

به نام خدا



دانشگاه صنعتی امیرکبیر
دانشکده مهندسی کامپیوتر
استاد درس: دکتر صفابخش

پاییز ۱۴۰۱

درس بینائی کامپیوتر

تمرین هفتم

هدف: آشنایی با روش‌های تخمین عمق، بازسازی سه‌بعدی و سنسور لیدار

کد: در پیاده سازی می‌توانید از زبان‌های پایتون، متلب یا سی پلاس پلاس استفاده کنید. همچنین در تمامی موارد می‌توانید از کتابخانه اپن سی وی و Open3D استفاده کنید مگر اینکه صراحتاً خلاف آن در صورت سوال ذکر شده باشد.

گزارش: توجه کنید ملاک اصلی برای ارزیابی گزارش تمرین می‌باشد. برای این منظور گزارش را در قالب PDF تهیه کنید و برای هر سوال، تصاویر ورودی، خروجی و توضیحات مربوط به آن را ذکر کنید. همچنین اگر فرض اضافه‌ای در نظر می‌گیرید حتماً در گزارش به آن اشاره کنید.

تذکر: مطابق قوانین دانشگاه هرگونه کپی برداری و اشتراک کار دانشجویان غیرمجاز بوده و شدیداً برخورد خواهد شد. استفاده از کدها و توضیحات اینترنت به منظور یادگیری بلامانع است، اما کپی کردن غیرمجاز است.

راهنمایی: در صورت نیاز سوالات خود را می‌توانید در گروه مربوط به درس در تلگرام یا با ایمیل زیر مطرح کنید.

E-mail: cv.ceit.aut@gmail.com

ارسال: فایل‌های کد و گزارش را در قالب یک فایل فشرده با فرمت studentID_HW07.zip تا تاریخ ده بهمن‌ماه ارسال نمایید.

تاخیر مجاز: در طول ترم، مجموعاً مجاز به حداکثر ده روز تاخیر برای ارسال تمرینات هستید (بدون کسر نمره). این تاخیر را می‌توانید برحسب نیاز بین تمرینات مختلف تقسیم کنید، اما مجموع تاخیرات تمام تمرینات شما نباید بیشتر از ده روز شود. پس از استفاده از این تاخیر مجاز، هر روز تاخیر باعث کسر ده درصد نمره خواهد شد.

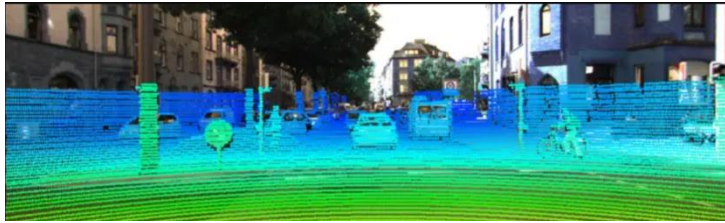
برخلاف بینایی دو بعدی کامپیوتر که در اکثر موارد تصاویر دوربین‌های معمولی به عنوان ورودی الگوریتم‌ها کافی است. در بینایی سه بعدی کامپیوتر معمولاً بازیابی بعد سوم تصویر، که در فرآیند تصویربرداری از بین می‌رود، یک گام مهم برای شروع حل مسئله است. در این تمرین با دو روش رایج بازیابی عمق، بینایی استریو و سنسور لیدار آشنا می‌شویم. به این منظور از داده‌های دیتاست KITTI (پیوست یک) استفاده خواهیم کرد.

برای شروع با داده‌های سنسور لیدار (ابر نقاط - پیوست یک) آشنا می‌شویم.

الف) داده‌های سنسور لیدار را برای یک فریم دلخواه بارگذاری کنید و به وسیله کتابخانه Open3D نمایش دهید. به نظر شما مزایا و معایب ابر نقاط چیست؟

در اکثر کاربردهای صنعتی (در مواردی که استفاده از لیدار صرفه اقتصادی دارد)، سنسورهای لیدار^۱ و دوربین به عنوان مکمل هم در نظر گرفته می‌شود. به فرآیند ترکیب داده‌های دو سنسور مختلف در کاربردهای بینایی ماشین، اصطلاحاً fusion گفته می‌شود. ساده‌ترین راه برای ترکیب این دو سنسور، نگاشت نقاط از ابر نقاط به تصویر است.

^۱ Lidar



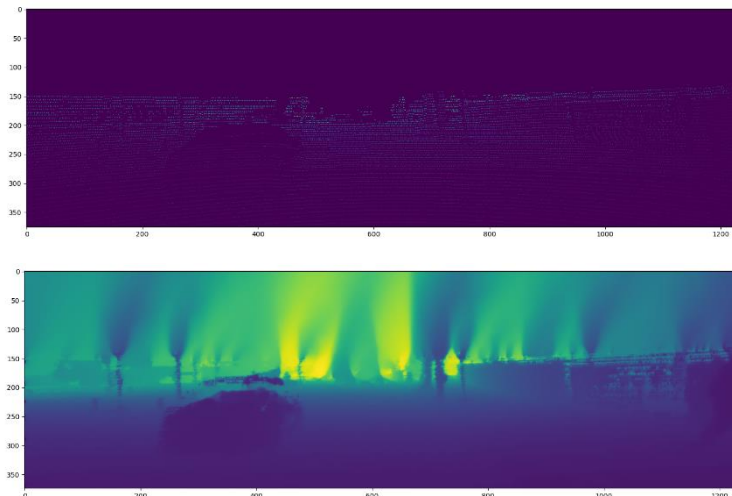
شکل ۱- نگاشت ابر نقاط به تصویر

ب) با استفاده از [این لینک](#) به عنوان راهنما، فرآیند نگاشت ابر نقاط به تصویر را به طور خلاصه توضیح دهید و پیاده سازی کنید. خروجی را برای یک فریم از دیتاست پیوست ارائه کنید. در خروجی ارائه شده باید مانند شکل ۱، رنگ نقاط نمایانگر عمق آن‌ها باشد.

علازمه دقت بالای سنسور لایدار، ابر نقاط خروجی آن معمولاً بسیار غیرمترکم‌تر از دوربین است، بنابراین در محل بسیاری از پیکسل‌های تصویر، عمق متناظر با آن در دسترس نخواهد بود. یک راه ساده برای حل این مشکل، استفاده از Interpolation برای این نقاط است.

ج) یک نقشه عمق^۲ متناظر با تصویر ورودی ایجاد و نمایش دهید (شکل ۲). در نقاطی که عمق برای آن‌ها در دسترس نیست، از یک روش Interpolation برای تعیین عمق استفاده کنید. به نظر شما چه روشی از Interpolation برای این کار مناسب‌تر است؟

همانطور که مشاهده می‌کنید، روش ذکر شده برای ایجاد نقشه عمق عملکرد مناسبی ندارد. درحقیقت ساخت نقشه عمق با استفاده از ابرنقاط یک حوزه فعال از بینایی ماشین است که تحت عنوان Depth Completion شناخته می‌شود.



شکل ۲- تبدیل ابر نقاط به نقشه عمق

^۲ Depth map



شکل ۳- بازسازی سه بعدی صحنه

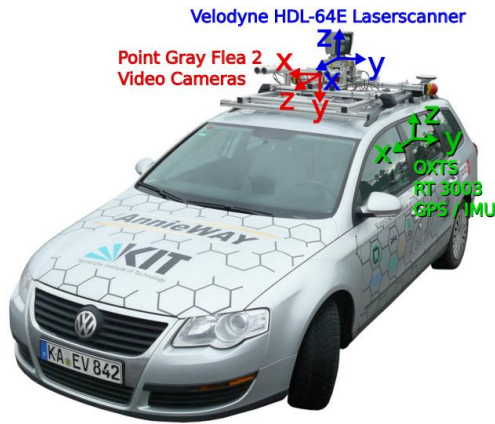
یک راه دیگر برای بازسازی عمق که نقشه‌های عمق مترکم تری را نتیجه می‌دهد، استفاده از بینایی استریو است که در درس با آن آشنا شدید. برای انجام سوال بعدی از تصاویر سطح خاکستری موجود در دیتاست استفاده کنید.

د) با استفاده از توابع مناسب از کتابخانه OpenCV و با روش بینایی استریو یک نقشه عمق متناظر با تصویر ورودی ساخته و نمایش دهید. توجه کنید که نقشه Disparity باید با روشی که در درس خواندید به نقشه عمق تبدیل شود. نقشه عمق ساخته شده را با نقشه عمق ساخته شده از ابر نقاط مقایسه کنید.

پس از تعیین عمق متناظر با هر پیکسل از تصویر، می‌توان فضای سه بعدی تصویر را بازسازی کرد. برای این کار هر نقطه از تصویر را مجدداً به یک نقطه سه بعدی نگاشت می‌کنیم و این بار یک ابر نقاط بسیار مترکم تر خواهیم داشت. این تبدیل معکوس، با روابطی که در بخش ب از آن‌ها استفاده کردید ممکن است؛ اما برای سادگی می‌توانید از [توابع آماده](#) کتابخانه Open3D استفاده کنید.

ه) از تصویر ورودی و نقشه عمق متناظر با آن برای ساخت نمایش سه بعدی صحنه استفاده کنید.

پیوست ۱



شکل 4- پلتفرم جمع‌آوری دیتاست KITTI

الف) دیتاست KITTI^۳

این دیتاست سال‌های طولانی استاندارد مهمی برای ارزیابی الگوریتم‌های بینایی کامپیوتر دو بعدی و سه بعدی بوده است. با تمرکز بر مسئله اتومبیل‌های خودران و سیستم‌ها کمک راننده، این دیتاست شامل سنسورهای مختلفی از جمله دوربین، لایدار، GPS و ... است (شکل 4).

داده‌های خام این دیتاست که حدود ۱۳۵ گیگابایت اطلاعات سنسورهای مختلف را شامل می‌شود، به مرور زمان برای چندین مسئله مهم بینایی ماشین توسعه یافته است. به همراه این تمرین، خروجی چهار فریم متوالی از دیتاست برای همه سنسورها با قالب اصلی دیتاست ارائه شده است. ساختار پوشه ارائه شده به صورت زیر است.

▼ kitti_sample	
▼ 2011_09_29	تاریخ برداشت داده‌ها
▼ 2011_09_29_drive_0026_sy...	شماره، توالی برداشت شده
> image_00	تصاویر اصلاح شده (rectified) دوربین سطح خاکستری چپ
> image_01	تصاویر اصلاح شده (rectified) دوربین سطح خاکستری راست
> image_02	تصاویر اصلاح شده (rectified) دوربین رنگی چپ
> image_03	تصاویر اصلاح شده (rectified) دوربین رنگی راست
> oxts	
> velodyne_points	اسکن‌های لایدار
≡ gps.txt	
≡ calib_cam_to_cam.txt	پارامترهای داخلی و خارجی دوربین‌ها و ماتریس‌های نگاشت بین آن‌ها
≡ calib_imu_to_velo.txt	
≡ calib_velo_to_cam.txt	ماتریس چرخش و انتقال برای لایدار به دوربین
❏ readme.txt	چریات بیشتر

همانطور که مشاهده می‌کنید، پلتفرم جمع‌آوری داده‌ها شامل دو جفت دوربین استریو (مجموعاً چهار دوربین) و یک سنسور لایدار است. برای انجام تبدیلات سه بعدی، تمامی اطلاعات از جمله ماتریس‌های پارامترهای داخلی و خارجی و ماتریس اصلاح^۴ ارائه شده است. توجه شود که دوربین شماره ۰۰ به عنوان مختصات مرجع تمامی سنسورها در نظر گرفته شده است. بنابراین ماتریس پارامترهای خارجی دوربین‌ها، نسب به این دوربین محاسبه شده‌اند. همچنین ماتریس چرخش و انتقال از لایدار نیز نسبت به همین دوربین محاسبه شده است. در مجموع، ماتریس‌هایی که در این تمرین به آن‌ها نیاز دارید به شرح زیر است:

^۳ Geiger, Andreas, et al. "Vision meets robotics: The kitti dataset." The International Journal of Robotics Research 32.11 (2013): 1231-1237.

^۴ rectification

calib_cam_to_cam.txt	
K _{xx}	ماتریس کالیبراسیون سه در سه دوربین xx پیش از اصلاح (rectification)
R _{rect_xx}	ماتریس اصلاح سه در سه دوربین xx
P _{rect_xx}	ماتریس نگاشت سه در چهار از دوربین xx به دوربین °° پس از اصلاح
calib_velo_to_cam.txt	
R, T	به ترتیب ماتریس سه در سه چرخش و بردار انتقال از لایدار به دوربین °°

پیشنهاد می‌شود برای آشنایی بیشتر با این دیتاست مقاله مربوطه را مطالعه کنید.

ب) سنسور لایدار و ابر نقاط



شکل 5- یک نمونه سنسور لایدار

تکنولوژی بسیار پیشرفته سنسورهای لایدار موجب شده که حتی ابتدایی ترین انواع آن، قیمت بسیار بیشتری از دوربین‌های رنگی رایج داشته باشند؛ بنابراین استفاده از این سنسورها همچنان در بسیاری از پروژه‌های صنعتی، صرفه اقتصادی ندارد. در این میان اهمیت حفظ ایمنی در خودروهای خودران، لایدار را به یک سنسور رایج در این خودروها تبدیل کرده است. این سنسور معمولاً از یک آرایه عمودی از لیزرها تشکیل شده که با نرخ بسیار بالایی به دور یک محور چرخیده و داده‌ها را با میدان دید 360° درجه برداشت می‌کند. یک برداشت از سنسور لایدار، معادل یک تصویر از دوربین، شامل تمامی نقاط ثبت

شده در یک چرخش کامل 360° درجه است. این برداشت به صورت یک ماتریس $N \times 4$ ارائه می‌شود که N تعداد نقاط برداشت شده (معمولاً چند صد هزار) و 4، بردار مربوط به یک نقطه شامل مختصات X، Y و Z و میزان بازتابش دریافتی است. در کاربردهای رایج بازتابش دریافتی استفاده نمی‌شود و بنابراین ماتریس $N \times 3$ به عنوان خروجی این سنسور در نظر گرفته می‌شود که به آن ابر نقاط⁵ نیز گفته می‌شود.

⁵ Point cloud