

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

تمرین پنجم درس پردازش تصویر

دکتر رحمتی

غلامرضا دار ۴۰۰۱۳۱۰۱۸

بهار ۱۴۰۱

فهرست مطالب

۳	سوال ۱)
۴	سوال ۲)
۱۴.	سوال ۳)
۲۴.	سوال ۴).

سوال ۱)

به صورت pdf مجزا در فایل زیپ موجود است.

سوال (۲)

در این سوال قصد داریم کاربردهای عملیات مورفولوژی مختلف بر روی تصاویر سطح خاکستری را بررسی کنیم. روابط محاسبه *Smoothig* و *Gradient*, *Second Derivative* درس برداشته شده اند.

2nd Derivative using Dilation and Erosion (A

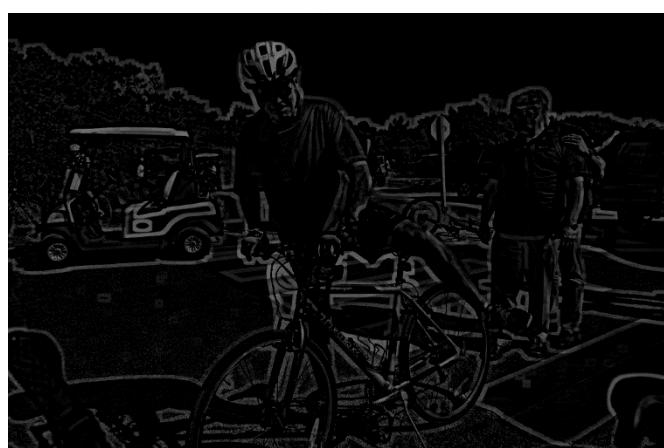
1 iteration



2 iterations

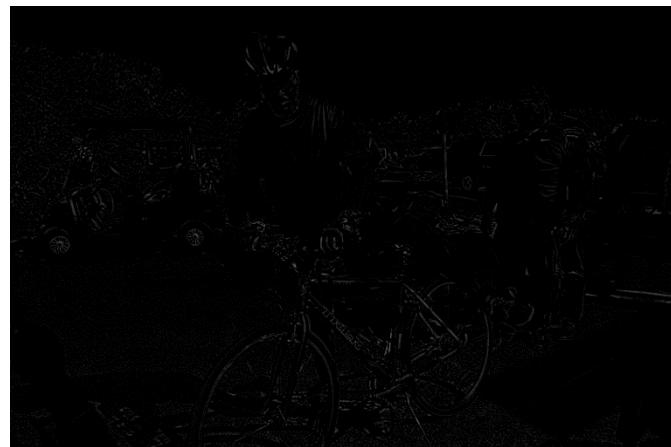


3 iterations

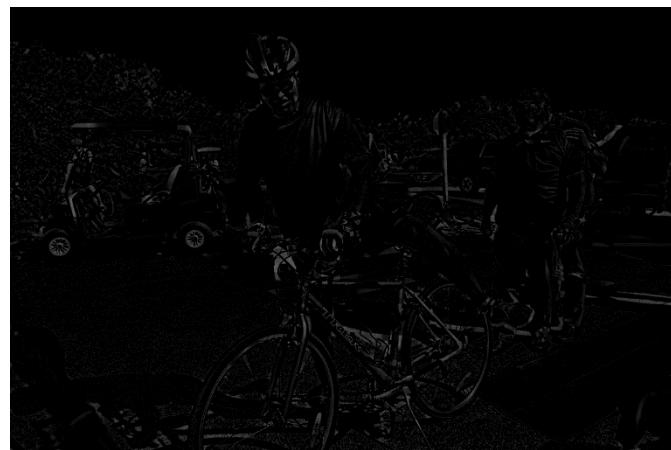


2nd Derivative using Opening and Closing (B

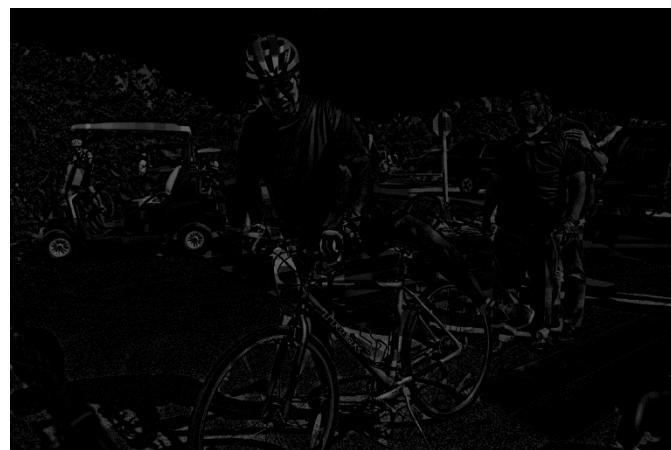
1 iteration



2 iterations



3 iterations



2nd Derivative using Dilation, Erosion, Opening and Closing (C)

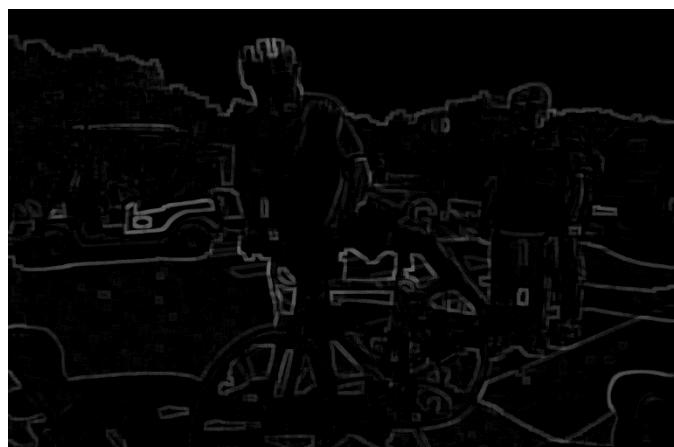
1 iteration



2 iterations



3 iterations



Gradient using Dilation and Erosion (D)

1 iteration



2 iterations



3 iterations



Gradient using Opening and Closing (E)

1 iteration



2 iterations



3 iterations



Gradient using Dilation, Erosion, Opening and Closing (F

1 iteration



2 iterations



3 iterations



Smoothing using Dilation and Erosion (G)

1 iteration



2 iterations



3 iterations



Smoothing using Opening and Closing ($H \otimes H$)

1 iteration



2 iterations



3 iterations

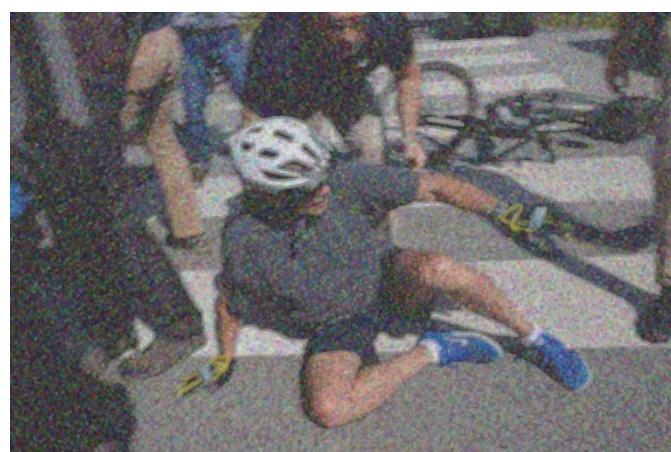


خروجی مراحل قبل به صورت رنگی

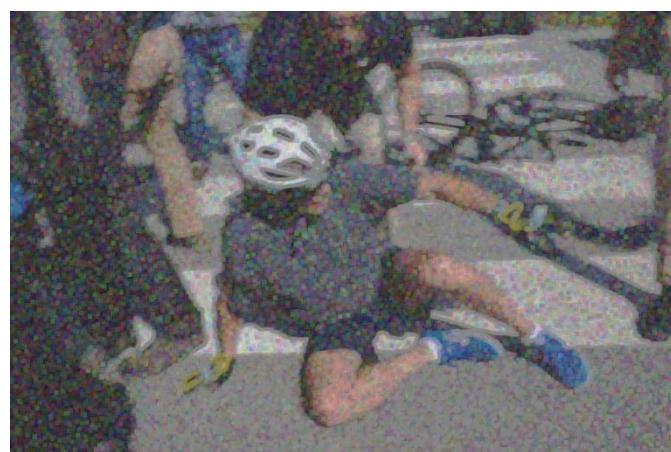
1 iteration



2 iterations



3 iterations



1 iteration



2 iterations

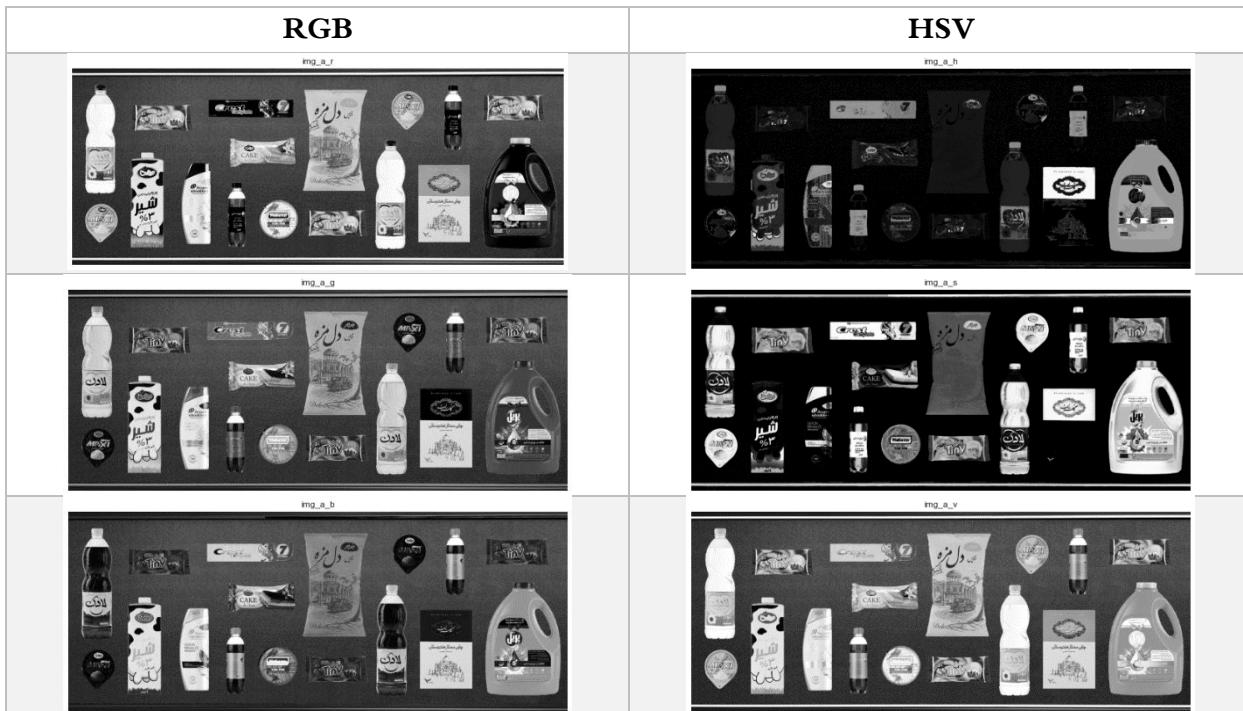


3 iterations

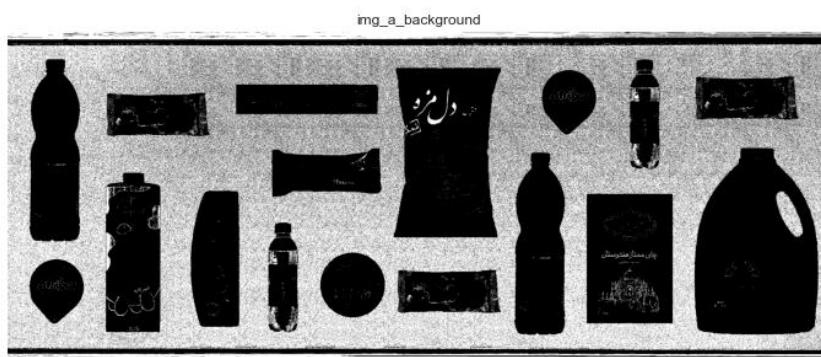


سوال (۳)

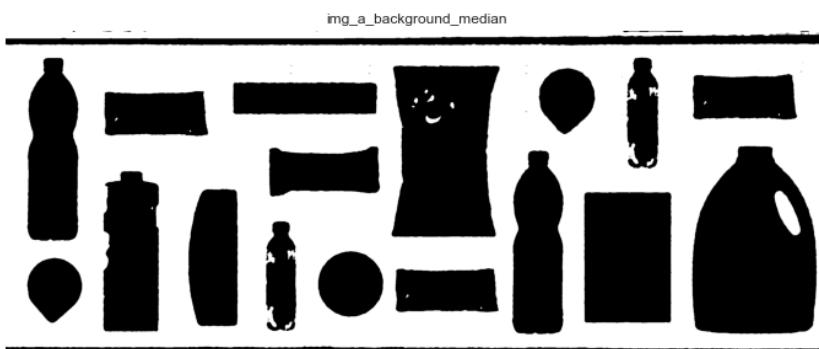
در این سوال قصد داریم با استفاده از ترکیب تکنیک‌های مختلف از جمله Morphology، Template Matching، Feature Matching، یک سبد خرید را بررسی کنیم و به سوالات مختلف در مورد ارزش سبد خرید پاسخ دهیم.
برای حل این سوال ابتدا از تکنیک‌های Morphology استفاده کردیم تا بتوانیم کالاهای را به خوبی از پس زمینه جدا کنیم.
برای این کار، ابتدا کanal‌های مختلف تصویر را مشاهده می‌کنیم.



با مقایسه دو فضای رنگی HSV و RGB متوجه می‌شویم که کanal V فضای رنگی HSV بهترین کanal برای جداسازی پس زمینه است. زیرا تقریباً تمام پیکسل‌های کالاهای از پس زمینه روشن‌تر هستند. به این ترتیب، با محدود کردن رنج مقادیر بین ۰ تا ۱۰۰ در این کanal، یک تخمین نسبتاً خوب از Background Extraction خواهیم داشت.



همین طور که مشاهده می کنید، این تصویر دارای مقدار زیادی نویز در پس زمینه است. برای کاهش این نویز از فیلتر Median Blur با اندازه کرنل ۱۱ استفاده می کنیم.



از آنجایی که احتمالا در همچین کاربردی، همواره مکان دوربین ثابت است، به صورت دستی نوار بالا و پایین تصویر را حذف کردیم (با کراپ کردن عمودی تصویر به اندازه ۵۰ پیکسل از هر جهت).



در مرحله بعد برای از بین بدن سوراخ های موجود در برخی از کالاهای، از Opening با اندازه کرنل ۲۷ استفاده می کنیم.



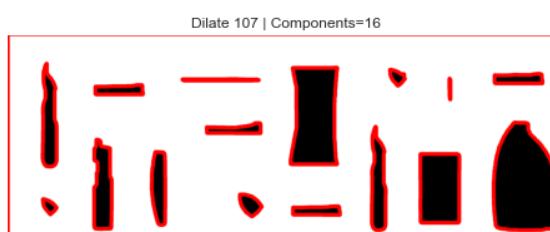
اگر با دقت به تصویر قبل نگاه کنید، یک سری نقطه مشکی رنگ در تصویر مشاهده می‌کنید. این نقاط را می‌توان به سادگی با استفاده از Closing از بین برد.



همچنین یک نسخه از تصویر که سوراخ موجود در دستگیره مایع ظرفشویی را ندارد نیز تولید می‌کنیم.



در این مرحله، تصویر برای شمارش کالاها آماده است. در این تمرین از تعدادی روش مختلف استفاده می‌کنیم و نتایج این روش‌ها را با یکدیگر مقایسه می‌کنیم. روش اولی که بررسی کردیم، کاهش اندازه کالاها به کمک عمل Dilate بود. به این ترتیب در هر مرحله با اعمال Dilate، اندازه کالاها کوچک‌تر می‌شوند. با اجرای این عمل به تعداد زیاد، کالاها به ترتیب اندازه، یکی پس از دیگری حذف می‌شوند. در ادامه با شمارش تعداد کالاهای باقی‌مانده، می‌توانیم مشخص کنیم که آیا نیازی به ادامه روند کاهش اندازه است یا خیر. برای شمارش کالاهای باقی‌مانده از Contour‌ها استفاده می‌کنیم.



Dilate 112 | Components=15



Dilate 150 | Components=12



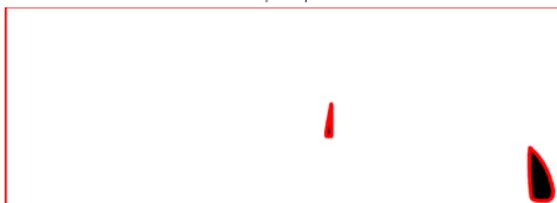
Dilate 182 | Components=4



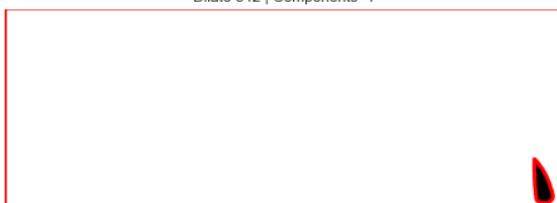
Dilate 201 | Components=3



Dilate 315 | Components=2



Dilate 342 | Components=1

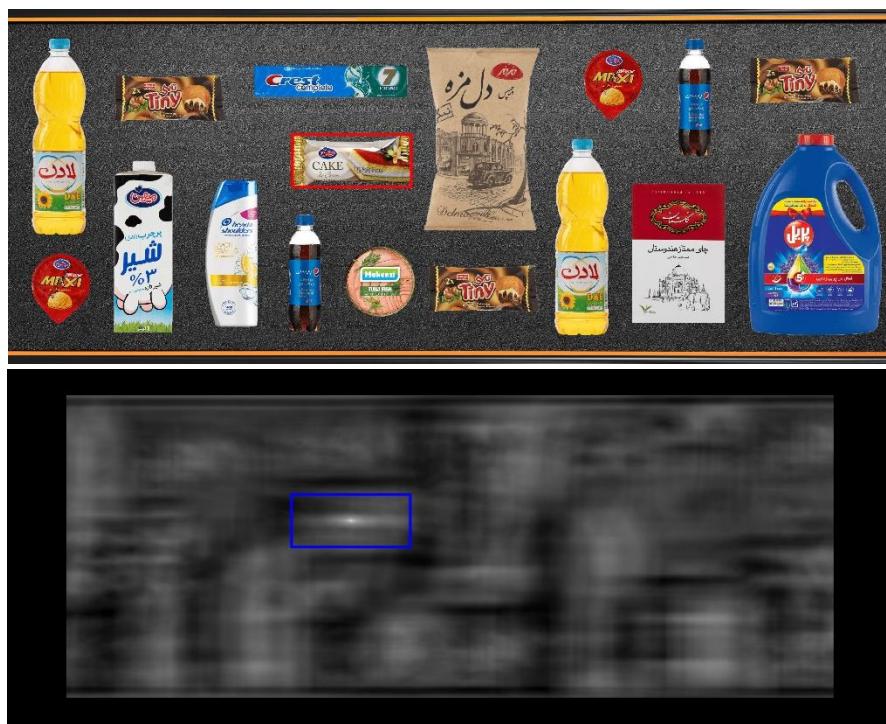


در مرحله بعد می‌توانیم تعداد مراحل لازم برای حذف شدن هر کالا را یادداشت کنیم و در مرحله تست، با توجه به اینکه تعداد کالاهای در چه مرحله‌ای تغییر می‌کند، می‌توانیم تشخیص دهیم که چه کالاهایی در سبد موجود بوده‌اند. این روش، روش بسیار جالبی نیست و برای بخش دوم سوال نیز کار نخواهد کرد. به همین دلیل از روش‌های دیگری برای حل این سوال استفاده خواهیم کرد.

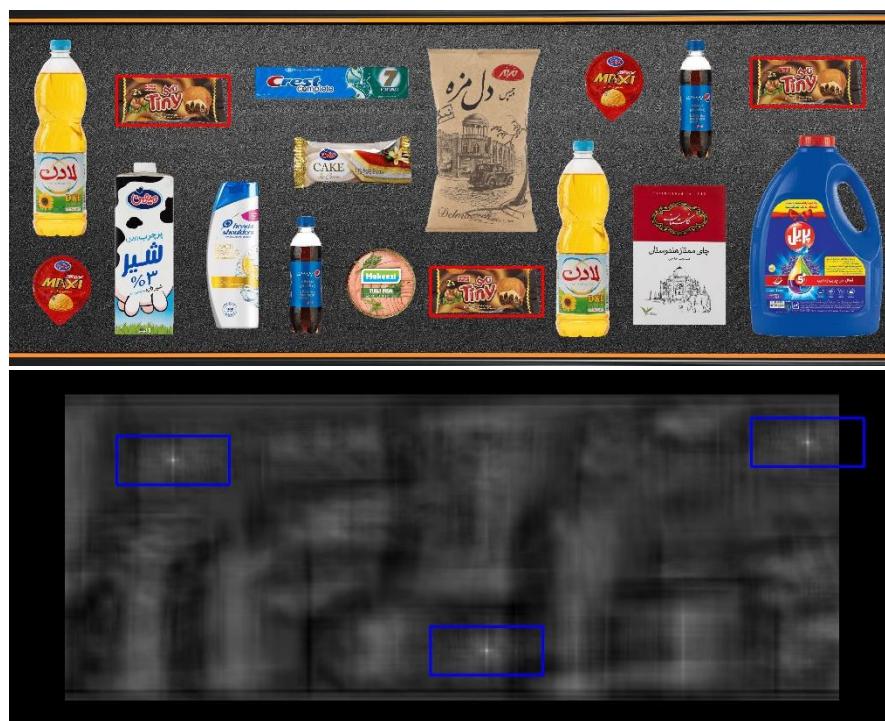
روش دومی که استفاده خواهیم کرد، روش Template matching است. برای این روش نیاز است ابتدا تصاویری از هر کالا را به طور جداگانه داشته باشیم. سپس تصویر هر کالا را بر روی تصویر اصلی (سبد خرید) حرکت می‌دهیم و میزان Correlation این دو تصویر را در این حالت بررسی می‌کنیم.

پس از اعمال Template Matching، روش‌ترین نقطه تصویر حاصل، مکان کالای مدنظر ما خواهد بود. مانند تصویر دوم، ناحیه‌ای به اندازه کالا و به مرکزیت نقطه روش‌یافته شده، در نظر می‌گیریم. این ناحیه را از تصویر اصلی حذف می‌کنیم و دوباره Template Matching را اجرا می‌کنیم. با این کار اگر تعداد بیشتر از یک کالا وجود داشته باشد نیز تشخیص داده می‌شود. در ادامه تعدادی از کالاهای یافت شده را مشاهده می‌کنیم.

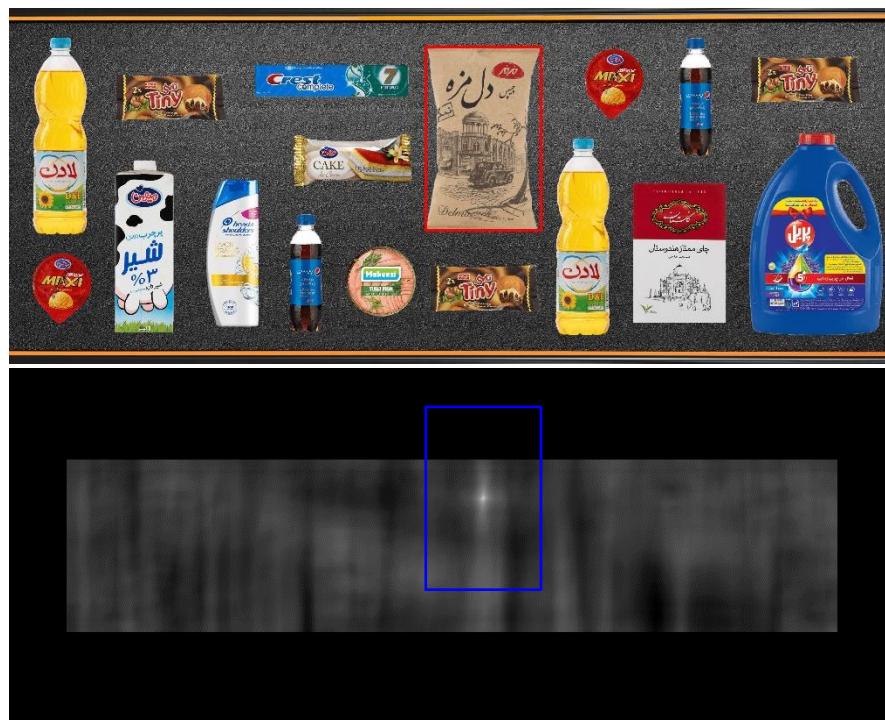
یافتن کیک میهن



یافتن کیک تانی



یافتن چیپس دل مزه مزمز



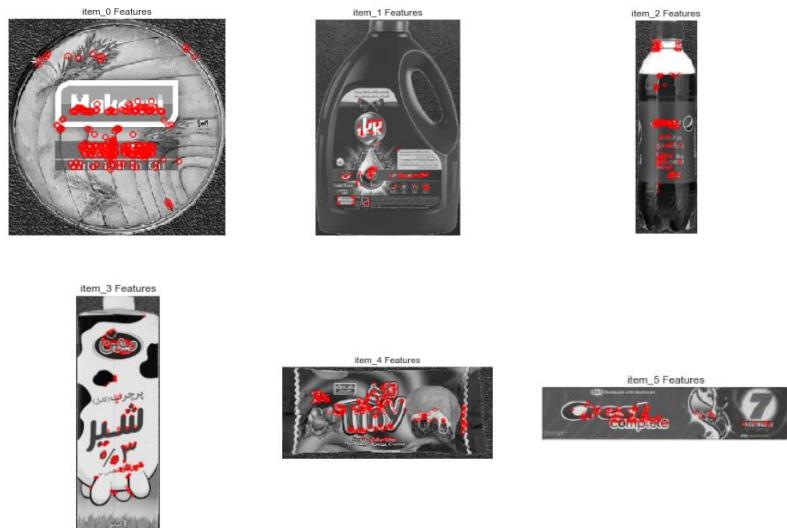
اما همان طور که انتظار می رفت، این روش برای تصویر دوم عملکرد خوبی نخواست داشت. روش Template Matching نمی تواند تصاویر با چرخش متفاوت یا مقیاس متفاوت را تشخیص دهد و این باعث می شود راه حل مناسبی برای مسئله ما نباشد.



بنابراین به راه حل سوم پیشنهادی یا Feature Matching می رسیم. این روش در تصویر سبد و تصویر کالا دنبال ویژگی های مشابه می گردد. این ویژگی ها را نظری می کند و با محاسبه یک تبدیل هندسی مناسب، کالا را در موقعیت خودش در تصویر سبد قرار می دهد. این روش نسبت به تغییر مقیاس و جهت مقاوم است و حتی اگر بخشی از کالا توسط کالای دیگری پنهان شده باشد باز هم تعدادی ویژگی مشترک بین کالا و بخش نمایان کالا در سبد پیدا می شود. اینها این روش را برای حل بخش اول و دوم این مسئله بسیار مناسب می کنند. در تصاویر زیر، می توانید نحوه استخراج ویژگی برای این روش را مشاهده کنید. همچنین لازم به ذکر است که از پیاده سازی ORB برای Feature Matching استفاده کردیم. ستون سمت راست پس از اعمال مقداری Gaussian Blur به تصاویر ویژگی ها را استخراج کرده است.



همین کار را برای تصویر کالاهای نیز انجام می‌دهیم.

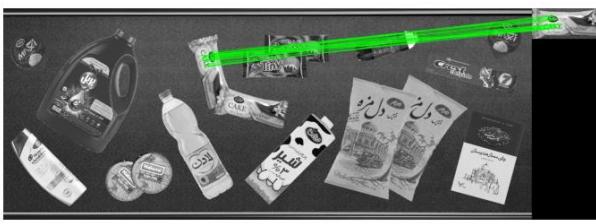


در مرحله بعد این ویژگی‌ها را با هم مچ می‌کنیم. حاصل مچ کردن تعدادی از کالاهای را مشاهده می‌کنید. همچنین برای تشخیص تعداد بیشتر از یک کالا، مانند روش Template Matching عمل می‌کنیم و پس از هر تشخیص آن کالا را از تصویر سبد حذف می‌کنیم.

نتایج Feature Matching برای سبد ۱



نتايج Feature Matching برای سبد ۲



در ادامه نتیجه شمارش دو سبد توسط این روش را مشاهده می‌کنید.



Basket 1 Summary	
<hr/>	
Tuna	: x 1
Peril	: x 1
Pepsi	: x 2
Milk	: x 1
Tiny	: x 3
Crest	: x 1
Cake	: x 1
Oil	: x 2
Maxi	: x 2
Shampoo	: x 1
Tea	: x 1
Chips	: x 1

Basket 2 Summary	
<hr/>	
Tuna	: x 2
Peril	: x 1
Pepsi	: x 1
Milk	: x 1
Tiny	: x 2
Crest	: x 1
Cake	: x 2
Oil	: x 1
Maxi	: x 2
Shampoo	: x 1
Tea	: x 1
Chips	: x 2

با داشتن تعداد کالاهای مختلف در هر سبد و همچنین دانستن قیمت آنها، میتوانیم به تمام سوالات پاسخ دهیم.

۱۷) کالا

۳ (B)

$۳۰۰۰۰ * ۲ + ۱۰۸۰۰۰ * ۲ + ۶۵۰۰ * ۲$ (C)

۴ (D)

E) لیست تعداد کالاهای موجود در صفحه قبل موجود است. آنها را نظیر به نظیر در قسمت هر کالا ضرب میکنیم.

۱۷) کالا

۲ (G)

$۶۰۰۰ * ۲$ (H)

I) لیست تعداد کالاهای قابل خوردن در صفحه قبل موجود است. آنها را نظیر به نظیر در قسمت هر کالا ضرب میکنیم.

J) لیست تعداد کالاهای موجود در صفحه قبل موجود است. آنها را نظیر به نظیر در قسمت هر کالا ضرب میکنیم.

سوال (۴)

(A) وقتی تصویر را به فضای YCbCr می‌بریم، مولفه روشنایی تصویر از مولفه‌های رنگی آن جدا می‌شوند. آزمایش‌ها نشان داده اند که مغز انسان، به سطح روشنایی یک تصویر، به نسبت رنگ، اهمیت بسیار بیشتری می‌دهد. این دانش ایده این کار را بوجود می‌آورد که مولفه رنگی تصویر را با میزان فشرده سازی بیشتری نسبت به مولفه روشنایی تصویر ذخیره کنیم.

با این کار، بدون تغییر زیاد در کیفیت تصویر، حجم مورد نیاز برای ذخیره سازی تصویر کاهش می‌یابد.

(B) این اتفاق در واقع یکی از مشکلات استفاده از اندازه کرnel زوج در عملیات Opening است. وقتی از کرnel با اندازه زوج استفاده کنیم، پیکسل مرکزی نخواهیم داشت و با هر بار اعمال این عملیات، پیکسل مرکزی تخمین زده می‌شود و جابجا می‌شود. بنابراین می‌توان برای شیفت دادن تصویر به سمت راست، به تعداد دلخواه، عملیات Opening را با یک کرnel اندازه زوج مانند ۲ در ۲، بر روی تصویر اعمال کرد.

(C) این دو عملیات Idempotent هستند. اعمال این دو عملیات بیش از یک بار نتیجه ای به همراه نمی‌آورد. البته در عمل این گونه نیست و در صورت استفاده از کرnel با اندازه زوج همانظور که در بخش B توضیح داده شد، تصویر شیفت می‌خورد.

(D) بله. عملیات مختلف مانند Erosion, Dilation, Opening, Closing, Thinning و Thickening توسط عملیات Hit and Miss و عملیات مجموعه‌ها قابل پیاده سازی هستند. میتوان از کرnel‌های زیر برای استخراج ویژگی‌های مختلف استفاده کرد و از ترکیب آنها با یکدیگر برای تولید Erosion استفاده کرد. کرnel شماره ۱ میتواند نقاط ایزوله را شناسایی کند. کرnel دو و چرخش‌های مختلفش میتواند انتهای یک مسیر در یک Skeleton را شناسایی کند و کرnel‌های ۳ و چرخش‌های آنها، میتوانند حالت سه راه را شناسایی کنند. تعداد بسیار زیادی از این نوع کرnel‌ها وجود دارند که میتوانند ویژگی‌های مختلفی را شناسایی کنند.

1)	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	0	0	0	0	1	0	0	0	0	2)	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td></td><td></td><td></td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>				0	1	0	0	0	0	3a)	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>1</td><td></td><td></td></tr> <tr><td>1</td><td></td><td></td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1			1			1	1	1	3b)	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>1</td><td></td><td></td></tr> <tr><td></td><td>1</td><td></td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1				1		1	1	1	3c)	<table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>*</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> <tr><td>*</td><td>1</td><td>*</td></tr> </table>	*	0	1	1	1	0	*	1	*
0	0	0																																																				
0	1	0																																																				
0	0	0																																																				
0	1	0																																																				
0	0	0																																																				
1																																																						
1																																																						
1	1	1																																																				
1																																																						
	1																																																					
1	1	1																																																				
*	0	1																																																				
1	1	0																																																				
*	1	*																																																				

به عنوان مثال برای erosion میتوان پیکسل‌هایی را انتخاب کرد که از چهار طرف توسط پیکسل‌های سفید محاصره شده اند و سایر پیکسل‌ها را نادیده گرفت. با این کار پیکسل‌های تنها از بین می‌روند و همچنین پیکسل‌های لبه نیز به دلیل اینکه کاملاً محاصره نشده اند از بین می‌روند و عمل Erosion شکل می‌گیرد. لینک [منبع](#)

(E) برای اینکار می‌توانیم از الگوریتم ارائه شده در این لینک استفاده کنیم. برای این منظور به صورت سطحی بر روی تصویر حرکت میکنیم تا به پیکسلی با مقدار ۱ برسیم. پس از یافتن همچنین پیکسلی، همسایه‌های آن که تا الان بررسی شده اند را مشاهده می‌کنیم. بر اساس این اطلاعات تصمیم می‌گیریم چه برچسبی به پیکسل فعلی بزنیم.

- If all four neighbors are 0, assign a new label to p , else
- if only one neighbor has $V=\{1\}$, assign its label to p , else
- if more than one of the neighbors have $V=\{1\}$, assign one of the labels to p and make a note of the equivalences.

پس از اتمام تصویر، هر پیکسل دارای یک لیبل که با تمام همسایه‌های چسبیده آن یکسان است خواهد شد.

لینک [منبع](#)