

Localização da direção principal de Corredores por Ponto de Fuga

Dimas Abreu Dutra
dimas@dcc.ufmg.br

Gustavo de Oliveira Corrêa
gustavoc@dcc.ufmg.br

Rogério de Faria Silva
rogeriofaria_ufg@hotmail.com

Resumo

A localização da direção de corredores é uma informação importante para a navegação de robôs em ambientes internos. Esse trabalho propõe um método para identificar a direção de corredores utilizando pontos de fuga. Para localização do ponto de fuga propomos também um método baseado no algoritmo RANSAC.

1. Introdução

Uma das principais funções de um robô é a percepção do meio no qual ele se encontra inserido. Só após esse passo alguma decisão pode ser tomada e executada pelo robô. Esse trabalho propõe um método de percepção do ambiente por visão computacional para fornecer uma referência de navegação para o robô em um corredor.

Corredores possuem, em geral, uma grande quantidade de retas paralelas ao seu eixo principal, que é a direção ao longo da qual um robô deve se deslocar, conforme mostrado na fig. 1. Em uma câmera que segue o modelo da projeção perspectiva, essas retas são projetadas em retas que se encontram em um único ponto da imagem, o ponto de fuga.

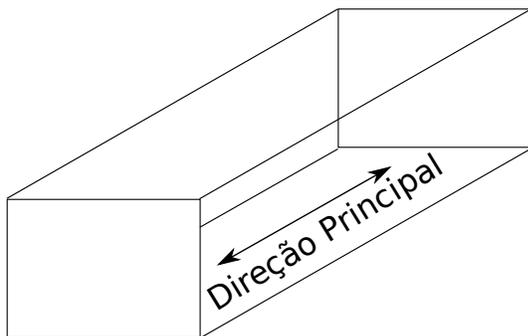


Figura 1. A direção principal de um corredor.

A posição x do ponto de fuga na imagem em uma câmera montada em um robô com uma configuração semelhante à mostrada na fig. 2 pode ser utilizada como referência



Figura 2. Configuração do robô e câmera utilizada

de ângulo de guinada para um sistema de controle para a navegação do robô. O ponto de fuga deve ser mantido no centro da imagem para que o robô siga em direção a ele.

Nesse trabalho é mostrado o método de localização de pontos de fuga baseado no algoritmo RANSAC que desenvolvemos, os passos realizados para localizar o ponto de fuga e os resultados que conseguimos com as imagens coletadas com o robô e câmera da fig. 2 nos corredores do prédio da nossa universidade em que se encontra nosso laboratório.

2. Trabalhos Relacionados

Vários trabalhos na literatura [4, 3, 5] utilizam o ponto de fuga formado pela interseção de retas paralelas ao eixo principal de um corredor como indicador da sua direção. De maneira semelhante, em [2] os pontos formados pela interseção das retas da parede esquerda e frontal, de um lado, e direita e frontal, do outro, fornecem a direção do corredor. Esses trabalhos extraem a informação do centro do corredor para

navegação robótica e diferem entre si principalmente pela maneira como identificam as retas.

Em [4] as retas são identificadas com o filtro Sobel e o algoritmo RANSAC. A orientação do gradiente é utilizada para selecionar as retas da parede esquerda e direita e sua interseção é o ponto de fuga. Em [5] é proposto um sistema para detecção da direção dominante para uma classe mais abrangente de caminhos que inclui corredores. Esse trabalho não utiliza apenas uma imagem para detectar a direção dominante, mas também a informação temporal da localização nos instantes anteriores e a odometria do robô. Nesse sistema, a câmera deve estar com seu eixo óptico a um ângulo conhecido do plano horizontal. Em [3] um sistema de geração e verificação de hipóteses é utilizado para localizar corredores e portas na imagem. O algoritmo *Fast Line Finder* (FLF) é utilizado para detectar gradientes de intensidade correspondente às bordas. Esses gradientes são separados por sua orientação como diagonais positivos, diagonais negativos e horizontais. A geração de hipóteses é feita aplicando algoritmo RANSAC para encontrar retas nos gradientes diagonais. As hipóteses são verificadas vendo se as linhas verticais (formadas pelos gradientes horizontais) terminam nas retas detectadas. Isso se baseia no fato de que em um corredor muitas linhas verticais terminam nas linhas da parede, como as linhas formadas por portas e quinas de paredes. Esse método tem a vantagem de ser mais robusto em relação a linhas retas espúrias e calcular o ponto de fuga com duas ou mais retas. Ele assume que a direção vertical é conhecida e não funciona para corredores com poucas linhas verticais terminando nas retas entre parede e chão.

Já [2] utiliza técnicas de mineração de dados através do algoritmo *k-means* para agrupar os elementos da imagem. O elemento que se quer agrupar são as paredes da imagem, que podem variar de uma a três (parede esquerda, frontal e direita). O método assume também que o corredor tem um rodapé negro, enquanto o restante da imagem seria mais clara. O método proposto por [6] consiste em gerar a imagem binária e extrair linhas usando o algoritmo de Hilditch para, em seguida, obter a borda correta da linha usando a transformada de Hough.

3. Metodologia

Para detectar o ponto de fuga de um corredor utilizamos os seguintes passos: detecção de bordas na imagem, detecção de retas a partir dos pontos de borda, e detecção de pontos de fuga a partir das retas. Nessa seção explicaremos cada passo em mais detalhe.

3.1. Detecção das bordas

Entre as retas paralelas ao eixo principal de um corredor, as mais notáveis e que delimitam a área que pode ser ex-

plorada pelo robô são as formadas na interface entre chão e parede, que chamaremos de retas da parede. Essas retas geram bordas na imagem que podem se formar devido à mudança de orientação da superfície, presença de um rodapé ou diferença de cor entre piso e paredes, entre outros motivos, e estão presentes na grande maioria dos corredores. Para detectar as bordas foi utilizado o algoritmo de Canny. Ele é um operador gaussiano de primeira derivada que suaviza os ruídos e localiza as bordas. A partir do resultado do detector de bordas pode-se detectar as linhas da imagem.

3.2. Detecção das Retas

Para detecção de retas foi utilizada a Transformada de Hough (HT). A HT é um método para determinação dos parâmetros de uma figura parametrizável que melhor representam um conjunto de pontos. Essa transformada se baseia na transformação dos pontos do espaço de dados para o espaço de parâmetros.

Para a detecção de retas através da transformada de Hough, foi utilizada a parametrização polar da reta, descrita na eq. 1:

$$x \cos(\theta) + y \sin(\theta) = \rho \quad (1)$$

onde θ é o ângulo que a normal da reta faz com o eixo x e ρ é a distância entre a reta e a origem. Nessa parametrização, um ponto no espaço de parâmetros $r = (\rho_p, \theta_p)$ determina um conjunto de pontos no espaço da imagem que formam a reta r . De maneira análoga, um ponto no espaço de imagem $P = (x_p, y_p)$ determina um conjunto de pontos no espaço de parâmetros que forma uma senóide no plano (ρ, θ) e representa todas as retas da imagem que podem passar por P .

Se vários pontos no espaço da imagem pertencem a uma reta, as diversas senóides que eles definem no espaço de parâmetros se interceptam num único ponto do plano (ρ, θ) que corresponde à reta comum entre eles. Esse fato é utilizado para a detecção de retas através da transformada de Hough. O espaço de parâmetros é discretizado em um intervalo que abrange todas as retas que podem existir na imagem, isto é:

$$0 \leq \rho \leq D \quad (2a)$$

$$-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}, \quad (2b)$$

onde D é a diagonal da imagem, e para cada célula desse espaço discretizado é então atribuído um contador $A(\rho, \theta)$. É então aplicada, para cada ponto do espaço da imagem, a transformação para o espaço de parâmetros e o contador A é incrementado em cada célula (ρ, θ) que as senóides correspondentes aos pontos na imagem passam. Como pontos que correspondem a retas geram senóides que se interceptam no ponto correspondente à reta comum, os picos no acumulador A correspondem às retas que melhor explicam os

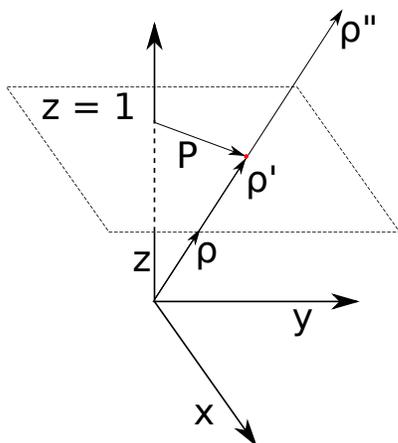


Figura 3. Representação gráfica das coordenadas homogêneas de um ponto.

pontos. Essas retas são entrada para o detector de pontos de fuga. A resolução dos parâmetros depende do número de células e o tempo computacional é linear e proporcional a este número.

3.3. Determinação do ponto de fuga

3.3.1. Representações Homogêneas Para determinação do ponto de fuga foram utilizadas coordenadas homogêneas. As coordenadas homogêneas se baseiam na representação de pontos P de uma imagem 2D por pontos ρ no espaço 3D que pela projeção perspectiva seriam projetados no plano $z = 1$ em P . Nessa representação, infinitos pontos ρ correspondem a um mesmo ponto P , conforme mostrado na fig. 3. Um ponto $P = (x_P, y_P)$ corresponde a $\rho = (wx_P, wy_P, w), \forall w \neq 0$.

De maneira semelhante, uma reta no plano 2D $z = 1$ pode ser representada no espaço como um vetor 3D em coordenadas homogêneas. Nesse caso a equação paramétrica da reta

$$Ax + By + C = 0 \quad (3)$$

se transforma em

$$Ax + By + Cz = 0 \quad (4)$$

sendo o ponto (x, y, z) uma representação em coordenadas homogêneas de um ponto 2D que pertence à reta. A eq. 4 também representa um plano no espaço que passa pela origem com o vetor $\vec{N} = [A, B, C]$ como normal. Como no caso do ponto, infinitos vetores normais representam o mesmo plano no espaço e, conseqüentemente, a mesma reta 2D. A interseção do plano normal a \vec{N} com o plano $z = 1$ é a reta 2D representada pela eq. 3.

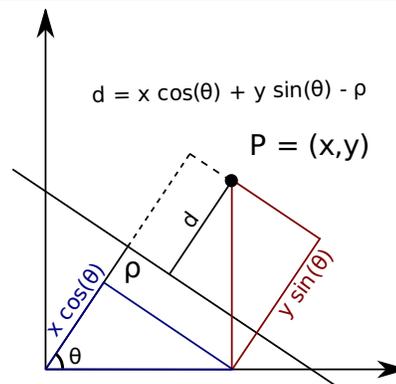


Figura 4. Representação gráfica da distância entre a reta e um ponto.

A representação homogênea de pontos e retas é conveniente para a determinação de pontos de fuga pois a interseção entre duas retas é o ponto homogêneo gerado pelo produto vetorial entre suas normais. Isso pode ser visualizado no espaço como o ponto homogêneo correspondente ao encontro das duas retas sendo o vetor diretor da reta formada pela interseção entre os dois planos correspondentes às retas homogêneas. Se as retas forem paralelas o resultado do produto vetorial entre suas normais será da forma $[a, b, 0]$. Esse resultado corresponde a representação homogênea de um ponto no infinito, que é a interseção de duas retas paralelas.

Comparando a eq. 1 com a eq. 3 é evidente a transformação entre a parametrização polar da reta e a homogênea:

$$\vec{N} = [\cos \theta, \sin \theta, -\rho]. \quad (5)$$

A representação da eq. 5 também é útil pois a distância 2D entre uma reta (ρ, θ) e um ponto é dada por

$$d = x \cos \theta + y \sin \theta - \rho, \quad (6)$$

conforme mostrado graficamente na fig. 4.

3.3.2. Algoritmo de Determinação O ponto de fuga das retas identificadas pela transformada de Hough é encontrado utilizando o algoritmo RANSAC (*Random Sample Consensus*) [1]. Ele é utilizado para encontrar o ponto na imagem que melhor representa a interseção das retas, apesar dos ruídos e incertezas. Seu funcionamento é o seguinte:

1. Transformar as retas retornadas pela Transformada de Hough para a representação da eq. 5.
2. Entre as retas, escolher duas aleatoriamente que não tenham sido escolhidas antes.
3. Encontrar a interseção delas através do produto vetorial.

4. Calcular a distância entre todas as outras retas e a interseção através da eq. 6.
5. Os *inliers* são as retas que estão a uma distância menor que um limiar da interseção.
6. Se o número de *inliers* for grande o suficiente, recalculer o ponto de fuga através da decomposição em valores singulares.
7. Voltar ao passo 2 e repetir N vezes.
8. O melhor ponto de fuga é aquele com maior número de *inliers*.

A decomposição em valores singulares para recalculer o ponto de fuga é feita montando-se uma matriz em que cada reta homogênea representa uma linha. O melhor ponto de fuga corresponde ao vetor \vec{v} correspondente ao menor valor singular da matriz. Por utilizar o RANSAC esse método funciona bem mesmo com uma grande quantidade de retas espúrias.

4. Resultados e discussões

Exemplos das imagens coletadas para validar nosso método são apresentados na fig. 5. Os resultados obtidos da aplicação do método proposto nessas imagens são apresentados na fig. 6, onde são apresentadas as imagens retornadas pelo detector de bordas com as retas detectadas pela Transformada de Hough sobrepostas em azul e o ponto de fuga como um asterisco em vermelho.

Nessas imagens é possível ver que todas retas paralelas à direção principal do corredor, não apenas as retas da parede, são utilizadas para localizar e identificar a direção principal do corredor. É possível ver também que, devido à utilização do RANSAC, as retas verticais, horizontais e mesmo diagonais espúrias que não passam pelo ponto de fuga não polarizam o resultado.

5. Conclusões

Esse artigo introduz um método para navegação em corredores através do ponto de fuga formado pelas retas do corredor. Os pontos de fuga são obtidos através da intercessão das retas no espaço de coordenadas homogêneas, retas esses detectadas pela Transformada de Hough, recebendo como entrada as bordas obtidas pelo algoritmo de Canny. O ponto de fuga das retas do corredor é então estimado usando RANSAC. Os resultados mostram a eficiência do método proposto. Embora nenhuma métrica para avaliação dos resultados tenha sido usada, o ponto de fuga estimado das retas do corredor são muito próximos do real. Entretanto a implementação do mesmo exigiu alguns parâmetros obtidos empiricamente. O método também não trata casos em

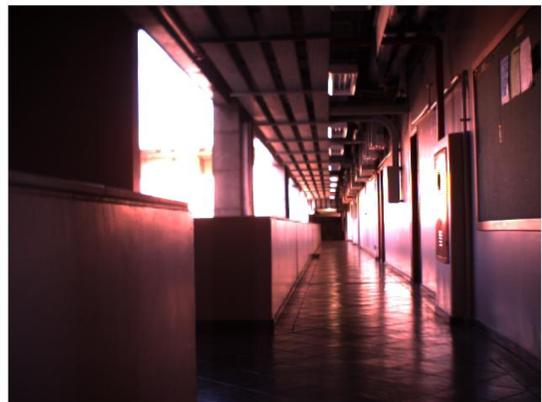


Figura 5. Imagens utilizadas para validar nosso método.

que não há um corredor da imagem. Nesse caso algumas soluções foram pensadas como a comparação com o ponto de fuga obtido anteriormente, dada uma seqüência de imagens, entretanto não foi implementado.

Referências

- [1] M. Fischler and R. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [2] H.-S. Lho, J.-W. Kwon, D. Chwa, and S.-K. Hong. Vision-based corridor line detection using k-means algorithm. In *International Conference on Control, Automation and Systems, 2007*, pages 1052–1056, 2007.
- [3] W. Shi and J. Samarabandu. Investigating the performance of corridor and door detection algorithms in different environments. In *International Conference on Information and Automation, 2006*, pages 206–211, 2006.
- [4] R. Vassallo, H. Schneebeli, and J. Santos-Victor. A purposive strategy for visual-based navigation of a mobile robot. In *Proceedings of Midwest Symposium on Circuits and Systems*, pages 334–337, 1998.
- [5] A. Zhang and R. Russell. Dominant orientation tracking for path following. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3885–3889, 2005.
- [6] A.-M. Zou, Z.-G. Hou, M. Tan, and D. Liu. Vision-Guided Mobile Robot Navigation. In *Proceedings of Int. Mediterranean Modeling Multiconference*, 2006.

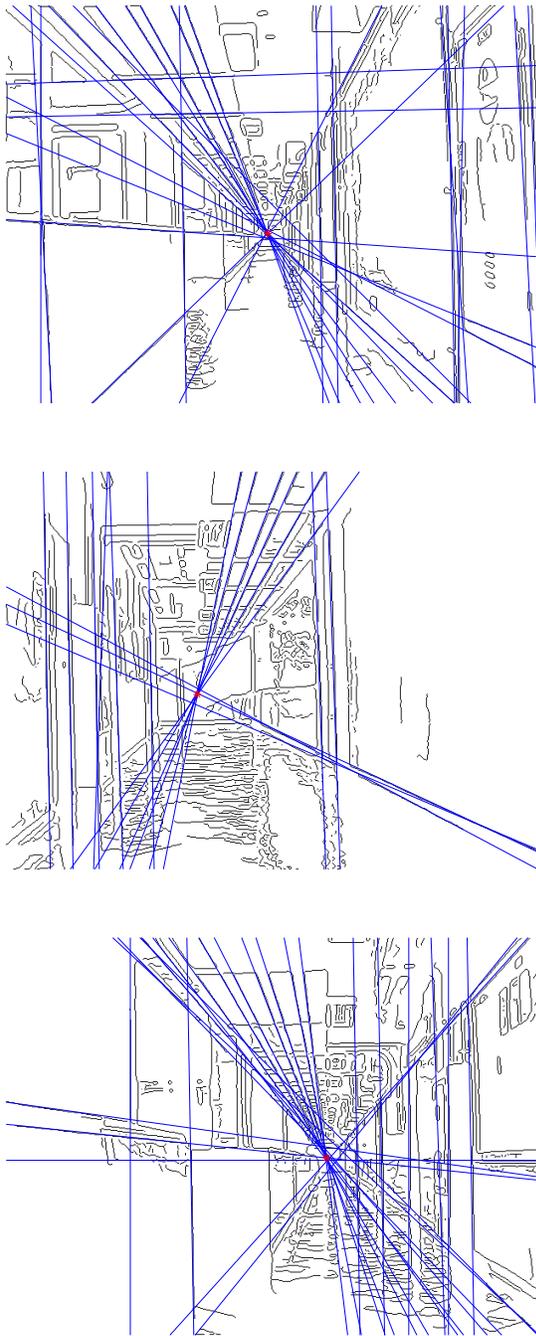


Figura 6. Resultados da aplicação do método proposto.