



University of Tehran

School of Electrical and Computer Engineering



Digital Image Processing

Instructor: Hamid Soltanian-Zadeh

Assignment 9

Chapter 11 – Representation and Description

Sasan Keshavarz

810199253

Spring 2022

فهرست

چکیده	۱
سوال ۱	۲
بخش اول	۲
بخش دوم	۳
بخش سوم	۳
بخش چهارم	۴
سوال ۲	۵
سوال 3	۶
بخش اول	۶
بخش دوم	۶
سوالات تحلیلی	۸
11.1.a	۸
11.1.b	۸
11.3.a	۸
11.3.b	۸
10.10.a	۹
10.10.b	۹
10.11.a	۹
10.11.b	۹
10.19.a	۱۰
10.19b	۱۰

۱۰ 11.26

۱۲ پیوست ۱: روند اجرای برنامه

۱۳ مراجع

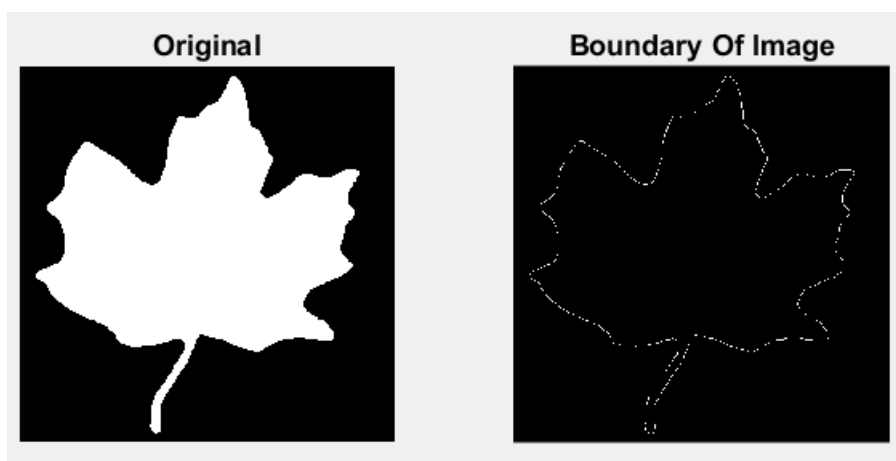
در این سری از تمرین هدف پیاده سازی یک سری روش ها برای توصیف و دستکاری مرز اشکال و نواحی با کمک یک سری الگوریتم ها و البته عملگرهای مورفولوژیکی است. همچنین خواص مربوط به بافت تصاویر و امضای جسم برای تشخیص نوع آن نیز بررسی خواهد شد.

سوال ۱

بخش اول

در این بخش می خواهیم الگوریتم تعقیب مرز که در کتاب آمده را در قالب تابع پیاده سازی کنیم. ابتدا پیکسل شروع (شمالی ترین و غربی ترین) را به دست می آوریم. سپس نقطه پس زمینه را انتخاب میکنیم که در اینجا ما نقطه west را در نظر میگیریم.

باقی مراحل در کد نوشته شده به صورت کامنت گذاری کامل توضیح داده شده است. برای حرکت ساعتگرد نیز از یک آرایه offset استفاده کرده ایم که همان اختلاف پیکسل های مجاور با پیکسل وسط با شروع از نقطه west به صورت ساعتگرد است. همچنین بعد از کشف هر پیکسل مرز جدید مختصات آن را برای استفاده در قسمت دوم در آرایه ی indx و indy ذخیره می نماییم. نتیجه حاصل به شرح زیر خواهد بود:



شکل ۱ شکل اصلی و مرز به دست آمده از آن

بخش دوم

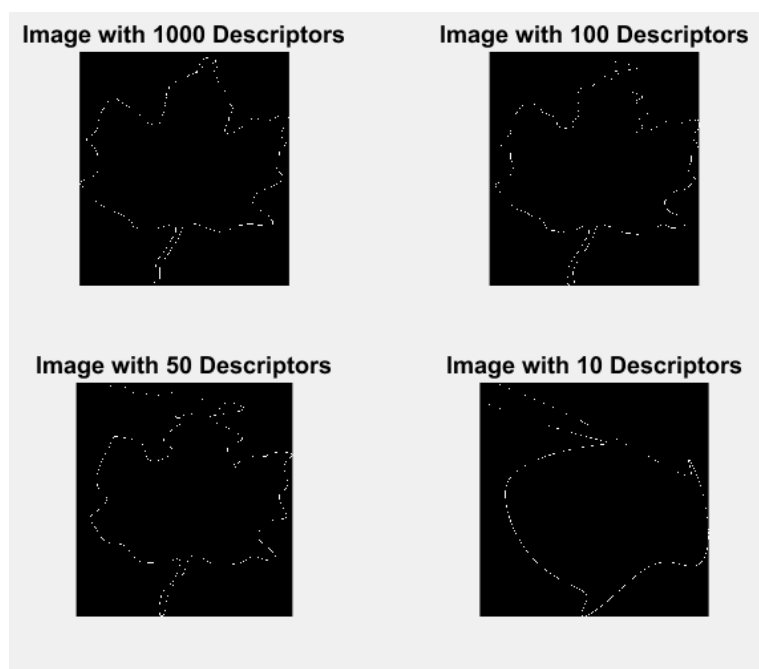
در این بخش با استفاده از مختصات به دست آمده از قسمت قبل و کدگذاری نتیجه به شرح زیر (بخشی از زنجیره) خواهد بود:

1113	1114	1115	1116	1117	1118	1119	1120	1121	1122	1123	1124	1125	1126	1127	1128
2	2	2	2	2	2	2	2	3	2	3	4	3	4	3	4

شکل ۲ بخشی از زنجیره به دست آمده

بخش سوم

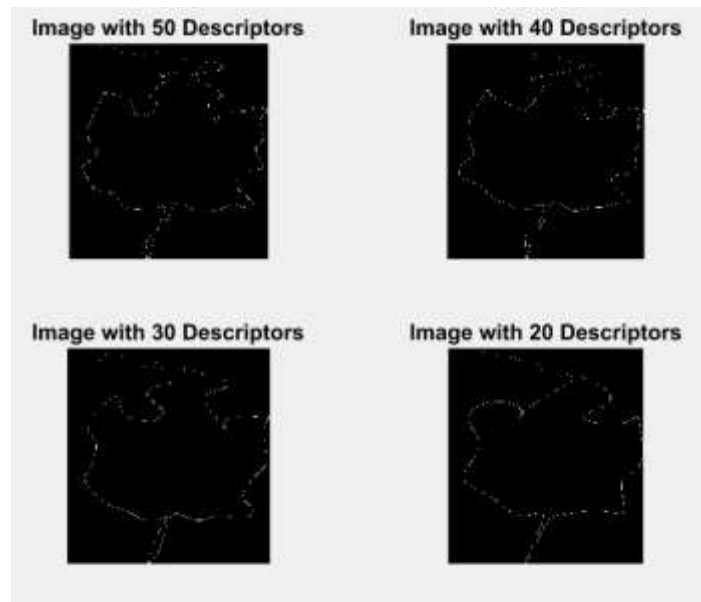
در این قسمت با استفاده از تابع `my_fdescriptor` که نوشته ایم مرزهای جسم را به وسیله تعداد مختلف خواسته شده از توصیفگرها دستکاری میکنیم و نتایج را به شرح زیر به دست می آوریم:



شکل ۳ توصیفگرهای فوریه

بخش چهارم

ملاحظه میکنیم که عدد ۴۰ تقریباً تعداد توصیفگرهای لازم را بیان میکند. برای این کار اعداد ۵۰، ۴۰، ۳۰ و ۲۰ انتخاب شدند که نتیجه مناسب را با ۴۰ میتوان گرفت.



شکل ۴ توصیفگرهای فوریه برای ایفتن کمترین توصیفگر مورد نیاز

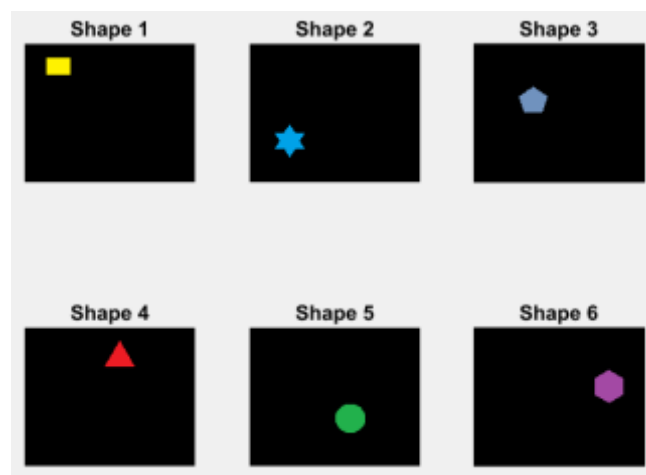
سوال ۲

در این سوال با استفاده از الگوریتم بیان شده در کتاب قصد داریم اسکلت اثر انگشت را به دست بیاوریم. ابتدا یک حلقه بینهایت ایجاد میکنیم. سپس مرحله به مرحله همسایگی ها را به دست آورده و الگوریتم شرح داده شده را روی آن اعمال می کنیم. با این کار یک حلقه به پایان می رسد. سپس حلقه دوم و الی آخر را اجرا میکنیم تا جایی که پیکسل جدیدی صفر نشود. در اینجا الگوریتم به پایان می رسد. کلیه این مراحل مجدد در تابع `my_bwskel` شرح داده شده است. متأسفانه نتوانستم از این کد نتیجه مطلوب بگیرم اما کد تابع و کد سوال اصلی را هرچقدر گشتم ایرادی نداشتند و باید منطقاً این الگوریتم جواب میداد.

بخش اول

با استفاده از تابع `bwconncomp` متلب و خواص مربوط به آن هر شکل را جدا می سازیم و در قالب

یک تصویر جداگانه به شرح زیر نمایش می دهیم:



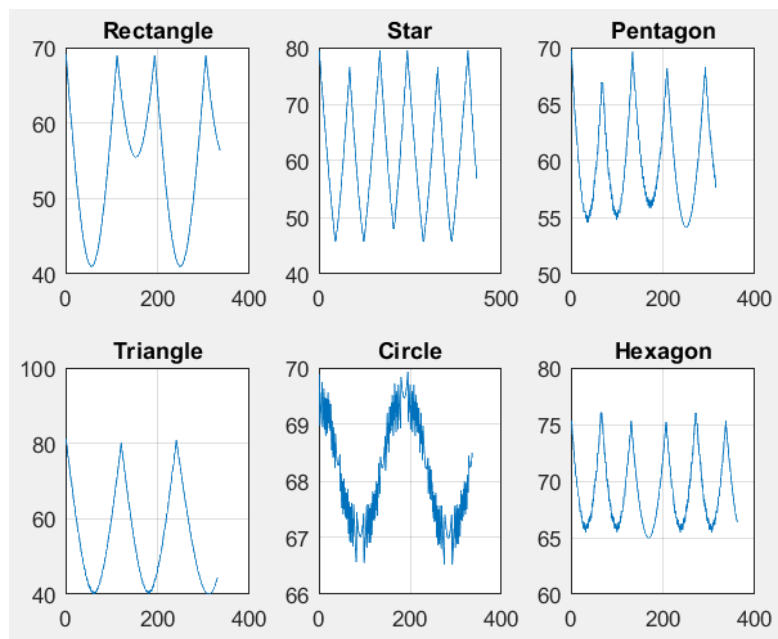
شکل ۵ شکل های جداسازی شده

بخش دوم

ابتدا با استفاده از تابع `regionprops` و متد `"Centroid"` آن مرکز اشکال را به دست می آوریم.

مرز اجسام را نیز با تابع `bwboundary` محاسبه میکنیم و حال با یک قضیه فیثاغورث ساده فواصل

را به دست می آوریم و همانند زیر نمایش می دهیم .



شکل ۵ امضای اشکال موجود در تصویر

متلب نقطه شروع را گوشه شمال غربی در نظر میگیرد بنابراین علت اختلاف شکل های بالا با کتاب به این علت خواهد بود (کتاب از وسط ضلع شرقی شروع میکند) و همچنین شکل ها ایده آل نیستند.

سوالات تحلیلی

11.1.a

برای حل این سواب باید مشخص کرد که ارزش هر عنصر در یک کد زنجیره ای نسبت به مقدار قبلی خود است. کد یک مرز که به روشی ثابت ردیابی می شود (مثلاً در جهت عقربه های ساعت) یک مجموعه دایره ای منحصر به فرد از اعداد است. شروع در مکان های مختلف در این مجموعه، ساختار دنباله دایره ای را تغییر نمی دهد. انتخاب کوچکترین عدد صحیح به عنوان نقطه شروع به سادگی همان نقطه را در دنباله مشخص می کند. حتی اگر نقطه شروع منحصر به فرد نباشد، این روش همچنان یک توالی منحصر به فرد ارائه می دهد. به عنوان مثال، دنباله ۱۰۱۰۱۰ دارای چند نقطه شروع ممکن است، اما همه آنها کوچکترین عدد صحیح ۰۱۰۱۰۱ را ایجاد می کنند.

11.1.b

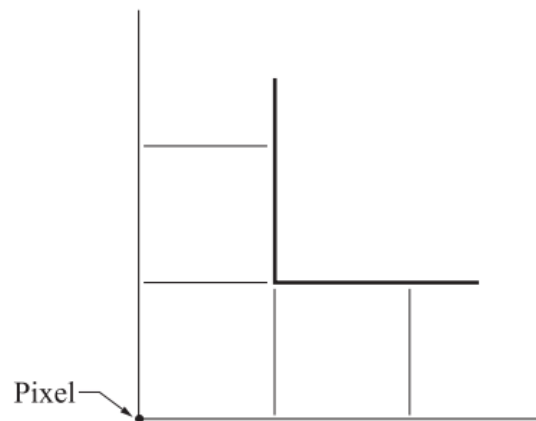
کد ۱۱۰۷۶۷۶۵۵۴۳۳۲۲۱۱ است. نقطه شروع استاندارد ۰ است که دنباله ۰۷۶۷۶۵۵۴۳۳۲۲۱۱ را به دست می دهد.

11.3.a

رویکرد نوار لاستیکی چند ضلعی را مجبور می کند تا رئوسی در هر خمیدگی دیواره سلولی داشته باشد. یعنی محل رئوس توسط ساختار دیواره های داخلی و خارجی ثابت می شود. از آنجایی که رئوس با خطوط مستقیم به هم متصل می شوند، این چند ضلعی حداقل محیطی را برای هر پیکربندی دیوار ایجاد می کند.

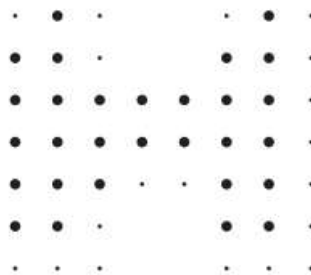
11.3.b

اگر گوشه ای از سلول در مرکز یک پیکسل در مرز باشد و سلول به گونه ای باشد که نوار لاستیکی در گوشه مقابل سفت شود، وضعیتی ماند شکل زیر خواهیم داشت. با فرض اینکه اندازه سلول $d \times d$ باشد، حداکثر اختلاف بین پیکسل و مرز در آن سلول $\sqrt{2}d$ است. اگر سلول ها روی پیکسل ها متمرکز شوند، حداکثر اختلاف $\sqrt{2}d/2$ است.



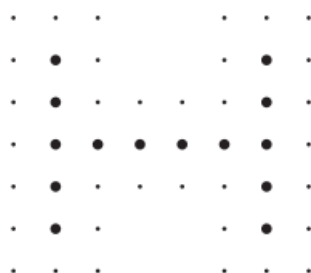
10.10.a

نتیجه مانند شکل زیر خواهد شد.



10.10.b

بعد از دو مرحله نتیجه زیر را خواهیم گرفت.



10.11.a

تعداد نمادها در تفاوت اول برابر است با تعداد قسمت های اولیه در مرز، بنابراین ترتیب شکل ۱۲ است.

10.11.b

با شروع از گوشه بالا سمت چپ نتایج به شرح زیر خواهند بود:

Chain code: 000332123211 , difference: 300303311330 , shape number: 003033113303

10.19.a

تصویر داده شده مطابق زیر است:

01010

10101

01010

10101

01010

فرض میکنیم $z_1 = 0$ و $z_2 = 1$. از آنجا که تنها دو سطح شدت وجود دارد، ماتریس G از مرتبه 2×2 است. g_{11} یعنی عنصراولین سطر و ستون از G تعداد پیکسل های با ارزش ۰ است که یک پیکسل در سمت راست ۰ قرار دارد. با بررسی، $g_{11} = 0$. به طور مشابه، $g_{12} = 10$ ، $g_{21} = 10$ و $g_{22} = 0$. تعداد کل پیکسل هایی که گزاره P را برآورده می کنند ۲۰ است، بنابراین ماتریس co-occurrence نرمایلازد شده مثل زیر است.

$$G = [0 \quad 1/2, \quad 1/2 \quad 0]$$

10.19b

در این مورد، g_{11} تعداد دو پیکسل ۰ در سمت راست یک پیکسل است. با بررسی شکل داده شده، $g_{11} = 8$ است. به طور مشابه، $g_{12} = 0$ ، $g_{21} = 0$ و $g_{22} = 7$ به دست می آید. تعداد پیکسل هایی که P را برآورده می کنند ۱۵ است، بنابراین ماتریس co-occurrence نرمایلازد شده مانند زیر است.

$$G = [8/15 \quad 0, \quad 0 \quad 7/15]$$

11.26

این مشکل چهار بخش عمده دارد. (۱) تشخیص همه بطری ها در یک تصویر؛ (۲) پیدا کردن بالای هر بطری. (۳) پیدا کردن گردن و شانه هر بطری. و (۴) تعیین سطح مایع در ناحیه بین گردن و شانه. (۱) پیدا کردن بطری ها : پس زمینه در تصویر نمونه بسیار تیره تر از بطری ها است. ما فرض می کنیم که این در همه تصاویر صادق است. سپس، یک راه ساده برای یافتن بطری های منفرد، یافتن نوارهای سیاه عمودی در تصویر است که عرض آن ها با میانگین جدایی بین بطری ها تعیین می شود، عددی که به راحتی از تصاویری که نشان دهنده تنظیمات واقعی در حین کار است، قابل محاسبه است. ما می توانیم این نوارها را به روش های مختلفی پیدا کنیم. یکی از راه ها صاف کردن تصویر برای کاهش اثرات نویز است (فرض می کنیم که مثلاً یک ماسک میانگین 3×3 یا 5×5 کافی است). سپس، یک خط اسکن افقی را از وسط

تصویر عبور می دهیم. مقادیر کم در خط اسکن با پس زمینه سیاه یا تقریباً سیاه مطابقت دارد. هر بطری افزایش و کاهش قابل توجهی از سطح شدت در خط اسکن برای عرض بطری ایجاد می کند. بطری هایی که به طور کامل در میدان دید دوربین قرار دارند، دارای عرض متوسط از پیش تعیین شده خواهند بود. بطری هایی که فقط تا حدی در میدان دید قرار دارند، نمایه های باریک تری خواهند داشت و می توان آنها را از تجزیه و تحلیل بیشتر حذف کرد.

(۲) یافتن قسمت بالای هر بطری : وقتی محل هر بطری (کامل یا تقریباً کامل) مشخص شد، دوباره می توانیم از کنتراست بین بطری ها و پس زمینه برای پیدا کردن قسمت بالای بطری استفاده کنیم. یکی از روش های ممکن، محاسبه یک تصویر گرادیان (فقط به لبه های افقی حساس) و جستجوی یک خط افقی در نزدیکی بالای تصویر گرادیان است. یک روش ساده تر این است که یک خط اسکن عمودی را از وسط مکان های یافت شده در مرحله قبل اجرا کنیم. اولین تغییر عمده در سطح خاکستری (از بالای تصویر) در خط اسکن، نشانه خوبی از محل بالای یک بطری است.

(۳) یافتن گردن و شانه یک بطری. در غیاب اطلاعات دیگر، ما فرض می کنیم که همه بطری ها همان اندازه هستند، همانطور که در تصویر نمونه نشان داده شده است. سپس، زمانی که در جایی که بالای یک بطری است، محل گردن و شانه در فاصله ثابتی از بالای بطری است.

(۴) تعیین سطح مایع: ناحیه ای که توسط پایین گردن و بالای شانه تعریف شده است تنها ناحیه ای است که باید بررسی شود تا سطح پر شدن قابل قبول در مقابل غیرقابل قبول در یک بطری مشخص شود. در واقع، همانطور که در تصویر نمونه نشان داده شده است، ناحیه ای از بطری که خالی از مایع است در یک تصویر کاملاً روشن به نظر می رسد، بنابراین گزینه های مختلفی داریم. ما می توانیم یک خط اسکن عمودی را دوباره اجرا کنیم، اما بطری ها دارای مناطق بازتابی هستند که می تواند این رویکرد را اشتباه بگیرد. یک روش این است که ناحیه ای را که در یک مستطیل قرار دارد که توسط پایین گردن، شانه و کناره های بطری مشخص شده است، در آستانه قرار دهیم. سپس تعداد پیکسل های سفید بالای نقطه وسط این مستطیل را می شماریم. اگر این عدد بیشتر از یک مقدار از پیش تعیین شده باشد، می دانیم که مایع کافی از دست رفته است و اعلام می کنیم بطری به درستی پر نشده است. یک تکنیک کمی پیچیده تر، یافتن سطح مایع است. این شامل جستجوی یک لبه افقی در ناحیه داخل بطری است که توسط دو طرف بطری، پایین گردن، و خطی که از وسط شانه و پایین گردن می گذرد، مشخص می شود. یک رویکرد پیوندی گرادیان/لبه، همانطور که در فصل ۱۰ توضیح داده شد، مناسب خواهد بود.

پیوست ۱: روند اجرای برنامه

پوشه تصاویر در فایل کدها قرار داده شده است و با انجام `set path` کدها اجرا خواهند شد. بخش‌های مختلف هر کد با %% از هم تفکیک شده‌اند. در صورت نیاز توضیحاتی در خود کد نوشته شده است.

- [1] R. C. Gonzalez and R. E. Woods, "Digital image processing." Prentice hall Upper Saddle River, NJ, 2002.
- [2] MATLAB help