



به نام خدا



دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

دانشکده مهندسی مکانیک

رباتیک و مکاترونیک

پروژه شماره 4

نام و نام خانوادگی	مهدیار بیان -- سید علی تقوی
شماره دانشجویی	810100106-810100101
تاریخ ارسال گزارش	1403/04/21

**فهرست گزارش سوالات** (لطفاً پس از تکمیل گزارش، این فهرست را به روز کنید.)

- سوال 1 – YOLO different versions..... 3
- سوال 2 – mAP score..... 4
- سوال 3 – Project dataset ..... 5
- سوال 4 – Object detection ..... 10
- سوال 5 – Object segmenation ..... 19
- سوال 6 – Grasp point generation ..... 22

## سوال 1 – YOLO different versions

YOLO has published different versions throughout time, describe one difference in each version in comparison with the older version. (Getting help from ChatGPT is valid for this question.)

نسخه YOLO-v2: در YOLO-v2، شبکه بهبود یافته و از Batch Normalization استفاده شده است که دقت تشخیص را افزایش می‌دهد و همچنین از Anchor Boxes برای بهبود پیش‌بینی استفاده می‌کند.

نسخه YOLO-v3: در YOLO-v3، ساختار شبکه به صورت چند مقیاسی تغییر یافته و از سه لایه خروجی مختلف برای پیش‌بینی استفاده می‌شود که دقت بیشتری در تشخیص اشیاء کوچک دارد.

نسخه YOLO-v4: در YOLO-v4، از روش‌های جدیدی مانند CSPDarknet53 به عنوان شبکه ستون فقرات استفاده شده و تکنیک‌های بهینه‌سازی مانند Mosaic Data Augmentation و Self-Adversarial Training اضافه شده‌اند که باعث بهبود عملکرد و دقت می‌شوند.

نسخه YOLO-v5: در YOLO-v5، بهینه‌سازی‌های بیشتری در سرعت و کارایی انجام شده و مدل به صورت پیش‌فرض با PyTorch پیاده‌سازی شده است که استفاده و اجرا را ساده‌تر می‌کند.

نسخه YOLO-v6: در YOLO-v6، بهینه‌سازی‌های بیشتری در زمینه یادگیری عمیق و استفاده از الگوریتم‌های جدیدتر انجام شده است که باعث افزایش دقت و سرعت تشخیص می‌شود.



تصویر 1-1 ورژن‌های YOLO در طی زمان

## سوال 2 – mAP score

YOLO uses mAP (mean average percision) score as an evaluation metrics. Describe how this score is calculated. (Getting help from ChatGPT is valid for this question.)

### 1- محاسبه Precision و Recall:

**Precision:** نسبت تعداد پیش‌بینی‌های صحیح (True Positives) به مجموع پیش‌بینی‌های مثبت (True Positives + False Positives) است.

**Recall:** نسبت تعداد پیش‌بینی‌های صحیح (True Positives) به مجموع تعداد اشیاء واقعی (True Positives + False Negatives).

### 2- ترسیم منحنی Precision-Recall :

برای هر کلاس، مقادیر Precision و Recall را در نقاط مختلف مقدار احتمال پیش‌بینی (Confidence Threshold) محاسبه کنید.

یک منحنی Precision-Recall برای هر کلاس ترسیم کنید که محور افقی نشان‌دهنده Recall و محور عمودی نشان‌دهنده Precision باشد.

### 3- محاسبه: Average Precision (AP)

مساحت زیر منحنی Precision-Recall (AUC) برای هر کلاس محاسبه می‌شود. این مساحت نشان‌دهنده Average Precision (AP) برای آن کلاس است.

### 4- محاسبه: mean Average Precision (mAP)

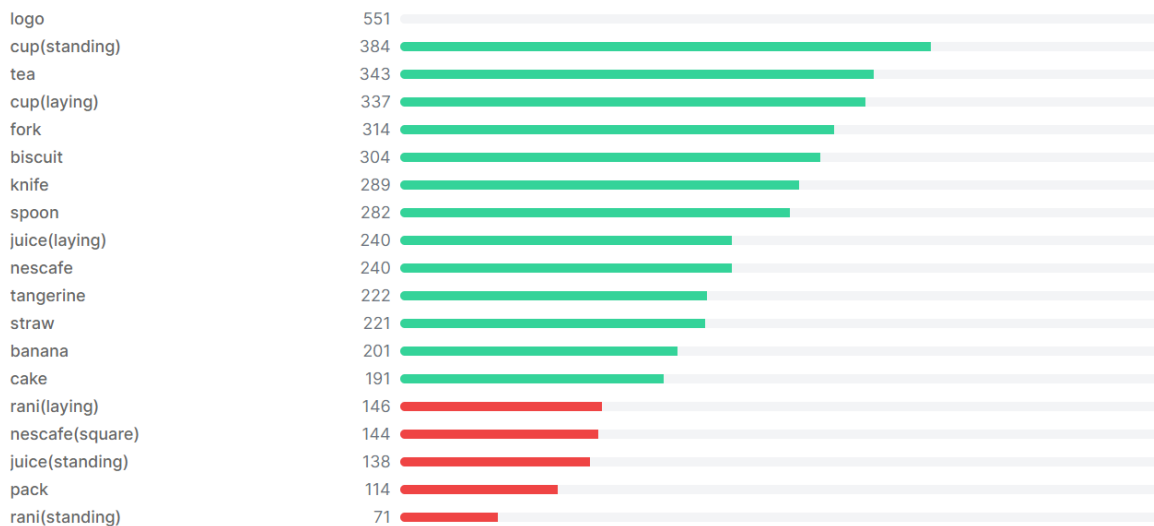
AP برای هر کلاس را محاسبه کرده و سپس میانگین این مقادیر را برای تمامی کلاس‌ها به دست آورید. فرمول:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$$

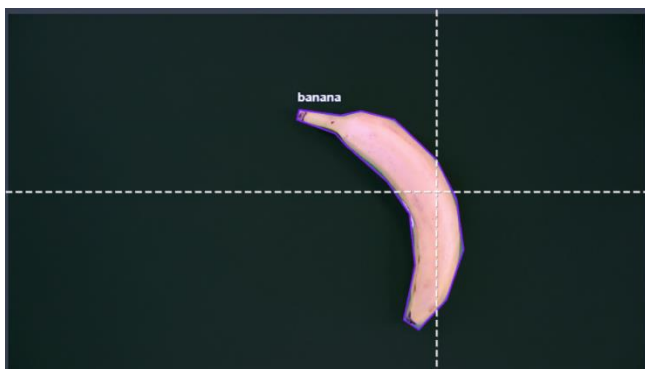
### سوال 3 – Project dataset

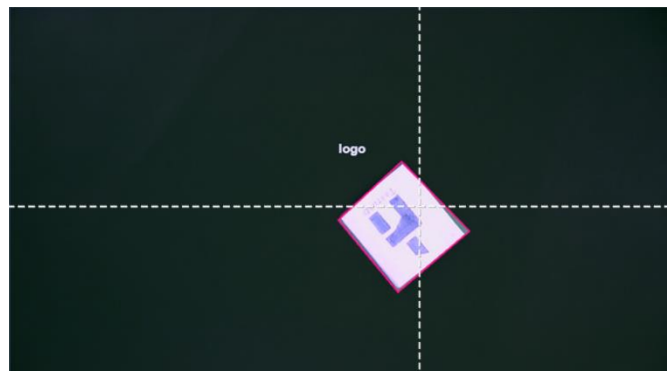
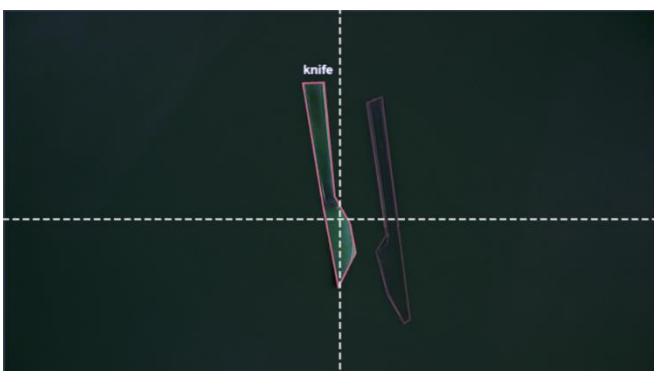
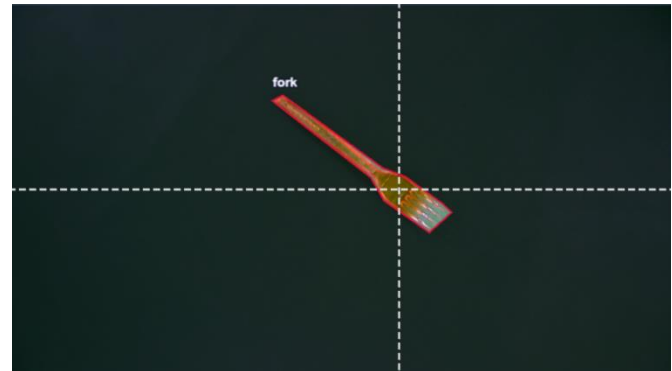
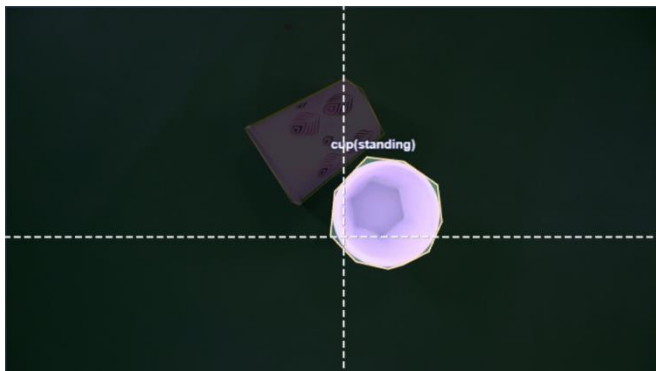
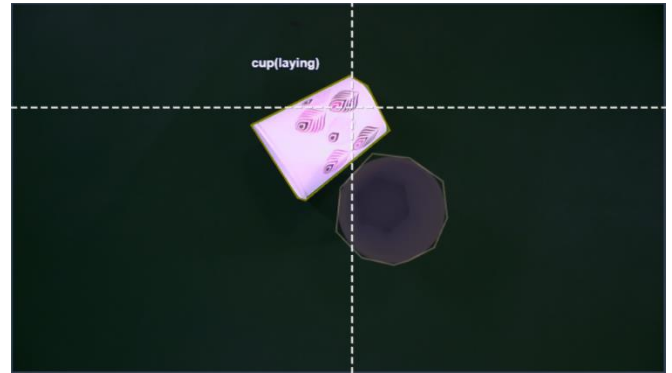
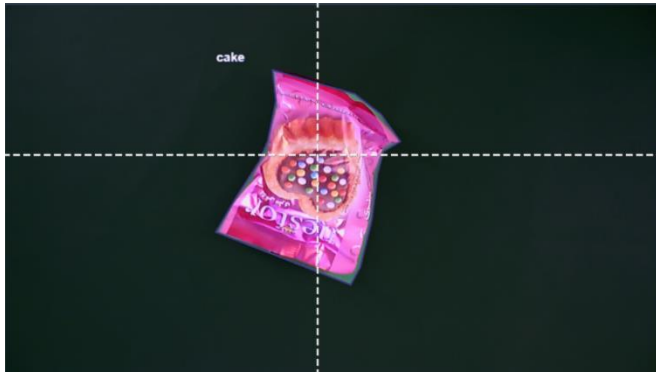
1. Open the link to the dataset in roboflow platform. Analyze this dataset regarding its even distribution throughout different classes. Report the number of images and instances of each class in this dataset.

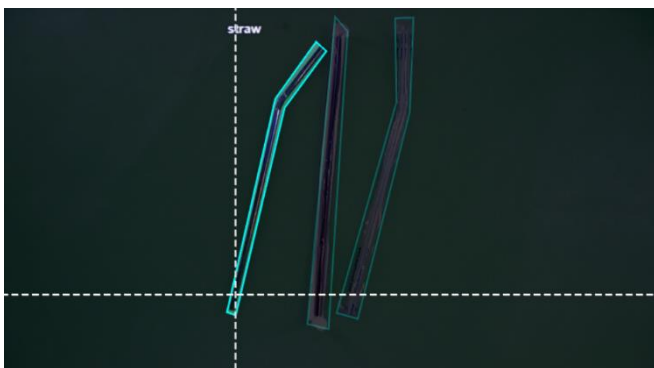
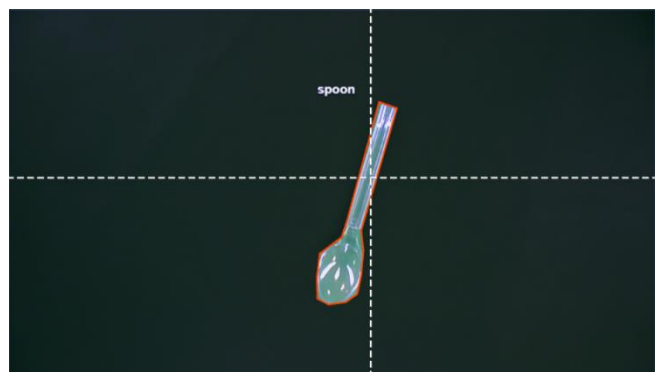
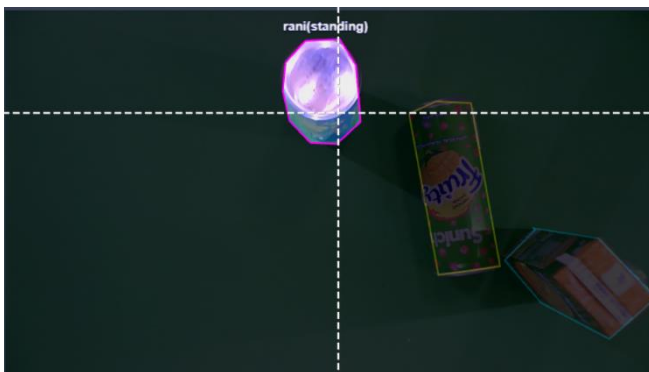
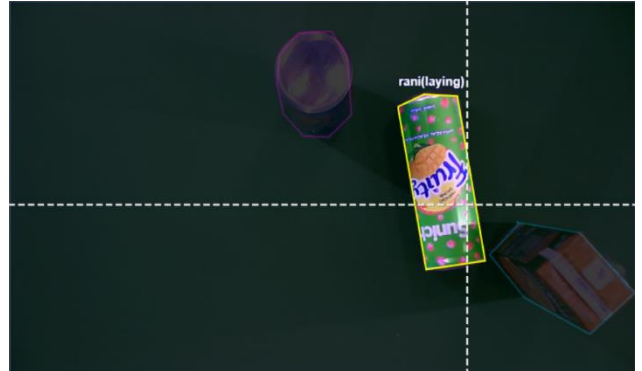
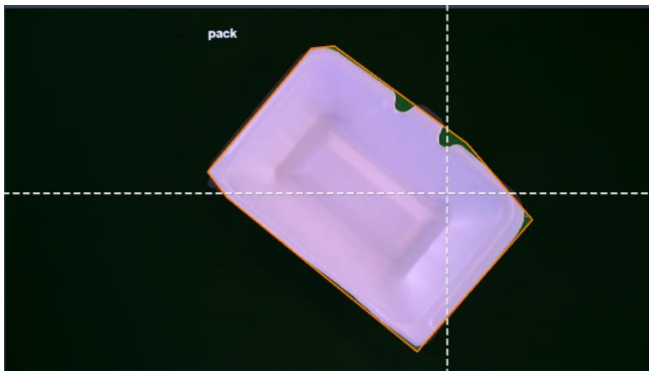
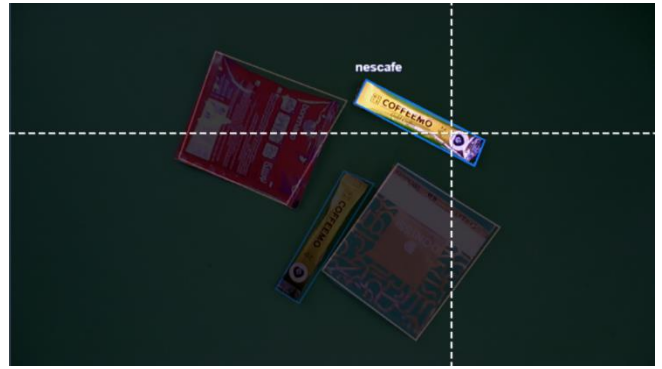
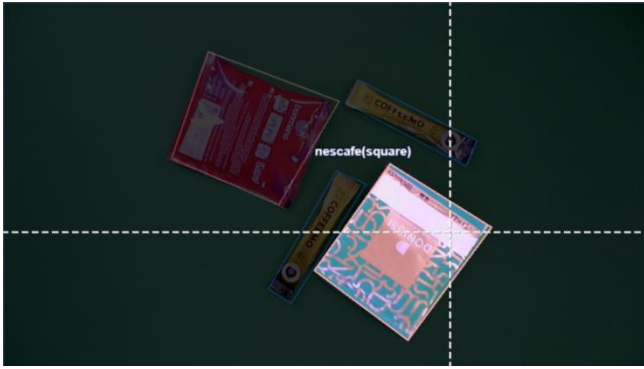
در سایت داده شده ما دارای یک دیتاست هستیم که شامل چندین کلاس و از هر کلاس تعدادی نمونه داریم تعداد کل دیتاهایی که داریم برابر است با 1408 عکس و از هر کدام از کلاس ها به تعداد زیر عکس داریم:

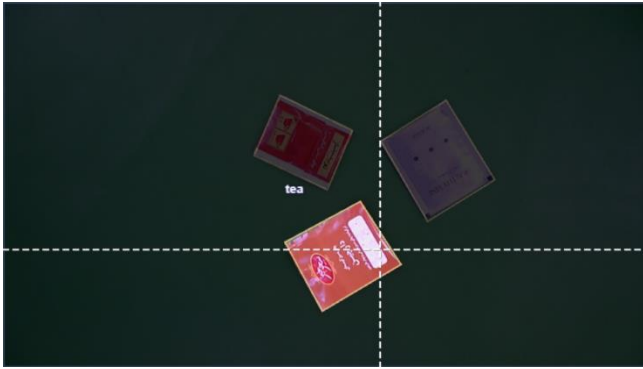


همانطور که در شکل بالا نیز مشاهده میکنید بصورت کلی در دیتاست دارای 19 کلاس مختلف هست که آنها را بصورت زیر داریم:









## 2. Why do we use augmentation steps in the process of developing a dataset?

در این قسمت می‌خواهیم به کاربرد های data augmentation بپردازیم.

یکی از کاربرد های آن افزایش تعداد داده‌ها هست بطور مثال برای train کردن یک سیستم شاید نیاز به تعداد خیلی بالایی از عکس باشد ولی میتوان با گرفتن تعداد عکس کمتری و انجام data augmentation روی آنها با تعداد عکس کمتری به تعداد داده مطلوب برسیم.

یکی دیگر از کاربرد های این روش مقاوم سازی سیستم به اتفاقاتی است که ممکن است در واقعیت اتفاق بیافتد بطور مثال یک عکس داریم و از آن چند عکس با طیف نور کمتر و یا بیشتر نیز به داده ها اضافه میکنیم اتفاقی که میافتد این است که سیستم در عمل نسبت به نور مقاوم تر میشود و با تغییر نور جزئی باز هم به درستی تشخیص خواهد داد و انیکار را میتوانیم برای هر نوع نویز دیگری هم انجام دهیم و نهایت سیستم را نسبت به همان نویز مقاوم تر کنیم.

یکی دیگر از کاربردها این است که اگر ما به دیتا با سایزهای مختلفی نیاز داشته باشیم نیاز نیست که با سایز مختلف عکس برداری کنیم و میتوانیم همان عکس ها را به سایز های مختلف تبدیل کنیم.



3. What are the augmentation steps used in this dataset? How are they useful for this project?

در این قسمت میخواهیم متوجه شویم که در دیتاست داده شده از چه روش های data augmentation استفاده شده است.

با توجه به توضیحاتی که در خود سایت نوشته شده است انواع data augmentation استفاده شده بصورت زیراند:

**Augmentations**      Outputs per training example: 3  
Rotation: Between  $-15^{\circ}$  and  $+15^{\circ}$   
Bounding Box: Noise: Up to 5% of pixels

که عکس بالا به این معنا هست که بطور میانگین برای هر عکس سه data augmentation انجام شده است و با دو روش چرخاندن با زاویه بین  $\pm 15^{\circ}$  و ایجاد نویز 5 درصدی روی پیکسل ها هست.

که این کار سیستم ما را نسبتا مقاوم تر میکنید هم نسبت به اینکه ممکن است زاویه دوربین کمی عوض شود و هم ممکن است اجسام کمی دچار چرخش شوند و با این کار سیستم را در برابر این حوادث مقاوم کردیم و همینطور نویز نیز سیستم را در برابر اینکه شاید دقت دوربین کمی کم و زیاد شود و دچار خطا شود مقاوم میکند.

## سوال 4 – Object detection

1. Import the dataset using the link to the dataset that has been provided in the last section. You can export a link to the dataset regarding to which versions of YOLO want to use in the next sections.

در این قسمت با توجه به مدل YOLO انتخابی لینک مورد نظر آن مدل را از سایت داده شده در سوال قبل میگیریم تا فایل داده های آن را به کد در ادامه سوال دهیم و از دیتاست آن در کدمان استفاده کنیم.

2. Install and import YOLO in your notebook. Although its up to use which version you want to use, it is highly recommended to use versions above v5.

ما برای این قسمت از ورژن v5 استفاده کردیم و در گوگل کولب کد را اجرا کردیم.

3. Go through the training process of training this dataset on the first version of YOLO you chose. After you're done training, you get a report of your training process. Report this table and discuss it. (Since you have limited GPU sources 50 epochs of training is enough for this project)

بعد از اجرای کد جدولی مطابق با جدول زیر به ما داده میشود:

Class	Images	Instances	P	R	mAP50	mAP50-95: 100% 7/7 [00:04<00:00, 1.50it/s]
all	212	620	0.858	0.841	0.92	0.785
banana	212	72	1	0.95	0.979	0.896
biscuit	212	8	0.64	1	0.954	0.72
cake	212	8	0.685	1	0.967	0.903
cup-laying-	212	8	0.477	0.875	0.821	0.764
cup-standing-	212	8	0.793	1	0.982	0.89
fork	212	12	1	0.839	0.995	0.766
juice-laying-	212	13	0.978	1	0.995	0.904
juice-standing-	212	31	0.979	1	0.995	0.877
knife	212	5	1	0.704	0.995	0.85
logo	212	37	0.763	0.892	0.803	0.681
nescafe	212	40	0.876	0.531	0.595	0.449
nescafe-square-	212	24	0.239	0.792	0.733	0.646
pack	212	34	1	0.965	0.995	0.906
rani-laying-	212	18	0.989	1	0.995	0.886
rani-standing-	212	13	0.915	1	0.951	0.873
spoon	212	14	0.979	1	0.995	0.868
straw	212	5	1	0.43	0.995	0.592
tangerine	212	79	0.987	0.971	0.983	0.869
tea	212	191	1	0.0202	0.746	0.574

\*در جدول بالا ستون اول مربوط به انواع کلاس ها هست.

\*ستون دوم مربوط به تعداد نمونه های استفاده شده در برای ارزیابی و بدست آوردن precision و recall و map و دیگر اطلاعات است.

\*ستون بعد تعداد عکس های هر کلاس هست که تعداد اینها مربوط به دیتاست است.

\*ستون بعد P(precision) هست که دقت مدل را به ما نشان میدهد و نسبت تعداد تشخیص های درست نسبت به کل تشخیص ها هست.

\*ستون بعد نیز R(recall) هست که دفعات فراخانی شدن داده را به ما میدهد یا به عبارتی تعداد تشخیص های درست نسبت به کل تعداد واقعی هر کلاس.

\*ستون بعدی MAP50 نام دارد که به معنی میانگین دقت مدل با معیاد  $IoU=50$  هست.

\*ستون بعدی نیز MAP50-95 هست که همان میانگین دقت هست با معیاد  $IoU$  بین 50 تا 95 هست.

در قسمت all ما عملکرد کلی سیستم را روی کل کلاس ها مشاهده میکنیم که بصورت زیر است:

$$Precision = 0.858$$

$$Recall = 0.841$$

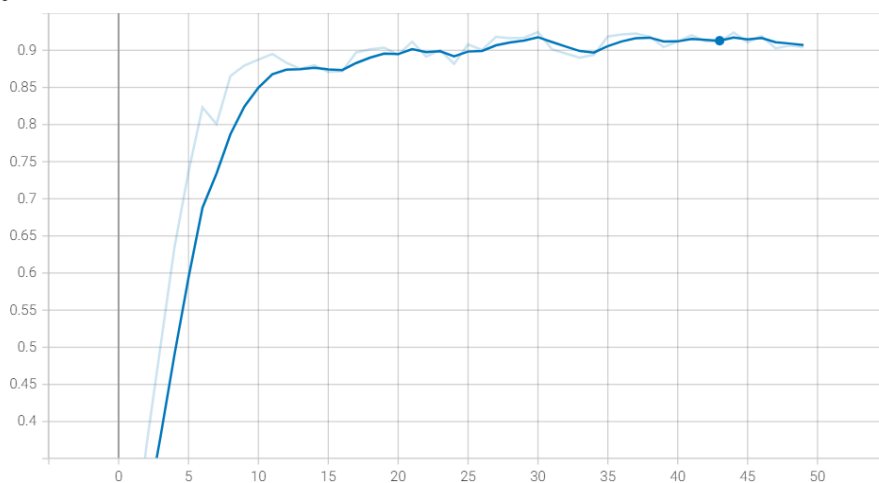
$$MAP50 = 0.92$$

$$MAP50 - 95 = 0.785$$

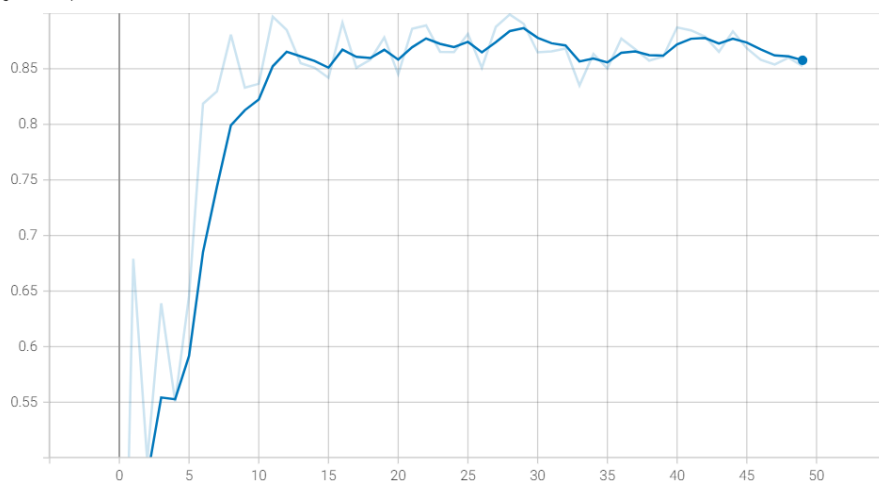
که میتوان گفت بطور کلی سیستم ما عملکرد خوبی داشته است و در ردیف های بعدی نیز بصورت دقیقتری به معیار ها برای هر کلاس میپردازد که برای هر کلاس بطور جدا نوشته شده است.

علت وجود تفاوت در دقت و دیگر معیار ها برای هر کدام از کلاس ها عوامل مختلفی هستند مثلاً عدم وجود توازن در تعداد داده ها و یا شباهت ظاهری بعضی از کلاس ها به یکدیگر و یا تفاوت ظاهری زیاد یک کلاس از بقیه کلاس ها و یا وجود تنوع در عکس های مربوط به یک کلاس و یا حتی پیچیدگی یک کلاس باعث ایجاد این تفاوت ها در نتایج برای کلاس های مختلف شده است.

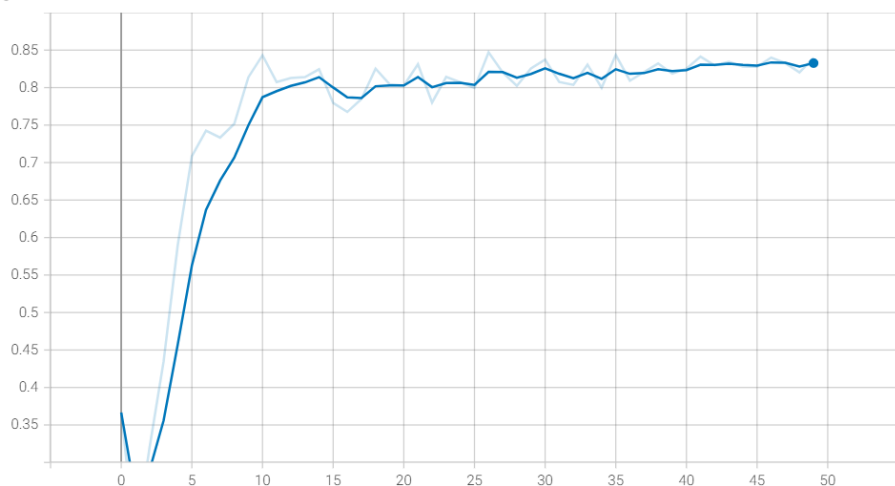
metrics/mAP\_0.5  
tag: metrics/mAP\_0.5



metrics/precision  
tag: metrics/precision



metrics/recall  
tag: metrics/recall



همانطور که در تصاویر بالا مشاهده میکنید در ابتدا دقت و دیگر معیارها مقادیر کمی را نشان میدهند ولی بعد از تعداد بیشتری عکس رو به بهبودی میروند.

4. Report the training time needed per epoch for this version.

زمان سپری شده برای هر epoch بطور میانگین برای 30 ثانیه بود.

Epoch	GPU_mem	box_loss	obj_loss	cls_loss	Instances	Size
49/49	1.74G	0.02044	0.01788	0.004758	42	416: 100% 158/158 [00:31<00:00, 5.00it/s]
	Class	Images	Instances	P	R	mAP50 mAP50-95: 100% 7/7 [00:01<00:00, 3.85it/s]
	all	212	620	0.852	0.84	0.904 0.768

و زمان سپری شده برای کل 50 epochs برابر با 0.466 ساعت یا 27.96 دقیقه.

50 epochs completed in 0.466 hours.

5. You have been provided with a folder of test images. Upload this folder to your colab and test the trained model on these folders. Talk about the results. Why does the model get some outputs wrong?

نتایج کل 357 داده تست در دیتاست اصلی را در فایل تکست با نام YOLO\_Results قرار دادم و حالا می‌خواهیم 10 تا از نمونه‌ها را نشان دهیم و صحت درستی نتایج را بررسی کنیم.

\*در عکس اول مشاهده می‌کنیم که به اشتباه تشخیص داده شده است و چیزی که در عکس هست با چیزی که به عنوان نتیجه داریم یکسان نیست که این اتفاق به علت وجود شباهت بین آبمیوه و لوگو است.



```

image 280/357 /content/datasets/grasp-6-7/test/images/IMG_1231_JPG.rf.1ca9e3378aa85117e70cd3e30d844b81.jpg: 416x416 4 cup-standing-s, 1
logo, 7.1ms
image 281/357 /content/datasets/grasp-6-7/test/images/IMG_1244_JPG.rf.e926a65574e4e736d782d197e94da2c1.jpg: 416x416 2 juice-laying-s, 2
logos, 7.1ms
image 282/357 /content/datasets/grasp-6-7/test/images/IMG_1246_JPG.rf.f9fb6f5dbef51f05ee2437b91c2d45ee.jpg: 416x416 3 juice-laying-s, 3
logos, 7.1ms
image 283/357 /content/datasets/grasp-6-7/test/images/IMG_1254_JPG.rf.21d1c2c10b4445a8adbdd76ff6ad1a4c.jpg: 416x416 2 juice-laying-s, 2
juice-standing-s, 1 logo, 7.1ms
image 284/357 /content/datasets/grasp-6-7/test/images/IMG_1256_JPG.rf.1e03496b7083c25ae0d50b0da81df548.jpg: 416x416 1 juice-laying-, 2
juice-standing-s, 1 logo, 1 straw, 7.1ms

```

\*در این قسمت نیز مشاهده میکنیم که یکی از آبمیوه ها تشخیص داده نشده است که به علت این است که دو جسم از بالا با هم تداخل دارند و یکی از آبمیوه ها بصورت کامل نشان داده نشده و است و به همین علت تشخیص هم داده نشده است.



```

image 315/357 /content/datasets/grasp-6-7/test/images/IMG_3021_JPG.rf.7b587c994d24058f19027af02e4b2a92.jpg: 416x416 2 juice-laying-s, 1
juice-standing-, 1 logo, 1 rani-laying-, 7.1ms
image 316/357 /content/datasets/grasp-6-7/test/images/IMG_3023_JPG.rf.2b1c9eec20ae06871d88763863219cf7.jpg: 416x416 2 juice-laying-s, 1
logo, 2 rani-laying-s, 7.1ms
image 317/357 /content/datasets/grasp-6-7/test/images/IMG_3025_JPG.rf.948b0342fd008f051b02fe612cc79d20.jpg: 416x416 1 juice-laying-, 1
logo, 1 rani-laying-, 1 rani-standing-, 7.1ms
image 318/357 /content/datasets/grasp-6-7/test/images/IMG_3033_JPG.rf.538d742213a513e90c10c5ac2bf371b2.jpg: 416x416 1 juice-laying-, 1
juice-standing-, 1 logo, 2 rani-laying-s, 1 rani-standing-, 7.1ms
image 319/357 /content/datasets/grasp-6-7/test/images/IMG_3036_JPG.rf.5c77a612d2c17e855e8b3d2b24673070.jpg: 416x416 1 juice-laying-, 1
juice-standing-, 1 logo, 2 rani-laying-s, 7.1ms

```



\*همانطور که مشاهده میکنید در این قسمت نیز یک کارد بیشتر تشخیص داده شده است.



image 332/357 /content/datasets/grasp-6-7/test/images/IMG\_3073.JPG.rf.9890783f1f919ab1052eaad331eea471.jpg: 416x416 1 juice-laying-, 1 logo, 2 rani-laying-s, 7.1ms  
image 333/357 /content/datasets/grasp-6-7/test/images/IMG\_3235.JPG.rf.f7516e23a5f2df03d3aa04e613d877a5.jpg: 416x416 5 knives, 7.1ms  
image 334/357 /content/datasets/grasp-6-7/test/images/IMG\_3236.JPG.rf.7566023fd25416a74b9c4e483989cd29.jpg: 416x416 5 knives, 7.1ms  
image 335/357 /content/datasets/grasp-6-7/test/images/IMG\_3241.JPG.rf.799254e56b226d08746420b7e4b3f8de.jpg: 416x416 1 cup-laying-, 2 spoons, 7.1ms  
image 336/357 /content/datasets/grasp-6-7/test/images/IMG\_3243.JPG.rf.f71939e15a9950b7c3ca352d898166bc.jpg: 416x416 1 fork, 3 spoons, 12.1ms

\*در این قسمت نیز دچار مقداری خطا در تشخیص شده ایم.



```

image 334/357 /content/datasets/grasp-6-7/test/images/IMG_3236_JPG.rf.7566023fd25416a74b9c4e483989cd29.jpg: 416x416 5 knives, 7.1ms
image 335/357 /content/datasets/grasp-6-7/test/images/IMG_3241_JPG.rf.799254e56b226d08746420b7e4b3f8de.jpg: 416x416 1 cup-laying-, 2
spoons, 7.1ms
image 336/357 /content/datasets/grasp-6-7/test/images/IMG_3243_JPG.rf.f71939e15a9950b7c3ca352d898166bc.jpg: 416x416 1 fork, 3 spoons,
12.1ms
image 337/357 /content/datasets/grasp-6-7/test/images/IMG_3248_JPG.rf.b7503675a3a5be17cc1b7704f069c050.jpg: 416x416 4 forks, 1 knife, 1
spoon, 7.1ms
image 338/357 /content/datasets/grasp-6-7/test/images/IMG_3249_JPG.rf.fae224187cdcd312cc6b13b4014d788.jpg: 416x416 3 forks, 7.1ms
image 339/357 /content/datasets/grasp-6-7/test/images/IMG_3250_JPG.rf.7e37d55fdb61562f120c3e5d73fb2cc6.jpg: 416x416 3 forks, 7.2ms

```

\*در این داده تست مشاهده میکنیم که به درستی تشخیص داده شده است.



```

image 339/357 /content/datasets/grasp-6-7/test/images/IMG_3250_JPG.rf.7e37d55fdb61562f120c3e5d73fb2cc6.jpg: 416x416 3 forks, 7.2ms
image 340/357 /content/datasets/grasp-6-7/test/images/IMG_E1332_JPG.rf.b6371cea90257af51842cad1de5f79bc.jpg: 416x416 2 forks, 7.1ms
image 341/357 /content/datasets/grasp-6-7/test/images/IMG_E1334_JPG.rf.2b22c624aade34fc1c01ab783dad9cdb.jpg: 416x416 2 forks, 7.1ms
image 342/357 /content/datasets/grasp-6-7/test/images/IMG_E1342_JPG.rf.f57949b3b1fcde7462f3868bc0dedf1c.jpg: 416x416 2 spoons, 7.1ms
image 343/357 /content/datasets/grasp-6-7/test/images/IMG_E1344_JPG.rf.85e838aa74e84fbbcca44c68d9c23109.jpg: 416x416 2 spoons, 7.1ms
image 344/357 /content/datasets/grasp-6-7/test/images/IMG_E1352_JPG.rf.e0484e3242c4d6cfeb96092f481a53da.jpg: 416x416 2 spoons, 7.1ms
image 345/357 /content/datasets/grasp-6-7/test/images/IMG_E1359_JPG.rf.56cb5cc189a38c421eb52b10810b8bd4.jpg: 416x416 2 knives, 7.1ms

```



\*در این قسمت نیز مشاهده میکنیم که به درستی تشخیص داده شده است.



```
image 344/357 /content/datasets/grasp-6-7/test/images/IMG_E1352.JPG.rf.e0484e3242c4d6cfeb96092f481a53da.jpg: 416x416 2 spoons, 7.1ms  
image 345/357 /content/datasets/grasp-6-7/test/images/IMG_E1359.JPG.rf.56cb5cc189a38c421eb52b10810b8bd4.jpg: 416x416 2 knives, 7.1ms  
image 346/357 /content/datasets/grasp-6-7/test/images/IMG_E1363.JPG.rf.b2d538915530b797ab65dc01d4d68d7d.jpg: 416x416 1 knife, 7.1ms  
image 347/357 /content/datasets/grasp-6-7/test/images/IMG_E1367.JPG.rf.f343ecadb8d7980f967b4e0af4cf3.jpg: 416x416 1 knife, 7.1ms  
image 348/357 /content/datasets/grasp-6-7/test/images/IMG_E1368.JPG.rf.135e7a241e868be6855f71ff4e4cacc8.jpg: 416x416 2 knives, 7.1ms  
image 349/357 /content/datasets/grasp-6-7/test/images/IMG_E1369.JPG.rf.2b535db1936948675c6eb57e50c60377.jpg: 416x416 2 knives, 7.1ms  
image 350/357 /content/datasets/grasp-6-7/test/images/IMG_E1370.JPG.rf.d143f31063431c35130eb5527270ffd6.jpg: 416x416 2 straws, 7.1ms
```

\*در این داده یک شیء در قسمت راست تصویر وجود دارد که آن نی تشخیص داده شده است.



image 347/357 /content/datasets/grasp-6-7/test/images/IMG\_E1367.JPG.rf.f343ecadb878d7980f967b4e0af4cf3.jpg: 416x416 1 knife, 7.1ms  
 image 348/357 /content/datasets/grasp-6-7/test/images/IMG\_E1368.JPG.rf.135e7a241e868be6855f71ff4e4cacc8.jpg: 416x416 2 knives, 7.1ms  
 image 349/357 /content/datasets/grasp-6-7/test/images/IMG\_E1369.JPG.rf.2b535db1936948675c6eb57e50c60377.jpg: 416x416 2 knives, 7.1ms  
 image 350/357 /content/datasets/grasp-6-7/test/images/IMG\_E1370.JPG.rf.d143f31063431c35130eb5527270fffd6.jpg: 416x416 2 straws, 7.1ms  
 image 351/357 /content/datasets/grasp-6-7/test/images/IMG\_E1376.JPG.rf.cac4eefe04dda987ab0ce55573000e8a.jpg: 416x416 2 straws, 7.1ms  
 image 352/357 /content/datasets/grasp-6-7/test/images/IMG\_E1383.JPG.rf.b8287a15813855f0dd724e251b1b640a.jpg: 416x416 2 straws, 7.1ms  
 image 353/357 /content/datasets/grasp-6-7/test/images/IMG\_E1386.JPG.rf.d7289bcd6306733570f8e0bbf16ef1f3.jpg: 416x416 1 cake, 8.7ms

این تصویر نیز به درستی تشخیص داده شده است.

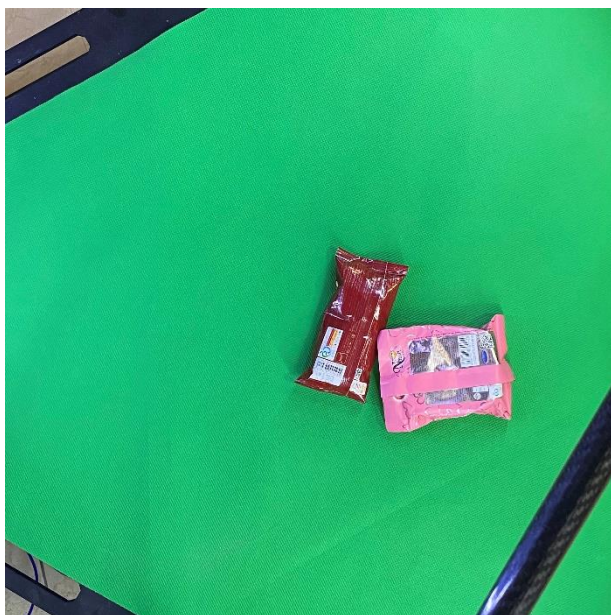


image 352/357 /content/datasets/grasp-6-7/test/images/IMG\_E1383.JPG.rf.b8287a15813855f0dd724e251b1b640a.jpg: 416x416 2 straws, 7.1ms  
 image 353/357 /content/datasets/grasp-6-7/test/images/IMG\_E1386.JPG.rf.d7289bcd6306733570f8e0bbf16ef1f3.jpg: 416x416 1 cake, 8.7ms  
 image 354/357 /content/datasets/grasp-6-7/test/images/IMG\_E1388.JPG.rf.abb7dcdbd42800df3cd9db2499aeea9.jpg: 416x416 1 cake, 7.1ms  
 image 355/357 /content/datasets/grasp-6-7/test/images/IMG\_E1392.JPG.rf.4aff19dadcd113275df74b50e3fb.jpg: 416x416 2 cakes, 7.1ms  
 image 356/357 /content/datasets/grasp-6-7/test/images/IMG\_E1397.JPG.rf.7e41184d20df2d238380009186d86409.jpg: 416x416 2 cakes, 7.1ms  
 image 357/357 /content/datasets/grasp-6-7/test/images/IMG\_E1503.JPG.rf.b139d0659c3617442b72dcd8ac719a47.jpg: 416x416 1 rani-laying-, 1 rani-standing-, 7.1ms

بصورت کلی دلیل عدم تشخیص درست میتواند وجود کلاس های مشابه مثل نی و کارد و یا قاشق و

چنگال باشد و یا وجود همپوشانی دو جسم در تصویر.

## سوال 5 – Object segmenation

1. Import FastSAM to your colab.

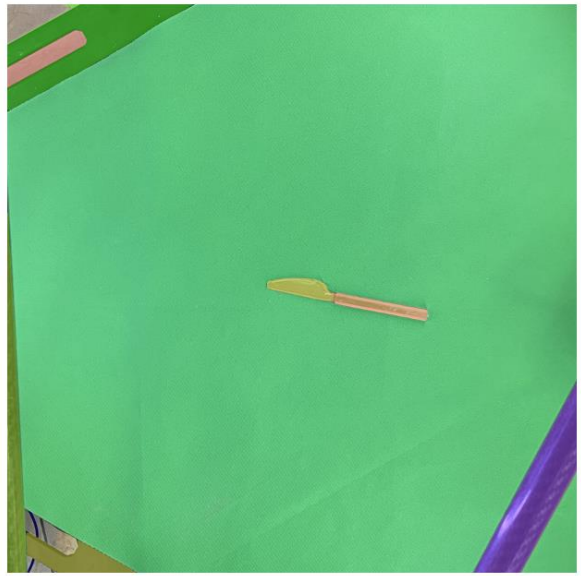
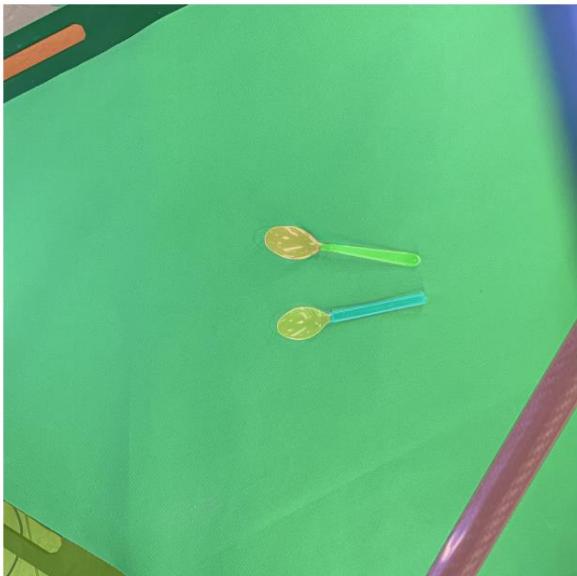
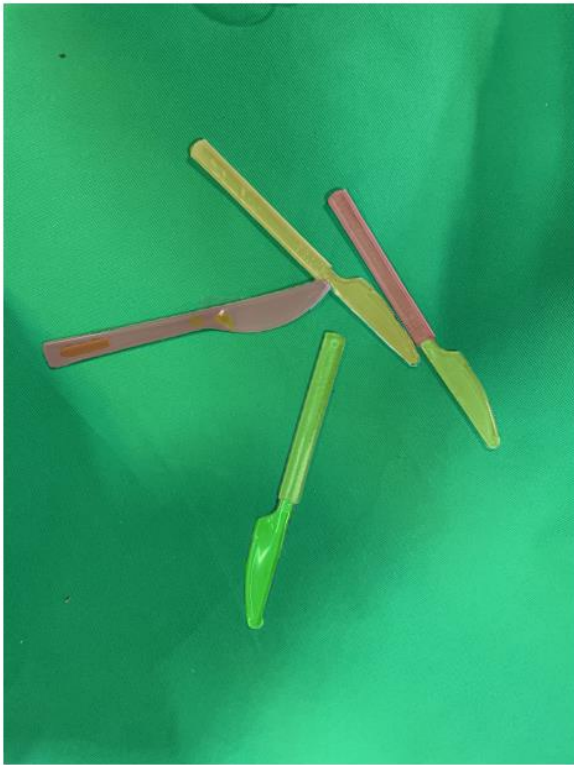
در این قسمت میخواهیم از روش دیگری استفاده کنیم تا بجای آنکه تنها مرز اجسام را تشخیص دهیم بتوانیم کل جسم را شناسایی کنیم برای این کار از SAM کمک میگیریم که برای کار ما ورژن FastSAM نیز کارآمد هست و از آن استفاده خواهیم کرد در ادامه.

2. You have tested your fine-tuned model in previous question on the test folder. Using FastSAM which is a zero-shot segmentation model and the bounding box outputs from YOLO for these images, visualize the output of SAM on these pictures.

حال میخواهیم داده هایی را که در قسمت قبل به YOLO دادیم را در این قسمت نیز به FastSAM بدهیم و نتایج آن را مشاهده کنیم.





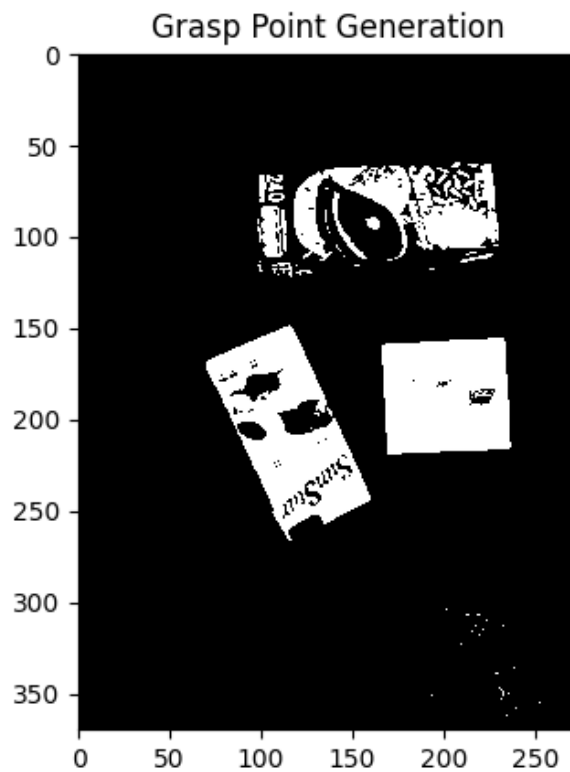




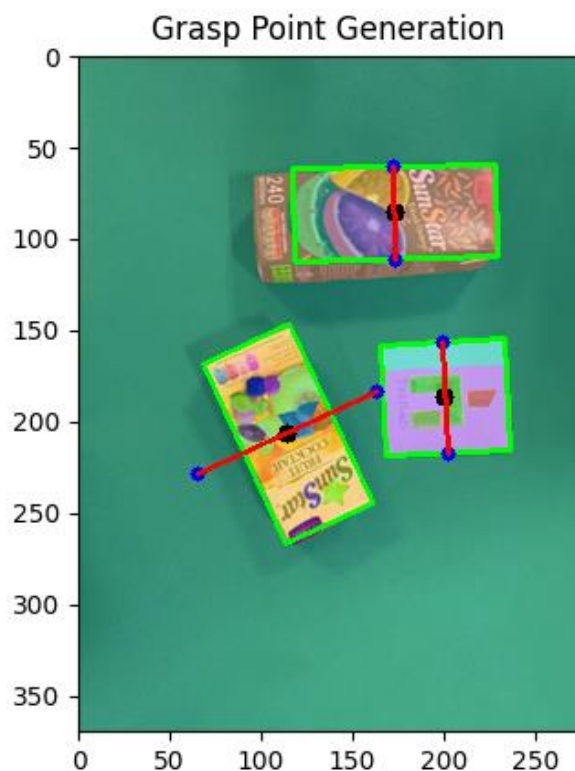
## سوال 6 – Grasp point generation

در این قسمت میخواهیم جسم را توسط ربات برداریم که برای اینکار باید بدانیم که از کدام نقاط برای بلند کردن استفاده کنیم و در این سوال میخواهیم که این نقاط را پیدا کنیم.

برای این قسمت اول از همه شکل را بصورت باینری تبدیل میکنیم که بصورت زیر میشود:



و حالا باید شکل های بدست آمده را در boundary box قرار دهیم و بعد مرکز و زاویه آنها را بدست آوریم و با داشتن مرکز و زاویه میتوان نقاط مورد نظرمان را روی شکل پیدا کنیم که شکل نهایی آن بصورت زیر میشود:



مراجع:

<https://colab.research.google.com/github/roboflow-ai/notebooks/blob/main/notebooks/how-to-segment-anything-with-fast-sam.ipynb#scrollTo=wwtmiH6pLA8N>

<https://colab.research.google.com/github/roboflow-ai/yolov5-custom-training-tutorial/blob/main/yolov5-custom-training.ipynb>