

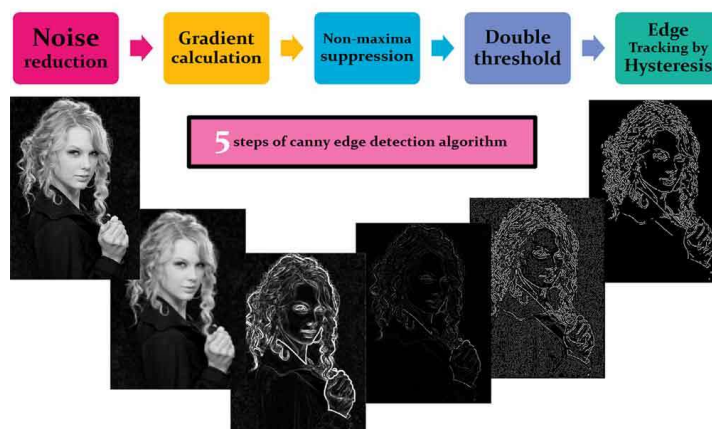
تمرین سری پنجم درس تصویربرداری رقمی

پوریا محمدی نسب

(۴۰۰۷۲۲۱۳۸)

## ۲- الف) تمام مراحل لبه یاب Canny را نام برده و دلیل استفاده از هر مرحله را توضیح دهید. (۵ امتیاز)

شناسایی لبه در روش Canny شامل شش مرحله است. **مرحله اول** فیلتر کردن تصویر اولیه و حذف نویز از تصویر است. برای این منظور فیلتر گوسین با یک ماسک ساده قابل استفاده است که به طور انحصاری در الگوریتم Canny استفاده می‌شود. **مرحله دوم** یافتن لبه‌های قوی با استفاده از gradient magnitude در هر نقطه است که برای این منظور معمولاً از ماسک سوئل استفاده می‌شود. **مرحله سوم** بدست آوردن جهت لبه‌های تصویر، با استفاده از مقدار gradient در راستای X و Y که در مرحله قبل محاسبه شده است. **مرحله چهارم** اختصاص دادن جهت‌های قابل قبول در تصویر به جهت‌های بدست آمده است. برای هر پیکسل در تصویر فقط ۴ جهت ۰، ۴۵ و ۹۰ و ۱۳۵ درجه امکان‌پذیر می‌کنیم. بنابراین جهت‌های بدست آمده را به یکی از این ۴ جهت Map می‌کنیم. **مرحله پنجم** به مرحله suppression non-maximum معروف است. این مرحله جهت لبه‌ها را بررسی می‌کند و از بین آن‌ها، لبه‌هایی را که نمی‌شود به عنوان لبه در نظر گرفت را حذف می‌کند. این مرحله یک خط باریک در تصویر نهایی می‌دهد. **مرحله ششم** آستانه‌گیری در تشخیص لبه است. در Canny از روشی به نام hysteresis استفاده می‌شود. برای این منظور دو آستانه بالا و پایین تعریف می‌کنیم هر پیکسل که دارای gradient بیشتر از حد بالا باشد به عنوان لبه پذیرفته می‌شود و در صورتی که دارای مقدار کمتر از حد پایین باشد، رد می‌شود و در صورتی که دارای مقداری بین این دو حد باشد، در صورتی پذیرفته می‌شود که یکی از همسایه‌های آن پذیرفته شده باشد. الگوریتم canny در مقایسه با الگوریتم‌های دیگر از دقت بالایی برخوردار است ولی پیچیدگی محاسباتی دارد و کمی کندتر عمل می‌کند.



## ۲- ج) لبه یاب Sobel را با لاپلاسیان مقایسه کنید. آیا لاپلاسیان لبه یاب مناسبی است؟ (۵ امتیاز)

به طور کلی دو روش برای تشخیص لبه داریم (sobel و laplacian). روش sobel از دو کرنل استفاده می‌کند که بینی لبه‌های جهت افقی و دیگری لبه‌های جهت عمودی را در تصویر تشخیص می‌دهند. این در حالی است که روش laplacian از یک کرنل استفاده می‌کند. هر دو روش، معمولاً تحت تاثیر نویز که یک جزء جدا نشدنی از تصویر است، قرار می‌گیرند. لاپلاسیان همانند مشتق (sobel) هستش و میزان تغییرات در نواحی مختلف تصویر را نشان می‌دهد ولی با حساسیت بیشتر از sobel. به این دلیل که لاپلاسیان مشتق دوم هست. یکی از تفاوت‌های لاپلاسیان نسبت به sobel شکل موج آنها هست در sobel ما peak را روی لبه داریم ولی در لاپلاسیان ما peak را قبل لبه و بعد از آن یک دره داریم. یعنی در سمت سیاه تصویر مثبت peak داریم و در سمت سفید (آبجکت) منفی دره داریم در مقایسه با sobel که در واقع zero crossing در لاپلاسیان در مرکز آن وجود دارد.

۳- الف) می‌خواهیم از الگوریتم RANSAC برای یافتن پارامترهای یک دایره استفاده کنیم. در صورتی که ۵۰ درصد از نقاط موجود مربوط به دایره و بقیه نقاط پرت باشند، برای اینکه دقت انتخاب پارامترها به ۰,۹۹۹۹۹۹ برسد به چند تکرار نیاز داریم؟ (۵ امتیاز)

$$k = \frac{\log(1 - P)}{\log(1 - w^2)}, P = 0.99999, w = 0.5$$

$$k = \frac{\log(0.000001)}{\log(0.75)} = \frac{-6}{-0.125} = 48$$

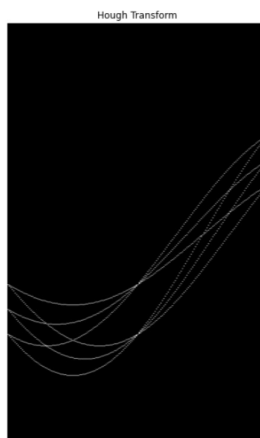
۳- ج) با مطالعه این منبع، Least Median of squares را توضیح دهید و با RANSAC مقایسه کنید. (۵ امتیاز)

الگوریتم LMedS پارامترها با حل کردن یک رابطه بهینه سازی تخمین میزند. درواقع در این روش قصد ما train کردن  $h_{cur}$  است و برای هر مشاهده مقدار باقی مانده ها (residuals) محاسبه میشود. سپس میانه این  $r$  ها بدست آمده و با  $r_{cur}$  مقایسه میشود اگر مقدار بیشتری داشت جایگزین  $r_{cur}$  میشود و همینطور تا انتها تکرار میکنیم تا در نهایت بهترین  $r$  و  $h$  را بازمیگرداند. تفاوت عمده آن با روش RANSAC این است که روش RANSAC به دنبال جمعیت پیکسل ها با بیشترین چگالی ممکن میگردد ولی روش LMedS شامل نازک ترین خطی است که شامل ۵۰٪ نقاط باشد.

۳- د) در یک تصویر تعدادی لبه از یک دایره موجود است. در بین لبه ها داده پرت هم وجود دارد. برای اینکه پارامترهای دایره را خیلی دقیق بدست بیاوریم چگونه از RANSAC باید استفاده کنیم؟ (۵ امتیاز) (راهنمایی: در RANSAC برای تعیین پارامترها از تعدادی محدودی نقطه استفاده می شود).

رابطه ی بدست آمده برای تعیین پارامترهای RANSAC مشخص میکند که برای افزایش احتمال (دقت) باید مقدار  $p$  افزایش یابد که در ادامه ی آن مقدار  $1 - p$  کاهش پیدا میکند. میتوانیم با دادن یک مقدار بسیار نزدیک به 1 ، پارامتر  $k$  یعنی تعداد تکرار را بالا برد. در بعضی منابع اشاره شده است که برای کاهش احتمال انتخاب نقاط پرت میتوان از تکنیکی شبیه به رای گیری، کاری کرد تا نقاط پرت در رابطه ی ما که بر پایه ی احتمالات است با احتمال کمتری انتخاب شوند البته مشکل این راه حل این است که سربار حافظه و زمان دارد.

۴ - الف) تبدیل hough زیر مربوط به چه تصویری است؟



با توجه به شکل که دو نقطه متمرکز در شکل داریم. این شکل با احتمال زیادی تبدیل hough دو خط موازی است. به این دلیل که زوایای برابری دارند ولی  $r$  های آنها متفاوت است که این مشخصات نشان دهنده دو خط موازی است. همینطور از شکل میتوان حدس زد که برای یافتن هر خط از سه نقطه استفاده شده است.

۴- ب) شبه کدی برای تبدیل Hough دایره و با استفاده از جهت گرادیان بنویسید (راهنمایی: هر نقطه بر روی دایره میتواند به تعداد زیادی دایره نگاشت شود که با استفاده از گرادیان این موضوع باید حل شود). (۱۰ امتیاز)

Initialize the accumulator ( $H[a,b,r]$ ) to all zeros

Find the edge image using any edge detector

For  $r = 0$  to diagonal image length

For each edge pixel  $(x,y)$  in the image

For  $\Theta = 0$  to  $360$

$$a = x - r \cdot \cos \Theta$$

$$b = y - r \cdot \sin \Theta$$

$$H[a,b,r] = H[a,b,r] + 1$$

Find the  $[a,b,r]$  value(s), where  $H[a,b,r]$  is above a suitable threshold value

## References

- 1) [www.matlab1.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%D9%84%D8%A8%D9%87-%DB%8C%D8%A7%D8%A8%DB%8C-canny/](http://www.matlab1.ir/%D8%A7%D9%84%DA%AF%D9%88%D8%B1%DB%8C%D8%AA%D9%85-%D9%84%D8%A8%D9%87-%DB%8C%D8%A7%D8%A8%DB%8C-canny/)
- 2) <https://towardsdatascience.com/canny-edge-detection-step-by-step-in-python-computer-vision-b49c3a2d8123>
- 3) <https://onlinebme.com/canny-edge-detection-algorithm/>
- 4) <https://cse442-17f.github.io/Sobel-Laplacian-and-Canny-Edge-Detection-Algorithms/>
- 5) <https://medium.com/@nikatsanka/comparing-edge-detection-methods-638a2919476e>
- 6) [www.7khatcode.com/2611/%D9%84%D8%A7%D9%BE%D9%84%D8%A7%D8%B3%DB%8C%D9%86-%D8%AA%D8%B5%D9%88%DB%8C%D8%B1-laplacian](http://www.7khatcode.com/2611/%D9%84%D8%A7%D9%BE%D9%84%D8%A7%D8%B3%DB%8C%D9%86-%D8%AA%D8%B5%D9%88%DB%8C%D8%B1-laplacian)
- 7) <https://medium.com/mllearning-ai/outlier-detection-using-the-ransac-algorithm-de52670adb4a>
- 8) <https://theailearner.com/tag/hough-circle-transform-algorithm/>