

desvio padrão em torno do valor médio de posição. Semelhantemente, a informação de cor é modelada por uma cor média e a sua variância. Portanto, uma blob é uma distribuição gaussiana com seis dimensões. As gaussianas D -dimensionais são representadas pela Equação 1.

Analisando o conceito de gaussiana, pode-se determinar o quão distante cada voxel está em relação a cada blob. Considerando um voxel V na posição X_V com as cores $C_V = \{C_V^1, \dots, C_V^{mc}, \dots, C_V^{N_c}\}$ vistas pelas N_c câmeras, a distância de Mahalanobis $D_M(V, B) = (V - \mu)\Sigma^{-1}(V - \mu)^T$ mede a distância deste voxel à blob $B = (\mu, \Sigma)$.

Supondo que não existe dependência entre as informações de cor e posição, pode-se simplificar $D_M(V, B)$:

$$D_M(V, B) = (X_V - \mu_X)\Sigma_X^{-1}(X_V - \mu_X)^T + (C_V^{mc} - \mu_C)\Sigma_C^{-1}(C_V^{mc} - \mu_C)^T \quad (6)$$

Para o cálculo de $D_M(V, B)$, utiliza-se apenas o vetor de cor C_V^{mc} que minimiza a distância ao modelo da blob, ou seja:

$$mc = \arg_{i=1\dots N_c} \min(C_V^i - \mu_C)\Sigma_C^{-1}(C_V^i - \mu_C)^T \quad (7)$$

Inicialização do modelo. Um procedimento de inicialização é necessário, a fim de fornecer uma estimativa inicial dos parâmetros das blobs para o algoritmo de rastreamento. Neste projeto, foi utilizada uma inicialização manual, na qual um usuário informa, para cada blob: a) dois pontos extremos do principal eixo de variância da blob em questão. Através destes pontos pode-se estimar o valor médio de posição da blob e a direção do principal eixo de variação da gaussiana. b) três valores $\alpha_x, \alpha_y, \alpha_z$ que informarão os desvios padrões em cada eixo de direção.

Denominando-se R a matriz de rotação do elipsóide (blob) em relação aos eixos X, Y, Z , a matriz de covariância de posição da blob pode ser calculada por:

$$\Sigma_X = R \cdot \begin{pmatrix} \alpha_x^2 & 0 & 0 \\ 0 & \alpha_y^2 & 0 \\ 0 & 0 & \alpha_z^2 \end{pmatrix} \cdot R^T \quad (8)$$

Para a obtenção da matriz R , pode-se utilizar o principal eixo de variação da blob, para calcular os ângulos desse eixo com relação aos eixos X, Y, Z da cena (ângulos de Euler), seguindo a Figura 4:

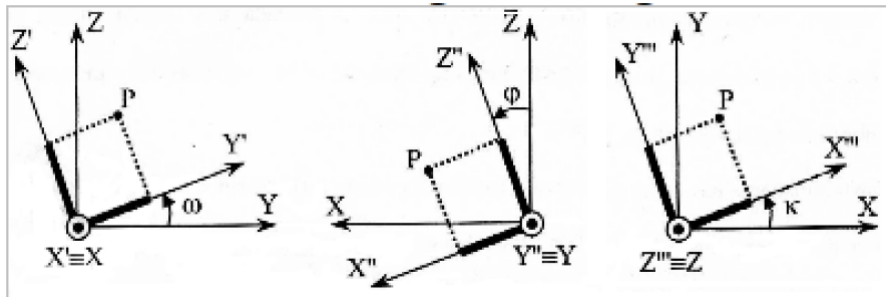


FIGURA 4 - Ângulos de Euler - para a determinação da matriz de rotação R .

A partir destes ângulos e considerando que a rotação inicial é dada por $R_X(\omega)$, uma secundária por $R_Y(\varphi)$ e uma final dada por $R_Z(\kappa)$, a matriz de rotação resultante pode ser calculada da seguinte maneira [19] (s representa o seno e c , o cosseno):

$$\omega = \arctg\left(\frac{z_b - z_a}{y_b - y_a}\right) \quad \varphi = \arctg\left(\frac{x_b - x_a}{y_b - y_a}\right) \quad \kappa = \arctg\left(\frac{y_b - y_a}{x_b - x_a}\right) \quad (9)$$

$$R = \begin{pmatrix} c\varphi c\kappa & s\omega s\varphi c\kappa + c\omega s\kappa & -c\omega s\varphi c\kappa + s\omega s\kappa \\ -c\varphi s\kappa & -s\omega s\varphi s\kappa + c\omega c\kappa & c\omega s\varphi s\kappa + s\omega c\kappa \\ s\varphi & -s\omega c\varphi & c\omega c\varphi \end{pmatrix}$$

5- RASTREAMENTO

O Rastreamento é executado pelo método Expectation-Maximization (EM) baseado em [5]. O Expectation-Maximization é composto pelos dois passos seguintes (executados repetidamente):

Expectation: para cada elemento de volume da cena discretizada (voxel), é determinado a qual parte do objeto este voxel pertence, ou seja, a qual blob. Para tal tarefa, se utiliza os parâmetros das blobs estimadas nas iterações anteriores do EM ou inicializados pelo usuário (caso seja a primeira iteração do EM).

Assim, para cada voxel, se computa a probabilidade de este pertencer a cada blob e, então, se designa este voxel a blob com maior probabilidade. Definindo que a blob B_j é representada pelo vetor médio μ_j e pela matriz de covariância Σ_j , a probabilidade $p(V_i|B_j)$ que V_i tem de pertencer a blob B_j é calculada por uma gaussiana de 6 dimensões:

$$p(V_i|B_j) = \frac{1}{(2\pi)^3 \sqrt{|\Sigma_j|}} \cdot e^{-\frac{1}{2}D_M(V_i, B_j)} \quad (10)$$

Onde $D_M(V_i, B_j)$ é a distância de Mahalanobis (calculada acima) e $|\Sigma_j|$ é o determinante da matriz de covariância de B_j . Supondo que não existe dependência entre as informações de cor e posição, $|\Sigma_j|$ pode ser reduzida por um produto dos determinantes das matrizes de covariância de posição e cor.

Uma otimização padrão é comparar os logaritmos das probabilidades, pois o máximo da função de probabilidade é também o máximo logaritmo desta função:

$$\log p(V_i|B_j) = -3 \log(2\pi) - \frac{1}{2} \log|\Sigma_j| - \frac{1}{2} D_M(V_i, B_j) \quad (11)$$

Desconsiderando os termos e fatores multiplicativos constantes, obtém-se a função que denominamos de $\phi()$, cuja minimização é equivalente à maximização da função de probabilidade original:

$$\phi(V_i, B_j) = \log|\Sigma_j| + D_M(V_i, B_j) \quad (12)$$

Minimizar $\phi()$ depende principalmente da distância de Mahalanobis $D_M()$. O termo $\log|\Sigma_j|$ é constante para cada blob e atua apenas como um fator de normalização. Este termo utiliza a variância codificada na matriz Σ_j para favorecer as blobs menores, fornecendo, a estas, maiores chances de serem escolhidas pelos voxels. Este termo é útil quando duas blobs são muito semelhantes (em posição e cor), pois, se apenas $D_M()$ fosse utilizada, a blob com maior variância sempre teria maior vantagem em ser escolhida.

Maximization: novos valores para o vetor média μ e para a matriz de covariância Σ de cada blob B são estimados, utilizando, para isto, o conjunto de voxels designados à cada blob (durante o passo Expectation).

Após, os parâmetros atualizados são, então, utilizados como estimativa inicial para o passo Expectation da próxima iteração do algoritmo.

5.1 Fluxo óptico

O Fluxo óptico é a distribuição 2D da velocidade aparente de movimento de padrões no plano da imagem [20]. Ou seja, o campo de fluxo óptico consiste em um campo denso de velocidade, onde cada pixel, no plano da imagem, está associado a um único vetor de velocidade [12].

Métodos baseados em fluxo óptico têm sido freqüentemente utilizados, na literatura, para a análise de movimento, tanto no domínio 2D [12], quanto no domínio 3D [13]. No entanto, até o momento, nenhum trabalho foi encontrado utilizando a técnica de Grid de Ocupação Probabilístico 3D em conjunto com o Fluxo óptico.

A partir da afirmação de que a agregação de mais informações ao grid probabilístico 3D (além da cor) é possível e pode favorecer a obtenção de maiores informações na reconstrução 3D [10]; considera-se que o fluxo óptico possa ser uma importante fonte de informação, capaz de auxiliar na etapa posterior de rastreamento.

A informação de velocidade pode ser adquirida individualmente para cada câmera e unificada em um único vetor para cada voxel; constituindo-se, assim, em uma informação que não realiza avaliações prematuras analisando apenas uma câmera, conforme o objetivo central da técnica de Grid Probabilístico. Após, então, pode-se calcular um vetor de velocidade para cada blob, utilizando os vetores de fluxo de todos os voxels da blob em questão. Assim, é possível estimar a localização das blobs, no decorrer do tempo, com maior precisão.

Levando tais fatos em consideração, estudos para a aplicação do fluxo óptico em conjunto com a técnica de Grid Probabilístico vêm sendo desenvolvidos.

Primeiramente, o fluxo óptico é calculado para cada imagem 2D em separado, utilizando o algoritmo Lucas Kanade [21]. Após, para cada pixel, é

feito um mapeamento das suas coordenadas 2D para o espaço 3D, de forma, a se obter um conjunto de N_c (número de câmeras) vetores de velocidade para cada voxel e, então, se calcula a média destes vetores, obtendo, como resultado, um único vetor de velocidade para cada voxel.

6- RESULTADOS

Um sistema foi desenvolvido com a finalidade de realizar: i. a reconstrução probabilística 3D, através da técnica de Grid de Ocupação Probabilístico; e ii. o rastreamento, através do método Expectation-Maximization.

Em relação à reconstrução volumétrica, a técnica de Grid de Ocupação Probabilístico apresentou vantagens consideráveis em relação ao método tradicional Shape-From-Silhouette. Este último apresentou falhas na seqüência de imagens de um vídeo, que não foram encontradas na reconstrução obtida pelo Grid de Ocupação Probabilístico (ver Figura 5).

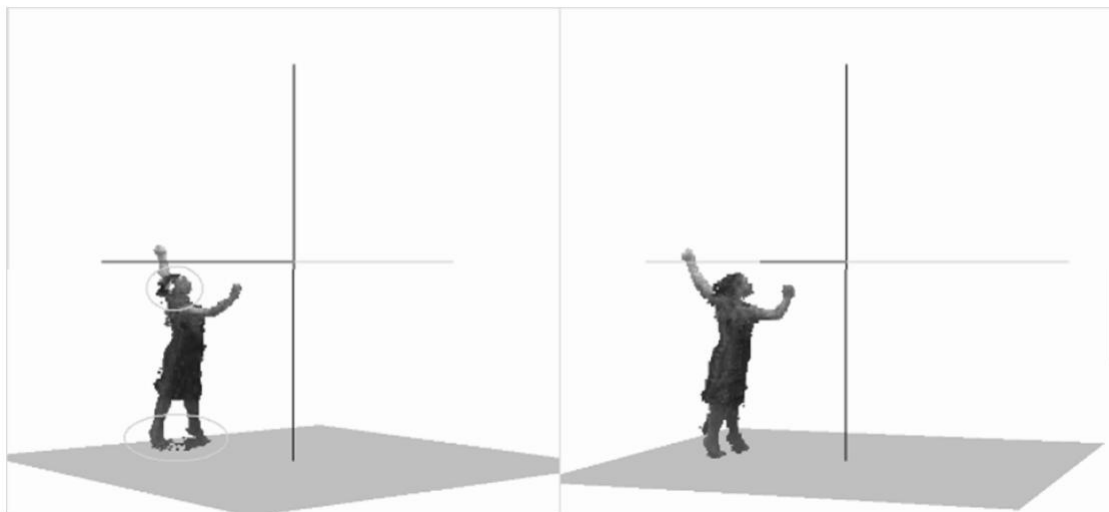


FIGURA 5 - Reconstrução volumétrica. Esquerda: método tradicional Shape-from-silhouette (falhas em verde). Direita: técnica Grid de Ocupação Probabilístico.

Em relação ao rastreamento, utilizou-se o método Expectation-Maximization, para seguir o objeto frame a frame. A Figura 6 apresenta uma seqüência de movimentos rastreados com sucesso. Os voxels estão representados com tonalidades arbitrárias, cada conjunto de voxels de uma mesma tonalidade pertence a uma única blob.

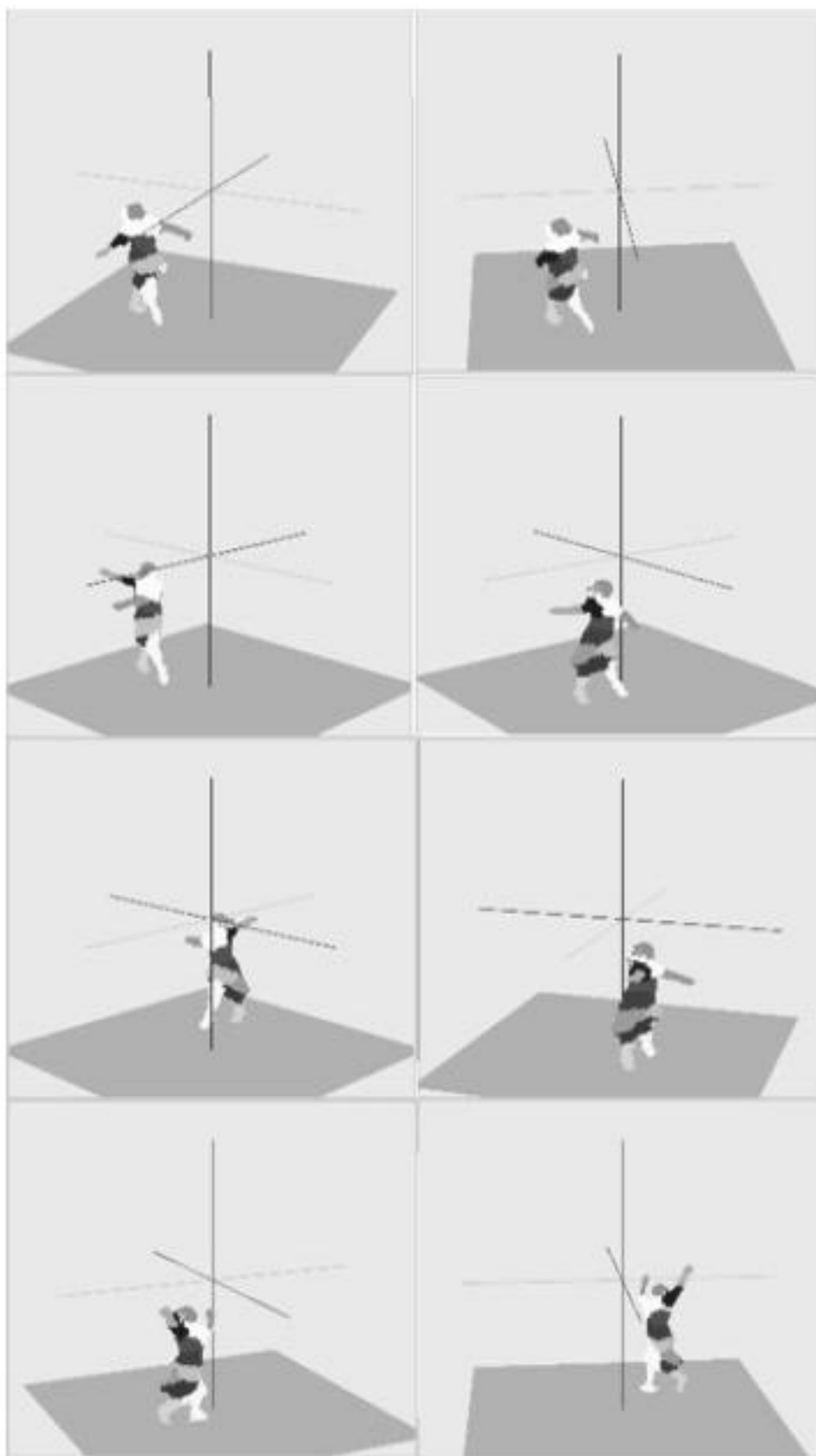


FIGURA 6 – Seqüência de rastreamento.

Na Figura 7, são apresentados os primeiros resultados do fluxo óptico.

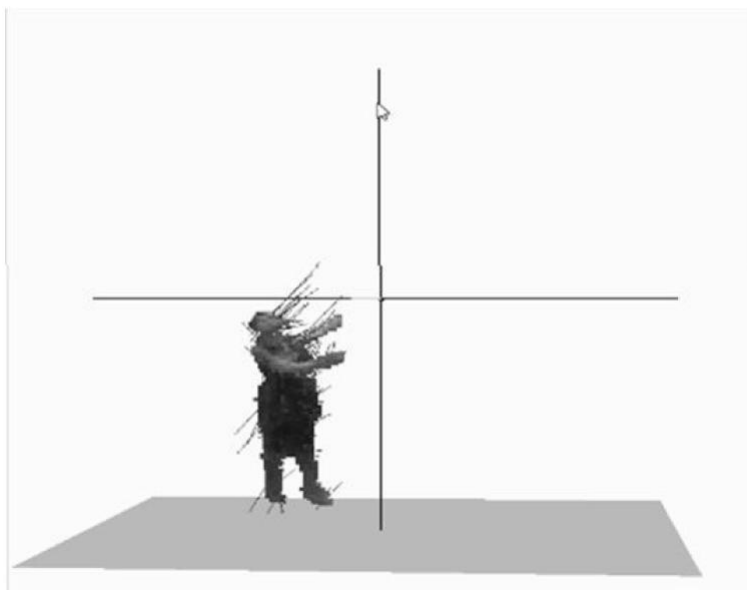


FIGURA 7 – Fluxo óptico.

7- CONCLUSÃO

Os principais focos deste trabalho foram: a técnica de reconstrução 3D empregada (recentemente publicada); e a utilização de um método de rastreamento de movimento. Este trabalho serve como uma base para se investigar a técnica de reconstrução 3D abordada: o Grid de Ocupação Probabilístico.

O sistema desenvolvido foi utilizado para testar as técnicas mencionadas, demonstrando-se eficiente. Principalmente, quanto à etapa de reconstrução probabilística 3D, que apresentou vantagens consideráveis em relação aos tradicionais métodos. Este fato deve-se à vantagem do método de reconstrução probabilística considerar as informações de todas as imagens em conjunto, sem realizar avaliações em separado para cada imagem. Contrariamente, os métodos tradicionais de reconstrução falharam na análise de partes do objeto de interesse cuja cor se assemelhava com a cor de fundo da cena, além disso, os métodos tradicionais apresentaram-se mais suscetíveis a variações de luminosidades (sombras).

REFERÊNCIAS

- [1] X. Zhao and Y. Liu, "Tracking 3D human motion in compact base space", IEEE WACV'07, 2007.
- [2] T. B. Moeslund and E. Granum, "A survey of computer vision based human motion capture", CVIU, vol. 81, no. 3, pp. 231–268, 2001.
- [3] P. Prez, J. Vermaak, and A. Blake, "Data fusion for visual tracking with particles", Proc. IEEE, vol. 92, no. 3, pp. 495–513, 2004.
- [4] Y. Kobayashi, D. Sugimura, K. Hirasawa, N. Suzuki, H. Kage, Y. Sato, and A. Sugimoto, "3D head tracking using the particle filter with cascaded classifiers", British Machine Vision Conference, pp. 1–37, 2006.
- [5] F. Caillette, "Real-time markerless 3D human body tracking", PhD Thesis, University of Manchester, 2006.
- [6] I. Mikic, M. Trivedi, E. Hunter, and P. Cosman, "Human body model acquisition and tracking using voxel data", IJCV, 2003.
- [7] R. Kehl, L. Gool, and D. Gavrilu, "Markerless motion capture of complex human movements from multiple views", Doctor dissertation, Swiss Federal Institute of Technology Zurich, 2006.
- [8] B. D. Allen, G. Bishop, and G. Welch, "Course 11 tracking: Beyond 15 minutes of thought", SIGGRAPH 2001 Courses, 2001.
- [9] F. Silva, "Um sistema de animação baseado em movimento capturado", M.Sc. Thesis, Federal University of Rio de Janeiro (UFRJ) - COPPE/Sistemas, march 1998.
- [10] J. S. Franco and E. Boyer, "Fusion of multi-view silhouette cues using a space occupancy grid", ICCV 05, 2005.
- [11] C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.
- [12] R. L. Barbosa, R. B. A. Gallis, J. F. C. Silva, and M. M. Júnior, "A computação do fluxo óptico em imagens obtidas por um sistema móvel de mapeamento terrestre", Revista Brasileira de Cartografia, 2005.
- [13] J. L. Barron and N. A. Thacker, "Tutorial: Computing 2D and 3D optical flow", in Tina Memo Internal, Imaging Science and Biomedical Engineering Divison, University of Manchester, 2005, no. 2004-012.
- [14] S. S. Beauchemin and J. L. Barron, "The computation of optical flow", ACM Computing Surveys, vol. 27, no. 3, pp. 433–467, 1995.
- [15] P. Group. (2008) Perception's website. [Online]. Available: <http://perception.inrialpes.fr>
- [16] R. A. Bem, E. Arnaud, and E. Boyer, "Intership report perception group", Perception Group, INRIA Rhöone-Alpes, Tech. Rep., 2007.

- [17] A. Elfes, "Using occupancy grids for mobile robot perception and navigation", IEEE Computer, Special Issue on Autonomous Intelligent Machines, vol. 22, no. 6, pp. 46–57, 1989.
- [18] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, 1997.
- [19] M. Galo and C. L. Tozzi, "A representação de matizes de rotação e o uso de quatérnions em ciências geodésicas", Série em Ciências Geodésicas, 2001.
- [20] B. K. P. Horn and B. G. Schunck, "Determining optical flow", Artificial Intelligence, vol. 17, pp. 185–203, 1981.
- [21] Open Source Computer Vision Library Reference Manual, Intel Corporation, U.S.A., 2001. [Online]. Available: <http://developer.intel.com>

AGRADECIMENTOS

Os autores agradecem à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq.