



دانشکده علوم ریاضی

بینایی کامپیوتر

نیم سال دوم ۱۴۰۰-۱۴۰۱

مدرس: دکتر مصطفی کمالی

---

تمرین سری سوم

---

شماره دانشجویی: ۹۸۱۰۸۱۲۴

نام و نام خانوادگی: نیما کلیدری

---

پرسش ۱

## پرسش ۲

برای حل این سوال، در ابتدا بردارهای ویژگی دو تصویر را با استفاده از *SIFT* استخراج میکنیم. سپس با استفاده از روش *FLANNKDTree* نقاط مشترک آن‌ها را پیدا کرده و با استفاده از روش *knn* با  $n = 2$ ، مچ‌ها را پیدا میکنیم. سپس نقاط نزدیک را پیدا میکنیم و آن‌هایی که از دومین بردار ویژگی نزدیکشان بیشتر از ضریب 0.9 فاصله دارند را انتخاب میکنیم. در نهایت نیز با استفاده از تابع *findFundamentalMat* در تابع *OpenCV*، یک *RANSAC* با مقدار آستانه ۴ و درصد اطمینان ۹۵ درصد و حداکثر تلاش ۵ میلیون، اجرا میکنیم و از آن ماسک *inlier* و *outlier*‌ها و ماتریس *Fundamental* را محاسبه میکنیم. مقادیر این ماتریس بصورت زیر است:

$$F = \begin{bmatrix} -1.12978881e-08 & -2.84874745e-08 & -1.76437229e-04 \\ -1.13712348e-07 & -3.90610135e-08 & -1.41002291e-03 \\ -1.34698817e-04 & 1.76159346e-03 & 1.00000000e+00 \end{bmatrix}$$

را استخراج میکنیم. سپس اینلایر و اوتلایر‌ها را در ۲ آرایه برای نقاط هر تصویر می‌ریزیم. در پایان این بخش نیز برای نقاط هر تصویر با استفاده از تابع *computeCorrespondEpilines*، خطوط اپیپولار برای هر تصویر را بدست می‌آوریم. این خطوط را هم برای نقاط اینلایر و هم اوتلایر به صورت جدا جدا بدست می‌آوریم. سپس نیز با استفاده از تابع *drawlines*، این نقاط را با رنگ‌ها گفته شده در دو تصویر مشخص کرده و در ادامه نیز، تصویرهای بدست آمده را در کنار هم ذخیره میکنیم.

در ادامه نیز با استفاده از رابطه  $Fe = 0$ ، رابطه  $null(F) = e$  را بدست آورده و پس از بدست آوردن  $e$  و همگن کردن آن، مقادیر مورد نظر را بدست می‌آوریم. بصورت مشابه  $e'$  نیز بدست می‌آید. این دو بردار، مقادیر زیر را دارند.

$$e = [-11892.54269039 \quad -1477.02151413 \quad 1]$$

$$e' = [73470.76809325 \quad -8484.24424148 \quad 1]$$

سپس از آنجا که این دو نقطه در خارج از صفحه تصویر خارج شده است، لازم است که دو تصویر خالی که به اندازه کافی بزرگ است را ساخته و نقطه‌ها را روی آن نمایش دهیم. پس از ساختن صفحات مناسب و قرار دادن تصاویر اصلی روی آن، و سپس نصف کردن ابعاد آن (که قابلیت تولید شدن داشته باشند)، و در نهایت رسم نقاط اپی پولار روی آن‌ها، نتایج این بخش را نیز بدست می‌آوریم. درون نتیجه تصویر اول نقطه اپی پولار قرمز در بالا سمت چپ تصویر و درون تصویر دوم (که بسیار بزرگتر است)، این نقطه قرمز در بالا سمت چپ تصویر نمایان است.

در بخش پایانی این سوال، لازم است از لیست نقاط اینلایر بدست آمده، ۱۰ نقطه بصورت تصادفی انتخاب کرده، و سپس با استفاده از تابع گفته شده برای رسم خط، خطوط اپی پولار گذرنده از نقاط مشخص شده را می‌سازیم و توسط تابع *drawlines*، نقاط را به رنگ آبی و خطوط را به رسم خاکستری رسم کرده و این دو تصویر را در کنار هم قرار داده و ذخیره میکنیم.

### پرسش ۳

برای این سوال، از چمدان درون راهروی خوابگاه ، ۷۴ عکس گرفته شده که پس از دادن به نرم افزار Meshroom ، پس از اجرای حدود ۲ ساعت، مدل سه بعدی موجود در فایل پاسخ که تکسچر بندی نیز شده است بدست آمده است.

## پرسش ۴

(۱) برای این بخش، ابتدا تصاویر از فولدر ها با کتابخانه *os* توسط چند حلقه تو در تو دریافت شده و بلافاصله تبدیل به یک بردار یک در  $s^2$  میشود.  $s$  در پاسخ نهایی این سوال بصورت تجربی برابر با ۸ قرار داده شد. سپس یک مدل *Knn* با داده های آموزش و برچسب آنها به تابع *KNearest.create* داده میشوند. سپس دقیقاً به همان صورتی که داده های تصاویر آموزش دریافت شد، داده های آزمایش نیز دریافت میشود و بلافاصله به بردار  $s^2$  تبدیل میشود. حال آگه این بردار را با تابع فاصله  $L2$  و  $n = 1$  به مدل *Knn* تشکیل شده بدهیم، نتیجه ای با دقت حدود ۲۴ درصد بدست می آید.

(۲) در بخش دوم، عیناً مانند بخش اول عمل میکنیم، با این تفاوت که هنگام دریافت یک تصویر، مستقیماً بردار ویژگی هایش را که تعداد آنها را به *limit* محدود میکنیم، توسط *SIFT* استخراج میشود. سپس هر کدام از بردار ها در یک لیست جامع ریخته میشود، اما اینکه مربوط به چه تصویری بوده در یک آرایه ذخیره میشود. سپس بین کل بردار ها، خوشه بندی به روش *k - means* انجام میگردد. بعد از اینکه مشخص شد هر بردار مربوط به کدام دسته است، یک آرایه تماماً صفر با طول تعداد خوشه ها میسازیم که هر کدام از این اعداد، بیانگر این هستند که هر یک از این خوشه ها در چند تصویر آمده اند. بعد از بدست آوردن رابطه  $0.1 - \frac{N}{p[i]}$  که  $N$  تعداد کل تصاویر آموزش هستند و  $p[i]$  نیز تعداد تصاویری که خوشه  $i$  ام در آن ظاهر شده است. سپس کل داده هایی که برای آموزش گردآوری شدند، بنابر اینکه در کدام خوشه هستند، این ضریب بدست آمده در ضرب میشود. به این ترتیب کلمات تصویری ای که کمتر ظاهر میشوند از اهمیت بیشتری برخوردار میشوند. حال یک مدل *Knn* تعریف میکنیم. مانند قبل تصاویر آزمایش را باز هم میخوانیم و هر تصویری که دریافت میشود را، بعد از آنکه بردار های ویژگی هایش را استخراج کردیم، تک تک بردار ها را تست میکنیم و میبینیم به مرکز کدام خوشه نزدیک تر است. به این صورت یک هیستوگرام برای تصویر تست نیز میسازیم. حال با دادن این بردار به *Knn* تعریف شده، میبینیم که در کدام دسته شناسایی شده است و اگر درست بود یکی به یک متغیر زیاد میکنیم و از تقسیم این متغیر بر تعداد کل داده های تست، دقت مورد نظر بدست می آید. حداکثر دقت در این بخش حدود ۴۵ درصد است. قابل ذکر است که تعداد خوشه ها در این بخش ۸۵ عدد و حداکثر تعداد بردار ویژگی ها، ۵۵۰ است. برای  $n$  مربوط به *Knn* نیز، مقدار ۱۵ را داریم.

(۳) در بخش آخر نیز، تنها کاری که کردیم، این است که همان مدل *Knn* درست شده را به مدل *svm* کتابخانه *scikit* تغییر دادیم. در این حالات خوشه ها برابر ۷۵ عدد و حداکثر تعداد بردار ویژگی ها مساوی ۵۵۰ است. در پایان کار نیز با ذخیره سازی دقیق پاسخ های خروجی، ماتریس کانفیوژن را بدست آورده و با دادن آن به کتابخانه *matplotlib*، تصویر مورد نظر سوال را بدست می آوریم. حداکثر دقت در این بخش حدود ۵۲ درصد میباشد.