**Multimédia**

**RELATÓRIO DO TRABALHO PRÁTICO 1**

**COMPRESSÃO DE IMAGEM**

Diogo Miguel Henriques Correia

uc2016219825@student.uc.pt

Telmo Filipe Queirós da Silva Correia

uc2019224775@student.uc.pt

Engenharia Informática

3.º Ano da Licenciatura

**Faculdade de Ciências e Tecnologia da Universidade de Coimbra**

**2.º Semestre - 2021/2022**

**Uma imagem com texto

Descrição gerada automaticamente**

**Índice**

Introdução

Exercício 1

Analise de resultados

Exercício 2

Exercício 3.1

Exercícios 3.2 e 3.3

Exercício 3.4

Exercício 3.5

Resultados

Exercício 4

Exercício 5

Resultados

Analise de resultados

Exercício 6

Exercício 7

Resultados para casos teste:

1. Canal completo:
2. Em blocos 8x8:
3. Em blocos 64x64:

Analise de resultados

Exercício 8

Resultados

Analise de resultados

Exercício 9

Resultados

Analise de resultados

Exercício 10

Resultados

Analise de resultados

Conclusão

**Introdução**

Este trabalho tem como objeto de estudo questões fundamentais de compressão de imagem destrutiva, onde é abordado uma versão do algoritmo JPEG.

Existem também conceitos teóricos que se abordam sendo o caso dos modelos de cor.

**Exercícios**

**Exercício 1**

Objetivo: Comprimir imagens no formato bmp para jpeg, com vários fatores de qualidade (alto 75%, medio 50%, baixo 25%) e comparar os resultados obtidos.

Tabela com os tamanhos das imagens:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Original | Compressão alta (75%) | Compressão media (50%) | Compressão baixa (25%) |
| barn\_mountains | 348Kb | 55,3Kb | 21,5Kb | 13,5Kb |
| logo | 411Kb | 12,4Kb | 7,6Kb | 6,18Kb |
| peppers | 576Kb | 53,9Kb | 20,3Kb | 13,3Kb |

Uma imagem com texto

Descrição gerada automaticamentePara calcular a taxa de compressão recorreu-se à seguinte fórmula:

Tabela com as taxas de compressão das imagens em relação à original

|  |  |  |  |
| --- | --- | --- | --- |
|  | Taxa de compressão (75%) | Taxa de compressão (50%) | Taxa de compressão (25%) |
| barn\_mountines | 6,29 | 16,19 | 25,78 |
| logo | 33,15 | 54,08 | 66,50 |
| peppers | 10,69 | 28,37 | 43,31 |

**Análise de resultados:** É possível ver que a taxa de compressão e a qualidade são influenciadas pelo fator qualidade, pois para todos os casos, quando o fator de qualidade diminui, a própria qualidade da imagem piora, pois existe muita compressão destrutiva. Ao existir muita compressão a diferença entre o tamanho da imagem original e da imagem comprimida aumenta o que torna a taxa de compressão mais elevada. Em relação às imagens em si podemos verificar que à medida que a taxa de compressão aumenta a imagem torna-se mais desfocada sendo que esta alteração é mais notável nas imagens “barn\_mountains” e “peppers”. Uma vez que a imagem “logo” contém muitos pixéis brancos consegue-se comprimir mais e, por isso, para todos os fatores de qualidade a sua taxa de compressão é muito mais alta.

**Exercício 2**

Objetivo: Desenvolver as funções encoder e decoder que irão encapsular as funções a desenvolver posteriormente.

**Exercício 3.1**

Objetivo: Desenvolver uma função para ler imagens no formato bmp.

**Exercício 3.2 e 3.3**

Objetivo: Desenvolver uma função que implemente um colormap a definir pelo utilizador e que permita a sua visualização.

Funções desenvolvidas: visualizar\_img\_colormap(img, nome,inicio, fim, niveis)

(print das funções)

**Exercício 3.4**

Objetivo: Desenvolver uma função que separe os canais RGB e a sua função inversa

Funções desenvolvidas: separar\_canais(img), juntar\_canais(r, g, b, converter=True)

(print das funções)

**Exercício 3.5**

**Resultados**

**(meter as imagens)**

**Exercício 4**

Objetivos: Desenvolver uma função para padding de imagens com dimensão múltipla de 16x16. sendo que as imagens que não cumpram esta condição tenho a replicação das últimas linhas e/ou colunas, até a satisfazerem. Desenvolver também a função inversa de maneira a obter a imagem original.

Funções desenvolvidas: padding(img), reverse\_padding(img, nl, nc)

(print das funções)

**Exercício 5**

Objetivo: Desenvolver 2 funções, uma que converta a imagem do modelo RGB para o modelo YCbCr e eu outra que convertae imagem do modelo YCbCr para o RGB.

Funções desenvolvidas: rgb\_ycbcr(img), ycbcr\_rgb(img)

(print das funções)

Uma imagem com texto, escuro

Descrição gerada automaticamenteUma imagem com texto, verde, exuberante

Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamente**Resultados:**

Fig. 3 – Canal B

Fig. 2 – Canal G

Fig. 1 – Canal R

Uma imagem com texto, exterior, edifício, branco

Descrição gerada automaticamenteUma imagem com texto, exterior, branco, montanha

Descrição gerada automaticamenteUma imagem com texto, exterior, casa, antigo

Descrição gerada automaticamente

Fig. 6 – Canal Cr

Fig. 5 – Canal Cb

Fig. 4 – Canal Y

**Análise de resultados:** Uma vez que o modelo RGB tem muita redundância de luminância e crominância nos 3 canais a transformação deste modelo no modelo YCbCr permite descorrelacionar os canais e aumentar a compressibilidade. Como é possível verificar através das imagens anteriores o canal Y apresenta mais detalhe do que as imagens obtidas pelos canais Cb e Cr. Isto deve-se ao facto de o olho humano ser mais sensível a variações de luminância do que de crominância. Também é possível verificar que as imagens do canal R e G apresentam mais luminância que a imagem do canal B, pelo facto das primeiras terem mais detalhe o que se pode confirmar com a comparação com a imagem do canal Y.

Isto beneficia o algoritmo jpeg pois permite que elimine informação redundante presente nos canais de crominância, como se irá ver nos passos seguintes.

**Exercício 6**

Objetivo: Desenvolver 2 funções, uma função para fazer subamostragem dos canais Y Cb Cr e outra para efetuar na função inversa.

Funções desenvolvidas: downsampling(img, caso), upsampling(y\_d, cb\_d, cr\_d)

(print das funções)

**Resultados para casos teste de downsampling (4, 2, 2) e (4, 2, 0):**

**(meter imagens)**

**Análise de resultados:** Uma vez que o downsampling vai ser aplicado apenas aos canais Cb e Cr a compressão apenas se vai notar nestes dois últimos. Ao fazer a variante 4:2:2 vamos reduzir a taxa de amostragem dos canais Cb e Cr para metade na direção horizontal. Ao fazer a variante 4:2:0 vamos reduzir para metade nas direções vertical e horizontal. Uma vez que esta redução é feita eliminando alternadamente linhas e colunas e, ao fazer o upsampling vamos duplicar linhas e colunas não vamos garantir que os valores sejam iguais aos valores presentes antes de ter sido aplicado o downsampling o que faz com que, nesta fase, o algoritmo seja destrutivo. Relativamente à taxa de compressão, vamos obter um melhor resultado ao usar a variante 4:2:0 pois estamos a reduzir o tamanho da imagem ao nível de linhas e colunas ao contrário da variante 4:2:2 que vai apenas reduzir o tamanho ao nível das colunas. Consecutivamente, a imagem reconstruída depois de usar o downsampling 4:2:0 não vai ser tão realista comparativamente ao uso da variante 4:2:2 pois iremos ter mais destrutividade.

Taxas de compressão do downsampling para os casos de teste e caso não seja aplicado

4:4:4 (sem subamostragem) = 1 – ((4 + 4 + 4) / 12) = 0 = 0 %

4:2:2 = 1 – ((4 + 2 + 2) / 12) = 0,33 = 33 %

4:2:0 = 1 – ((4 + 2 + 0) / 12) = 0,50 = 50 %

**Exercício 7**

Objetivo: Desenvolver funções para o calculo da transformada discreta do cosseno (DCT) e a sua inversa aplicando de pois a vários casos de teste (aos canais completos da imagem e depois a blocos 8x8 e blocos 64x64).

Funções desenvolvidas: dct(y, cb, cr), dct\_inverso(y\_dct, cb\_dct, cr\_dct), dct\_em\_blocos(canal, bloco), dct\_inversa\_em\_blocos(canal, bloco):

(print das funções)

**Resultados para casos teste:**

1. **Canal completo:**
2. **Em blocos 8x8:**
3. **Em blocos 64x64:**

**(meter imagens)**

**Análise de resultados:** Ao aplicar-se a DCT á imagem toda é possível que se perda potencial de compressão. Ao analisar-se as imagens obtidas verifica-se que estas têm altas frequências presentes, o que se deve às mudanças de cor. Logicamente deduzimos que aplicando a DCT a blocos de imagem mais pequenos iriamos fazer um melhor aproveitamento no que diz respeito ao potencial de compressão.

Nota se também que a dimensão do canal também influencia a suavidade da imagem, sendo que a DCT do canal Y é menos pixalizada que as dos canais Cb e Cr. (consequência do downsampling da alínea anterior)

Ao aplicar-se a DCT a blocos 8x8 e fazer a analise das imagens obtidas nota-se que a imagem é mais suave, é possível destingir os contornos da imagem original. Estes resultados confirmam a tese proposta na alínea anterior. Concluímos que a divisão em blocos evita a concentração de altas frequências, resultantes das mudanças de cor.

Ao aplicar-se DCT a blocos 64x64 e fazer a analise das imagens obtidas nota-se já não se não se distinguem os contornos da imagem original, embora a esta dimensão do bloco seja melhor que a imagem toda, os blocos 8x8 ainda tem um poder compressivo mais alto.

**Exercício 8**

Objetivo: Desenvolver as funções que façam a quantização dos coeficientes DCT para os blocos 8x8 aplicando o fator de qualidade, e sua função inversa.

Funções desenvolvidas: quantizacao\_Qualidade(qf, canal, Y=True), inversa\_quantizacao\_Qualidade(canal, qsY, qsCbCr, Y=True) (print das funções)

**Resultados**

**(meter imagens)**

**Análise de resultados:** É possível verificar ao que existe uma relação entre os fatores de qualidade e o potencial de compressão, podendo chegar se á conclusão de que quanto maior o fator de qualidade menor vai ser o potencial de compressão.

Ao fazer se uma análise mais detalhada, verificamos que para todos os blocos em que se aplica a quantização obtemos um elevado numero de “0” o que faz com que caso sejam aplicados algoritmos aritméticos ou códigos de Huffman

**Exercício 9**

Objetivo: Desenvolver as funções que façam a codificação DPCM dos coeficientes DC para cada bloco 8x8 e sua função inversa.

Funções desenvolvidas: codificacao\_dpcm(matriz,bloco), inversa\_codificacao\_dpcm(matriz,bloco)

(print das funções)

**Resultados**

**(meter imagens)**

**Análise de resultados:** Ao utilizarmos a codificação DPCM dos coeficientes DC vamos estar a reduzir a gama de valores utilizada e a variância o que faz com que a entropia seja menor permitindo assim uma maior compressibilidade ao utilizar uma codificação de Huffman ou aritmética, por exemplo.

**Exercício 10**

*barn\_moutains.bmp*

Uma imagem com texto, montanha

Descrição gerada automaticamenteUma imagem com texto, exterior

Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=10

Fig. – Imagem Reconstruída. qf=10

Uma imagem com texto, montanha, de madeira

Descrição gerada automaticamente

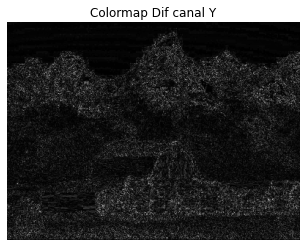
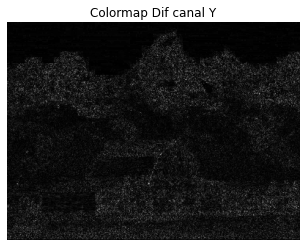


Fig. – Imagem Reconstruída. qf=25

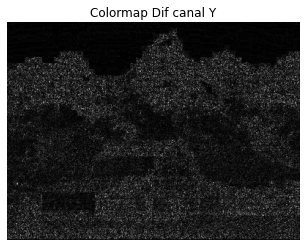
Fig. – Diferenças no canal Y. qf=25

Uma imagem com texto, montanha, casa, de madeira

Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=50

Fig. – Imagem Reconstruída. qf=50

Uma imagem com texto, montanha, de madeira

Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=75

Fig. – Imagem Reconstruída. qf=75

Uma imagem com texto, céu noturno

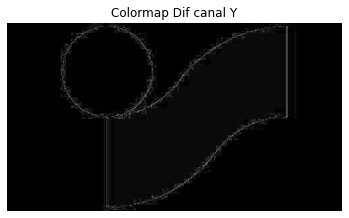
Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=100

Fig. – Imagem Reconstruída. qf=100

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 10 | 25 | 50 | 75 | 100 |
| MSE | 745.10 | 430.2 | 293.21 | 187.51 | 55.54 |
| RMSE | 27.3 | 20.74 | 17.12 | 13.69 | 7.45 |
| SNR | 18.46 | 20.85 | 22.51 | 24.46 | 29.74 |
| PSNR | 19.41 | 21.79 | 23.46 | 25.4 | 30.68 |

*logo.bmp*

Uma imagem com ClipArt

Descrição gerada automaticamente

Fig. – Imagem Reconstruída. qf=10

Fig. – Diferenças no canal Y. qf=10

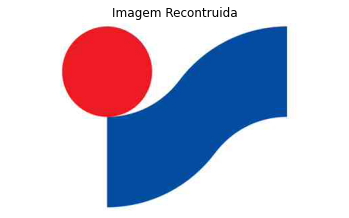
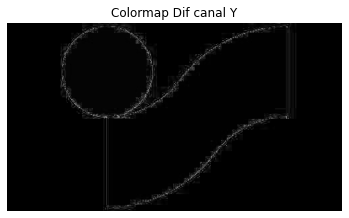


Fig. – Diferenças no canal Y. qf=25

Fig. – Imagem Reconstruída. qf=25

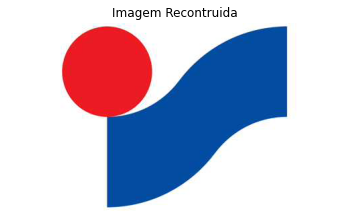
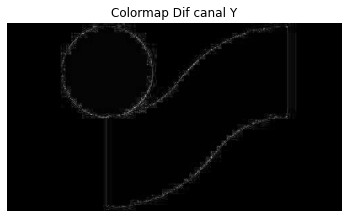
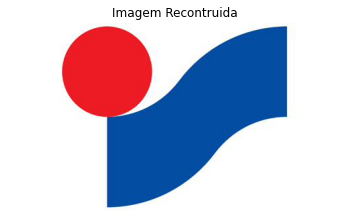


Fig. – Diferenças no canal Y. qf=50

Fig. – Imagem Reconstruída. qf=50



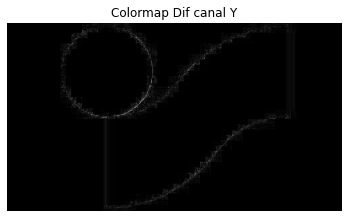


Fig. – Diferenças no canal Y. qf=75

Fig. – Imagem Reconstruída. qf=75

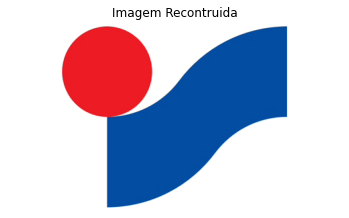
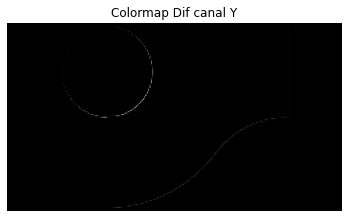
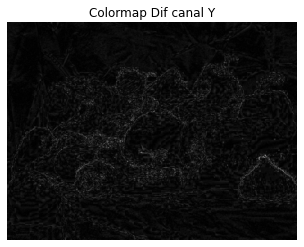


Fig. – Diferenças no canal Y. qf=100

Fig. – Imagem Reconstruída. qf=100

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 10 | 25 | 50 | 75 | 100 |
| MSE | 207.82 | 100.94 | 81 | 60.74 | 43.07 |
| RMSE | 14.42 | 10.05 | 9 | 7.79 | 6.56 |
| SNR | 28.16 | 31.29 | 32.25 | 33.5 | 34.99 |
| PSNR | 24.95 | 28.09 | 29.05 | 30.3 | 31.79 |

*peppers.bmp*

Uma imagem com texto, fruta, diferente, vegetal

Descrição gerada automaticamente

Fig. – Imagem Reconstruída. qf=10

Fig. – Diferenças no canal Y. qf=10

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto, interior, vegetal, diferente

Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=25

Fig. – Imagem Reconstruída. qf=25

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto, interior, vegetal, diferente

Descrição gerada automaticamente

Fig. – Imagem Reconstruída. qf=50

Fig. – Diferenças no canal Y. qf=50

Uma imagem com texto, branco

Descrição gerada automaticamenteUma imagem com texto, interior, vegetal, diferente

Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=75

Fig. – Imagem Reconstruída. qf=75

Uma imagem com texto

Descrição gerada automaticamenteUma imagem com texto, interior, vegetal, diferente

Descrição gerada automaticamente

Fig. – Diferenças no canal Y. qf=100

Fig. – Imagem Reconstruída. qf=100

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 10 | 25 | 50 | 75 | 100 |
| MSE | 366.56 | 188.40 | 133.52 | 104.56 | 62.28 |
| RMSE | 19.15 | 13.73 | 11.55 | 10.23 | 7.89 |
| SNR | 19.28 | 22.18 | 23.67 | 24.73 | 26.98 |
| PSNR | 22.49 | 25.38 | 26.88 | 27.94 | 30.19 |

**Analise de resultados:** Através da visualização da imagem com os erros de forma visual podemos confirmar que quando o fator de qualidade é demasiado baixo (10) temos uma quantidade de alterações maior sendo possível visualizar algum ruido nas imagens. Quando o fator de qualidade aumenta a quantidade de alterações vai diminuindo, e as alterações da imagem reconstruída deixam de ser percetíveis. Numa imagem com o fator de qualidade demasiado elevado (100) temos alterações mínimas nas imagens reconstruídas, confirmadas pela imagem de erro quase toda preta para o canal Y. Em relação ao MSE como já era esperado os valores diminuem conforme o fator de qualidade aumenta, por sua vez o SNR dimimui.