

**Research Paper**

# Localization of Aerial Robot Based on Fisheye Cameras in a Virtual Lab

Mohammad Ali Amiri Atashgah<sup>1\*</sup>, and S. Mohammad Jafar Tabib<sup>2</sup>

1, 2. Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran

\* atashgah@ut.ac.ir

*This research represents the localization of an aerial robot using fisheye cameras on walls in a simulation environment. The virtual testbed in this work is a quadrotor simulated in MATLAB Simulink. Subsequently, the simulation outputs as flight records are used in a virtual lab developed in 3DsMAX. Then, the virtual fisheye cameras (here two) are installed at different points on the walls, and the related images from the cameras are received offline. Open CV will process the gathered images in a C++ environment. For external calibration, each fisheye camera takes an image from a known pattern consisting of some lights placed in the virtual lab. We execute the Perspective-n-Point method on the images to obtain the pierce direction/position of the camera. For more, the aerial robot is localized by computing the nearest point between two lines of sight. In brief, the outcomes exhibit an accuracy of 4cm in the center of the virtual-room room.*

**Keywords:** Virtual environment, Visual navigation, Fisheye calibration, Fisheye camera, Perspective-n-point, Aerial robot.

---

1. Associate Professor (Corresponding Author)  
2. PhD Student

مقاله علمی - پژوهشی

# پیاده‌سازی الگوریتم در حلقه سامانه موقعیت‌یابی با دوربین‌های چشم‌ماهی

محمدعلی امیری آتشگاه<sup>۱\*</sup> و سیدمحمدجعفر طیب<sup>۲</sup>

۱ و ۲- دانشکده علوم و فنون نوین، دانشگاه تهران، تهران، ایران

\* atashgah@ut.ac.ir

در این مقاله رویکرد مکان‌یابی یک ربات پرنده در یک فضای محدود توسط دوربین‌هایی با دید وسیع ارائه می‌شود. در ابتدا شبیه‌سازی دینامیکی ۶ درجه آزادی وسیله در محیط Simulink انجام شده و حرکت آن در یک محیط مجازی در 3DsMAX پیاده‌سازی می‌شود. در این محیط تعدادی دوربین چشم‌ماهی در چند نقطه مستقر شده و تصاویر دریافتی آنها از پرواز ربات پرنده به صورت برون‌خط توسط کتابخانه‌های OpenCV تجزیه و تحلیل می‌شوند. سپس به منظور کالیبراسیون نرونی دوربین‌ها از الگویی مشخص شامل تعدادی نقطه روشن که مختصات معلوم دارند تصویربرداری شده و با کمک روشی مبتنی بر نقاط پرسپکتیو، جهت و مکان خود دوربین‌ها بدست می‌آید. نتایج تحقیق حاکی از این است که با دقتی در حدود ۴ سانتیمتر می‌توان موقعیت وسیله را تخمین زد. شایان ذکر است که از روش پیشنهادی می‌توان در ناوبری ربات‌های فضایی و مریخ‌نورد نیز بهره برد.

واژه‌های کلیدی: محیط مجازی، ناوبری تصویری، کالیبراسیون دوربین چشم‌ماهی، الگوریتم مبتنی بر نقاط پرسپکتیو (PnP)، ربات پرنده ربات پرنده.

		علائم و اختصارات	
$P_1, P_2$	ضرایب اعوجاج مماسی دوربین	$g$	شتاب جاذبه ( $ms^{-2}$ )
$f_x, f_y$	طول فاصله کانونی دوربین در راستای افقی و عمودی تصویر	$\theta$	زاویه پیچ
$C_x, C_y$	نقطه مرکز دید دوربین در تصویر	$\varphi$	زاویه غلت
$torted, Y_{distorted}$	مختصات نقطه در تصویر دارای اعوجاج	$\psi$	زاویه گردش
		$R$	ماتریس تبدیل بین دو دستگاه مختصات
		$CT$	دستگاه مختصات دوربین
		$RN$	دستگاه مختصات جهانی (مرجع)
		$C$	مختصات محل حضور دوربین
		$\tau$	دستگاه مختصات دوربین
		$\eta$	دستگاه مختصات جهانی
		$k_1, k_2, k_3$	ضرایب اعوجاج شعاعی دوربین

## مقدمه

امروزه ربات‌های پروازی در سطح زمین و یا سیاره دارای طیف گسترده‌ای از مأموریت‌های بعضاً حیاتی هستند که بخش مهمی از آنها در فضای داخلی تعریف می‌شود؛ با توجه به اهمیت و لزوم دقت ناوبری در فضای داخلی مسئله ناوبری ربات‌های پرنده حائز اهمیت ویژه‌ای است [1]. این پژوهش از آنجا اهمیت دارد که می‌توان با کمک روش ارائه شده، سامانه‌های ناوبری یک ربات پرنده مانند سامانه اینرسی تلفیق شده با GNSS را از طریق مقایسه موقعیت بدست آمده از دوربین‌های چشم‌ماهی مورد ارزیابی قرار داد [2].

۱. دانشیار (نویسنده مخاطب)

۲. دانشجوی دکتری

گریز در تصاویر پرسپکتیو محاسبه شده و با استفاده از آن زاویه سمت و انحراف عرضی پهپاد برای ناوبری در محیط‌های داخلی بدست می‌آید. در مرجع [14] داده‌های اودومتري حاصل از یک دوربین نصب شده روی پهپاد و الگوریتم SLAM با داده‌های حاصل از شبکه UWB<sup>۳</sup> آمیخته شده اند تا وسیله بتواند در یک محیط بسته صنعتی به کار ارزیابی و تعقیب اقلام موجود بپردازد. در مرجع [15] با استفاده از یک دوربین نصب شده روی یک ربات پرنده کوچک و تعدادی نقطه شناخته شده، ناوبری در داخل فضای بسته انجام شده است. در مرجع [16] یک سنسور عمق-سنج روی ربات پرنده نصب شده و با استفاده از الگوریتم SLAM به طور همزمان نقشه برداری و مکان‌یابی انجام می‌پذیرد. در مرجع [17] با کمک تعدادی نشانه‌های زمینی در مکان‌های از پیش معین، ناوبری داخلی یک ربات پرنده مجهز به دوربین، پیاده‌سازی شده و برای صحت سنجی آن نیز ۳ دوربین ثابت در آزمایشگاه نصب شده و داده‌های آنها با هم یکپارچه‌سازی شده است.

همانگونه که در منابع مختلف بررسی شده است، ناوبری در محیط داخلی و یا خارجی اغلب با تکیه بر کشف الگوهای موجود در محیط خارجی و یا شناسایی نمودن فانوس‌های فعال یا غیر فعال انجام می‌پذیرد. همچنین روش‌های پیشنهادی بر مبنای چند دوربین با زاویه دید کوچک انجام می‌شود. در انجام این پژوهش از لنز چشم‌ماهی کالیبره شده استفاده شده است؛ با استفاده از تعدادی دوربین چشم‌ماهی ثابت شده روی دیوارهای اتاق (حداقل دو عدد)، شیء پرنده بدون اینکه نیازی به تعلقات مکانیکی برای چرخاندن دوربین‌ها داشته باشد ردیابی می‌شود. همچنین در این تحقیق برای اثبات توانمندی روش پیشنهادی، یک سامانه مجازی با قابلیت مدل سازی دوربین‌های با زاویه دید بالای ۱۸۰ درجه (چشم ماهی) توسعه داده شده است که در ابزار شبیه سازی‌های موجود این قابلیت وجود ندارد. شایان ذکر است که هم اکنون فعالیت‌های متنوعی در زمینه روش پیشنهادی در این مقاله در مجامع علمی همچنان در حال انجام است که خود تاییدکننده بروز بودن و باز بودن موضوعات تحقیقاتی در این عرصه است [18] [19] [20].

## کالیبراسیون دوربین چشم ماهی

برای هر دوربین (در اینجا از نوع چشم‌ماهی) دو نوع از ویژگی‌ها بیان می‌گردد: ویژگی‌های درونی و ویژگی‌های برون. ویژگی‌های درونی شامل کیفیت ساخت و میزان پارامترهای اعوجاج دوربین و همچنین مکان قرارگیری ۲ بعدی نقطه کانونی و نقطه دید دوربین می‌باشد. ویژگی برون نیز شامل جهت قرارگیری و مکان قرارگیری دوربین در دستگاه مختصات خارجی می‌باشد. در مورد هر نوع از این ویژگی‌ها اصطلاحاً عمل کالیبراسیون باید انجام پذیرد تا

این راستا در مرجع [3] از فانوس‌های غیر فعال متشکل از ۳ رنگ مختلف استفاده شده تا بتوان به راحتی در سلسله تصاویر ویدئوی ثبت شده توسط دوربین ربات، فرآیند موقعیت‌یابی را انجام داد. در ابتدا ربات، فانوس‌ها را که هرکدام دارای کد و مکان مشخص هستند نسبت به دوربین خود یافته و سپس موقعیت خود را در نقشه مطلق پیدا می‌نماید. در مرجع [4] یک سیستم ناوبری یکپارچه پیاده سازی شده که در درون ساختمان از فانوس‌های کم مصرف بلوتوث<sup>۴</sup> و در خارج ساختمان از سیستم GNSS استفاده می‌نماید. مرجع [5] درباره موقعیت‌یابی یک ربات با اسکنر ارزان قیمت ۳ بعدی می‌باشد که در یک فضای داخلی با استفاده از الگوریتم SLAM<sup>۵</sup> از اتاق‌ها نقشه برداری می‌نماید و از این نقشه‌ها برای ناوبری در حضور بعدی خود استفاده می‌نماید. در مرجع [6] با استفاده از بینایی استریو<sup>۶</sup>، روشی سریع و واکنشی برای جلوگیری از برخورد یک ریزپهپاد با موانع پیاده سازی شده است. مرجع [7] با استفاده از فانوس‌های مغناطیسی فعال که در آن هر کدام امضای مخصوص خود را دارد، روشی را برای ناوبری گوشه‌های هوشمند در فضای داخلی ساختمان، ارائه داده است. در مرجع [8] با استفاده از سنسورهای مسافت‌یاب لیزری در امتداد صفحه و سامانه ناوبری اینرسی، یک سیستم ناوبری متن-باز<sup>۷</sup> برای یک ربات پرنده پیاده‌سازی شده که قابلیت نقشه برداری ۲ بعدی و ۳ بعدی برای استفاده در ناوبری SLAM را دارد. در مرجع [9] ناوبری ربات زمینی با تلفیق سامانه‌های INS و Wi-Fi توسط فیلتر کالمن<sup>۸</sup> و فیلتر کالمن<sup>۸</sup> انجام شده است. در مرجع [10] یک ربات زیرسطحی برای نقشه برداری از تونل‌های زیر آب مربوط به دوران باستان مطرح گردیده و به دلیل فقدان روش‌های معمول ناوبری، از روشی فعال مبتنی بر حرکت در نزدیکی سطوح و بررسی شکل‌ها و استفاده از الگوریتم استخراج دیوار - با فرض آنکه نقشه محیط از پیش کامل یا به صورت جزئی مشخص باشد - استفاده شده است. در مرجع [11] ناوبری یک ربات پرنده با کمک دوربین ۷۲۰ پیکسل نصب شده روی آن و در محیط از قبل ناشناخته و بدون دسترسی GNSS مطرح گردیده تا بتواند با کمک نشانه‌های زمینی به یک نقطه هدف از قبل مشخص شده در محیط برسد. در مرجع [12] با استفاده از یکپارچه‌سازی حسگر لیزری و اینرسی، کار نقشه‌برداری و مکان‌یابی یک پهپاد در محیط ناشناخته و دسته‌بندی نمودن خط لوله‌کشی‌های موجود در آنجا انجام پذیرفته است. در مرجع [13] با تجزیه تحلیل ساختارهای موجود در تصاویر دریافتی دوربین یک ریزپهپاد، نقطه

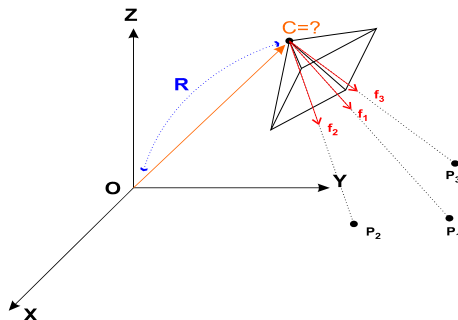
3. Beacon  
4. Bluetooth  
5. Simultaneous localization and mapping  
6. Stereo  
7. Open-Source  
8. Adaptive Kalman Filter

9. Ultra-Wide Band

کالیبراسیون، تعدادی تصویر نمونه (کمتر از ۱۰ عدد) نیاز است که در بردارنده یک ساختار از پیش تعریف شده واضح و ساده و خوب باشند.

### کالیبراسیون برونی با الگوریتم P3P

همانطور که گفته شد ویژگی های برونی هر دوربین شامل جهت و مکان دوربین در دستگاه جهانی می باشد که امکان سنجش دقیق این ویژگی ها به سادگی امکان پذیر نیست. روشی که برای حالت مجهول بودن جهت و مکان دوربین پیشنهاد شده است، مبتنی بر شناسایی چیدمان تعدادی نقطه مرجع بر روی کف زمین است. این مسئله که چگونه می توان موقعیت و زوایای دوربین را از روی نقاط از پیش معلوم شده بدست آورد تحت عنوان PnP مطرح است [21]. در این پژوهش از راه حل ارائه شده توسط کینپ [22] استفاده شده که به صورت فرم بسته می باشد و در یک مرحله، مستقیماً مکان و جهت دوربین را در مختصات جهانی به عنوان تابعی از مختصات نقاط موجود در تصویر آنها محاسبه می نماید. لازم به ذکر است که کارایی روش پیشنهادی در حد روش ارائه شده در مرجع [23] می باشد که از پرکاربردترین و مقاومترین الگوریتم در مسئله P3P می باشد. در این الگوریتم قرار است - مطابق شکل (۱)- موقعیت دقیق نقطه C و ماتریس تبدیل دستگاه (O, X, Y, Z) نسبت به دستگاه دوربین R محاسبه گردند. مختصات نقاط  $P_1, P_2$  و  $P_3$  نیز در دستگاه جهانی داده شده است. همچنین فرض بر این است ویژگی های داخلی دوربین نیز معلوم بوده و کالیبره شده است. بنابراین ۳ بردار  $f_1, f_2, f_3$  ارائه شده در شکل (۱) نیز در دستگاه دوربین موجود هستند. در این الگوریتم مطابق شکل (۲)، دو دستگاه واسط به نام های T و  $\Pi$  تعریف می گردند که دستگاه دوربین (۳) با ماتریس T به دستگاه T و دستگاه  $\Pi$  نیز با ماتریس N به دستگاه مرجع یا همان (O) تبدیل می گردد. لازم به ذکر است شرط وجود داشتن دستگاه واسط  $\Pi$  عدم هم خط بودن ۳ نقطه  $P_1, P_2$  و  $P_3$  می باشد. لازم به ذکر است که اساس کار الگوریتم P3P بر پایه یافتن تبدیل بین دستگاه  $\Pi$  و T استوار است. به این منظور یک نیم صفحه مطابق شکل (۳) به نام  $\Pi$  تعریف می گردد.



شکل ۱- مسئله P3P در حالی که نقاط معلوم اند

ویژگی ها دانسته شوند و بدین ترتیب مختصات نقاط موجود در تصاویر بدست آمده از دوربین قابل تفسیر شوند.

### کالیبراسیون داخلی

در تصاویر بدست آمده از دوربین های چشم ماهی معمولاً دو نوع اعوجاج به شرح زیر وجود دارد که طبیعتاً شدت این اعوجاج ها بستگی به کیفیت ساخت دوربین و تعداد المان های اپتیکی آن دارد.

#### الف- اعوجاج شعاعی

بر اثر این نوع اعوجاج خطوط صاف در صحنه واقعی به شکل خمیده در تصویر ظاهر می گردند و هر چه از نقطه مرکزی تصویر به اطراف تصویر حرکت شود میزان این اعوجاج شدیدتر خواهد شد. مدل ریاضی این اعوجاج از درجه ۶ است و به فرم زیر تعریف می گردد [21]:

$$x_{\text{distorted}} = x(1 + k_1r^2 + k_2r^4 + k_3r^6) \quad (1)$$

$$y_{\text{distorted}} = y(1 + k_1r^2 + k_2r^4 + k_3r^6) \quad (2)$$

#### ب- اعوجاج مماسی

این نوع اعوجاج در دوربین های با کیفیت پایین وجود دارد و ناشی از این است که عدسی دوربین با صفحه فرضی تصویر تراز نشده باشد و به همین جهت ممکن است برخی نواحی تصویر نزدیکتر از آنچه که هستند به نظر بیایند. مدل ریاضی این نوع اعوجاج به شکل زیر تعریف می گردد [21]:

$$x_{\text{distorted}} = x + (2p_1xy + p_2(r^2 + 2x^2)) \quad (3)$$

$$y_{\text{distorted}} = y + (p_1(r^2 + 2y^2) + 2p_2xy) \quad (4)$$

با توجه به مدل های اعوجاج ذکر شده در روابط (۱) الی (۴) برای تصحیح اعوجاج های یک تصویر و کالیبره کردن آن باید این ضریب پیدا شوند  $(k_1, k_2, p_1, p_2, k_3)$ . همچنین باید برخی پارامترهای دیگر نیز مانند پارامترهای درونی و برونی دوربین بدست آیند. پارامترهای درونی اختصاص به خود دوربین دارند و اگر یکبار به درستی محاسبه شوند برای تمامی تصاویر دریافت شده از آن دوربین قابل استفاده هستند که به مجموعه این ضرایب، ماتریس دوربین نیز گفته می شود:

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} = \text{ماتریس دوربین} \quad (5)$$

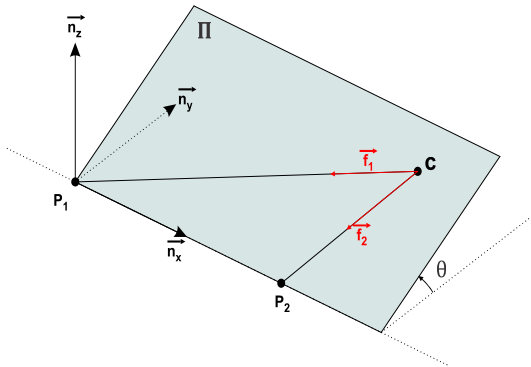
شایان توجه است که پارامترهای  $(f_x, f_y)$  در واقع نشان دهنده طول فاصله کانونی دوربین در راستای افقی و عمودی در صفحه تصویر می باشند. پارامترهای  $(c_x, c_y)$  نیز بیانگر مختصات پیکسلی نقطه ای در تصویر است که طبق برآوردهای توابع کالیبراسیون، دوربین به آن نقطه نگاه مستقیم می کند. به منظور انجام عمل

$$P_3^{\eta} = Q(\alpha, \theta) \cdot (P_3^{\eta} - C^{\eta}(\alpha, \theta)) = \begin{bmatrix} -\cos \alpha \cdot p_1 - \sin \alpha \cos \theta \cdot p_2 + d_{12}(\sin \alpha \cdot b + \cos \alpha) \\ \sin \alpha \cdot p_1 - \cos \alpha \cos \theta \cdot p_2 \\ -\sin \theta \cdot p_2 \end{bmatrix} \quad (8)$$

در نهایت می‌توان با بسط رابطه ۸، زاویه  $\theta$  را بر حسب  $\alpha$  نوشت و در نهایت به معادله درجه ۴ به فرم زیر رسید [22]:

$$a_4 \cdot \cos^4 \theta + a_3 \cdot \cos^3 \theta + a_2 \cdot \cos^2 \theta + a_1 \cdot \cos \theta + a_0 = 0 \quad (9)$$

پس از محاسبه ۴ جواب (معمولاً حقیقی) این معادله، ۴ مجموعه جواب شامل مختصات و جهت دوربین بدست می‌آید. البته باید حداقل از یک نقطه دیگر علاوه بر ۳ نقطه موجود کمک گرفت تا یک پاسخ به عنوان پاسخ نهایی و درست انتخاب گردد.

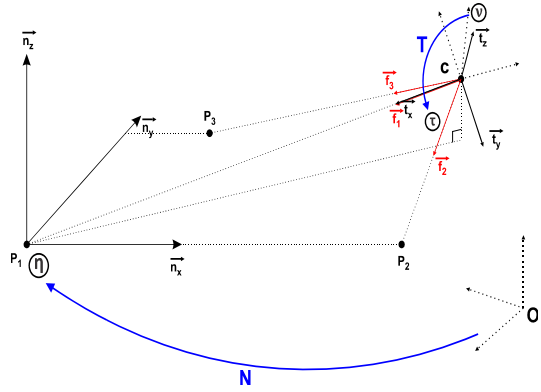


شکل ۴- دوران صفحه  $\Pi$  حول محور  $\vec{n}_x$  به اندازه  $\theta$

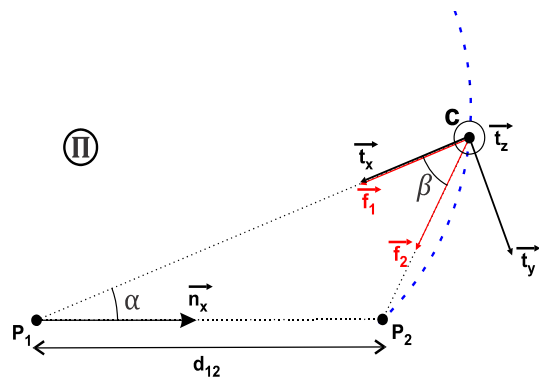
### پیاده‌سازی الگوریتم P3P و تحلیل تصاویر

برای تعیین مکان و جهت دوربین‌های ثابت روی دیوار از یک الگوی ثابت و با مکان مشخص از نقاط نورانی مطابق شکل (۵) استفاده می‌شود. با توجه به اینکه فقط با ۳ نقطه نمی‌توان به پاسخ قطعی رسید، الگوریتم P3P سه مرتبه بر روی ۳ دسته مختلف ۳-تایی نقاط پنجگانه اجرا می‌گردد تا علاوه بر رفع ابهام، افزونگی و افزایش دقت نیز ایجاد شود. همچنین در ادامه، شکل (۶)، نمایی از زاویه دید دو دوربین که تصاویرشان کالیبره شده است را به تصویر می‌کشد.

الگوریتم کاربردی برای یافتن نقطه شاخص این است که ابتدا مختصات ۲ بعدی برای ۵ نقطه به شکل پادساعت‌گرد مرتب شوند تا ساختار ۵ ضلعی در صفحه تصویر (و نه مختصات واقعی) شناسایی شود؛ با این فرض که نقطه شاخص همان رأس با زاویه ۱۸۰ درجه خواهد بود. به منظور پیدا نمودن نقطه شاخص از بین ۵ نقطه موجود، ابتدا باید رئوس ۵ ضلعی را به صورت ساعت‌گرد مرتب نمود. به همین منظور یک تابع جهت مرتب سازی نقاط تعریف شده است که نقاط



شکل ۲- دستگاه واسط دوربین ( $\tau$ ) و دستگاه واسط جهانی ( $\eta$ )



شکل ۳- نیم صفحه  $\Pi$  و مکان هندسی محتمل برای نقطه  $C$

همانگونه که در شکل (۳) قابل مشاهده است، از آنجا که ضلع  $d_{12}$  و زاویه  $\beta$  هر دو معلوم هستند، بنابراین مکان هندسی نقطه  $C$ ، کماتی از یک دایره خواهد بود که با خطچین آبی رنگ در آن نمایش داده شده است. به منظور پارامتری نمودن موقعیت نقطه  $C$  بر حسب زاویه  $\alpha$ ، از قانون سینوس‌ها به شکل زیر استفاده می‌شود [22]:

$$\frac{\|CP_1\|}{d_{12}} = \frac{\sin(\pi - \alpha - \beta)}{\sin \beta} \quad (6)$$

و در نهایت مختصات نقطه  $C$  در صفحه  $\Pi$  به صورت زیر حاصل می‌شود [22]:

$$\Rightarrow C^{\Pi}(\alpha) = \begin{pmatrix} d_{12} \cos \alpha (\sin \alpha \cdot \cot \beta + \cos \alpha) \\ d_{12} \sin \alpha (\sin \alpha \cdot \cot \beta + \cos \alpha) \\ 0 \end{pmatrix} \quad (7)$$

سپس برای اینکه نیم صفحه  $\Pi$  را بتوان به دستگاه  $\eta$  ارتباط داد زاویه  $\theta$  مطابق شکل (۴) تعریف می‌گردد.

در نهایت ماتریس تبدیل  $Q$  برای تبدیل دستگاه  $\eta$  به  $\tau$  به صورت  $Q(\alpha, \theta)$  خواهد بود. حال نقطه  $P_3^{\eta} = [p_1 p_2 0]^T$  را می‌توان به شکل زیر محاسبه نمود:

و هم مکان نقاط الگو مشخص هستند، بنابراین بردار متصل کننده دوربین به نقاط الگو نیز در دستگاه مختصات جهانی (در اینجا متصل به اتاق) مشخص است. در نتیجه بردارهای متصل کننده دوربین به نقاط در هر دو دستگاه دوربین و جهانی مشخص هستند و فقط باید ماتریس تبدیل بین این دو بدست آید. فرض می‌نماییم ماتریس تبدیل دستگاه مختصات اتاق به دوربین به شکل رابطه (۱۰) باشد. در ادامه حرف  $C$  نماد دستگاه دوربین و حرف  $R$  نماد دستگاه مرجع متصل به اتاق (دستگاه جهانی) است.

$$R_C^R = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \quad (10)$$

همانطور که گفته شد، در این بخش فرض می‌کنیم مکان دوربین و  $n$  نقطه از قبل معلوم می‌باشند و همچنین بردار یکه مختصات این نقاط در دستگاه دوربین نیز مشخص می‌باشد. به این ترتیب بردارهای  $V^C_1$  الی  $V^C_n$  همان بردارهای متصل از دوربین به نقاط  $n$  گانه در دستگاه مختصات دوربین می‌باشد و بردارهای  $V^R_1$  الی  $V^R_n$  نیز نظیر آنها در دستگاه مختصات اتاق هستند. مؤلفه‌های هر یک از این بردارها نیز با  $V^R_{3x}, V^R_{3y}$  و  $V^R_{3z}$  بیان می‌شود. از آنجا که ماتریس  $R_C^R$  در واقع بیانگر جهت گیری دوربین است، بنابراین  $V^R_j = R_C^R \times V^C_j$  می‌باشد. در ادامه برای بردار  $z$ -ام داریم:

$$\begin{bmatrix} V^R_{jx} \\ V^R_{jy} \\ V^R_{jz} \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \times \begin{bmatrix} V^C_{jx} \\ V^C_{jy} \\ V^C_{jz} \end{bmatrix} \quad (11)$$

و با در نظر گرفتن روابط مربوط به تمام  $n$  نقطه خواهیم داشت:

$$\begin{bmatrix} V^R_{1x} \\ \vdots \\ V^R_{nx} \end{bmatrix} = \begin{bmatrix} V^C_{1x} & V^C_{1y} & +V^C_{1z} \\ \vdots & \vdots & \vdots \\ V^C_{nx} & V^C_{ny} & +V^C_{nz} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad (12)$$

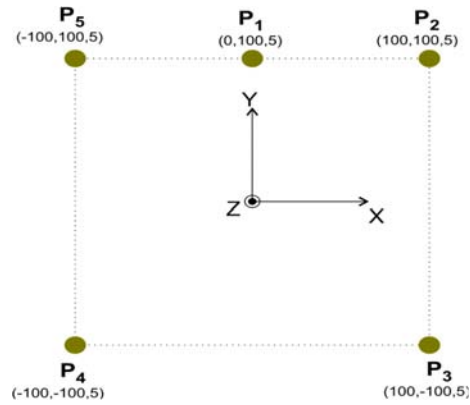
در رابطه (۱۲) اگر ماتریس  $A$  را  $\begin{bmatrix} V^C_{1x} & V^C_{1y} & +V^C_{1z} \\ \vdots & \vdots & \vdots \\ V^C_{nx} & V^C_{ny} & +V^C_{nz} \end{bmatrix}$  بنامیم که ماتریسی معلوم است آنگاه خواهیم داشت:

$$U \triangleq \begin{bmatrix} V^R_{1x} \\ \vdots \\ V^R_{nx} \end{bmatrix} = A \times \begin{bmatrix} a \\ b \\ c \end{bmatrix} \quad V \triangleq \begin{bmatrix} V^R_{1y} \\ \vdots \\ V^R_{ny} \end{bmatrix} = A \times \begin{bmatrix} d \\ e \\ f \end{bmatrix} \quad W \triangleq \begin{bmatrix} V^R_{1z} \\ \vdots \\ V^R_{nz} \end{bmatrix} = A \times \begin{bmatrix} g \\ h \\ i \end{bmatrix} \quad (13)$$

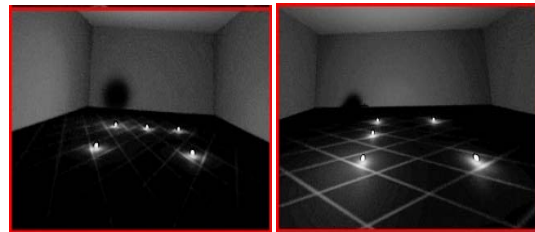
که در آن  $U, V, W$  و ماتریس‌هایی معلوم هستند و فقط  $\begin{bmatrix} a \\ b \\ c \end{bmatrix}, \begin{bmatrix} d \\ e \\ f \end{bmatrix}, \begin{bmatrix} g \\ h \\ i \end{bmatrix}$  مجهول هستند. با توجه به غیرمربع بودن ماتریس  $A$ ، برای یافتن وارون آن باید از عملگر شبه‌وارون استفاده نمود:

$$A^+ = (A^T A)^{-1} A^T \quad (14)$$

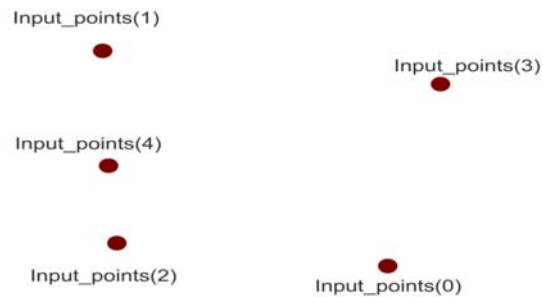
نامرتب را گرفته و به صورت مرتب شده تحویل می‌دهد. به عنوان مثال خروجی این تابع برای نقاط معرفی شده در شکل (۷) به صورت  $\{0, 2, 4, 1, 3\}$  می‌باشد.



شکل ۵- نقاط الگوی پنج‌گانه



شکل ۶- تصویر مورد استفاده در کالیبراسیون برون دوربین‌ها



شکل ۷- نمایی از چیدمان نقاط

به دلیل اینکه حجم محاسبات الگوریتم P3P زیاد است و ممکن است در شرایط بهنگام، هر سخت افزاری نتواند محاسبات مربوطه را انجام دهد، در این پژوهش الگوریتمی ساده‌تر، صرفاً به منظور محاسبه جهت دوربین ارائه می‌شود که فرض اساسی در آن دانستن موقعیت دقیق دوربین می‌باشد. اساس عملکرد این الگوریتم که در اینجا از آن با برچسب‌های "روش دوم" و "Estimate DCM" نام برده خواهد شد، بر این اصل استوار است که چون هم مکان دوربین

و در نتیجه :

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = A^+U \quad \begin{bmatrix} d \\ e \\ f \end{bmatrix} = A^+V \quad \begin{bmatrix} g \\ h \\ i \end{bmatrix} = A^+W \quad (15)$$

بدین ترتیب ماتریس دوران  $R_C^R$  محاسبه می‌شود.

## پیاده‌سازی الگوریتم مکان‌یابی

تا اینجا راهکارهای لازم برای کالیبراسیون درونی و برون‌ی دوربین و روش‌های بدست آوردن ماتریس تبدیل بین چارچوب مختصات اتاق و دوربین بیان گردید. در ادامه روند مکان‌یابی نقاط مرجع روی ربات پرنده که دارای موقعیت مجهول است، توسط دوربین‌های چیده شده در اتاق ( شکل ۸) بیان می‌شود.

### روند تطبیق نقاط

در صورتی که دوربین‌ها از نظر ویژگی درونی کالیبره شده باشند، می‌توان هر پیکسل از تصاویر آنها را به عنوان یک بردار ۳-بعدی در دستگاه آن دوربین در نظر گرفت. در این پژوهش فرض بر این است که کلیه اجسام و نشانه‌هایی که قرار است مورد کشف و تعقیب قرار بگیرند، نقاطی نورانی در اتاقی تاریک هستند. به منظور یافتن این نقاط، ابتدا باید تصویر دریافتی از حالت رنگی (RGB) به حالت طیف خاکستری تبدیل گردد؛ B، G و R به ترتیب شدت نور آبی، سبز و قرمز در هر پیکسل تصویر رنگی می‌باشند [24]:

$$Y = 0.299 \times R + 0.587 \times G + 0.114 \times B \quad (16)$$

سیس تصویر و یا قطعه تصویر مورد نظر آستانه‌گذاری می‌شود و یک تصویر باینری<sup>۱</sup> بدست می‌آید که باید نقاط یا توده نقاط موجود در آن با کمک تابع تشریح شده در [25] شناسایی شوند. این تابع برای پیدا نمودن الگوها از الگوریتم ارائه شده در [26] بهره می‌برد. پس از بدست آمدن مختصات ۲ بعدی توده نقاط موجود در تصویر (مرکز این توده‌ها) می‌توان بردار یک ۳-بعدی متناظر با آنها را در دستگاه مختصات دوربین بدست آورد. بدین منظور از مدل دوربین سوراخ سوزنی<sup>۲</sup> بردار یک ۳-بعدی مختصات نقطه ۳-بعدی (P) در دستگاه دوربین (K) - که در تصویر ۲-بعدی به صورت نقطه Q دیده می‌شود - از رابطه زیر محاسبه می‌گردد:

$$\vec{V}_P^K = \begin{bmatrix} Q_x - C_x \\ -(Q_y - C_y) \times \left(\frac{F_x}{F_y}\right) \\ -F_x \end{bmatrix}, \quad \vec{v}_P^K = \frac{\vec{V}_P^K}{\|\vec{V}_P^K\|} \quad (17)$$

شایان ذکر است که چون این دستگاه مختصات در تصاویر رایانه‌ای به صورت چپگرد مورد استفاده است، بنابراین مقدار مولفه y در آن باید منفی گردد.

## محاسبات موقعیت

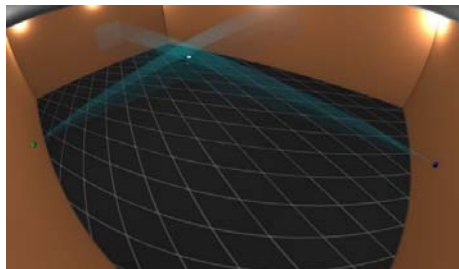
در اینجا فرض بر وجود فقط ۲ دوربین می‌باشد تا حالت حداقلی رعایت شود. اما امکان آزمایش وجود تعداد دوربین‌های بیشتر نیز در توابع فراهم گردیده است. به منظور پیدا کردن مکان ربات پرنده بایستی ابتدا ربات پرنده در تصویر هر کدام از دوربین‌ها یافت شود. سپس باید بردار یک ۳-بعدی مختصات نسبی بین ربات پرنده و دوربین‌ها در چارچوب مختصات هر کدام از دوربین‌ها بیان شود. در مرحله بعد، این بردارها به چارچوب مختصات مرجع (اتاق) منتقل می‌شوند. مطابق با روش ارائه شده در [27] برای نقاط الگو و خطوط مرتبط با المانها داریم:

$$[\sum_i [n_i * n_i^T - I]] * p = \sum_i [n_i * n_i^T - I] * a_i \quad (18)$$

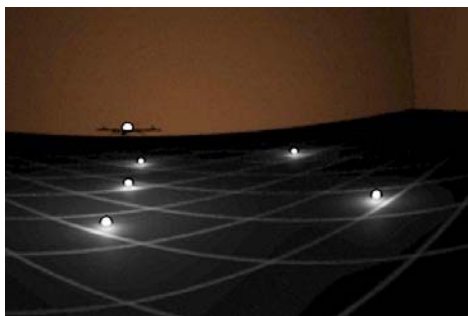
که در آن I ماتریس همانی می‌باشد و مبداء هر خط با  $a_i$  و بردار یک ۳-بعدی آن با  $n_i$  نمایش داده می‌شود. فرم کلی این معادله به صورت  $S * p = C$  است که در نتیجه داریم:

$$p = S^{-1} * C \quad (19)$$

همانطور که گفته شد، در تصاویر شبیه سازی شده، ربات پرنده دارای یک منبع نور در مرکز خود می‌باشد و اتاق نیز تاریک است (شکل ۹) که از فواصل خیلی دور فقط توده ای متمرکز از چند پیکسل دیده خواهد شد. در صورتی که در اتاق تاریک، فقط چراغ ربات پرنده در حال پرواز روشن باشد کار شناسایی آن آسان خواهد بود و تنها نقطه روشن در صحنه دریافت شده توسط هر دوربین همان ربات پرنده خواهد بود. اما حالت دیگر این است که همزمان چراغ‌های ۵-گانه الگو نیز روشن باشند تا در صورت جایجایی، امکان تصحیح بالادرنگ ماتریس تبدیل بین چارچوب دوربین و چارچوب اتاق میسر باشد.



شکل ۸- نمایی از چیدمان دوربین‌ها در محیط مجازی

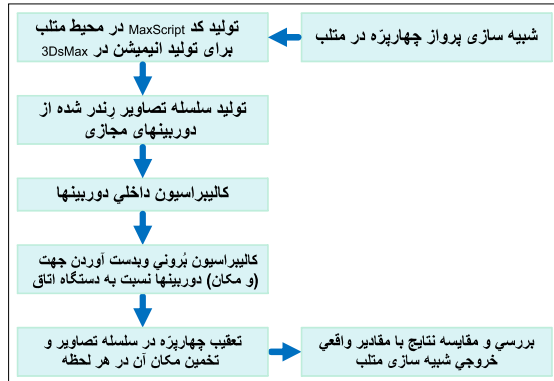


شکل ۹- تصویر شبیه سازی شده از محیط و نقاط الگو

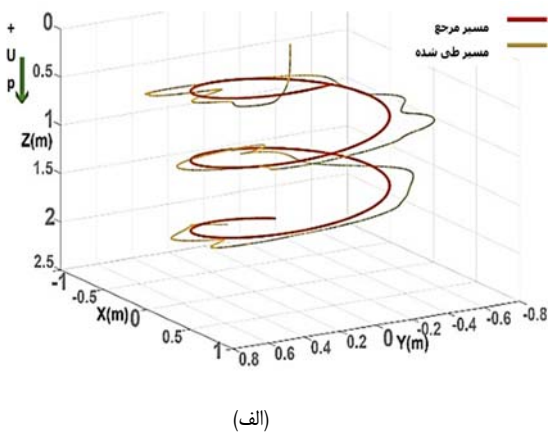
11. Binary  
12. Pinhole

## شبیه‌سازی و نتایج

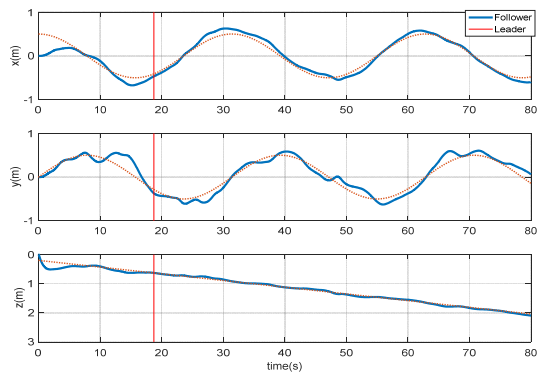
در ادامه، مقدار خطای روش  $P3P$  و "روش دوم" برای تخمین زوایای جهت دوربین را می‌توان در جدول‌های ۲ و ۳ مشاهده کرد. همچنین خطای الگوریتم  $P3P$  برای یافتن مکان دو دوربین در جدول‌های ۴ و ۵ قابل ملاحظه می‌باشد. شایان ذکر است که واحد موقعیت در اینجا سانتی‌متر می‌باشد.



شکل ۱۰- مراحل انجام شبیه‌سازی مجازی



(الف)



(ب)

شکل ۱۱- مسیر مرجع (Leader) و مسیر طی شده توسط ربات (Follower) (الف) به صورت سه-بعدی و (ب) تفکیک شده

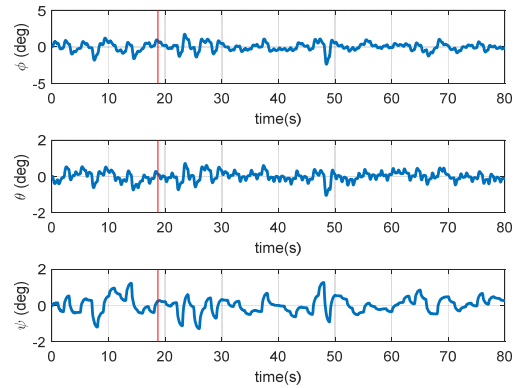
قبل از اعمال روش پیشنهادی، در ابتدا تصاویر خام و کالیبره نشده در 3DsMAX با کمک ابزار ویژه Vray که دارای توان بالایی در شبیه‌سازی صحنه‌های دوربین‌های چشم‌ماهی مطابق واقعیت دارد، تولید می‌شود. در این محیط، کلیه اجسام ثابت و متحرک و نورپردازی‌ها و همچنین نقاط الگو پیش‌بینی شده‌اند و در صورت لزوم در زمان‌های مختلف، مکان و جهت المانهای شبیه‌سازی، توسط کدهای اختصاصی maxscript تعیین می‌گردد. در ادامه، دیاگرام روند شبیه‌سازی و تولید مجازی تصاویر در شکل (۱۰) نشان داده شده است. به منظور تکمیل پروسه شبیه‌سازی، سلسله تصاویر تولید شده و داده‌های پروازی، بین نرم افزار متلب، 3DsMAX و برنامه ++C مورد تبادل واقع می‌گردد؛ به این صورت که پرواز ربات پرنده در سیمولینک متلب شبیه‌سازی می‌گردد و نتایج این شبیه‌سازی پس از انتقال به 3DsMAX توسط کدهای maxscript به محیط نمایش ارسال شده و متناسب با تغییرات در حرکت وسیله، تصویر تولید شده بروز می‌شود. پس از تولید سلسله تصاویر از منظر دوربین‌ها توسط 3DsMAX، این تصاویر مطابق آنچه پیشتر گفته شد توسط کتابخانه‌های OpenCV و در محیط برنامه ++C مورد پردازش واقع می‌شوند. همچنین برای تولید تصاویر مرتبط با پرواز ربات پرنده، ابتدا یک پرواز مارپیچی با قطر کم (حدود ۱ متر) و مدت زمان حدود ۸۰ ثانیه در مدل دینامیکی ربات پرنده در سیمولینک طراحی گردید که نتایج آن در شکل‌های (۱۱) و (۱۲) قابل ملاحظه است. توضیح اینکه تست‌های الگوریتم در حلقه صرفاً برای ۱۸ ثانیه اول پرواز انجام شده است که این محدوده زمانی با خط قرمز عمودی در شکل‌های مربوطه نمایان است.

با توجه به اینکه تخمین ماتریس دوران دوربین‌ها باید به دقت انجام شود تا مانع از خطای بعدی شود، تصمیم گرفته شد از رزولیشن ۲۴۰۰ در ۲۴۰۰ پیکسل برای تصاویر لازم برای الگوریتم‌های تخمین مکان و جهت دوربین استفاده گردد. همچنین با در نظر گرفتن این نکته که صرفاً با ۳ نقطه نمی‌توان به پاسخ قطعی رسید (معمولاً معادله درجه ۴، ۴ پاسخ مختلف دارد)، تابع  $P3P$ ، مرتبه ۳ بر روی ۳ دسته مختلف ۳-تایی از نقاط پنج‌گانه اجرا می‌گردد تا بدین ترتیب ۳ دسته جواب ۴-تایی بدست آید. جواب صحیح جوابی است که در هر ۳ دسته جواب با اندکی تفاوت ناشی از خطاها حضور داشته باشد که به منظور کاهش خطا از هر ۳ مقدار بدست آمده آن میانگین گرفته می‌شود. در شکل‌های ۱۳ و ۱۴ تصاویر اولیه تولید شده نمایش داده شده که در آن‌ها دور نقاط شاخص، یک دایره زرد رنگ ترسیم شده است.



جدول ۳- نتایج تخمین جهت دوربین شماره ۲ با روش‌های مختلف

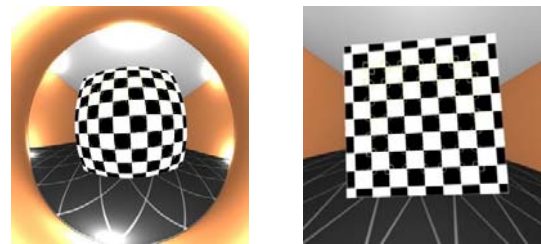
پارامتر (deg)	مقدار مرجع	نتایج روش P3P	خطای مطلق روش P3P	نتایج روش دوم	خطای مطلق روش دوم
زاویه حول محور X	۹۴/۹۸۸۴	۹۴/۹۳۵۱	-۰/۰۵۳۳	۹۴/۹۳۰۷	-۰/۰۵۷۷
زاویه حول محور Y	-۲/۷۵۸۶	-۲/۷۳۴۱	-۰/۰۲۴۵	-۲/۸۰۸۸	-۰/۰۵۰۲
زاویه حول محور Z	۰/۹۴۱۳	۰/۹۳۴۸	-۰/۰۰۶۵	۰/۸۵۰۷	-۰/۰۹۰۶
زاویه مطلق با محور X اتاق	۹۵/۰۴۵	۹۴/۹۹۰۹	-۰/۰۵۴۱	۹۵/۰۳۴۱	-۰/۱۰۰۹



شکل ۱۲- مولفه‌های جهت گیری ربات پرنده بر حسب زمان

جدول ۴- نتایج تخمین مکان دوربین شماره ۱ توسط الگوریتم P3P

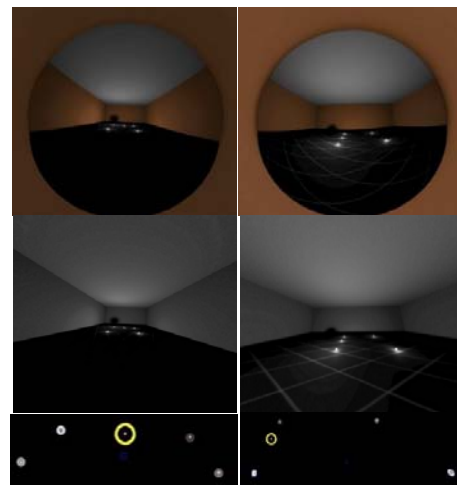
مقدار مرجع	خروجی روش P3P	خطای روش P3P
مختصات X (cm)	-۳۱۴/۶۸۳۹	-۱/۲۶۱۹
مختصات Y (cm)	۴۴/۹۷۳۷	-۰/۱۷۸۲
مختصات Z (cm)	۱۰۰/۰۰۰۰	-۰/۸۳۱۷
مقدار مطلق خطا (cm)		۱/۵۲۱۷



جدول ۵- نتایج تخمین مکان دوربین شماره ۲ توسط الگوریتم P3P

مقدار مرجع	خروجی روش P3P	خطای روش P3P
مختصات X (cm)	۱۲/۸۶۰۲	-۰/۶۵۸۴
مختصات Y (cm)	-۴۸۴/۹۵۳۹	-۰/۱۲۹۸
مختصات Z (cm)	۱۰۰/۰۰۰۰	-۰/۳۴۴۲
مقدار مطلق خطا (cm)		-۰/۷۱۴۱

شکل ۱۳- نمونه‌ای کالیبره شده از یک صفحه شطرنج مجازی

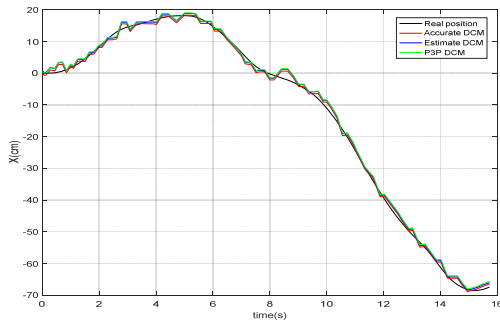


شکل ۱۴- نتیجه کالیبراسیون و کشف الگوی چراغ‌ها توسط دوربین ۱ (سمت راست) و دوربین ۲ (سمت چپ)

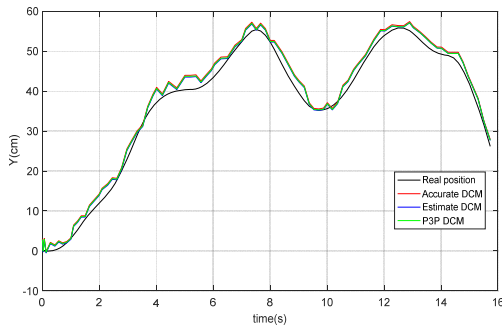
جدول ۲- نتایج تخمین جهت دوربین شماره ۱ با روش‌های مختلف

پارامتر (deg)	مقدار مرجع	نتایج روش P3P	خطای مطلق P3P	نتایج روش دوم	خطای مطلق روش دوم
زاویه حول محور X	۹۹/۹۸۷۴	۱۰۰/۰۸۴۳	/۰۹۶۹	۹۹/۹۹۶۵	-۰/۰۰۹۱
زاویه حول محور Y	۰/۹۶۰۷	۰/۹۷۹۶	/۰۱۸۹	-۰/۹۷۴۴	-۰/۰۱۳۷
زاویه حول محور Z	-۸۸/۱۳۱۶	-۸۸/۱۳۷۳	/۰۱۵۷	-۸۸/۱۳۲	-۰/۰۰۰۴
زاویه مطلق با محور X اتاق	۱۲۵/۵۵۵	۱۲۵/۶۳۵	-۰/۰۸	۱۲۵/۵۳۶	-۰/۰۱۹

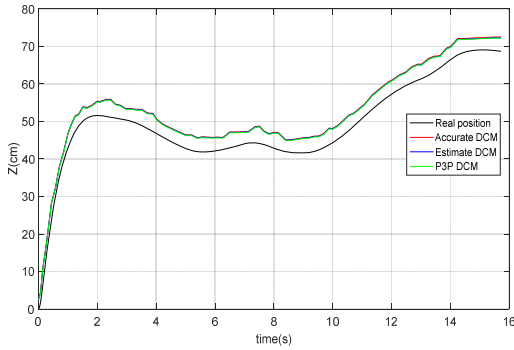
در شکل (۱۵) تعدادی از صحنه‌های متوالی پرواز ربات پرنده و شناسایی آن نشان داده شده است در شکل ۱۵- ب که مربوط به دوربین شماره ۱ است، ملاحظه می‌شود که با وجود عبور ربات پرنده از روی یکی از چراغ‌ها (از منظر دوربین) باز هم عملیات تعقیب آن با موفقیت ادامه یافته است. توضیح اینکه ابعاد پنجره تعقیب ثابت بوده و به اندازه  $\frac{1}{3}$  طول تصویر تعیین می‌گردد. همچنین نقطه مرکز پنجره نیز در هر لحظه، با فرض ثابت ماندن سرعت پیکسلی ربات پرنده در لحظه‌ی فعلی نسبت به لحظه قبل، تعیین می‌گردد و پس از شناسایی مکان ۲ بعدی ربات پرنده در تصویر، مقدار سرعت پیکسلی، برورسانی می‌شود. همچنین در صورتی که ربات پرنده در پنجره تعقیب یافت نشود، کل تصویر برای یافتن مکان ربات بررسی می‌گردد. در ادامه، نتایج خطای تخمین مکان ربات پرنده در یک مسیر برای ۶۸ صحنه پشت سر هم که مجموعاً در فاصله زمانی ۱۸/۵۷ ثانیه است، در جدول‌های ۶ الی ۸ و در شکل‌های ۱۶ الی ۱۸ قابل مشاهده است که در تصاویر ارائه شده، منحنی‌های با برچسب "Accurate DCM" در شرایط ایده‌آل که زاویه نصب و مکان دوربین‌ها به صورت پیش فرض مشخص است، رسم شده‌اند. آرایش قرارگیری دوربین‌ها در شبیه‌سازی‌ها در شکل (۱۹) نمایان



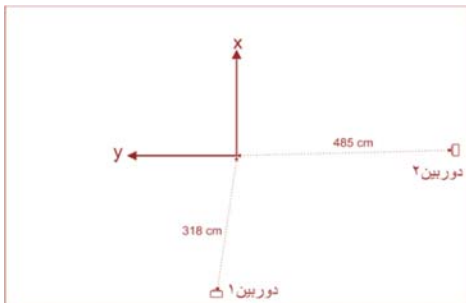
شکل ۱۶- مقایسه تخمین موقعیت ربات پرنده در بُعد X از ۳ روش مختلف



شکل ۱۷- تخمین موقعیت ربات پرنده در بُعد Y از ۳ روش مختلف

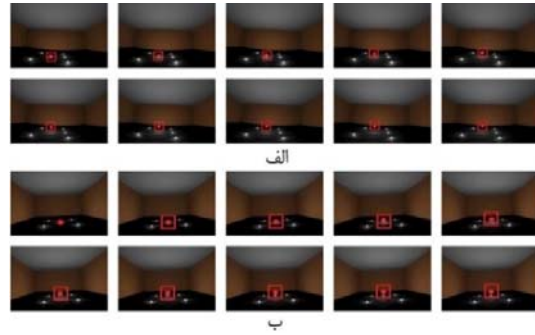


شکل ۱۸- تخمین موقعیت ربات پرنده در بُعد Z از ۳ روش مختلف



شکل ۱۹- نمایی از مکان دوربین‌ها نسبت به مبدأ مختصات در مثال شبیه‌سازی شده

است، می‌توان چنین تفسیر نمود که دوربین شماره ۱ بیشتر وظیفه کمک به تخمین مؤلفه Y از مختصات ربات پرنده و دوربین شماره ۲ وظیفه کمک به تعیین مؤلفه X از مختصات ربات پرنده را بر عهده دارد و البته هر دو دوربین در افزایش دقت محاسبات در مؤلفه Z سهیم هستند.



شکل ۱۵- تعدادی صحنه متوالی از پرواز ربات پرنده و ردیابی آن در پنجره تعقیب «الف» مربوط به دوربین ۱ و «ب» مربوط به دوربین ۲

جدول ۶- خطای تخمین مکان با معلوم بودن مکان و جهت دوربین

خطای کل (cm)	خطا در جهت محور z (cm)	خطا در جهت محور y (cm)	خطای جهت محور x (cm)	
۴/۱۱۷۲	۳/۷۱۳۳۹۵	۱/۷۷۸۱۷۶	۰/۰۲۵۹۷۳	میانگین
۴/۴۳۴	۳/۷۴۹۵۶	۲/۰۶۳۹۲	۱/۱۵۸۱۹	میانگین مربعی (RMS)
۲/۷۰۸۴	۰/۲۶۹۹	۱/۰۹۷۸	۱/۳۴۰۷	واریانس
۱/۶۴۵۷	۰/۵۱۹۵	۱/۰۴۷۷	۱/۱۵۷۹	انحراف از معیار

جدول ۷- خطای تخمین مکان با داشتن صرفاً مکان دوربین

خطای کل (cm)	خطا در جهت محور z (cm)	خطا در جهت محور y (cm)	خطای جهت محور x (cm)	
۳/۹۱۰۵	۳/۶۰۰۸	۱/۴۱۲۴	۰/۵۷۵۷	میانگین
۴/۲۳۹۸	۳/۶۳۷۳	۱/۷۵۲۵	۱/۲۹۴۲	میانگین مربعی (RMS)
۲/۶۸۴۰	۰/۲۶۴۲	۱/۰۷۶۲	۱/۳۴۳۵	واریانس
۱/۶۴۳۳	۰/۵۱۴۰	۱/۰۳۷۴	۱/۱۵۹۱	انحراف از معیار

جدول ۸- خطای تخمین مکان با الگوریتم P3P بدون داشتن مکان و جهت دوربین

خطای کل (cm)	خطا در جهت محور z (cm)	خطا در جهت محور y (cm)	خطا در جهت محور x (cm)	
۳/۸۹۵۵	۳/۴۹۸۴	۱/۵۳۸۱	۰/۷۵۵۳	میانگین
۴/۲۲۸۳	۳/۵۳۶۱	۱/۸۵۸۴	۱/۳۸۵۹	میانگین مربعی (RMS)
۲/۷۰۳۰	۰/۲۶۴۸	۱/۰۸۸۰	۱/۳۵۰۲	واریانس
۱/۶۴۴۱	۰/۵۱۴۶	۱/۰۴۳۱	۱/۱۶۱۹	انحراف از معیار

- [4] Cheng, R.-S., W.-J. Hong, J.-S. Wang, and K. W. Lin, "An Indoor Guidance System Combining Near Field Communication and Bluetooth Low Energy Beacon Technologies," *World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng.*, vol. 10, no. 9, pp. 1639–1645, 2016.
- [5] Bergeon, Y., I. Hadda, V. Krivánek, J. Motsch, and A. Štefek, "Low cost 3D mapping for indoor navigation," in *Military Technologies (ICMT), 2015 International Conference on*, 2015, pp. 1–5.
- [6] Oleynikova, H., D. Honegger, and M. Pollefeys, "Reactive avoidance using embedded stereo vision for MAV flight," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, 2015, pp. 50–56.
- [7] Sheinker, A., B. Ginzburg, N. Salomonski, L. Frumkis, B.-Z. Kaplan, and M. B. Moldwin, "A method for indoor navigation based on magnetic beacons using smartphones and tablets," *Measurement*, vol. 81, pp. 197–209, 2016.
- [8] Dryanovski, I., R. G. Valenti, and J. Xiao, "An open-source navigation system for micro aerial vehicles," *Auton. Robots*, vol. 34, no. 3, pp. 177–188, 2013.
- [9] Chai, W., C. Chen, E. Edwan, J. Zhang, and O. Loffeld, "INS/Wi-Fi based indoor navigation using adaptive Kalman filtering and vehicle constraints," in *Positioning Navigation and Communication (WPNC), 2012 9th Workshop on*, 2012, pp. 36–41.
- [10] Simpson, A. K., "Autonomous robot control for underwater tunnel mapping," *Sr. thesis, Princet. Univ.*, 2012.
- [11] S. Biswas and R. Sharma, "Goal-aware Navigation of Quadrotor UAV for Infrastructure Inspection," in *AIAA Scitech 2019 Forum*, 2019, p. 1691.
- [12] Kumar, G. A., A. Patil, R. Patil, S. Park, and Y. Chai, "A LiDAR and IMU integrated indoor navigation system for UAVs and its application in real-time pipeline classification," *Sensors*, vol. 17, no. 6, p. 1268, 2017.
- [13] Balasubramanian, A. and A. Ganesan, "Vision-based heading and lateral deviation estimation for indoor navigation of a quadrotor," *IETE J. Res.*, vol. 63, no. 4, pp. 597–603, 2017.
- [14] Tiemann, J., A. Ramsey, and C. Wietfeld, "Enhanced UAV indoor navigation through SLAM-augmented UWB localization," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2018, pp. 1–6.
- [15] Yin, Y., Y. Chen, L. Wan, Y. Zhang, and Y. Yang, "Research on indoor multipoint data acquisition with a micro UAV," in *2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC/1&CPS Europe)*, 2017, pp. 1–4.
- [16] Zhang, P., R. Li, Y. Shi, and L. He, "Indoor Navigation for Quadrotor Using RGB-D Camera," in *Proceedings of 2018 Chinese Intelligent Systems Conference*, 2019, pp. 497–506.
- [17] Nahangi, M., A. Heins, B. McCabe, and A. Schoellig, "Automated Localization of UAVs in GPS-Denied Indoor Construction Environments Using Fiducial Markers," in *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, 2018, vol. 35, pp. 1–7.
- [18] Zhou, B., Z. Chen, and Q. Liu, "An Efficient Solution to the Perspective-n-Point Problem for Camera With Unknown Focal Length," *IEEE Access*, vol. 8, pp. 162838–162846, 2020.
- [19] Adli, S.E., M. Shoran, and S. M. S. Noorani, "GSP n P: simple and geometric solution for P n P problem," *Vis. Comput.*, vol. 36, no. 8, pp. 1549–1557, 2020.
- [20] Bai, L., Y. Yang, C. Guo, C. Feng, and X. Xu, "Camera assisted received signal strength ratio algorithm for indoor visible light positioning," *IEEE Commun. Lett.*, vol. 23, no. 11, pp. 2022–2025, 2019.
- [21] Grunert, J.A., "Das pothenotische problem in erweiterter

همانگونه که در جدول‌های ۶ الی ۸ ملاحظه شد، مقدار واریانس خطا در محور  $x$  با توجه به روش‌های مختلف استفاده شده ۲۲ الی ۲۵ درصد بیشتر از مقدار واریانس خطا در محور  $y$  است که می‌توان دلیل آن را دورتر بودن دوربین شماره ۲ نسبت به ربات پرنده دانست (شکل ۲۰). همچنین به دلیل بالاتر قرار داشتن چراغ ربات پرنده نسبت به نقطه مرکزی آن، در حدود ۳ سانتیمتر خطای آفست در تخمین موقعیت عمودی (محور  $z$ ) ربات پرنده وجود دارد. در مجموع می‌توان گفت خطای میانگین در حدود ۴ سانتیمتر برای فواصل داخل یک اتاق نسبتاً بزرگ و با وجود دوربینهای فرض شده با وضوح ۸۰۰ پیکسل قابل قبول می‌باشد.

### جمع بندی و نتیجه گیری

در مقاله حاضر فعالیت‌های مرتبط با موقعیت‌یابی یک ربات پرنده در فضای محدود به یک اتاق در محیط شبیه ساز مجازی ارائه گردید. به منظور انجام این کار ابتدا دینامیک پرواز ربات پرنده شبیه سازی شده و در محیط 3DsMAX پرواز آن در مقابل تعدادی (در اینجا ۲ عدد) دوربینهای چشم‌ماهی با میدان دید نزدیک به ۱۸۰ انجام گرفت. در قدم اول کالیبراسیون درونی و برونی دوربین‌ها انجام شد. در ادامه نیز نحوه تعقیب نقطه نورانی روی ربات پرنده در هر کدام از دوربین‌ها و ردگیری آن ارائه شده و سپس تلفیق داده‌های دوربین‌ها با یکدیگر و تخمین موقعیت ربات پرنده تشریح گردید. پس از تحلیل تصاویر شبیه‌سازی شده مشخص شد با خطایی در حدود ۴ سانتیمتر می‌توان مکان ربات پرنده را در یک اتاق تخمین زد. با توجه به اینکه در این فعالیت، کلیه عملیات بینایی ماشین تحت زبان ++C و با بهره‌گیری از کتابخانه‌های OpenCV انجام پذیرفته است تا به زبان ماشین نزدیکتر باشد و سرعت اجرای بیشتری را به ارمغان آورد؛ بنابراین در تکمیل این فعالیت با اتصال دوربین‌ها به یک پردازنده نیز می‌توان به سادگی روش پیشنهادی را به صورت سخت افزار در حلقه و تست میدانی در زمان واقعی مورد آزمون قرار داد.

### مراجع

- [1] Lee, C.-Y. "Cooperative Drone Positioning Measuring in Internet-of-Drones," in *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, 2020, pp. 1–3.
- [2] Aminzadeh, A., M. A. Amiri Atashgah, and A. Roudbari, "Software in the loop framework for the performance assessment of a navigation and control system of an unmanned aerial vehicle," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 33, no. 1, pp. 50–57, Jan. 2018.
- [3] Tyukin, A. L., A. L. Priorov, and I. M. Lebedev, "Research and development of an indoor navigation system based on the digital processing of video images," *Pattern Recognit. Image Anal.*, vol. 26, no. 1, pp. 221–230, 2016.

- [24] Available, [on Line]: "[https://docs.opencv.org/3.2.0/d7/d1b/group\\_imgproc\\_misc.html](https://docs.opencv.org/3.2.0/d7/d1b/group_imgproc_misc.html)".
- [25] Available, [on Line]: "[https://docs.opencv.org/3.2.0/d3/dc0/group\\_imgproc\\_shape.html](https://docs.opencv.org/3.2.0/d3/dc0/group_imgproc_shape.html)".
- [26] Suzuki, S. "Topological structural analysis of digitized binary images by border following," *Comput. vision, Graph. image Process.*, vol. 30, no. 1, pp. 32-46, 1985.
- [27] Available, [on Line]: "<https://math.stackexchange.com/questions/61719/finding-the-intersection-point-of-many-lines-in-3d-point-closest-to-all-lines>".
- gestalt nebst bber seine anwendungen in der geodasie," *Grunerts Arch. fur Math. und Phys.*, pp. 238-248, 1841.
- [22] Kneip, L., D. Scaramuzza, and R. Siegwart, "A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation," 2011.
- [23] Gao, X.-S., X.-R. Hou, J. Tang, and H.-F. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 930-943, 2003.