

به نام خدا

گزارش تمرین ۸

نیما کمبرانی ۹۸۵۲۱۴۲۳

مبانی بینایی کامپیوتر

استاد: دکتر محمدی

سوال ۱)

(الف)

برای الگوریتم **split and merge** نیاز داریم که میزان همگن بودن یک ناحیه را بدست آوریم. در این حالت برای بدست آوردن همگن بودن ناحیه با توجه به توصیه کتاب از واریانس میانگین های هر یک ۴ زیر ناحیه استفاده می کنیم. با توجه به اینکه سطح آستانه برابر ۳ است اگر در یک ناحیه مقدار بدست آمده بزرگتر از ۳ بود ناهمگن و در غیر این صورت همگن است.

۱. برای اجرای الگوریتم در ابتدا کل تصویر را بعنوان یک ناحیه در نظر میگیریم و این از حالت ادامه می دهیم.
 ۲. ناحیه هایی که شرط همگن بودن را نداشته باشند، به ۴ قسمت بخش تقسیم می کنیم. هر ۴ ناحیه که از یک والد تولید شده باشند و شرط همگن بودن را داشته باشند، باهم ادغام می کنیم.
- با شروع از کل تصویر بعنوان یک ناحیه شرط همگن بودن را بررسی میکنیم.

برای کل تصویر میزان همگنی برابر ۳,۲ (واریانس میانگین ۴ بخش) است که از آستانه ۳ بزرگتر است پس به ۴ بخش تقسیم می شود.

برای بخش ۰ میزان همگنی برابر ۰,۳ بدست می آید در نتیجه این بخش همگن است و برای بخش دوم میزان همگن بودن ۱,۸ و برای بخش ۳ برابر ۰,۰۳ و بخش ۴ میزان همگن بودن برابر ۲,۳ بدست می آید که در نتیجه تمامی بخش ها همگن هستند. پس ناحیه بندی بصورت ۴ بخش بدست می آید.

۰	۱
۲	۳

۳. اگر دو ناحیه مجاور شرط همگنی را داشتند، باهم ادغام میکنیم.

میانگین هر یک از نواحی ۰ تا ۳ از چپ به راست برابر است با (۵,۵), (۴,۸۱۲۵), (۰,۸۱۲۵), (۳,۶۲۵)

با توجه به میانگین نواحی ۰ و ۱ و نواحی ۳ و ۲ مجاور هستند و هم شرط همگن بودن را دارند در نتیجه می توانند با یک دیگر ادغام شوند.

۰	۱
۲	۳

شکل نهایی ادغام نواحی بصورت جدول بالا خواهد بود.

(ب)

در گسترش ناحیه با شروع از یک نقطه تا جایی که همچنان اندازه نقاط در فاصله نزدیک به نقطه اولیه یا نقطه مجاور باشد گسترش را ادامه می‌دهیم. در **split and merge** با شروع از تعدادی ناحیه با توجه به تابع سنجش همگن بودن نواحی تعدادی از این نواحی را ادغام و تعدادی دیگر را به نواحی کوچکتر تقسیم می‌کنیم. در روش گسترش ناحیه، یه ناحیه بصورت پیکسل به پیکسل گسترش می‌یابد و در ادامه ناحیه‌های ساخته شده یا موجود تقسیم نمی‌شوند. اما در روش دوم گسترش نواحی به صورت ادغام چند ناحیه با یک دیگر است و ممکن است به صورت پیکسل به پیکسل بررسی انجام نشود.

در نتیجه در تصاویری مانند صفحه شطرنج با استفاده از گسترش ناحیه هر خانه شطرنج بصورت جدا ناحیه بندی می‌شود اما در روش دوم با توجه به برابر بودن ۴ بخش صفحه شطرنج کل صفحه بصورت یک ناحیه در نظر گرفته میشود.

سوال ۲)

(الف)

در اجرای اول عملگر باز نقاط و جزییات کوچک در تصویر حذف می‌شوند اما در بار دوم تغییر در نتیجه ایجاد نمی‌شود و نتیجه به بار اول اجرا برابر خواهد بود.

(ب)

۱. افزودن **padding** به اطراف تصویر اولیه با تکرار اندازه پیکسل آخر

۱	۱			۱	۱
۱	۱			۱	۱
۱	۱			۱	۱
۱	۱	۱	۱	۱	۱
۱	۱	۱	۱	۱	۱
۱	۱			۱	۱
۱	۱			۱	۱
۱	۱			۱	۱

۲. انجام عملیات سایش با استفاده از فیلتر داده شده

۱	۱				۱
۱	۱				۱
۱	۱				۱
۱	۱			۱	۱
۱	۱	۱	۱	۱	۱
۱	۱				۱
۱	۱				۱
۱	۱				۱

۳. معکوس کردن فیلتر و انجام عملیات گسترش

۱	۱			۱	۱
۱	۱			۱	۱
۱	۱			۱	۱
۱	۱	۱	۱	۱	۱
۱	۱	۱	۱	۱	۱
۱	۱			۱	۱
۱	۱			۱	۱
۱	۱			۱	۱

۴. اعمال سایش برای دومین مرتبه

۱	۱				۱
۱	۱				۱
۱	۱				۱
۱	۱			۱	۱

۱	۱	۱	۱	۱	۱
۱	۱				۱
۱	۱				۱
۱	۱				۱

۵. انجام عملیات گسترش برای مرتبه دوم

۱	۱			۱	۱
۱	۱			۱	۱
۱	۱			۱	۱
۱	۱	۱	۱	۱	۱
۱	۱	۱	۱	۱	۱
۱	۱			۱	۱
۱	۱			۱	۱
۱	۱			۱	۱

در این مسئله با استفاده از فیلتر داده شده عملیات باز تغییری در شکل ایجاد نکرد و در پایان با حالت اولیه خود باقی ماند.

سوال (۳)

(الف)

روش otsu با توجه به اینکه بصورت اتوماتیک و با استفاده از واریانس سطح آستانه را مشخص می کند، نسبت به استفاده از میانگین و میانه برای تعیین سطح آستانه بهتر عمل می کند. همچنین باتوجه به اینکه تنها یکبار سطح آستانه برای کل تصویر محاسبه می شود، سرعت مطلوبی دارد و درزمان کم پایان می یابد. در حالتی که نور پردازی تصویر به درستی انجام نشده باشد و میزان سایه در بخش های مختلف تصویر متفاوت باشد، بدلیل اینکه سطح آستانه بدست آمده برای تمامی پیکسل ها یکسان است ، بخش های تاریک تر بطور کامل سیاه و بخش های روشن تر کاملاً سفید می شوند و جزئیات مهم از دست می روند.

روش adaptive threshold باتوجه به اینکه نیاز به محاسبه آستانه برای هریک از خانه ها به صورت جدا است، از لحاظ محاسباتی پیچیدگی بیشتری دارد و زمان بیشتری می برد. باتوجه به اینکه برای هر پیکسل بصورت محلی آستانه گذاری انجام می پذیرد، تاثیر سایه و نورپردازی در آن تاحد زیادی از بین میرود و در خروجی جزئیات هر بخش از تصویر باقی می ماند.

(ب)

اجرای عملیات adaptive threshold دارای ۳ مرحله است. ابتدا با استفاده از کانولوشن با فیلتر گاوسی میانگین نقاط در یک همسایگی را برای هریک از نقاط تصویر بدست می‌آوریم. در مرحله دوم میانگین بدست آمده را از تصویر اصلی کم می‌کنیم تا اختلاف هر تصویر با اطرافش را بدست آوریم. در مرحله آخر با اعمال آستانه گذاری با مقدار C بر روی تصویر تفاوت بدست آمده در مرحله دوم، خروجی نهایی را بدست می‌آوریم.

ورودی های تابع adaptiveThreshold بصورت زیر هستند:

Src: تصویر ۸ بیتی ورودی که عملیات آستانه گذاری روی آن انجام می‌پذیرد.

Max_value: مقداری که در خروجی نهایی به پیکسل های بزرگتر از آستانه داده می شود.

AdaptiveMethod: روشی است که برای بدست آوردن میانگین استفاده می‌شود و میتواند بصورت وزندار گاوسی یا میانگین معمولی باشد.

blockSize: اندازه پنجره اطراف پیکسل که برای بدست آوردن میانگین در نظر گرفته میشود.

thresholdType: نحوه آستانه گذاری پیکسل ها را مشخص می کند که میتواند سفید یا سیاه شدن پس زمینه را تنظیم کند.

C: مقدار ثابتی است که آستانه گذاری پس کم کردن میانگین از اندازه پیکسل با توجه به آن انجام می شود.

سوال (۴)

در تصویر royan با توجه به وجود سایه و نورپردازی غیر یکسان در تصویر با استفاده از otsu برای آستانه گذاری بخش زیادی از تصویر کاملاً سیاه و بسیاری جزئیات حذف میشوند. با استفاده از adaptive threshold تاثیر سایه در حذف جزئیات کمتر می شود. در مرحله بعد با استفاده از عملگر گسترش بر روی تصاویر با توجه به اینکه نوشته ها سیاه هستند و پس زمینه سفید باعث نازک شدن نوشته ها می شود اما برای حروفی مانند "ه" که وسط آن ها خالی است باعث بهتر دیده شدن آن می‌شود. با توجه به اینکه اتسو بسیاری از متن را بطور کامل سفید میکند باعث از بین رفتن آن میشود اما در روش انطباقی متن هایی که در سایه قرار دارند نیز در خروجی نمایش داده می شوند و توسط الگوریتم OCR قابل تشخیص هستند.

عنصر ساختاری استفاده شده در این سوال بصورت زیر است:

۰	۱	۰
۱	۱	۱
۰	۱	۰

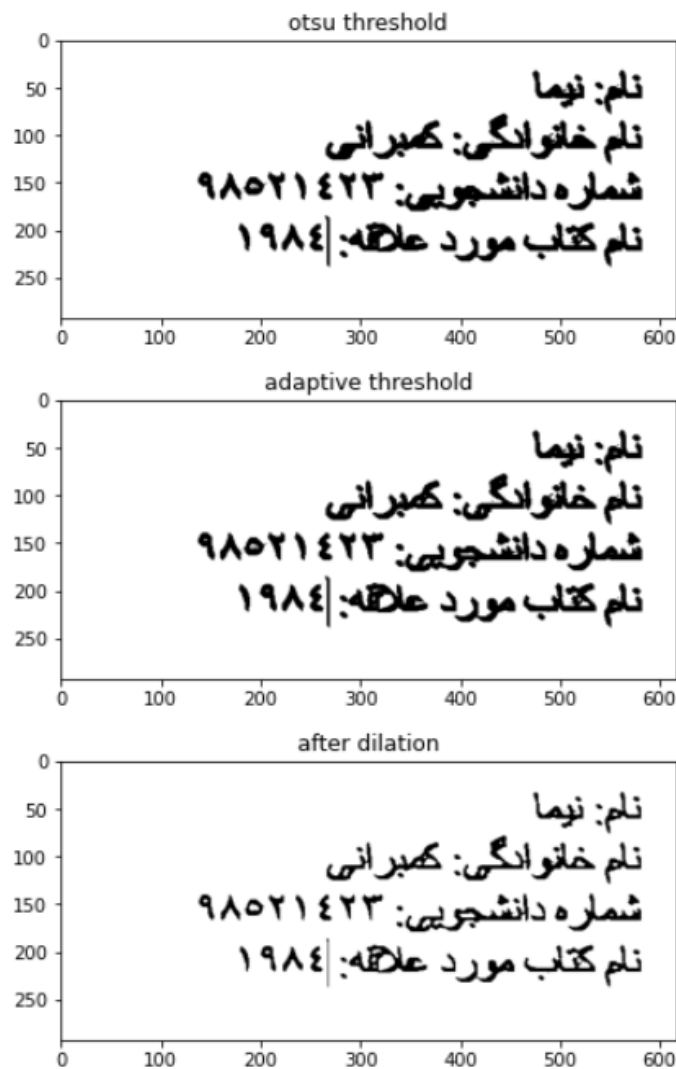
در شکل یک تصویر خروجی پس از اعمال آستانه گذاری از دو روش otsu و adaptive threshold را نشان می دهد.



شکل ۱. تصویر بالا خروجی آستانه گذاری otsu و تصویر پایین خروجی آستانه گذاری Adaptive را نشان می دهد.

(ب)

با توجه به اینکه تصویر ورودی مشکل نور پردازی ندارد و تفاوت رنگ پیکسل ها زیاد است خروجی دو الگوریتم otsu و adaptive تفاوت چندانی نخواهد داشت. اما در این حالت استفاده از گسترش باعث بهبود جزئیات حروف می شود. در شکل ۲ خروجی آستانه گذاری و همچنین عملگر گسترش نمایش داده شده است. در این حالت الگوریتم ocr متن را با دقت زیادی تشخیص می دهد.



شکل ۲. خروجی آستانه گذاری با دو الگوریتم و اعمال عملگر گسترش بر روی خروجی الگوریتم *adaptive*