

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

گروه مستقل مهندسی رباتیک

تمرین چهارم درس بینایی ماشین

استاد درس:

دکتر صفابخش

تدریس یار:

مهندس مجد

نام دانشجو:

نوید خزاعی

۹۲۱۳۵۰۰۸

خرداد ۱۳۹۳

فهرست مطالب

۲	فهرست مطالب
۳	۱ بخش اول
۳	۱.۱ SIFT
۷	۲.۱ SURF
۸	۲ HOG
۸	۱.۲ تشخیص انسان
۱۰	۲.۲ بهبود HOG
۱۲	مراجع

۱ بخش اول

۱.۱ SIFT

پرسش: از کاربرد های الگوریتم SIFT پیدا کردن تطابق بین تصاویر است. برای این کار کافی است الگوریتم SIFT را بر روی هر دو تصویر اعمال کرده و سپس با استفاده از یک الگوریتم انطباق ویژگی میزان انطباق دو تصویر و یا محل وقوع یک تصویر در دیگری را تعیین کرد. با استفاده از الگوریتم SIFT و یکی از روش های انطباق پیاده سازی شده در open-cv مثل Brut-Force Matcher، نقاط منطبق در دوجفت تصویر ۱۰ (۱-۱۰ و ۲-۱۰) و ۱۱ (۱-۱۱ و ۲-۱۱) را بیابید. تصاویر را پیش از استفاده به تصاویر خاکستری تبدیل کنید.

پاسخ: برای استفاده از الگوریتم SIFT در OpenCV دو روش وجود دارد. یکی استفاده از خود کلاس SIFT است و دیگری استفاده از SiftFeatureDetector و SiftDescriptorExtractor. روال کار به این شکل است که باید ابتدا الگوریتم را بر روی هر تصویر اجرا نموده و نقاط کلیدی^۱ را استخراج کنیم و سپس هر کدام از نقاط را توسط یک بردار توصیف گر^۲ نمایش دهیم. در هر دو فاز از الگوریتم SIFT استفاده می کنیم. سپس این بردارها را به یک سیستم تطبیق ویژگی^۳ می دهیم تا نقاط موجود در هر بردار را با دیگری تطبیق دهد. سپس بهترین تطبیق ها، که بر اساس فاصله از نقطه ی متناظر در بردار دیگر مشخص می شوند، انتخاب می شوند. در صورت وجود تطبیق های زیاد، می توان نتیجه گرفت که شی مورد نظر، در تصویر دوم نیز وجود دارد.

روش استفاده از SIFT در این تمرین در زیر آورده شده است:

○ با استفاده از کلاس SIFT یک نمونه ایجاد می کنیم:

```
SIFT mysift(0, 3, 0.04, 10, 1);
```

که پارامترهای آن به ترتیب در زیر آورده شده است. مقادیر پیش فرض هر کدام نیز مشخص شده است:

- `int nfeatures = 0`: تعداد ویژگی هایی که در انتها باقی می ماند. انتخاب این تعداد بر اساس رتبه بندی خود الگوریتم در حین اجرا مشخص می شود.

- `int nOctaveLayers = 3`: تعداد لایه ها در هر اکتاو را مشخص می کند که در ایجاد

هرم تصویر به کار برده می شود. در الگوریتم اصلی [۱] این تعداد به صورت پویا بر اساس رزولوشن

^۱ Key points

^۲ Descriptor vector

^۳ Feature matching

تصویر محاسبه می‌شود.

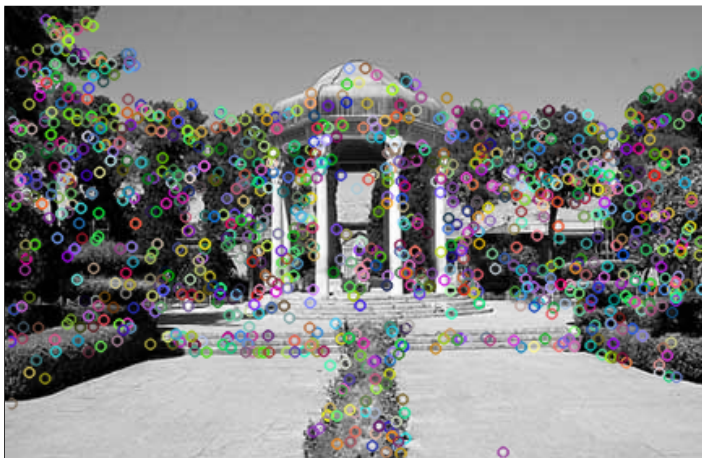
- $\text{double contrastThreshold} = 0.04$: برای حذف ویژگی‌های ضعیف در نقاط با کنتراست کم استفاده می‌شود.

- $\text{double edgeThreshold} = 10$: حد آستانه برای حذف ویژگی‌های گوشه‌مانند است. به این ترتیب که هر چقدر بیشتر باشد ویژگی‌های کمتری حذف می‌شوند.

- $\text{double sigma} = 1.6$: واریانس فیلتر گوسی اعمال شده بر تصویر در اکتاو صفر را مشخص می‌کند. اگر تصویر با دوربین ضعیف گرفته شده است بهتر است این مقدار را کم کنیم. سپس برای استخراج نقاط کلیدی می‌توانیم از یک الگوریتم تطبیق‌دهنده‌ی ویژگی^۴ مانند FLANN استفاده کنیم. پس از آن، تعدادی از بهترین تطبیق‌ها را انتخاب می‌کنیم و نقاط نظیر را به هم متصل می‌کنیم.

برای هر تصویر، نتیجه‌ی اجرا به همراه پارامترهای تنظیم شده آورده شده‌است:

اجرا با پارامترهای پیش‌فرض:

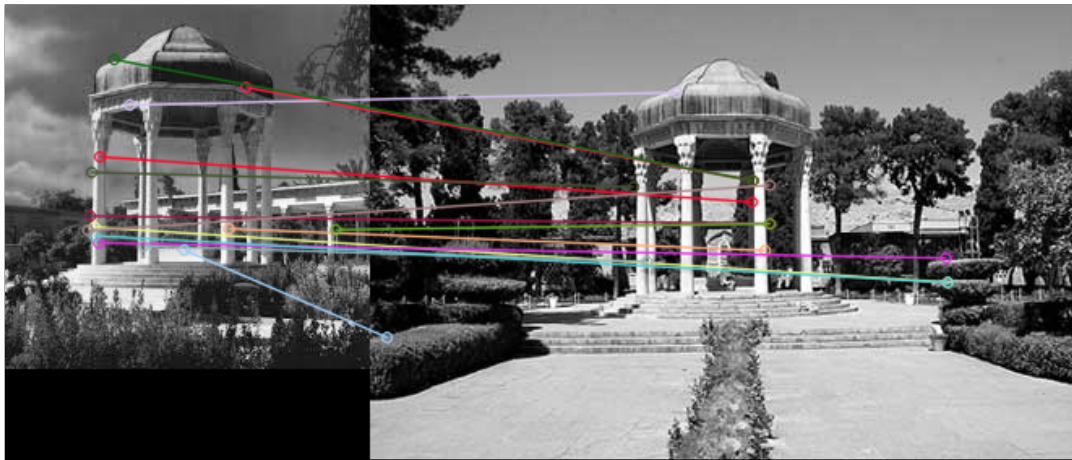


تصویر مقایسه



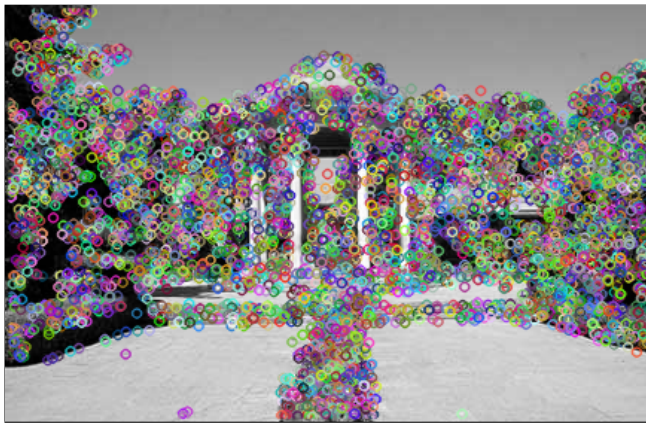
تصویر اصلی

^۴ Feature matching



نقاط منطبق با فاصله‌ی کمتر از ۱۰۰

اجرا با مقدار واریانس یک:



تصویر مقایسه

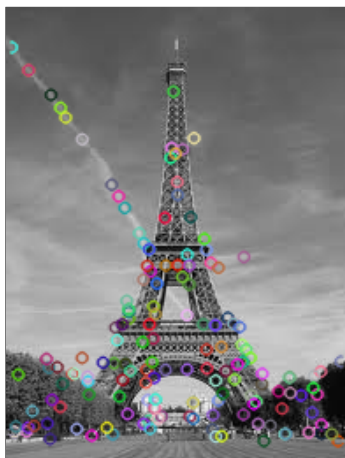


تصویر اصلی

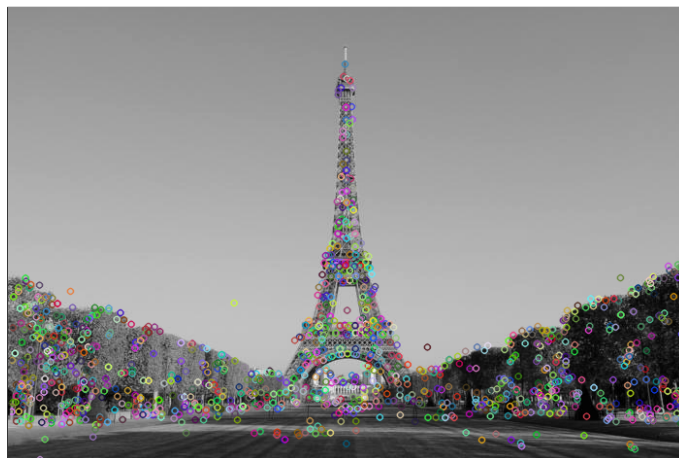


نقاط منطبق با فاصله‌ی کمتر از ۸۰

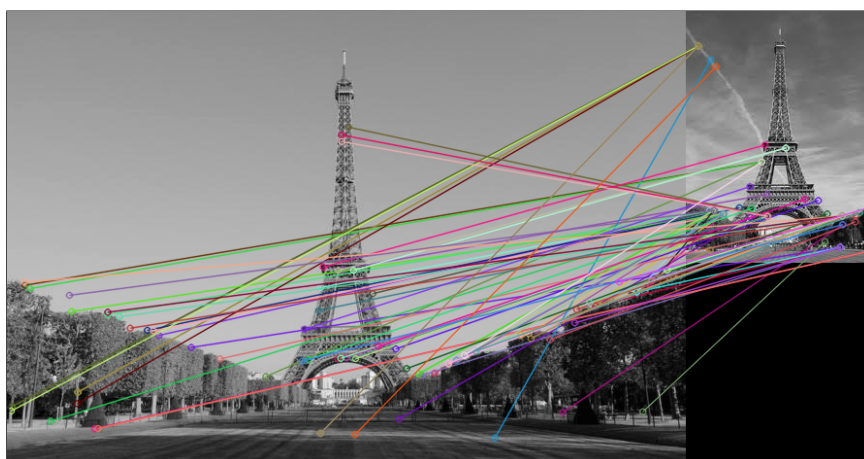
نتایج برای تصویر دوم و با پارامترهای پیش فرض به این شکل است:



تصویر مقایسه



تصویر اصلی

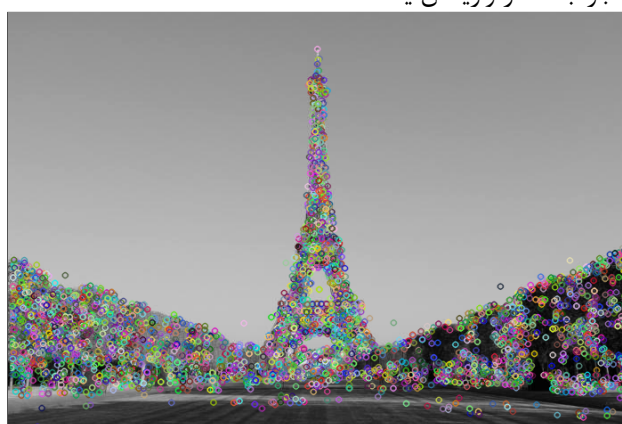


نقاط منطبق با فاصله‌ی کمتر از ۱۰۰

اجرا با مقدار واریانس یک:



تصویر مقایسه



تصویر اصلی



نقاط منطبق با فاصله‌ی کمتر از ۱۰۰

۲.۱ SURF

پرسش: الگوریتم SURF یکی از چندین الگوریتمی است که بر پایه الگوریتم SIFT نوشته شده است. در دو پاراگراف و به طور خلاصه تفاوت دو روش و بهبودهای بدست آمده را شرح دهید.

پاسخ:

در SIFT از تفاضل گاسی^۵ استفاده می‌شود در صورتی که در SURF از ماتریس هسی^۶ استفاده می‌شود که دقت و کارایی بیشتری دارد. در SIFT هرمی از رزولوشن‌های تصویر می‌سازد و سپس فیلترهای گاسی با افزایش سیگما (واریانس) را به آن‌ها اعمال می‌کند و سپس تفاضل آن‌ها را محاسبه می‌نماید. این عمل معادل لاپلاس گرفتن از تابع گاسی و اعمال آن به تصویر است. در نهایت با محاسبه‌ی جهت گرادیان در اطراف نقطه‌ی کلیدی پیدا شده در مرحله‌ی قبل، به آرایه‌ای از هیستوگرام‌ها (برای نمونه 4×4) برای نواحی مختلف اطراف تصویر می‌رسیم که هر کدام، برای نمونه، ۸ جهت نماینده‌ی بازه‌هایی از تمامی جهات را شامل می‌شوند و به این شکل، یک توصیف ۱۲۸ بعدی از نقطه‌ی کلیدی خواهیم داشت که نسبت به تبدیلات هندسی و تغییرات شدت روشنایی و نویز، تا حد بسیار زیادی مقاوم است.

اما در SURF چنین نیست. در اصل در SURF از موجک‌های دو بعدی هار^۷ و از تصویر انتگرال^۸

^۵ DoG - Difference of Gaussian

^۶ Hessian Matrix

^۷ 2D Haar wavelet

^۸ Integral Image

استفاده می‌شود. در SURF با استفاده از یک تقریب خطی از لکه‌یاب هسی^۹ که توسط تصویر انتگرال محاسبه می‌شود، سرعت بسیار بیشتر می‌شود و در اطراف این نقطه‌ی مهم، از جمع موجک‌های هار استفاده می‌کند تا نقطه را توصیف کند و این نیز به وسیله‌ی تصویر انتگرال قابل محاسبه است. از آن‌جا که این روش به نوعی از Look up Table استفاده می‌کند بار پردازشی آن کاهش یافته است. این روش تعداد نقاط کمتری پیدا می‌کند ولی سرعت آن به مراتب از SIFT بیشتر است.

۲ HOG

۱.۲ تشخیص انسان

پرسش: یکی از کاربردهای اصلی روش HOG تشخیص افراد پیاده در تصاویر است. این روش در opencv پیاده‌سازی شده و آماده استفاده است. در مجموعه داده موجود در سایت دانشگاه MIT به آدرس <http://cbcl.mit.edu/cbcl/software-datasets/PedestrianData.html>

با استفاده از HOG پیاده‌ها را بیابید.

پاسخ: پیاده‌سازی ضمیمه شده‌است.

پرسش: نتایج اعمال روش را با تصاویر نمونه از موارد موفقیت و شکست و همچنین با درصد بیان کنید.

پاسخ:

در این روش نمونه‌های موفق بسیاری دیده شد که در زیر آورده شده‌است:



^۹ Hessian blob detector

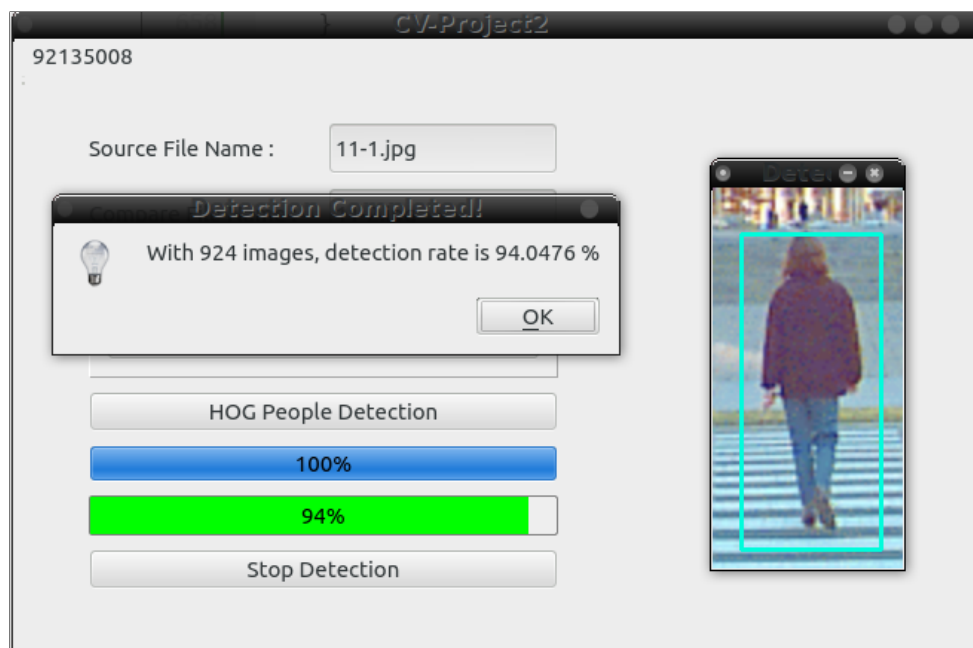
مواردی از شکست نیز آورده شده است:



مواردی از تشخیص اشتباه نیز آورده شده است:



در کل دقت تشخیص را حدود ۹۴ درصد به دست آوردیم:



به نظر می‌رسد با پایین آمدن رزولوشن یا اثر لرزش، و نیز وجود سایه و یا شلوغی بیش از حد، این روش خطای زیادی دارد.

۲.۲ بهبود HOG

پرسش: چه مزایا و معایبی در این روش می‌بینید؟ تحقیقات زیادی به افزایش کارایی روش HOG پرداخته و تاکنون الگوریتم‌های متعددی بر این پایه ارائه شده‌اند. یکی از روش‌های تعمیم یافته از روش HOG را به طور خلاصه توضیح دهید.

پاسخ:

در یکی از کارهای کاملاً به‌روز در این زمینه [۲]، از روشی نوین برای تنظیم پارامترهای این الگوریتم استفاده شده‌است. به این ترتیب که از یک LSVM^{۱۰} استفاده شده‌است و اندازه‌ی پنجره‌ی تشخیص، اندازه‌ی سلول و نسبت ظاهری^{۱۱} را تنظیم می‌کند. در این الگوریتم نمونه‌های آموزشی بهینه می‌شوند تا ماشین بردار پشتیبان بهترین مقادیر برای پارامترهای یادشده برای الگوریتم را یاد گیرد و از یک روش منفی‌کاوی سخت^{۱۲}

^{۱۰} Latent SVM

^{۱۱} Aspect Ratio

^{۱۲} Hard negative mining method

برای جلوگیری از تشخیص‌های اشتباه استفاده شده‌است. با توجه به این نکات، این الگوریتم یک ماشین بردار پشتیبان را برای کاربرد تشخیص انسان بهینه نموده‌است.

مراجع

- [1] Lowe, David G. "Distinctive image features from scale-invariant keypoints." International journal of computer vision 60, no. 2 (2004): 91-110.
- [2] Gao, Chao, Fengcai Qiao, Xin Zhang, and Hui Wang. "An Improved HOG Based Pedestrian Detector." In Foundations and Practical Applications of Cognitive Systems and Information Processing, pp. 577-590. Springer Berlin Heidelberg, 2014.