

## Introdução

A segmentação de imagens tem como objetivo facilitar o entendimento de uma imagem para simplificar a sua análise. É uma etapa crucial em diversas aplicações de visão computacional, como reconhecimento de objetos, análise de imagens médicas e detecção de mudanças. Ela divide a imagem em regiões com o objetivo de extrair as partes homogêneas. A ideia é separar as informações para realizar uma análise precisa.

## Quais são as principais bibliotecas e frameworks para segmentação de imagens?

Dentre os principais bibliotecas e frameworks destacam se:

**OpenCV:** possui funções que permite implementar técnicas como segmentação por limiar, segmentação por região e detecção de bordas.

**Scikit-image:** Possui implementações de diversos algoritmos de segmentação, como Watershed, K-means e segmentação ativa.

## Aplicação

Os métodos tradicionais de segmentação de imagens incluem segmentação por limiar, baseada em bordas, baseada em regiões, baseada em cluster e baseada em cores. As técnicas modernas usam redes neurais e Deep Learning, como segmentação semântica e segmentação de instância. Neste estudo foi aplicado a segmentação por limiar. A segmentação por limiar transforma a **imagem em preto e branco**. Cada ponto dessa nova imagem (pixel) possui um valor numérico que representa sua intensidade de cinza.

A segmentação por limiar divide uma imagem em duas partes distintas com base em um valor de intensidade. Devemos definir um valor limiar que servirá de comparação com cada pixel. Para classificar identificar as regiões é definido valores para que ao ser comparado seja possível determinar se pertencem a uma região.

```
import cv2
from google.colab.patches import cv2_imshow

img = cv2.imread('panda.png', 0)

ret, thresh = cv2.threshold(img, 127, 255, cv2.THRESH_BINARY)

cv2_imshow(thresh) |
cv2.waitKey(0)
cv2.destroyAllWindows()
```



Após realizar a importação da biblioteca OpenCV e de carregar a imagem utilizando a função **cv2.imread**, este dado foi armazenado na variável **img**. O trecho do código **cv2.threshold(img, 127, 255, cv2.THRESH\_BINARY)** realiza a segmentação por limiar da imagem define o valor do limiar e dos pixels.