
DESENVOLVIMENTO DE ESQUEMA DE CONTROLE COM REALIMENTAÇÃO VISUAL PARA UM ROBÔ MANIPULADOR

Allan Aminadab André Freire Soares

Departamento de Engenharia de Computação e Automação - CT - UFRN - 59072-970 - Natal - RN – Brasil
aminadab@dca.ufrn.br

Pablo Javier Alsina

Departamento de Engenharia de Computação e Automação - CT - UFRN - 59072-970 - Natal - RN – Brasil
pablo@dca.ufrn.br

Adelardo Adelino Dantas de Medeiros

Departamento de Engenharia de Computação e Automação - CT - UFRN - 59072-970 - Natal - RN – Brasil
adelardo@dca.ufrn.br

Resumo Este artigo descreve um esquema baseado em técnicas de visão computacional para determinação da pose (posição e orientação) de um braço robótico a partir de uma imagem obtida através de câmera CCD. Para facilitar o cálculo da pose, rótulos triangulares coloridos são dispostos na garra do manipulador. Regras heurísticas eficientes são utilizadas para obter os vértices dos rótulos na imagem. Métodos numéricos são usados para recuperação da posição e orientação do robô a partir desses vértices. Um esquema de calibração de cores baseado em rede neural competitiva é implementado de modo a garantir a robustez do sistema de visão na presença de variações na iluminação. O esquema proposto será utilizado para fins de controle de um braço manipulador de cinco graus de liberdade como uma alternativa ao controle tradicional baseado em medição das posições de juntas através de codificadores óticos.

Palavras Chaves – braço manipulador, servo-visão, controle mão-olho.

Abstract: This paper describes a scheme for robot arm pose calculation based on computer vision techniques. Triangular colored labels are disposed on the robot grip. The label's vertices are extracted through efficient real time heuristic rules and numerical methods are adopted for reconstruction of the robot pose (position and orientation). A color calibration scheme based on competitive neural network is implemented to guarantee robustness in the presence of illumination changes. The proposed scheme will be applied to the position control of a five DOF manipulator as an alternative to classical manipulator's control that uses *encoder's* as joint position sensors.

Keywords – manipulator's arm, visual servoing, hand-eye control.

1 INTRODUÇÃO

Os métodos convencionais de controle de posição para braços robóticos são baseados na realimentação das posições das juntas medidas através de codificadores óticos. De acordo com esta abordagem, os ângulos de juntas são controlados visando atingir uma determinada pose no espaço cartesiano previamente especificada para a garra do manipulador. Esses métodos baseiam-se no pressuposto que, dada uma pose especificada no espaço cartesiano, os ângulos de juntas de referência correspondentes podem ser obtidos com precisão utilizando-se o modelo cinemático inverso do manipulador. Os métodos convencionais de controle dependem então da precisão deste modelo cinemático. Assim, a pose é controlada em malha aberta, de modo indireto, através do controle de juntas. Uma alternativa a este tipo de controle é utilizar a realimentação da posição e orientação da garra medida diretamente em espaço cartesiano. Esta medição pode ser realizada por meio de técnicas de visão computacional. Atualmente várias pesquisas estão sendo realizadas na área de controle robótico usando realimentação visual (Corke, 1996), (Wang, 1992), (Wei *et alii*, 1998), (Castaño e Hutchinson, 1994), (Popovic e Blechschmidt, 1996), (Sharma e Hutchinson, 1997), (Haralick, e Shapiro, 1993). etc.

Este trabalho propõe um método alternativo para cálculo da pose de um braço robótico. Regras heurísticas são aplicadas para extração, a partir da imagem do robô, dos vértices de rótulos triangulares dispostos na garra. Métodos numéricos são posteriormente utilizados para obtenção da pose a partir desses vértices. A seção 2 deste artigo descreve o problema estudado neste trabalho. A seção 3 contém as técnicas propostas para rotulagem e escolha de cores, calibração de cores, extração dos vértices dos rótulos triangulares na imagem, calibração da câmera, e cálculo da pose. Na seção 4, os resultados obtidos são analisados. As conclusões e perspectivas de trabalhos futuros são apresentadas na seção 5.

2 DESCRIÇÃO DO PROBLEMA

O objetivo de um esquema de controle para um manipulador robótico é posicionar e orientar sua garra de forma precisa no espaço de trabalho. O esquema de controle convencional em espaço de juntas é mostrado na figura 1.

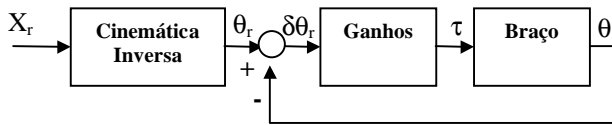


Figura 1. Controle convencional em espaço de juntas

Neste esquema, as referências angulares de junta (θ_r) são calculadas usando o modelo cinemático inverso a partir da pose de referência do manipulador (X_r) especificada em espaço cartesiano. As posições das juntas (θ) são medidas através de codificadores óticos e comparadas às referências de junta obtidas. O erro resultante é utilizado numa malha de controle que garante que as posições de junta convirjam para as referências de juntas calculadas. Entretanto, devido às imprecisões cinemáticas, isto não garante que a garra atinja efetivamente a pose final especificada.

Para contornar o problema introduzido pelas imprecisões do modelo cinemático, propõe-se um esquema baseado em realimentação visual para controle do braço diretamente em espaço cartesiano, conforme mostrado na figura 2.

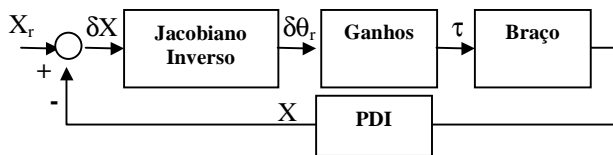


Figura 2. Controle baseado em realimentação visual.

Neste esquema, a pose do manipulador X é diretamente obtida a partir de uma imagem do braço usando técnicas de visão computacional e comparada à pose de referência. Um algoritmo de controle em espaço cartesiano pode então ser utilizado para minimizar o erro entre a pose medida e a pose desejada. As técnicas de controle em espaço cartesiano baseiam-se no jacobiano inverso do manipulador, o qual depende dos parâmetros cinemáticos. Apesar disto, a imprecisão destes parâmetros não impede que o controlador minimize o erro na pose, visto que, existindo um erro em espaço cartesiano, o mesmo será mapeado para o espaço de juntas e uma ação de controle será tomada de modo a reduzi-lo. Assim, mesmo uma representação grosseira do jacobiano é suficiente para garantir o mapeamento entre erros em espaço cartesian e espaço de juntas.

3 MÉTODO DE REALIMENTAÇÃO VISUAL

3.1 Rotulagem e Escolha das Cores

No presente trabalho, a pose do robô deve ser extraída a partir de imagens do mesmo capturadas por uma câmera CCD. Para facilitar esta tarefa, um cubo é fixado na garra do manipulador. Cada face deste cubo apresenta rótulos triangulares de diferentes cores sobre um fundo preto. Todos os triângulos apresentam uma faixa adjacente a um dos lados. A cor desta faixa é a mesma para todos os triângulos. Sua função é facilitar a identificação do lado a que os vértices pertencem. A técnica de rotulagem proposta é descrita na figura 3.

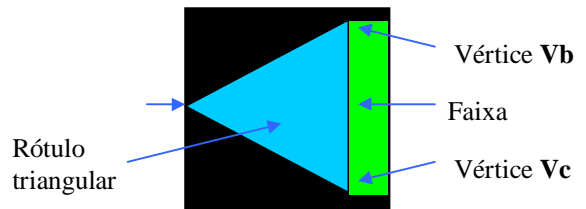


Figura 3. Rotulagem do cubo.

As cores dos rótulos foram escolhidas a partir da análise das cores que compõem o manipulador e seu espaço de trabalho através de uma rede neural artificial competitiva básica (Costa, 1999). O modelo de cores utilizado é o modelo RGB (González & Woods, 1992). Cada *pixel* da imagem é aplicado à rede neural na forma de um vetor (3 x 1) contendo os valores para seus parâmetros R, G e B. A partir desses vetores a rede determina o agrupamento dos *pixels* da imagem em um determinado número de classes previamente definido (classes de cores).

A análise de várias imagens fornece vários centros para cada uma dessas classes de cores. O valor médio dos centros obtidos para cada classe é tomado como o valor final que representa o centro da classe. A partir desses valores, escolheram-se os centros das classes de cores, e conseqüentemente as cores, que preenchem os rótulos. Os centros das classes de cores dos rótulos foram escolhidos de forma que, no modelo RGB, todos os centros para as classes de cores fossem os mais distantes possíveis, de modo a facilitar a segmentação da imagem. A classe de cor à qual pertence um determinado *pixel* é aquela cujo centro se encontra à menor distância euclidiana do valor RGB do mesmo.

3.2 Calibração de Cores

Devido a mudanças na luminosidade, os centros das classe de cores são sujeitos a variações correspondentes. Por isso, previamente ao controle, é necessário calibrar o sistema visando determinar os centros das classes de cores (robô, rótulos e espaço de trabalho) para as condições de iluminação atuais (Aires, 2001). O algoritmo de calibração de cores adotado usa uma rede neural artificial competitiva básica para analisar um conjunto de imagens captadas do manipulador em diferentes configurações. Essas configurações para o manipulador são determinadas previamente de modo que todos os rótulos triangulares fixados no cubo sobre a garra possam ser captados. Cada *pixel* de cada imagem é aplicado à entrada da rede neural na forma de um vetor (3 x 1) contendo os valores para seus parâmetros R, G e B. A partir desses vetores a rede determina o agrupamento dos *pixels* da imagem nas classes de cores definidas na etapa anterior (seção 3.1). O valor médio dos centros obtidos para cada classe de cor em cada imagem é tomado como o valor final que representa o centro da classe.

Foi utilizada uma primeira rede neural competitiva para fornecer o centro das classes de cores existentes no manipulador e no seu espaço de trabalho visando a escolha das cores dos rótulos. Esta escolha foi feita de maneira a maximizar a distância euclidiana entre todos os centros das classes de cores. Uma segunda rede neural competitiva, foi usada para a calibração de cores, fornecendo os centros de todas as classes de cores presentes durante a operação do robô. As classes obtidas desta maneira serão utilizadas para classificar os *pixels* em tempo real.

3.3 Extração dos Vértices do Triângulo

A projeção do rótulo triangular da face do cubo no plano da imagem corresponde a um triângulo de forma normalmente irregular. A imagem pode conter de um a três triângulos dependendo da orientação da garra. O algoritmo de extração dos vértices busca, por meio de regras heurísticas, determinar os vértices desses triângulos em referencial de imagem.

A seguinte heurística será usada para detectar os vértices do triângulo no plano da imagem, de maneira eficiente em tempo real, sem a necessidade de analisar todos os *pixels* da imagem, e processando apenas uma pequena fração dos *pixels* do triângulo:

1. Percorre-se a imagem em passos fixos quanto ao número de *pixels* até encontrar um *pixel P* cuja classe de cor corresponda a uma das classes de cores definidas para os rótulos triangulares. O passo de varredura é escolhido o maior possível (para minimizar o esforço computacional) de modo a que pelo menos quatro *pixels* do mesmo triângulo sejam analisados quando a garra se encontrar na sua posição mais afastada da câmera. Com isto, minimiza-se o risco de não detectar o triângulo.
2. A partir de **P**, percorre-se a imagem *pixel a pixel*, para cima, para baixo, para a direita e para a esquerda até alcançar as bordas do triângulo (pontos **Pc**, **Pb**, **Pd** e **Pe** respectivamente). Duas situações podem ocorrer, conforme mostrado na figura 4: a) dois dos pontos pertencem a um mesmo lado do triângulo e os outros dois estão em lados isolados, b) dois pontos pertencem a um mesmo lado do triângulo e os outros dois pertencem a um outro lado, ficando o terceiro lado sem nenhum ponto. Observação: para cada *pixel* analisado nesta fase calcula-se a média RGB da sua vizinhança (*pixels* 8-conectados), de modo a garantir uma maior robustez frente a ruídos no processo de classificação do *pixel* como pertencente ao rótulo ou não. Caso o valor obtido seja muito diferente ao valor do centro de cores do rótulo, analisam-se os dois *pixels* seguintes na direção de busca. Persistindo esta situação para os mesmos, o *pixel* original é classificado como pertencente à borda do triângulo e a busca nessa direção é concluída.

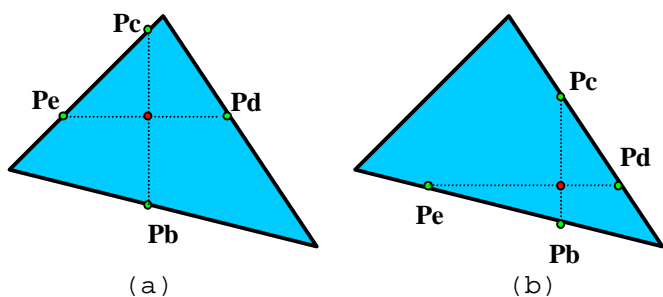


Figura 4. Detecção das bordas do triângulo.

a) Um lado com dois pontos. b) Dois lados com dois pontos.

3. Determinam-se os pontos que pertencem a um mesmo lado. Isto é feito tomando os pontos dois a dois, da seguinte forma: (**Pe**, **Pc**), (**Pe**, **Pb**), (**Pd**, **Pc**) e (**Pd**, **Pb**). Caso algum destes pares ordenados contenha pontos pertencentes a um mesmo lado, estes pontos não serão mais testados. Observação: os pontos são considerados como pertencentes a um mesmo lado se o ponto médio do segmento formado por eles for um ponto de borda.
4. Configurando-se o caso a), com apenas dois pontos pertencentes a um mesmo lado, é necessário determinar

mais um ponto auxiliar **Po** em um dos outros lados, redundando no caso b). Isto é feito varrendo o triângulo a partir do ponto **P**, na direção diagonal oposta ao lado já detectado, como mostrado na figura 5. A seguir, testa-se qual dos dois pontos isolados que resultaram do passo 3) está no mesmo lado do triângulo em que se encontra **Po**. Com isto, obtêm-se dois pares de pontos, os quais determinam as retas suporte, **L12** e **L13**, de dois lados do triângulo respectivamente.

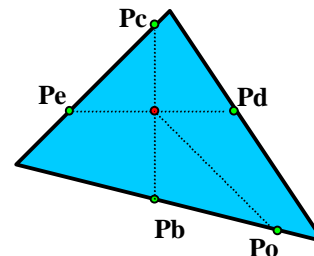


Figura 5. Obtenção do ponto auxiliar na diagonal oposta.

5. Determinar o vértice **V1** do triângulo contido na intersecção das duas retas suporte **L12** e **L13**.
6. Determinar para cada uma das retas suporte **L12** e **L13**, qual dentre os pontos detectados sobre cada uma delas é o mais distantes do vértice **V1**.
7. Determinar a inclinação da reta **Lb**, bissetriz do ângulo formado pelas retas suporte obtidas nos passos 3 e 4.
8. Determinar as equações de duas retas, **Lb'** e **Lb''**, cuja inclinação é igual à da bissetriz obtida no passo 7 e que passam respectivamente por cada um dos dois pontos obtidos no passo 6. A partir destes dois pontos, varrer o triângulo sobre estas duas retas no sentido oposto ao vértice **V1**. Continuar a varredura até atingir dois pontos de borda, **Po'** e **Po''**, os quais obrigatoriamente pertencem ao lado do triângulo oposto a esse mesmo vértice, conforme mostra a figura 6. Esta varredura pode ser realizada de maneira eficiente usando o algoritmo de Bresenham (Bresenham, 1965).
9. Determinar os vértices remanentes, **V2** e **V3**, nas intersecções das retas suporte **L12** e **L13** com a reta suporte **L23** que passa por **Po'** e **Po''** (Ver figura 6).

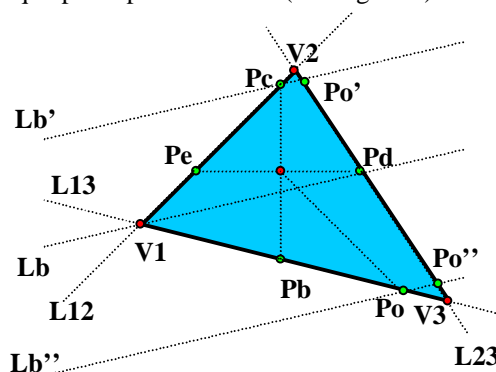


Figura 6. Obtenção dos vértices remanentes.

Após extraídos os vértices, é necessário determinar a que lado do triângulo eles pertencem. Para isto será utilizada a informação da faixa colorida adjacente a um dos lados. O vértice **Va** é aquele na intersecção dos lados não adjacentes à faixa. Os vértices remanentes, que denominaremos **V'** e **V''**, são identificados da seguinte forma: realiza-se o produto vetorial $(V''-Va) \times (V'-Va)$; caso o produto seja positivo, $V^2=Vb$ e $V''^2=Vc$, caso contrário, teremos $V^2=Vc$ e $V''^2=Vb$.

3.4 Calibração da Câmera

A calibração da câmera consiste na determinação da transformação de coordenadas do referencial cartesiano da base do robô para o referencial cartesiano do plano da imagem. Essa transformação permite que se obtenha a equivalência entre as dimensões encontradas no mundo real e suas correspondentes medidas no plano da imagem. Para realizar a calibração da câmera é utilizado o método apresentado por Schilling (Schilling,1990) que considera constante a orientação entre o referencial da base do robô e o referencial da câmera (X,Y,Z). Este, por sua vez, está perfeitamente alinhado com o referencial do plano da imagem (x,y,z). Este método requer o conhecimento de dois pontos fixos (**A** e **B**) do robô ou do seu espaço de trabalho cujas coordenadas cartesianas em relação ao referencial da base são conhecidas. No nosso caso, os dois pontos fixos correspondem aos vértices **V_b** e **V_c** do rótulo triangular na face do cubo paralela à garra quando o robô se encontra na sua posição de repouso (*home-position*). A câmera será disposta perpendicularmente à base do robô.

Denominando (x_0, y_0, z_0) como a posição da origem do referencial da base do robô expresso no referencial da câmera e λ como a distância focal, a transformação do referencial da base do robô para o referencial de imagem pode ser obtida em função das transformações câmera-imagem, $\mathbf{T}_{c\grave{a}m\acute{e}r\acute{a}}^{i\acute{m}a\acute{g}e\acute{m}}$, e base-c\grave{a}m\acute{e}r\acute{a}, $\mathbf{T}_{b\acute{a}s\acute{e}}^{c\acute{a}m\acute{e}r\acute{a}}$, (Gonz\’ales & Woods, 1992):

$$\begin{aligned} \mathbf{T}_{b\acute{a}s\acute{e}}^{i\acute{m}a\acute{g}e\acute{m}} &= \mathbf{T}_{c\grave{a}m\acute{e}r\acute{a}}^{i\acute{m}a\acute{g}e\acute{m}} \cdot \mathbf{T}_{b\acute{a}s\acute{e}}^{c\acute{a}m\acute{e}r\acute{a}} \Rightarrow \\ \Rightarrow \mathbf{T}_{b\acute{a}s\acute{e}}^{i\acute{m}a\acute{g}e\acute{m}} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{-1}{\lambda} & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 0 & x_0 \\ 0 & 0 & 1 & y_0 \\ 1 & 0 & 0 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \\ \Rightarrow \mathbf{T}_{b\acute{a}s\acute{e}}^{i\acute{m}a\acute{g}e\acute{m}} &= \begin{bmatrix} 0 & 1 & 0 & x_0 \\ 0 & 0 & 1 & y_0 \\ 1 & 0 & 0 & 0 \\ \frac{-1}{\lambda} & 0 & 0 & \frac{(\lambda - z_0)}{\lambda} \end{bmatrix} \quad (1) \end{aligned}$$

Na matriz de transformação $\mathbf{T}_{b\acute{a}s\acute{e}}^{i\acute{m}a\acute{g}e\acute{m}}$ apenas os valores para x_0 , y_0 e z_0 s\’ao desconhecidos. O processo de calibração consiste na determina\’cao destes valores. No m\’etodo apresentado por Schilling (Schilling, 1990) os pontos previamente conhecidos no referencial da base do rob\’o t\’em suas coordenadas cartesianas dadas por **A**(A_1, A_2, A_3) e **B**(B_1, B_2, B_3). As proje\’c\’oes desses pontos no referencial da imagem s\’ao respectivamente **a**(a_1, a_2, a_3) e **b**(b_1, b_2, b_3). Sabendo-se disso monta-se a partir da matriz $\mathbf{T}_{b\acute{a}s\acute{e}}^{i\acute{m}a\acute{g}e\acute{m}}$ e dos vetores **A**, **B**, **a** e **b** um sistema de equa\’c\’oes cuja solu\’c\’ao fornece as express\’oes para o c\’alculo de x_0 , y_0 e z_0 :

$$\begin{aligned} x^0 &= [\lambda(\lambda(a^1 - A^1)(a^2 - b^2) - a^1(A^1(a^2 - b^2) + (a^1 - A^1 - b^1 + B^1))) + a^1 A^1 + a^1 b^1 B^1] / \lambda(a^1 - b^1) \quad (2) \\ y^0 &= [\lambda(b^2(a^2 - A^2) - a^2(b^2 - B^2)) - a^2 b^2 A^2 - a^2 b^2 B^2] / \lambda(b^2 - a^2) \\ z^0 &= [\lambda(a^1 - A^1 - b^1 + B^1) - a^1 A^1 - b^1 B^1] / (a^1 - b^1) \end{aligned}$$

3.5 Determina\’c\’ao da Pose

O comprimento dos lados dos r\’otulos triangulares nas faces do cubo \’e igual a $\Delta l = \Delta l_{ab} = \Delta l_{bc} = \Delta l_{ca}$.

As coordenadas dos v\’ertices do tri\’angulo no referencial do plano da imagem e no referencial da c\’am\’era (desconhecidas) s\’ao dadas respectivamente por: obtidas na subsec\’\’ao anterior s\’ao dadas por:

$$\mathbf{v}_i = [x_i \quad y_i \quad 0]^T \quad (3)$$

$$\mathbf{v}_i = [X_i \quad Y_i \quad Z_i]^T \quad (4)$$

onde $i = a, b, c$. A rela\’c\’ao entre as coordenadas dos v\’ertices nos diferentes referenciais apresentados acima \’e fornecida pela transforma\’c\’ao de perspectiva (Gonz\’ales & Woods, 1992):

$$\begin{cases} X_i = \frac{x_i}{\lambda}(\lambda - Z_i) & Y_i = \frac{y_i}{\lambda}(\lambda - Z_i) \end{cases} \quad (4)$$

O comprimento dos lados pode ser expresso da seguinte forma:

$$\left\{ \Delta l_{ij} = \left[(\Delta X_{ij})^2 + (\Delta Y_{ij})^2 + (\Delta Z_{ij})^2 \right]^{1/2} = \Delta l \quad (5)$$

onde $(i, j) = (a, b), (b, c), (c, a)$, e

$$\begin{cases} \Delta X_{ij} = X_i - X_j \\ \Delta Y_{ij} = Y_i - Y_j \\ \Delta Z_{ij} = Z_i - Z_j \end{cases} \quad (6)$$

Das express\’oes em (5) monta-se o seguinte sistema de equa\’c\’oes quadr\’aticas em Z_i ($i = a, b, c$):

$$\begin{cases} (x_i^2 + y_i^2 + \lambda^2)Z_i^2 + (x_j^2 + y_j^2 + \lambda^2)Z_j^2 + 2\lambda(x_i x_j + y_i y_j - x_i^2 - y_i^2)Z_i + \\ + 2\lambda(x_i x_j + y_i y_j - x_j^2 - y_j^2)Z_j - 2(x_i x_j + y_i y_j + \lambda^2)Z_i Z_j = \\ = \lambda^2(\Delta l^2 + 2(x_i x_j + y_i y_j) - x_i^2 - y_i^2 - x_j^2 - y_j^2) \end{cases} \quad (7)$$

onde $(i, j) = (a, b), (b, c), (c, a)$. Assim, temos um sistema de tr\’es equa\’c\’oes n\’ao lineares e tr\’es inc\’ognitas, Z_a, Z_b e Z_c , o qual pode ser resolvido utilizando m\’etodos num\’ericos, tal como o m\’etodo Newton-Raphson, (Ruggiero, 1996). Como condi\’c\’ao inicial para o m\’etodo num\’erico, utilizam-se os valores de Z_a, Z_b e Z_c obtidos no per\’iodo de amostragem anterior, de modo a permitir uma r\’apida converg\’encia do m\’etodo num\’erico. Substituindo os valores obtidos para Z_a, Z_b e Z_c nas equa\’c\’oes de perspectiva (4) determina-se os valores para as demais coordenadas (X_a, X_b, X_c) e (Y_a, Y_b, Y_c).

Determina-se dessa forma os valores para as coordenadas dos v\’ertices de um dos r\’otulos triangulares em rela\’c\’ao ao referencial da c\’am\’era: $\mathbf{v}_a, \mathbf{v}_b$ e \mathbf{v}_c . Essas coordenadas podem ser descritas no referencial da base do rob\’o multiplicando-se cada uma delas pela matriz de transforma\’c\’ao $\mathbf{T}_{c\grave{a}m\acute{e}r\acute{a}}^{b\acute{a}s\acute{e}}$. Essa matriz corresponde a inversa da matriz de transforma\’c\’ao $\mathbf{T}_{b\acute{a}s\acute{e}}^{c\acute{a}m\acute{e}r\acute{a}}$ determinada na subsec\’\’ao 3.4.

$$\mathbf{V}_i^b = \mathbf{T}_{c\grave{a}m\acute{e}r\acute{a}}^{b\acute{a}s\acute{e}} \cdot \mathbf{v}_i \quad (8)$$

onde $i = a, b, c$ e \mathbf{V}_i^b s\’ao as coordenadas do v\’ertice em referencial de base. Com as coordenadas de um v\’ertice conhecidas em rela\’c\’ao ao referencial da base do rob\’o e em rela\’c\’ao ao referencial da garra, (\mathbf{V}_i^g), estas \’ultimas fixas e conhecidas, pode-se calcular a posi\’c\’ao da garra em rela\’c\’ao referencial da base, ${}^b \mathbf{P}_g$, da seguinte forma:

$${}^b \mathbf{P}_g = \mathbf{V}_i^b - \mathbf{V}_i^g \quad (9)$$

As coordenadas dos vértices dos rótulos triangulares no referencial da garra são invariáveis ao deslocamento da mesma e previamente conhecidas. A partir de cada rótulo triangular, podem ser obtidos três vetores com a mesma direção dos três vetores unitários dos eixos coordenados da referencial da garra. O primeiro deles é obtido diretamente da diferença $V_b - V_c$. O segundo, a partir da mediatriz do lado adjacente à faixa colorida, ou seja $V_a - (V_b + V_c)/2$. O terceiro pode ser obtido pelo produto vetorial dos dois primeiros. Estes vetores compõem a matriz de orientação do referencial da garra em relação ao referencial da base do robô (Craig, 1989). A partir dessa matriz obtêm-se os ângulos de rolamento (*roll*), arfagem (*pitch*) e guinada (*yaw*) (Yoshikawa, 1990) que representam a orientação da garra em relação ao referencial da base do robô.

Obtidas a posição e orientação da garra estas são comparadas com seus respectivos valores de referência, gerando os erros de posição e orientação (ΔP , ΔO):

$$\Delta P = [\text{POSIÇÃO DE REFERENCIADA GARRA} - \text{POSIÇÃO DA GARRA}]_{3 \times 1} \quad (10)$$

$$\Delta O = [\text{ORIENTAÇÃO DE REFERENCIADA GARRA} - \text{ORIENTAÇÃO DA GARRA}]_{3 \times 1}$$

Esses erros são agrupados na forma de um único vetor e aplicados ao Jacobiano inverso do robô produzindo os incrementos angulares necessários a cada junta para que o manipulador alcance a posição e orientação final desejada (Craig, 1989):

$$[\delta \theta]_{5 \times 1} = [J(\theta)^{-1}]_{5 \times 6} \cdot \begin{bmatrix} \Delta P \\ \Delta O \end{bmatrix}_{6 \times 1} \quad (11)$$

4 RESULTADOS OBTIDOS

Na figura 8 mostra-se um dos resultados obtidos na análise dos valores RGB de uma imagem capturada do robô (figura 7). Foram estabelecidas três classes de cores para os *pixels* da imagem: amarelo (robô), preto (fundo) e uma classe destinada a classificar *pixels* espúrios. Foi utilizada uma rede neural competitiva básica com três neurônios, cujos pesos correspondem aos centros das classes de cores. Como entrada utilizou-se um conjunto de dezesseis mil *pixels* escolhidos aleatoriamente na imagem. Nos vários testes realizados com a imagem da figura 7, a rede convergiu em, no máximo, sete iterações.

Como as cores presentes na imagem se encontram na região central do cubo RGB, as cores dos rótulos foram escolhidas nos vértices do cubo, de modo a garantir o maior afastamento: vermelho (1,0,0), verde (0,1,0), azul (0,0,1), magenta (1,1,0), ciano (0,1,1) e, para a faixa lateral, branco (1,1,1).

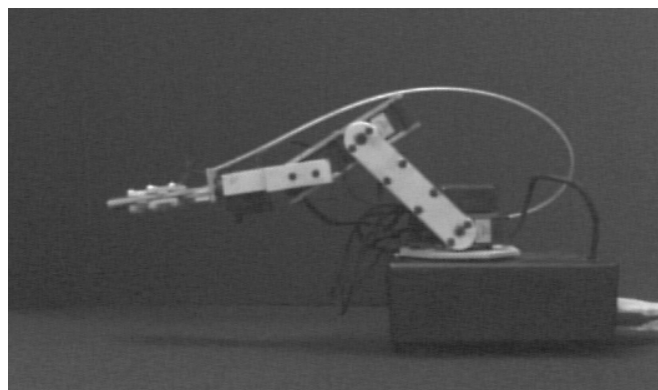


Figura 7. Imagem capturada para análise de cores

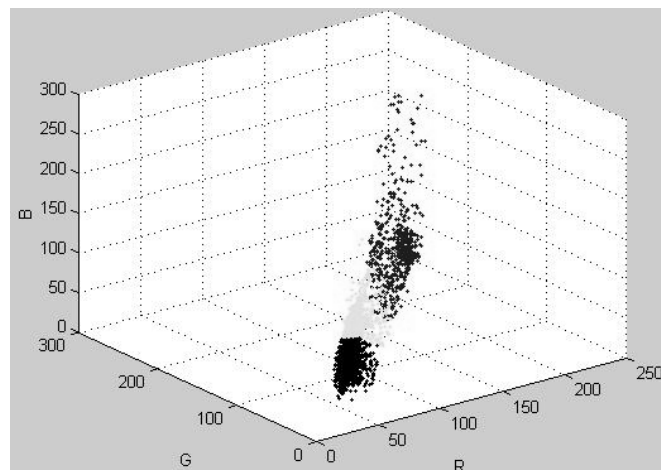


Figura 8. Agrupamento em classe de cores.

5 CONCLUSÃO

Neste artigo foi proposto um esquema para cálculo da pose de um braço manipulador a partir de técnicas de visão computacional para fins de controle baseado em realimentação visual. Um algoritmo baseado em heurísticas foi proposto para o cálculo em tempo real da posição e orientação do robô a partir de imagens captadas do mesmo. Uma vantagem do sistema proposto é que o cálculo da pose do manipulador é realizado usando apenas uma única câmera.

A utilização da rotina desenvolvida para implementação da rede neural artificial competitiva básica demonstrou que as cores do robô e do seu espaço de trabalho podem ser agrupadas em três classes distintas permitindo desta forma a escolha segura das cores que preenchem os rótulos.

Atualmente, as rotinas relacionadas à extração dos vértices dos rótulos e do cálculo da pose estão em fase de implementação. Após esta etapa, o algoritmo proposto para extração dos vértices será comparado, em termos de desempenho e precisão com outro algoritmo baseado em transformada de Hough (Jain, 1989). O sistema de controle baseado em realimentação visual será comparado, em termos de precisão, com um sistema de controle convencional baseado em modelo cinemático.

6 REFERÊNCIAS BIBLIOGRÁFICAS

- Aires, K. R. T. (2001). Desenvolvimento de um Sistema de Visão Global para uma Frota de Mini-Robôs Móveis. Dissertação de Mestrado, UFRN, Natal – RN.
- Bresenham, J. E. (1965). Algorithm for computer control of Digital Plotter. *IBM System Journal* (Jan.), Vol. 4, pp. 25-30.
- Castaño, A. & S. Hutchinson (1994). Visual Compliance: Task-Directed Visual Servo Control. *IEEE Transactions on Robotics and Automation* (Jun.), Vol. 10, pp. 334-342.
- Corke, P. I., (1996). *Visual Control of Robots: High Performance Visual Servoing*. Research Studies Press Ltd.
- Costa, J. A. F. (1999). Classificação Automática e Análise de Dados por Redes Neurais Auto-Organizáveis. Tese de Doutorado, UNICAMP, Campinas - SP.

- Craig, John J. (1989). *Introduction to Robotics: Mechanics and Control*. 2ª Edição, Addison-Wesley Publishing Company.
- Gonzáles, R. C. & Woods, R. E. (1992). *Digital Image Processing*. Addison-Wesley Publishing Company.
- Haralick, R. M. & Shapiro, L. G. (1993). *Computer and Robot Vision*. Volume 2, Addison-Wesley Publishing Company.
- Jain, A. K. (1989). *Digital Image Processing*. Prentice Hall.
- Popovic, D. & R. Blechschmidt (1996). Hand-Eye Coordination of a Robot System Using Neural Networks. *The Fourth International Conference on Control, Automation, Robotics and Vision*, Singapore.
- Ruggiero, M. A. & V. L. R. Lopes (1996). *Cálculo Numérico*. 2ª Edição, Makron Books do Brasil.
- Schilling, R. J. (1990). *Fundamentals of Robotics: Analysis and Control*. Prentice-Hall.
- Sharma, R. & S. Hutchinson (1997). Motion Perceability and its Application to Active Vision-Based Servo Control. *IEEE Transactions on Robotics and Automation* (Aug.). Vol. 13, pp. 607-617.
- Wang, Ching-Cheng (1992). Extrinsic Calibration of a Vision Sensor Mounted on a Robot. *IEEE Transactions on Robotics and Automation* (Apr.). Vol. 8, pp. 161-175.
- Wei, Guo-Qing, K. Arbter & G. Hirzinger (1998). Active Self-Calibration of Robotics Eyes and Hand-Eye Relationships with Model Identification. *IEEE Transactions on Robotics and Automation* (Feb.). Vol. 14, pp. 158-166.
- Yoshikawa, Tsuneo (1990). *Foundations of Robotic: Analysis and Control*. The MIT Press.