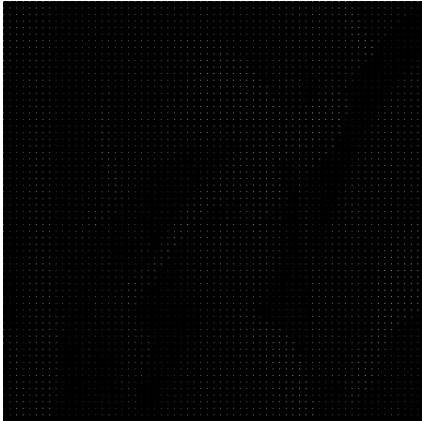


Figura 8: Codificação utilizando DCT com quantificação



Figura 9: Decodificação utilizando IDCT com quantificação

Comparando os resultados obtidos na Figura 3 e na Figura 8, é possível observar a olho "nu" que os níveis de frequências na Figura 8 são muito menores que na Figura 3. Isto acontece pois a informação irrelevante da imagem é descartada, logo, existe perdas. É possível observar também pela Figura 8, que utilizando um fator de compressão de 75% conseguimos obter uma imagem decodificada com uma qualidade muito aproximada da imagem original.

Com o diagrama de blocos da Figura 6 implementado, foram efetuados os cálculos das SNR's (Signal-to-noise ratio) para diferentes fatores de qualidade, respectivamente, 25%, 50% e 75%. A expressão analítica da SNR que foi utilizada define-se do seguinte modo:

$$SNR = 10 \log_{10} \frac{P_{sinal}}{P_{ruído}}$$

A interpretação dos valores da SNR deve fazer-se do seguinte modo:

- Valores elevados de SNR significam maior qualidade da imagem, isto é, menos distorção.
- Valores baixos de SNR implicam menor qualidade de imagem, isto é, maior nível de distorção ou perda.

Tendo estes dois pontos em consideração podemos observar na Tabela 1 os resultados obtidos de valores de SNR para os diferentes fatores de compressão:

Fator de qualidade	Relação sinal-ruído
25%	28.06
50%	30.17
75%	32.24

Tabela 1: Valores de SNR utilizando codificação DCT com quantificação

Como é possível observar através dos valores obtidos, a relação sinal-ruído é proporcional ao fator de qualidade utilizado, sendo que, quanto maior a qualidade da imagem maior o seu valor de SNR. Um aspeto a ter em conta na norma JPEG deve-se ao facto de, apenas existir perdas na quantificação, logo, na terceira fase do trabalho prático o valores da relação sinal-ruído devem-se manter iguais. Caso isto não aconteça, pode significar problemas na implementação do algoritmo, visto que, a codificação de entropia não introduz perdas na codificação de imagens.

2.4 Codificação/descodificação com perdas JPEG baseada na DCT

É possível observar na Figura 10 um diagrama de blocos do codificador JPEG baseado na DCT para uma imagem em tons de cinzento (apenas um componente de cor).

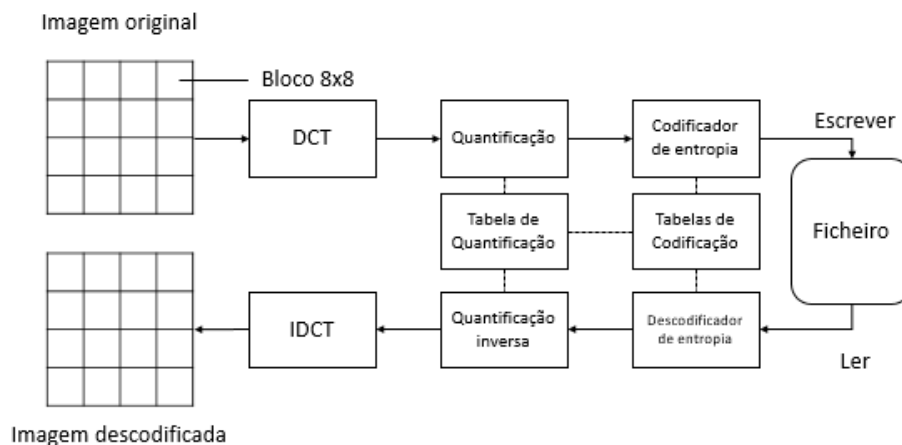


Figura 10: Diagrama de blocos do codificador/descodificador JPEG

Grande parte do processo presente no diagrama de blocos da Figura 9 já foi descrito nos subcapítulos 2.2 e 2.3, logo, a terceira fase do trabalho requer a implementação do codificador de entropia para os valores resultantes da quantificação da imagem.

Resumidamente, é realizada a divisão da imagem em blocos de dimensão 8×8 que não se sobrepõem. Em seguida, transforma-se cada bloco de píxeis para o domínio das frequências através de uma transformada DCT (2-D). Os coeficientes que resultam da DCT são posteriormente quantificados e, a seguir, codificados sem perdas por um codificador de entropia.

Na norma JPEG, a **codificação de entropia** consiste em **duas fases**:

1. A **primeira** fase envolve a realização de uma codificação preditiva para o coeficiente DC da DCT quantificada, isto é, para o valor $[0,0]$ da matriz, seguida por uma codificação RL em zig-zag dos coeficientes AC da DCT quantificados.
2. A **segunda** fase corresponde a uma codificação de entropia dos valores resultantes da primeira fase, que pode ser realizada recorrendo ao método de Huffman.

No que diz respeito ao processo de descodificação de um fluxo de dados JPEG comprimido, o processo é referido na segunda parte no diagrama de blocos da Figura 10. No caso do trabalho prático implementado é realizada uma leitura de um ficheiro de texto onde a imagem foi codificada, seguidamente, os dados referentes ao píxeis da imagem passam por um descodificador de entropia, obtendo-se os coeficientes da DCT quantificados. Com estes valores obtidos é realizada a quantificação inversa explicada anteriormente no subcapítulo 2.3 de modo a obter a imagem descodificada.

2.4.1 Codificação de Huffman dos coeficientes DC

Como referido anteriormente, os coeficientes DC representam os elementos $[0,0]$ da matriz de quantificação, pois estes são os coeficientes com maior frequência. A codificação destes coeficientes é realizado utilizando o DC_i e DC_{i-1} , para tal, é realizada uma codificação diferencial (DPCM) para estes coeficientes, calculando o diferença entre DC_i e DC_{i-1} , ou seja, o componente DC atual e o anterior.

Com a diferença entre os coeficientes calculada é realizada a sua codificação, para tal, cada diferença passa a ser descrita por dois parâmetros distintos (grandeza, amplitude), a grandeza representa o número de bits necessários para definir a amplitude, e o parâmetro amplitude representa o valor da diferença calculada.

Existem duas noções a ter em consideração quando é realizado o cálculo da diferença:

- Se a diferença for um valor positivo, a amplitude corresponde a esse valor em binário, sendo que, o tamanho desse valor binário será representado pela grandeza.
- Se a diferença for um valor negativo, é transformado esse valor para binário e de seguida efetuado o seu complemento, ou seja, os bits que contiverem valor 1 passam a 0 e os que tiverem valor 0 passam a 1, de seguida, é representada a grandeza do mesmo modo descrito anteriormente.

Podemos utilizar como exemplo o primeiro valor de grandeza que obtemos no nosso trabalho, visto que para o primeiro bloco 8×8 da imagem não existe um bloco anterior, então é feita a subtração do coeficiente DC atual por 0. Para um fator de compressão de 75% o valor do primeiro coeficiente DC é 160, passando o valor 160 para binário é obtida a amplitude 10100000, de seguida, é obtido o comprimento da amplitude que toma o valor 8. Assim, o valor 160 passa a ser descrito pelos parâmetros (8,10100000). O próximo passo é consultar a tabela de Huffman e obter o código para uma grandeza de 8, com este valor obtido o número 160 passa a ser codificado com os bits 11111010100000, grandeza + amplitude.

A descodificação dos componentes DC resulta no processo inverso à codificação, ou seja, é descodificado em primeiro lugar a grandeza, desde modo, conseguimos obter o comprimento dos bits que representam a amplitude.

2.4.2 Codificação de Huffman dos coeficientes AC

No que diz respeito aos coeficientes AC existem algumas diferenças na sua codificação, tais como os coeficientes DC, estes também são descritos pelos parâmetros (grandeza, amplitude). A grande diferença entre este tipo de codificação e a anterior, deve-se ao facto do modo em que os coeficientes são processados, sendo estes processados em modo de zigzag. É possível observar este modo na Figura 11:

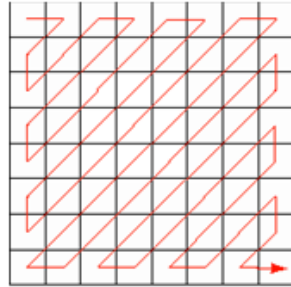


Figura 11: Processamento em modo zigue-zague

Os coeficientes AC são processados obtendo o próximo valor de um AC diferente de 0, é então calculada uma *run* que representa o número de coeficientes AC iguais a 0 que precedem o coeficiente não-nulo atual. Deste modo, cada coeficiente passa a ser descrito por os parâmetros (*run*/grandeza, amplitude). Com estes dois parâmetros obtidos o método de codificação é igual ao dos coeficientes DC.

Existem porem algumas particularidades a ter em conta, das quais, as seguintes:

- Caso o valor de uma *run* seja superior a 15, é utilizado o valor (15/0) de modo a representar uma *run* de 15 zeros, seguido por um zero.
- É utilizado uma marca EOB (end-of-block) caso todos os coeficientes restantes após um coeficiente não nulo forem nulos, para tal, é utilizado o símbolo 0/0 que é representado pelo valor 1010.

2.4.3 Resultados obtidos

Implementados os métodos que realizam a codificação e a decodificação seguindo o diagrama de blocos da Figura 10, resta apresentar os resultados obtidos e alguns exemplos do funcionamento do algoritmo.

Tendo como exemplo o primeiro bloco 8×8 resultante da DCT quantificada da Figura 8, podemos observar a vermelho os coeficientes DC e a verde os coeficiente AC.

160	1	1	0	0	0	0	0
1	0	0	-1	0	0	0	0
-1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figura 12: Primeiro bloco 8×8 resultante da DCT

Procedendo à codificação deste bloco são obtidos os seguintes resultados:

- DC: (grandeza, amplitude) [160] = (8, 10100000)
- AC: [(run/grandeza, amplitude)] = [(0,1),(0,1),(0,-1),(1,1), (7,-1), EOB]

Neste caso o resultado final da codificação do primeiro bloco é representado do seguinte modo, onde é possível observar o coeficiente DC codificado, seguido dos 5 coeficientes AC e o bloco de terminação EOB:

111111011010100000 00 001 000 11001 111110100 1010

Nas figuras seguintes podemos observar as imagens obtidas dos componentes DCT e das imagens decodificadas, para fatores de qualidade de 25%, 50% e 75%, respetivamente:

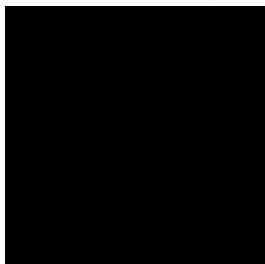


Figura 13: DCT e decodificação para um fator de qualidade de 25%

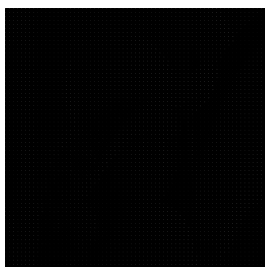


Figura 14: DCT e decodificação para um fator de qualidade de 50%

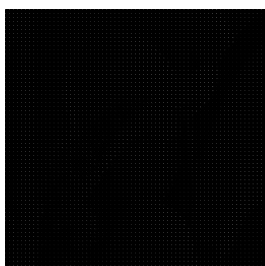


Figura 15: DCT e decodificação para um fator de qualidade de 75%

Como é possível observar nas imagens, mesmo com uma fator de qualidade de 25% a imagem decodificada apresenta uma boa qualidade. De seguida, foram realizado os cálculos das Taxas de Compressão e das SNR's para os diferentes valores de compressão, sendo possível observar os valores na Tabela 2:

Fator de Compressão	25%	50%	75%
Relação Sinal-Ruído	28.06	30.17	32.24
Taxa de Compressão	5.34	4.61	4.26
Tempo de Codificação	1.53	2.16	3.06
Tempo de Decodificação	1.63	2.06	2.91

Tabela 2: Resultados obtidos

Observado os resultados obtidos na Tabela 2 podemos retirar as seguintes conclusões:

- No que diz respeito aos valores obtidos na SNR podemos confirmar que são idênticos aos valores do capítulo 2.3, logo, não existiu perda de informação na codificação de entropia, apenas na quantificação. Ou seja, os resultados são os esperados.
- No que diz respeito à Taxa de Compressão seria de esperar que à medida que o fator de qualidade aumentasse a Taxa de Compressão diminuiria, visto que, quanto mais aproximada a imagem decodificada for da imagem original, mais o valor da Taxa de Compressão se aproxima de 1.
- Por fim, os Tempos de Codificação/Decodificação são relativamente baixos, salientando o facto de existir um aumento proporcional ao fator de qualidade, isto devesse ao facto de existirem mais amostras para quantificar.

É possível ainda reforçar o que foi referido nos pontos anteriores com o gráfico presente na Figura 15, que representa a Taxa de Compressão em função da SNR.

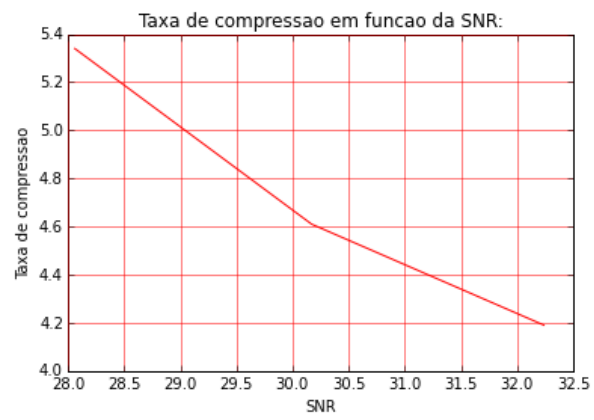


Figura 16: Taxa de Compressão em função da SNR

3 Conclusões

Com a finalização do trabalho e os resultados obtidos analisados podemos concluir que o trabalho foi implementado com sucesso. Os valores referentes às SNR's e Taxas de Compressão são os pretendidos e mesmo com um fator de compressão de 25% é possível obter uma qualidade de imagem muito aproximada da original, apenas quando se desce o fator de qualidade para os 10/15% é que se deixa de ter uma qualidade aceitável. No que diz respeito aos tempos de processamento estes mantiveram-se relativamente baixos, como referido anteriormente, sendo que podem existir otimizações no desenvolvimento do código de modo a reduzir ainda mais estes tempos. Em suma o grupo obteve conhecimentos referentes à compressão de imagens digitais e o funcionamento da norma JPEG.