

Projeto Demonstrativo 02 - Calibração de Câmeras

Thúlio Noslen Silva Santos (14/0164090)

Departamento de Ciência da Computação

Universidade de Brasília

Brasília, Brasil

thulionoslen_@hotmail.com

Resumo—Para que possam ser extraídas características físicas sobre o mundo a partir de imagens capturadas por uma câmera, é necessário que se conheçam as características internas da câmera e qual é sua posição no mundo. Neste trabalho, métodos de calibração intrínseca e extrínseca serão utilizados a fim de conseguir estimar o tamanho de objetos no mundo a partir das imagens da câmera. Os fatores com maior influência na calibração são a distância do plano $Z=0$ do mundo até a câmera e a orientação do padrão de calibração com relação ao plano da imagem.

Index Terms—visão computacional, OpenCV, calibração de câmeras

I. INTRODUÇÃO

A partir do momento em que se conhecem as características de uma câmera, é possível determinar funções de mapeamento que levam de dimensões em pixels no plano da imagem para coordenadas em 3D no mundo e vice-versa. Para que se conheçam estas características, é necessário realizar os procedimentos chamados de calibração intrínseca e extrínseca.

A. Calibração intrínseca

A calibração intrínseca consiste em determinar a matriz K tal que:

$$P_{retina} = KP_{image} \quad (1)$$

Ou, escrita de outra forma:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

Veja que K pode ser vista como uma matriz de transformação que leva de coordenadas no plano da imagem, em metros, para o plano físico da retina, em pixels. Os planos podem ser conferidos na figura 1, retirada de [1].

Os parâmetros f_x e f_y estão relacionados à distância focal da câmera, enquanto c_x e c_y estão relacionados ao centro da imagem, visto que é comum que as medidas em pixels comecem de um dos cantos da imagem ao invés do centro. Por fim, o parâmetro γ é uma deformação nos eixos da retina física. Para este trabalho, assumiu-se que γ é zero, uma vez que as funções em OpenCV implementadas usam isto como padrão.

Além disto, outros parâmetros considerados intrínsecos são os de distorção radial e tangencial, dados por k_1, k_2, k_3 e p_1, p_2 , respectivamente. Sendo r o raio do círculo circunscrito

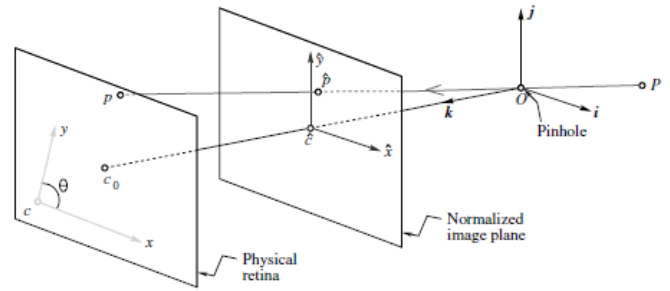


Figura 1. Coordenadas do mundo e planos físico e da imagem.

pela imagem, para a distorção radial, os parâmetros são tais que:

$$\begin{aligned} x_{corr} &= x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ y_{corr} &= y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{aligned} \quad (3)$$

E para a distorção tangencial:

$$\begin{aligned} x_{corr} &= x + [2p_1 xy + p_2(r^2 + 2x^2)] \\ y_{corr} &= y + [p_1(r^2 + 2y^2) + 2p_2 xy] \end{aligned} \quad (4)$$

Geralmente, os coeficientes são agrupados num vetor de coeficientes, tal como:

$$D = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3] \quad (5)$$

B. Calibração extrínseca

A calibração extrínseca consiste em determinar a matriz $M = R|t$ tal que:

$$P_{image} = MP_{world} \quad (6)$$

Ou, escrita de outra forma:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = [R \quad t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (7)$$

Veja que M pode ser vista como uma matriz de transformação que leva de coordenadas no mundo para coordenadas no plano, ambas em metros. Novamente, estas representações podem ser conferidas na figura 1. Os parâmetros R e t são matrizes de rotação e translação que fazem este mapeamento.

C. Soluções dos sistemas

Observe que a calibração pode ser considerada um problema de otimização. Imagens de um padrão conhecido são capturadas para posterior processamento. No caso da calibração intrínseca, são necessárias ao menos 2 imagens (4 coordenadas) para que se consiga estimar a matriz K . Neste caso, métodos de solução de equações lineares ou regressão linear podem ser utilizados. Como também há de ser realizada uma otimização não linear por conta da distorção, métodos de regressão não linear precisam ser usados, e mais imagens precisam ser capturadas. No OpenCV, a função de calibração intrínseca é implementada com métodos baseados em regressões lineares e métodos de declive de gradiente (*gradient descent*) [2]. Veja que mesmo que poucas imagens sejam necessárias, é recomendado que se usem ao menos 10, com o padrão em diferentes posições na imagem.

Para a calibração extrínseca, o problema é parecido, mas os métodos de solução são diferentes. A otimização precisa levar em consideração que a distância do padrão até a imagem também precisa ser estimada, que gera a equação:

$$\alpha \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = [R \quad t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (8)$$

Com isto, em OpenCV são implementados algoritmos como a otimização de Levenberg-Marquardt, que é baseada em minimizar o erro de reprojeção. Em outras palavras, queremos otimizar $\hat{\beta}$:

$$\hat{\beta} = \operatorname{argmin}_{\beta} \sum_{i=1}^m [y_i - f(x_i, \beta)]^2 \quad (9)$$

Em que $y_i - f(x_i, \beta)$ é o erro de reprojeção.

II. MATERIAIS E METODOLOGIA

A. Materiais

O programa desenvolvido deve atender aos seguintes requisitos:

- Permitir que seja medida uma distância em pixels numa imagem com o mouse;
- Realizar o procedimento de calibração intrínseca e permitir que sejam medidas distâncias em pixels na imagem original e corrigida de distorções.
- Realizar o procedimento de calibração extrínseca para 3 distâncias diferentes do padrão.
- Permitir que sejam medidas distâncias em pixels na imagem original e corrigida de distorções e transformar estas distâncias em medidas nas coordenadas do mundo.

Foi utilizado um computador com as seguintes configurações:

- Ubuntu 16.04.03 LTS 64-bits;
- OpenCV 3.4.0;
- numpy 1.14.2;
- Python 3.5

B. Metodologia

Foram utilizados tutoriais de Python, numpy, OpenCV, calibração de câmeras na elaboração do projeto. A câmera utilizada foi a câmera frontal de um Moto G4 Plus.

Foram desenvolvidos 4 módulos, um para cada requisito:

- line.py
- intrinsic.py
- extrinsic.py
- ruler.py

Mais informações sobre a implementação do trabalho pode ser conferida no código fonte e no arquivo README.md.

Para a calibração intrínseca, 50 imagens foram coletadas de diferentes vídeos com o auxílio de scripts em Python. A posição do padrão xadrez foi variada assim como sua inclinação, e as 50 imagens foram separadas em 5 pastas de 10 imagens cada.

Para a calibração extrínseca, 3 imagens foram coletadas para 3 distâncias diferentes, cada imagem com uma pequena variação na distância do padrão à câmera. O padrão foi mantido paralelo ao plano da imagem. Com isto, as 9 imagens foram separadas em 3 pastas para posterior processamento.

III. RESULTADOS

Todos os comandos usados para obter os resultados podem ser conferidos no arquivo README.md.

A. Requisito 1

Na figura 2 pode ser conferida uma medição de distância em pixels para uma das imagens de calibração. Uma vez que a imagem tem 960 pixels de largura, a distância medida de 227 pixels está coerente.

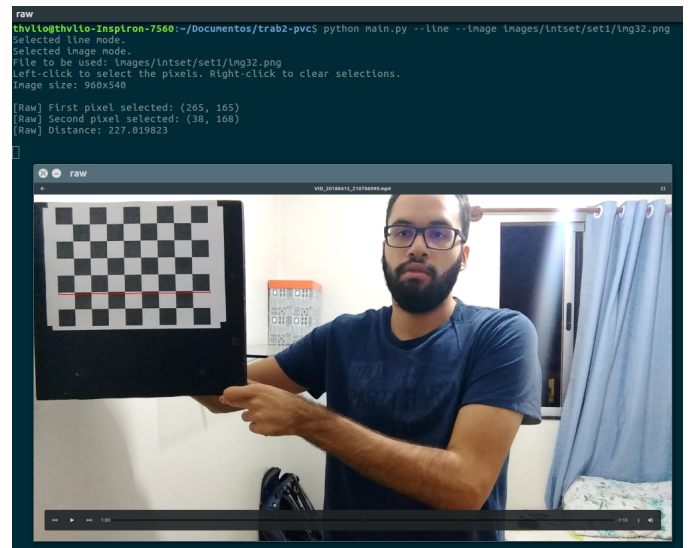


Figura 2. Resultado para o primeiro requisito.

B. Requisito 2

Na figura 3 pode ser conferida a medição da distância em pixels nas duas janelas, *raw* e *undistorted*. Veja que como a câmera não apresenta muita distorção, a distância se preserva bastante na imagem, obtendo-se 281 e 278 pixels para uma reta similar nas duas janelas. O efeito da correção da distorção é mais pronunciado nas bordas das imagens, como na cortina à direita na figura.

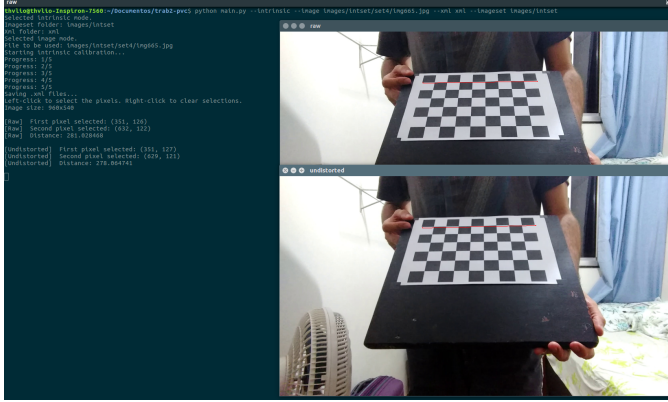


Figura 3. Resultado para o segundo requisito.

A matriz K e o desvio padrão obtidos foram:

$$K = \begin{bmatrix} 1288.04 & 0 & 952.51 \\ 0 & 1287.24 & 540.55 \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

$$\sigma_K = \begin{bmatrix} 11.42 & 0 & 3.70 \\ 0 & 13.87 & 13.02 \\ 0 & 0 & 1 \end{bmatrix} \quad (11)$$

O vetor de distorções e o desvio padrão obtidos foram: A matriz K e o desvio padrão obtidos foram:

$$D = [0.084 \quad -0.21 \quad -0.000028 \quad 0.00038 \quad 0.097] \quad (12)$$

$$\sigma_D = [0.032 \quad 0.17 \quad 0.0027 \quad 0.0022 \quad 0.24] \quad (13)$$

Os resultados para as médias se encontram nos arquivos *intrinsics.xml* e *distortions.xml*, respectivamente.

C. Requisito 3

O resultado da calibração para as 9 imagens foi armazenado no arquivo *extrinsics.xml*. Foi calculada uma média de 3 imagens para fins de estimar a confiabilidade dos resultados. Para cada distância, obteve-se a norma de t e seu desvio padrão, seguido da distância original e seu desvio padrão:

$$\begin{bmatrix} |t_{d_{min}}| \\ |t_{d_{med}}| \\ |t_{d_{max}}| \end{bmatrix} = \begin{bmatrix} 313.06 \\ 1126.09 \\ 2471.71 \end{bmatrix}, \quad \begin{bmatrix} \sigma_{|t_{d_{min}}|} \\ \sigma_{|t_{d_{med}}|} \\ \sigma_{|t_{d_{max}}|} \end{bmatrix} = \begin{bmatrix} 14.97 \\ 11.11 \\ 37.24 \end{bmatrix} \quad (14)$$

$$\begin{bmatrix} d_{min} \\ d_{med} \\ d_{max} \end{bmatrix} = \begin{bmatrix} 290.00 \\ 1123.33 \\ 2343.33 \end{bmatrix}, \quad \begin{bmatrix} \sigma_{d_{min}} \\ \sigma_{d_{med}} \\ \sigma_{d_{max}} \end{bmatrix} = \begin{bmatrix} 16.33 \\ 17.00 \\ 36.82 \end{bmatrix} \quad (15)$$

IV. REQUISITO 4

Um exemplo de medição para um objeto pode ser conferido na figura 4. Foi escolhida uma imagem correspondente à distância d_{max} , e foi medida a caixa perto da borda e a largura de tábua que serviu de suporte ao padrão. Os resultados de distância estão apresentados na tabela I.

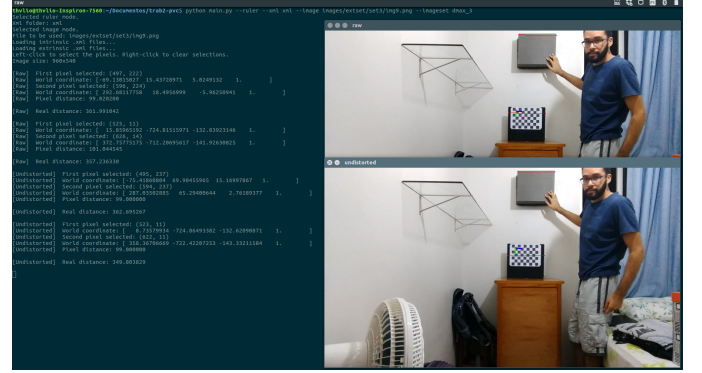


Figura 4. Resultado para o quarto requisito.

Tabela I
RESULTADOS PARA MEDIDAS DE OBJETOS

Posição do padrão de calibração	d_{min}	d_{med}	d_{max}
ltl, medida pela trena	290.00	1123.33	2343.33
ltl, calculada pela cal. extrínseca	313.06	1126.09	2471.71
$l_{raw,centre}$	198.67	356.18	361.99
$l_{raw,perifery}$	188.64	385.31	357.24
$l_{undistorted,centre}$	196.60	354.62	362.70
$l_{undistorted,perifery}$	190.68	388.52	349.80
l_{real}	170	355	355

Os objetos usados foram uma capa de um leitor digital para perto, e a tábua de suporte para o padrão e uma caixa cuja largura é igual à largura da tábua para médio e longe.

V. DISCUSSÃO E CONCLUSÕES

Observe pelos resultados do requisito 2 que o desvio padrão é pequeno com relação à média, indicando bons resultados. Os parâmetros de centro são próximos dos valores esperados visto que a imagem foi capturada em 1080p, e os de distância focal também.

Para a matriz D, observe que os parâmetros de distorção tangencial e o último de distorção radial tem desvio padrão muito alto, indicando que não foram bem modelados pela calibração.

De acordo com a tabela I, pode-se notar que o valor ltl é próximo mas sempre maior que os valores determinados pela calibração. Isto ocorre porque existem componentes x e y em t que surgem por conta de desalinhamentos do padrão com relação ao plano paralelo à imagem. Observe que, apesar disto, os desvios de ltl medidos são próximos de ltl estimados, indicando boas estimativas.

Pelo tamanho medido dos objetos, pode-se notar que as melhores medidas centrais são para a distância intermediária.

Para as distâncias periféricas, as melhores medidas são as da distância máxima.

Para obter melhores resultados, seria interessante remover os parâmetros de distorção, uma vez que a câmera aparenta já vir com boas especificações de fábrica. Outra consideração é que as melhores medições se deram na distância intermediária.

REFERÊNCIAS

- [1] David A. Forsyth, Jean Ponce, “Computer Vision: A Modern Approach,” Second Edition, Prentice Hall, 2011.
- [2] OpenCV, “Camera Calibration and 3D Reconstruction — OpenCV 3.0.0-dev documentation,” https://docs.opencv.org/3.0-beta/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html. Acessado em 17/04/2018.
- [3] OpenCV, “Camera Calibration — OpenCV 3.0.0-dev documentation,” https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_calib3d/py_calibration/py_calibration.html. Acessado em 17/04/2018.
- [4] OpenCV, “Pose Estimation — OpenCV 3.0.0-dev documentation,” https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_calib3d/py_pose/py_pose.html. Acessado em 17/04/2018.
- [5] OpenCV, “OpenCV: cv::FileStorage Class Reference,” https://docs.opencv.org/3.4.0/da/d56/classcv_1_1FileStorage.html. Acessado em 17/04/2018.
- [6] Satya Mallick, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” <https://www.learnopencv.com/head-pose-estimation-using-opencv-and-dlib/>. Acessado em 17/04/2018.