

---

**UNIVERSIDADE FEDERAL DE ALAGOAS**  
**INSTITUTO DE COMPUTAÇÃO**

**Visão Computacional**  
**Professor: Thales Vieira**

---

**4a lista de exercícios**

**04 de junho de 2022**

---

**Instruções:**

A lista deve ser respondida por grupos de até 2 pessoas (graduação) e individualmente (mestrado).

Resoluções idênticas de grupos distintos serão desconsideradas.

O código e as imagens devem ser anexadas a cada questão.

Data limite para entrega: 19/06/2022.

**1.** Leia o seguinte tutorial de calibração de câmera:

[https://docs.opencv.org/4.5.2/dc/dbb/tutorial\\_py\\_calibration.html](https://docs.opencv.org/4.5.2/dc/dbb/tutorial_py_calibration.html)

Adapte o código do tutorial e implemente um programa que funcione o mais próximo possível do tempo real com imagens adquiridas por uma web-cam ou smartphone. Você vai precisar de um tabuleiro de xadrez (se não tiver um, pode imprimir uma imagem numa folha A4, e colar num papelão ou emplastificar para a geometria ficar fixa). Exiba na tela a matriz intrínseca, e os vetores de rotação e translação da câmera em tempo real, enquanto varia a posição da câmera, mantendo o tabuleiro sempre visível na imagem. Salve o vídeo e o print da tela.

**2.** As fotografias em modo retrato se popularizaram nos últimos anos.

Elas consistem em segmentar *foreground* e *background* em uma fotografia, e borrar o *background*, simulando o efeito de uma câmera DSLR, como na imagem abaixo.



O objetivo dessa questão é simular este efeito, usando mapas de disparidade gerados por duas imagens. Para um tutorial de como gerar mapas de disparidade usando OpenCV, veja [https://docs.opencv.org/4.5.2/dd/d53/tutorial\\_py\\_depthmap.html](https://docs.opencv.org/4.5.2/dd/d53/tutorial_py_depthmap.html). Após a aquisição do mapa de disparidade, você deve usá-lo adequadamente para detectar o foreground, aplicar um filtro gaussiano para borrar apenas o background, e combinar as duas partes para obter um efeito como na figura acima. Experimente seu algoritmo em três pares de imagens adquiridas através de pequenas variações de ponto de vista da câmera.

**3.** Com a experiência (e códigos) resultantes das questões anteriores, o objetivo agora é realizar reconstrução 3D. Leia cuidadosamente o tutorial em três partes que pode ser encontrado com código em <https://becominghuman.ai/stereo-3d-reconstruction-with-opencv-using-an-iphone-camera-pair>. A parte 3 é a mais importante. Gere três exemplos de reconstruções 3D em formato .ply. Você deve submeter os arquivos ply e imagens com visualizações destas reconstruções, usando o software MeshLab, que pode ser baixado em <https://www.meshlab.net>.