

فریمه رشیدی(99222040) - تمرین سری دوم - سوالات تئوری

مسئله ۱)

هرکدام از کرنل های زیر، این وظیفه را به عهده دارند که لبه ها را در یک جهت خاص پیدا کنند و نشان دهند. بنابراین تفاوت آنها در جهتی است که لبه ها را نشان می دهند. کرنل ها و تصاویر مربوط به آن ها از چپ به راست:

1.

0	-1	0
0	0	0
0	1	0

این کرنل همانطور که از ظاهرش پیداست، لبه های در جهت عمودی را پیدا می کند. این کرنل در روش های لبه یابی ساده استفاده می شود. نحوه ی کارش به این صورت است که با سنجیدن تفاوت **intensity** بین دو پیکسل(بالا و پایین)، وجود لبه در پیکسل مرکزی را پیدا می کند. از بین تصاویر داده شده، این کرنل مربوط به تصویر 4 است.

2.

0	0	0
-1	0	1
0	0	0

این کرنل همانطور که از ظاهرش پیداست، لبه های در جهت افقی را پیدا می کند. در این کرنل مانند کرنل لبه یاب عمودی، زمانی که در تصویر از تاریکی به روشنایی از چپ به راست عبور می کنیم، مقادیر مثبت تولید می شوند. برعکس، زمانی که در تصویر از روشنایی به تاریکی از چپ به راست عبور می کنیم، مقادیر منفی تولید می شوند. با توجه به توضیحات بالا، این کرنل مربوط به تصویر 2 است.

3.

0	0	-1
0	0	0
1	0	0

این کرنل لبه ها را به صورت اُریب تشخیص داده و لبه ها را نشان می دهد. از ظاهر کرنل قابل حدس است که تغییرات از گوشه ی بالایی راست به گوشه ی پایینی چپ را تشخیص می دهد و لبه ها را اینگونه می یابد. با توجه به این توضیحات، این کرنل به تصویر 1 مربوط است.

4.

-1	0	0
0	0	0
0	0	1

عملکرد این کرنل برعکس کرنل قبلی است. یعنی به صورت مورب از گوشه ی پایینی چپ به گوشه ی بالایی راست، تغییرات را تشخیص داده و لبه یابی می کند. بنابراین مربوط به تصویر 3 است.

مسئله ۲)

فیلتر لاپلاسین نسبت به **sobel** و **canny** معایبی دارد که باعث می شود در عمل کمتر استفاده شود. ابتدا به توضیح مختصری از هر کدام از این فیلترها میپردازیم:

فیلتر لاپلاسین: این فیلتر در واقع لبه ها را بر اساس محاسبه ی مشتقات دوم تصویر، پیدا می کند. فیلتر لاپلاسین با استفاده از عملگر لاپلاسین، تغییرات **intensity** روشنایی را در عکس شناسایی می کند و آن پیکسل هایی که تغییرات ناگهانی دارند، به عنوان لبه ارائه می کند.

فیلتر **sobel**: این فیلتر در واقع لبه ها را بر اساس محاسبه ی مشتقات اول تصویر، پیدا می کند. دو ماسک عمودی و افقی روی تصویر اعمال می شود و این دو ماسک تغییرات شدت روشنایی را در دو جهت عمودی و افقی اندازه می گیرند و با هم ترکیب می شوند.

فیلتر **canny**: این فیلتر از چندین مرحله برای تشخیص لبه ها بهره می برد. ابتدا یک فیلتر گوسی برای کاهش نویز روی عکس اعمال می شود. سپس گرادیان تصویر در جهات مختلف حساب می شود. سپس در مرحله ی بعد لبه ها نازک تر و دقیق تر می شوند. و در نهایت یک آستانه ای برای تشخیص لبه های قوی از ضعیف معین شده و طبق آن، لبه های ضعیفی که به لبه های قوی متصل اند شناسایی شده و باقی لبه های ضعیف حذف می شوند.

حال به دلایل ارجحیت فیلترهای sobel و canny بر لاپلاسین میپردازیم:

1. حساسیت نسبت به نویزها:

فیلتر canny با روش های نرم افزاری مثل گوسی و sobel با استفاده از مشتقات اول کار می کنند. بنابراین نسبت به لاپلاسین که با محاسبه ی مشتقات دوم کار می کند، بهتر نویزها را کاهش می دهند و باعث می شود در برابر نویز مقاوم تر باشند. ولی فیلتر لاپلاسین خیلی نسبت به نویزها حساسیت دارد (چون با مشتقات دوم کار می کند). نویزها در این روش تقویت می شوند و در نتیجه لبه ها اشتباهاً شناسایی می شوند.

2. عملکرد ضعیف در تشخیص لبه های ضعیف:

فیلتر لاپلاسین در نقاطی که تغییرات شدید دارند، حساسیت بالایی دارد ولی معمولاً لبه های ضعیف و با شدت کم را به خوبی تشخیص نمی دهد. بنابراین می توان گفت در تشخیص لبه های ضعیف، عملکرد ضعیفی دارد. در مورد تشخیص لبه های ضعیف، فیلتر canny عملکرد خوبی دارد چون در یک مرحله لبه ها را نازک تر و دقیق تر می کند که Non-maximum Suppression نام دارد.

3. عدم جهت گیری لبه ها:

فیلتر لاپلاسین بدون در نظر گرفتن جهت تغییرات شدید در شدت روشنایی، فقط تغییرات شدید در شدت روشنایی را تشخیص می دهد. یعنی درباره ی جهت لبه ها، اطلاعاتی نمی دهد. ولی فیلتر canny جهت گیری لبه ها را مشخص کرده و به طور دقیقی لبه ها را شناسایی می کند. فیلتر sobel نیز جهت گیری لبه ها را مشخص می کند.

مسئله ۳)

یک ماتریس 2×2 است که از مشتقات تصویر محاسبه می شود. در تعریف M ، در واقع I_x و I_y به ترتیب مشتقات تصویر در جهت x و y هستند. M را structure tensor می گوئیم.

به تابع زیر توجه کنید:

$$R = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2 = \det(M) - k \operatorname{tr}(M)^2$$

این تابع R که متشکل از مقادیر ویژه ی structure tensor است، گوشه را بیان می کند.

مقادیر ویژه مشخص می کنند که یک ناحیه، لبه یا گوشه یا مسطح (flat) است.

اگر هر دو مقادیر ویژه بزرگ باشند و $\lambda_1 = \lambda_2$ ، پس R بزرگ و ناحیه مورد نظر یک گوشه است.

اگر هر دو مقادیر ویژه کوچک باشند، پس R کوچک است و ناحیه مسطح (flat) است.

اگر $\lambda_1 < \lambda_2$ برقرار باشد یا برعکس، R از صفر بزرگتر و ناحیه یک لبه است.

حال اگر بخواهیم ارتباط تصاویر را با فرایند بگوییم، می توان گفت تصویر داده شده مربوط به مقادیر ویژه در ماتریس M است. اگر $\lambda_1 = \lambda_2$ پس می توان گفت که گوشه ای در منطقه وجود دارد. بنابراین از این روش از طریق محاسبه ی مشتقات، می توان مشخص کرد که ناحیه ای دارای گوشه هست یا نه.

برای بخش سوم سوال، ابتدا باید I_x را حساب کنیم.

ما I_x را به شکل زیر از صورت سوال داریم.

0	0	1	4	9
1	0	5	7	11
1	4	9	12	16
3	8	11	14	16
8	10	15	16	20

در اینجا مرکز هر d/dx را روی هر پیکسل هایلایت شده در تصویر قرار دادیم.

از سوال d/dx را داریم. اگر d/dx را روی هر پیکسل هایلایت شده اعمال کنیم، I_x به شکل زیر به دست می آید:

4	7	6
8	8	7
8	6	5

سپس از تکرار همین مراحل برای d/dy خواهیم داشت:

ly :

4	8	8
8	6	7
6	6	4

حال ماتریس هریس را حساب می کنیم. بنابراین مقادیر ماتریس زیر را محاسبه می کنیم.

lx^2	$lxly$
$lxly$	ly^2

H :

403	385
385	381

برای محاسبه ی R داریم:

$$R = (403 \cdot 381 - 385 \cdot 385) - 0.04 \cdot (403 + 381)^2 = -19268.24$$

حاصل R منفی شد پس در تصویر ذکر شده لبه اتفاق افتاده است.

مسئله 4)

1. کاهش نویز

ما از فیلتر گاوسی 3×3 استفاده خواهیم کرد.

$$\frac{1}{16} * \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

این فیلتر را بر روی هر پیکسل تصویر اعمال می‌کنیم. نتیجه نهایی تصویر زیر است:

0	0	0	0	1	1	1	0	0	0	0
0	0	0	1	1	2	1	1	0	0	0
1	1	4	5	7	13	7	5	4	1	1
1	1	7	6	8	14	8	6	7	1	1
2	2	7	5	11	15	11	5	7	2	2
2	5	11	9	12	15	12	9	11	5	2
2	2	7	5	11	15	11	5	7	2	2
1	1	7	6	8	14	8	6	7	1	1
1	1	4	5	7	13	7	5	4	1	1
0	0	0	1	1	2	1	1	0	0	0
0	0	0	0	1	1	1	0	0	0	0

2. محاسبه گرادیان

اپراتورهای سوبل را برای تصویر صاف شده اعمال می‌کنیم تا گرادیان‌ها را پیدا کنیم. اپراتور سوبل افقی یا G_x :

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

اپراتور سوبل عمودی یا Gy:

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

حالا برای هر پیکسل جهت گرادیان را محاسبه می‌کنیم. برای مثال محاسبات برای پیکسل (۱ و ۱) به شکل مقابل است:

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 2 \\ 1 & 3 & 5 \end{bmatrix}$$

$$G_x = (0 \cdot -1 + 0 \cdot 0 + 1 \cdot 1) + (0 \cdot -2 + 1 \cdot 0 + 2 \cdot 2) + (1 \cdot -1 + 3 \cdot 0 + 5 \cdot 1) = 1 + 4 + 4 = 9$$

$$G_y = (0 \cdot 1 + 0 \cdot 2 + 1 \cdot 1) + (0 \cdot 0 + 1 \cdot 0 + 2 \cdot 0) + (1 \cdot -1 + 3 \cdot -2 + 5 \cdot -1) = 1 + 0 - 12 = -11$$

$$G = (9^2 + 11^2)^{1/2} = 14.21$$

$$\text{Theta} = \tan^{-1}(G_y/G_x) = -0.014$$

3. Non maximum suppression

لبه‌ها را با حذف پیکسل‌هایی که در جهت گرادیان، ماکزیمم محلی نیستند، نازک‌تر می‌کنیم. به عنوان مثال، اگر پیکسل (۱ و ۱) با زاویه $\theta = -0.014$ (نزدیک به قطر) باشد، آن را با همسایگان قطری‌اش مقایسه می‌کنیم. اگر بزرگی گرادیان در (۱ و ۱) بیشتر از هر دو همسایه نباشد، نازک‌تر می‌شود.

4. آستانه گذاری

با استفاده از دو آستانه minVal و maxVal:

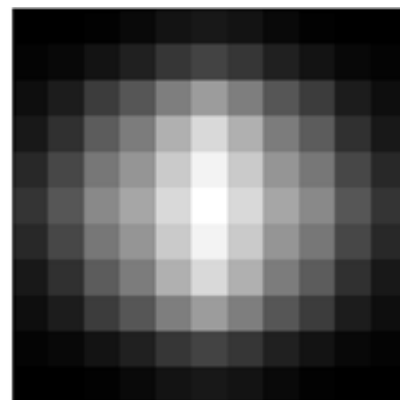
- پیکسل‌هایی با بزرگی گرادیان بالاتر از maxVal به عنوان لبه‌های قوی نگه داشته می‌شوند.
- پیکسل‌هایی با بزرگی گرادیان کمتر از minVal حذف می‌شوند.
- پیکسل‌هایی با بزرگی گرادیان بین minVal و maxVal نگه داشته می‌شوند اگر به لبه‌های قوی متصل باشند.

نتیجه نهایی تصویر زیر با استفاده از کد پایتون:

عکس اصلی:



عکس با فیلتر گوسی:



عکس نهایی:



