Relatório PIV

Descrição Geral do Projeto

Neste projeto o objetivo é reconhecer objetos que se movem num fundo estático com base numa sequência de imagens de duas câmaras rgb+d (com perspectivas diferentes). O programa desenvolvido reconhece e segue os objetos que se movem num espaço em 3D desenhando uma caixa à volta de cada um nas sequências de imagens de ambas as câmaras. Esta caixa é atualizada ao longo da sequência de imagens e o programa deteta vários objetos simultaneamente.

Na primeira parte do projeto são-nos fornecidas as transformações entre as imagens das câmaras e sistema de coordenadas do mundo em 3D. Dado estes inputs o programa reconhece e segue os objetos que se movem.

Funções

**bg\_fg\_detect.m**: detecta o background usando filtros medianos e elimina o ruído. Para além disso para cada fotografia detecta o foreground e faz algum processing da imagem. No fim mostra o resultado (põem em 3D usando o get\_xyz\_asus.m)

**get\_T\_R\_world.m:** vai buscar as matrizes de transformação das coordenadas de uma câmara para a outra. Usa SIFT e RANSAC juntamente com Procrustes para fazer isto.

**sift\_match:** usa a biblioteca *VL\_FEAT* que contém o SIFT para ter matches de pontos entre as duas imagens. Depois retorna os melhores pontos.

**ransac\_iteration:** provavelmente vou renomeá-lo só “ransac” mas basicamente implementa o processo RANSAC. Ainda estou a laminar este

**minboundbox:** devolve os 8 pontos que minimizam a bounding box de um certo conjunto de pontos. A função plot\_min\_box faz plot dessa bodega

THAT’S IT SO FAR! TO DO: Acabar get\_T\_R\_world, conseguir ter consenso entre duas câmaras e ter correspondência de pontos entre frames com o SIFT (probably)

Divisão do projeto em partes a implementar

Thresholds a ter cuidado (files a bold)

**bg**\_**fg**\_**detect.m**

fg\_depth\_bin = fg\_depth > 700

mag\_grad(mag\_grad < 2000) = 0;

filtered\_fg = imopen(segmented\_fg,strel('disk',2));

filtered\_fg2 = bwareaopen(filtered\_fg,2000);

**sift\_match**

thres = 7;