# UTILIZANDO VISÃO COMPUTACIONAL PARA RECONSTRUÇÃO PROBABILÍSTICA 3D E RASTREAMENTO DE MOVIMENTO

GISELE M. SIMAS[1], GUILHERME P. FICKEL[2], LUCAS NOVELO[3], RODRIGO A. DE BEM[4], SILVIA S. C. BOTELHO[5]

Universidade Federal do Rio Grande – FURG

[1] Mestranda em Modelagem Computacional – gisele_simas@yahoo.com.br

[2] Mestrando em Computação - guilhermefickel@gmail.com

[3] Graduando em Engenharia de Computação – lucasnovelo@gmail.com

[4] Professor Mestre em Engenharia Elétrica - rodrigo.bem@gmail.com

[5] Professora Doutora em Informática e Telecomunicações - silviacb@furg.br

## RESUMO

Este artigo apresenta um método não intrusivo para rastreamento de movimento 3D em ambientes monitorados por múltiplas câmeras. Primeiramente, se obtém uma reconstrução volumétrica 3D, através da técnica de Grid de Ocupação Probabilístico, tal técnica ainda foi pouco explorada no contexto de rastreamento de movimento. Então, utiliza-se o algoritmo Expectation-Maximization em conjunto com um modelo de representação do corpo do objeto de interesse baseado em blobs Gaussianas, para identificar e rastrear o movimento das partes do corpo do objeto de interesse.

**Palavras-Chaves:** reconstrução volumétrica probabilística 3D, rastreamento de movimento, múltiplas câmeras, Grid de Ocupação Probabilístico.

## ABSTRACT

## USING COMPUTER VISION FOR 3D PROBABILISTIC RECONSTRUCTION AND MOTION TRACKING

This paper presents an approach to the 3D visual tracking problem into multi-camera environments. This proposal executes the markerless visual tracking observing the environment through a model based in a volumetric reconstruction technique, called 3D Probabilistic Occupancy Grid, which is still seldom used for this purpose. The target is tracked by the use of Expectation-Maximization algorithm with an object representation model constructed with Gaussians blobs representing the object body parts.

**Keywords:** 3D probabilistic volumetric reconstruction, motion tracking, multiple cameras, Probabilistic Occupancy Grid.

## 1- INTRODUÇÃO

A **captura de movimento** é o processo de determinação de parâmetros da postura de um determinado objeto - muitas vezes associado ao movimento humano - no espaço 3D [1]. É um problema complexo, que pode ser decomposto nas seguintes etapas: inicialização, rastreamento, estimação de pose e reconhecimento do movimento [2].

A etapa de **inicialização** abrange as ações necessárias para assegurar que o sistema inicie a sua operação com uma correta interpretação da cena atual. O **rastreamento** é a detecção e a localização recursiva de objetos ou, mais geralmente, de padrões em seqüências de imagens (vídeos) [3]. Em sua forma mais simples o rastreamento compõe-se de um modelo de observação do espaço sensoriado, um modelo de representação do objeto rastreado e um algoritmo de rastreamento. Enquanto que as etapas de **estimação de pose** e de **reconhecimento** são, respectivamente, a identificação de como um corpo humano (ou outro objeto) está configurado no espaço 3D (ângulos e orientações de juntas); e a classificação do tipo de movimento capturado.

O **rastreamento visual** é uma tarefa importante na área de visão computacional [4] e pode ser empregado na captura de movimento, preparando os dados das imagens para as etapas posteriores da captura de movimento (estimação de pose e reconhecimento) [2].

A **reconstrução volumétrica 3D**, por sua vez, consiste na determinação do volume e da localização de objetos de interesse. Estas informações são úteis para o método de rastreamento, bem como, para outras aplicações, tais como: na construção de animações realísticas, análise espacial, reconstrução de ambientes e visualização científica. Além disso, a reconstrução volumétrica tem sido freqüentemente utilizada em conjunto com captura de movimento [5], [6], [7].

O rastreamento aplicado à captura de movimento vem sendo estudado amplamente na literatura, constituindo-se em um problema em aberto e fonte de diversas pesquisas [5], [6], [7]. Este interesse advém da variedade de aplicações para este problema, tais como: interface homem-máquina, estudos antropológicos, entretenimento, educação e vigilância.

A execução do rastreamento de movimento com emprego de técnicas de visão computacional sem marcas apresenta vantagens em relação a outros métodos, tais como sistemas mecânicos, magnéticos e acústicos [8], [9], visto

que: a) não é uma abordagem intrusiva (não necessita de equipamentos colocados na pessoa em movimento); b) não exige a replicação de equipamentos para o rastreamento de múltiplas pessoas simultaneamente; c) permite a obtenção de informações adicionais quanto às características físicas dos objetos (como cor e textura).

No entanto, rastrear pessoas ou objetos usando imagens 2D é um problema com dificuldades a serem tratadas, tais como a natureza complexa do movimento 3D; a perda de informações em imagens devido às restrições do espaço 2D; variação de intensidade dos pixels devido à luminosidade do meio; existência de outros objetos no ambiente em que a pessoa/objeto se movimenta. Assim, para minimizar alguns destes problemas, múltiplas câmeras podem ser utilizadas no sensoriamento, de forma a permitir a aferição de profundidade, bem como, aproveitar a redundância das informações de uma mesma região, diminuindo a incerteza da informação a ser tratada.

Assim, o sistema de rastreamento visual proposto, neste trabalho, utiliza como entrada um conjunto de imagens obtidas por múltiplas câmeras sincronizadas, dispostas ao redor de uma sala. Para a fusão das informações contidas nas imagens 2D, é utilizada a técnica de Grid de Ocupação [10], a qual posterga a definição de pertinência dos pixels das imagens ao objeto em movimento para depois da junção das informações obtidas por todas as câmeras, o que é realizado através de Inferência Bayesiana [11].

Dessa forma, se propõe um sistema capaz de realizar: i. a reconstrução probabilística 3D, através da técnica de Grid de Ocupação Probabilístico; e ii. o rastreamento, através do método Expectation-Maximization.

Também se apresenta, neste artigo, os primeiros resultados para agregar a informação de fluxo óptico [12], [13], [14] ao grid probabilístico, analisando os pixels de cada imagem 2D, antes de fazer qualquer segmentação ou fusão de imagens, a fim de auxiliar a tarefa posterior de rastreamento.

Inicialmente, na Seção 2, é apresentada a arquitetura da solução proposta. Prosseguindo, a Seção 3 relata o modelo de observação da cena: a reconstrução volumétrica 3D obtida pela técnica de Grid de Ocupação Probabilístico. Após, a Seção 4 é dedicada ao modelo de representação do objeto rastreado: as Blobs Gaussianas. A Seção 5 trata sobre o algoritmo empregado no rastreamento: o Expectation-Maximization. A seção 5.1 apresenta

a contribuição relacionada ao Fluxo óptico para reconstrução e rastreamento 3D. Finalizando, as Seções 6 e 7 apresentam, respectivamente, alguns resultados e a conclusão do trabalho.

## 2- SOLUÇÃO PROPOSTA

Um sistema de rastreamento pode ser especificado como tendo os seguintes componentes: inicialização, modelo de observação; modelo de representação; e algoritmo de rastreamento.

Em um primeiro momento, é necessária a realização de algumas inicializações com relação ao modelo de fundo da cena a ser tratada e do modelo de representação a ser adotado. Posteriormente, é necessária a especificação do modelo de observação das informações contidas nas imagens das câmeras. Algumas possíveis fontes de informação são: cores, contornos dos objetos de interesse, reconstrução volumétrica 3D, etc. Logo após, é necessário estabelecer um modelo de representação do objeto a ser rastreado, como modelos de representação podem-se citar: formas geométricas (elipsóides), estruturas articuladas (esqueletos), modelos gaussianos (blobs), etc. Finalizando, tem-se o rastreamento propriamente dito, que pode ser realizado por diversos tipos de algoritmos (Filtro de Kalman, Filtros de Partículas, Expectation-Maximization, etc).

A solução adotada, neste trabalho, é apresentada na Figura 1. O sistema desenvolvido utiliza, como entrada, uma seqüência de imagens 2D, capturadas por múltiplas câmeras sincronizadas, dispostas ao redor de um ambiente cujo objeto de interesse se movimenta. Para evitar a necessidade de se montar esta estrutura complexa de câmeras, foram utilizadas imagens do repositório público[15].

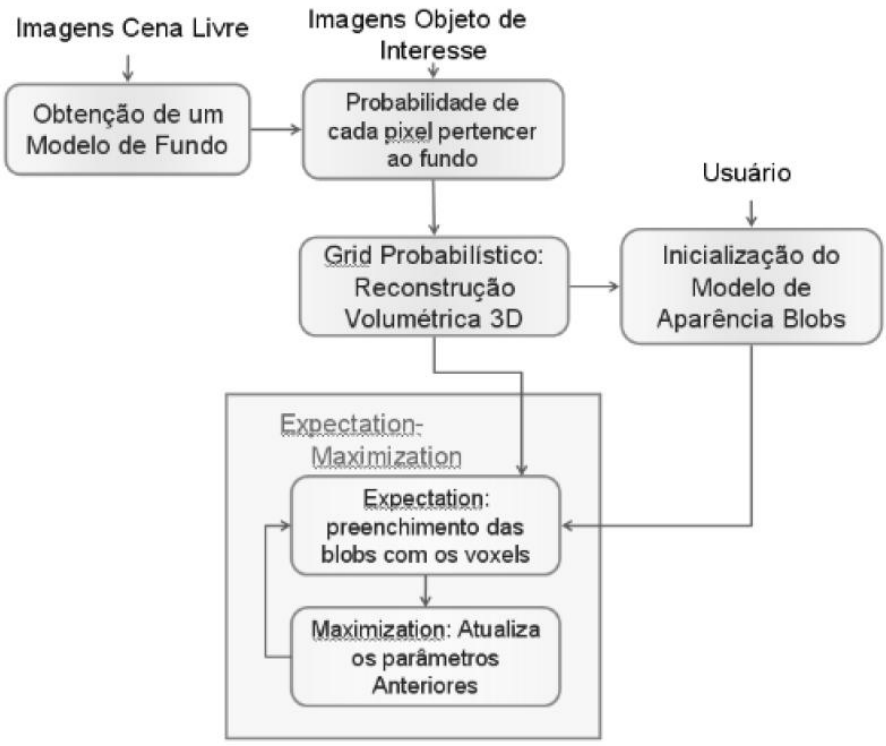


FIGURA 1 - Esquema da Solução Adotada.

**Inicialização.** Neste trabalho, é utilizado um método semi-automático de inicialização. Um modelo estatístico de fundo da cena é construído, a partir de imagens livres de objetos em movimento, para permitir a posterior segmentação entre pixels referentes ao fundo da cena e pixels referentes ao objeto de interesse. Além disso, um usuário deve fornecer um modelo inicial do objeto a ser rastreado com informações das dimensões, posições e orientações de cada parte do corpo deste objeto.

O **modelo de observação** utilizado é constituído da reconstrução volumétrica 3D e das informações de intensidade luminosa das imagens. Ou seja, o espaço da cena analisada é discretizado em elementos de volume (voxels): para cada voxel, é associado um valor de probabilidade de este estar ocupado por algum objeto de interesse; e um valor de intensidade luminosa.

A reconstrução volumétrica (valor de probabilidade de cada voxel estar preenchido pelo objeto de interesse) é realizada com base em [10] e obtida pela técnica de Grid de Ocupação Probabilístico.

O **modelo de representação** adotado é constituído por entidades estatísticas (modelos gaussianos) [5] que consideram informações de posição e

intensidade luminosa. Ou seja, cada parte do objeto de interesse é modelada por um modelo gaussiano nomeado de blob.

**Rastreamento.** Após terem sido obtidos os modelos de observação e representação, estas informações são utilizadas para rastrear o movimento do objeto de interesse, através do método Expectation-Maximization (EM) [5].

## 3- MODELO DE OBSERVAÇÃO

O problema tratado nesta seção é a reconstrução volumétrica probabilística 3D a partir de múltiplas câmeras [10], [16]. Na reconstrução volumétrica, em ambientes monitorados por múltiplas câmeras, freqüentemente se utiliza uma subtração binária do fundo, analisando cada imagem em separado. Este método pode dramaticamente alterar a percepção 3D que se teria, caso se observasse todas as imagens em conjunto. Intuitivamente, o conhecimento de todas as imagens simultâneas traz mais informações que o conhecimento de apenas uma imagem [10].

Analisando este contexto, Franco [10] propôs calcular a fusão das informações de todas as imagens no espaço 3D, antes de realizar avaliações sobre cada imagem individualmente. Para isto, Franco [10] utilizou a técnica de Grid de Ocupação Probabilístico. Esta técnica tem sido amplamente utilizada, na comunidade de robótica, para representar o ambiente de navegação de robôs monitorado por sensores de profundidade e de medidas de orientação [17]. Franco [10] propôs, então, estender o conceito do Grid de Ocupação para sensores baseados em imagens.

Nesta técnica, cada pixel da câmera é tratado como um sensor estatístico suscetível a incertezas. O problema é, então, tratado como uma estimativa Bayesiana. O espaço 3D é discretizado em elementos de volumes, nomeados de voxels, como ilustrado na Figura 2, e para cada voxel se calcula a probabilidade deste estar ocupado pelo objeto de interesse.

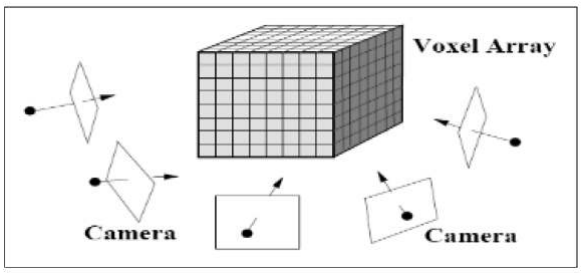


FIGURA 2 – Espaço 3D discretizado em voxels.

**Modelo de fundo.** Primeiramente, é necessária a construção de um modelo de fundo da cena livre de objetos em movimento. Assim, é adquirido um conjunto de imagens da cena livre de objetos. A partir destas se constrói um modelo estatístico de fundo [18]: cada pixel é modelado por uma distribuição gaussiana representada por um vetor média $\mu_p^i$ e uma matriz de covariância $\Sigma^i$. Para cada pixel $p$ de cada câmera $i$, é calculada uma média $\mu_p^i$ da sua cor no espaço YUV. A matriz de covariância $\Sigma^i$ é calculada apenas uma para cada câmera $i$. Sendo a dimensão da gaussiana $D = 3$, cada pixel terá a seguinte probabilidade de pertencer ao fundo:

$$N(\mu, \Sigma) \cong \frac{1}{(2\pi)^{\frac{D}{2}}\sqrt{|\Sigma|}} exp\left\{-\frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)^{\mathrm{T}}\right\} \quad (1)$$

No entanto, um grande número de multiplicações de pequenos valores pode sofrer underflow de precisão numérica; assim a expressão acima é transformada em uma soma de logaritmos (Equação 2). Além disso, a função logarítmica da probabilidade é mais conveniente, uma vez que o logaritmo uma função que cresce monotonicamente [11].

$$\log N(\mu, \Sigma) \cong -\frac{3}{2}\log(2\pi) - \frac{1}{2}\log(|\Sigma|) - \frac{1}{2}(x-\mu)\Sigma^{-1}(x-\mu)^{\mathrm{T}} \quad (2)$$

**Inferência Bayesiana.** As ocupações dos voxels, por questões de custo computacional, são consideradas estatisticamente independentes. E, portanto, se calcula a probabilidade de ocupação de cada voxel independentemente.

Como entradas do sistema, são utilizadas as seguintes informações: $I_p^i$

(cor relativa ao pixel $p$ da câmera $i$); $B^i$ (modelo estatístico de fundo da câmera $i$); $P^i$ (matriz de projeção da câmera $i$); $\tau$ (simboliza o conhecimento a priori que se tem sobre a cena ou sobre às características da câmera, neste trabalho, não se utilizou qualquer conhecimento a priori).

Como saída, o sistema de reconstrução deve fornecer: $G$, o grid de ocupação (Figura 2). Para cada ponto $X$ do grid é associada uma variável binária de ocupação $G_X \epsilon \{0,1\}$, o valor $0$ informa que o voxel está livre de qualquer objeto de interesse e o valor 1 que está ocupado.

Além disso, se define, para cada pixel $p$ de cada câmera $i$, uma variável binária $F_p^i$ que representa a detecção da silhueta do objeto de interesse. $F_p^i = 1$ se a câmera $i$ informa que o pixel $p$ pertence a um objeto de interesse.

As relações de dependências entre as variáveis foram modeladas de acordo com [10]:

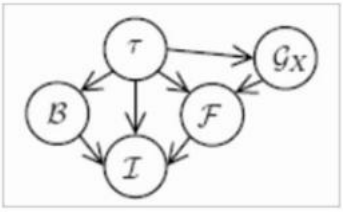


FIGURA 3 – Relações de dependência entre as variáveis.

Assim, para inferir a probabilidade de ocupação $G_X$, deve-se computar a probabilidade conjunta $p(G_X, I, B, F, \tau)$:

$$p(G_X, I, B, F, \tau) = p(\tau)p(B|\tau)p(G_X|\tau)p(F|G_X, \tau)p(I|F, B, \tau) \quad (3)$$

$p(\tau)$, $p(B|\tau)$, $p(G_X|\tau)$ são, respectivamente, a probabilidade a priori do conjunto de parâmetros; a probabilidade prévia do fundo da cena; e a probabilidade prévia de ocupação do voxel $X$. Estes parâmetros foram definidos como distribuições uniformes.

$p(F|G_X, \tau)$ é o termo de probabilidade da silhueta, depende do conhecimento prévio do sistema e da ocupação do voxel relacionado ao pixel $(p, i)$, pois a ocupação de um voxel que é projetado sobre $p$ na imagem da câmera $i$ explica a detecção do objeto nesta mesma imagem.

$p(I|F,B,\tau)$ é o termo de probabilidade da imagem. As cores $I$ nas imagens são condicionadas pela detecção $F$ de objetos nestas imagens e pelo modelo de fundo da cena $B$.

Desconsiderando as distribuições uniformes e tratando as observações de cor $I$ de cada pixel como independentes uma das outras, pode-se simplificar a Equação 3:

$$p(G_X, I, B, F, \tau) = \prod_{i,p} p\left(F_p^i | G_X, \tau\right) p(I_p^i | F_p^i, B_p^i, \tau) \quad (4)$$

**Termo da Formação da Silhueta.** $p\left(F_p^i | G_X, \tau\right)$ modela a resposta da detecção da silhueta do pixel $(p,i)$ para o estado de ocupação $G_X$ do voxel $X$.

a) Se $G_X = 1$, o voxel $X$ está ocupado por um objeto de interesse, a detecção da silhueta é, então, regida pela distribuição de probabilidade $P_d\left(F_p^i\right)$. Na prática, esta distribuição é definida utilizando uma constante $P_D \in [0,1]$, que é parâmetro do sistema: i. $P_d\left(\left[F_p^i = 1\right]\right) = P_D$ é a taxa de detecção correta de silhueta em um pixel, e ii. $P_d\left(\left[F_p^i = 0\right]\right) = 1 - P_D$ é a taxa de falha nesta detecção. Falhas na detecção ocorrem quando o pixel relata que não existe qualquer objeto sobre a sua linha de visão quando na verdade existe. Esta consideração é útil para este problema, pois algumas vezes a extração da silhueta falha localmente e a consideração de incertezas permite que o sistema recupere a informação correta do voxel.

b) Se $G_X = 0$ o voxel $X$ não está ocupado e, portanto, não podemos tirar conclusões sobre o seu estado de detecção. Logo, a detecção é regida pela distribuição $P_f\left(F_p^i\right)$. Esta distribuição é definida utilizando uma constante $P_{FA} \in [0,1]$, parâmetro do sistema: i. $P_f\left(\left[F_p^i = 1\right]\right) = P_{FA}$ é a taxa de falsos alarmes de um pixel. Falsos alarmes ocorrem quando erroneamente um pixel demonstra a existência de algum objeto sobre sua linha de visão, quando na verdade não existe. ii. $P_f\left(\left[F_p^i = 0\right]\right) = 1 - P_{FA}$ é a taxa com a qual esperamos que o pixel $(p,i)$ informe corretamente a não-detecção de um objeto.

**Termo da Formação da Imagem.** $p(I_p^i | F_p^i, B_p^i, \tau)$ pode ser interpretado da seguinte forma:

a) Se $F_p^i = 1$, um objeto é encontrado no pixel $(p, i)$ e, portanto, o conhecimento sobre o fundo é irrelevante para definir a cor esperada para este pixel (o fundo é obstruído por um objeto, cuja cor não foi definida previamente). Neste caso, este termo é considerado como uma distribuição uniforme: $p(I_p^i|F_p^i, B_p^i, \tau) = U(I_p^i)$.

b) Se $F_p^i = 0$, nenhum objeto se encontra no pixel $(p, i)$, então a cor observada neste pixel deve ser similar à cor do modelo de fundo. Neste caso, o valor de probabilidade é calculado utilizando a distribuição gaussiana do modelo de fundo: $p(I_p^i|[F_p^i = 0], [B_p^i = (\mu_p^i, \Sigma^i)], \tau) = N(I_p^i|\mu_p^i, \Sigma^i)$.

**Inferência da Ocupação de cada Voxel.** Uma vez que as probabilidades conjuntas tenham sido determinadas, é possível utilizar o teorema de Bayes para inferir a probabilidade da variável buscada $G_X$:

$$p(G_X|I, B, \tau) = \frac{\sum_F p(G_X, I, B, F, \tau)}{\sum_{G_X,F} p(G_X, I, B, F, \tau)} = \frac{\prod_{i,p} \sum_{F_p^i} p(F_p^i|G_X, \tau) p(I_p^i|F_p^i, B_p^i, \tau)}{\sum_{G_X} \prod_{i,p} \sum_{F_p^i} p(F_p^i|G_X, \tau) p(I_p^i|F_p^i, B_p^i, \tau)} \quad (5)$$

Após, define-se um limiar para a probabilidade $p(G_X|I, B, \tau)$: todos os voxels cuja probabilidade passar desse limiar serão utilizados no rastreamento.

## 4- MODELO DE REPRESENTAÇÃO

O rastreamento de um objeto envolve encontrar a sua posição global, bem como, a posição relativa entre cada parte do corpo de este objeto, em cada frame da seqüência de vídeo. Objetos articulados, como o corpo humano, são, em sua maioria, formados por partes rígidas, que individualmente não apresentam modificações significativas em suas formas. Partindo desta suposição, neste trabalho, cada parte do objeto é representada por um modelo gaussiano, nomeado de blob.

Com referência a informação espacial, uma blob é freqüentemente representada por uma forma elipsoidal, tendo a sua superfície definida pelo

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 $D_M(V,B)$:

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

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:

$$mc = arg_{i=1\dots N_C} min\left(C_V^i - \mu_C\right)\Sigma_C^{-1}\left(C_V^i - \mu_C\right)^T \qquad (7)$$

**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:

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

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:

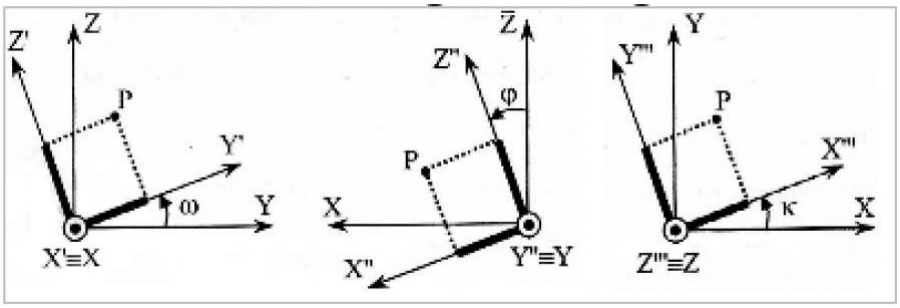


FIGURA 4 - Ângulos de Euler - para a determinação da matriz de rotação R.

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):

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

$$R = \begin{pmatrix} c\varphi c\kappa & s\omega s\varphi c\kappa + c\omega s\kappa & -c\omega s\varphi c\kappa + s\omega s\kappa \\ -c\varphi s\kappa & -s\omega s\varphi s\kappa + c\omega s\kappa & c\omega s\varphi s\kappa + s\omega c\kappa \\ s\varphi & -s\omega c\varphi & c\omega c\varphi \end{pmatrix} \tag{9}$$

## 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 $\mu_j$ e pela matriz de covariância $\Sigma_j$, 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:

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

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:

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

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:

$$\phi(V_i, B_j) \ = \log|\Sigma_j| + D_M(V_i, B_j) \tag{12}$$

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 $\Sigma_j$ 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 $\mu$ e para a matriz de covariância $\Sigma$ 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).

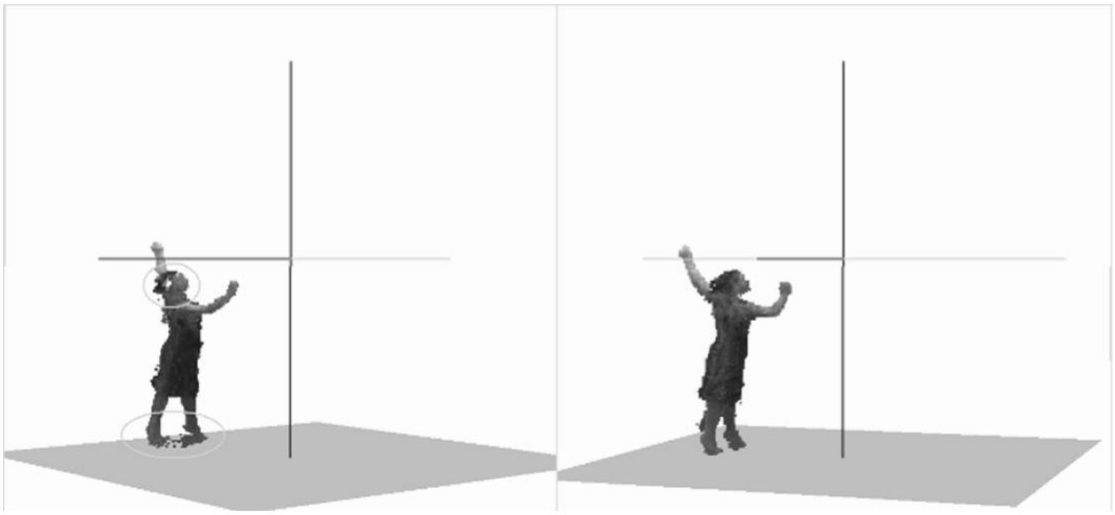

FIGURA 5 - Reconstrução volumétrica. Esquerda: método tradicional Shape-from-silhouette (falhas em verde). Direita: técnica Grid de Ocupação Probabilístico.

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.

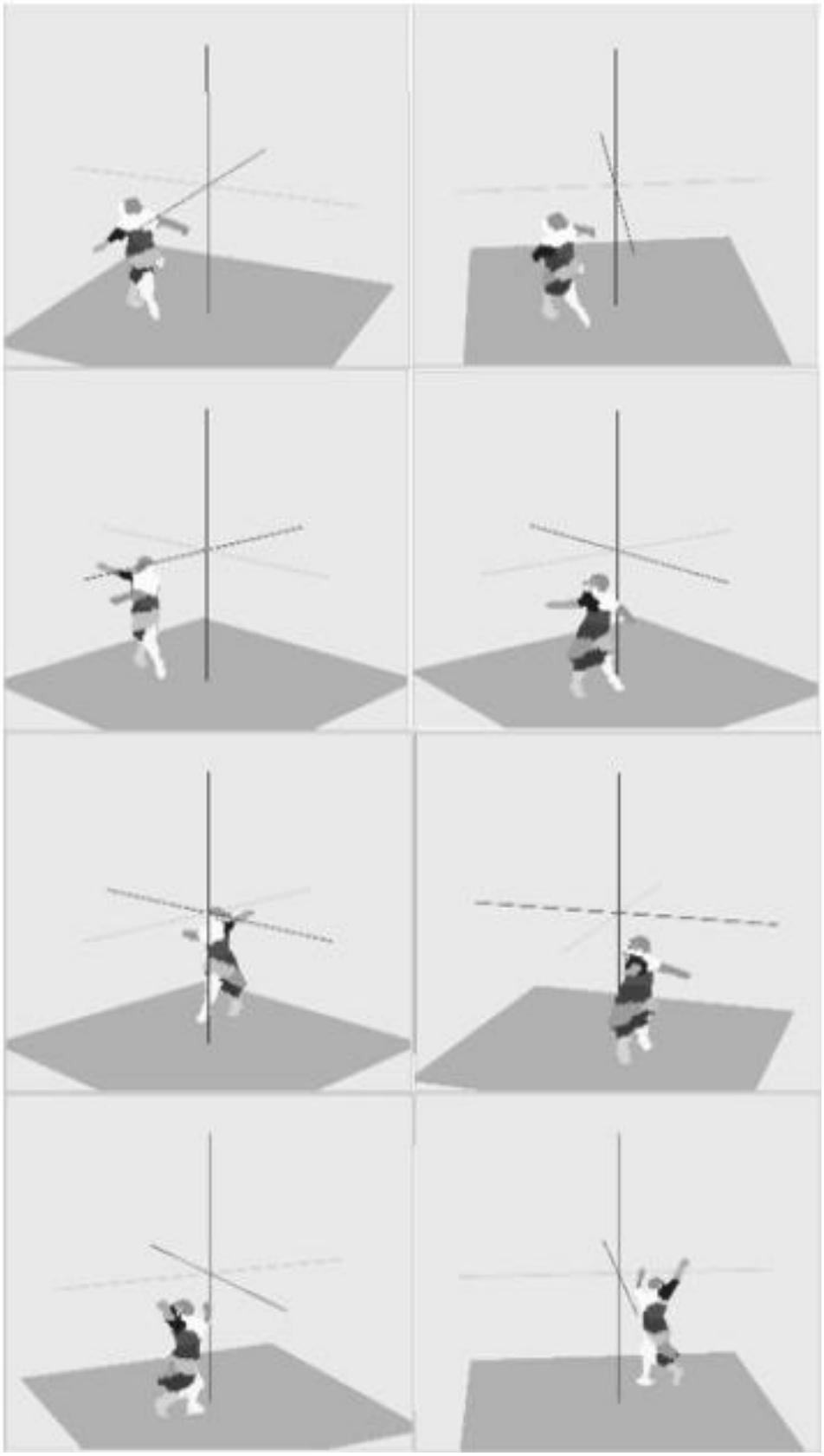

FIGURA 6 – Seqüência de rastreamento.

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

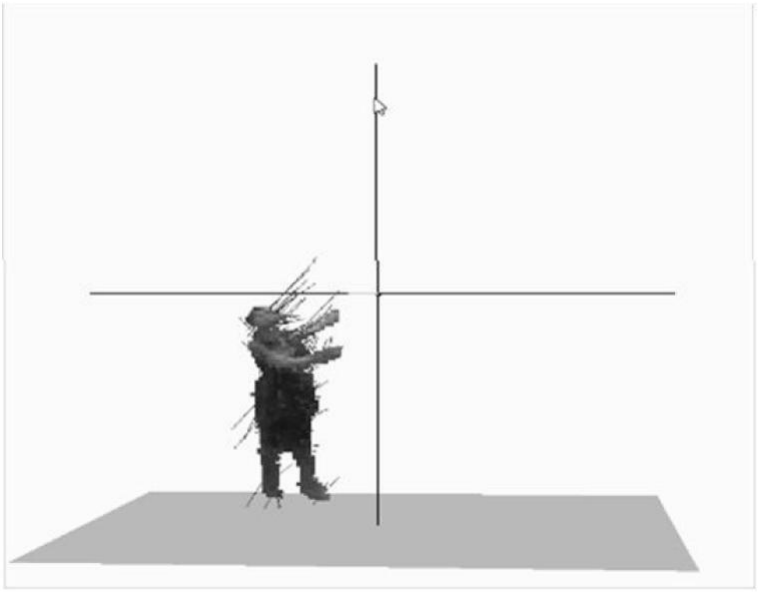

FIGURA 7 – Fluxo óptico.

## 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. Gavrila, “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. Azarbayeajani, 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.