

EXTRAÇÃO DE CARACTERÍSTICAS MÉTRICAS LOCAIS A PARTIR DE GRADE DE OCUPAÇÃO USANDO MÉTODO DE VOTAÇÃO PONDERADA

DIOGO P. F. PEDROSA,* ADELARDO A. D. MEDEIROS,* PABLO J. ALSINA*

*DCA – CT – UFRN
59078-900, Natal, RN, Brasil

Emails: diogo@dca.ufrn.br, adelardo@dca.ufrn.br, pablo@dca.ufrn.br

Abstract— Nowadays there is an increasing interest about cognitive description of indoor environments. This kind of representation is obtained by classifying all local indoor spaces as rooms or corridors. To accomplish this local spaces classification, it is necessary the knowledge of local geometry. This geometry can be summarized by the best geometric figure that fits the free local space. This paper introduces a method to compute the parameters of the geometric figure by a weighted voting process. This method is inspired on the Generalized Hough Transform. The value of each vote is determined by the probability of a small region of local environment belonging to an obstacle. The probability is given by a occupancy grid previously constructed by the robot. Results obtained from actual sensor data are presented.

Keywords— Mobile robot, mapping, Generalized Hough Transform.

Resumo— Atualmente há um crescente interesse na descrição cognitiva de ambientes internos semi-estruturados, a qual se dá através da classificação de espaços locais em salas ou corredores. Um espaço local pode ser descrito através do conhecimento de parâmetros que descrevem uma figura geométrica que melhor se ajusta ao seu espaço livre. Este artigo apresenta uma forma de calcular estes parâmetros através de um procedimento de votação ponderada, inspirado na Transformada Generalizada de Hough. A ponderação de cada voto se dá pela probabilidade de que uma determinada região do espaço local pertença a um obstáculo. Esta informação é dada por uma grade de ocupação previamente construída pelo robô. Resultados obtidos a partir de informações sensoriais reais são apresentados para a validação do método.

Palavras-chave— Robô móvel, Mapeamento, Transformada Generalizada de Hough.

1 Introdução

Fazer um robô móvel obter uma representação do seu ambiente de trabalho é uma das tarefas mais importantes que se pode executar em robótica móvel. Um ambiente pode ser basicamente descrito através de duas formas distintas: (1) no formato métrico, utilizando representação em grade de ocupação (Moravec e Elfes, 1985; Borenstein e Koren, 1991) ou mapa de características (Pfister et al., 2003); ou (2) no formato topológico (Fabrizi e Saffiotti, 2000; Wetherbie III e Christopher, 2001). Neste caso, o mapa tem uma descrição mais abstrata do ambiente, armazenando informações sobre regiões sensorialmente homogêneas e suas relações de conectividade. Uma terceira corrente de representação agrega os formatos métrico e topológico em uma descrição única (Buschka e Saffiotti, 2004). Diversas propostas para a construção de mapas utilizando estes formatos foram apresentadas, inclusive considerando abordagens para resolver a localização de forma simultânea (Durrant-Whyte e Bailey, 2006a; Durrant-Whyte e Bailey, 2006b).

Recentemente, a representação cognitiva de ambientes vem despertando atenção por causa da popularização dos robôs domésticos. Embora a idéia de mapas cognitivos já tenha sido anteriormente utilizada (Arleo et al., 1999; Monteiro e Ribeiro, 2002), atualmente este conceito busca utilizar uma descrição de ambientes que seja compreensível tanto por robôs quanto por humanos. Por exemplo, é desejável que um robô doméstico possa identificar e classificar apropriadamente ambientes locais como “sala”, “cozinha” ou “quarto”. O trabalho de Vasudevan et al. (2007)

busca apresentar soluções para este tipo de mapeamento, usando visão e sensores de alcance para encontrar objetos característicos em salas distintas e, com isso, classificá-las adequadamente.

Fabrizi e Saffiotti (2002) também propuseram um método para obter as características de espaços livres dentro de um ambiente semi-estruturado, e com isso obter uma classificação de salas e corredores. No entanto, eles partiram de uma idéia baseada apenas nas características geométricas, como as dimensões de uma sala. Para realizar isto, eles utilizaram um mapa na forma de grade de ocupação previamente construído e o trataram como uma imagem em tons de cinza. Ferramentas de processamento de imagens eram aplicadas sobre este mapa para extrair as características métricas de regiões que permaneciam homogêneas após o seu processamento. Com isto, um mapa topológico enriquecido com informações métricas era construído.

Pedrosa et al. (2006) seguiram esta mesma linha de representação topológica. Eles descreveram um ambiente interno a partir da identificação dos espaços locais, baseando-se também em parâmetros geométricos, e das passagens entre estes espaços. Os dados utilizados eram obtidos diretamente de sensores de alcance e não sofriam qualquer tipo de pré-processamento. A construção do mapa era realizada durante a exploração do ambiente, diferentemente de Fabrizio e Saffiotti (2002) que a faziam *off-line*. A classificação dos ambientes locais baseou-se nos valores de parâmetros geométricos que descreviam completamente a área livre de uma sala ou de um corredor. Para isto, foi utilizado um método de votação

inspirado na Transformada Generalizada de Hough (Ballard, 1981). Pedrosa et al. (2006) consideravam que a forma geométrica padrão das salas exploradas era conhecida, o que permitia que poucos pontos fossem coletados pelos sensores de alcance e, conseqüentemente, utilizados pelo método de votação proposto.

Este artigo apresenta uma modificação sobre o método de votação apresentada por Pedrosa et al. (2006), em cada instância da figura geométrica desejada recebia um voto unitário. Aqui, a ferramenta de votação utilizada tem voto ponderado para cada instância encontrada. Esta ponderação depende da probabilidade de que uma determinada região do espaço esteja ocupada por um obstáculo. Tais probabilidades são dadas pelos valores das células de uma grade de ocupação local, construída pelo robô usando informações de alcance de seus sensores, e que deve ter o tamanho suficiente para englobar uma única sala ou corredor.

A razão para utilizar uma grade é porque ela consegue representar com maior fidelidade um espaço local interno (salas ou corredores), quando comparada com uma nuvem de pontos simples e não-tratada. Assim, por ter-se uma melhor definição dos limites do ambiente local, o cálculo dos parâmetros da figura geométrica desejada tende a ser mais preciso.

A estrutura deste trabalho está organizada da seguinte forma: na seção 2 está descrito a forma como todos os parâmetros geométricos que descrevem o espaço livre de uma sala ou corredor são calculados. Na seção 3 é mostrada como a solução pode ser iterativamente refinada. Em seguida, alguns resultados são apresentados na seção 4, utilizando a ferramenta proposta e, por fim, as conclusões obtidas estão na seção 5.

2 Cálculo da Geometria Local

Uma ferramenta comumente utilizada para encontrar formas geométricas que não possuem descrição analítica, em uma imagem qualquer, é a Transformada Generalizada de Hough (Ballard, 1981). Tendo uma imagem de referência com a forma desejada e uma imagem de busca, a Transformada faz com que um determinado elemento de um espaço acumulador multidimensional receba um voto, para cada instância da figura que é encontrada na imagem de busca. O número de dimensões deste espaço acumulador é igual ao número de parâmetros que descrevem a figura geométrica, e os seus elementos descrevem todas as suas possíveis configurações, de acordo com a precisão adotada. O valor que cada elemento armazena corresponde ao número de vezes que a forma foi encontrada na imagem de busca. No final do processo de voto, o elemento de maior valor no espaço acumulador indicará a configuração de forma geométrica de maior destaque.

Na área da robótica, a Transformada Generalizada de Hough tem sido pouco utilizada para mapeamento de ambientes. O motivo disso são os resul-

tados satisfatórios apresentados com métodos que utilizam a Transformada Padrão de Hough para retas, e o fato da Transformada Generalizada ser uma ferramenta que exige um custo computacional alto. Uma exceção na literatura foi o trabalho de Chen e Tsai (1998) que apresentou uma Transformada Generalizada Ponderada para atualizar a pose do robô, dado o conhecimento do mapa do ambiente.

Aqui, neste trabalho, procurou-se utilizar um procedimento de voto inspirado na proposta de Ballard (1981), mas sem utilizar informações dadas a partir de imagem de referência, ou outros elementos que existem no algoritmo original da Transformada Generalizada de Hough, como a tabela-R (Ballard, 1981). A idéia principal é utilizar este procedimento de voto para encontrar os melhores parâmetros da forma geométrica que descreve a área livre de espaços locais internos, como salas ou corredores. Tendo uma descrição compacta da geometria destes espaços locais, é possível implementar um mapeamento topológico, ou ser coadjuvante em uma metodologia de mapeamento cognitivo do espaço.

É assumido que a forma geometria de um ambiente interno local, como sala ou corredor, pode ser satisfatoriamente representada por um retângulo. Como esta figura não possui uma forma geral analítica, pode-se caracterizá-la com a seguinte forma paramétrica:

$$\mathbf{r} = [x, y, \theta, l, c]^T \quad (1)$$

onde (x, y) é o centro do retângulo, θ é a orientação deste retângulo em relação ao eixo x , e l e c são, respectivamente, a largura e comprimento do retângulo (ver figura 1).

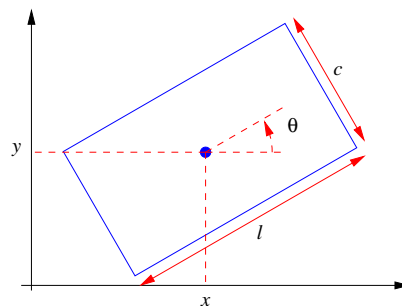


Figura 1: Parâmetros de um retângulo em relação a um sistema de coordenadas.

Como a geometria local é descrita por cinco parâmetros, então cria-se um espaço vetorial acumulador $A(x, y, \theta, l, c)$, com cinco dimensões, para receber os votos dos retângulos que são encontrados na grade local. Cada dimensão possui, respectivamente, N_x , N_y , N_θ , N_l e N_c intervalos de discretização. Inicialmente, todos os elementos deste espaço acumulador têm valores nulos. Os intervalos de busca para cada dimensão são delimitados da seguinte forma:

- Para as coordenadas x e y do centro do retângulo, os espaços de busca serão $[x_{\min}, x_{\max}]$ e

$[y_{\min}, y_{\max}]$, onde estes valores são os respectivos valores mínimos e máximos das coordenadas da grade de ocupação;

- Para a orientação, o espaço será $[\theta_{\min}, \theta_{\max}]$, com $\theta_{\min} = -\pi/4$ e $\theta_{\max} = \pi/4$. Para $|\theta| > \pi/4$, os valores de l e c são permutados;
- Para o comprimento e largura, os espaços serão dados por $l_{\min} = c_{\min} = 0$ e $l_{\max} = c_{\max} = d_{\max}$, onde d_{\max} é a medida da diagonal da grade.

O algoritmo 1 descreve o método de votação ponderada para determinar o retângulo que se ajusta à informação presente na grade de ocupação. Os dados de entrada são, portanto, a grade local de ocupação e o número de intervalos de discretização para cada dimensão.

Definir os espaços de busca para os parâmetros do retângulo;
 Criar um espaço acumulador $A(N_x + 1, N_y + 1, N_\theta + 1, N_l + 1, N_c + 1)$ com todos os seus elementos iguais a zero;
 Para cada (x_i, y_j) da grade:
 Se a probabilidade $p(x_i, y_j) > \gamma$
 Para cada $\theta_k \in [-\pi/4, \pi/4]$:
 Para cada $l_m \in [0, d_{\max}]$:
 Para cada $c_n \in [0, d_{\max}]$:
 Para cada (x_q, y_p) nos limites do retângulo $\mathbf{r} = [x_i, y_j, \theta_k, l_m, c_n]^T$, incrementar o elemento $A(p, q, k, m, n)$ com $p(x_i, y_j)$.
 Buscar o elemento de maior valor no espaço A e encontrar os parâmetros do retângulo \mathbf{r} desejado.

Algoritmo 1: Método de votação ponderada para detecção de retângulos.

O valor de $p(x_i, y_j) \in [0, 1]$ é a probabilidade de uma determinada célula, centrada em (x_i, y_j) , estar ocupada por obstáculos. O limiar γ foi utilizado com o objetivo de reduzir a quantidade de pontos da grade a serem analisados. Como o retângulo é calculado sobre a informação de ocupação das células, aquelas que apresentarem valores próximos a zero, ou valores que indiquem indeterminação sobre a presença de obstáculos, podem ser naturalmente descartadas no processo de votação. Ressalta-se aqui que não há processamento prévio da grade com quaisquer outros tipos de ferramentas de processamento de imagens.

O maior problema deste algoritmo é o seu esforço computacional. Somente para o armazenamento de A é exigido um custo da ordem $O[(N_x + 1) \cdot (N_y + 1) \cdot (N_\theta + 1) \cdot (N_l + 1) \cdot (N_c + 1)]$. O custo total do algoritmo 1 também depende do número de elementos na grade. Mesmo com a adoção do limiar γ que reduz a quantidade de pontos a serem analisados, também é preciso adotar uma baixa discretização em todas as dimensões do espaço acumulador para diminuir o esforço utilizado no algoritmo 1. Porém, a redução da discretização implica a redução da precisão do retângulo. Desta forma, foi proposto um método de refinamento iterativo para que a solução encontrada estivesse dentro dos limites de precisão desejado, mesmo

trabalhando com uma baixa discretização no espaço acumulador.

3 Refinamento Iterativo da Solução

A abordagem utilizada para o refinamento do retângulo consiste em manter uma baixa discretização em todas as dimensões do espaço acumulador e aplicar o algoritmo 1 de forma iterativa. Em cada iteração, há uma redefinição nos limites dos espaços de busca de cada parâmetro do retângulo. Os novos limites são definidos baseando-se no retângulo encontrado na iteração anterior e na largura do intervalo de discretização. Este processo é repetido até que a norma euclidiana entre o retângulo encontrado na iteração atual e o retângulo dado pela iteração anterior seja menor que um limite pré-especificado. Percebe-se, portanto, que o número de intervalos de discretização permanece com um valor baixo e constante, mas a largura de cada intervalo é iterativamente alterada, visando o afunilamento dos seus limites em torno dos valores reais dos parâmetros do retângulo. O efeito produzido por esta metodologia é similar ao aumento da resolução do espaço vetorial acumulador dentro da região de interesse. É um procedimento inspirado no método da bisseção para encontrar raízes de funções não-lineares.

Para a iteração inicial, os limites dos intervalos de busca são calculados como indicado anteriormente na seção 2. Cada subdivisão dos intervalos de busca tem uma largura igual a:

$$\begin{aligned} \delta_x &= \frac{|x_{\max} - x_{\min}|}{N_x} & \delta_y &= \frac{|y_{\max} - y_{\min}|}{N_y} \\ \delta_l &= \frac{|l_{\max} - l_{\min}|}{N_l} & \delta_c &= \frac{|c_{\max} - c_{\min}|}{N_c} \\ \delta_\theta &= \frac{|\theta_{\max} - \theta_{\min}|}{N_\theta} \end{aligned} \quad (2)$$

Sendo o algoritmo 1 repetidamente aplicado, então na k -ésima iteração tem-se $\mathbf{r}_k = [x_k, y_k, \theta_k, l_k, c_k]^T$. Porém, por causa da baixa discretização, valores mais precisos dos parâmetros do retângulo podem ser encontrados numa iteração seguinte, realizando o afunilamento do espaço de busca dos parâmetros do retângulo em torno do valor atual encontrado. Assim, os limites dos espaços de busca devem ser recalculados de acordo com as equações 3 para serem aplicados na iteração seguinte. Em seguida reapplicam-se as eqs. 2 para calcular as novas larguras dos intervalos de discretização e, com isso, dar continuidade ao método.

$$\begin{aligned} x_{\min} &= x_k - \delta_x & x_{\max} &= x_k + \delta_x \\ y_{\min} &= y_k - \delta_y & y_{\max} &= y_k + \delta_y \\ l_{\min} &= l_k - \delta_l & l_{\max} &= l_k + \delta_l \\ c_{\min} &= c_k - \delta_c & c_{\max} &= c_k + \delta_c \\ \theta_{\min} &= \theta_k - \delta_\theta & \theta_{\max} &= \theta_k + \delta_\theta \end{aligned} \quad (3)$$

Como em cada iteração tem-se um vetor retângulo como resposta, o critério de parada implementado ve-

rifica a norma entre \mathbf{r}_k e \mathbf{r}_{k-1} :

$$N = |\mathbf{r}_k - \mathbf{r}_{k-1}| \quad (4)$$

Quando ela tiver um valor inferior a uma precisão previamente definida, o processo iterativo é suspenso e tem-se o resultado final do retângulo. O algoritmo 2 sintetiza todo este processo.

Aplicar o alg. 1 para obter a estimativa inicial do vetor \mathbf{r} ;
 Enquanto a norma euclidiana for maior que uma tolerância pré-definida, faz-se:
 Redefinir os espaços de busca (eqs. 3);
 Recalcular larguras dos intervalos de discretização (eqs. 2);
 Aplicar o algoritmo 1;
 Calcular a norma (eq. 4).
 Apresentar o último retângulo encontrado como solução.

Algoritmo 2: Refinamento da precisão do método de votação ponderada.

A figura 2 mostra um exemplo do resultado obtido pela aplicação do algoritmo 2 em uma grade de ocupação construída manualmente. Nesta simulação, não foram incluídas as incertezas sobre a representação, devido a sensores exteroceptivos de robôs, nem buscou-se destacar questões relativas à presença de obstáculos no ambiente. A área livre da grade simulada possui dimensões iguais a (2.5, 3.1). O retângulo obtido pelo algoritmo foi:

$$\mathbf{r} = [1.645, 2.009, 0^\circ, 2.725, 3.197]$$

e está representado pela linha vermelha na figura 2, sobreposta à grade.

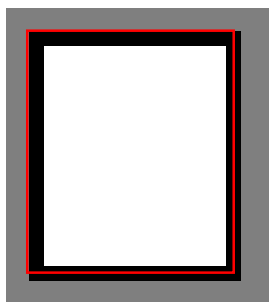


Figura 2: Retângulo obtido (linha vermelha) a partir de uma grade de ocupação simulada.

4 Resultados

Alguns resultados da aplicação do algoritmo 2 são apresentados nesta seção. As grades de ocupação utilizadas foram montadas a partir de dados de odometria e alcance, medido por sonares, obtidos de um robô *Pioneer 3AT*, em um dos corredores do Departamento de Engenharia de Computação e Automação da UFRN.

Para esta experiência, o robô foi manualmente controlado ao longo do corredor, que embora não possuísse obstáculos em seu interior, tinha irregularidades nas suas paredes que poderiam influenciar nas medidas de alcance dos sonares. Uma representação da planta baixa deste corredor é mostrada na figura 3. Todas as portas foram mantidas fechadas durante a coleta de dados. As suas dimensões são 23m de largura e 1.6m de comprimento, aproximadamente.

Para criar as grades, implementou-se o método de mapeamento proposto por Thrun et al. (2005). Foi utilizada uma resolução de 10cm para as grades. Para a aplicação do algoritmo 1, adotou-se $N_x = N_y = N_\theta = N_l = N_c = 15$. O limiar γ foi mantido em 0,5 e a tolerância para verificar a norma (eq. 4) foi igual a 0,025.

A figura 4 mostra a superposição do retângulo encontrado para uma grade de ocupação de um corredor, montada a partir de todas as medidas de sonar do *Pioneer 3AT*. Os valores obtidos dos parâmetros foram:

$$\mathbf{r} = [11.5, -0.45, -0.0133^\circ, 25.59, 3.30]^T$$

onde as unidades da posição central do retângulo e suas dimensões são dadas em metros.

Para este resultado, percebe-se que os valores encontrados para a largura e para o comprimento do corredor não correspondem à área livre real apresentada pela figura 3. Os erros relativos para a largura e comprimento são, respectivamente, 11.26% e 106.25%. O motivo desta discrepância de medidas é a quantidade de pontos marcados como “ocupados” na grade, devido ao modelo de incerteza do sonar, além das características próprias deste tipo de sensor, como falsas reflexões. Percebe-se na figura 4 que o retângulo “passa” por regiões ocupadas na grade que não são do corredor, mas sim originadas do modelo em cone do sonar.

Quando a grade de ocupação é construída com uma maior precisão, então o resultado dado pelo algoritmo 2 tende a apresentar uma melhor descrição da área livre do espaço local. Isto pode ser verificado na figura 5. Nela, a grade foi construída a partir de informações de alcance dadas apenas pelos sonares laterais, frontal e traseiro do *Pioneer 3AT*. Pelo fato do movimento do robô ter sido realizado ao longo da maior dimensão do corredor sem efetuar rotações significativas, os sonares laterais permaneceram perpendiculares às paredes durante toda a coleta de dados. Assim, a grade foi montada com uma maior fidelidade ao ambiente. Sobreposto a esta grade está o retângulo desenhado a partir dos seguintes valores que foram calculados pelo método proposto:

$$\mathbf{r} = [11.18, 0.13, 1.0^\circ, 24.83, 1.64]^T$$

com x , y , l e c em metros. Para este retângulo, os erros relativos são 7.96% para a largura (maior dimensão do corredor) e 2.5% para o comprimento.

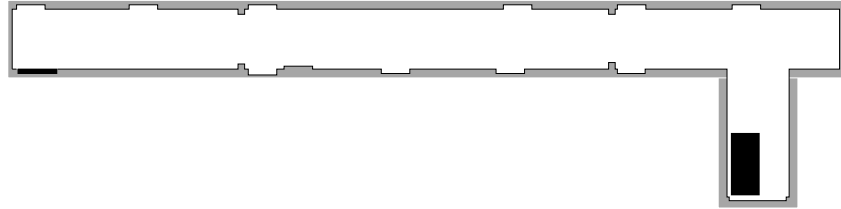


Figura 3: Modelo do corredor utilizado para a coleta de dados.

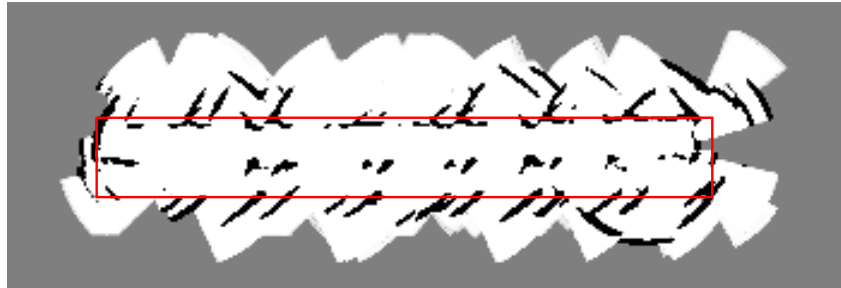


Figura 4: Retângulo encontrado para a representação em grade do corredor apresentado na figura 3.

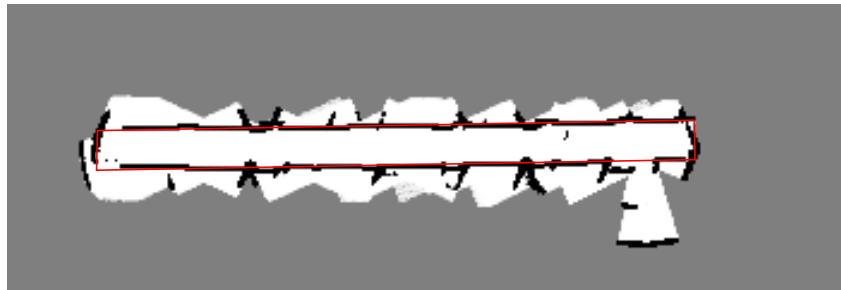


Figura 5: Retângulo para representação do corredor a partir de uma grade mais precisa.

5 Considerações

A primeira consideração a ser feita sobre o método de votação ponderada apresentado é relativo ao esforço computacional gasto para obter o retângulo. O problema não está necessariamente nos algoritmos 1 e 2, mas na grande quantidade de pontos da grade de ocupação que está sendo analisada. Quando este método de votação foi mostrado por Pedrosa et al. (2006), a utilização da nuvem de pontos e o método de refinamento permitiam que a determinação do retângulo pudesse ser realizada durante a fase de exploração local, pois havia poucos pontos coletados por causa do prévio conhecimento do formato geométrico do espaço local. Neste trabalho é necessário utilizar uma grade do ambiente local. Caso todos os pontos desta grade fossem analisados no procedimento de voto ponderado, o tempo gasto para encontrar a geometria de uma única sala seria excessivamente grande. Isto poderia inviabilizar a aplicação do método para mapeamento *on-line*. Porém, a idéia de aplicar um *threshold* para descartar células no voto reduziu o tempo gasto para calcular o retângulo, permitindo que este método seja aplicado para a construção incremental de todo o

mapa.

Outra consideração importante é que grades mais precisas implicam retângulos mais precisos, como mostrado pelas figuras 4 e 5. Mas também deve-se atentar à maneira como a grade é construída. Neste caso, três fatores devem ser analisados. O primeiro é sobre o tipo de sensor utilizado. Muitas das medidas do sonar são incertas, o que influencia na qualidade da grade. Sensores como *laser* podem retornar dados de alcance mais precisos, e conseqüentemente, permitir a construção de grades mais fieis ao ambiente. Como sonares foram utilizados para a aquisição de dados deste trabalho, o segundo fator a ser analisado é o modelo em cone do sonar, que é uma simplificação do comportamento acústico da onda sonora. Para futuras experiências pretende-se elaborar outros modelos que sejam mais precisos. Por fim, o terceiro fator é construir a grade com algoritmos que permitem uma maior fidelidade de representação, como proposto por Thrun (2003).

Futuros trabalhos estão sendo direcionados para uma maior generalização do método. Por exemplo, pretende-se utilizar esta votação ponderada não apenas para calcular retângulos, mas também outras fi-

guras geométricas que melhor se ajustem aos pontos ocupados da grade. Além disso, pretende-se expandir o método para a construção de todo o mapa do ambiente. Assim, é necessário definir heurísticas para determinar tamanhos para grade de forma que ela possa representar um único espaço local por vez, e com isso aplicar o algoritmo 2 corretamente.

Uma questão que também deverá ser analisada para a representação local proposta é a presença de obstáculos no ambiente. Dependendo da configuração dos móveis em uma cena real, uma única sala pode ser representada por duas, ou mais, figuras geométricas simples. Estudos estão sendo dirigidos para obter um melhor tratamento deste aspecto.

Por fim, pretende-se evoluir da tarefa de mapeamento simples para o mapeamento realizado simultaneamente com a localização do robô, obtendo uma representação métrica final composta por elementos geométricos simples. A adoção de grades de ocupação para a extração de características geométricas do espaço livre local permite a utilização de técnicas bem fundamentadas de SLAM, como o trabalho de Hänel et al. (2003).

Agradecimentos

O autores agradecem a Anderson Abner (mestrando do PPGEE-UFRN) por dispor seus conjuntos de dados para este trabalho. O autor Diogo Pedrosa agradece, em especial, à Capes pelo apoio financeiro.

Referências

Arleo, A., Millán, J. d. R. e Floreano, D. (1999). Efficient Learning of Variable-resolution Cognitive Maps for Autonomous Indoor Navigation, *IEEE Trans. on Robotics and Automation* **15**(6): 990–1000.

Ballard, D. H. (1981). Generalizing the Hough Transform to Detect Arbitrary Shapes, *Pattern Recognition* **13**(2): 111–122.

Borenstein, J. e Koren, Y. (1991). Histogramic in-Motion Mapping for Mobile Robot, *IEEE Trans. on Robotics and Automation* **7**(4).

Buschka, P. e Saffiotti, A. (2004). Some Notes on the Use of Hybrid Maps for Mobile Robots, *Proc. of the 8th Int. Conf. on Intelligent Autonomous Systems*, Amsterdã, NL, pp. 547–556.

Chen, G.-Y. e Tsai, W.-H. (1998). An Incremental Learning by Navigation Approach to Vision-based Autonomous Land Vehicle Guidance in Indoor Environments using Vertical-line Information and Multiweighted Generalized Hough Transform Technique, *IEEE Trans. on Systems, Man and Cybernetics, Part B: Cybernetics* **28**(5): 740–748.

Durrant-Whyte, H. e Bailey, T. (2006a). Simultaneous Localization and Mapping (SLAM): Parte I, *IEEE Robotics and Autonomous Magazine* **13**(2): 99–108.

Durrant-Whyte, H. e Bailey, T. (2006b). Simultaneous Localization and Mapping (SLAM): Parte II, *IEEE Robotics and Autonomous Magazine* **13**(3): 108–117.

Fabrizi, E. e Saffiotti, A. (2000). Extracting Topology-based Maps from Gridmaps, *Proc. of the IEEE ICRA*, San Francisco, CA, USA, pp. 2972–2978.

Fabrizi, E. e Saffiotti, A. (2002). Augmenting Topology-based Maps with Geometric Information, *Robotics and Autonomous Systems* **40**: 91–97.

Hänel, D. e Fox, D. e Burgard, W. e Thrun, S. (2003). A Highly Efficient FastSLAM Algorithm for Generating Cyclic Maps of Large-Scale Environments from Raw Laser Range Measurements, *Proc. of the Conf. on Intelligent Robots and Systems*, Las Vegas, USA.

Monteiro, S. T. e Ribeiro, C. H. C. (2002). Obtenção de Mapas Cognitivos para o Robô Móvel Magellan Pro, *Anais do XII CBA*, Natal, RN.

Moravec, H. e Elfes, A. (1985). High Resolution Maps from Wide Angle Sonar, *1985 IEEE Int. Conf. on Robotics and Automation*, Vol. 2, St Louis, EUA, pp. 116–121.

Pedrosa, D. P. F., Medeiros, A. A. D. e Alsina, P. J. (2006). Uma Proposta de SLAM com Determinação de Informações Geométricas do Ambiente, *Anais do XIV CBA*, Salvador, BA.

Pfister, S. T., Roumeliotis, S. I. e Burdick, J. W. (2003). Weighted Line Fitting Algorithms for Mobile Robot Map Building and Efficient Data Representation, *Proc. of the 2003 IEEE ICRA*, Taipei, Taiwan, pp. 1304–1311.

Thrun, S. (2003). Learning Occupancy Grid Maps with Forward Sensor Models, *Autonomous Robots* **15**: 111–127.

Thrun, S., Burgard, W. e Fox, D. (2005). *Probabilistic Robotics*, MIT Press, Cambridge, MA, USA.

Vasudevan, S., Gächter, S., Nguyen, V. e Siegwart, R. (2007). Cognitive Maps for Mobile Robots – An Object Based Approach, *Robotics and Autonomous Systems* **55**(5): 359–371.

Wetherbie III, J. O. e Christopher, E. S. (2001). Large-scale Feature Identification for Indoor Topological Mapping, *2001 IEEE Int. Conf. on Systems, Man, and Cybernetics*, Vol. 5, pp. 2852–2857.