

# SISTEMA DE MEDIÇÃO DE DISTÂNCIA PARA ROBÔS MÓVEIS UTILIZANDO ALGORITMO SURF E VISÃO ESTÉREO

**Guilherme Costa HOLANDA (1); Edson CAVALCANTI NETO (1); Antonio Themoteo VARELA (1); André Luiz Carneiro de ARAÚJO (1); Auzuir Ripardo de ALEXANDRIA (1)**

(1) Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Av. Treze de Maio, 2081 - Benfica - Fortaleza/CE, +55 (85) 3307-3666,

{guilhermecosta@ifce.edu.br; edsoncavalcanti@ifce.edu.br; themoteo@ifce.edu.br; andreluiz@ifce.edu.br; auzuir@ifce.edu.br}

## RESUMO

O atual desenvolvimento da Robótica Móvel é destaque no entretenimento do ser humano e nas realizações de tarefas insalubres e perigosas para o mesmo. No entanto, há uma gama enorme de possibilidades entreabertas de aplicações. Um robô móvel típico é constituído dos subsistemas a seguir: percepção, localização, navegação e controle de movimento. O sistema de percepção é o responsável pela aquisição de dados do ambiente que o robô atua. A medição de distância é algo de fundamental importância na localização e navegação de robôs móveis. A forma clássica de medição de distância é através de sonares e radares *lasers*. Porém, o uso de visão computacional em estéreo, baseado em duas câmeras é bastante difundido na Robótica. Algoritmos recentes de detecção de pontos permitem o desenvolvimento de diferentes métodos de visão estéreo. Dentre esses novos algoritmos destaca-se o SURF (*Speeded Up Robust Features*) que pode ser utilizado em conjunto com algoritmos clássicos de visão em estéreo, no desenvolvimento de sistemas de Visão Computacional 3D. Este projeto possui como objetivo o desenvolvimento de um sistema de medição de distância através de visão computacional em estéreo, utilizando o algoritmo SURF. Os resultados obtidos são interessantes, uma vez que o maior erro percentual obtido nos testes foi menor que 10%, o que torna o sistema promissor para esta aplicação. O algoritmo SURF realmente apresenta-se como um algoritmo de rápido processamento, uma vez que a rotina leva em média 0,2 segundos para ser processada.

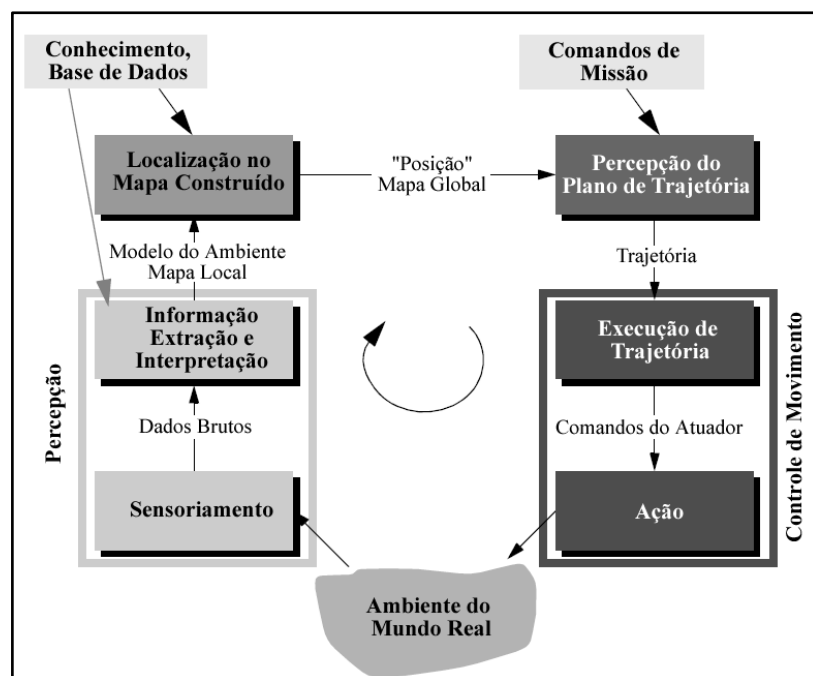
**Palavras-chave:** robótica, surf, visão 3d, cálculo profundidade

## 1. INTRODUÇÃO

Atualmente os robôs móveis autônomos vêm ganhando bastante popularidade na área da robótica, pois estes são capazes de tomar decisões sem necessitarem de ajuda contínua do ser humano. Para isto, técnicas de visão computacional tornam-se viáveis para a extração de informações do ambiente.

Em ambientes insalubres e perigosos os robôs móveis têm ganhado bastante popularidade. Um robô móvel típico é composto dos seguintes subsistemas: percepção, localização, navegação e controle de movimento que é ilustrado no diagrama da Figura 1. O sistema de percepção é responsável pela obtenção de dados do meio que o robô atua. Estes dados, além de serem armazenados como parte da própria tarefa executada pelo robô, são utilizados na geração de informações relativas à localização do mesmo em seu ambiente, através de seu sistema de localização. Com base na localização, o robô é capaz de gerar um mapa do ambiente que atua e comparar sua posição local com uma posição global no mapa gerado. A capacidade de navegar utilizando o referido mapa, de posse da missão que deve desempenhar também é tarefa extremamente necessária e realizada pelo seu sistema de navegação. Esse último informa qual trajetória o sistema de controle de movimento deve desempenhar (SIEGWART; NOURBAKHS, 2004).

Alguns robôs móveis comerciais não operam em ambientes perigosos ou insalubres, porém compartilham espaço com os humanos no ambiente humano. Estes robôs não se destacam por razões de mobilidade, mas pela autonomia e a capacidade de manterem uma percepção da posição e navegarem sem a intervenção de humanos (SIEGWART; NOURBAKHS, 2004).



**Figura 1 – Esquema de um robô móvel (SIEGWART; NOURBAKHS, 2004).**

Uma das informações importantes para um robô móvel é a medição de distância de obstáculos, que pode ser obtida através de vários tipos de sensores ou até por imagens do ambiente. A partir dessa distância, processos como localização, navegação, desvio de obstáculos e outras podem ser efetuadas. Uma das maneiras de se realizar medição de distâncias é através de imagens digitais. Como os robôs móveis normalmente possuem sistema de visão, a este se pode introduzir uma nova funcionalidade que é a medição de distância de obstáculos através de imagens digitais em estéreo. Sistemas de imagens em estéreo consistem em efetuar a aquisição simultânea ou não de duas imagens da mesma cena, de forma semelhante aos animais que possuem dois olhos. Com esse sistema é possível resgatar a informação de profundidade presente nas imagens adquiridas (RUSSEL; NORVIG, 2004).

Na literatura há trabalhos em que se usa sensor de infravermelho para a medição de distância, porém uma das desvantagens do mesmo é o fato do fecho de detecção ser bem estreito (BENET et al., 2002). Por um lado isso é uma vantagem porque pode mapear com maior precisão o ambiente ao redor, mas em compensação ele não consegue detectar um obstáculo pequeno. Há também trabalhos utilizando Visão Estéreo, como por exemplo o de Coman et al. (2010) que não aplicam algoritmos de detecção de pontos-chaves e descritores, e sim segmentação do objeto de desejo, detecção de cores e detecção de bordas na imagem.

Este projeto possui como objetivo o desenvolvimento de um sistema de medição de distância através de visão computacional em estéreo, utilizando o algoritmo SURF.

Na seção 2 encontra-se a fundamentação teórica com conceitos de Visão Computacional, Algoritmo SURF e sistema de visão em estéreo cálculo de profundidade. Na seção 3 encontra-se a metodologia, resultados e análise de resultados. Na seção 4 encontram-se considerações finais acerca do trabalho e na Seção 5 os agradecimentos.

## 2. FUNDAMENTAÇÃO TEÓRICA

### 2.1 Sistema de Visão Computacional

Segundo Filho e Neto (1999), um sistema de Visão Computacional é definido como um sistema computadorizado capaz de adquirir, processar e interpretar imagens correspondentes a cenas reais.

O primeiro processo em um sistema de visão é a aquisição de imagem. Neste processo, uma iluminação adequada a ser utilizada na cena é de fundamental importância, proporcionando uma imagem, cuja qualidade pode facilitar os outros processos do sistema. Outras modalidades diferentes da luz visível, como o infravermelho, microondas, raios-X, raios gama, entre outras radiações eletromagnéticas, ou ondas acústicas (sistemas de imagens ultrassom) ou ainda campo magnético (ressonância magnética), podem ser utilizadas

para se constituir a imagem de entrada do sistema (HEIDJEN, 1994; TSUJI et al., 2002). Para a aquisição de imagens, um dispositivo, adequadamente especificado para esta aplicação, deve ser utilizado, normalmente uma câmera de vídeo ou máquina fotográfica. Dispositivos com tecnologia CCD e CMOS são os mais utilizados atualmente. Em caso de saída analógica da câmera de vídeo, a utilização de um conversor analógico/digital se faz necessária (XILINX, 1999; CARNEIRO; RODRIGUES; GARCIA, 2004).

Após a aquisição da imagem, esta deve ser submetida a um pré-processamento, que consiste normalmente em operações de Processamento Digital de Imagens realizadas para extrair ruído, efetuar restauração ou realce (MARSHALL e MARTIN, 1992).

## 2.2 Algoritmo SURF (*Speeded Up Robust Features*)

O algoritmo SURF (BAY et al., 2006) inclui detector de pontos-chaves e descritores. O SURF utiliza uma representação especial de imagem conhecido como *integral image* a fim de acelerar o processo de detecção. A detecção de pontos-chaves é efetuada a partir da construção de uma representação em espaço-escala da imagem.

A representação em espaço-escala é realizada a partir da convolução da imagem com um núcleo gaussiano. Essa representação é um tipo especial de representação em multi-escala que abrange um parâmetro contínuo da escala e conserva a mesma amostragem espacial para todas as escalas. Desta forma, pode-se entender o espaço-escala de uma imagem como um conjunto de imagens do mesmo tamanho da imagem original (SCHWEIGER, 2009; ZANIN, 2003; LINDBERG, 1994).

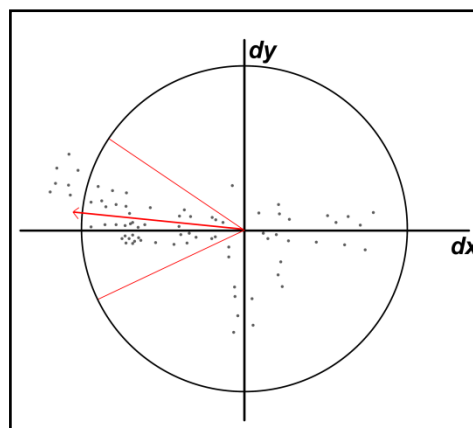
Na detecção de características no espaço de escala, o SURF utiliza o determinante da matriz Hessiana. Dado um ponto  $x = [x, y]$  em uma imagem  $I$ , a matriz Hessiana  $D(x, \sigma)$  em  $x$  na escala  $\sigma$  é definida como

$$D_{\text{SURF}}(x, \sigma) = \begin{bmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{bmatrix}, \quad [\text{Eq. 01}]$$

em que  $L_{xx}(x, \sigma)$  é a convolução da derivada de segunda ordem da gaussiana com a imagem  $I$ , no ponto  $x$  e similarmente para  $L_{xy}(x, \sigma)$  e  $L_{yy}(x, \sigma)$ . Essas derivadas são conhecidas como Laplaciano de gaussianas (EVANS, 2009).

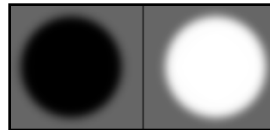
Para garantir a invariância à rotação, identificam-se os pontos-chaves com uma orientação calculada a partir da resposta da Transformada Wavelet de Haar nas direções  $x$  e  $y$  em uma vizinhança circular centrada no ponto de interesse e de raio dependente da escala (BAY et al, 2006). De acordo com Lima (2002), as *wavelets* podem distinguir as características locais de um sinal em diferentes escalas e, por translações, elas cobrem toda a região na qual o sinal é estudado.

Para aumentar a robustez diante de deformação geométrica e erros de localização, ponderam-se as respostas da transformada com um núcleo gaussiano centrado no ponto-chave. Após isso, as respostas são representadas como pontos no espaço vetorial. A orientação dominante é obtida a partir da rotação de um segmento de círculo de ângulo  $60^\circ$  em torno da origem. Em cada posição, as respostas dentro do segmento são somadas e usadas para formar um novo vetor. O maior vetor indica a orientação que deve ser atribuída ao ponto-chave, conforme a Figura 2 (BAY et al, 2006).



**Figura 2 – Atribuição da orientação.**

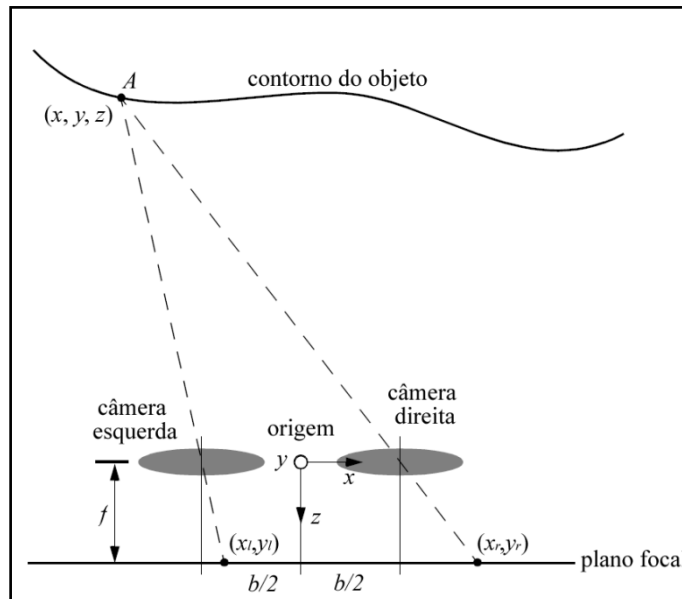
Para encontrar correspondências entre pontos-chaves de duas imagens a partir de seus respectivos descritores, é necessário localizar os melhores candidatos a serem seus equivalentes. Isto é realizado procurando-se o vizinho mais próximo do descritor do ponto entre todos os possíveis candidatos. Utiliza-se a mínima distância euclidiana entre os descritores. Mas nem sempre o vizinho mais próximo é o correspondente, por isso é necessária a utilização da distância euclidiana com o segundo vizinho mais próximo, sendo excluídas correspondências que têm a razão entre o vizinho mais próximo e o segundo maior que um determinado limiar, removendo assim grande parcela de falsos pontos correspondentes. Utiliza-se também o sinal do laplaciano, que distingue o brilho de pontos em fundos escuros da situação inversa, conforme a Figura 3. Na fase de correspondência de pontos, comparam-se apenas pontos-chaves que têm os mesmos tipos de contraste. Assim, essa pequena informação permite uma fase de correspondência mais rápida, sem reduzir o desempenho do descritor. Resumindo, se o contraste entre dois pontos de interesse é diferente, o ponto candidato não é considerado um possível ponto correspondente (BAY et al., 2008).



**Figura 3 – Distinção de brilho de pontos em fundos escuros da situação inversa a partir do sinal do laplaciano.**

### 2.3 Sistema de visão em estéreo e cálculo de profundidade

Um sistema de visão em estéreo é um sistema em que duas imagens da mesma cena são adquiridas em condições diferentes, de forma a ser possível efetuar medições em três dimensões. Sistemas de visão animais realizam essa tarefa facilmente, sendo essa a razão dos animais possuírem normalmente dois olhos. Sendo assim, utiliza-se o sistema de geometria apresentado na Figura 4 constituído de duas câmeras de distância focal  $f$  equidistantes de uma origem a uma distância  $b/2$ . Sendo os símbolos  $A$ ,  $b$ ,  $f$ ,  $x_l$ ,  $y_l$ ,  $x_r$ ,  $y_r$  e  $z$  correspondentes a, respectivamente, ponto do objeto analisado, distância entre centros ópticos das câmeras, distância focal, coordenada  $x$  do ponto  $A$  na imagem da câmera esquerda, coordenada  $y$  do ponto  $A$  na imagem da câmera esquerda, coordenada  $x$  do ponto  $A$  na imagem da câmera direita, coordenada  $y$  do ponto  $A$  na imagem da câmera direita e distância entre a origem e o ponto  $A$  (profundidade). As câmeras da esquerda e da direita com seus sistemas de lentes são denominadas  $l$  e  $r$ , respectivamente (SIEGWART; NOURBAKHS, 2004).



**Figura 4 – Geometria de um sistema de visão em estéreo (SIEGWART; NOURBAKHS, 2004).**

Ao visualizar-se um objeto, cada ponto de seu contorno visível possui uma coordenada  $(x, y, z)$  no espaço físico em relação a uma origem localizada entre as lentes esquerda e direita. A medição de profundidade na coordenada  $z$  é dada por (SIEGWART; NOURBAKHS, 2004):

$$z = b \frac{f}{x_l - x_r}, \quad [\text{Eq. 02}]$$

em que a diferença das coordenadas  $x_l - x_r$  das imagens é chamada de *disparidade*.

Pode-se destacar algumas consequências da Eq. 02, tais como:

- a distância é inversamente proporcional à disparidade. A distância de objetos próximos pode ser medida com mais precisão do que para objetos distantes. Em geral, isso é aceitável para a Robótica Móvel, porque para a navegação o desvio de obstáculos mais próximos é de maior importância;
- a disparidade é proporcional à  $b$ . Para um dado erro de disparidade, a precisão da estimativa de profundidade aumenta com o aumento da distância  $b$  entre as câmeras;
- quanto mais a distância  $b$  aumenta, devido a separação física entre as câmeras, alguns objetos podem aparecer em uma das câmeras, mas não na outra. Tais objetos, por definição, não possuem uma disparidade e, portanto, o cálculo da profundidade não é calculado com êxito (SIEGWART; NOURBAKHS, 2004).

Além da distância  $b$  entre as câmeras e da distância focal  $f$  de seus conjuntos ópticos, necessitam-se das coordenadas  $x_l$  e  $x_r$  de um determinado ponto para se calcular a profundidade no eixo  $z$ . Para isso, obtêm-se uma imagem da câmera esquerda e uma da câmera direita, ambas contendo o ponto no contorno do objeto visualizado (SIEGWART; NOURBAKHS, 2004). O desafio é identificar qual ponto na imagem da direita é correspondente a um determinado ponto da imagem da esquerda. Uma maneira clássica de se efetuar essa relação entre  $x_l$  e  $x_r$  é utilizando correlação, outra forma bastante em pauta é a utilização de detectores invariantes de características, por exemplo, o SURF (BAY et al., 2006), sendo este mais robusto, em razão da sua promessa de invariância a translação, rotação e escala, além de intensidade luminosa dos pixels.

Dessa forma, pode-se, baseado nesse detector, implementar um sistema capaz de realizar medição de distâncias utilizando um sistema de visão computacional estéreo. Para isso, necessita-se de duas câmeras idênticas (mesmas dimensões e características digitais e óticas), espaçadas de uma distância  $b/2$ . A partir dos algoritmos dos detectores, identificam-se pontos-chaves nas imagens das duas câmeras e encontram-se os pontos correspondentes de ambas. Após o uso do detector e de posse das coordenadas dos pontos correspondentes, calcula-se a distância  $z$  para cada ponto através da Eq. 02. O referido processo é resumidamente apresentado no fluxograma da Figura 5.

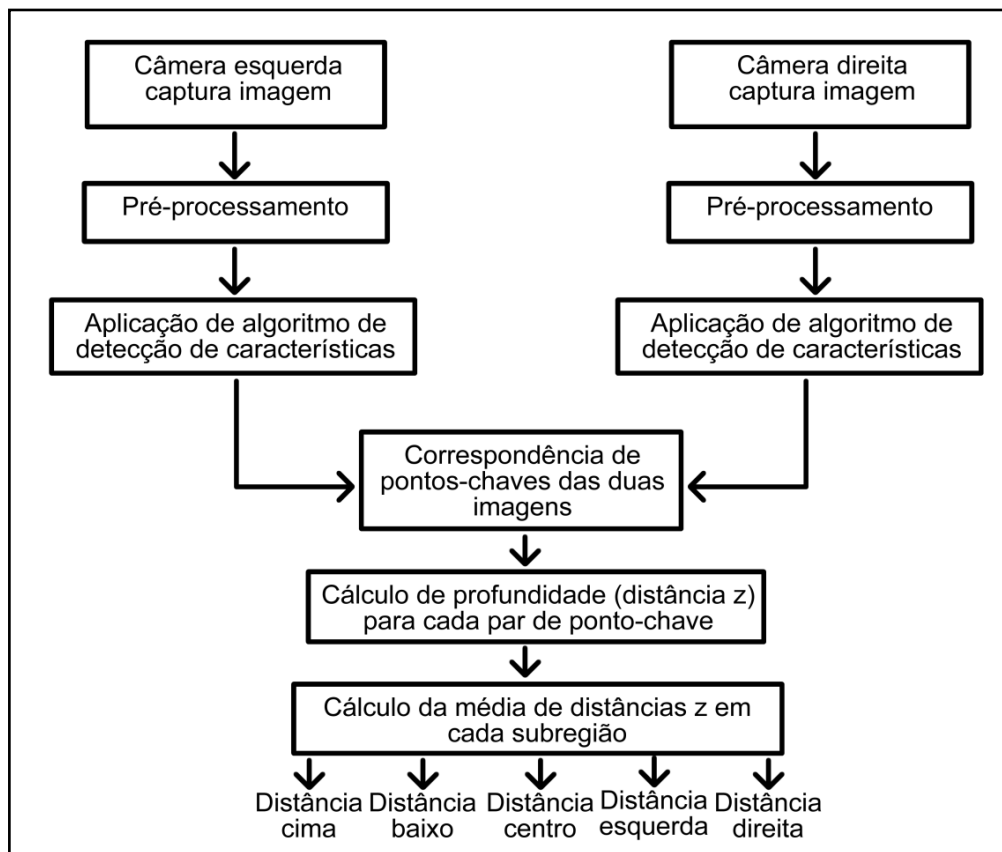


Figura 5 – Etapas do processo de cálculo de profundidade.

Abaixo segue um resumo das etapas do processo de cálculo de profundidade:

- Captura: obtêm-se as imagens das câmeras esquerda e direita na escala RGB de cores de resolução 320x240 *pixels*. As duas imagens devem conter o objeto que se deseja calcular a distância.
- Pré-processamento: as imagens na escala RGB são convertidas para tons de cinza, este processo é útil para que se possa haver uma continuidade radiométrica entre os *pixels* e para que as imagens possam servir de entrada para o algoritmo SURF.
- Detecção de características da imagem: a partir do algoritmo SURF são obtidos pontos-chaves de ambas imagens com seus respectivos descritores.
- Correspondência de pontos-chaves: a partir dos pontos-chaves com seus respectivos descritores locais, O algoritmo SURF identifica pontos correspondentes nas duas imagens, ou seja, identifica que ponto da imagem direita é correspondente a um determinado ponto da imagem esquerda.
- Cálculo de profundidade: de posse dos pares de pontos correspondentes, utiliza-se a Eq. 02 a fim de calcular a distância  $z$  (profundidade) para cada par de pontos correspondentes.
- Divisão da imagem em sub-regiões: criam-se cinco sub-regiões: cima, baixo, centro, esquerda e direita conforme a Figura 6 e calcula-se a média da distância  $z$  para cada uma, a fim de se definir uma aproximação da profundidade de cada sub-região.



Figura 6 – Sub-regiões definidas: esquerda, direita, cima, baixo e centro.

### 3. METODOLOGIA, RESULTADOS E ANÁLISE

Implementam-se as rotinas de medição de distância a partir de uma versão disponível do algoritmo SURF da OpenCV (INTEL) para o software MATLAB 2008 (SURFMEX, 2010) executada no Windows XP.

A estrutura do Sistema é montada com duas *webcams* da marca Bright modelo Atlanta. Esta câmera possui conexão USB, sensor CMOS, suporta 24 bits de cores e é capaz de capturar 30 quadros por segundo. A resolução da câmera é de 1,3 *Megapixels* e possui foco ajustável. A distância  $b$  entre os centros ópticos é de 116 mm. Um desenho do sistema de visão estéreo montado é ilustrado na Figura 7.

Uma dificuldade enfrentada foi se descobrir a distância focal  $f$  do sistema, uma vez que o fabricante não especifica essa característica de cada câmera, então foi utilizada uma *toolbox* de calibração de câmera para MATLAB a fim de se obter uma aproximação para essa distância focal que foi de aproximadamente 520 *pixels* (BOUGUET), porém uma posterior calibração mais detalhada é necessária para aumentar a robustez do sistema.

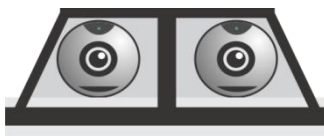
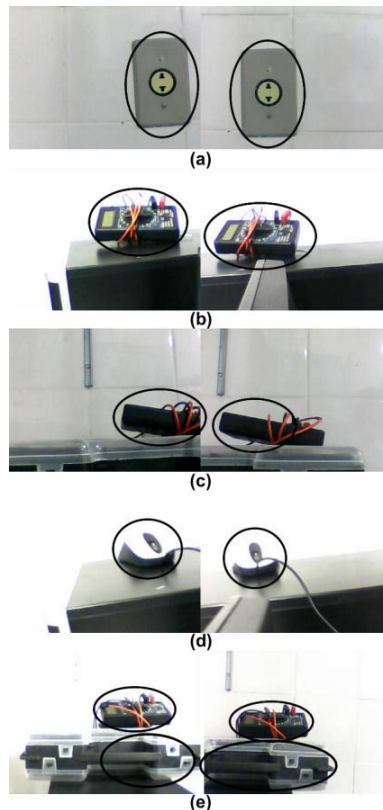


Figura 7 – Desenho do sistema de visão estéreo montado.

Para a verificação da rotina nos testes realizados são utilizados cinco pares de imagens obtidos pelas *webcams*, com alguns objetos conhecidos, como tomadas, *mouses*, multímetros, conforme ilustra a Figura 8. No par de imagens da Figura 8 (e), diferentemente dos outros, são calculadas as distâncias de dois objetos. As imagens são convertidas para tons de cinza e o algoritmo SURF usa-as como entrada a fim de obter pontos correspondentes de ambas as imagens. De posse dos pontos correspondentes, torna-se possível o cálculo de profundidade para cada par de pontos, conforme foi explanado na Seção 2.3. Através do cálculo da média de distancias, são obtidas aproximações de distâncias para as cinco sub-regiões predefinidas. Os resultados obtidos são exibidos na Tabela 1.



**Figura 8 – Pares de imagens utilizados para teste (a) tomada (b) multímetro (d) mouse (e) objetos.**

Resultado dos testes realizados					
Teste	Distância real (mm)	Distância calculada pelo Sistema (mm)	Número de pontos correspondentes	Erro percentual relativo (%)	Tempo de processamento (s)
a) tomada	400	401	15	0,25	0,142
a) multímetro	460	447	9	2,83	0,311
b) multímetro 2	330	310	5	6,06	0,132
c) mouse	340	368	7	8,24	0,184
d) objetos	445 / 560	486 / 580	8	9,21 / 3,57	0,256
<b>Erro médio absoluto (mm):</b>		21	<b>Tempo médio de processamento (s)</b>		0,205

**Tabela 1 – Resultado dos testes realizados**

Os resultados obtidos são interessantes, uma vez que o erro médio absoluto foi de 21 mm e o maior erro percentual obtido foi menor que 10%, o que torna o sistema promissor para esta aplicação. O algoritmo SURF realmente apresenta-se como um algoritmo de rápido processamento, uma vez que a rotina leva em média 0,2 segundos para ser processada. Uma calibração mais precisa nas câmeras reduz o erro e consequentemente proporciona um desempenho maior no sistema.

#### 4. CONSIDERAÇÕES FINAIS

Para a utilização do sistema de visão do robô para localização e navegação, a informação da distância em relação a obstáculos, paredes ou quaisquer outros objetos do ambiente é bastante útil. Como trabalhos futuros são sugeridos a mudança do sistema da plataforma de simulação MATLAB para o ambiente C++ Builder e uma calibração mais precisa do sistema.

#### 5. AGRADECIMENTOS

Agradecimentos aos Laboratórios LEM (Laboratório de Ensaio Mecânicos) e ITTI (Instituto de Telemática) do Instituto Federal de Educação, Ciência e Tecnologia do Ceará.

## REFERÊNCIAS

- BAY, H.; TUYTELAARS, T.; GOOL, L. V. **SURF: Speeded Up Robust Features**. In European Conference on Computer Vision, pp. 404–417, 2006.
- BAY, H.; ESS, A.; TUYTELAARS, T.; GOOL, L. V. **SURF: Speeded Up Robust Features**, Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359, 2008.
- BENET, G.; BLANES, F.; SIMO, J. E.; PEREZ, P. **Using infrared sensors for distance measurement in mobile robots**. *Robotics and Autonomous Systems*, 40 (4), pp. 255-266, 2002.
- BOUGUET, J. Y. **Complete camera calibration toolbox for Matlab**. Disponível em: <[http://www.vision.caltech.edu/bouguetj/calib\\_doc](http://www.vision.caltech.edu/bouguetj/calib_doc)>. Acesso em: 14 jul. 2010.
- CARNEIRO, C. A.; RODRIGUES, D. L.; GARCIA, F. M. P. **Plataforma de Aquisição de Sinais de Vídeo com Processamento Digital de Imagens em Hardware**. Anais do XV Congresso Brasileiro de Automática – CBA 2004, Gramado-RS, set., 2004.
- COMAN, M.; STAN, S. D.; MANIC, M.; BALAN, R. **Application of distance measuring with Matlab/Simulink**. HSI'10, the 3rd International Conference on Human System Interaction, Rzeszow, Poland, May 13-15, 2010.
- FILHO, O. M.; NETO, H. V. **Processamento Digital de Imagens**, Rio de Janeiro: Editora Brasport, 1999.
- HEIDJEN, F. V. **ImageBased Measurement Systems Object Recognition**. Inglaterra: John Wiley and Sons Ltda, 1994.
- INTEL. **Intel Open Source Computer Vision Library**. Intel Research. Disponível em: <<http://opencv.willowgarage.com/wiki/>>. Acesso em: 14 jul. 2010.
- MARSHALL, A. D.; MARTIN, R. R. **Computer Vision, Models and Inspection**. Londres - Inglaterra: World Publishing Company - Series in Robotics and Automated Systems, 1992.
- RUSSEL, S.; NORVIG, P. **Inteligência Artificial**. Rio de Janeiro: Elsevier, 2004.
- SCHWEIGER, F.; ZEISL, B.; GEORGEL, P.; SCHROTH, G.; STEINBACH, E.; NAVAB, N. **Maximum Detector Response Markers for SIFT and SURF**. In *Vision, Modeling and Visualization Workshop (VMV)*, Braunschweig, 2009.
- SIEGWART, R.; NOURBAKHSH, I. R. **Introduction to Autonomous Mobile Robots**. Cambridge: The MIT Press, 2004.
- SURFMEX. **A MATLAB SURF interface**. Disponível em: <<http://www.maths.lth.se/matematiklth/personal/petter/surfmx.php>> Acesso em: 14 jul 2010.
- TSUJI, T. et al. **Development of Night-Vision System**. IEEE Transaction on Intelligent Transportation Systems IEEE Transaction on Dielectrics and Electrical Insulation, v. 3, n. 3, p. 203-209, set., 2002.
- XILINX. **The Design of a Video Capture Board Using the Spartan Series**. USA, mar., 1999.
- ZANIN, R. B. **Metodologia Automática para Extração de Cruzamentos de Rodovias em Imagens de Alta Resolução**. RBC – Revista Brasileira de Cartografia, N° 55/2, 2003.