

Ficha de Trabalho 3

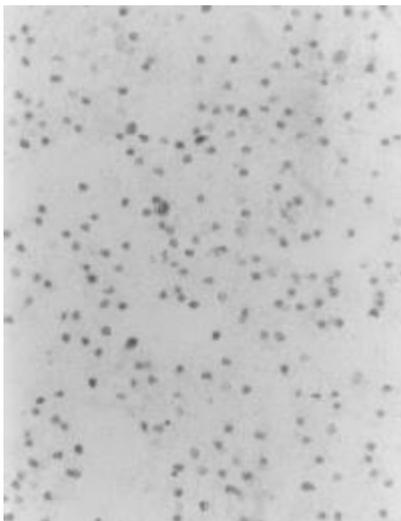
Docente: Miguel Tavares Coimbra

1. Histogramas.

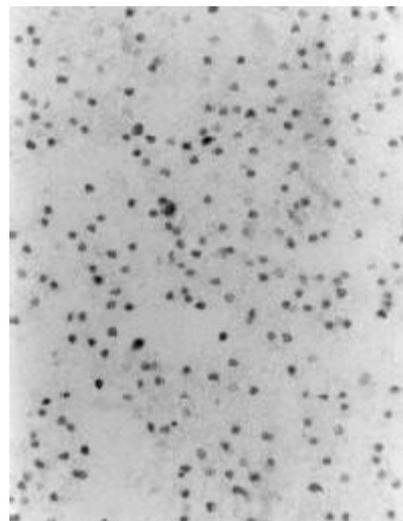
- Crie uma função que calcule o histograma da *Intensity (HSI)* de uma imagem.
[Sugestão] Crie um vector *histograma* de tamanho 256, iniciando-o a zero. Percorra a imagem, somando um ao *bin* correcto de cada histograma por cada pixel da imagem.
- Compare o histograma que obteve da imagem “*lena_intensity*” com a imagem “*Histogram_lena_intensity*” (ver imagens de apoio).
- [Opcional] Este deverá ser visualizado numa nova janela (*Frame*).
 - i. Trabalhe a função *paint (public void paint(Graphics g))* da nova janela, que deverá desenhar um histograma baseado numa variável interna do tipo *int[256]*.
 - ii. Classe *java.awt.Graphics: drawLine, drawRect, etc.*
- [Opcional] Crie uma função que calcule os três histogramas de cor (espaço RGB) de uma imagem. Estes deverão ser visualizados numa única janela.

2. Manipulação da gama dinâmica

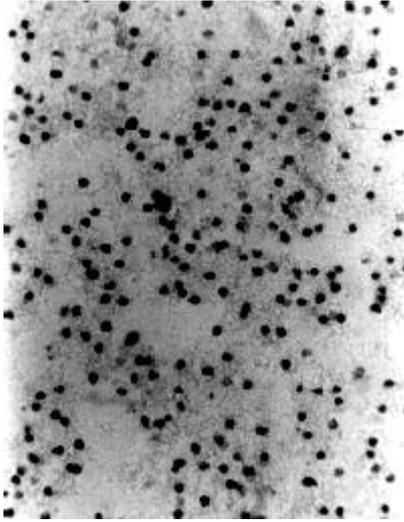
- Crie uma função que aplique a técnica de *contrast stretching* a uma imagem.
 - i. Use o valor mínimo e máximo do histograma.
 - ii. Use os valores entre 5% e 95% da distribuição de pontos no histograma (i.e. ignore 5% dos pontos de cada um dos lados do histograma).
- Aplique esta função à imagem “*dots.jpg*” e compare-a com a imagem “*dotsEnhancedContrast.jpg*”, onde foram usados os valores entre 5% e 95%.
- [Opcional] Crie uma função onde aplique a técnica de *histogram equalization*. Aplique-a à função “*dots.jpg*” e compare-a com os resultados anteriores.



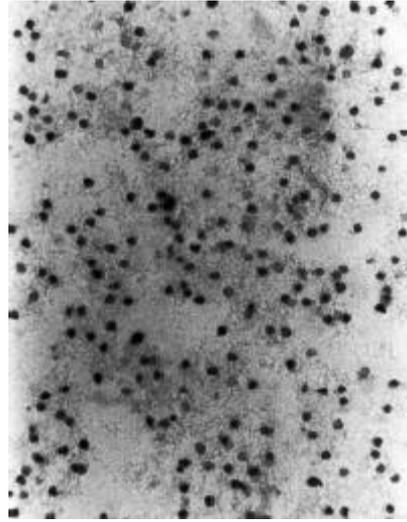
a) Dots.jpg



b) Contrast stretching (0-100%)



c) Contrast stretching (5%-95%).



d) Histogram equalization