

الف) ابتدا تصویر مورد نظر را با استفاده از دستور `cv2.imread` می‌خوانیم. سپس با دستور `cv2.putText` شماره دانشجویی را به گوشه بالا چپ تصویر اضافه می‌کنیم. در انتها با دستور `cv2.cvtColor` حالت Grayscale تصویر را تولید می‌کنیم.



96101165



همچنین با استفاده از دستور `cv2.waitKey` کلیدهای ورودی را بررسی و در صورت فشرده شدن کلید `s` یا `e` برای ذخیره یا بستن تصاویر اقدام می‌کنیم.



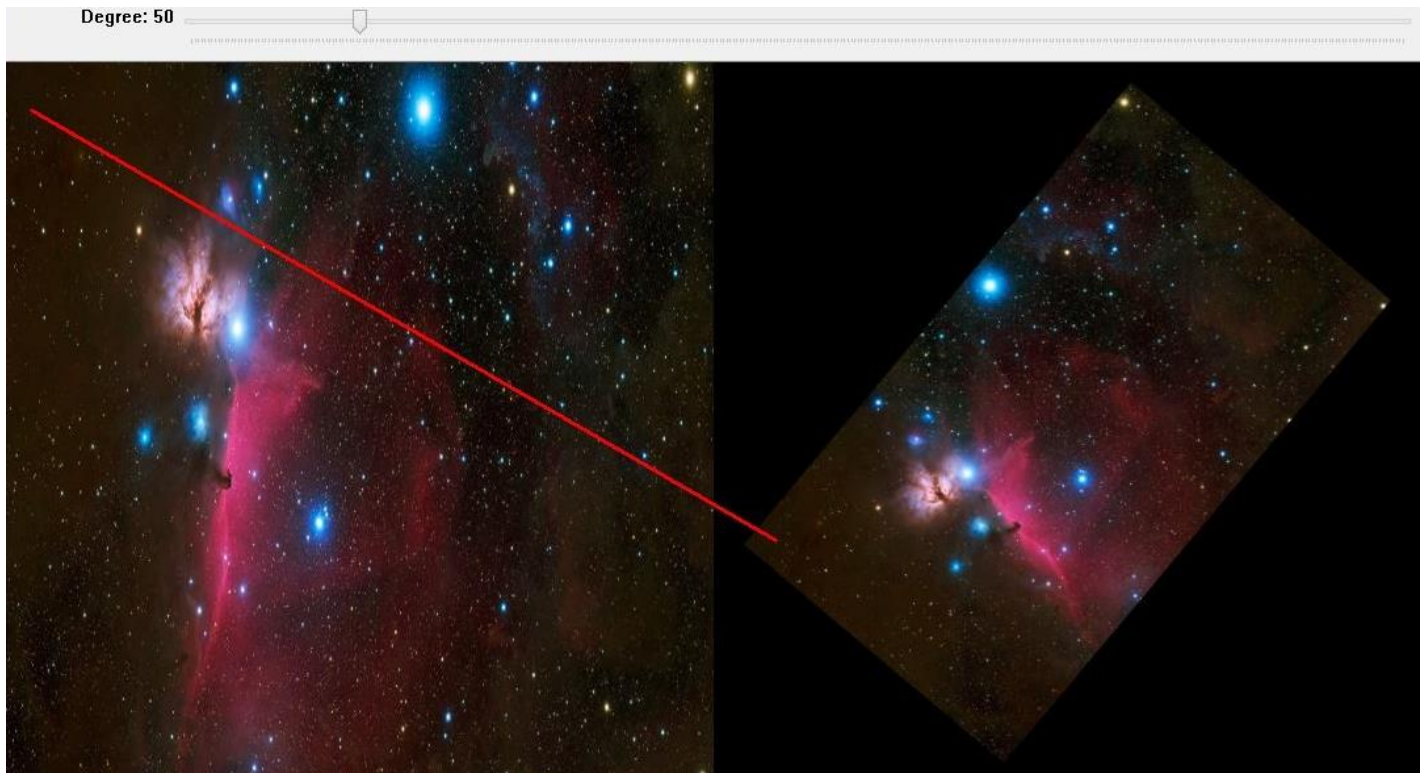
ب) ابتدا با استفاده از دستور `cv2.rectangle` یک مستطیل تقریبی دور توپ رسم می‌کنیم. سپس با کپی کردن پیکسل‌های مربوط به توپ در قسمت خالی چمن، توپ دیگری به تصویر اضافه می‌کنیم.



## سوال ۲:

ابتدا با دستور `cv2.namedWindow` پنجره‌ای برای کاربردهای بعدی ایجاد می‌کنیم. سپس با دستور `cv2.createTrackbar` یک اسلایدر به پنجره ایجاد شده، اضافه می‌کنیم. این اسلایدر تابعی دارد که در هنگام تغییر اسلایدر، فراخوانی می‌شود و تصویر دوران یافته تحت زاویه مورد نظر را ایجاد می‌کند.

برای نمایش بهتر تصویر دوران یافته، اندازه تصویر ثابت را به گونه‌ای تغییر می‌دهیم که مربع نزدیک شود. سپس برای اتصال تصویر ثابت و تصویر تحت دوران به بالا و پایین تصویر تحت دوران به میزان یکسان کادر سیاه اضافه می‌کنیم. در آخر یک نقطه ثابت مثلاً (۲۰ و ۲۰) را در تصویر دوران یافته، دنبال می‌کنیم و با استفاده خط متصل کننده دو تصویر را رسم می‌کنیم.



## سوال ۵:

نحوه عملکرد فیلتر Erosion:

این فیلتر ماترسی به نام `kernel` دریافت می‌کند. سپس برای هر یک از پیکسل‌های تصویر اصلی، مرکز ماتریس `kernel` را بر روی آن پیکسل قرار می‌دهد و مینیمم مقدار پیکسل‌هایی از تصویر اصلی را که مقدار متناظر آن‌ها در ماتریس `kernel` مخالف صفر است، جایگزین مقدار آن پیکسل می‌کند و در نتیجه باعث می‌شود بخش‌های تاریک تصویر بزرگتر شوند. هر چه ابعاد ماتریس `kernel` بزرگتر باشد، میزان بزرگ شدن بخش‌های تاریک هم بیشتر است.



نحوه عملکرد فیلتر dilation:

نحوه عملکرد این فیلتر دقیقا مانند فیلتر Erosion است با این تفاوت که ماکسیمم مقدار جایگزین می شود و به طبع باعث می شود بخش های روشن تصویر بزرگتر شوند. به طور مشابه هر چه ابعاد ماتریس kernel بزرگتر باشد، میزان بزرگ شدن بخش های روشن هم بیشتر است.





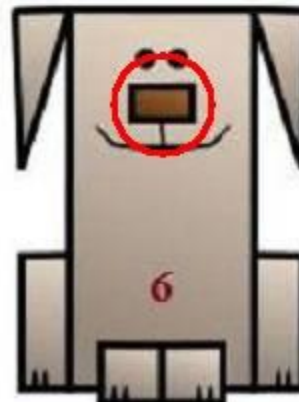
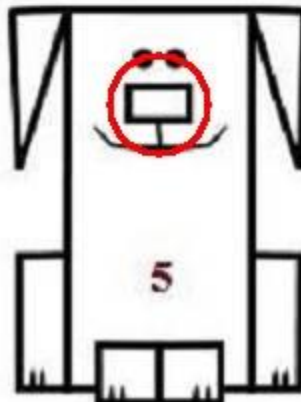
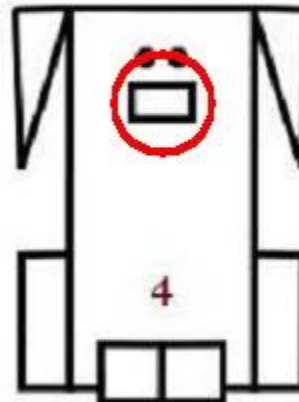
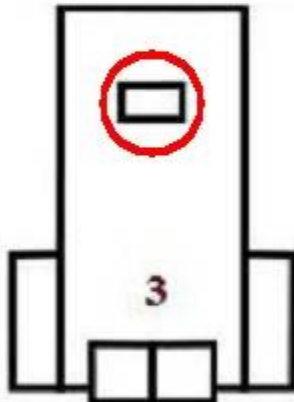
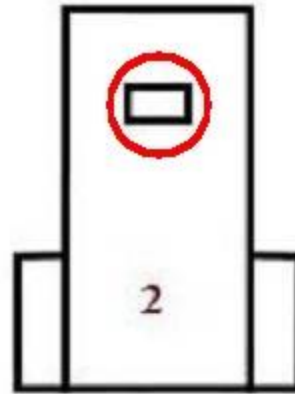
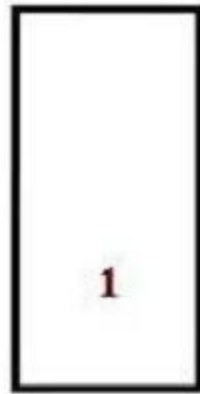
تبدیل Closing: این تبدیل در واقع دو بار فیلتر کردن، اول با فیلتر Dilation و سپس با فیلتر Erosion است. در اثر فیلتر اول نویز های سیاه تصویر حذف می شوند و سپس با فیلتر دوم ابعاد اشکال مختلف تقریبا به حالت اول باز می گردد.



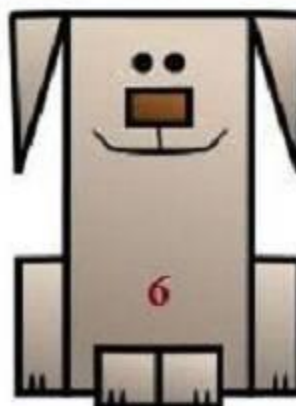
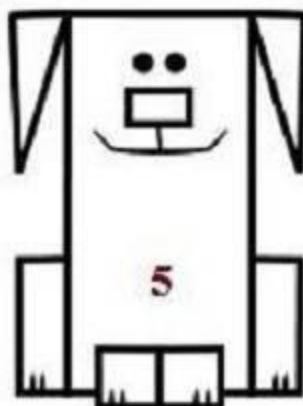
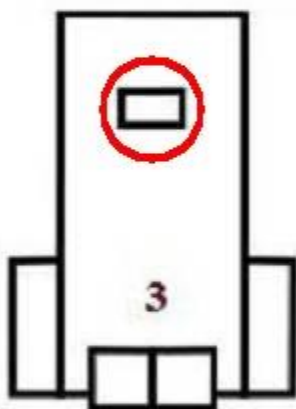
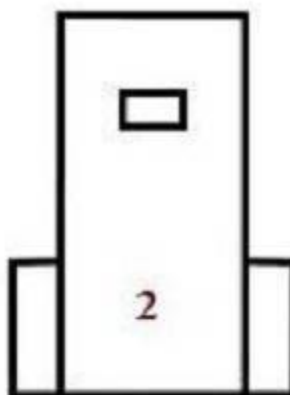
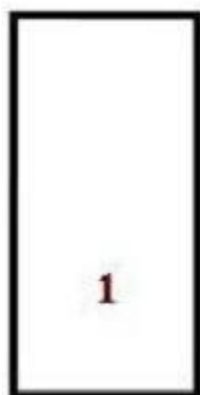
تبدیل Opening: این تبدیل دقیقا مانند تبدیل Closing است با این تفاوت که برای حذف نویز سفید استفاده می شود.



برای تشخیص اجزای مشابه، ابتدا یک نمونه از شکل مورد نظر را تولید می‌کنیم. سپس با تصویر اصلی را با استفاده از این نمونه، فیلتر می‌کنیم. در آخر با قرار دادن یک حد آستانه پیکسل‌های مرکز اجزای مشابه را شناسایی و در یک ماتریس ذخیره می‌کنیم. در انتها برای هریک از این پیکسل‌ها دایره‌ای با شعاع مناسب در تصویر اصلی رسم می‌کنیم.



برای تشخیص دقیق مستطیل مشخص شده روی مختصات پیکسل‌ها محدودیتی اعمال می‌کنیم و فقط به مرکز پیکسل‌هایی که مختصات  $x$  آن‌ها بین  $0/33$  تا  $0/66$  ارتفاع تصویر و مختصات  $y$  آن‌ها کوچکتر از  $0/50$  عرض تصویر است، دایره‌هایی رسم می‌کنیم.





(ب)

سوال ۱:

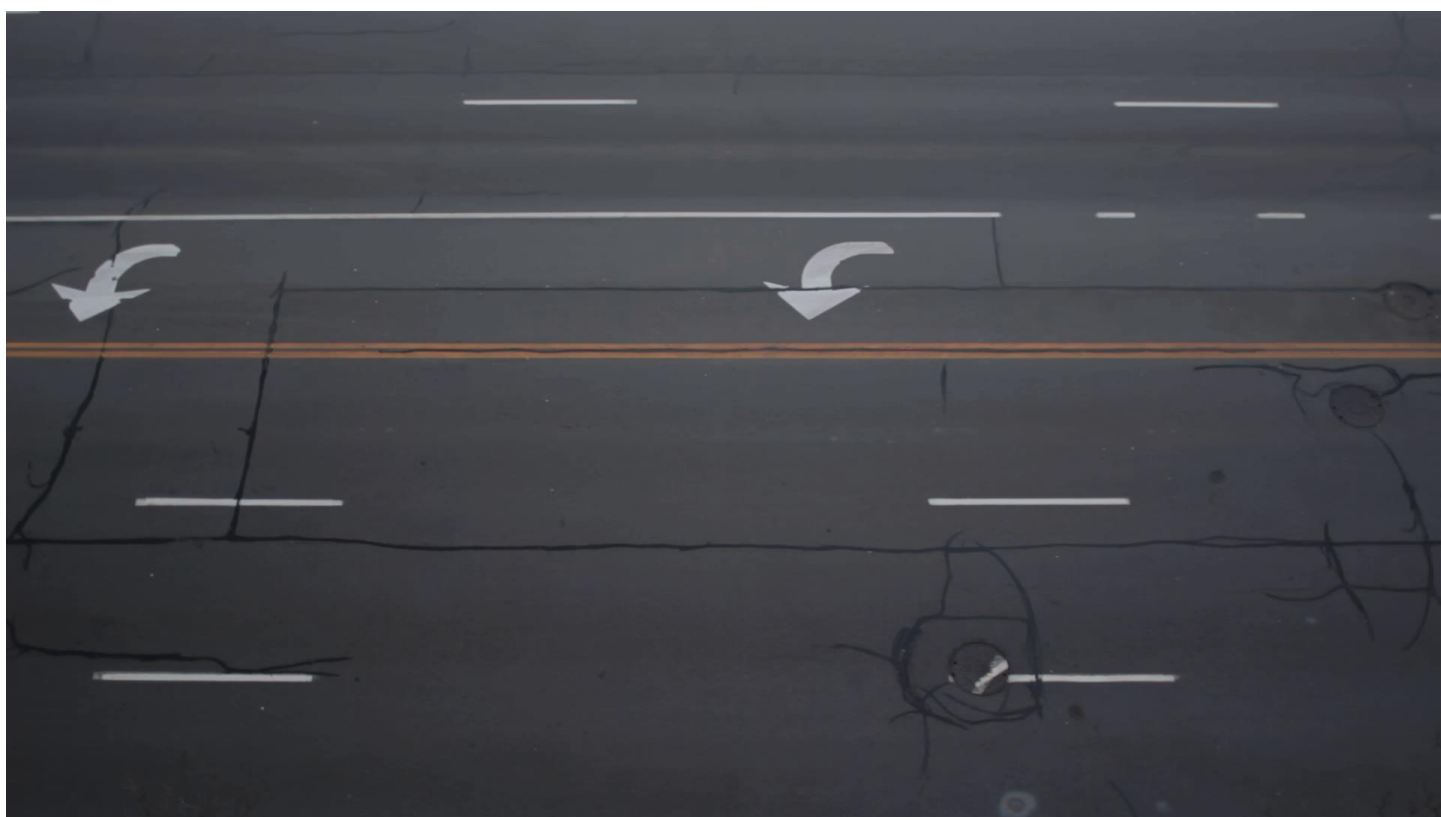
برای گرفتن ویدئو از کلاس VideoCapture و برای ذخیره ویدئو از کلاس VideoWriter استفاده می‌کنیم. سپس در یک حلقه که تا زمان فعال بودن object ساخته شده از کلاس VideoCapture برقرار است، frame های متوالی را دریافت و پخش می‌کنیم.

همچنین با استفاده از دستور cv2.waitKey کلیدهای ورودی را بررسی و در صورت فشرده شدن کلید s یا e برای ذخیره یا بستن ویدئو اقدام می‌کنیم.

سوال ۲:

چون در صورت ثابت دوربین، تصویر پس زمینه در طول ویدئو ثابت است، می‌توان با جدا کردن تمام frame های ویدئو و سپس انتخاب میانه هر پیکسل در تمام تصاویر، تصویر پس زمینه را بدست آورد. علت آن است که در این ویدئو در اکثر تصاویر برای هر پیکسل تصویر پس زمینه را مشاهده می‌کنیم و فقط در تعداد اندکی از تصاویر ماشین‌های عبوری به جای پس زمینه دیده می‌شوند.

روش دیگری که برای تشخیص تصویر پس زمینه می‌توان استفاده کرد، انتخاب مد هر پیکسل است.



در انتها با استفاده از دستور cv2.subtract برای هر تصویر تفاوت آن تصویر با تصویر پس زمینه را محاسبه می‌کنیم و با تغییر تصویر با سرعت مناسب، ویدئو مورد نظر را تولید می‌کنیم.



Activate Windows  
Go to PC settings to activate Windows.