



UNIVERSIDADE DE AVEIRO

DEPARTAMENTO DE ELECTRÓNICA, TELECOMUNICAÇÕES E  
INFORMÁTICA

47053- COMPUTAÇÃO VISUAL

---

# Imagens RGB - Histogramas

## Implementação em OpenCV e Qt

---

8240 - MESTRADO INTEGRADO EM ENGENHARIA DE  
COMPUTADORES E TELEMÁTICA

António Rafael da  
Costa Ferreira  
NMec: 67405

Rodrigo Lopes  
da Cunha  
NMec: 67800

Docente: Joaquim João Estrela Ribeiro Silvestre Madeira

Janeiro de 2016  
2015-2016

# Conteúdos

1	Introdução . . . . .	2
2	Instalação e Projeto Qt . . . . .	3
3	Implementação . . . . .	3
3.1	Ler imagens . . . . .	3
3.2	Visualizar as diferentes componentes RGB . . . . .	4
3.3	Visualizar a apenas a Luminância . . . . .	4
3.4	Guardar uma imagem ou um histograma . . . . .	5
3.5	Abrir um novo histograma para comparação . . . . .	6
3.6	Mudar o contraste da imagem . . . . .	6
3.7	Mudar o brilho da imagem . . . . .	7
3.8	Contrast Limited Adaptive Histogram Equalization . .	7
3.9	Histogram Equalization . . . . .	8
4	Conclusão . . . . .	9

## 1 Introdução

O trabalho proposto para o projeto da unidade curricular de Computação Visual é um Programa OpenCV com Qt que produza histogramas (R, G, B e luminância) a partir de imagens RGB. Também foi proposto que fosse possível fazer alterações nas imagens e guardar em disco, assim como os seus histogramas. Os histogramas podem ser guardados em ficheiros de texto com os sucessivos valores ou como imagens.

Para elaborar a interface gráfica foi proposto que usássemos QT para fazer todos os controlos e opções disponíveis.

Foi pensada uma implementação baseada na experiência do utilizador ao utilizar a aplicação, para isso desenvolveu-se uma interface atraente e com algumas funcionalidades, tanto como de visualização como de opções de manipulação do histograma.

O relatório reflete todos os passos e decisões tomadas na criação da implementação, assim como uma explicação do que foi usado e da interface do utilizador.

## 2 Instalação e Projeto Qt

Para instalar e usar o nosso programa é preciso ter as bibliotecas do Qt 5.4.2 GCC 64bits e o OpeCV 2.4.9. A preparação da máquina virtual para usar foi demorada e um pouco difícil para compilar as duas bibliotecas de forma correta. Foi usado um tutorial<sup>1</sup> que explica passo a passo como fazer essa instalação, no entanto a máquina que se usou foi uma instalação de um colega do mesmo ano e do curso.

Para iniciar o programa:

```
# cd RGBHistogram
# ./RGBHistogram
```

Para abrir o Projeto Qt, abrir o ficheiro: RGBHistogram.pro

## 3 Implementação

A implementação permite que o utilizador leia, visualize e salvaguarde imagens e histogramas, visualize as imagens das diferentes componentes, determine, visualize os histogramas da luminância e aplique diferentes operações baseadas nesses histogramas.

### 3.1 Ler imagens

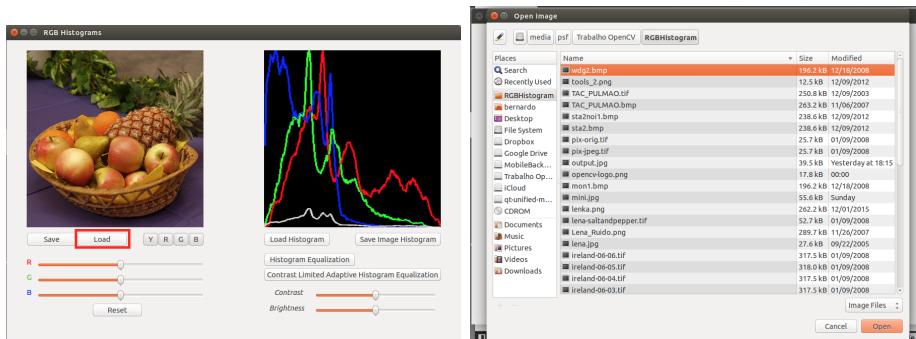


Figura 1: Abrir uma nova imagem

Para abrir uma nova imagem o utilizador deve clicar em "Load". Irá aparecer uma janela que permitirá escolher uma imagem dos seguintes formatos: \*.png, \*.jpg, \*.bmp, \*.tif. Após isso irá aparecer uma nova janela com a nova imagem.

<sup>1</sup> <https://www.youtube.com/watch?v=qA46fvP305A>

### 3.2 Visualizar as diferentes componentes RGB

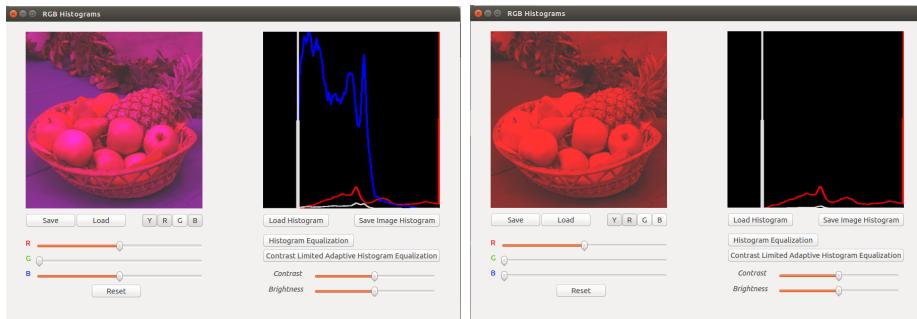


Figura 2: Visualizar as diferentes componentes de forma separada

Foi dada a possibilidade ao utilizador de selecionar as componentes que pretende visualizar, assim como os histogramas das mesmas. Nos botões "R", "G", "B", que são do tipo "checkable", se estiverem selecionados irá ser apresentado a componente correspondente, sempre que não estiverem selecionados, essa componente não irá aparecer.

Por exemplo, na primeira imagem da figura 3, apenas estão selecionadas as componentes vermelho e azul e por isso a componente verde não aparece nem na imagem, nem no histograma.

Na segunda imagem da mesma figura, apenas está selecionada a componente vermelha, sendo que apenas esta aparece na imagem e no histograma.

#### Implementação

Para implementação do histograma com as diferentes componentes foi usado um tutorial<sup>2</sup> do OpenCV que explica como se deve dividir as diferentes componentes e criar assim histogramas para cada uma.

### 3.3 Visualizar a apenas a Luminância

O utilizador pode também apenas visualizar a luminância para uma determinada imagem. Para isso, tem de não selecionar os botões vermelho, verde e azul. Sendo assim, apenas o botão da luminância fica selecionado, aparecendo assim no histograma apenas a luminância.

#### Implementação

Para determinar a luminância<sup>3</sup> foi criado um array com 256 entradas para guardar cada valor de intensidade e inicializados a zero. Depois foram

<sup>2</sup> [http://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/histogram\\_calculation/histogram\\_calculation.html?highlight=histogram](http://docs.opencv.org/2.4/doc/tutorials/imgproc/histograms/histogram_calculation/histogram_calculation.html?highlight=histogram)

<sup>3</sup> <http://www.programming-techniques.com/2013/01/intensity-histogram-using-c-and-opencv.html>

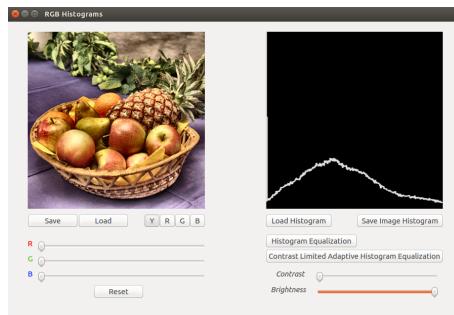


Figura 3: Visualizar apenas a luminância num histograma

calculados o número de pixels para cada valor de intensidade. De seguida foi encontrado o valor máximo de intensidade do histograma e foi também normalizado o histograma entre 0 e o número de linhas do histograma.

### 3.4 Guardar uma imagem ou um histograma

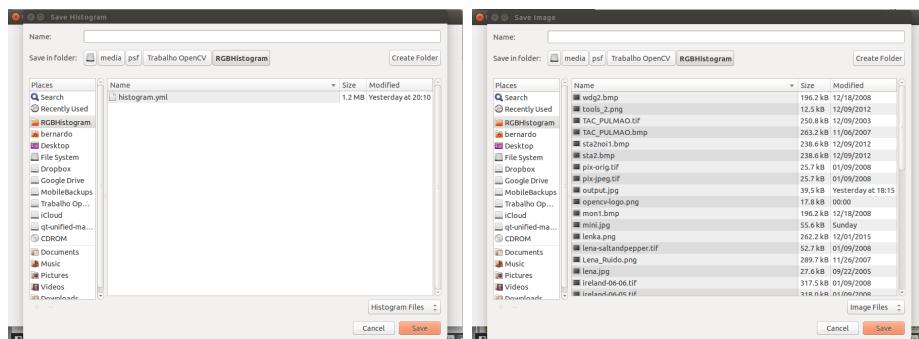


Figura 4: Janelas para o utilizador guardar uma imagem ou histograma

O utilizador pode exportar as imagens em diferentes formatos (\*.png, \*.jpg, \*.bmp, \*.tif) e guardar o histograma no formato \*.yml<sup>4</sup>.

<sup>4</sup> <http://stackoverflow.com/questions/10277439/opencv-load-save-histogram-data>

### 3.5 Abrir um novo histograma para comparação

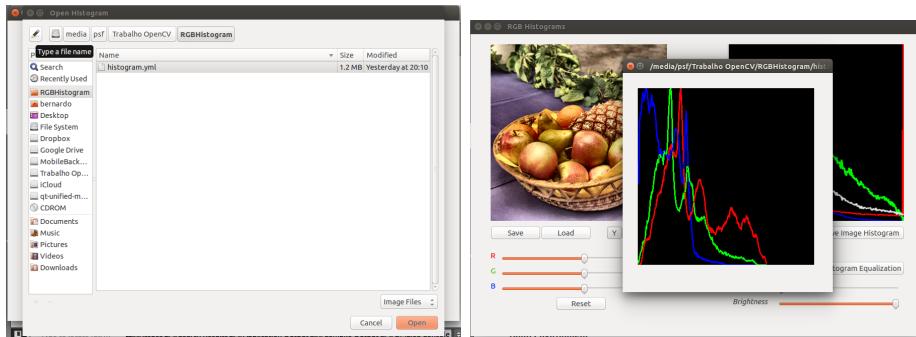


Figura 5: O utilizador abre o histograma numa nova janela

Foi dada a possibilidade de o utilizador abrir um histograma numa nova janela para conseguir comparar esse histograma com o que está a ser mostrado na imagem aberta na janela principal.

### 3.6 Mudar o contraste da imagem

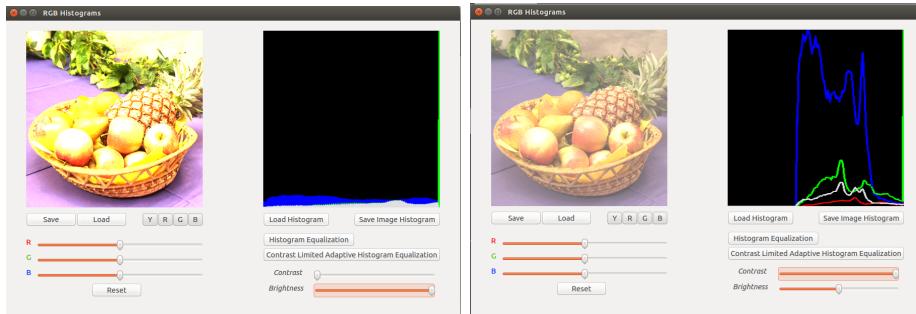


Figura 6: Mudar o contraste usando os sliders

Caso o utilizador pretenda, pode alterar o contraste da imagem, por exemplo, na figura 6 o utilizador pode selecionar qual o contraste que a imagem apresenta, se o máximo, se o mínimo contraste ou outro valor.

### 3.7 Mudar o brilho da imagem

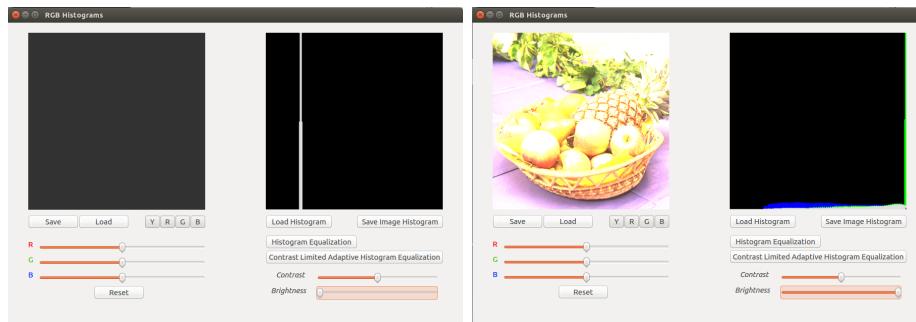


Figura 7: Mudar o brilho usando os sliders

Também foi dada a possibilidade para que dado o utilizado pretenda, pode alterar o brilho da imagem, por exemplo, na figura 9 o utilizador pode selecionar qual o brilho que a imagem apresenta, se o máximo, se o mínimo brilho ou outro valor.

### 3.8 Contrast Limited Adaptive Histogram Equalization

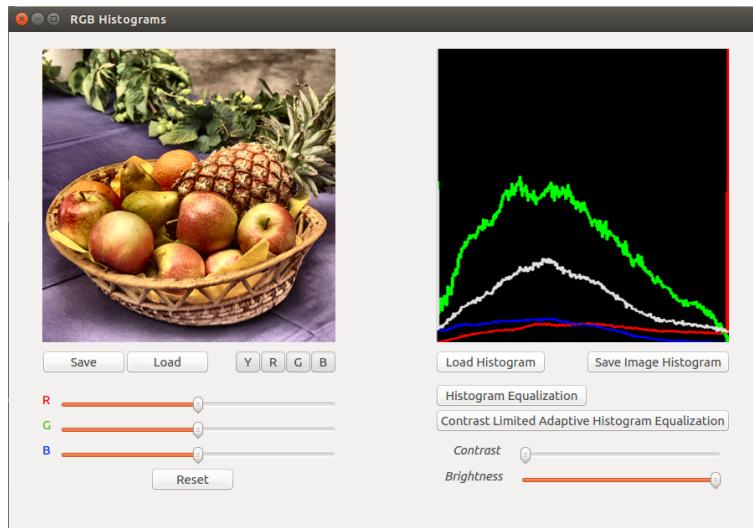


Figura 8: Contrast Limited Adaptive Histogram Equalization

Para manipulação da imagem e do histograma foi criada uma funcionalidade para o utilizador fazer a equalização do histograma de forma simples. A primeira funcionalidade<sup>5</sup> foi a "Contrast Limited Adaptive Histogram Equalization".

<sup>5</sup> [http://docs.opencv.org/master/d5/daf/tutorial\\_py\\_histogram\\_equalization.html](http://docs.opencv.org/master/d5/daf/tutorial_py_histogram_equalization.html)

ization".

### Implementação

A imagem é dividida em pequenos blocos chamados "tiles" e cada desses blocos são depois equalizados de forma igual. Portanto numa pequena área, o histograma vai confinar-se a uma pequena região, se existir ruído, então era ampliar-se. Para evitar isto, é então aplicado "contrast limiting".

Para esta funcionalidade foi usado código disponível no site do OpenCV, usando o CLAHE.

### 3.9 Histogram Equalization

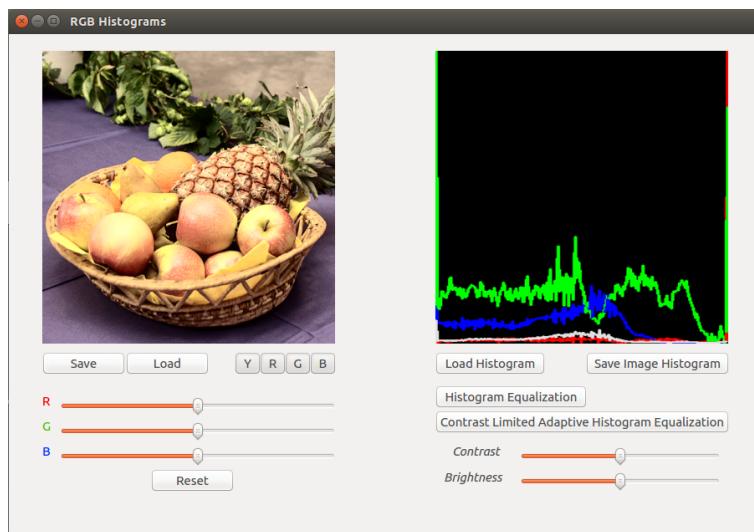


Figura 9: Histogram Equalization

A outra funcionalidade desenvolvida foi a equalização do histograma RGB. O utilizador, apenas com um clique, consegue equalizar o histograma e visualizar os seus resultados.

### Implementação

A equalização de um histograma RGB<sup>6</sup> não pode ser feito da mesma forma que um a preto e branco ou dividindo o R, G e B e equalizar cada uma das componentes, isso seria incorreto. A equalização envolve valores de intensidade da imagem e não pode ser aplicado diretamente à imagem. Necessita-se então de ser aplicada de uma maneira que os valores de intensidade são equalizados de maneira a que não modifiquem as cores da imagem. Portanto, o primeiro passo é converter a imagem RGB para um formato que

<sup>6</sup> <http://prateekvjoshi.com/2013/11/22/histogram-equalization-of-rgb-images/>

tenha mais valores sobre a intensidade e os separe, por exemplo HSV/HLS, YUV, YCbCr. Escolheu-se então o YCbCR porque está desenhado para imagens digitais. Depois de equalizado converte-se novamente para RGB.

## 4 Conclusão

O principal objetivo foi conseguido, inicialmente teve-se um pouco de receio ao usar Qt, devido à inexperiência com o mesmo e a dificuldade com que se conseguia juntar o Qt e o OpenCV. Não se pode deixar de agradecer aos colegas Bernardo Ferreira e ao Bruno Silva por disponibilizar a imagem do Parallels da máquina recém instalada o que nos permitiu avançar para a implementação, pois não tinha sido até então montar uma máquina.

Usou-se algum código externo para fazer alguma das funcionalidades implementadas, contudo, todas as fontes estão enunciadas no relatório.

Este trabalho permitiu ter uma diferente visão sobre a utilidade do OpenCV e mesmo da facilidade com que se cria uma interface com o Qt e se trabalha sobre a mesma, sendo assim, o projeto acabou por exceder as expetativas iniciais e foi divertido e interessante de ser realizado.