

## Aula Prática 2

Docente: Miguel Tavares Coimbra

### 1. RGB color space

- Build a function that can display the R, G, B individual components of an image.
- Hints:
  - i. Create 3 new images using objects of type *ImagePanel*.
  - ii. Create 3 button/functions, one per RGB dimension, which processes the visualized image.

### 2. HSI color space

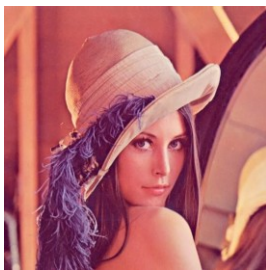
- Create a function that can display the H, S, I individual components of an image.
- Hints:

$$\text{i. } H = \begin{cases} \theta & \leftarrow B \leq G \\ 2\pi - \theta & \leftarrow B > G \end{cases} \quad \theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G) + (R-B)]}{\left[ (R-G)^2 + (R-B)(G-B) \right]^{1/2}} \right\}$$

$$\text{ii. } S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)]$$

$$\text{iii. } I = \frac{1}{3}(R+G+B)$$

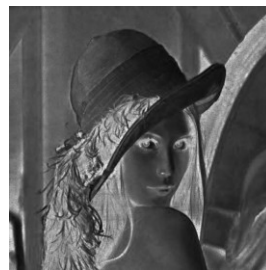
- There are values for which H and S are undefined. Why?



RGB



H



S



I

### 3. Pseudocolor

- Create a function that applies a pseudocolor algorithm to a black and white image. Use any technique you want (intensity slicing, intensity to color)



#### 4. Histograms

- Build a function that calculates the *Intensity (HSI)* histogram of an image. This should be displayed on a new window (*Frame*).
- Hints:
  - i.  $I = (R+G+B)/3$
  - ii. Change the function *paint* (*public void paint(Graphics g)*) of the new window, that should draw a histogram based on an internal variable of type *int[256]*.
  - iii. Use classes from *Java.awt.Graphics*: *drawLine*, *drawRect*, etc.

