

Maciej STANIAK, Cezary ZIELIŃSKI

POLITECHNIKA WARSZAWSKA, WYDZIAŁ ELEKTRONIKI I TECHNIK INFORMACYJNYCH,
INSTYTUT AUTOMATYKI I INFORMATYKI STOSOWANEJ

Serwomechanizmy wizyjne - część 2

Mgr inż. Maciej STANIAK

Doktorant. Absolwent Wydziału Elektroniki i Techniki Informatycznych Politechniki Warszawskiej w 2002 r.



e-mail: M.Staniak@elka.pw.edu.pl

Prof. nzw. dr hab. inż. Cezary ZIELIŃSKI

Kierownik Zespołu Programowania Robotów i Systemów Rozpoznających, Instytutu Automatyki i Informatyki Stosowanej, Wydziału Elektroniki i Techniki Informatycznych, Politechniki Warszawskiej.



e-mail: C.Zielinski@ia.pw.edu.pl

Streszczenie

Obecnie uwaga naukowców zajmujących się robotyką koncentruje się na robotach usługowych oraz terenowych. W obu przypadkach do sprawnego wykonywania zadań roboty te potrzebują mnogości czujników, ale podstawowe stanowią kamery. Są one szczególnie istotne przy chwytaniu. Do realizacji tego zadania trzeba skonstruować serwomechanizm wizyjny. W pierwszej części tego artykułu wprowadzono podstawowe pojęcia niezbędne do analizy struktur serwomechanizmów wizyjnych. Ta część przedstawia poszczególne struktury wraz z analizą ich wad i zalet.

Słowa kluczowe: serwomechanizmy wizyjne, sterowanie robotów

Visual servo control - part 2

Abstract

The first part of this paper, published in PAK 5/2006, was devoted to basic concepts used in the analysis of visual-servo control and the classification of the resulting structures of controllers. This part gives a detailed description of each structure. Moreover, it presents the advantages and drawbacks of each of those structures, especially from the point of view of calibration (identification of parameter values) of the kinematic model of the manipulator and the relative location of the camera and the robot.

Keywords: visual servoing, robot control

1. Wstęp

Ta część artykułu bazuje na pojęciach i oznaczeniach wprowadzonych w części pierwszej, która ukazała się w PAK nr 5 z 2006 roku. Tam też dokonano klasyfikacji serwomechanizmów wizyjnych z punktu widzenia trzech kryteriów: przestrzeni, w której wyznaczany jest uchyb, miejsca zamocowania kamery oraz tego, czy w trakcie ruchu obserwowana jest końcówka efektora. W tej części zostaną przedstawione struktury sterowania według powyższej klasyfikacji oraz zostaną przeanalizowane ich wady i zalety - szczególnie z punktu widzenia kalibracji poszczególnych podsystemów układów sterowania.

2. Serwomechanizmy wizyjne wyznaczające uchyb w przestrzeni zadaniowej (PB)

2.1. Kamera nieruchoma, nie obserwująca końcówki (PB-EOL-SAC)

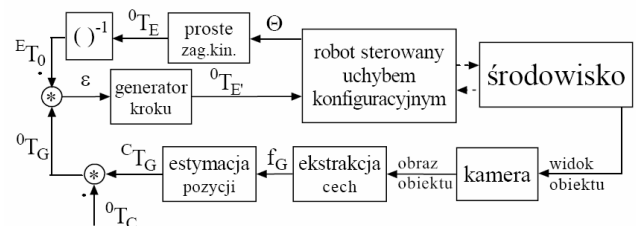
Dla układu z nieruchomą kamerą, pozycję chwytanego obiektu względem globalnego układu odniesienia 0T_G można wyznaczyć na podstawie uzyskanego obrazu:

$${}^0T_G = {}^0T_C {}^C T_G(f_G) \quad (1)$$

a uchyb zadaniowy ε pomiędzy pozycją tego obiektu a pozycją końcówki robota jako:

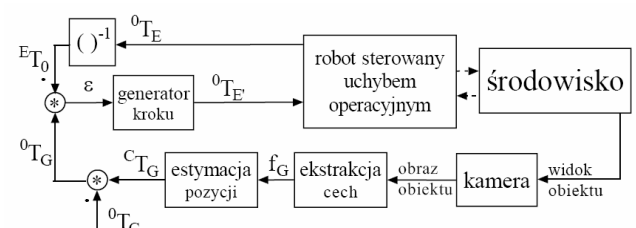
$$\varepsilon = {}^E T_G = {}^E T_0(\Theta) {}^0 T_G = {}^E T_0(\Theta) {}^C T_G(f_G) \quad (2)$$

Zazwyczaj uchyb ε jest duży, bo obiekt znajduje się w znacznej odległości od chwytaka, a więc uchyb ten musi być podzielony na mniejsze części poprzez generator kroku. Generator kroku na podstawie uchybu $\varepsilon ({}^E T_G)$ wyznacza położenie zadane końcówki ${}^0 T_E$ dla danej iteracji algorytmu regulacji. Zamiast niwelowania dużego uchybu w jednym kroku, realizowana jest jedynie jego część ${}^E T_E$ (czyli uchyb operacyjny), która pomnożona przez ${}^0 T_E$ daje ${}^0 T_G$. Cząstkowe zadane położenia końcówki ${}^0 T_E$ wykorzystane są jako wartości zadane zarówno przez regulator z uchybem konfiguracyjnym (rys. 1, przykładowa implementacja [5]), jak i z uchybem operacyjnym (rys. 2).



Rys. 1. Serwomechanizm wizyjny nie obserwujący końcówki efektora, z robotem sterowanym uchybem konfiguracyjnym, z nieruchomą kamerą. Ze względu na nieprzemienność mnożenia macierzy, kropka przy znaku mnożenia określa, która macierz jest pierwsza w kolejności. Oznaczenie to jest stosowane również na kolejnych schematach

Fig. 1. Endpoint open loop, configuration (manipulator joint) error control using a stand-alone camera. Since matrix multiplication is not commutative the dot next to the multiplication symbol specifies the matrix which is the first argument of the operation. The symbol is used in the subsequent diagrams



Rys. 2. Serwomechanizm wizyjny nie obserwujący końcówki efektora, z robotem sterowanym uchybem operacyjnym, z nieruchomą kamerą

Fig. 2. Endpoint open loop, operational (task) error control using a stand-alone camera

W przypadku serwomechanizmów typu PB-EOL-SAC zarówno kalibracja kamera-robot (0T_C), jak i kalibracja samego robota (${}^0T_E(\Theta) = {}^E T_0^{-1}(\Theta)$) jest istotna dla działania serwomechanizmu [5]. Jeśli kalibracja kamera-robot jest wadliwa, a więc pozycja kamery względem globalnego układu odniesienia będzie wyznaczona błędnie, to pozycja obiektu zostanie źle oceniona w globalnym układzie odniesienia, a w konsekwencji robot (nawet dobrze skalibrowany, a więc z dobrze wyznaczoną macierzą 0T_E) sięgnie w złe miejsce i nie złapie obiektu. Gdy kalibracja modelu kinematycznego robota 0T_E jest niepoprawna, to mimo dobrze wyznaczonego położenia obiektu względem kamery ${}^C T_G$, robot również sięgnie w niewłaściwe miejsce.

2.2. Kamera nieruchoma, obserwująca końcówkę (PB-ECL-SAC)

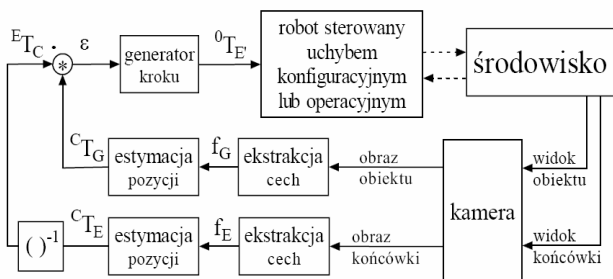
Jeśli obserwowana jest zarówno końcówka jak i obiekt, pozycja obiektu względem globalnego układu odniesienia 0T_G może być wyznaczona w następujący sposób:

$${}^0T_G = {}^0T_E(\Theta) {}^E T_C(f_E) {}^C T_G(f_G) = {}^0T_E(\Theta) {}^C T_E^{-1}(f_E) {}^C T_G(f_G) \quad (3)$$

gdzie macierze ${}^E T_C$ oraz ${}^C T_G$ wyznaczone są na podstawie cech obrazu końcówki i obiektu, ale uchyb zadaniowy ε zależy jedynie od:

$$\varepsilon = {}^E T_G = {}^E T_C(f_E) {}^C T_G(f_G) = {}^C T_E^{-1}(f_E) {}^C T_G(f_G) \quad (4)$$

W tym przypadku ogólne schematy układów regulacji zarówno dla robota sterowanego uchybem konfiguracyjnym jak i operacyjnym są identyczne (rys. 3).



Rys. 3. Serwomechanizm wizyjny obserwujący końcówkę efektora, z robotem sterowanym uchybem konfiguracyjnym lub operacyjnym, z nieruchomą kamerą

Fig. 3. Endpoint closed loop, configuration or operational error control using a stand-alone camera

Ponieważ powyższe równania nie zawierają macierzy transformacji 0T_C , nie trzeba wyznaczać położenia kamery względem robota 0T_C , a więc rozważany serwomechanizm wizyjny jest niewrażliwy na kalibrację kamera-robot. Wprawdzie obliczony uchyb nie jest zależny od kalibracji robota, ale ruch robota wyznaczony na podstawie tego uchybu jest zależny od tej kalibracji, gdyż wymaga rozwiązania odwrotnego zadania kinematyki w przypadku sterowania z uchybem konfiguracyjnym, bądź jacobianu robota dla uchybu operacyjnego.

2.3. Kamera na końcówce (PB-EIH)

Jeśli kamera zamontowana jest na końcówce robota, pozycja obiektu względem globalnego układu odniesienia 0T_G może być wyznaczona w następujący sposób:

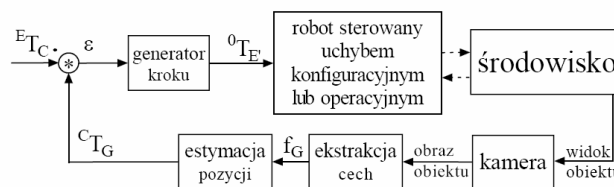
$${}^0T_G = {}^0T_E(\Theta) {}^E T_C {}^C T_G(f_G) \quad (5)$$

a uchyb ε jako:

$$\varepsilon = {}^E T_G = {}^E T_C {}^C T_G(f_G) \quad (6)$$

Jeśli końcówka nie jest obserwowana (PB-EOL), to mimo, że powyższe równania są prawie identyczne z (3) i (4), to macierz ${}^E T_C$ ma inny charakter. Transformacja ta nie powstaje w wyniku rozpoznania końcówki na obrazie. Jest ona wyznaczona na podstawie kalibracji kamery względem tejże końcówki (dlatego mamy ${}^E T_C$, a nie ${}^E T_C(f_E)$). Kalibracja kamera-kończówka jest więc niezbędna.

Tutaj ponownie ogólne struktury układów regulacji z uchybem konfiguracyjnym i operacyjnym są takie same (rys. 4). Dokładniejszy opis sterowania z wykorzystaniem jacobianu można znaleźć w pracy [11].



Rys. 4. Serwomechanizm wizyjny nie obserwujący końcówki efektora, z robotem sterowanym uchybem konfiguracyjnym lub operacyjnym, z kamerą na końcówce

Fig. 4. Endpoint open loop, configuration or operational error control using an eye-in-hand setup

W przypadku, gdy kamera obserwuje końcówkę (PB-EOL), transformację kamera-kończówka (w powyższych wzorach) możemy zapisać symbolicznie ${}^E T_C(f_E)$, gdzie $f_E = \text{const}$, gdyż jest wyznaczona w wyniku rozpoznania końcówki na obrazie, względem którego końcówka jest nieruchoma. Może to być zrobione raz, gdyż położenie końcówki na obrazie się nie zmienia, a więc ${}^E T_C$ jest stałe, lub wiele razy by wyeliminować szum pomiarowy związany z akwizycją i rozpoznaniem obrazu. Ponieważ robot sterowany jest przyrostowo w oparciu o obliczenie ${}^E T_G$, rozważania z poprzedniego podpunktu dotyczące kalibracji robota dotyczą również tego przypadku.

W przypadku kamery umieszczonej na końcówce należy tak sterować, aby obiekt znajdował się cały czas w polu widzenia kamery. Istnieją jednak algorytmy przewidujące, kiedy na skutek sterowania końcówką (wraz z kamerą na niej) obserwowany obiekt (lub jego część) może się znaleźć poza polem widzenia kamery [4].

3. Serwomechanizmy wizyjne wyznaczające uchyb w przestrzeni cech obrazu (IB)

3.1. Kamera nieruchoma, obserwująca końcówkę (IB-ECL-SAC)

Ruch końcówki zmienia cechy obrazu f według wzoru:

$$\dot{f}_E = {}^f J_{C,E} ({}^C r_E) \dot{r}_E \quad (7)$$

gdzie ${}^C r_E$ jest wektorem prędkości końcówki względem kamery, a ${}^f J_{C,E} ({}^C r_E)$ jest jacobianem obrazu.

Związek między prędkościami liniowymi i -tego punktu na końcówce $[{}^C \dot{x}_E, {}^C \dot{y}_E, {}^C \dot{z}_E]^T$ wyznaczonymi względem kamery a prędkościami liniowymi $[{}^c \dot{x}_E, {}^c \dot{y}_E, {}^c \dot{z}_E]^T$ i kątowymi $[{}^c \dot{\alpha}_E, {}^c \dot{\beta}_E, {}^c \dot{\gamma}_E]^T$ układu odniesienia związanego z końcówką względem kamery:

$$\begin{bmatrix} {}^c \dot{x}_{E_i} \\ {}^c \dot{y}_{E_i} \\ {}^c \dot{z}_{E_i} \end{bmatrix} = \begin{bmatrix} {}^c \dot{\alpha}_E \\ {}^c \dot{\beta}_E \\ {}^c \dot{\gamma}_E \end{bmatrix} \times \begin{bmatrix} {}^c x_{E_i} \\ {}^c y_{E_i} \\ {}^c z_{E_i} \end{bmatrix} + \begin{bmatrix} {}^c \dot{x}_E \\ {}^c \dot{y}_E \\ {}^c \dot{z}_E \end{bmatrix} \quad (8)$$

Powyższe równanie można zapisać w postaci układu równań:

$${}^c \dot{x}_{E_i} = {}^c z_{E_i} {}^c \dot{\beta}_E - {}^c y_{E_i} {}^c \dot{\gamma}_E + {}^c \dot{x}_E \quad (9)$$

$${}^c \dot{y}_{E_i} = {}^c x_{E_i} {}^c \dot{\gamma}_E - {}^c z_{E_i} {}^c \dot{\alpha}_E + {}^c \dot{y}_E \quad (10)$$

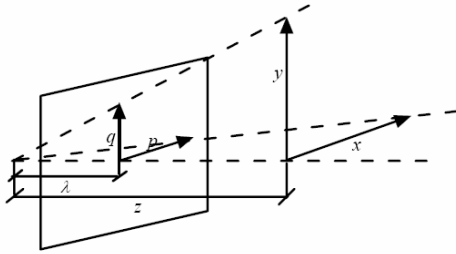
$${}^c \dot{z}_{E_i} = {}^c y_{E_i} {}^c \dot{\alpha}_E - {}^c x_{E_i} {}^c \dot{\beta}_E + {}^c \dot{z}_E \quad (11)$$

Zależność pomiędzy położeniem i -tego punktu końcówki w przestrzeni zadaniowej względem kamery $[{}^c x_{E_i}, {}^c y_{E_i}, {}^c z_{E_i}]^T$, a położeniem i -tego punktu końcówki w przestrzeni cech obrazu $[p_i, q_i]^T$ opisują równania (rys. 5):

$$p_i = \lambda \frac{{}^c x_{E_i}}{{}^c z_{E_i}} \quad (12)$$

$$q_i = \lambda \frac{{}^c y_{E_i}}{{}^c z_{E_i}} \quad (13)$$

gdzie λ jest ogniskową soczewki.



Rys. 5. Rzut perspektywiczny wektorów położenia obiektu w przestrzeni zadaniowej na płaszczyznę obrazu

Fig. 5. Perspective projection of object position vectors from task space onto the image plane

Prędkość położenia cech obrazu otrzymujemy różniczkując (12) i (13) w układzie kamery:

$$\begin{bmatrix} \dot{p}_i \\ \dot{q}_i \end{bmatrix} = \begin{bmatrix} \lambda \frac{{}^c z_{E_i} {}^c \dot{x}_{E_i} - {}^c x_{E_i} {}^c \dot{z}_{E_i}}{{}^c z_{E_i}^2} \\ \lambda \frac{{}^c z_{E_i} {}^c \dot{y}_{E_i} - {}^c y_{E_i} {}^c \dot{z}_{E_i}}{{}^c z_{E_i}^2} \end{bmatrix} \quad (14)$$

Po podstawieniu za ${}^c \dot{x}_{E_i}$, ${}^c \dot{y}_{E_i}$ i ${}^c \dot{z}_{E_i}$ równań (9)-(11) oraz korzystając z faktu, że $z_{E_i} \approx z_E$ otrzymujemy:

$$\begin{bmatrix} \dot{p}_i \\ \dot{q}_i \end{bmatrix} = \begin{bmatrix} \frac{\lambda}{{}^c z_E} & 0 & -\frac{p_i}{{}^c z_E} & -\frac{p_i q_i}{\lambda} & \frac{\lambda^2 + p_i^2}{\lambda} & -q_i \\ 0 & \frac{\lambda}{{}^c z_E} & -\frac{q_i}{{}^c z_E} & \frac{\lambda^2 - q_i^2}{\lambda} & \frac{p_i q_i}{\lambda} & p_i \end{bmatrix} \begin{bmatrix} {}^c \dot{x}_E \\ {}^c \dot{y}_E \\ {}^c \dot{z}_E \\ {}^c \dot{\alpha}_E \\ {}^c \dot{\beta}_E \\ {}^c \dot{\gamma}_E \end{bmatrix} \quad (15)$$

Powyższe równanie można uogólnić na k obserwowanych punktów:

$$\begin{bmatrix} \dot{p}_1 \\ \dot{q}_1 \\ \vdots \\ \dot{p}_k \\ \dot{q}_k \end{bmatrix} = \begin{bmatrix} \frac{\lambda}{{}^c z_E} & 0 & -\frac{p_1}{{}^c z_E} & -\frac{p_1 q_1}{\lambda} & \frac{\lambda^2 + p_1^2}{\lambda} & -q_1 \\ 0 & \frac{\lambda}{{}^c z_E} & -\frac{q_1}{{}^c z_E} & \frac{\lambda^2 - q_1^2}{\lambda} & \frac{p_1 q_1}{\lambda} & p_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\lambda}{{}^c z_E} & 0 & -\frac{p_k}{{}^c z_E} & -\frac{p_k q_k}{\lambda} & \frac{\lambda^2 + p_k^2}{\lambda} & -q_k \\ 0 & \frac{\lambda}{{}^c z_E} & -\frac{q_k}{{}^c z_E} & \frac{\lambda^2 - q_k^2}{\lambda} & \frac{p_k q_k}{\lambda} & p_k \end{bmatrix} \begin{bmatrix} {}^c \dot{x}_E \\ {}^c \dot{y}_E \\ {}^c \dot{z}_E \\ {}^c \dot{\alpha}_E \\ {}^c \dot{\beta}_E \\ {}^c \dot{\gamma}_E \end{bmatrix} \quad (16)$$

Wektor $[\dot{p}_1, \dot{q}_1, \dots, \dot{p}_k, \dot{q}_k]^T$ jest oczywiście wektorem pochodnych po czasie cech obrazu \dot{f} a wektor $[{}^c \dot{x}_E, {}^c \dot{y}_E, {}^c \dot{z}_E, {}^c \dot{\alpha}_E, {}^c \dot{\beta}_E, {}^c \dot{\gamma}_E]^T$ wektorem prędkości końcówki w powyższym przykładzie.

Gdy wymiar wektora cech jest większy niż wymiar wektora prędkości końcówki (redundancja informacji) należy dokonać aproksymacji tego ostatniego. W tym celu wyznaczamy macierz pseudoinwersji:

$$J^+ = (J^T J)^{-1} J^T \quad (17)$$

Powstaje ona poprzez następujące rozumowanie:

$$\dot{f} = J \dot{r} \quad (18)$$

mnożąc lewostronnie przez J^T :

$$J^T \dot{f} = J^T J \dot{r} \quad (19)$$

i uwzględniając, że $J^T J$ jest macierzą kwadratową, a więc jeżeli jest również macierzą nieosobliwą, to:

$$\dot{r} = (J^T J)^{-1} J^T \dot{f} \quad (20)$$

wiec J^+ można wyliczyć z wzoru (17).

Gdy wymiar wektora cech i wymiar wektora prędkości końcówki są równe (dokładna informacja) należy po prostu obliczyć macierz odwrotną:

$$J^+ = J^{-1} \quad (21)$$

Aby zmienić wartości cech obrazu należy przemieścić końcówkę manipulatora zgodnie z równaniem:

$${}^c \dot{r}_E = {}^f J_{C,E}^+ ({}^c r_E) \dot{f} \quad (22)$$

gdzie ${}^f J_{C,E}^+ ({}^c r_E)$ jest macierzą pseudoinwersji jacobianu, gdyż w ogólnym przypadku jacobian może nie być kwadratowy [12, 13].

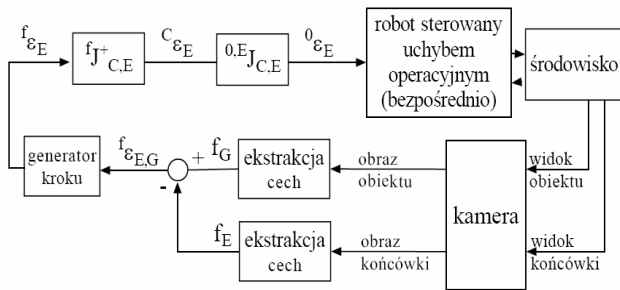
Równanie (22) możemy zapisać w postaci:

$$\lim_{\Delta t \rightarrow 0} \frac{{}^c \varepsilon_E - {}^f J_{C,E}^+ ({}^c r_E) \lim_{\Delta t \rightarrow 0} \frac{{}^f \varepsilon_E}{\Delta t}}{\Delta t} \quad (23)$$

Stąd wynika, że dla małych uchybów w przestrzeni cech ${}^f \varepsilon_{E,E}$ uchyb w przestrzeni zadaniowej ${}^c \varepsilon_{E,E}$ wynosi:

$${}^c \varepsilon_E = {}^f J_{C,E}^+ ({}^c r_E) {}^f \varepsilon_E = {}^f J_{C,E}^+ ({}^c r_E) (f_E - f_E) \quad (24)$$

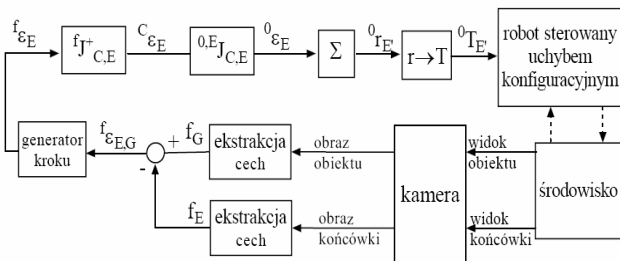
gdzie f_E jest zadany (docelowym) wektorem cech, a f_E zmierzonym. Schemat układu regulacji znajduje się na rys. 6.



Rys. 6. Serwomechanizm wizyjny wyznaczający uchyb wizyjny w przestrzeni cech obrazu, z robotem sterowanym uchybem operacyjnym, z nieruchomą kamerą

Fig. 6. Image based, operational error control using a stand-alone camera

Schemat serwomechanizmu z robotem sterowanym uchybem konfiguracyjnym znajduje się na rys. 7.

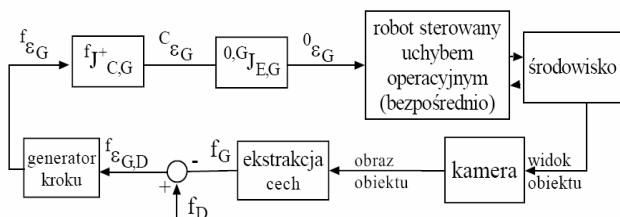


Rys. 7. Serwomechanizm wizyjny wyznaczający uchyb wizyjny w przestrzeni cech obrazu z nieruchomą kamerą, z robotem sterowanym uchybem konfiguracyjnym. Przyrost ${}^0_{E,E}$ jest na bieżąco sumowany, by otrzymać ${}^0_{r,E}$.

Fig. 7. Image based, configuration error control using a stand-alone camera. The ${}^0_{E,E}$ increments are constantly being summed to produce ${}^0_{r,E}$.

3.2. Kamera ruchoma, nie obserwująca końcówki (IB-EOL-EIH)

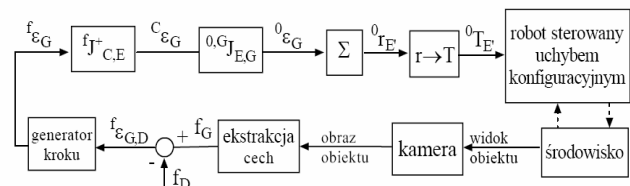
W tym wypadku sterowanie efektem polega na tym, że cechy obrazu obserwowanego obiektu f_G należy sprowadzić do z góry ustalonych cech f_D . Cechy f_D mogą być wyznaczone poprzez „uczenie przez pokazywanie” (ang. teaching by showing). Metoda ta polega na tym, że stawiany jest przed kamerą obiekt w pozycji docelowej i wyznaczane są cechy obrazu f_D . Można również wyznaczyć cechy obrazu na podstawie ${}^C T_E$ (gdyż chcemy, aby obiekt osiągnął pozycję względem kamery taką samą jaką ma końcówka). Powyższa metoda jest metodą odwrotną do poprzedniej metody estymacji pozycji. Stosowana jest rzadziej ze względu na potrzebę wyznaczania estymacji pozycji. Schemat układu regulacji znajduje się na rys. 8.



Rys. 8. Serwomechanizm wizyjny wyznaczający uchyb w przestrzeni cech obrazu z robotem sterowanym uchybem zadaniowym, z ruchomą kamerą

Fig. 8. Image based, operational error control using an eye-in-hand setup

Jako uchyb dla robota podawany jest ${}^0_{E,G}$ a nie ${}^0_{E,E}$. Jednak oba wektory leżą na tej samej prostej, więc mają ten sam kierunek. Poza tym mają ten sam zwrot. Ponieważ dodatkowo długość wektora ${}^0_{E,G}$ jest ustalana arbitralnie przez generator kroku (przetwarzana tylko potem liniowo przez jacobiany), staje się przyrostem położenia końcówki ${}^0_{E,E}$. Aby przekształcić przyrost ${}^E_{E,G}$ w ${}^0_{E,G}$ powinno się użyć jacobianu ${}^0_G J_{C,G}$. Jednak wyznaczenie takiego jacobianu wymaga znajomości pozycji obiektu względem kamery. Chcemy tego uniknąć, więc używamy jacobianu ${}^0_G J_{E,G}$ zamiast ${}^0_G J_{C,G}$. Ma to sens, gdy orientacja kamery jest taka sama jak orientacja końcówki. Schemat serwomechanizmu z robotem sterowanym uchybem konfiguracyjnym znajduje się na rys. 9.



Rys. 9. Serwomechanizm wizyjny wyznaczający uchyb w przestrzeni cech obrazu z ruchomą kamerą, z robotem sterowanym uchybem konfiguracyjnym. Przyrost ${}^0_{E,G}$ jest traktowany jako równy przyrostowi ${}^0_{E,E}$, więc może być na bieżąco sumowany, by otrzymać ${}^0_{r,E}$.

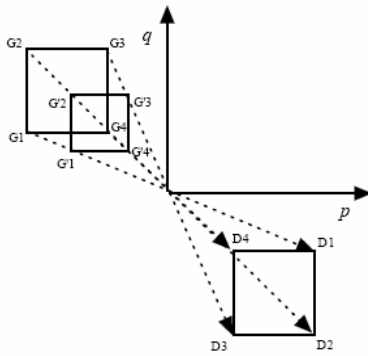
Fig. 9. Image based, configuration error control using an eye-in-hand setup. The increment ${}^0_{E,G}$ is treated as equal to ${}^0_{E,E}$, so it can be summed up to obtain ${}^0_{r,E}$.

3.3. Wady i zalety wyznaczania uchybu w przestrzeni cech obrazu

Zaletą tej metody jest mniejsza wrażliwość na szum obrazu oraz błędne wyznaczenie wewnętrznych parametrów kamery, takie jak ogniskowa (typy IB-EOL,ECL-SAC,EIH). Metoda pozwala również trzymać obserwowany obiekt w polu widzenia, dlatego że śledzone są cechy obrazu, przez które dany obiekt jest reprezentowany (dla typu IB-EOL-EIH).

Wadami natomiast jest posiadanie przez Jacobian odwrotny osobliwości, dla których uchyb w przestrzeni cech obrazu jest zerowy przy równoczesnym niezerowym uchybie w przestrzeni zadaniowej (typy IB-EOL,ECL) [2] oraz nieintuicyjne ruchy w przestrzeni zadaniowej, w szczególności dla wielkich ruchów kamery (dla typu IB-EOL).

Ponieważ uchyb w przestrzeni obrazu liczony jest jako odległość euklidesowa, powstaje problem z ruchem obrotowym kamery. Jednym ze znanych przypadków jest „problem cofania się kamery” (ang. camera retreat problem) [2]. Gdy dany obiekt jest obrócony względem osi optycznej kamery o π rad, prawo regulacji oparte na danym jacobianie obrazu nie wygeneruje w przestrzeni obrazu trajektorii po łukach, tak aby każda cecha obrazu została odpowiednio obrócona, a będzie generował trajektorie po liniach prostych, które przecinają się w środku optycznym kamery. Poszczególne cechy obrazu generowane dla kolejnego kroku trajektorii będą przesuwać się w kierunku środka. Jak widać (rys. 10) generowane odległości na obrazie między tymi cechami będą się zmniejszać, co spowoduje ruch kamery do tyłu, tak aby przy danej ogniskowej kamera obserwowała zmniejszający się obiekt. W ekstremalnym przypadku kamera oddali się do nieskończoności, gdy wszystkie cechy obrazu zbiegną się w jednym punkcie. W praktyce jednak zatrzyma się, gdy końcówka manipulatora osiągnie granicę przestrzeni roboczej robota. Oczywiście jest to przypadek szczególny, jednak pokazuje, jak wbrew intuicji może być generowana trajektoria w przestrzeni zadaniowej. W ogólnym przypadku każdy obrót obiektu generuje dodatkowy ruch końcówki wzdłuż osi optycznej kamery.



Rys. 10. „Problem cofania się kamery” - punkty G1-4 widziane są w chwili początkowej ruchu, a punkty D1-4 są punktami zadanymi i są obrócone o π rad wokół osi optycznej, a więc kamera powinna obrócić się o π rad w kierunku przeciwnym do wskazówek zegara (w układzie zadaniowym). Ponieważ trajektorie ruchu w przestrzeni obrazu są prostoliniowe (przerwane strzałki), obiekt w przestrzeni obrazu musi maleć (G¹⁻⁴), co w przestrzeni zadaniowej skutkuje cofaniem się kamery

Fig. 10. The camera retreat problem. The points G1-4 can be seen in the first step of motion, the points D1-4 are the desired ones (goal of motion) and are rotated exactly π rad around the optical axis (in task space). Since the trajectories in the image space are represented as straight lines (dashed arrows), the object in the image space instead of turning will become smaller and smaller (G¹⁻⁴) what causes the camera to retreat in the task space

Jest kilka sposobów radzenia sobie z nieprzewidywalnymi ruchami kamery związanymi z ruchami obrotowymi. Jednym z nich jest stosowanie innych cech obrazu niż pojedyncze punkty, z których trudno wnioskować o rotacji obiektu. Propozycjami były odcinki pomiędzy danymi punktami [2] oraz metody momentowe [1]. Wadą tych propozycji była znaczna komplikacja jakobianu obrazu, gdyż opis matematyczny danej cechy musiał być zawarty w jakobianie. Innym pomysłem jest sterowanie rotacją w przestrzeni zadaniowej [7, 10]. W literaturze można znaleźć również inne sposoby [3, 8, 9].

4. Podsumowanie

Cechy opisanych serwomechanizmów wizyjnych można zamieścić w tabeli (pojawienie się macierzy T oznacza konieczność jej kalibracji, ε oznacza, że uchyb jest zależny od kalibracji robota).

Tab. 1. Porównanie serwomechanizmów wizyjnych
Tab. 1. A comparison of the visual servos

		PB		IB	
		SAC	EIH	SAC	EIH
JEC	EOL	${}^0T_E, {}^0T_C, \varepsilon$	${}^0T_E, {}^E T_C$	nierealizowalny	${}^0T_E, {}^E T_C$
	ECL	0T_E	${}^0T_E, {}^E T_C(f_E)$ = const	0T_E	nierealizowalny
PEC	EOL	${}^0T_E, {}^0T_C, \varepsilon$	${}^0T_E, {}^E T_C$	nierealizowalny	${}^0T_E, {}^E T_C$
	ECL	0T_E	${}^0T_E, {}^E T_C(f_E)$ = const	0T_E	nierealizowalny

Jeśli kamera jest ruchoma i końcówka jest nieobserwowana (PB-EOL-SAC) potrzebna jest kalibracja kamera-robot 0T_C i kalibracja robota 0T_E . Jeśli dodatkowo obserwowana jest

kończówka (PB,IB-ECL-SAC) kalibracja kamera-robot nie jest konieczna, natomiast istotna jest kalibracja robota; uchyb jednak nie zależy od kalibracji robota. Gdy kamera jest na końcówce i jej nie obserwuje (PB,IB-EOL-EIH) ważna jest kalibracja kamera-kończówka ${}^E T_C$ i kalibracja robota, ale uchyb jest niezależny od kalibracji robota. Gdy końcówka jest obserwowana przez kamerę (PB-ECL-EIH) kalibracja kamera-kończówka nie jest potrzebna, gdyż położenie końcówki względem kamery można określić dzięki jej lokalizacji na obrazie i estymacji pozycji w przestrzeni zadaniowej.

Serwomechanizmy IB-EOL-SAC są nierealizowalne, gdyż w typach IB-SAC końcówka jest z definicji obserwowana, w celu wyznaczenia uchybu w przestrzeni cech obrazu. Serwomechanizmy IB-ECL-EIH nie są realizowane, gdyż końcówka jest nieruchoma względem kamery, więc widok końcówki nie zmienia cech obrazu.

Praca jest finansowana przez grant MNiI: 4 T11A 003 25.

5. Literatura

- [1] F. Chaumette: Image Moments: A General and Useful Set of Features for Visual Servoing. IEEE Transactions on Robotics, vol. 20, no. 4, August 2004
- [2] F. Chaumette: Potential problems of stability and convergence in image-based and position-based visual servoing. The Confluence of Vision and Control, Lecture Notes in Control and Information Systems, New York: Springer-Verlag, 1998 vol. 237, pp. 66-78
- [3] G. Chesi, A. Vicino: Visual Servoing for Large Camera Displacements. IEEE Transactions on Robotics, vol. 20, no. 4, August 2004
- [4] G. Chesi, K. Hashimoto, D. Prattichizzo, A. Vicino: Keeping Features in the Field of View in Eye-In-Hand Visual Servoing: A Switching Approach. IEEE Transactions on Robotics, vol. 20, no. 5, October 2004
- [5] W. Czajewski, M. Staniak, C. Zielinski: Pewne aspekty wykorzystania informacji wizyjnej w robotach usługowych. Materiały VIII Krajowej Konferencji Robotyki - Polanica Zdrój 23 - 25 czerwca 2004
- [6] Lingfeng Deng, W. J. Wilson, F. Janabi-Sharifi: Dynamic Performance of the Position-Based Visual Servoing Method in the Cartesian and Image Spaces. Proceedings of the 2003 IEEE/RSJ Intl. Conference on Intelligent Robots and Systems, Las Vegas, Nevada - October 2003
- [7] K. Deguchi: Optimal motion control for image-based visual servoing by decoupling translation and rotation. Proc. Int. Conf. Intelligent Robots and Systems, Oct. 1998, pp. 705-711
- [8] N. R. Gans S. Hutchinson, P. I. Corke: Performance Tests for Visual Servo Control Systems, with Application to Partitioned Approaches to Visual Servo Control. The International Journal of Robotics Research, Vol. 22, No. 10-11, 955-981 (2003)
- [9] M. Iwatsuki, N. Okijama: A New Formulation of Visual Servoing Based on Cylindrical Coordinate System. IEEE Transactions on Robotics, vol. 21, no. 2, April 2005
- [10] E. Malis, F. Chaumette, S. Boudet: 2-1/2-D Visual Servoing. IEEE Transactions on Robotics and Automation, vol. 15, no. 2, April 1999
- [11] W. J. Wilson, C. C. Hulls, G. S. Bell: Relative End-Effector Control Using Cartesian Position-Based Visual Servoing. IEEE Transactions on Robotics and Automation, vol. 12, no. 5, October 1996
- [12] <http://mathworld.wolfram.com/LeastSquaresFitting.html>
- [13] <http://mathworld.wolfram.com/Moore-PenroseMatrixInverse.html>