



دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی برق

بینایی ماشین (Machine Vision)

## Assignment 3

نگارش

محمد مهدی نجفی زاده

علی دانشپور

استاد درس

دکتر زهرا سادات شریعتمداری مرتضوی

آذر ۱۴۰۳

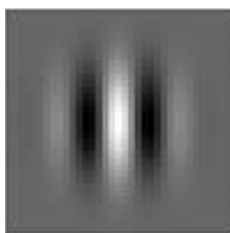
بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

## تمرین‌های برنامه‌نویسی

۱ - یکی دیگر از فیلترهای پر کاربرد در پردازش تصویر، فیلترهای گابور است. کرنل‌های این فیلتر در اندازه و جهات مختلف می‌توانند با تصویر ورودی کانوالو شوند و ویژگی‌های متمایزی را از تصویر استخراج نمایند.

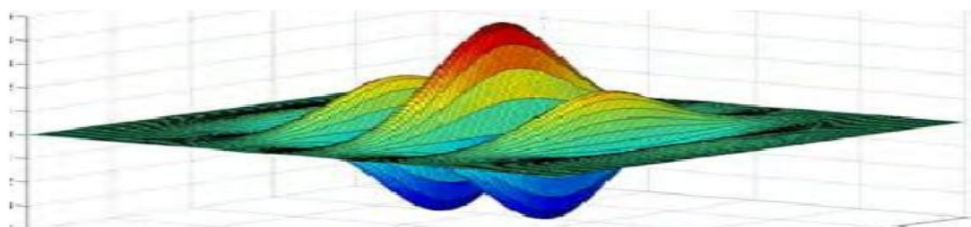
الف) عملکرد این فیلترها را بصورت مختصر توضیح دهید.

یکی از روش‌های استخراج ویژگی‌های بافتی از تصاویر، فیلترهای گابور می‌باشد. این روش رقیبی برای روش‌های LPQ، LBP و دیگر روش‌های استخراج ویژگی‌های بافتی می‌باشد. فیلترهای گابور، فیلترهای خطی هستند که برای تحلیل بافت استفاده می‌شود. این روش فیلترهایی طراحی می‌کند و تصویر را با این فیلترها کانوالو می‌کند. هر فیلتر گابور محتوای فرکانسی در یک جهت خاص در هر محدوده محلی از تصویر را تحلیل می‌کند.



شکل ۱: یک نمونه فیلتر گابور

در شکل ۱ یک نمونه فیلتر گابور در راستای افقی مشخص است. همچنین در شکل ۲، شکل سه بعدی از یک نمونه فیلتر گابور مشخص است.



شکل ۲: شکل ۳ بعدی یک نمونه فیلتر گابور

همچنین فیلترهای گابور با توجه به رابطه زیر ساخته می‌شوند. با تغییر پارامترهای این فرمول مانند  $\sigma$  و  $\theta$  می‌توانیم فیلترهای مختلفی بسازیم. اگر این فیلترها را با زوایای مختلف مانند کرنل روی تصویر بلغزانیم، می‌توانیم اطلاعات خطوط تصویر را استخراج کنیم.

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right)$$

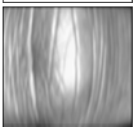
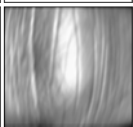
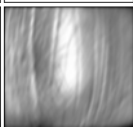
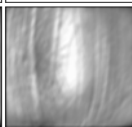
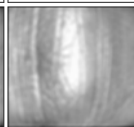
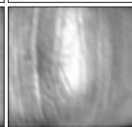
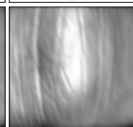
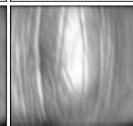

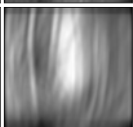
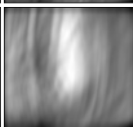

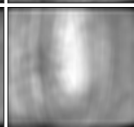
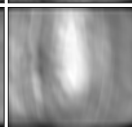
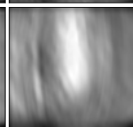

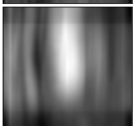
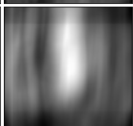
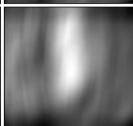
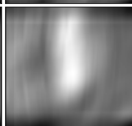
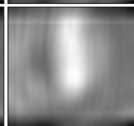
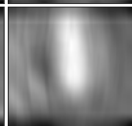
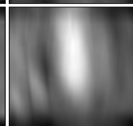
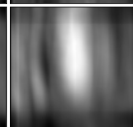

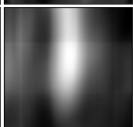
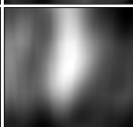
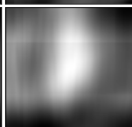
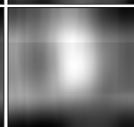
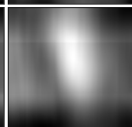
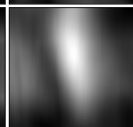
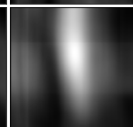
where :

$$x' = x \cos \theta + y \sin \theta$$

$$y' = -x \sin \theta + y \cos \theta$$

ب) این فیلترها را در ۴ اندازه و ۸ جهت مختلف در تصویر پیوست شده image1 کانوالو نمائید و نتیجه را نمایش دهید.

با در نظر گرفتن ۴ اندازه و ۸ زاویه مختلف و اعمال آن‌ها بر روی تصویر image1، شکل ۳ به عنوان خروجی به دست آمد. مقادیر کرنل‌ها و زوایای مختلف در شکل ۳ به وضوح مشخص است.

$\theta$ Size	0	$\frac{\pi}{8}$	$\frac{2\pi}{8}$	$\frac{3\pi}{8}$	$\frac{4\pi}{8}$	$\frac{5\pi}{8}$	$\frac{6\pi}{8}$	$\frac{7\pi}{8}$
7								
15								
31								
63								

شکل ۳: اعمال فیلتر گابور در ۴ اندازه و ۸ جهت مختلف

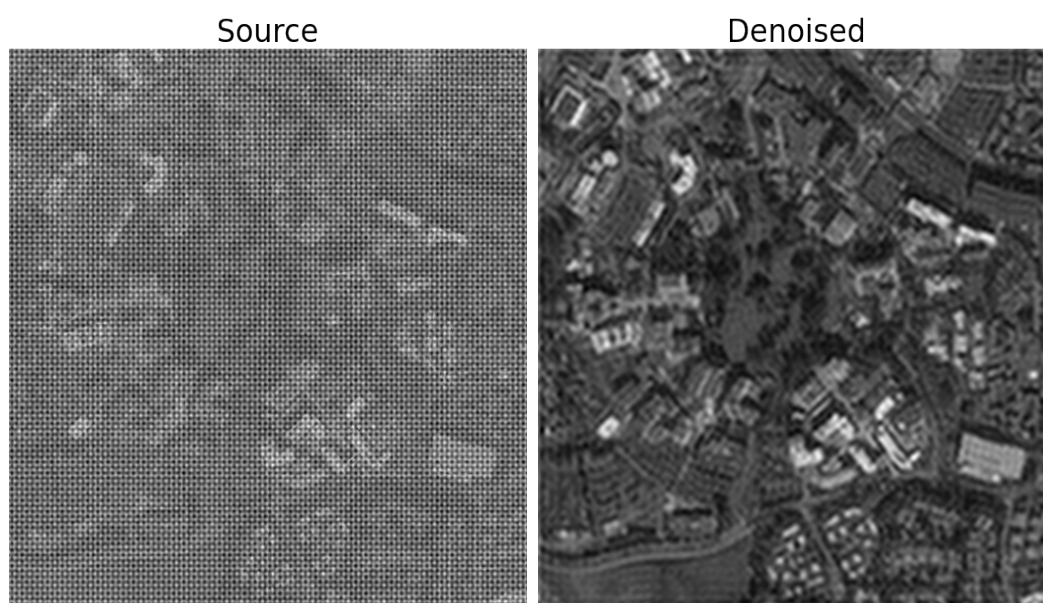
همانطور که از شکل ۳ مشخص است، با توجه به اینکه خطوط تصویر image1 عمودی هستند، فیلترهای در راستای افقی ( $\frac{4\pi}{8}$ ) محوتر و فیلترهای در راستای عمودی (۰) جزئیات بیشتری از تصویر را مشخص کردند. همچنین کرنل‌های با اندازه کوچک‌تر جزئیات دقیق‌تر و شارپ‌تری از تصویر را در مقایسه با کرنل‌های با سایز بزرگ‌تر ثبت کرده‌اند.

### ج) این فیلتر معادل با عملکرد کدام بخش از سیستم بینایی انسان است؟

برخی از پژوهشگران معتقد هستند که قشر بینایی<sup>۱</sup> مغز انسان را می‌توان با عملکرد فیلترهای گابور مدل‌سازی کرد. بنابراین، تجزیه و تحلیل تصویر با فیلترهای گابور را می‌توان شبیه به ادراک در سیستم بینایی انسان در نظر گرفت. در نتیجه فیلتر گابور را می‌توان معادل با عملکرد سیستم قشر بینایی مغز انسان در نظر گرفت.

### ۲- تصویر image2 آغشته به نویز متناوب است. نویز متناوب با ایجاد پترن‌های مشابه باعث تخریب کیفیت تصویر می‌شود. با استفاده از مفاهیم فضای فرکانسی تصویر، این نویز را تشخیص داده و تصویر بهبود یافته را نمایش دهید.

در این سوال از تبدیل فوریه برای حذف نویز تصویر image2 استفاده کردیم. ابتدا با استفاده از تبدیل فوریه تصویر را به حوزه فرکانس منتقل کرده و اجزای فرکانس بالا (که معمولاً نویز هستند) را فیلتر کردیم. سپس با استفاده از تبدیل فوریه معکوس، تصویر بدون نویز را در اختیار داریم. تصویر اولیه و تصویر رفع نویز شده در شکل ۴ مشخص است.



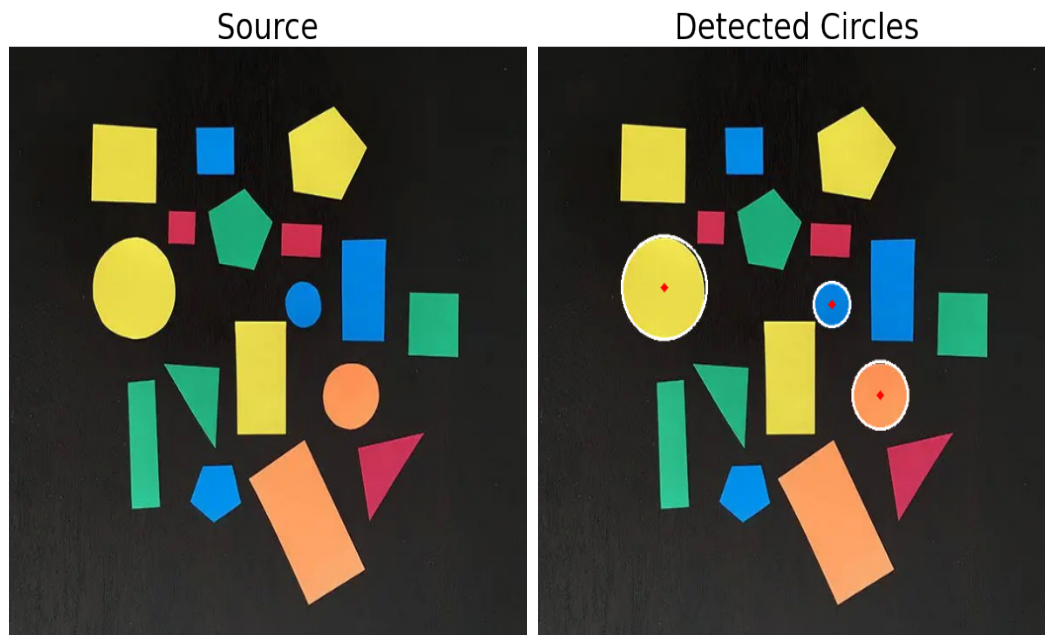
شکل ۴: تصویر اصلی در کنار تصویر بهبود یافته

<sup>1</sup>Visual Cortex

### ۳- دایره‌های موجود در تصویر image3 را دیتکت کنید. در صورت استفاده از توابع آماده تمام پارامترها را به تفکیک مشخص نمایید.

در این سوال از الگوریتم Hough Circle Transform برای تشخیص دایره‌ها در تصویر image3 استفاده کردیم. برای این منظور از کتابخانه OpenCV و تابع cv2.HoughCircles استفاده کردیم. در ادامه پارامترهای تابع cv2.HoughCircles و مقادیر هر پارامتر که در این تمرین استفاده کردیم را، معرفی می‌نمائیم.

- image: تصویر ورودی (image3)
  - $dp = 1$ : معکوس نسبت وضوح accumulator به وضوح تصویر
  - $minDist=20$ : حداقل فاصله بین مراکز دایره‌های شناسایی شده
  - $param1=50$ : لبه‌یاب Canny به دو پارامتر  $minVal$  و  $maxVal$  برای تشخیص لبه احتیاج دارد. مقدار  $maxVal$  برابر با  $Param1$  و مقدار  $minVal$  برابر با  $\frac{Param1}{3}$  می‌باشد.
  - $param2=30$ : این پارامتر، مقدار آستانه accumulator برای شناسایی دایره‌های کاندید را مشخص می‌کند. مقادیر بالاتر منجر به شناسایی حلقه‌های کمتر و دقیق‌تر می‌شود.
  - $minRadius = 1$ : حداقل شعاع دایره
  - $maxRadiuss = 100$ : حداکثر شعاع دایره
- تصویر اولیه و تصویر نهایی که در آن دایره‌ها مشخص شده‌اند، در شکل ۵ مشخص است. محیط دایره‌ها با رنگ سفید و مرکز آن‌ها با رنگ قرمز مشخص شده‌اند.



شکل ۵: تشخیص دایره‌ها در تصویر

## تمرین‌های تشریحی

۱- برای تشخیص خط در الگوریتم RANSAC هر بار دو نقطه به صورت تصادفی انتخاب می‌شود. بنابراین در صورت وجود مجموعه داده زیاد، به تعدادی زیادی تکرار از این الگوریتم نیاز است. با اعمال فرضیات لازم، چگونه می‌توان تحلیل نمود که استفاده از این الگوریتم به تعداد زیادی تکرار نیاز نخواهد داشت؟

در تشخیص خط، روش‌هایی مثل MSE حساسیت بسیار زیادی به داده‌های پرت دارند. یعنی اگر همه‌ی نقاط روی یک خط باشند و حتی یک نقطه پرت داشته باشیم، معادله نامناسبی به دست می‌آورند. الگوریتم RANSAC ایده‌ای برای مقابله با داده‌های پرت می‌باشد. اگر  $N$  تا نقطه داشته باشیم با تعداد تکرار  $\binom{N}{2}$ ، می‌توانیم ۱۰۰ درصد مطمئن باشیم که بهترین خط را انتخاب می‌کنیم. اما این تعداد تکرار بسیار زیاد و از  $O(N^2)$  می‌باشد. بنابراین به دنبال روش‌هایی می‌رویم که بتوانیم حد معقولی روی تعداد تکرارها بذاریم و در واقع به صورت احتمالی با این مسئله برخورد می‌کنیم. فرض می‌کنیم  $W$  نسبت نقاط inlier به کل نقاط می‌باشد.

$$W = \frac{inlier}{inlier + outlier} \quad (1)$$

با توجه به رابطه ۱ به وضوح مشخص است که احتمال اینکه یک مجموعه کاملاً از نقاط inlier تشکیل شود،  $W^K$  می‌باشد.

همچنین  $P$  را احتمال یافتن حداقل یک مجموعه از نقاط بدون Outlier بعد از  $K$  تکرار در نظر می‌گیریم. اگر  $K$  تعداد تکرارها باشد، احتمال آنکه هیچ مجموعه بدون Outlier ای انتخاب نشده باشد  $(1 - P)$ ، با توجه به رابطه ۱ به صورت زیر محاسبه می‌شود:

$$1 - P = (1 - W^K)^K \quad (2)$$

ما به دنبال این هستیم که  $P$  عدد بزرگی شود. در واقع باید ببینیم تعداد تکرار ( $K$ ) را چه عددی در نظر بگیریم تا با احتمال خیلی زیادی ( $P$ ) بگوییم که حتماً یک مجموعه inlier داریم. بنابراین با توجه به رابطه ۲، مقدار  $K$  از رابطه ۳ به دست می‌آید.

$$K = \frac{\log(1 - P)}{\log(1 - W^K)} \quad (3)$$

حال می‌خواهیم ببینیم که اگر  $W = 0.8$  (یعنی ۲۰ درصد داده‌ها پرت باشند) و  $P = 0.999$  (یعنی بخواهیم با دقت ۹۹/۹ درصد مطمئن باشیم که حتماً خط مناسب را پیدا می‌کنیم)، چه تعداد تکرار با توجه به رابطه ۳ مورد نیاز است.

$$K = \frac{\log(1 - 0.999)}{\log(1 - 0.8^7)} = 6.81 \Rightarrow K \simeq 7$$

یعنی تنها با ۷ تکرار می‌توان با دقت ۹۹/۹ درصد از جواب درست اطمینان داشت. در حالی که برای اطمینان ۱۰۰ درصدی باید تعداد تکرار بسیار بسیار بیشتری ( $\frac{N \times (N-1)}{2}$ ) انجام می‌دادیم.

## ۲ - الگوریتمی معرفی نمایید که بتواند بیضی‌های موجود در تصاویر را شناسایی کند.

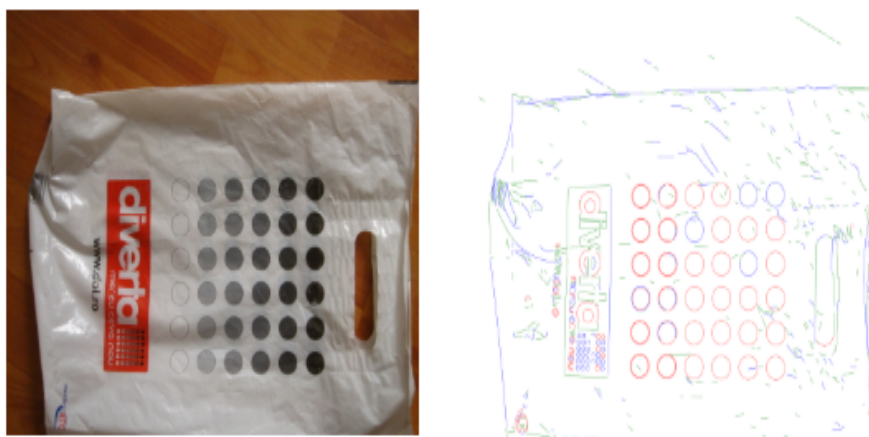
استفاده از الگوریتم Enhanced Line Segment Detector (ELSD) یکی از روش‌های شناسایی بیضی در تصاویر می‌باشد. الگوریتم ELSD یک الگوریتم بینایی ماشین برای تشخیص خط و بیضی در تصاویر دو بعدی می‌باشد. این الگوریتم در سال ۲۰۱۲ و در مقاله [۱] ارائه شده است. تشخیص خط و بیضی اغلب یک پیش نیاز برای کارهای سطح بالا است. از این رو ما به آشکارسازهای خودکار (بدون تنظیم پارامتر) نیاز داریم. همین موضوع انگیزه نویسندگان برای طراحی الگوریتم ELSD شد. در واقع الگوریتم ELSD گسترش یافته الگوریتم LSD می‌باشد.



این الگوریتم با محاسبه گرادیان‌های تصویر از طریق مشتقات مرتبه اول شروع می‌شود. ابتدا تغییر رنگ‌های پیکسل‌ها در سراسر تصویر را بررسی کرده و سپس پیکسل‌های مجاور که الگوهای تغییر مشابهی دارند را گروه‌بندی می‌کند. این گروه‌بندی‌ها به شناسایی لبه‌ها یا منحنی‌های بالقوه در تصویر کمک می‌کند. سپس الگوریتم ELSD این پیکسل‌های گروه‌بندی شده را با دقت بیشتری تحلیل می‌کند تا مشخص کند آیا آن‌ها خطوط یا منحنی‌های واقعی را تشکیل می‌دهند یا تنها الگوهای تصادفی هستند که به طور اتفاقی ظاهر شده‌اند.

پس از شناسایی ویژگی‌های بالقوه، الگوریتم ELSD تلاش می‌کند تا اشکال هندسی مختلفی نظیر خطوط مستقیم، کمان‌های دایره‌ای، یا کمان‌های بیضوی را با گروه‌های پیکسلی تطابق دهد. سپس، شکلی را که بهترین تطابق با داده‌ها دارد، انتخاب می‌کند. سپس با ادغام اشکال مشابهی که در نزدیکی یکدیگر قرار دارند و حذف تشخیص‌های تکراری، نتایج را تصحیح و بهینه می‌کند. این رویکرد باعث می‌شود که الگوریتم ELSD عملکرد قابل توجهی در تحلیل تصاویر فنی، نقشه‌های معماری، و سایر تصاویر نیازمند تشخیص دقیق اشکال هندسی، حتی در شرایطی که تصاویر دارای نویز یا کیفیت پایین هستند، داشته باشد.

یک نمونه از خروجی الگوریتم ELSD در کنار تصویر اصلی در شکل ۶ مشخص است.



شکل ۶: نتیجه استفاده از الگوریتم ELSD

## منابع و مراجع

- [1] Patraucean, Viorica, Gurdjos, Pierre, and Gioi, Rafael. *A Parameterless Line Segment and Elliptical Arc Detector with Enhanced Ellipse Fitting*, volume 7573, pages 572–585. 01 2012.