

VISÃO COMPUTACIONAL

Lista de Exercícios 01 –

Projeções e Homografias

I Projeções

Um objeto de calibração conhecido foi colocado em frente a uma câmera desconhecida. Neste exercício, usaremos um sistema de coordenadas com origem no centro ótico da câmera. Assim, cada ponto de coordenadas $[X \ Y \ Z]^T$ no espaço é projetado no plano da imagem na posição $[x \ y]^T$ ($[x \ y \ 1]^T$ em coordenadas homogêneas).

Considere os conjuntos de dados:

- MATLAB: *Pontos2D_1.mat*, *Pontos3D_1.mat* e *Pontos2D_2.mat*, *Pontos3D_2.mat*.
- Python (Numpy): *Pontos2D_1.npy*, *Pontos3D_1.npy* e *Pontos2D_2.npy*, *Pontos3D_2.npy*.

Cada conjunto contém (1) as coordenadas $[X \ Y \ Z]^T$ de N pontos marcados em um objeto de calibração e (2) as coordenadas $[x \ y]^T$ dos seus pontos correspondentes projetados num plano de imagem.

Podemos relacionar os pontos no espaço e no plano da imagem através da matriz de projeção \mathbf{P} tal que

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \\ p_{20} & p_{21} & p_{22} & p_{23} \end{bmatrix}}_{\mathbf{P}} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

para cada ponto i no conjunto de dados.

Para cada conjunto de dados:

1. Visualize os pontos no espaço 3D.
2. Escreva uma rotina para encontrar a matriz de projeção \mathbf{P} para cada conjunto de dados.
Sugestão: Expresse as equações para as coordenadas x_i e y_i no plano da imagem em função dos elementos da matriz \mathbf{P} e das coordenadas X_i , Y_i e Z_i e utilize decomposição SVD para encontrar a solução para a equação $\mathbf{A}\mathbf{p} = \mathbf{0}$, onde \mathbf{A} é uma matriz $N \times 12$ e \mathbf{p} é o vetor de 12×1 contendo os elementos da matriz \mathbf{P} .
3. Verifique se a matriz \mathbf{P} encontrada por você está correta (calcule o erro de projeção para os N pontos). Visualize os pontos no plano de projeção da imagem.
4. Recalcule \mathbf{P} , mas desta vez adicione ruído branco aos pontos 3D (use $\sigma = 0.05 * \max_{3D}$, onde \max_{3D} é o valor máximo absoluto das coordenadas dos pontos 3D);
5. Repita o item anterior, mas substituindo 20% dos pontos 3D por valores aleatórios na faixa $[-\max_{3D}/2, +\max_{3D}/2]$. Comente seus resultados.

II Homografias

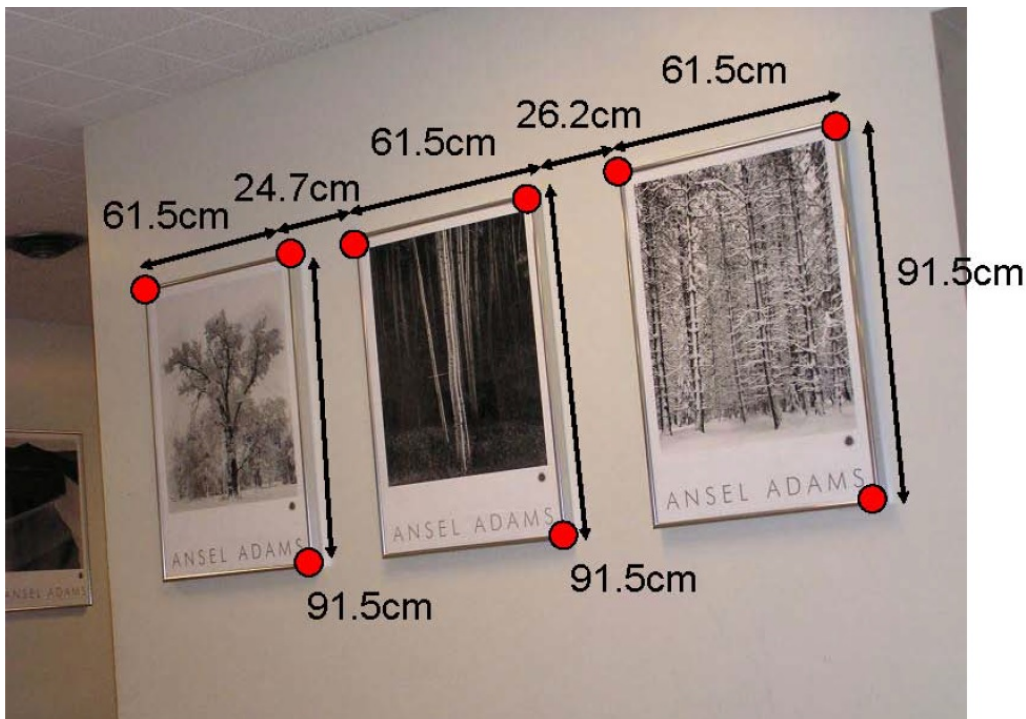
Os arquivos *homografia_1.txt* e *homografia_2.txt* contém coordenadas XY de N pares de pontos correspondentes nas imagens *img_homografia_1.png* e *img_homografia_2.png* (note que a primeira coluna contém pontos Y e a segunda pontos X).

1. Visualize cada imagem, superpondo a cada uma delas os pontos correspondentes. (**Sugestão:** no MATLAB, utilize a função `dlmread` para ler os arquivos *txt*).
2. Assuma que os pontos no espaço 3D são aproximadamente coplanares. Escreva uma função para determinar a homografia H_{21} entre a imagem 2 e a imagem 1 e a homografia H_{12} que transforma a imagem 1 na imagem 2. (**Sugestão:** utilize decomposição SVD).
3. Para cada homografia, compare (visualize) a imagem original e a resultante da transformação da segunda imagem. Comente seus resultados.

III Correção de Perspectiva

É possível utilizar os conhecimentos dos itens anteriores para corrigir distorções de perspectiva em imagens planares se estabelecermos a correspondência entre quatro pontos não-colineares na imagem e no mundo real. A partir destas correspondências e dos dados de relação de distâncias entre os pontos no mundo real, podemos calcular a homografia que mapeia os pontos da imagem distorcida no plano no mundo real.

A imagem *Quadros_ref.jpg* mostra a relação de distância no mundo real entre pontos não-colineares específicos nas imagens *Quadros_01.jpg*, *Quadros_02.jpg* e *Quadros_03.jpg*.



Para cada imagem *Quadros_01.jpg*, *Quadros_02.jpg* e *Quadros_03.jpg*:

1. Escolha manualmente um mínimo quatro pontos de referência nas imagens distorcidas. Mostre as imagens com os pontos escolhidos por você em destaque.
2. Utilizando os pontos escolhidos e suas correspondências, calcule a homografia H que transforma um ponto no plano da imagem no plano do mundo real.
3. Utilize H para remover a distorção de perspectiva, gerando a imagem resultante. Uma vez que as coordenadas no mundo real são reais, mas as coordenadas no plano da imagem (pixels) são inteiras, utilize técnicas de interpolação para estimar os valores de pixels no plano real.
4. Repita os itens 1 a 3 para uma imagem tirada por você mesmo com uma câmera digital. Será necessário usar uma trena ou régua para medir as distâncias no mundo real.