

تست عملکردی ربات ۶ محوره به کمک معادلات سینماتیکی و تعیین محدوده کاری آن

اعظم السادات نوربخش^۱، محرم حبیب نژاد کورایم^۲

^۱ کارشناس ارشد، هوش مصنوعی و رباتیک، دانشگاه آزاد اسلامی واحد لاهیجان، لاهیجان، ایران

A.S.Nourbakhsh@gmail.com

^۲ استاد، گروه مکانیک، دانشگاه علم و صنعت ایران، تهران، ایران

hkorayem@IUST.AC.IR

چکیده

در این مقاله سعی شده که از سیستم بینایی در جهت کنترل یک ربات 6R صنعتی استفاده شود و سیستم بینایی با گرفتن تصویر از محیط و پردازش بر روی تصویر، هدف را تشخیص داده و ربات را کنترل کند. جهت تست عملکرد ربات از معادلات سینماتیکی استفاده می گردد. در تست سینماتیکی مستقیم، مقادیر زوایای مفاصل مشخص است. براساس زوایای داده شده، مکان پنجه توسط سیستم براساس ماتریس انتقال، به دست می آید. در تست سینماتیکی معکوس، مختصات مکان پنجه در دسترس است. با استفاده از معادلات سینماتیکی معکوس، مقدار زوایای مفاصل مشخص شده و ربات به همان مقادیر می چرخد و براساس آن، مکان پنجه در فضا مشخص می گردد. مشخص نمودن بازه مقدار دهی ربات جهت تست عملکرد، ضروری به نظر می رسد. از این رو جهت یافتن نقاط تکین، دترمینان ماتریس ژاکوبین بازوی مکانیکی را برابر صفر قرارداده و فضای کاری ربات را با کمک شبیه ساز Matlab مشخص نموده ایم. همچنین نتایج به دست آمده توسط استاندارد 2-ANSI-RIA R15.05 تست شده اند.

کلمات کلیدی: ربات ۶ محوره، دوربین، کنترل بینایی، معادلات سینماتیکی مستقیم و معکوس، فضای کاری ربات، تست های اجرائی.

۱ مقدمه

سیستم های بینایی معمولاً بر اساس تعداد دوربین و موقعیت آن ها دسته بندی می شوند. سیستم های بینایی تک دوربینه، به علت ارزانی و آسانی ساخت، نسبت به سیستم های بینایی چند دوربینه بیشتر مورد استفاده قرار می گیرند. از طرف دیگر، استفاده از دوربین به صورت یک ساختار استریو (به عنوان مثال مشاهده محیط کاری بوسیله دو دوربین) بسیاری از مشکلات بینایی کامپیوتر را آسانتر می سازد. Gregory Flandin چگونگی استفاده از دوربین بصورت های (Eye-in-hand / Eye-to-hand) برای کنترل ربات تشریح کرده است [4]. این دو مساله

که کاربردهای متفاوتی در صنعت دارند کاملاً متفاوت هستند. در روش Eye-in-hand دوربین بر روی پنجه قرار می گیرد ولی در روش Eye-to-hand دوربین کل محیط کاری ربات را زیر نظر دارد و در بیرون ربات نصب می گردد. یک مرور روی visual servoing توسط Hutchinson انجام شده است [6]. وی اصول visual servoing مبتنی بر خصوصیات هندسی را تشریح کرده است. Croke نشان داد که کارآیی الگوریتم های visual servoing می تواند توسط بکار گرفتن مدلی از دینامیک سیستم بهبود پیدا کند [7]. Skaar و همکارانش، پیشنهاد کردند که بسیاری از نیازهای دنیای حقیقی با استفاده از عملکرد محدوده فضای یک یا چند دوربین می تواند پوشش داده شود [8]. نقطه دید و خصوصیات رباتی که یک دوربین روی پنجه آن نصب شده، توابعی از نقاط وابسته دوربین با توجه به شیء هدف هستند. عموماً این تابع غیرخطی و به صورت عمودی کوپل شده است. بگونه ای که حرکت درجه آزادی پنجه از حرکت پیچیده بسیاری از خصوصیات منتج می شود. برای سادگی و سهولت، این روابط می توانند در مورد نقطه مورد نظر خطی شوند. تحقیقات زیادی از این روش Feature-based برای کنترل بینائی استفاده کرده اند.

در این تحقیقات یک حلقه بسته برای کنترل مفاصل ربات مورد استفاده قرار گرفته است. Feddama از مفهوم یک مولد مسیر فضای خصوصیت و کنترل مفصل حلقه بسته برای غلبه بر مشکل نرخ پائین نمونه گیری بینائی استفاده کرده است. کارهای آزمایشگاهی کنترل بینائی Feature-based برای ربات ۴ درجه آزادی ثابت شده است. [5] Hashimoto و همکاران، یک شبیه ساز برای مقایسه روش های Position-based و Feature-based ارائه نموده اند [9]. کنترل بینائی ربات 3P، با استفاده از روش Feature-based توسط کورایم و همکاران شبیه سازی شده است [2]. در این تحقیق، روش Feature-based جهت کنترل پنجه برای رسیدن به شیء هدف مورد استفاده قرار گرفته است. در آزمایشگاه رباتیک دانشگاه علم و صنعت رباتی ساخته شد که دارای ۶ درجه آزادی بوده و همه مفاصل آن به صورت چرخشی حرکت می کنند. برای کنترل بینائی این ربات به طریقه Position-based سیستم شبیه سازی ارائه شده است. صحت عمل در این سیستم شبیه ساز، بوسیله استانداردهای ISO 9283 و ANSI-RIA R15.05-2 مورد آزمایش قرار گرفته است. [10]

در این نرم افزار شبیه ساز، از سه دوربین در محیط استفاده شده است. کنترل ربات به طریقه Feature-based انجام شده و توسط دوربینی که روی پنجه ربات نصب شده است، خصوصیات شیء انتخاب شده چک شده و عملیات تشخیص و گرفتن شیء صورت می گیرد. در هر مرحله از عملیات توسط دو دوربین ۲۰۱ تصویر برداری شده و این تصاویر ثبت می گردند. جهت تست ربات، از دو دوربین ۳۰۱ که در فاصله ثابت روی زمین نصب شده اند و ربات را از جلو و از دید سمت راست مشاهده می کنند، استفاده می شود. این دو دوربین از ربات و اطراف آن تصویر می گیرند. پس از تجزیه و تحلیل تصویر و تشخیص شیء مورد نظر و پنجه، مکان آن ها در مختصات سطح تصویر تخمین زده می شود و سپس سیستم بینائی، پنجه را به سمت شیء مورد نظر

هدایت می کند. از آنجا که برای داشتن مکان های پنجه و شیء در مختصات مرجع کروی^۱ نگاشت مختصات نقشه تصویر برای مختصات مرجع مورد نیاز است، از شبکه عصبی به جای نگاشت، استفاده شده است.

۲ معرفی ربات 6R

رباتی که در آزمایشگاه دانشگاه علم و صنعت طراحی شده، ۶ درجه آزادی داشته و تمام مفاصل آن از نوع چرخشی می باشند. همانگونه که در شکل ۱ دیده می شود، ۳ درجه آزادی در بدنه، شانه و سر و همچنین ۳ درجه آزادی در پنجه دارد که چرخش های roll، pitch و yaw را انجام می دهند. بازوی اول، در سطح افق، حول محور قائم می چرخد. بازوی دوم در یک محور عمودی قائم نسبت به سطح چرخش بازوی اول چرخش نموده و اتصال سوم در یک سطح موازی چرخش اتصال دوم می چرخد.



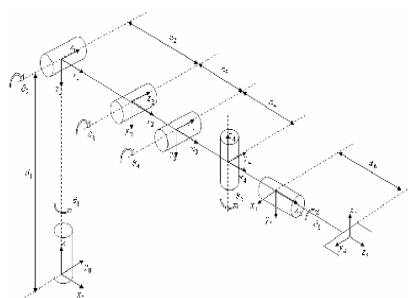
شکل ۱: ساختار ربات شش درجه آزادی

اعمال ربانی بنا بر تعریف، عبارت از به حرکت در آوردن قطعات و ابزارها در فضا به وسیله نوعی مکانیزم است. استفاده بهینه از عملکرد بازو منوط به کنترل دقیق حرکت ها می باشد و کنترل دقیق حرکت ها با شناسایی و تعریف آن ها به وسیله کمیت های ریاضی مناسب قابل انجام است. در ادامه، معادلات سینماتیکی ربات طراحی شده، ارایه می شوند.

۲-۱ سینماتیک مستقیم

مطابق این سیستم صورت تبدیل کلی که چارچوب های متصل به رابط های مجاور را به یکدیگر مربوط می سازد، به دست می آید. با ادغام تبدیل های کلی منفرد، مکان و جهت گیری رابط ۶ را نسبت به رابط صفر به دست می آوریم. برای حل مدل سازی ربات 6R، نمادسازی دینامیت هارتنبرگ مورد استفاده قرار گرفته است. سیستم های مختصات متصل به هر مفصل و مدل ساده شده ربات 6R در شکل (۲) نمایش داده شده است. پارامترهای دینامیت هارتنبرگ ربات 6R نیز در جدول (۱) آورده شده اند.

^۱ Global Reference Coordinate



شکل ۲: نمادسازی دینامیت-هارتبرگ ربات 6R

محورها	θ	d	a	a
1	θ_1	d	0	$-\pi/2$
2	θ_2	0	a_2	0
3	θ_3	0	a_3	0
4	θ_4	0	a_4	$\pi/2$
5	θ_5	0	0	$-\pi/2$
6	θ_6	d	0	0

جدول ۱: پارامترهای حاصل از نامگذاری دینامیت-هارتبرگ

می توان با ضرب ماتریسی، ماتریس های تبدیل رابط ها، تبدیل کلی T_6^0 ، را محاسبه کرد، که عبارتست از:

$$T_6^0 = T_1^0 T_2^1 T_3^2 \dots T_6^5 \quad (1)$$

ماتریس T_6^0 ، موقعیت و جهت پنجه را نسبت به مختصات مرجع مشخص می کند. اگر مکان مفاصل با استفاده از حس گر مشخص شوند، می توان موقعیت پنجه را به وسیله ماتریس T_6^0 بدست آورد:

$$T_6^0 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

که در آن:

$$\begin{aligned} n_x &= C_1(C_5 C_6 C_{234} - S_6 S_{234}) - S_1 S_5 C_6 \\ n_y &= S_1(C_5 C_6 C_{234} - S_6 S_{234}) + C_1 S_5 C_6 \\ n_z &= -C_5 C_6 C_{234} - S_6 C_{234} \end{aligned} \quad (3)$$

$$o_x = -C_1(C_5 C_6 C_{234} - S_6 S_{234}) - S_1 S_5 S_6 \quad (4)$$

$$o_y = -S_1(C_5C_6C_{234} - S_6S_{234}) - C_1S_5S_6$$

$$o_z = S_{234}C_5S_6 - C_6C_{234}$$

$$a_x = -C_1S_5C_{234} - S_1C_5$$

$$a_y = -S_1S_5C_{234} + C_1C_5$$

$$a_z = S_{234}S_5$$

(۵)

$$P_x = -C_1(d_6S_5C_{234} - a_4C_{234} - a_3C_{23} - a_2C_2) - d_6S_1C_5$$

$$P_y = -S_1(d_6S_5C_{234} - a_4C_{234} - a_3C_{23} - a_2C_2) - d_6C_1C_5$$

$$P_z = -C_1(d_6S_5S_{234} - a_4C_{234} - a_3S_{23} - a_2C_2) + d_1$$

(۶)

نکته قابل توجه اینکه جهت رعایت اختصار از خلاصه نویسی زیر استفاده شده است:

$$\cos \theta_i = C_i, \quad \sin \theta_i = S_i$$

$$\cos(\theta_i + \theta_j) = C_{ij}, \quad \sin(\theta_i + \theta_j) = S_{ij}$$

۲-۲ سینماتیک معکوس

اگر یک نقطه دلخواه و جهت حرکت پنجه ربات مشخص باشد، برای یافتن مقادیر زوایای مفصل که در معادله مستقیم (۲) صدق می کنند، نیاز به حل سینماتیک معکوس ربات داریم. برای ربات 6R، معادلات سینماتیک معکوس به صورت زیر خواهند بود:

$$\theta_1 = \tan^{-1} \left(\frac{P_y - d_6 a_y}{P_x - d_6 a_x} \right) \quad \text{and} \quad \theta_1 = \theta_1 + \pi \quad (7)$$

$$\theta_2 = -\tan^{-1} \left(\frac{\pm [1 - (w/q)^2]^{1/2}}{w/q} \right) + \tan^{-1} \left(\frac{u}{t} \right) \quad (8)$$

$$\theta_3 = \tan^{-1} \left(\frac{u - a_2 S_2}{t - a_2 C_2} \right) - \theta_2 \quad (9)$$

$$\theta_{234} = \tan^{-1} \left(\frac{-a_z}{a_x C_1 + a_y S_1} \right), \quad \text{if } \theta_3 > 0, \text{ else } \theta_{234} = \theta_{234} + \pi \quad (10)$$

$$\theta_4 = \theta_{234} - \theta_2 - \theta_3 \quad (11)$$

$$\theta_5 = \tan^{-1} \left(\frac{\pm d_6 [1 - (a_y C_1 - a_x S_1)^2]^{1/2}}{P_y C_1 - P_x S_1} \right) \quad (12)$$

$$\theta_6 = \tan^{-1} \left(\frac{o_x S_1 - o_y C_1}{n_y C_1 + n_x S_1} \right) \text{ for } \theta_5 > 0 \text{ and } \theta_6 = \theta_6 + \pi \text{ for } \theta_5 < 0 \quad (13)$$

که در این معادلات مقادیر زیر لحاظ شده اند:

$$t = a_2 C_2 + a_3 C_{23} = C_1 P_x + S_1 P_y + d_6 S_5 C_{234} - a_4 C_{234}$$

$$u = a_2 S_2 + a_3 S_{23} = -P_z + d_1 - a_4 S_{234} + d_6 S_5 S_{234}$$

$$w = t C_2 - u S_2 = \frac{-a_3^2 + t^2 + u^2 + a_2^2}{2a_2}$$

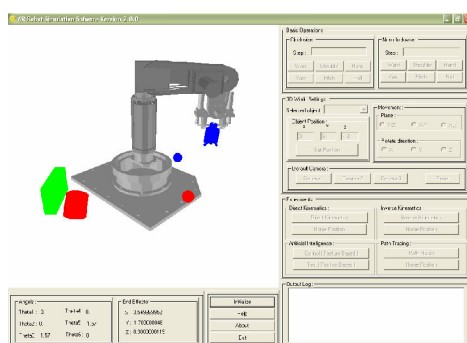
$$q = \sqrt{t^2 + u^2}$$

معادلات فوق برای شبیه سازی حرکت ربات در مراحل تست و کنترل مورد استفاده قرار می گیرند.

۳ معرفی نرم افزار شبیه ساز

۳-۱ آشنائی با نرم افزار

یکی از اهداف این پروژه، به کارگیری مجموعه مسایل مربوط به بینائی ماشین، از نحوه تصویر برداری توسط دوربین گرفته تا پردازش تصویر و شناخت شیء مورد نظر و در نهایت کنترل ربات توسط بینائی می باشد و لذا با ایجاد محیط سه بعدی کار شبیه سازی ربات 6R انجام شده است. در این نرم افزار اقدام به استفاده از سه دوربین شده است. کاربر در هر لحظه می تواند با سوئیچ کردن روی هر یک از سه دوربین مونیتورینگ داشته باشد. نرم افزار شبیه ساز توسط دوربینی که روی پنجه نصب شده است، اطلاعات را دقیقاً مشابه حالت واقعی از دوربین گرفته و پس از پردازش اطلاعات تصویری، اشیاء مشاهده شده را شماره گذاری می نماید و به دنبال شیء ای با خصوصیات شیء هدف جستجو می کند. چنانچه شیء در اولین تصویر گرفته شده مشاهده نشد، سیستم بینائی، پنجه ربات را در فضای کاری حرکت می دهد و دوباره تصویر می گیرد. این عمل آنقدر ادامه پیدا می کند تا سیستم بینائی شیء مورد نظر را مشاهده نماید یا اینکه تمام فضای کاری ربات جستجو شده باشد و هدف پیدا نشود.



شکل ۳: شمائی از نرم افزار ارابه شده کنترل بینائی

پس از شناسایی هدف توسط سیستم بینایی، ربات بوسیله یک الگوریتم کنترلی که توسط سیستم بینایی اعمال می‌شود، شروع به حرکت می‌نماید و به طور مداوم با دوربین نصب شده روی پنجه، از فضای کاری ربات تصویر برداری می‌شود و عملیات آنقدر ادامه پیدا می‌کند تا پنجه در فاصله معینی از هدف قرار گیرد. هدف از نصب دو دوربین در محیط، مشاهده کل فضایی که ربات در آن قرار دارد توسط ناظر می‌باشد. این دوربین‌ها به صورت ثابت نبوده و می‌توانند توسط کاربر در هر نقطه از فضا به حرکت درآیند. دوربین شماره یک از سمت راست و دوربین شماره سه از روبه‌رو به ربات نگاه می‌کنند. در این نرم افزار امکان حرکت شیء انتخاب شده نسبت به صفحات مختلف محورهای مختصات امکان پذیر است. در این حالت می‌توان با استفاده از دکمه سمت چپ ماوس، مکان شیء انتخاب شده را در پلان تصویر تغییر داد. همچنین هر یک از اشیاء انتخاب شده، حول محور x ، y و یا z می‌چرخند. این عمل با استفاده از دکمه سمت راست ماوس انجام می‌شود.

۲-۳ تشخیص فضای کاری ربات 6R

از آن جا که موقعیت و جهت پنجه مستقیماً با تغییر مکان های مفصل در ارتباط است، برای متناسب کردن حرکات مفاصل، روابط دیفرانسیلی بین تغییر مکان های مفصل و مکان پنجه را استخراج نموده و سپس برای حرکت هر مفصل جداگانه آن را حل می‌کنیم.

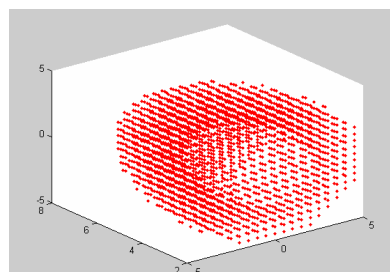
ماتریس شامل مشتقات جزئی موقعیت پنجه نسبت به تغییر مکان های مفاصل، ژاکوبین بازوی مکانیکی نامیده می‌شود. نقاطی که در مینان ماتریس ژاکوبین برابر صفر می‌شود، نقاط تکین نامیده شده و حرکت ربات در یک راستای قائم صورت می‌گیرد. از آنجا که حرکت هر یک از مفاصل دامنه محدودی دارد و اتصال مفاصل بازه حرکتی شان را محدود می‌کند. [1]

این مسایل مشکلاتی را در زمینه حرکت ربات ایجاد می‌کند و باعث می‌شود که در شبیه سازی به نتایج مطلوب دست پیدا نکنیم. به همین دلیل، محاسبه نقاط تکین و فضای کاری ربات، جهت آگاهی از بازه مقادیر داده، ضروری به نظر می‌رسد.

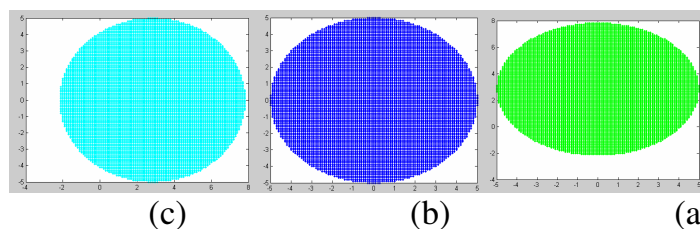
برای به دست آوردن نقاط تکین؛ با توجه به اینکه مختصات موقعیت پنجه با P_x , P_y , P_z طبق رابطه ۶ در نظر گرفته شده است، ماتریس ژاکوبین 6×6 برای ربات طبق روابط زیر به دست آمده و در مینان آن برابر صفر در نظر گرفته می‌شود.

$$J = \{ \{ D[px, \theta_1], D[px, \theta_2], D[px, \theta_3], D[px, \theta_4], D[px, \theta_5], D[px, \theta_6] \}, \{ D[py, \theta_1], D[py, \theta_2], D[py, \theta_3], D[py, \theta_4], D[py, \theta_5], D[py, \theta_6] \}, \{ D[pz, \theta_1], D[pz, \theta_2], D[pz, \theta_3], D[pz, \theta_4], D[pz, \theta_5], D[pz, \theta_6] \}, \{ 0, -\sin[\theta_1], -\sin[\theta_1], -\sin[\theta_1], -\cos[\theta_4] \} \cdot \{ -\cos[\theta_1] \cos[\theta_3] \sin[\theta_2] - \cos[\theta_1] \cos[\theta_2] \sin[\theta_3] + (\cos[\theta_1] \cos[\theta_3] - \cos[\theta_1] \sin[\theta_2] \sin[\theta_3]) \sin[\theta_4] - \cos[\theta_5] \sin[\theta_1] - (\cos[\theta_4] (\cos[\theta_1] \cos[\theta_2] \cos[\theta_3] - \cos[\theta_1] \sin[\theta_2] \sin[\theta_3]) + (-\cos[\theta_1] \cos[\theta_3] \sin[\theta_2] - \cos[\theta_1] \cos[\theta_2] \sin[\theta_3]) \sin[\theta_4]) \sin[\theta_5] \}, \{ 0, \cos[\theta_1], \cos[\theta_1], \cos[\theta_1], -\cos[\theta_4] \} \cdot \{ -\cos[\theta_3] \sin[\theta_1] \sin[\theta_2] - \cos[\theta_2] \sin[\theta_1] \sin[\theta_3] + (\cos[\theta_2] \cos[\theta_3] \sin[\theta_1] - \sin[\theta_1] \sin[\theta_2] \sin[\theta_3]) \sin[\theta_4], \cos[\theta_1] \cos[\theta_5] - (\cos[\theta_4] (\cos[\theta_2] \cos[\theta_3] \sin[\theta_1] - \sin[\theta_1] \sin[\theta_2] \sin[\theta_3]) + (-\cos[\theta_3] \sin[\theta_1] \sin[\theta_2] - \cos[\theta_2] \sin[\theta_1] \sin[\theta_3]) \sin[\theta_4]) \sin[\theta_5] \}, \{ 1, 0, 0, 0, -\cos[\theta_4] \} \cdot \{ -\cos[\theta_2] \cos[\theta_3] + \sin[\theta_2] \sin[\theta_3] + (-\cos[\theta_3] \sin[\theta_2] - \cos[\theta_2] \sin[\theta_3]) \sin[\theta_4] \} \sin[\theta_5] \} \} \quad (6)$$

براین اساس، اقدام به مشخص کردن فضای کاری ربات به وسیله Matlab Simulink نمودیم. با توجه به شکل سه بعدی بدست آمده، بازه مقادیر صحیح مشخص می گردد. برای درک بهتر، محدوده کاری ربات در سه فضای دو بعدی مشخص شده اند.



شکل ۴: نمایش فضای کاری ربات 6R (یک کره توخالی)



شکل ۵: (a) تصویر فضای کاری ربات روی دو محور X و Y

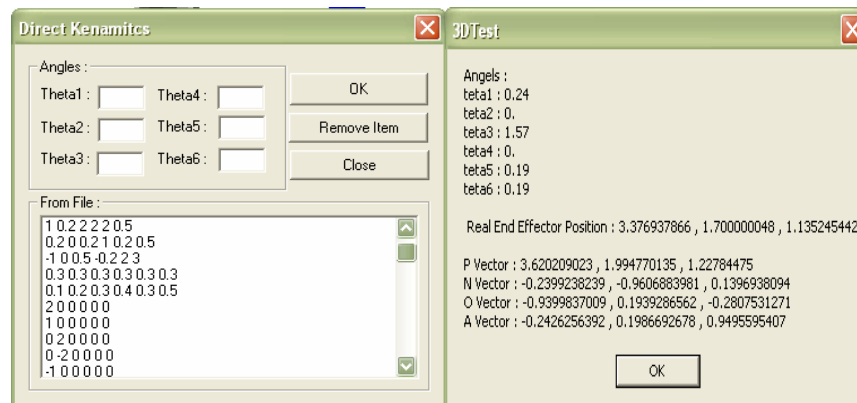
(b) تصویر فضای کاری ربات روی دو محور X و Z

(c) تصویر فضای کاری ربات روی دو محور Y و Z

۴ تست های اجرایی

تست های اجرایی روی ربات در دو حالت سینماتیک مستقیم، سینماتیک معکوس، انجام شده اند. در تست سینماتیک مستقیم، مقادیر زوایای مفاصل مشخص است. براساس زوایای داده شده، مکان پنجه توسط سیستم بدست می آید. این عمل در برنامه شبیه ساز توسط تابع DirectKinematics صورت می گیرد. از آن جایی که سینماتیک مستقیم در ربات 6R با شش زاویه تعیین می شود و از روی آن ها جهت و مکان end-effector بدست می آید، این تابع نیز شش زاویه را از کاربر خوانده و جهت و مکان end-effector را در فضای سه بعدی بر می گرداند.

این تابع وظیفه چرخاندن مفصل ها را به اندازه زوایای وارد شده دارد و پس از آن نتایج محاسبات جهت و مکان end-effector را در فایل DKOut.txt ذخیره می کند. پنجره طراحی شده برای این تابع، قابلیت استفاده از مقادیر قبلی وارد شده را به کاربر می دهد و آن ها را در فایل DK.txt ذخیره می کند. از تفاوت مکان واقعی پنجه در صفحه و مکان به دست آمده تئوریک، خطای مکانی سیستم مشخص می گردد. شکل ۷، نمودار خطای مکانی سیستم در سینماتیک مستقیم را نشان می دهد.

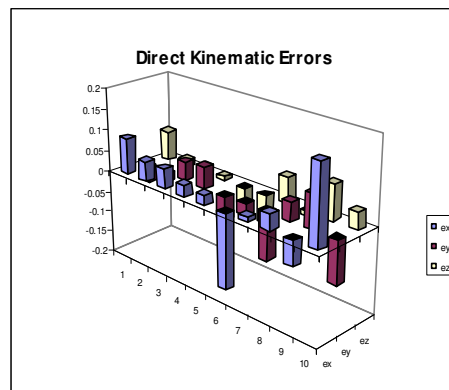


(a)

(b)

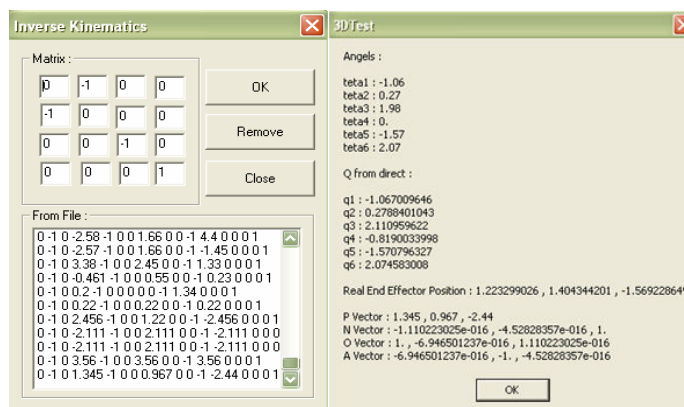
شکل ۶: (a) پنجره ورود اطلاعات در تست سینماتیک مستقیم

(b) پنجره اطلاعات خروجی در تست سینماتیک مستقیم



شکل ۷: نمودار خطای مکانی در سینماتیک مستقیم

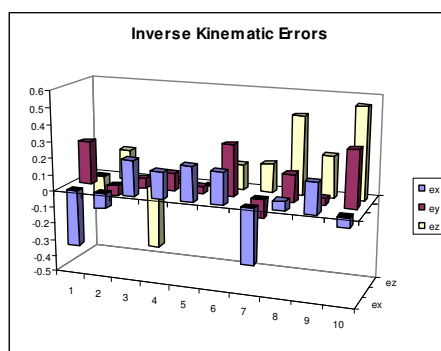
در تست سینماتیک معکوس، مختصات مکان پنجه در دسترس است. با استفاده از معادلات سینماتیک معکوس، مقدار زوایای مفاصل مشخص شده و ربات به همان مقادیر می چرخد. و براساس آن، مکان پنجه در فضا مشخص می گردد. تفاوت مکان داده شده و مکان به دست آمده، مشخص کننده خطای مکانی تست سینماتیک معکوس است. این خطا در نمودار شکل ۶ نشان داده شده است. این اعمال در برنامه شبیه ساز توسط تابع InverseKinematics انجام می شود. این تابع برای شبیه سازی سینماتیک معکوس ربات 6R، به عنوان ورودی یک ماتریس ۴ در ۴ را جهت مشخص نمودن مکان و جهت end-effector دریافت می کند و از روی آن شش زاویه مفصل های ربات را به دست می آورد. هم چنین در صورتی که نقطه داده شده خارج از محدوده دسترسی ربات باشد، کاربر را با دادن هشدار مطلع می سازد.



a

b

شکل ۸: (a) پنجره ورود اطلاعات در تست سینماتیک معکوس
(b) پنجره اطلاعات خروجی در تست سینماتیک معکوس



شکل ۹: نمودار خطای مکانی در سینماتیک معکوس

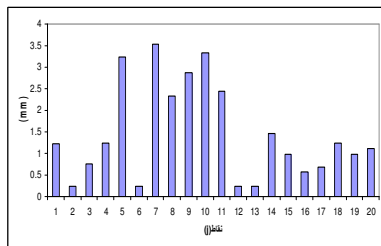
۵ آزمون اندازه گیری عملکرد ربات بر اساس استاندارد بین المللی آمریکا-ANSI-RIA R15.05-2

برای بررسی عملکرد ربات، روش های استاندارد گوناگونی وجود دارد که عملکرد مختلف ربات را ارزیابی می نماید.

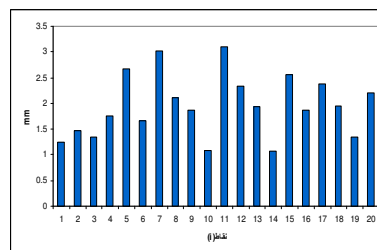
هدف این استاندارد مهیا کردن اطلاعات تکنیکی برای کمک به کاربران در راستای انتخاب مناسب ترین ربات برای کار مورد نظر می باشد [10]. این استاندارد، معیارهای مهمی بر مبنای مسیر را تعریف می نماید و سپس روش هایی برای ارزیابی آن ها پیش رو قرار می دهد. این معیارها عبارتند از: دقت نسبی مسیر، دقت مطلق مسیر، قابلیت تکرار مسیر، خصوصیات سرعت مسیر و انحراف گوشه ای. ارزیابی معیارهای ذکر شده یکی از مناسب ترین روش ها برای ارزیابی کل کارآیی بر مبنای مسیر روی ربات های صنعتی می باشد. همچنین، محاسبه این معیارها امکان مقایسه کارآیی ربات های مشابه را بوجود می آورد.

انجام تست های عملکرد ربات، از دو دوربین که در محیط قرار داده شده استفاده شده و به وسیله تست سینماتیک مستقیم، نقاط متناظر نقاط مطلوب به دست می آید. که نمودار این نقاط نشان دهنده دقت حرکت مسیر ربات است.

همچنین مقادیر پارامترهای تست برای ربات 6R برای ۲۰ نقطه با ده بار تکرار بدست آمده اند: پارامتر ماکزیمم دقت مسیر برابر $AC = 0.633451$ به دست آمده است. میانگین دقت مسیر نیز برابر $AC = 0.267432$ محاسبه می گردد. مقادیر \bar{u}_{a_j} و \bar{v}_{a_j} نیز محاسبه شده که هیستوگرام آن ها در شکل های ۱۱ و ۱۰ قابل مشاهده می باشد.

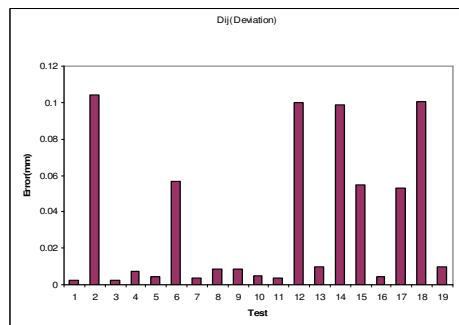


شکل ۱۰: نمودار مراکز ثقل طول



شکل ۱۱: نمودار مراکز ثقل عرض

مقادیر D_{ij} انحراف مسیر نیز محاسبه شده و هیستوگرام مربوطه در شکل (۱۲) رسم گردیده است. با استفاده از مقادیر به دست آمده برای D_{ij} ماکزیمم تکرار پذیری برابر $PR = 0.00034$ و میانگین تکرار پذیری برابر $PR = 0.005943$ بدست آمده است.



شکل ۱۲: نمودار میزان انحراف D_{ij} برای ربات 6R

همچنین مقادیر CR (خطای انحراف گوشه) و CO (پرتاب گوشه ای) برای مسیر مستطیلی شکل به صورت زیر به دست آمده اند:

$$CO=0.240985 \text{ mm}$$

$$CR=0.0650 \text{ mm}$$

۶ نتیجه گیری

در این نوشتار چگونگی استفاده از یک سیستم بینایی برای کنترل یک ربات 6R شبیه سازی گردید. یک نسخه تلفیقی از روش های مختلف کنترل بینایی در نرم افزار شبیه سازی کننده تولید و معرفی شده که این نرم افزار با استفاده از محیط گرافیکی سه بعدی توانسته تا حد زیادی نزدیک به واقعیت عملیات کنترل ربات توسط بینایی را انجام دهد. امکان استفاده از امکانات فضای سه بعدی در حرکت و چرخش هر یک از اشیاء محیط، به درک موقعیت و آشنائی با محیط سه بعدی کمک فراوانی می کند. فضای کاری ربات نیز جهت سهولت کار و عدم دریافت نتایج غلط، مشخص شده است. از تست های عملکردی سینماتیک ربات استفاده شده و حرکات ربات در سینماتیک مستقیم و معکوس مورد آزمایش قرار گرفته است. برای انجام تست های عملکردی با توجه به استانداردهای موجود، نیاز به بکار گیری دو دوربین قرار گرفته در محیط، جهت پردازش اطلاعات و به دست آوردن موقعیت پنجه نسبت به مختصات مرجع (زمین) است که در واقع دو دوربین به کمک هم این کار را انجام داده اند. از میان استانداردهای موجود در این پروژه استاندارد ANSI-RIA پرداخته شده و ضوابط تعیین شده در استاندارد ANSI-RIA (که استاندارد بین المللی آمریکا می باشد) به تجزیه و تحلیل آماری و علمی عملکرد ربات پرداخته شده است.

منابع

- [۱] آسادا و اسلوتاین، ترجمه دکتر محرم حبیب نژاد کورایم، خرداد ۱۳۷۶، تجزیه و تحلیل و کنترل ربات، انتشارات دانشگاه علم و صنعت ایران.
- [۲] شبیه سازی بینایی و ارابه الگوریتم نرم افزاری پردازش تصویر و بکارگیری آن در ربات، ۱۳۷۸، هادی علی اکبر پور، دانشگاه آزاد اسلامی واحد علوم و تحقیقات، پایان نامه کارشناسی ارشد.
- [3] Heidari, Fatemeh.S. , October 2006, Simulation of Visual Servo Control and Performance Tests of the 6R Robot Using Position Based and Image Based Approach, Iran University of Science and Technology
- [4] Geregory Flandin, 2000, Eye-in-hand/Eye-to-hand cooperation for Visual Servoing, IEEE Press.
- [5] J. T. Feddema, C.S. G Lee, O. R. Mitchell, 1991, Weighted Selection of image Features for resolved Rate Visual feedback Control, IEEE Transaction on Robotics and Automation, Vol. 7, No 1, pp. 31-47
- [6] Seth Hutchinson, Gregory Hanger, and Peter I. Croke, May 1996, A Tutorial on Visual Servo Control, pp.6-12.
- [7] Peter I. Croke, Malcom C. Good, Oct 1996, Dynamic Effects in Visual Closed Loop Systems, IEEE Trans. on Robotics and Automation., Automation, pp. 671-683.
- [8] S.Skaar, W.Brockman, and R.Hanson, 1987, Camera-Space Manipulation. Int.J. Robot. Res., 6(4), pp.20. 32.
- [9] H.Hashimoto, T.Kimoto, T.Ebin, Manipulator Control with Image Based Visual Servoing, In Proc. IEEE, Conf. Robotics and Automation , pp.2267-2272, 1991.
- [10] American National Standard for Industrial Robots and Robot Systems Path-Related and Dynamic Performance Characteristics Evaluation. ANSI/RIA R15.05-2. Apr. 16 2002