

Relatório

## Experiência 5 - Codificação de imagens segundo o padrão JPEG

PSI3531 - Processamento de Sinais Aplicado (2020)

Matheus Bordin Gomes - 9838028

Essa experiência visa a implementação de princípios básicos do padrão JPEG de compressão de imagens, utilizando transformação pela DCT e codificação de Huffman.

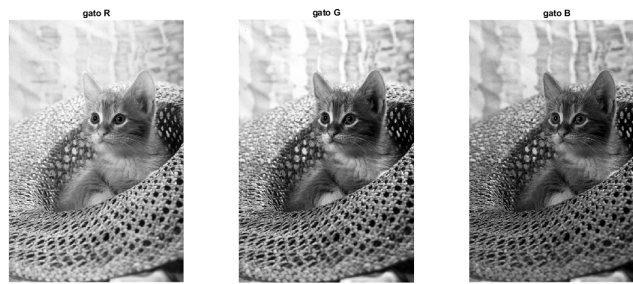
### 1. Sub-amostragem do sinal de cromaticidade

Esse exercício foi implementado no *script* "exp5\_1.m", entregue juntamente com este relatório. Nessa etapa, foi implementada a sub-amostragem dos sinais de cromaticidade, além da recuperação da imagem a partir desses sinais.

1. Pode-se ver na figura 1 a imagem original utilizada. Além disso, na figura 2 pode-se ver cada uma das componentes RGB separadamente. Por fim, na figura 3 é possível ver as componentes YCbCr obtidas após a transformação.



Figura 1. Imagem original utilizada no programa.

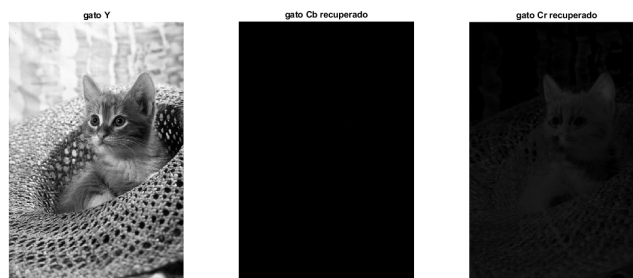


**Figura 2.** Componentes RGB da imagem original.



**Figura 3.** Componentes YCbCr da imagem original obtidas após as transformações.

2. Após a sub-amostragem dos canais Cb e Cr, foi obtida uma taxa de compressão de 49.98% em relação ao tamanho original.
3. Seguem as componentes YCbCr da imagem recuperada e as componentes RGB obtidas após a anti-transformação nas figuras 4 e 5, respectivamente.



**Figura 4.** Componentes YCbCr da imagem recuperada após a subamostragem dos canais Cb e Cr.



**Figura 5.** Componentes RGB da imagem recuperada.

4. A imagem recuperada pode ser vista na figura 6. Quanto a comparação das imagens, obteve-se um erro quadrático médio (MSE) = 1.4949 e um *peak signal-to-noise ratio* (PSNR) = 46.3848 dB.



**Figura 6.** Imagem recuperada.

## 2. Quantização da DCT

Esse exercício foi implementado no *script* "exp5\_2.m", entregue juntamente com este relatório. Nessa etapa, foi implementada a transformação DCT dos sinais YCbCr e a quantização dos coeficientes da transformada.

1. A implementação desse item pode ser vista no *script* mencionado anteriormente, nas linhas 51 a 53.
2. A implementação desse item pode ser vista no *script* mencionado anteriormente, nas linhas 72 a 77.
3. A implementação desse item pode ser vista no *script* mencionado anteriormente, nas linhas 79 a 93.
4. Pode-se ver na figura 7 as componentes YCbCr obtidas na recuperação da imagem, após a sub-amostragem, a transformação DCT e a quantização dos coeficientes da transformada. Já as componentes RGB da imagem recuperada podem ser vistas na figura 8. Por fim, a imagem recuperada está disponível na figura 9.



**Figura 7.** Componentes YCbCr da imagem recuperada após a subamostragem dos canais Cb e Cr, utilizando a quantização dos coeficientes da DCT.



**Figura 8.** Componentes RGB da imagem recuperada, utilizando a quantização dos coeficientes da DCT.



**Figura 9.** Imagem recuperada, utilizando a quantização dos coeficientes da DCT.

Quanto a comparação das imagens, obteve-se um erro quadrático médio ( $MSE$ ) = 33.2848 e um *peak signal-to-noise ratio* ( $PSNR$ ) = 32.9083 dB. Dessa forma, temos uma relação  $PSNR$  menor do que no caso anterior, porém ainda é satisfatória visto que valores de  $PSNR$  maiores ou iguais a 30 dB são dados como razoáveis na apostila da disciplina sobre o tema. Vale notar que esses resultados foram obtidos para  $k = 1$ ;

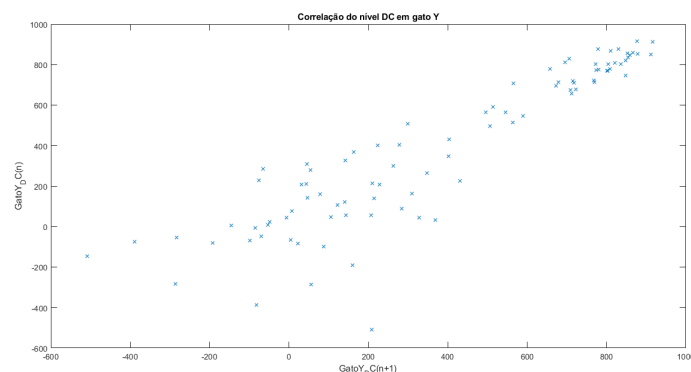
5. A razão entre o número de pixels da imagem original e o número de coeficientes nulos da DCT, considerando-se todos os canais, é de 2.2758. Em um caso ideal, em que os coeficientes

nulos não precisassem ser armazenados, poderíamos descartar o inverso dessa razão em dados desnecessários, ou seja, 43,94% dos dados poderiam ser descartados. Porém, isso não é possível, visto que é necessária a informação de quais coeficientes foram descartados para recuperar a imagem. c

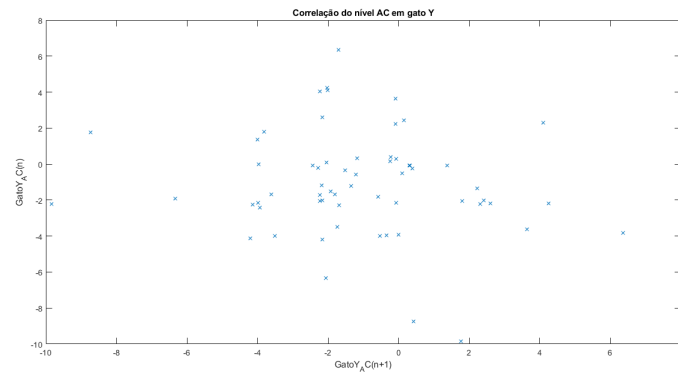
### 3. Cálculo da entropia do sinal

Esse exercício foi implementado no *script* "exp5\_3.m", entregue juntamente com este relatório. Nessa etapa, foi calculada a entropia da imagem e foi feita uma estimativa do tamanho que a compressão JPEG da imagem irá ocupar.

1. O cálculo da probabilidade de cada valor de código dos coeficientes DC aparecer pode ser visto nas linhas 110 e 118 do *script* supracitado. Já o cálculo referente a probabilidade de cada valor de código dos coeficientes AC pode ser visto nas linhas 123 a 131 do mesmo *script*.
2. A entropia resultante dos coeficientes DC é de 7.9378 bits, enquanto a entropia resultante dos coeficientes AC é de 0.9626 bits. Isso mostra que o comprimento médio da informação DC é maior do que o comprimento médio da informação AC. Vale notar que esse resultado foi obtido para  $k = 1$ . Para valores maiores de  $k$ , a entropia seria menor.
3. Multiplicando o tamanho médio dos coeficientes DC pelo número de coeficientes DC na imagem e multiplicando o tamanho médio dos coeficientes AC pelo número de coeficientes AC na imagem, obtemos a quantidade de bits estimada para cada um dos casos. No total, seriam necessários 586.77 kb para armazenar a imagem usando codificação de Huffman. Vale notar que esse resultado foi obtido para  $k = 1$ . Para valores maiores de  $k$ , o tamanho seria menor, visto que a entropia seria menor.
4. Na figura 10, pode-se observar dos coeficientes DC dos blocos em função dos coeficientes DC dos blocos seguintes. É notável que existe uma correlação significativa entre os valores, que tendem a formar uma reta. Já na figura 11, pode-se notar os valores de um coeficiente AC dos blocos em função do mesmo coeficiente dos blocos seguintes. Pela grande dispersão dos pontos, é notável que não há correlação significativa entre os coeficientes AC.



**Figura 10.** Valores dos coeficientes DC de um bloco em função dos valores dos coeficientes DC dos blocos seguintes.



**Figura 11.** Valores de um coeficiente AC de um bloco em função do valores dos coeficientes AC dos blocos seguintes.