



دانشگاه صنعتی امیرکبیر  
دانشکده مهندسی کامپیوتر  
گروه هوش مصنوعی و رباتیکز



## ربات‌های متحرک خودگردان هواپایه

### پروژه بینایی سه‌بعدی

#### بخش اول) کالیبراسیون دوربین

در تمام پروژه‌های بینایی کامپیوتر سه‌بعدی، گام اول کالیبراسیون سنسورها است. تئوری مربوط به کالیبراسیون پیش از این در کلاس درس تشریح شد.

۱) با کمک توضیحات و کدهای مربوطه در این [لینک](#)، فرآیند کالیبراسیون را برای دوربین موبایل خود انجام دهید و نتایج را ارائه کنید. الگوهای کالیبراسیون را می‌توانید از این [سایت](#) دریافت کنید.

#### بخش دوم) بینایی استریو

بینایی استریو کم هزینه‌ترین راه دسترسی به بازسازی سه بعدی متریک است. در این تمرین قصد داریم از این روش برای تخمین عمق تصاویر از دیتاست [Kagaru Airborne](#) استفاده کنیم. فایل [100831\\_155323\\_MultiCamera0\\_subset\\_db](#) که مربوط به ۱۰۰ فریم از این دیتاست است را دانلود کنید. پارامترهای مربوط به دو دوربین دیتاست در وبسایت مذکور ارائه شده است.

۲) با استفاده از کتابخانه OpenCV، تصاویر عمق مربوط به این ۱۰۰ فریم را استخراج و به صورت یک ویدیو در پیوست گزارش ارائه کنید.

۳) تصاویر عمق را با کمک پارامترهای داخلی دوربین به ابرنقاط تبدیل کنید.

۴) ابرنقاط به دست آمده را به کمک الگوریتم ICP موجود در کتابخانه Open3D تطبیق دهید و مسیر حرکت دوربین را نمایش دهید.

**راهنمایی:** استفاده از پیاده‌سازی‌های جایگزین از سایر کتابخانه‌ها و سایر زبان‌های برنامه‌نویسی مجاز است.

#### بخش سوم) ساختار از حرکت (امتیازی)

هدف از مسئله [ساختار از حرکت](#)<sup>۱</sup> استخراج نقشه سه بعدی محیط با استفاده از یک ویدیو است. یکی از معروف ترین کتابخانه های این مسئله، کتابخانه [COLMAP](#) است که چارچوب کامل SfM را ارائه می کند. در این تمرین قصد داریم برای استخراج نقشه از یکی از دنباله های دیتاست [UZH FPV](#) از این کتابخانه استفاده کنیم.

از بخش مربوط به (Indoor 45 degree downward facing) در این [لینک](#) دنباله #۲ را به صورت فایل zip دانلود کنید. برای ساخت نقشه تنها به تصاویر دوربین Snapdragon و برای ارزیابی الگوریتم به دنباله موقعیت Leica نیاز داریم (توجه کنید که فرض می شود به دنباله موقعیت دسترسی نداریم و تنها برای ارزیابی خروجی های کتابخانه COLMAP از آن استفاده می کنیم).

۵) کتابخانه COLMAP را بر روی تصاویر فوق اجرا کنید. خروجی را به صورت نقشه ابر نقاط و مسیر حرکت ربات در گزارش خود نشان دهید.

۶) مسیر حرکت به دست آمده را با دنباله موقعیت مقایسه و نمایش دهید. توجه کنید مقیاس مسیر حرکت به دست آمده نامشخص است؛ بنابراین می توانید برای نمایش، مقیاس هر دو داده را یکسان کنید.

راهنمایی: استفاده از واسط کاربری گرافیکی COLMAP مجاز است.

### بخش سوم) هندسه تک نمایی<sup>۲</sup> (امتیازی)

اگرچه دوربین قابل به برداشت اطلاعات هندسی از صحنه نیست، با داشتن اطلاعات پیشین، برخی از جزئیات هندسی با روابط جبری و مدل دوربین قابل استنباط است. فرض کنید در تصویر زیر، ارتفاع یکی از اجسام موجود در صحنه (مانند چراغ راهنمایی) معلوم است.

۷) با فرض داشتن پارامترهای داخلی دوربین، نشان دهید ارتفاع دوربین از سطح زمین قابل محاسبه است.



<sup>1</sup> Sturcture-from-motion

<sup>2</sup> Single-view Geometry

**راهنمایی:** استفاده از حداقل یک Vanishing Point برای محاسبه زاویه دوربین نقطه آغاز مناسبی است (پیوست).

### پیوست: Vanishing Points

خطوط موازی در یک صفحه هیچگاه همدیگر را قطع نمی کنند، اما در هندسه جبری، خطوط موازی در جایی خارج از صفحه همدیگر را قطع می کنند! محل های برخورد خطوط موازی اصطلاحاً Vanishing Points یا نقاط گریز نامیده می شوند. مشابه شکل زیر، محاسبه محل یک نقطه گریز به سادگی رسم حداقل دو خط موازی (در اینجا خطوط موازی زمین فوتبال) انجام می شود. هر خط دیگری که به طور دلخواه از این نقطه رسم کنید، با سایر خطوط موازی است. استفاده از این ویژگی برای تشخیص آفساید یکی از کاربردهای واقعی Vanishing Points است.

کاربردهای VP در مسائل بینایی کامپیوتر سه بعدی بسیار زیاد است. از مهم ترین کاربردها میتوان به تخمین پارامترهای داخلی دوربین، تخمین اندازه با کمک روابط هندسه تک نمایی و ... اشاره کرد. به طور خاص برای حل تمرین آخر می توان ثابت کرد که با داشتن VP مربوط به صفحه کف (صفحه زمین)، زاویه Pitch دوربین از رابطه زیر قابل محاسبه است:

$$\theta_{pitch} = \tan^{-1} \frac{v_y - c_y}{f}$$

در رابطه بالا  $v_y$  و  $c_y$  به ترتیب مختصات پیکسلی VP و مرکز تصویر در راستای y (عمودی) است. توجه کنید که مختصات پیکسلی VP می تواند خارج از تصویر باشد.

