

CÁLCULO DO FLUXO ÓPTICO EM TEMPO REAL E SUA UTILIZAÇÃO NA NAVEGAÇÃO DE ROBÔS MÓVEIS

MÁRIO SARCINELLI FILHO, HANSJORG ANDREAS SCHNEEBELI, ELIETE M. DE O. CALDEIRA

Universidade Federal do Espírito Santo
Av. Fernando Ferrari, s/n, 29060-900, Vitória, ES, BRASIL
E-mail: mario.sarcinelli@ele.ufes.br

Resumo— Este trabalho trata do uso de fluxo óptico na navegação de robôs móveis. A aplicação consiste em calcular o fluxo óptico a bordo do robô e utilizá-lo para controlar a navegação do mesmo. Este cálculo deve ser feito em tempo real, o que torna a quantidade de cálculo necessária e o número de quadros de imagem armazenados fatores importantíssimos a serem considerados, além da confiabilidade do fluxo calculado, por si só fundamental. Tendo em vista esses aspectos, alguns algoritmos foram selecionados da literatura. Os resultados da comparação entre eles, obtida a partir de dois experimentos, são reportados para os dois algoritmos com melhor desempenho e para uma versão modificada de um deles que é aqui proposta. Uma medida de confiabilidade disponível na literatura é também implementada para excluir, do fluxo resultante, vetores estimados com variância muito grande, restando assim apenas a informação de fluxo mais confiável. Um processo de segmentação baseada no fluxo é então usado, gerando uma imagem subdividida em vários "pseudo-objetos". Baseando-se no tempo para choque calculado para os diversos "pseudo-objetos" resultantes da segmentação, uma estratégia para desviar o robô dos objetos mais próximos está em discussão.

Abstract— This work addresses the use of optical flow in the navigation of a mobile robot. The application in mind consists on calculating the optical flow onboard the robot and using these data to control its navigation. This calculation should be performed in real time, thus making the amount of calculation and the number of image frames to be stored very important features, as well as the reliability of the estimated optical flow vectors. Regarding these three aspects, some algorithms available in the literature are analyzed, through the use of two simple experiments here described, whose results are discussed in connection with two good candidates to be the algorithm to calculate the optical flow. Besides these two algorithms, a third one here proposed is also analyzed. It merges the good features of the two algorithms previously mentioned and is a very fast one, thus being selected for the application in mind. In the sequence of the work, a measurement of the optical flow reliability yet available in the literature is applied to the optical flow vectors yet available, in order to eliminate those vectors whose variance is above a certain threshold. An image segmentation process based on the remaining optical flow vectors is then implemented, which results in cataloging the different objects in the image frame considered. For each object, it is calculated a time to collision, which can be used to control the mobile robot heading.

Keywords— Optical Flow; Mobile Robots; Computer Vision; Moving Detection; Segmentation; Vision-based Obstacle Avoidance.

1 Introdução

A literatura de visão computacional mostra que é possível determinar o movimento relativo entre um objeto em uma cena e a câmara usada para capturar uma seqüência de quadros de imagem 2D da mesma, através do cálculo do fluxo óptico correspondente (Horn e Schunck, 1981; Barron et al., 1993; Fleet e Jepson, 1990; Lai e Vemuri, 1998). Tal fluxo óptico pode também ser usado na navegação de robôs móveis (Dev et al., 1997a). Entretanto, nos experimentos relatados até o momento o cálculo do fluxo óptico é executado fora do robô, o que demanda transmissão/recepção de imagens e sinais de controle.

Este trabalho, porém, tem como objetivo calcular o fluxo óptico a bordo do robô para usá-lo no controle da sua navegação, o que melhorará sensivelmente a autonomia do robô, dado que ele não terá mais a restrição de distância em relação a um computador externo de controle.

Um algoritmo adequado para o cálculo do fluxo a bordo do robô é então pesquisado. Ele deve exigir apenas o armazenamento de dois quadros de imagem (número mínimo), assim como exigir o menor esforço computacional possível e gerar vetores de fluxo óptico que sejam confiáveis. Na busca de um algoritmo com essas características foi feita uma análise

comparativa entre diversos algoritmos da literatura de fluxo óptico. Esta análise comparativa (Sarcinelli-Filho et al., 2001) é baseada em dois experimentos relativamente simples: no primeiro o robô se move paralelamente a uma parede, enquanto no segundo ele é programado para aproximar-se da mesma. Nos dois experimentos a câmara fixa sobre o robô está voltada para a parede e adquirindo imagens da mesma a 30 quadros por segundo. No primeiro caso é verificado que o fluxo óptico calculado é horizontal e de módulo constante, como esperado. No segundo, o tempo para choque é calculado e usado para comparar os algoritmos. Foi usado nos experimentos o robô PIONEER 2DX da ActivMedia, que tem sobre ele uma câmara CCD colorida com 48,8° de abertura máxima, fornecendo imagens em padrão *bitmap* de 640x480 pixels.

A variância calculada a partir de variáveis da imagem é usada como medida de confiabilidade (Sarcinelli-Filho et al., 2001) dos vetores de fluxo óptico gerados, e aqueles menos confiáveis são descartados, gerando resultados mais coerentes, ainda que tornando o fluxo resultante um pouco mais esparsos.

Um processo de segmentação baseado no fluxo óptico assim calculado é também implementado, resultando no aparecimento de "pseudo-objetos" para

os quais pode ser calculado o tempo para choque (Dev et al., 1997a). Uma estratégia para desviar o robô dos obstáculos mais próximos pode então ser implementada, baseada nesta informação, o que é objeto deste trabalho.

2 Cálculo do Fluxo Óptico

O fluxo óptico é definido como a distribuição de velocidades aparentes do movimento do padrão de brilho em uma imagem, e aparece devido ao movimento relativo entre os objetos e a câmara (Horn e Schunck, 1981), considerando uma seqüência de quadros de imagem ao longo do tempo. O fluxo é dado por um conjunto de vetores (u, v) , sendo u a velocidade na direção x e v a velocidade na direção y , no instante t correspondente a um dado quadro de imagem.

Barron et al. (1993) fizeram um estudo comparativo entre diversos algoritmos de cálculo de fluxo óptico. Eles concluíram que os algoritmos de Horn e Schunck (1981), uma versão modificada dele (Barron et al., 1993) e o algoritmo de Fleet e Jepson (1990) apresentaram os melhores resultados, seguidos pelo algoritmo de Lucas e Kanade (1981), que também apresentou bons resultados.

O algoritmo de Fleet e Jepson não é adequado para a aplicação em navegação de robôs, por usar muitos quadros de imagem simultaneamente (dezenas em alguns exemplos (Lai e Vemuri, 1998)) e exigir grande esforço computacional. Da mesma maneira, o algoritmo de Horn e Schunck modificado (Barron et al., 1993) também é inadequado, pois utiliza cinco quadros de imagem no cálculo da derivada no tempo, exigindo muita memória e esforço computacional.

Outros algoritmos de cálculo do fluxo óptico foram propostos na literatura mais recente, os quais são mais robustos ou computacionalmente mais eficientes. Entre eles foram selecionados os algoritmos propostos por Grossmann e Santos-Victor (1997), por Nesi et al. (1995) e por Lai e Vemuri (1998). Estes algoritmos foram implementados e comparados com o algoritmo de Lucas e Kanade (1981) e o algoritmo clássico de Horn e Schunck (1981).

A comparação entre eles, usando os dois experimentos básicos referidos na introdução, levou à conclusão de que, apesar de alguns destes algoritmos apresentarem resultados extremamente confiáveis, como é o caso da versão clássica do algoritmo de Horn e Schunck (1981), do algoritmo de Lai e Vemuri (1998) e do algoritmo de Nesi et al. (1995), o tempo para a computação dos resultados é excessivo para uma aplicação em tempo real. Os algoritmos que se mostraram mais eficientes, sob o aspecto do reduzido tempo de computação, foram o algoritmo de Lucas e Kanade (1981) e o de Grossmann e Santos-Victor (1997).

Assim, na seqüência deste trabalho apenas estes dois algoritmos vão ser considerados: é apresentada

uma análise simplificada dos seus fundamentos teóricos e uma discussão rápida sobre sua implementação. Para tanto, as imagens vão ser caracterizadas pelo padrão de brilho $I(x,y,t)$, onde x e y são as coordenadas de um pixel em um quadro de imagem, e t designa um quadro de imagem específico em uma seqüência de quadros.

2.1 Algoritmo de Mínimos Quadrados (Lucas e Kanade, 1981)

A suposição básica para o cálculo do fluxo óptico é a de que o padrão de brilho da imagem é quase constante (Horn e Schunck, 1981), o que significa que

$$\frac{dI(x,y,t)}{dt} = 0 \quad \text{ou} \quad I_x u + I_y v + I_t = 0 \quad (1)$$

onde I_x , I_y e I_t são as derivadas da imagem nas direções x , y e t , e u e v são as componentes do fluxo óptico no instante t , para o pixel (x,y) sob consideração. Entretanto, a equação (1), chamada restrição do fluxo óptico, não é o bastante para determinar u e v em cada pixel. Isto caracteriza um problema mal condicionado, e alguma restrição adicional deve ser introduzida (Horn e Schunck, 1981; Grossmann e Santos-Victor, 1997).

O método proposto por Lucas e Kanade (1981) considera que o fluxo óptico é constante em uma região consistindo de um grupo de $N_x N_y$ pixels. Então, escrevendo a restrição de fluxo óptico para cada pixel na região, obtém-se um sistema de equações sobre-determinado, onde cada equação é dada por

$$I_{x_i} u + I_{y_i} v + I_{t_i} = 0 \quad \text{para} \quad i = 1, \dots, N_x N_y \quad (2)$$

Estas equações são usadas para gerar a estimativa de Mínimos Quadrados do vetor de fluxo óptico $(\hat{u} \quad \hat{v})$ nesta região, a qual é dada por

$$(\hat{u} \quad \hat{v}) = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (3)$$

onde a matriz \mathbf{A} é dada por

$$\mathbf{A} = \begin{bmatrix} I_{x_1} & \dots & I_{x_{N_x N_y}} \\ I_{y_1} & \dots & I_{y_{N_x N_y}} \end{bmatrix}^T \quad (4)$$

e o vetor \mathbf{b} é dado por

$$\mathbf{b} = -[I_{t_1}, \dots, I_{t_{N_x N_y}}]^T \quad (5)$$

Este método se mostra muito atrativo quando se considera o custo computacional necessário para obter a estimativa do fluxo óptico, por não ser um método iterativo, assim como pela simplicidade do método em si, que envolve apenas a inversão de uma matriz 2×2 para qualquer tamanho de região. Desta maneira, este método se torna mais atrativo quando o tamanho da região é aumentado. Porém, há um compromisso entre o tamanho da região e a precisão da estimativa, dada a suposição de fluxo constante na região. Por outro lado, as derivadas devem ser calculadas em todos os pixels da imagem, independentemente do tamanho da região. Sendo assim, mesmo aumentando o valor de N não é possível uma grande redução no tempo de cálculo.

2.2 Algoritmo de Grossmann e Santos-Victor (1997)

O método proposto por Grossmann e Santos-Victor se baseia na consideração de que o fluxo óptico em determinada região pode ser aproximado por um modelo afim, que é um modelo linear, ou seja, o fluxo tem as derivadas em relação a x e a y constantes, ou seja

$$u(x, y) = u_0 + (x - x_0)u_x + (y - y_0)u_y \quad (6)$$

$$v(x, y) = v_0 + (x - x_0)v_x + (y - y_0)v_y \quad (7)$$

Esta é uma aproximação, uma vez que, em geral, não há vetor afim que satisfaça a restrição de brilho constante expressa pela equação (1). Em seu artigo, tais autores consideram a restrição do fluxo óptico como sendo

$$I(x + u, y + v, t + I) = I(x, y, t) \quad \forall x, y, t \quad (8)$$

que é equivalente à equação (1).

Os autores propõem dois algoritmos para estimar os parâmetros do fluxo óptico afim, denominados estimadores anisotrópicos, os quais são baseados em um esquema de diferenciação direcional das imagens ao longo da direção da estimativa corrente do fluxo. Para determinar a equação de observação, eles consideram uma versão mais realista da equação (1), em que se considera um termo v correspondente ao ruído aleatório, ou seja,

$$-I_t = I_x u + I_y v + v \quad (9)$$

Aproximando linearmente o lado esquerdo da equação (8) obtém-se

$$I(x + u, y + v, t + I) = I(x + \tilde{u}, y + \tilde{v}, t + I) + \begin{bmatrix} I_x & I_y \end{bmatrix} \begin{bmatrix} u - \tilde{u} & v - \tilde{v} \end{bmatrix}^T + h(u - \tilde{u}, v - \tilde{v}) \quad (10)$$

onde (\tilde{u}, \tilde{v}) é um par arbitrariamente escolhido e h é o erro da expansão de primeira ordem. Juntando as equações (9) e (10) resulta em

$$I(x, y, t) - I(x + \tilde{u}, y + \tilde{v}, t + I) + I_x \tilde{u} + I_y \tilde{v} + h(u - \tilde{u}, v - \tilde{v}) + v = I_x u + I_y v \quad (11)$$

Uma vez que se tem as imagens nos instantes sucessivos t e $t + I$, e que as derivadas I_x e I_y podem ser calculadas, os únicos termos desconhecidos desta equação são h e v , que são a parte aleatória do modelo de observação. Em adição, os autores mostram que a variância de tais termos diminui quando se usa uma estimativa (\hat{u}, \hat{v}) de (u, v) para substituir o valor de (\tilde{u}, \tilde{v}) . Esta estimativa pode ser obtida pelo fluxo determinado para as imagens prévias, ou de maneira iterativa, usando regressão robusta.

Considerando que o fluxo e suas derivadas em relação às coordenadas da imagem são agrupados em um vetor de parâmetros definido por

$$\Theta = \begin{bmatrix} u_0 & u_x & u_y & v_0 & v_x & v_y \end{bmatrix}^T \in \mathfrak{R}^6 \quad (12)$$

a equação de observação torna-se

$$-I(x + \hat{u}, y + \hat{v}, t + I) + I(x, y, t) + \begin{bmatrix} I_x & I_y \end{bmatrix} \cdot \begin{bmatrix} \hat{u} & \hat{v} \end{bmatrix}^T = \begin{bmatrix} I_x & xI_x & yI_x & I_y & xI_y & yI_y \end{bmatrix} \Theta \quad (13)$$

Então, uma vez que a estimativa (\hat{u}, \hat{v}) seja conhecida, Θ pode ser calculado para a região desejada.

Para cobrir toda a imagem basta selecionar tantas regiões quanto necessário, escolhendo um tamanho apropriado para as regiões. Um número mínimo de seis pixels deve ser considerado na região desejada, a fim de estimar o modelo afim do fluxo naquela região. Aqui trabalha-se com um sistema sobre-determinado, usando-se 50 observações para cada região, cuja dimensão é 20x20 pixels. Isto significa que as derivadas da imagem devem ser calculadas apenas em alguns pixels por região, o que reduz sensivelmente o custo computacional. Outro fato importante é que uma estimativa de mínimos quadrados é usada como valor inicial a ser refinado por um algoritmo de regressão robusta. Pelo método Anisotrópico, as equações de observação são atualizadas antes das iterações do método de regressão robusta e no Método Anisotrópico-M elas são atualizadas a cada iteração. Um problema neste método, porém, é a necessidade de se inverter uma matriz 6x6 pelo menos uma vez (no caso do algoritmo Anisotrópico-M é necessário inverter uma matriz 6x6 a cada iteração do método de regressão robusta).

2.3 Algoritmo de Mínimos Quadrados Modificado

Considerando os métodos propostos por Lucas e Kanade (1981) e Grossmann e Santos-Victor (1997), descritos anteriormente, um outro método é aqui proposto, para o qual se considera a aproximação de fluxo constante em uma região de $N_x N_y$ pixels como no primeiro, mas no qual se toma um número de observações em cada região menor do que o número total de pixels na mesma, como é feito no segundo método. Isto faz com que as derivadas espaciais e temporal da imagem sejam calculadas apenas em alguns pixels, reduzindo sensivelmente o tempo de cálculo da estimativa dos vetores de fluxo.

2.4 Resultados Comparativos

Para comparar os três métodos acima descritos foram realizados dois experimentos usando o robô PIONEER 2DX. No primeiro o robô se move a 30mm/s paralelamente a uma parede, enquanto a câmara adquire uma seqüência de imagens da mesma a 30 quadros por segundo. Neste caso, espera-se que o fluxo óptico resultante tenha valores diferentes de zero apenas na direção horizontal. Mais precisamente, se a parede está à esquerda do robô, a direção dos vetores de fluxo óptico deve ser de 180 graus.

Os resultados deste experimento para os três métodos sob análise podem ser vistos na Tabela 1, onde em todos os algoritmos se considera uma região de 20x20 pixels, e as derivadas I_x , I_y e I_t são calculadas por diferenças finitas como sugerido por Horn e Schunck (1981). Os algoritmos de Grossmann e Santos-Victor (GSV) e o de Mínimos Quadrados Modificado (MQModificado) consideram 50 observações por região, tomadas aleatoriamente. A tabela apresenta o valor médio \bar{u} dos vetores de fluxo (em

Tabela 1 - Resultados Numéricos para o Primeiro Experimento

Algoritmo	$\bar{\theta}$	ε_{θ}	\bar{u}	σ_u
MQ	178.56	6.15	1.30	0.48
GSV 50	179.99	12.99	1.03	0.54
MQModificado 50	182.54	10.42	1.26	0.52

pixels/segundo), o desvio padrão σ_u associado a ele (em pixels/segundo), o valor médio $\bar{\theta}$ do ângulo dos vetores de fluxo óptico (em graus) e o erro médio ε_{θ} deste ângulo (em graus).

Como pode ser visto a partir da Tabela 1, o algoritmo de Mínimos Quadrados apresenta menor erro médio do ângulo e menor desvio padrão no módulo, seguido pelo algoritmo de Mínimos Quadrados Modificado.

No segundo experimento o robô se move com velocidade constante de 60mm/s perpendicularmente à parede, enquanto a câmara adquire uma seqüência de imagens da mesma. O fluxo óptico é calculado quando o robô está a 30 cm da parede, e assim o tempo para choque esperado é de 5 segundos.

Usando o fluxo óptico gerado por cada um dos três algoritmos sob análise, calculou-se o tempo para choque para cada região 20x20 pixels do quadro de imagem, cuja mediana está representada na Tabela 2.

Uma vez mais, as duas implementações do algoritmo de Mínimos Quadrados apresentam os resultados que mais se aproximam do esperado. Além disso, eles subestimam o tempo para choque, o que é melhor em termos de segurança do robô, já que ele vai antecipar, e não atrasar, qualquer manobra de evasão.

A fim de concluir a análise dos algoritmos considerados, a Tabela 3 apresenta o valor relativo do tempo gasto para executar estes algoritmos, tomando o algoritmo de Mínimos Quadrados como referência. A partir de tal tabela, percebe-se que o algoritmo de Mínimos Quadrados Modificado é mais adequado para a aplicação em mente. Ele explora a vantagem de não precisar calcular as derivadas do brilho da imagem em todos os pixels e de inverter apenas uma matriz 2x2, o que o torna muito mais rápido. Por outro lado, a precisão dos vetores de fluxo óptico obtidos usando este algoritmo é um pouco reduzida, mas ainda assim os resultados são satisfatórios quando se pensa em navegação de robôs móveis, onde

Tabela 2 - Resultados Numéricos para o Segundo Experimento.

Algoritmo	τ (s)
MQ	4.93
GSV 50	5.92
MQModificado 50	4.90

Tabela 3 - Tempo de Execução

Algoritmo	Tempo (%)
MQ	100
GSV 50	105
MQModificado 50	6

uma grande precisão não é necessária.

3 Medida de Confiabilidade do Fluxo Óptico

Dev et al. (1997b) sugerem uma medida de confiabilidade, baseada em uma análise das principais fontes de erro da estimativa do fluxo. Tal medida pode ser usada para descartar estimativas muito grosseiras do fluxo óptico, minimizando os efeitos dos vetores de fluxo com pouca confiabilidade em futuras etapas de processamento. De acordo com eles, uma estimativa da variância máxima no cálculo do fluxo óptico pelo método de mínimos quadrados é dada por

$$\sigma_{max}^2 = \frac{e^T e}{M \lambda_{min}(A^T A)} \quad (14)$$

onde $e_i = I_x \hat{u} + I_y \hat{v} + I_t$ é o erro resultante da aproximação do fluxo óptico na região pelos valores \hat{u} e \hat{v} , M é o número de pixels usados no cálculo do fluxo e λ_{min} é o menor autovalor da matriz $A^T A$.

4 Segmentação Baseada no Fluxo Óptico

Com o objetivo de facilitar a obtenção de informações do fluxo óptico e a sua posterior utilização no controle de robôs móveis, foi implementado um processo de segmentação baseada no fluxo, também chamada de segmentação de movimentos, que busca agrupar pixels que participam do mesmo movimento (Borshukov et al., 1997).

Existem vários métodos para executar esta segmentação (Borshukov et al., 1997; Wang e Adelson, 1994; Bergen et al., 1992). O método proposto por Wang e Adelson (1994) considera um modelo afim para o fluxo em regiões, e de acordo com este modelo classifica a região como pertencendo a um pequeno número de classes base usando um algoritmo "k-mean" adaptativo.

O método proposto por Borshukov et al. (1997) combina segmentação baseada em movimento dominante de Bergen et al. (1992) com "affine clustering" de Wang e Adelson (1994).

O procedimento aqui implementado se baseia no algoritmo proposto por Borshukov et al. (1997), mas utiliza um modelo constante do fluxo na região e executa o processo de agrupamento em apenas um nível, porém usando a mesma medida de semelhança.

Considerando que se tem um vetor (\hat{u}, \hat{v}) calculado em uma região $N \times N$ usando o método de Mínimos Quadrados Modificado, o qual representa toda a região, um procedimento de varredura por linhas é então iniciado, de região em região, com o objetivo de identificar entre as regiões adjacentes (vizinhança oito) quais executam movimento similar. O procedimento é o seguinte:

1- a região considerada deve ser inicialmente marcada como pertencente a um objeto. Se tal região for a primeira da imagem, então o contador de obje-

to deve ser iniciado ($n = 1$) e ela deve ser marcada com o valor de n . Se já existe algum objeto na lista ($n \geq 1$) e esta região ainda não está marcada, inicia-se um procedimento de busca na vizinhança por alguma região marcada cujo fluxo seja semelhante. A medida de semelhança usada é a distância (Borshukov et al., 1997)

$$\left[\left(u_{(k,l)} - u_{(i,j)} \right)^2 + \left(v_{(k,l)} - v_{(i,j)} \right)^2 \right] < T \quad (15)$$

onde T é um valor de limiar (da ordem de 20% da norma do vetor de fluxo na região central), (i,j) indica a região sob análise e (k,l) indica uma região em sua vizinhança. Se existir tal região similar, o seu marcador será copiado; se existirem várias regiões, o menor marcador será copiado e se não existir nenhuma região semelhante um novo marcador será criado e o número de objetos atualizado ($n=n+1$);

2- a seguir inicia-se um procedimento de busca na vizinhança por uma região cujo fluxo seja semelhante ao fluxo na região central, usando a medida apresentada na equação (15). As regiões semelhantes são marcadas como pertencentes ao mesmo objeto que a região central.

A Figura 1a mostra um quadro da seqüência de imagens considerada e a Figura 1b mostra o resultado da segmentação, sendo que cada tom de cinza representa um "pseudo-objeto" e as regiões mostradas em branco são aquelas onde a variância do fluxo é muito grande, ou seja, onde o fluxo não atende à medida de confiabilidade considerada. Como pode ser visto na Figura 1b, o processo de segmentação não leva à identificação correta dos objetos, associando mais de um "pseudo-objeto" a um único objeto real, o que é conhecido como "oversegmentation" (Borshukov et al., 1997).

Duas estratégias podem ser usadas para reduzir o número de "pseudo-objetos". A primeira, consiste em aumentar o valor de T . Porém, isto pode causar a marcação de objetos realmente distintos como um só. A segunda maneira seria introduzir uma nova etapa

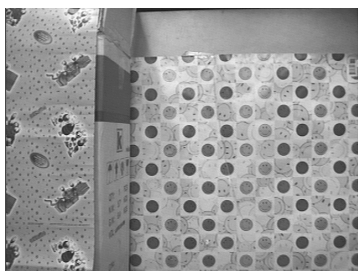


Figura 1a - Quadro da seqüência de imagens



Figura 1b - Segmentação baseada no fluxo aplicando medida de confiabilidade

no processo de segmentação, onde grupos de regiões com características semelhantes fossem remarcados. Este processo, porém, demandaria mais tempo de processamento, além de tornar necessária a criação de um novo modelo representativo dos objetos a serem comparados e uma nova medida de semelhança.

Por outro lado, o que ocorreu no processamento representado na Figura 1 foi a divisão de um objeto em partes. Porém, nenhum dos "pseudo-objetos" na imagem corresponde a partes de dois objetos reais diferentes. Sendo assim, optou-se por não modificar o procedimento de segmentação implementado.

5 Uso do Fluxo Óptico na Navegação do Robô

Para cada região é então calculado o tempo para choque (Dev et al., 1997a). Em seguida, um único valor de tempo para choque foi determinado para cada "pseudo-objeto" obtido pelo processo de segmentação, o qual corresponde ao valor mediano dos tempos para choque das regiões a ele pertencentes. Este valor indica quais objetos estão mais próximos do robô, se algum deles oferece risco ao mesmo caso sua trajetória seja mantida, e, em caso de necessidade de mudança de trajetória, qual a melhor estratégia para evitar o obstáculo.

O gráfico na Figura 2 mostra o tempo para choque mínimo nas 32 regiões em que foi dividida horizontalmente a imagem da Figura 1a. Por ele pode-se perceber que entre as colunas 0 e 9 existe um obstáculo mais próximo ao robô, e que entre as regiões 10 e 31 os obstáculos estão mais distantes, o que pode ser confirmado examinando a Figura 1a. Esta figura foi obtida considerando-se apenas os vetores de fluxo que atenderam à medida de confiabilidade e descartando-se os "pseudo-objetos" compostos por apenas uma ou duas regiões (provavelmente provenientes de falsas medições).

Na seqüência deste trabalho, uma estratégia de controle está sendo discutida, a qual se baseia em criar uma velocidade de rotação associada a cada direção do campo visual do robô. Cada uma destas velocidades angulares deve ser calculada de modo que sua intensidade seja inversamente proporcional ao tempo para choque do objeto mais próximo naquela direção e com sinal capaz de levar o robô a mudar sua direção. A velocidade angular gerada para cada direção tem que ser capaz de fazer o robô desviar do obstáculo nesta direção com intensidade

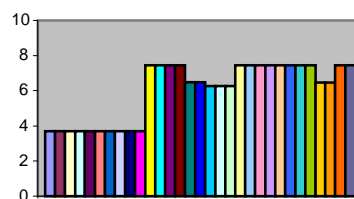


Figura 2 - Tempo para choque mínimo com confiabilidade, excluindo-se os "pseudo-objetos" menores que 3 regiões.

proporcional ao perigo que o mesmo representa. Uma velocidade de rotação resultante poderá então ser calculada levando-se em conta tais valores.

Por outro lado, a velocidade linear poderá ser controlada levando-se em conta o tempo para choque mínimo obtido para a região central da imagem: ela pode se manter em um valor preestabelecido, se o tempo para choque nesta região se mantiver acima de um valor mínimo, e ir diminuir à medida que o tempo para choque diminui.

6 Conclusão e Trabalhos Futuros

O uso da técnica de fluxo óptico para implementar o controle de navegação a bordo de um robô móvel é aqui tratado. O objetivo principal é detectar a presença de objetos próximos ao robô com base exclusivamente na informação visual.

Testes de algoritmos selecionados da literatura foram apresentados, e um algoritmo adequado para implementação a bordo do robô foi proposto, o qual é computacionalmente eficiente sem apresentar perda significativa de precisão.

Uma medida de confiabilidade (a variância associada a cada vetor de fluxo óptico) foi utilizada para eliminar do processamento os vetores que correspondem a erros de estimativa muito grandes.

Um procedimento de segmentação de imagem baseada no fluxo óptico foi então implementado, com o objetivo de delimitar os objetos na imagem capturada pela câmara a bordo do robô, e os resultados da segmentação da imagem foram apresentados para um caso exemplo.

Por fim, a próxima etapa, que está sendo iniciada, culminará com a proposta de uma estratégia de controle das velocidades linear e angular do robô, baseada nos valores de tempo para choque correspondentes a cada objeto identificado na imagem.

Agradecimentos

Os autores agradecem ao CNPq o financiamento deste trabalho, que se deu através de um auxílio à pesquisa aprovado dentro do Edital 001/2000, correspondente ao Processo número 462.882/2000-4.

Referências Bibliográficas

Barron, J. L., Fleet, D. J. e Beauchemin, S. S. (1993). Performance of optical flow techniques, *International Journal of Computer Vision*, Vol. 12, pp. 43-77.

Bergen, J. R., Burt, P. J. e Hanna, K. (1992). Dynamic multiple-motion computation, In: *Artificial Intelligence and Computer Vision* (Y. A. Feldman and A. Bruckstein, Eds.). Holland:Elsevier, pp. 147-156.

Borshukov, G. D., Bozdagi, G., Altunbasak, Y. e Tekalp, A. M. (1997). Motion Segmentation by Multi-stage Affine Classification, *IEEE Trans. On Image Processing*, vol. 6, pp. 1591-1594.

Dev, A., Kröse, B. J. A. e Groen, F. C. A. (1997a). Navigation of a mobile robot on the temporal development of the optic flow, *Proceedings of the 1997 IEEE/RSJ/GI International Conference on Intelligent Robots and Systems (IROS'97)*, Grenoble, France, pp. 558-563.

Dev, A., Kröse, B. J. A. e Groen, F. C. A. (1997b). Confidence Measures for Image Motion Estimation, *Proceedings of the 1997 RWC Symposium*, Japan, pp.199-206.

Dev, A., Kröse, B. J. A. e Groen, F. C. A. (1997c). Heading Direction for a Mobile Robot from Optical Flow, *Proceedings of the IEEE Int.Conf. on Robotics and Automation*, Leuven, Belgium, pp.558-563.

Fleet, D. J. e Jepson, A. D. (1990). Computation of component image velocity from local phase information, *International Journal of Computer Vision*, Vol. 5, pp. 77-104.

Grossmann, E. e Santos-Victor, J. A., (1997). Performance Evaluation of Optical Flow Estimators: Assessment of a New Affine Method, *Robotics and Autonomous Systems*, Vol. 21, pp. 69-82.

Horn, B. K. P. e Schunck, B. G. (1981). Determining optical flow, *Artificial Intelligence*, Vol. 17, pp. 185-203.

Lai, S. H. e Vemuri, B. C. (1998). Robust and efficient computation of optical flow, *International Journal of Computer Vision*, Vol. 29, pp. 87-105.

Lucas, B. e Kanade, T. (1981). An Iterative Image Registration Technique with an Application to Stereo Vision, *Proceedings of DARPA IU Workshop*, pp. 121-130.

Nesi, P., Del Bimbo, A. e Bem-Tzvi, D. (1995). A Robust Algorithm for Optical Flow Estimation, *Journal on Computer Vision, Graphics and Image Processing: Image Understanding*, Academic-Press, Vol. 61, No. 2, pp.59-68.

Sarcinelli-Filho, M., Schneebeli, H. A. e Caldeira, E. M. O. (2001). On Real-Time Optical Flow Calculation for Mobile Robot Navigation, *Proceedings of the 5th World Multi-Conference on Systemics, Cybernetics and Informatics – SCI2001*, vol. IX, pp. 60-65, Orlando, FL, USA.

Wang, J. Y. A. e Adelson, E. H. (1994). Representing moving images with layers, *IEEE Trans. Image Proc.*, vol. 3, pp. 625-638.