

Calibração de Câmeras

ALGEMIRO A. S. NETO

Pontifícia Universidade Católica do Rio de Janeiro

aneto@inf.puc-rio.br

Abstract

Este trabalho aborda o problema de calibração de câmeras apresentando uma breve descrição teórica e uma implementação para o mesmo com base na biblioteca desenvolvida por Flávio Szenberg, "Juiz Virtual". O método para calibração de câmeras implementado foi o Tsai2D e para ilustrar o resultado da calibração foi inserido um objeto virtual sobre o modelo utilizado como referência no mundo real. Para validar os resultados foram recuperados parâmetros de câmeras conhecidas. São apresentados alguns resultados práticos utilizando imagens sintéticas e reais capturadas com uma câmera Sonny P200 Cyber-shot 7.2 Mega pixels e uma Cannon Rebel.

I. INTRODUCAO

O interesse por métodos de processamento digital de imagens vem principalmente de duas importantes áreas de aplicação: aprimoramento da informação percebida por um ser humano (melhoramento) e interpretação de cenas por máquinas autônomas (percepção). Por outro lado, pesquisas em computação gráfica têm como grande impulsionador principalmente aplicações na indústria do entretenimento. Recentemente, o desenvolvimento de hardware permitiu a implementação de idéias que permitem a fusão de técnicas das duas áreas: computação gráfica e processamento de imagens, em uma nova área que convencionou-se chamar de Realidade Aumentada.

Realidade Aumentada (RA) é a aplicação de técnicas de CG em ambientes reais, capturados com câmeras digitais com o objetivo de aumentar as informações das imagens reais. RA pode ser considerada também como a tecnologia entre a Realidade Virtual (RV) e a telepresença. Enquanto na RV o ambiente é completamente sintético e na telepresença ele é completamente real, na RA o usuário vê o mundo real “aumentado” com objetos virtuais.

Sistemas RA podem tirar vantagens de displays HMD (*Head-Mounted-Displays*), sensores de posição 3D (*3D pointers*), câmeras digitais e os tradicionais displays bidimensionais para mostrar o resultado mixado. A idéia de superpor imagens sintéticas geradas por computador em imagens reais visualizadas pelo usuário tem sido explorada enormemente nos dias de hoje.

Entre diversas aplicações, a detecção de linhas com o objetivo de calibração de câmeras para projeção de objetos virtuais sobre cenas reais está entre os maiores e mais importantes desafios. Neste trabalho, apresentamos uma técnica desenvolvida por [2] para recuperar, em tempo real e sem utilizar qualquer informação adicional, a posição e os parâmetros da câmera em uma sequência de imagens contendo a visualização de modelos conhecidos. Para tal, é explorada a existência, nessas imagens, de segmentos de retas que compõem a visualização do modelo cujas posições são conhecidas no mundo tridimensional. Quando se trata de uma partida de futebol, por exemplo, o modelo em questão é composto pelo conjunto das linhas do campo, segundo as regras que definem sua geometria e dimensões.

O restante deste trabalho está dividido da seguinte maneira: Na Seção 2 é apresentado o modelo de câmera considerado neste trabalho, na Seção 3 são abordadas as transformações de câmera envolvendo mudança referencial e projeção perspectiva. Na Seção 4 é definido o problema de calibração de câmera e o Método utilizado neste trabalho para resolvê-lo. A Seção 5 descreve as etapas no desenvolvimento de um sistema para calibração de câmeras e na Seção 6 são apresentados os resultados obtidos.

projetada passa a se formar entre o objeto tridimensional e o centro de projeção O , também chamado de Centro da Câmera. O plano onde a imagem é formada, chamado aqui de Plano da imagem projetada, fica a uma distância da câmera chamada de distância focal, denotada por f . A linha que parte de O e é perpendicular ao plano da imagem é o eixo óptico. O centro da imagem C , é a intersecção entre o eixo óptico e o plano da imagem. O ponto p , imagem de P , é o ponto no qual a linha reta entre O e P intercepta o plano da imagem. A Figura 2 define um sistema de coordenadas tridimensional $OXYZ$, com a origem no centro óptico da câmera, o eixo Z perpendicular ao plano de projeção onde a imagem é formada e os eixos X e Y paralelos aos lados da imagem. A projeção desse sistema no plano da imagem induz um sistema de coordenadas $C\tilde{u}\tilde{v}$ nesse plano.

Com relação a esses sistemas de coordenadas, a projeção perspectiva de um ponto $P = (X, Y, Z)$ do espaço no sistema de coordenadas $OXYZ$ é dada por:

$$(\tilde{u}, \tilde{v}) = (f \frac{X}{Z}, f \frac{Y}{Z}) \quad (1)$$

onde

$$(\tilde{u}, \tilde{v}) = (u - u_0, v - v_0) \quad (2)$$

e

$$C = (u_0, v_0) \quad (3)$$

Essa equação é facilmente formulada por semelhança de triângulos sobre o modelo apresentado na Figura 2 e será usada mais adiante em projeções nesse modelo.

III. TRANSFORMAÇÃO DE CÂMERA

Considerando o Modelo de Câmera apresentado na Seção anterior, podem ser especificados quatro sistemas de coordenadas:

- Sistema de Coordenadas do Mundo (SCM)
- Sistema de Coordenadas da Câmera (SCC)
- Sistema de Coordenadas da Imagem (SCI)
- Sistema de Coordenadas em Pixels (SCP)

Geralmente um ponto P no espaço não é descrito no sistema de coordenadas $OXYZ$ (SCC), e sim no sistema xyz (SCM). Sendo assim, para utilizar a equação 1 para um ponto P no sistema xyz , deve-se efetuar uma transformação que mapeie este ponto para o SCC. Tal transformação é dada pela composição de uma rotação que alinhe os eixos e uma translação que iguale as origens. A Matriz 3×4 abaixo descreve esta transformação:

$$[R|T] = \begin{bmatrix} r_{1x} & r_{1y} & r_{1z} & t_x \\ r_{2x} & r_{2y} & r_{2z} & t_y \\ r_{3x} & r_{3y} & r_{3z} & t_z \end{bmatrix} \quad (4)$$

onde r_1, r_2 e r_3 definem a direção e inclinação da câmera formando uma base ortonormal e o ponto $T = (t_x, t_y, t_z)$ representa a origem do sistema xyz escrita em coordenadas do sistema da câmera. Combinando as equações 1 e 2 temos a projeção, no plano da imagem (SCI), de um ponto no sistema de coordenadas do mundo.

$$\begin{bmatrix} \tilde{u}s \\ \tilde{v}s \\ s \end{bmatrix} \begin{bmatrix} fr_{1x} & fr_{1y} & fr_{1z} & ft_x \\ fr_{2x} & fr_{2y} & fr_{2z} & ft_y \\ r_{3x} & r_{3y} & r_{3z} & t_x \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (5)$$

IV. CALIBRAÇÃO DE CÂMERA

Consiste em escrever as equações de projeção (rotação e translação) de um referencial conhecido no mundo real, em relação a um referencial da câmera (parâmetros extrínsecos), e com isto determinar os parâmetros da câmera (parâmetros intrínsecos). Isto é importante para que se possa desenvolver, posteriormente, aplicações com esta câmera que dependem do conhecimento preciso de seus parâmetros intrínsecos. Nem todas as aplicações necessitam de calibração da câmera, casos onde a precisão não é fator primordial podem exemplificar esta questão, como, por exemplo, a navegação robótica, onde se deseja somente identificar a ocorrência ou não de um obstáculo, ou onde a distância relativa entre o robô e um obstáculo não necessita precisão milimétrica.

A. Método de Tsai

Neste método o conjunto de pontos é subdividido para que novos pontos sejam criados. O método de Tsai nos fornece uma matriz de projeção e outra de *modelview* usadas no OpenGL para visualizarmos um modelo virtual em 3D.

Dividindo a primeira e a segunda linha da equação (5) pela terceira linha, obtemos as seguintes expressões:

$$\tilde{u} = f \frac{r_{1x}x + r_{1y}y + r_{1z}z + t_x}{r_{3x}x + r_{3y}y + r_{3z}z + t_z} \quad (6)$$

$$\tilde{v} = f \frac{r_{2x}x + r_{2y}y + r_{2z}z + t_y}{r_{3x}x + r_{3y}y + r_{3z}z + t_z} \quad (7)$$

Dividindo a equação (6) pela (7) obtemos a seguinte expressão:

$$\frac{\tilde{u}}{\tilde{v}} = \frac{r_{1x}x + r_{1y}y + r_{1z}z + t_x}{r_{2x}x + r_{2y}y + r_{2z}z + t_x} \quad (8)$$

Esta equação apresenta uma não-linearidade com relação às variáveis r_{ix} e r_{iy} . Para contornar esse não

linearidade, reescrevemos essa equação da seguinte forma:

$$\frac{r_{1x}}{t_y}x_i\tilde{v}_i + \frac{r_{1y}}{t_y}y_i\tilde{v}_i - \frac{r_{2x}}{t_y}x_i\tilde{u}_i - \frac{r_{2y}}{t_y}y_i\tilde{u}_i + \frac{t_x}{t_y}\tilde{v}_i = \tilde{u}_i \quad (9)$$

Assim, obtemos um sistema do tipo $Au = b$, onde A é uma matriz $n \times 5$ e cada linha A_i é dada por $(x_i\tilde{v}_i, y_i\tilde{v}_i, -x_i\tilde{u}_i, -y_i\tilde{u}_i, \tilde{v}_i)$, u é um vetor dado por:

$$(U_1, U_2, U_3, U_4, U_5) = \left(\frac{r_{1x}}{t_y}, \frac{r_{1y}}{t_y}, \frac{r_{2x}}{t_y}, \frac{r_{2y}}{t_y}, \frac{t_x}{t_y} \right) \quad (10)$$

Uma vez que r_1, r_2 e r_3 são ortonormais e definindo $\alpha = \frac{r_{1x}}{t_y}$ e $\beta = \frac{r_{2x}}{t_y}$, temos que

$$\left\{ \begin{array}{l} \alpha\beta = -U_1U_3 - U_3U_4 \\ \alpha^2 + U_1^2 + U_2^2 = \beta^2 + U_3^2 + U_4^2 \end{array} \right\} \quad (11)$$

onde cada elemento do vetor b é dado por $\tilde{u}_i$.

Resolvendo o sistema acima, calcula-se t_y a partir de

$$t_y^2 = \frac{U - \sqrt{U^2 - 4(U_1U_4 - U_2U_3)^2}}{2(U_1U_4 - U_2U_3)^2} \quad (12)$$

onde $U = U_1^2 + U_2^2 + U_3^2 + U_4^2$.

Assim, com t_y já definido, podemos rescrever a equação (9) em forma de um sistema linear:

$$\begin{bmatrix} x_i\tilde{v}_i & y_i\tilde{v}_i & x_i\tilde{u}_i & y_i\tilde{u}_i & \tilde{v}_i \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n\tilde{v}_n & y_n\tilde{v}_n & x_n\tilde{u}_n & y_n\tilde{u}_n & \tilde{v}_n \end{bmatrix} \begin{bmatrix} \frac{r_{1x}}{t_y} \\ \frac{r_{1y}}{t_y} \\ -\frac{r_{2x}}{t_y} \\ -\frac{r_{2y}}{t_y} \\ \frac{t_x}{t_y} \end{bmatrix} = \begin{bmatrix} \tilde{u}_i \\ \vdots \\ \tilde{u}_n \end{bmatrix} \quad (13)$$

cujas resoluções encontram as variáveis $r_{1x}, r_{1y}, r_{2x}, r_{2y}$ e t_x . Pelo fato dos vetores r_1 e r_2 terem normas iguais a 1, encontram-se também os valores r_{1z} e r_{2z} , e conseqüentemente calcula-se r_3 , pois r_1, r_2 e r_3

são ortonormais. Finalmente, usando os valores já encontrados e as equações (6) e (7), calculamos f e t_z .

V. IMPLEMENTAÇÃO

Esta Seção descreve os passos envolvidos na implementação do sistema de calibração de câmera. Um objeto Virtual simples, foi inserido em uma cena real e parâmetros de câmeras conhecidas foram recuperados para validação do método.

A. Captura do Marcador

A primeira etapa consiste em capturar o marcador. Neste programa isso pode ser feito em real com uma câmera ou carregando uma imagem do arquivo. É através do marcador que são fornecidos os pontos de referência no mundo para que o aplicativo possa inserir o objeto virtual na cena. As Figuras 3 e 4 ilustram respectivamente um marcador capturado com uma Web Cam e um marcador sintético.

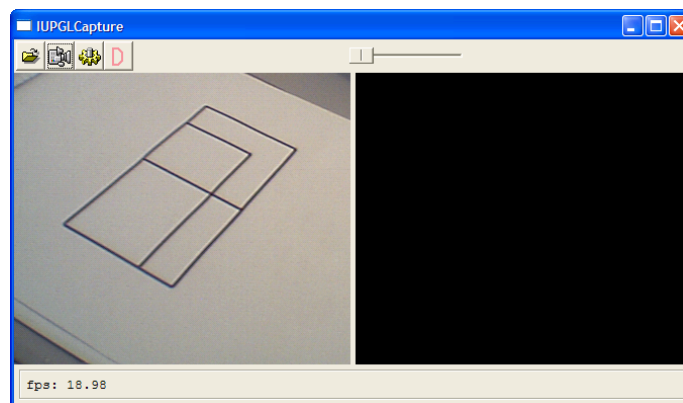


Fig. 3. Marcador Real

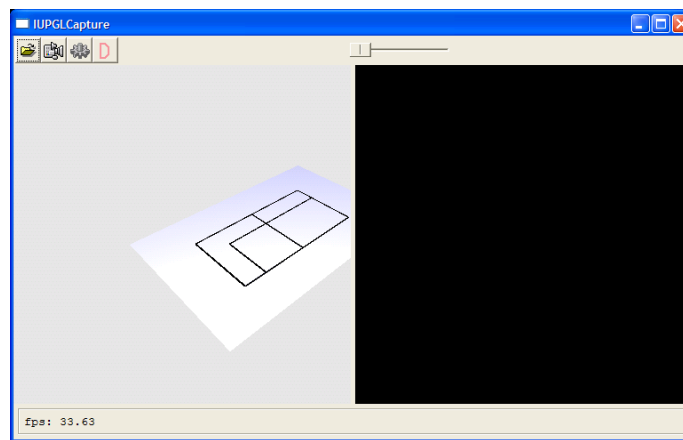


Fig. 4. Marcador Sintético

B. Detecção de Linhas

O primeiro passo do algoritmo implementado é a aplicação de dois filtros: um filtro gaussiano seguido de um filtro laplaciano. Esta técnica é chamada de Laplaciano da Gaussiana, ou LoG. Essa filtragem é aplicada sobre o negativo da luminância da imagem a fim de realçar linhas e dar suporte a uma segmentação por limiar.

C. Extração de Segmentos de Retas Longos

O resultado do primeiro passo do algoritmo é outra imagem, que apresenta pontos com intensidades maiores como sendo candidatos a estarem sobre algum segmento de reta presente na imagem original (mostrar um exemplo). Esta imagem resultante é passada para o segundo passo, onde são extraídos segmentos de retas longos, candidatos a serem parte da imagem do modelo em questão. O resultado desse segundo passo é um conjunto de segmentos de retas, definidos pelos seus pontos extremos (aqui pode ter outro exemplo).

D. Reconhecimento de segmentos

Os segmentos de reta extraídos no passo anterior são finalmente passados ao terceiro passo onde cada segmento de reta extraído é identificado como representante de um dado segmento de reta do modelo. Para isso foi utilizada a função `juvInterpret` da biblioteca `Juiz Virtual` que se baseia em uma árvore de interpretação para estabelecer uma correspondência entre as linhas do modelo. Com base nos segmentos são definidos os vértices do modelo e a partir desses vértices pode ser feita a reconstrução do modelo conforme ilustra a Figura 5.

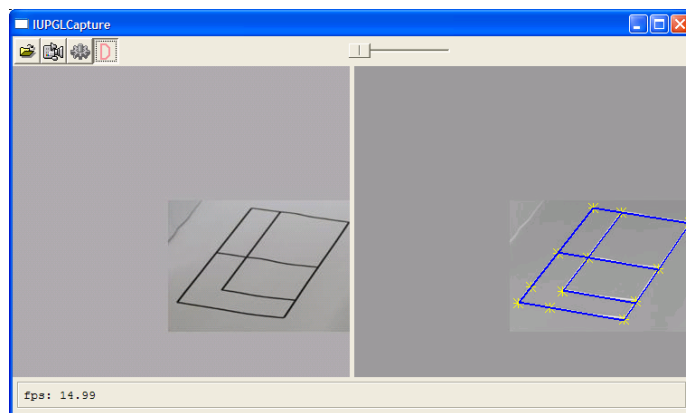


Fig. 5. A esquerda, Modelo reconstruído com base nos vértices definidos a partir das intersecções entre os segmentos reconhecidos

VI. RESULTADOS

Esta seção mostra alguns resultados obtidos com a implementação do método. Inicialmente, a Figura 6 mostra um objeto virtual inserido em uma cena capturada com uma web cam. No lado direito do canvas a projeção do modelo capturado.

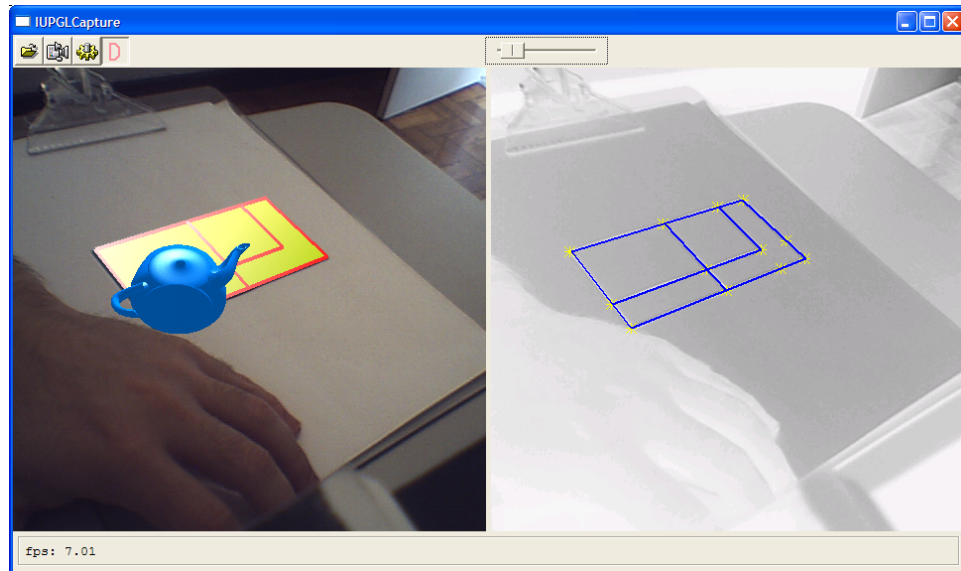


Fig. 6. Inserção de objeto virtual sobre um marcador na cena

REFERENCES

- [1] M. Gattass. Visão computacional e realidade aumentada, <http://www.tecgraf.puc-rio.br/~mgattass/ra/ra.html>, puc - rio de janeiro. 2006.2.
- [2] F. Szemberg. Acompanhamento de cenas com calibração automática de câmeras. In *Tese de Doutorado*, Pontifícia Universidade Católica do Rio de Janeiro, 2001.