

پاسخ سوال اول

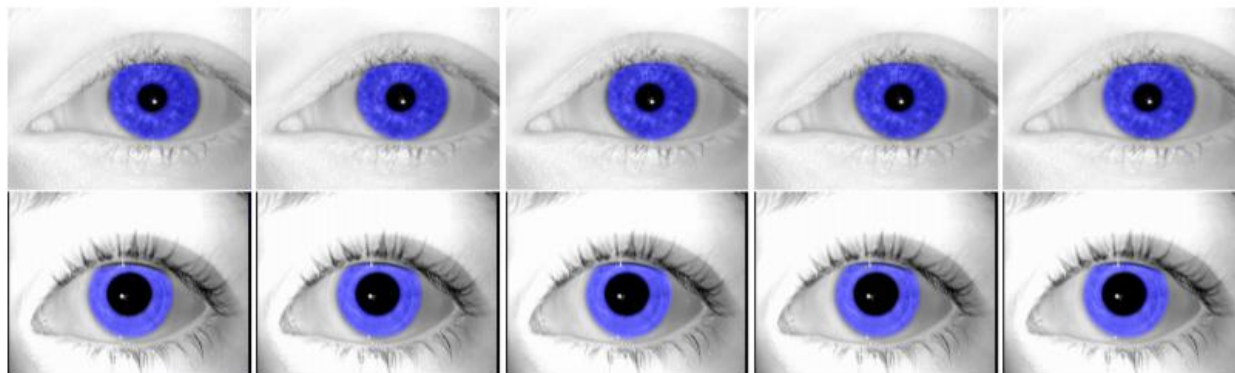
برخی از کاربرد های آن موارد زیر است:

1. تصویربرداری پزشکی از جمله حجم رندر تصاویر از سی تی اسکن و تصویر برداری رزونانس مغناطیسی.
2. تعیین محل تومور و دیگر بیماری ها
3. برنامه ریزی برای عمل جراحی
4. شبیه سازی جراحی مجازی
5. تشخیص عابر پیاده
6. تشخیص چهره
7. مکان یابی اشیاء در تصاویر ماهواره ای (جاده، جنگل، محصولات و غیره).
8. شناخت وظایف
9. تشخیص اثر انگشت
10. تشخیص عنبیه
11. سیستم های کنترل ترافیک
12. نظارت ویدئویی

حال به توضیح تشخیص عنبیه میپردازیم:

شناسایی عنبیه یک روش زیست سنجشی است که در آن با پردازش داده های به دست آمده از تصاویر عنبیه چشم افراد و مقایسه آنها با داده های بانک اطلاعاتی، می توان به هویت آنها پی برد. (عنبیه انسان ها مانند اثر انگشت منحصر به فرد است.

در این روش با استفاده از دوربین می توان به صورت خودکار از افرادی که در محدوده دید دوربین هستند را شناسایی کرد. این روش اگرچه در شرایطی غیر ایده آل نوری، فاصله چشم و متحرک بودن سوژه همچنان به دقت بسیار بالایی نرسیده اما هم اکنون در شماری از کشورهای جهان مورد استفاده قرار می گیرد .



که با توجه به طرح و بافت و رنگ این ناحیه بندی صورت میگیرد.

منبع:

[لینک اول](#)

[لینک دوم](#)

پاسخ سوال دوم

در این سوال از zero-padding استفاده کردیم.

سایش:

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

افزایش:

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1
1	1	1	0	1	1	1	1
0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	0
0	1	1	1	1	1	0	0
0	0	1	1	1	1	1	0

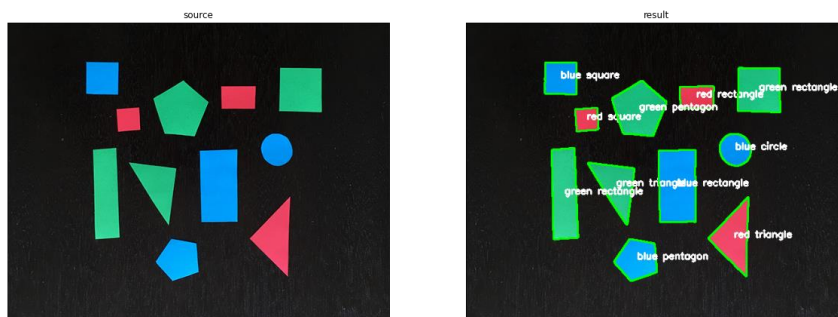
پاسخ سوال سوم

در این سوال ابتدا با توجه به لینک اول مراکز اشکال را بدست آوردیم، که همان ناحیه بندی است. ابتدا باید هموار میکنیم و یک حد آستانه اعمال میکنیم تا یک تصویر باینری بدست آوریم. `cv2.findContours` مجموعه ای از رئوس را برمی گرداند، از `imutil` استفاده میکنیم تا اطلاعات مربوط به هر کانتور را بدست آورده.

سپس در یک حلقه، در هر کانتور ابتدا مرکز آن شکل را به دست می آوریم سپس شکل و رنگ آن را بدست می آوریم با توجه به کلاس هایی که پیاده سازی کرده ایم.

برای انجام تشخیص شکل، من از contour approximation استفاده کردم. همانطور که از نام آن مشخص است، تقریب کانتور الگوریتمی برای کاهش تعداد نقاط منحنی با کاهش مجموعه نقاط است. برای انجام تقریب کانتور، ابتدا محیط کانتور را محاسبه کردم و به دنبال آن تقریب واقعی کانتور را ساختم. یک کانتور از لیستی از رئوس تشکیل شده است. تعداد ورودی های این لیست را بررسی کردم تا شکل یک شی را تعیین کنم.

به منظور برچسب گذاری و برچسب گذاری مناطق یک تصویر به عنوان یک رنگ خاص، من فاصله اقلیدسی را بین مجموعه ای از رنگ های شناخته شده و میانگین رنگی یک بخش تصویر خاص محاسبه کردم. برخلاف فضاها رنگی RGB و HSV، فاصله اقلیدسی بین رنگهای $L * a * b$ معنای واقعی ادراکی دارد. رنگ شناخته شده ای که فاصله اقلیدسی را به حداقل برساند به عنوان شناسایی رنگ انتخاب می شود.



پاسخ سوال چهارم

در این سوال افزایش و ساییش را پیاده سازی کردیم. تفاوت مشهودی ندارند خروجی های کد من و خروجی OpenCV.

ممکن کد opencv به دلیل سرعت و دقت بیشتر در `convolve` یا `padding` سرعت و دقت بیشتری داشته باشد.

در عملگر افزایش ابتدا `convolve` کردم تصویر را با اپراتور داده شده و نقاطی که صفر نبودند را مقدارشان را برابر یک قرار دادم.

در عملگر کاهش نیز ابتدا `convolve` کردم سپس با توجه به تعداد یک های موجود در `element` مقایسه کردم و به دو گروه تقسیم کردم اگر مقدارش از مجموع یک های `elemnt` بیشتر بود برابر یک قرار میدادم و در غیر اینصورت صفر که خروجی زیر حاصل شد.

