

تمرین شماره ۰۸ بینایی ماشین

فاطمه توکلی - ۴۰۰۱۳۱۰۱۶

Cv2.goodFeaturesToTrack (image, maxCorners, qualityLevel, minDistance, corners, mask, blockSize, useHarrisDetector, k)

در تابع فوق پارامترها به شکل زیر هستند:

- یک تصویر خاکستری ۸ بیتی یا ۳۲ بیتی به عنوان ورودی در نظر گرفته میشود.
- سقف تعداد گوشه‌ها. در اینجا اگر تعداد از تعداد تشخیص داده شده برای گوشه بیشتر باشد جواب‌های قوی برای کاندید گوشه بودن نیز در نظر گرفته میشود. اگر این مقدار از کوچکتر مساوی صفر باشد یعنی برای سقف تعداد گوشه بودن محدودیتی در نظر گرفته نشده‌است.
- سطح کیفی: در اینجا اگر به طور مثال میزان بهترین کیفیت ۱۳۰۰ باشد و سطح کیفی ۰.۰۲ باشد، تمام میزان کیفیت‌های بدست آمده که از ۲۶ کمتر باشند در نظر گرفته نشده و رد میشوند. یا به تعریفی دیگر حداقل کیفیت قابل قبول گوشه‌ها را بیان می‌کند. مقدار زیاد آن سبب کاهش نقاط گوشه تشخیصی و مقدار کم آن نیز سبب افزایش میشود که هر کدام به نوبه خود میتواند مضر باشد.
- کمترین فاصله: در اینجا کمترین فاصله اقلیدسی ممکن بین گوشه‌های بازگشتی و قابل قبول فرض میشود. مقدار زیاد آن باعث میشود گوشه‌های مهمی از دست برود، این مورد به درد اشکال بزرگ می‌خورد. مقدار بسیار کم آن نیز سبب میشود که در صورت بالا بودن تعداد گوشه‌ها نقاط غیر مهم و در کنار هم به عنوان گوشه اعلام شوند، البته به درد مواردی که جزئیات ریز و در کنار هم دارد میتواند موثر باشد.
- ماسک: ماسکی هم اندازه با تصویر در اینجا اگر قرار داشته باشد به این معنا است که ناحیه مورد توجه ما یا به عبارتی مکانی که از آنها قصد تشخیص گوشه‌ها را داریم مشخص میکنیم.
- اندازه بلوک: میانگین سائز بلوکی که برای محاسبه ماتریس کواریانس مشتق بر روی هر همسایگی پیکسل استفاده میشود.
- مشخص کردن اینکه از گوشه یاب هریس استفاده شود یا خیر که به طور پیش فرض false است.
- K اندازه پارامتر آزاد گوشه یاب هریس را مشخص میکند.

سؤال ۲:

ابتدا ۳ فریم اخیر وبکم را ذخیره کرده و سپس یک بار کد مربوط به پیدا کردن مسیر را برای جفت فریم ۱ و ۲ و یک بار هم برای جفت فریم ۲ و ۳ اجرا میکنیم و مسیر حرکتی جسم متحرک را پیدا میکنیم.

برای پیدا کردن مسیر به این صورت عمل شده است که ابتدا تمام عکس های ورودی به سطح خاکستری تبدیل شده است. سپس با استفاده از تابع `cv2.goodFeaturesToTrack`، تعداد گوشه های دلخواه از تصویر ورودی را استخراج کرده و سپس با استفاده از دو تصویر ورودی برای دنبال کردن جسم متحرک در این دو فریم متوالی و گوشه های استخراج شده به عنوان ورودی تابع `cv2.calcOpticalFlowPyrLK`، مسیر حرکت اجسام متحرک مشخص میشود. مسیر فریم های متوالی ۱ و ۲ با نقاط و خطوط زرد رنگ و مسیر فریم های متوالی ۲ و ۳ با نقاط و خطوط قرمز رنگ در فیلم ضبط شده مشخص شده است.

خروجی با نام `q2.avi` ضمیمه شده است.

سؤال ۳:

در روش متراکم نسبت به حالت پراکنده یا روش قبلی به جای در نظر گرفتن تنها نقاط گوشه یا مورد علاقه برای دنبال کردن از تمام نقاط برای تشخیص حرکت استفاده میشود. در اینجا در نظر گرفتن تمام نقاط با اینکه زمان بیشتری را به خود اختصاص میدهد اما دقت بهتری در دنبال کردن شی متحرک در تصویر نیز دارد. از همین رو برای هر نقطه شار یا میزان تغییرات بین دو فریم را در تمام دو بعد به عنوان خروجی الگوریتم به ما میدهد. در روش قبلی همینطور مرحله تشخیص نقاط کلیدی داریم که امکان خطا در دنبال کردن آنها در هر قاب نسبت به روشی که تمام پیکسل ها را در نظر میگیرد زیادتر است.

به صورت تجربی به دست آمده که دقت `Lucas Kanade` ممکن است برای کاربرد های `real-time` کافی باشد. شار نوری متراکم دقت بالاتری به قیمت کند بودن و بالا بودن هزینه محاسباتی دارد. و همچنین شار نوری نسبت به تغییرات روشنایی مقاوم نیست.

برای این سوال هم مانند سوال ۲، سه فریم اخیر را در آرایه ای ذخیره کرده و یک بار مسیر حرکت برای فریم ۱ و ۲ و یک بار هم برای فریم ۲ و ۳ به دست آمده است. برای پیدا کردن مسیر به این صورت عمل شده است که ابتدا تمام عکس های ورودی به سطح خاکستری تبدیل شده است. سپس با استفاده از دو تصویر ورودی برای دنبال کردن جسم متحرک در این دو فریم متوالی به عنوان ورودی تابع `cv2.calcOpticalFlowFarneBack`، مسیر حرکت اجسام متحرک مشخص میشود. برای نمایش دادن مسیر طی شده توسط جسم ابتدا شار نوری را با تابع `cv2.calcOpticalFlowFarneBack` اندازه و جهت این شار را هم در فریم های متوالی به دست آورده و بعد از تبدیل مقادیر به دست آمده به ابعاد فضای رنگی `HSV`، تصویر به دست آمده را به فضای رنگی `RGB` منتقل کرده و در نهایت این تصویر را نمایش می دهیم. تصویر حاوی پیکسل های رنگی متحرک می باشد که این پیکسل های رنگی، حرکت جسم متحرک را به نمایش می گذارند و پیکسل های مشکی رنگ، به این معنا هستند که در آن قسمت های فریم ها جسم متحرکی وجود نداشته و هیچ تغییری بین دو فریم مشاهده نشده است.

خروجی این قسمت با نام q3_a.avi پیوست شده است.

روش دیگری نیز پیاده سازی شده است که با استفاده از گرید هر چند پیکسل را با یک نقطه نمایش می دهد و سپس اندازه و جهت flow مشخص شده را با استفاده از این نقاط به نمایش می گذارد. جسم متحرک به هر سمتی که حرکت کند، نقاط حوالی آن جسم متحرک، جهت و اندازه‌ی حرکت را به نمایش می گذارند و اگر در قسمتی از تصویر تغییراتی وجود نداشت، نقاط آن محدوده هم ساکن نمایش داده میشوند. برای این نحوه‌ی نمایش، مسیر فریم‌های متوالی ۱ و ۲ با نقاط و خطوط زرد رنگ و مسیر فریم‌های متوالی ۲ و ۳ با نقاط و خطوط قرمز رنگ در فیلم ضبط شده مشخص شده است.

خروجی این قسمت با نام q3_b.avi پیوست شده است.

سؤال ۴:

استفاده از روش‌های شار نوری به نسبت توصیفگرها هم دقت بهتری دارند هم از نظر مقاوم بودن، مقاومت بهتری را از خود در برابر چرخش و حرکت با سرعت بالاتر را نشان می دهند. از طرفی سرعت نیز در هر دو روش شار نوری به نسبت روش‌های توصیفگر بیشتر است که همین برتری‌ها استفاده از روش‌های شارنوری را در اولویت اول نسبت به روش‌های توصیفگر قرار می دهد. از طرفی در روش‌های توصیفگر برای زوایایی که هیستوگرام در نظر گرفته است به صورت بازهای عمل میکنند و به صورت بسیار دقیق امکان دنبال کردن حرکات را ندارند. برای توصیفگرها همچنین به تنظیم بسیار دقیق تر پارامترها برای خروجی بهتر احتیاج هست که این تنظیم در شار نوری کمتر تاثیر حیاتی از خود نشان می دهد. با این حال اگر تغییرات نوری شدید داشته باشیم استفاده از روش شار نوری توصیه نمیشود. همینطور اگر تعداد بسیاری جسم در فضا در حال حرکت داشته باشد که قصد در نظر گرفتن آنها را نداریم استفاده از روش توصیفگرها بهتر است.