



CONTROLE DE MANIPULADOR REDUNDANTE UTILIZANDO REALIMENTAÇÃO VISUAL

Anfranserai Morais Dias

Universidade Federal do Rio Grande do Norte, DCA - CT - UFRN,
Lagoa Nova, Natal - RN, Cep.: 59072-970
xamd@dac.ufrn.br

Pablo Javier Alsina

Universidade Federal do Rio Grande do Norte, DCA - CT - UFRN,
Lagoa Nova, Natal - RN, Cep.: 59072-970
pablo@dac.ufrn.br

Resumo. *No controle convencional de manipuladores robóticos, a trajetória desejada é especificada em espaço cartesiano e mapeada para espaço de juntas através do modelo cinemático inverso do manipulador. As novas referências assim geradas são utilizadas para fins de controle dinâmico em espaço de juntas, desde modo a posição da garra é controlada efetivamente em malha aberta e a precisão do controle depende diretamente da precisão do modelo cinemático disponível. Neste artigo, um novo algoritmo de controle cinemático para braços redundantes baseado em realimentação visual é proposto. No sistema proposto, a imagem do robô é captada por uma câmera e processada, para obter a posição e orientação de cada um dos elos do robô. A trajetória do robô é especificada na forma de uma seqüência temporal de imagens de referência do braço robótico. Assim, ambas as poses, a medida e a de referência são especificadas no mesmo espaço de imagem e a sua diferença é utilizada para gerar um erro em espaço cartesiano para propósitos de controle cinemático. O esquema de controle proposto foi aplicado a um manipulador redundante planar de quatro graus de liberdade, resultados experimentais são apresentados.*

Palavras-chave: *Realimentação Visual, Controle, Manipulador Redundante, Cinemática.*

1 INTRODUÇÃO

Braços robóticos manipuladores redundantes são utilizados em aplicações que requerem elevada capacidade de manipulação. Graus de liberdade adicionais permitem que estes robôs consigam manipular objetos em espaços de trabalho povoados de obstáculos complexos. Definida uma trajetória desejada para a ferramenta no espaço de trabalho, a sua execução requer que a mesma seja mapeada numa trajetória correspondente em espaço de junta, *Controle Cinemático*. A execução da trajetória é realizada pelos servos controladores existentes nas juntas, *Controle Dinâmico*. Num manipulador redundante, o mapeamento não é bijetivo, ou seja, existem infinitas soluções para o problema de controle cinemático. Para resolver este problema é necessário determinar algum critério de otimização, de modo a obter uma única solução ótima dentro das infinitas possíveis. Um método clássico para solução do problema de controle cinemático de manipuladores redundantes é controle baseado em matriz pseudo-inversa (Nakamura, 1991; Chiaverini, 1997).

Em contrapartida, a maioria dos braços manipuladores encontrados no mercado, utilizam encoders para medir os ângulos de junta para determinar a posição do braço com objetivos de controle. Desta maneira, o manipulador é, de fato, controlado em espaço de juntas, sendo a trajetória da ferramenta controlada indiretamente através do mapeamento definido pelo controle cinemático. Assim, teoricamente, se for imposto um erro de rastreamento de trajetória nulo em espaço de juntas, deveria ter a princípio, um erro de rastreamento nulo também na trajetória da ferramenta. Isto é um pressuposto válido apenas se o modelo cinemático inverso disponível do robô é exato. Quanto mais impreciso é este modelo, menor é a garantia de que a ferramenta será controlada com precisão, apesar de que um erro nulo seja garantido em espaço de juntas. Uma alternativa para contornar este problema é utilizar sensores que forneçam diretamente a posição da ferramenta em espaço cartesiano. Um sensor apropriado para este tipo de medição é um sistema de visão (Hutchinson et al., 1996), a partir da imagem do robô captada por uma câmera, calcule diretamente a posição do braço em espaço cartesiano. Diversos sistemas de controle de braços manipuladores baseados em realimentação visual, seguindo a filosofia descrita acima, vêm sendo propostos na literatura (Corke, 1996; Zha et al., 1996). Processamento de imagem é uma atividade que consome muito tempo de computação. Assim, o maior desafio para este tipo de controladores é a implementação de algoritmos de visão computacional o suficientemente rápidos para possibilitar o controle em tempo real.

Este artigo propõe um sistema de controle baseado em realimentação visual para um manipulador redundante planar de quatro graus de liberdade. A Figura 1, mostra o fluxo de informação do sistema proposto. A visão é responsável pelo cômputo da posição e orientação dos elos e da ferramenta do manipulador, as quais serão comparadas com os valores definidos como trajetória de referência, de modo a gerar um erro de realimentação em espaço cartesiano para objetivos de controle. O controle, responsável pelos movimentos do manipulador, verifica se o objetivo foi alcançado, caso contrário, ele gera novos valores para os ângulos de junta que são aplicados aos atuadores, uma nova imagem é processada repetindo o processo até que o objetivo seja atingido.

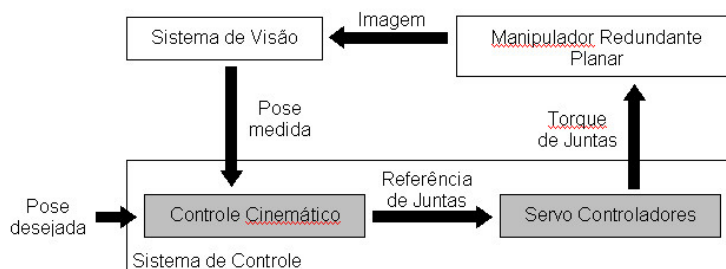


Figura 1: Diagrama de blocos para o sistema proposto.

A seção 2 descreve o funcionamento e os algoritmos utilizados para realizar o processamento das imagens captadas. A seção 3 apresenta as técnicas de controle utilizadas e por fim alguns resultados inerentes a proposta desenvolvida.

2 SISTEMA DE VISÃO

No sistema de visão proposto a aquisição de imagens é feita utilizando uma câmera CCD (*Charged Coupled Device*) colorida, que transforma a informação ótica em sinal elétrico. Esse sinal é digitalizado por uma placa de aquisição de imagens PCI 1411, esta placa fornece ao

sistema de processamento uma representação matricial da imagem, podendo ser representada em dois modelos de cores, RGB ou HSL. Essas matrizes são armazenadas por um computador pessoal que é responsável pelo processamento da imagem.

O tratamento da imagem adquirida se dá em dois momentos distintos. Num primeiro momento é feito uma calibração do sistema para estabelecer parâmetros ótimos que permitem a classificação precisa dos pixels, isto é feito considerando as variações das condições de iluminação de cada ambiente onde o sistema venha a ser usado, tornando o sistema robusto. Com os dados da calibração o sistema pode realizar a segunda fase do processamento, que consiste na localização dos objetos na cena adquirida.

2.1 Rótulos

Rótulos circulares nas cores: vermelho, verde, azul, amarelo e magenta, são dispostos sobre os eixos das juntas e sobre extremidade do último elo do robô. Os rótulos são usados no processamento de imagens, tanto na calibração quanto no processo de localização.

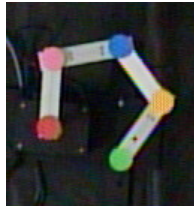


Figura 2: Descrição do Ambiente.

A Figura 2 ilustra o ambiente de trabalho, que é caracterizado por um fundo preto, pelos elos do manipulador, que tem sua cor próxima da branca e pelos rótulos. Com base nas características do ambiente é definido o modelo de cor utilizado. Um modelo de cor é um subespaço de um sistema de coordenadas tridimensionais, onde cada ponto do subespaço representa uma cor. O digitalizador empregado pode trabalhar com os modelos RGB e HSL. No ambiente de trabalho há uma predominância das cores preto e branco, que leva a um processamento de imagem mais complexo quando utilizado o modelo HSL (Aires, 2001), o que resultou na adoção do modelo RGB.

2.2 Processo de Calibração

A calibração é um processo automático realizado *off-line*, isto é, com o sistema fora de operação. A calibração consiste no ajuste de parâmetros como resolução, contraste, definição das cores, determinação dos limites do espaço de trabalho.

O processo de calibração pode ser visto em dois momentos. O primeiro momento é realizado manualmente, onde o usuário informa ao sistema alguns parâmetros como diâmetro e tamanho dos rótulos, limites da área vista pela câmera, em pixels e centímetros, altura da câmera em relação ao plano de trabalho. Alguns ajustes no sistema câmera/digitalizador também são necessários, como foco, brilho, saturação, contraste, frequência da placa de aquisição de imagens.

O segundo momento da calibração consiste no ajuste e classificação das cores. A calibração tem como finalidade aumentar a robustez do sistema, suportando pequenas variações de luminosidade existentes na imagem. Para classificação das cores se faz uso de técnicas de agrupamento de padrões semelhantes denominadas de técnicas de clusterização. Uma dessas técnicas é

o algoritmo *K-means*, é um algoritmo baseado em heurística, eficiente e de rápida convergência, característica necessária para um sistema que trabalha em tempo real (Jain, 1989).

O algoritmo *K-means* trabalha da seguinte forma:

Passo 1 - Inicialize os K centros (w_1, \dots, w_k);

Passo 2 - Associe cada centro w_j a uma classe distinta C_j ;

Passo 3 - Para cada padrão de entrada p_i , calcule a distância entre o padrão p_i e cada um dos centros w_j ;

Passo 4 - Classifique o padrão p_i como pertencente a classe C_j do centro w_j mais próximo;

Passo 5 - Atualize os centros pela média dos padrões pertencentes a cada classe;

Passo 6 - Calcule a função erro:

$$erro = \sum_{j=1}^k \sum_{p_i \in C_j} |p_i - w_j| \quad (1)$$

Passo 7 - Caso o erro seja menor que uma dada tolerância, pare o algoritmo, caso contrário retorne ao passo 3.

O número de centros deve ser maior que o número de cores a serem classificadas, com isso, tem-se cinco centros definidos pela quantidade de rótulos, um centro para cor de fundo, o preto, um centro para a cor branca presente nos elos do manipulador, e mais três classes para elementos espúrios, totalizando dez classes. Para uma boa definição das classes, tem-se o número de vinte centros para a implementação do *K-means*.

O resultado do *K-means* é dado pelos vinte centros processados, associado a cada um deles certa quantidade de padrões de entrada. Desses vinte centros são escolhidos para o processo de localização as cinco classes dos rótulos, que contém número de padrões próximo a quantidade de pixels dos rótulos. Acrescido a elas cinco outras classes que concentram as maiores quantidades de padrões, duas dessas classes definem a classe preto e a branco, as restantes agrupam padrões espúrios. Para a obtenção de um resultado ótimo com o *K-means*, é necessário uma boa distinção entre as classes, para isso, as cores dos rótulos são escolhidas o mais distante possível uma das outras.

2.3 Localização

O processo de localização de um rótulo consiste em duas fases: varredura e cálculo do centróide. O procedimento de varredura visa encontrar na imagem um pixel pertencente ao rótulo procurado, a partir da análise de uma fração de pixels escolhidos dentro de uma área da imagem na qual há grande probabilidade de encontrar o rótulo. No caso do rótulo da base, que é fixo, a área é um retângulo definido pelo usuário, Fig. 3(a). Para os demais rótulos, a área de varredura é definida por uma coroa circular centrada no rótulo precedente na cadeia cinemática, Fig. 3(b). Dois procedimentos diferentes de varredura foram utilizados: seqüencial e aleatório. No procedimento seqüencial, os pixels são escolhidos dentro da área de varredura a intervalos regulares. No procedimento aleatório, os pixels são escolhidos aleatoriamente na área de varredura.

A partir do pixel do rótulo encontrado pela varredura, um pequeno número de pixels vizinhos serão analisados de modo a determinar a posição do centróide do rótulo. O centróide é um pixel próximo ou no centro da imagem do rótulo, esta possível diferença entre a posição do centro e do centróide se deve a digitalização da imagem. Na imagem capturada, o rótulo é um conjunto

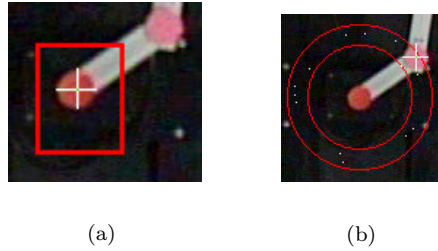


Figura 3: Áreas de Varredura.

de pixels, pequenos quadrados que tendem a formar uma circunferência. O algoritmo para o cálculo do centróide procede da seguinte forma:

- Passo 1** - Dado o pixel P_{ini} encontrado na varredura, trace um segmento de reta horizontal até os limites P_1 e P_2 do rótulo;
- Passo 2** - De posse dos limites calcule o ponto central P_{ch} do segmento de reta;
- Passo 3** - A partir do ponto central P_{ch} , trace um segmento de reta vertical, até os limites P_3 e P_4 do rótulo;
- Passo 4** - De posse desses novos limites calcule o ponto central P_c desta reta;
- Passo 5** - O ponto central P_c calculado no passo anterior define o centróide do rótulo.

A posição do centro pode diferir do centróide caso uma ou as duas retas tenham um número par pixels ou caso ocorra alguma pequena deformação na imagem adquirida.

A localização pode ser vista em duas etapas distintas. A primeira etapa é realizada com o sistema de controle fora de operação, localizando o primeiro rótulo, situado sobre a junta da base. Manualmente, através do *mouse* ou de coordenadas dadas pelo usuário via teclado, é demarcada uma pequena região da imagem que contém o primeiro rótulo. Dentro dessa região procura-se por um pixel com a cor do rótulo da base definida na calibração. Ao encontrar o pixel procede-se o cálculo do centróide.

No segundo momento, o primeiro rótulo sobre a junta da base já tem sua localização conhecida. A distância entre os centros dos rótulos também é conhecida, com estes valores é demarcada uma área de varredura no formato de coroa circular em torno do centróide conhecido. O algoritmo para esta varredura é dado por:

- Passo 1** - Definir como centróide atual, o centróide encontrado para o rótulo da base P_c ;
- Passo 2** - Definir centróide atual como centro da varredura e o rótulo a ser encontrado;
- Passo 3** - Calcular do ponto P_t pertencente a coroa circular a ser testado;
- Passo 4** - Atualizar o ponto P_t até encontrar o rótulo;
- Passo 5** - Encontrado o novo rótulo, calcular o centróide P_{cn} ;
- Passo 6** - Atualizar o ponto P_c com o valor de P_{cn} ;
- Passo 7** - Se todos os rótulos tiverem sido encontrados fim da localização, caso contrário continuar a varredura retornando ao passo 2.

3 CONTROLE CINEMÁTICO

O sistema de controle proposto para o manipulador redundante apresentado neste artigo, calcula apropriadamente os ângulos a serem aplicados aos atuadores para que seja possível a execução do movimento.

O sistema de controle é baseado na diferença entre poses, uma adquirida a cada instante pelo sistema de visão e a pose de referência. Num primeiro instante o sistema conhece os valores das posições e orientação da pose de referência e a pose atual obtida através do sistema de visão. O problema do sistema de controle resume-se então a mapear a diferença entre poses, que está representada em espaço cartesiano numa representação em espaço de juntas. Este mapeamento é estabelecido pela matriz Jacobiano do robô (Craig, 1989).

O propósito deste trabalho é realizar o controle não apenas da pose da ferramenta, mas da pose do braço como um todo. O cálculo do Jacobiano é realizado com base na geometria do manipulador. A junta da base é utilizada como referencial, restando à calcular a posição das três juntas restantes e a posição da ferramenta.

$$x_i = x_{i-1} + L_i \cos\left(\sum_{j=1}^i \theta_j\right); \quad y_i = y_{i-1} + L_i \sin\left(\sum_{j=1}^i \theta_j\right) \quad (2)$$

Como o apresentado em tópicos anteriores, o ambiente de trabalho está contido num plano, portanto, as posições são representadas por pares de coordenadas, (x_i, y_i) , onde i é o número da junta que varia de um a quatro, θ_i é o ângulo da junta i , e L_i é o comprimento do elo i , como o mostrado em Fig. 4.

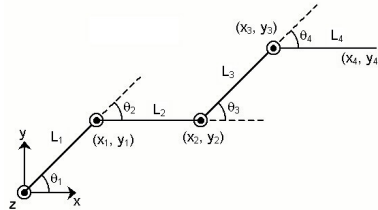


Figura 4: Descrição cinemática do manipulador.

Derivando a Equação 2, e assumindo que os elementos diferenciais são pequenos, a derivada pode ser aproximada por diferenças, ficando da seguinte forma:

$$\Delta x_i = \Delta x_{i-1} - L_i \sin\left(\sum_{j=1}^i \theta_j\right) \cdot \sum_{j=1}^i \Delta \theta_j; \quad \Delta y_i = \Delta y_{i-1} + L_i \cos\left(\sum_{j=1}^i \theta_j\right) \cdot \sum_{j=1}^i \Delta \theta_j \quad (3)$$

Adequando Eq. 3 a uma notação matricial:

$$\Delta \mathbf{P}_i = \Delta \mathbf{P}_{i-1} + \begin{bmatrix} -L_i \sin\left(\sum_{j=1}^i \theta_j\right) \cdot \sum_{j=1}^i \Delta \theta_j \\ L_i \cos\left(\sum_{j=1}^i \theta_j\right) \cdot \sum_{j=1}^i \Delta \theta_j \end{bmatrix}; \quad \Delta \mathbf{P}_i = \begin{bmatrix} \Delta x_i \\ \Delta y_i \end{bmatrix} \quad (4)$$

O Jacobiano resulta numa matriz $J_{8 \times 4}$, é apresentado na forma matricial como:

$$\begin{bmatrix} \Delta P_1 \\ \Delta P_2 \\ \Delta P_3 \\ \Delta P_4 \end{bmatrix} = \begin{bmatrix} J_{11} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ J_{21} & J_{22} & \mathbf{0} & \mathbf{0} \\ J_{31} & J_{32} & J_{33} & \mathbf{0} \\ J_{41} & J_{42} & J_{43} & J_{44} \end{bmatrix} \begin{bmatrix} \Delta \theta_1 \\ \Delta \theta_2 \\ \Delta \theta_3 \\ \Delta \theta_4 \end{bmatrix}; \quad \mathbf{0} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; \quad J_{ij} = \begin{bmatrix} -\sum_{i=1}^j L_i \sin(\sum_{g=1}^i \theta_g) \\ \sum_{i=1}^j L_i \cos(\sum_{g=1}^i \theta_g) \end{bmatrix} \quad (5)$$

Para o problema descrito neste artigo os valores ΔP são conhecidos através da realimentação visual e da pose de referência, é necessário calcular os valores de $\Delta \theta$ a serem aplicados aos atuadores, o problema recai num problema de cinemática inversa. A solução é inverter o Jacobiano para obter $\Delta \theta$ em função de ΔP . Considerando que o Jacobiano não é uma matriz quadrada, portanto devem ser usadas técnicas de cálculo para obter a matriz pseudo-inversa do Jacobiano. A resolução de Eq. 6 encerra o problema de controle cinemático.

$$\Delta \theta = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \Delta P \quad (6)$$

4 RESULTADOS

Esta seção apresenta os resultados do sistema de visão, descrevendo os experimentos realizados, num computador pessoal *PentiumIIITM* 800MHz e 128MB de memória RAM.

4.1 Calibração

O processo de calibração tem como resultado dez classes, caracterizadas por centros, que são representadas em componentes de cores do modelo RGB. Os centros são usadas para classificar os pixels da imagem adquirida, através da menor distância entre a classe e o pixel da imagem. A Tabela 1 apresenta os centros resultantes do processo de calibração.

Tabela 1: Resultado da Calibração.

Classe	Centros	Classe	Centros
vermelho	173 71 57	branco	190 192 189
magenta	212 134 150	preto	12 16 11
azul	70 111 177	espúrio	43 39 37
amarelo	225 185 114	espúrio	48 44 48
verde	118 174 110	espúrio	31 35 40

A Figura 5 apresenta a disposição espacial dos agrupamentos de uma imagem classificada, onde os eixos representam as cores primárias no modelo RGB, vermelho, verde e azul. Na Figura 5, cada agrupamento é representado pela cor formada por suas componentes RGB. O agrupamento preto é composto por quatro classes, a preta e as três classes de elementos espúrios.

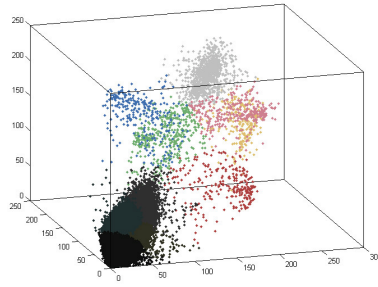


Figura 5: Agrupamentos.

4.2 Métodos de Varredura

O processo de localização usa dez classes resultantes da calibração para classificar os pixels da área de varredura. A varredura foi testada de maneira seqüencial e aleatória. No modo seqüencial, para o rótulo da base foi dado espaçamento de cinco pixels tanto na largura quanto na altura da imagem, para os rótulos restantes o espaçamento de cinco graus foi aplicado. O modo aleatório foi implementado e comparado ao modo seqüencial. A Figura 6 apresenta um dos resultados deste estudo comparativo.

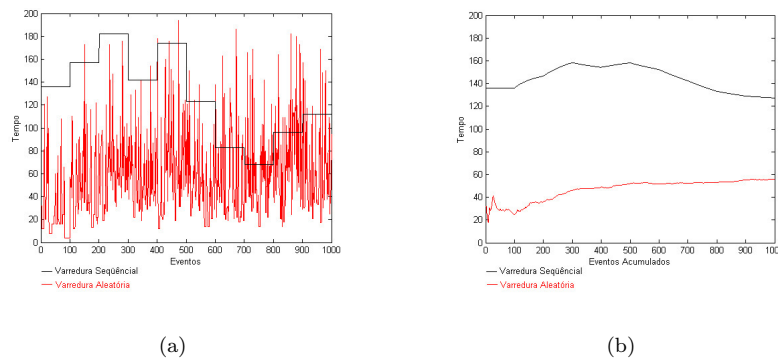


Figura 6: Estudo comparativo das abordagens de varredura.

O eixo das ordenadas, de ambos os gráficos, é uma representação temporal dada pela quantidade de *pixels* testados até encontrar o *pixel* da classe desejada. O eixo das abscissas, para o gráfico de Fig. 6(a), representa o número de eventos de localização realizados; para o gráfico de Fig. 6(b), representa o a média dos tempos até o evento atual. A linha vermelha caracteriza a varredura aleatória. Para obtenção dos gráficos de Fig. 6, foram realizados 1000 eventos de localização dos quatro rótulos variáveis, para dez poses diferentes. Para um evento isolado a comparação é inconclusiva, mas considerando a média dos eventos a varredura aleatória se mostra mais rápida que a seqüencial.

4.3 Cálculo do Centróide

O sistema de visão é responsável pela obtenção da posição das juntas, dentro do sistema de visão a posição é dada pelo cálculo do centróide. Para teste do cálculo do centróide, foi captada a imagem de um rótulo amarelo com seu centro marcado em preto. Foi delimitada uma área de

varredura retangular, como a mostrada em Fig. 7(a). Utilizando o modo de varredura aleatório. O cálculo do centróide foi realizado 10000 vezes, para cada vez foi realizada uma busca diferente. Foi obtido como resultado uma taxa de acerto de 56,24% e o maior erro observado foi de um pixel, Fig. 7(b).

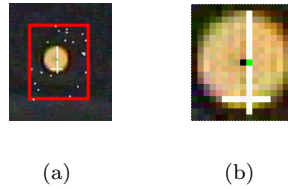


Figura 7: Centróide.

4.4 Localização

O processo de localização é a união da varredura e do cálculo do centróide. A base, por ser fixa, não é considerada no processo de localização para efeitos de análise baseado na velocidade de localização. O experimento repetiu a localização dos rótulos, excluindo o da base. Para um número de 1000 repetições o tempo médio gasto para localização foi de 2,02ms, para 10000 repetições foi de 2,14ms e para 100000 foi de 2,11ms. A Figura 8 ilustra o resultado de uma localização.

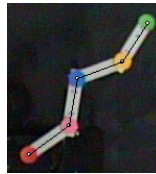


Figura 8: Resultado da Localização.

5 CONCLUSÕES

Os resultados obtidos com o sistema de visão apresentados na seção anterior são necessários para verificação de desempenho do sistema. O algoritmo *K-means* mostrou-se eficiente no processo de calibração conseguindo agrupar de forma eficiente as características principais da imagem adquirida. Para a localização, o modo de varredura aleatório se mostrou mais rápido para um grande número de iterações, portanto, ele foi o adotado para o sistema, e o cálculo do centróide mostrou boa precisão. O processo de localização leva em média 2,1 milissegundos, um resultado satisfatório considerando que uma imagem é adquirida a cada 33 milissegundos aproximadamente. O controle está sendo testado, mas o estudo comparativo com métodos clássicos não está concluído.

Referências

Aires, K. R. T.: 2001, *Desenvolvimento de um sistema de visão global para uma frota de mini-robôs móveis*, Master's thesis, Universidade Federal do Rio Grande do Norte.

- Chiaverini, S.: 1997, Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators, *IEEE Trans. On Robotics and Automation*. **13**(3).
- Corke, P. I.: 1996, Visual control of robots - high performance visual servoing, Research Studies Press LTD.
- Craig, J. J.: 1989, *Introduction to Robotics: mechanics and control*, 2nd edn, Addison-Wesley Publishing Company Inc.
- Gonzalez, R. C. and Woods, R. E.: 2000, *Processamento de Imagens Digitais*, 1st edn, Edgard Blücher Ltda.
- Haykin, S. S.: 2000, *Redes Neurais - Princípios e Prática*, 2 edn, Bookman Companhia ED.
- Hutchinson, S., Hager, G. D. and Corke, P. I.: 1996, A tutorial on visual servo control, *IEEE Transactions on Robotics and Automation* **12**(5).
- Jain, A. K.: 1989, *Fundamentals of Digital Image Processing*, 1 edn, Prentice Hall.
- Nakamura, Y.: 1991, *Advanced Robotics - Redundancy and Optimization*, Addison-Wesley Publishing Company Inc.
- Zha, H., Onitsuka, T. and Nagata, T.: 1996, A visuo-motor coordination algorithm for controlling arm's movements in environments with obstacles., Proc. Of the Fourth International Conference on Control, Automation, Robotics and Vision (ICARCV'96), Singapore.

REDUNDANT MANIPULATOR CONTROL USING VISUAL SERVOING

Anfranserai Morais Dias

Universidade Federal do Rio Grande do Norte, DCA - CT - UFRN,
Lagoa Nova, Natal - RN, Cep.: 59072-970
xamd@dac.ufrn.br

Pablo Javier Alsina

Universidade Federal do Rio Grande do Norte, DCA - CT - UFRN,
Lagoa Nova, Natal - RN, Cep.: 59072-970
pablo@dac.ufrn.br

Abstract. *In conventional robot manipulator control, the desired path is specified in cartesian space and converted to joint space through inverse kinematics mapping. The joint references generated by this mapping are utilized for dynamic control in joint space. Thus, the end-effector position is, in fact, controlled indirectly, in open-loop and the accuracy of grip position control directly depends on the accuracy of the available kinematic model. In this paper, a new scheme for redundant manipulator kinematic control, based on visual servoing is proposed. In the proposed system, a robot image acquired through a CCD camera is processed in order to compute the position and orientation of each link of the robot arm. The robot task is specified as a temporal sequence of reference images of the robot arm. Thus, both the measured pose and the reference pose are specified in the same image space, and its difference is utilized to generate a cartesian space error for kinematic control purposes. The proposed control scheme was applied in a four degree-of-freedom planar redundant robot arm, experimental results are shown.*

Index Terms: *Visual Servoing, Control, Redundant Manipulator, Kinematics.*