Single View Metrology

Projeto para a disciplina de Tópicos Avançados em Mídias e Interfaces

por Luca Ananias e André De' Carli

1. INTRODUÇÃO

O que estamos fazendo?

OBJETIVO

- Extrair medidas a partir de uma única imagem em perspectiva, dadas medidas de referência do usuário
  - Não é necessário retificar a imagem

O QUE É PRECISO

Plano de Referência

Escolhendo um plano de referência (usualmente o do chão) na imagem nós podemos computar distâncias em planos paralelos a ele.

Direção de Referência

Escolhendo uma direção de referência (não paralela ao plano de referência) nós podemos computar distâncias entre planos.

CIÊNCIA FORENSE

2.

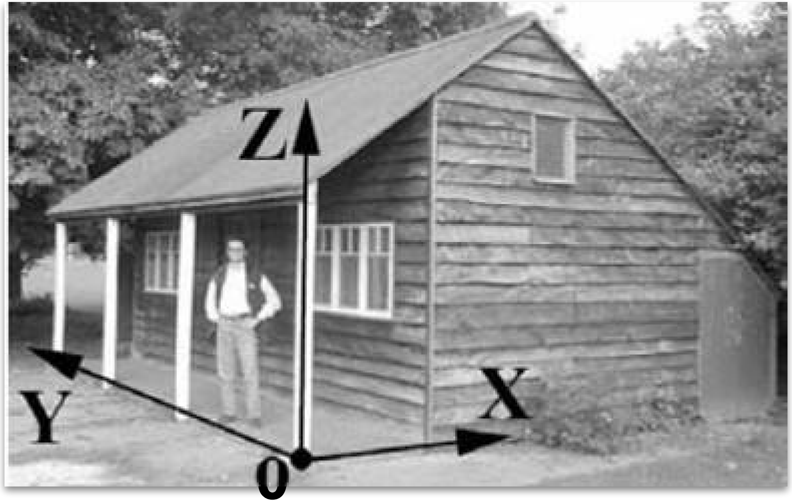
COMO FAZEMOS

Idéias gerais

SISTEMA DE COORDENADAS MUNDIAL

Definimos um sistema de coordenadas **XYZ** mundial da seguinte maneira:

1. Colocamos a origem no plano de referência;
2. Deixemos que os eixos **X** e **Y** gerem o plano de referência;
3. Deixemos que o eixo **Z** seja a direção de referência



ENTRADA DE DADOS

Precisamos que o usuário entre com as seguintes informações sobre a imagem:

- Um par de retas paralelas para cada eixo
- Uma medida real da imagem para cada eixo

ENTRADA DE DADOS

Utilizamos um objeto para servir como referência em nossos testes



MATRIZ DE PROJEÇÃO

Uma matriz de projeção , ou matriz de câmera têm a forma:

$$P = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad \mathbf{p}_3 \quad \mathbf{p}_4]$$

Onde **p1**, **p2** e **p3** são os pontos de fuga dos eixos **X**, **Y** e **Z**, mundiais, respectivamente. **p4** é a projeção da origem do sistema mundial.

Um ponto **X** mundial, tem imagem **x** de acordo com **x = PX**.

$$P = \begin{bmatrix}\mathbf{v}_X & \mathbf{v}_Y & \alpha\mathbf{v} & \hat{\mathbf{l}}\end{bmatrix}$$

MEDIÇÕES **ENTRE** PLANOS PARALELOS

Queremos obter a distância entre dois planos na cena especificados por pontos de referência **B = (X, Y, 0)^T** e **T = (X, Y, Z)^T** e suas imagens são **b** e **t**.

$$\mathbf{b} = P \begin{bmatrix} \mathrm{X} \\ \mathrm{Y} \\ 0 \\ 1 \end{bmatrix} = \rho(X\mathbf{p}_1 + Y\mathbf{p}_2 + \mathbf{p}_4)$$

$$\mathbf{t} = P \begin{bmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{bmatrix} = \mu(X\mathbf{p}_1 + Y\mathbf{p}_2 + Z\mathbf{p}_3 + \mathbf{p}_4)$$

MEDIÇÕES **ENTRE** PLANOS PARALELOS

Fazendo o produto escalar de **b** por **Î** temos:

$$\hat{\mathbf{l}}^\top \cdot \mathbf{b} = \rho(X\hat{\mathbf{l}}^\top \mathbf{p}_1 + Y\hat{\mathbf{l}}^\top \mathbf{p}_2 + \hat{\mathbf{l}}^\top \mathbf{p}_4)$$

Porém $\mathbf{v}_X$ e $\mathbf{v}_Y$ pertencem a **l**, e **Î** tem norma 1, logo:

$$\rho = \hat{\mathbf{l}}^\top \cdot \mathbf{b}$$

MEDIÇÕES **ENTRE** PLANOS PARALELOS

Ou seja:

$$\frac{\mathbf{b}}{\hat{\mathbf{l}}^\top \cdot \mathbf{b}} = (X\mathbf{p}_1 + Y\mathbf{p}_2 + \mathbf{p}_4)$$

Daí:

$$\mathbf{t} = \mu(X\mathbf{p}_1 + Y\mathbf{p}_2 + Z\mathbf{p}_3 + \mathbf{p}_4) = \mu\left(\frac{\mathbf{b}}{\hat{\mathbf{l}}^\top \cdot \mathbf{b}} + Z\alpha\mathbf{v}\right)$$

MEDIÇÕES **ENTRE** PLANOS PARALELOS

Reescrevendo:

$$\mu Z \alpha \mathbf{v} = \mathbf{t} - \frac{\mu \mathbf{b}}{\hat{\mathbf{l}}^\top \cdot \mathbf{b}}$$

Fazendo produto vetorial por t:

$$\mu Z \alpha \left(\mathbf{v} \times \mathbf{t}\right) = \left(\mathbf{t} \times \mathbf{t}\right) - \frac{\mu}{\hat{\mathbf{l}}^\top \cdot \mathbf{b}} \left(\mathbf{b} \times \mathbf{t}\right)$$

MEDIÇÕES **ENTRE** PLANOS PARALELOS

(1) $$\alpha Z = \frac{-\|\mathbf{b} \times \mathbf{t}\|}{(\hat{\mathbf{l}}^{\top} \cdot \mathbf{b})\|\mathbf{v} \times \mathbf{t}\|}$$

MEDIÇÕES **ENTRE** PLANOS PARALELOS

3.
ADAPTAÇÕES

MEDIÇÕES DE CUBOIDES

Queremos extrair medidas de objetos cuboides!

MEDIÇÕES **ENTRE** PLANOS PARALELOS

$$\text{(1)} \quad \alpha Z = \frac{-\|\mathbf{b} \times \mathbf{t}\|}{(\hat{\mathbf{l}}^\top \cdot \mathbf{b})\|\mathbf{v} \times \mathbf{t}\|}$$

Temos agora 3 planos e 3 direções de referência. Sendo um par por direção. Esta abordagem traz facilidades pois não exige nenhuma espécie de retificação na imagem.

4
DIFICULDADES ENCONTRADAS

DETECÇÃO DE PONTOS DE FUGA

Utilizamos o OpenCV para realizar uma detecção de arestas e inferir os pontos de fuga

- Grande número de outliers
  - Arestas que não são arestas de fato
  - Arestas que não pertencem aos eixos desejados
- Alta sensibilidade a parâmetros

DETECÇÃO DE PONTOS DE FUGA

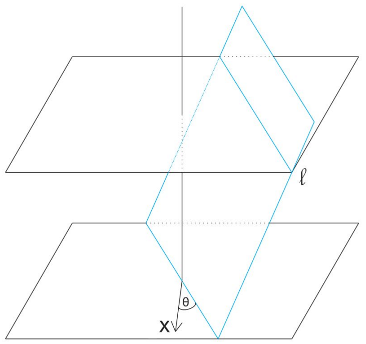
Fizemos uso do método RANSAC para encontrar os pontos de fuga

- Grande número de outliers resulta em falsos positivos
- Aumento na quantidade de amostras diminui drasticamente o desempenho do programa

DETECÇÃO DE PONTOS DE FUGA

Tentativa de agrupar os pontos de fuga

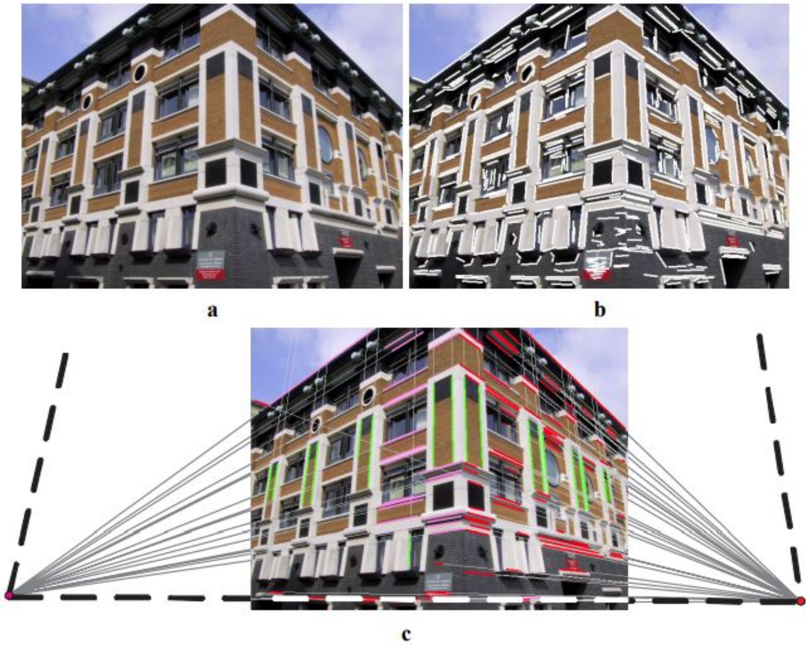
- Pontos visíveis, usamos distância euclidiana
- Pontos no infinito, usamos distância angular



5.
EM DESENVOLVIMENTO

Pequenas melhorias para serem implementadas no futuro

DETECÇÃO DE PONTOS DE FUGA



ERROS DE MEDIDAS

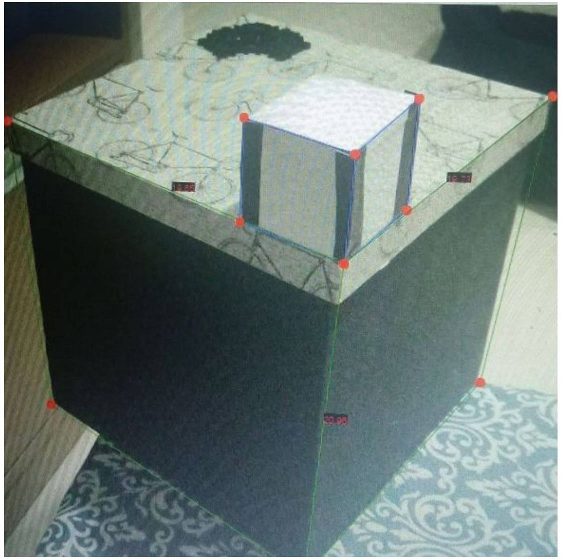
- Cálculo das incertezas associadas a cada medida
- Permitir que o usuário entre com mais de uma medida para aumentar a precisão

MEDIÇÕES ENTRE PLANOS

- Cálculo de proporções entre diferentes planos

DETECÇÃO AUTOMÁTICA DE OBJETOS

- Detecção de objetos de referência
- Medidas já conhecidas, diminui entrada do usuário



OBRIGADO!

Perguntas?

{ lams3, acms }@cin.ufpe.br