

هدف از انجام این پروژه تشخیص پلاک و تحلیل ۳ مدل یولو (Small و Medium، Large) بر روی دیتاست Licenseplate انجام شده است. این دیتاست شامل ۲۱۵ تصویر در زوایای مختلف و در روشنایی های متفاوت از روز عکسبرداری شده است.

در این پروژه ۱۵ تصویر (که سعی بر آن شده که تمامی حالات در آن گنجانده شده باشد) برای داده های تست، ۱۶۰ تصویر برای داده های آموزش و ۴۰ تصویر برای داده های اعتبارسنجی تقسیم بندی کرده ایم.

با استفاده از نرم افزار label-studio پلاک های ماشین ها را لیبل بندی کرده و تنها یک کلاس پلاک را به آن تخصیص داده ایم.

در ابتدا یک فولدر به نام Dataset ساخته و تصاویر train, validation و test را به همراه لیبل های آن ها قرار می دهیم.

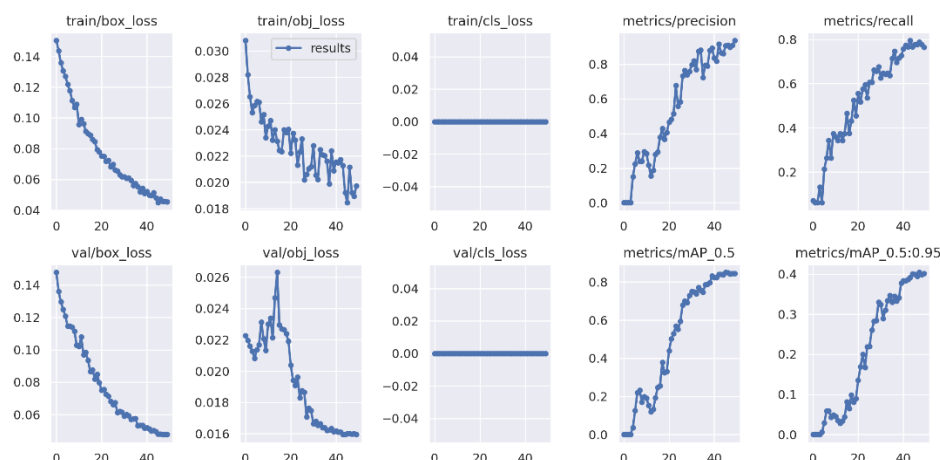
کتابخانه های موجود در requirements یولو و برنامه Wandb را نصب؛ و یک فایل به نام dataset.yaml ساخته و آدرس تصاویر به همراه لیبل های آن ها را مشخص میکنیم.

در YOLOv5 از مدل small, medium و large استفاده کردیم.

Model	size (pixels)	mAP ^{val} 0.5:0.95	mAP ^{val} 0.5	Speed CPU b1 (ms)	Speed V100 b1 (ms)	Speed V100 b32 (ms)	params (M)	FLOPs @640 (B)
YOLOv5n	640	28.0	45.7	45	6.3	0.6	1.9	4.5
YOLOv5s	640	37.4	56.8	98	6.4	0.9	7.2	16.5
YOLOv5m	640	45.4	64.1	224	8.2	1.7	21.2	49.0
YOLOv5l	640	49.0	67.3	430	10.1	2.7	46.5	109.1

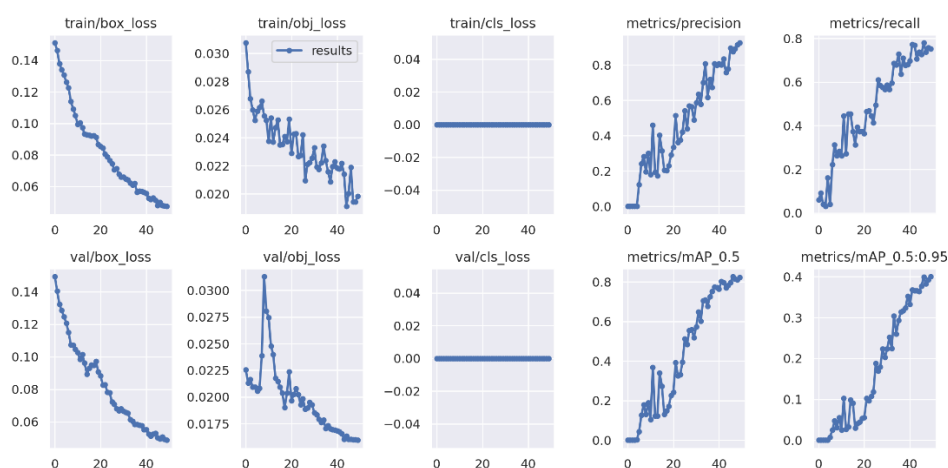
در جدول زیر مقادیر به همراه نتایج آن برای مدل small آورده شده است.

Train	Batch= 25	Epochs= 50	Inference
wandb:	best/epoch 47		Class: all
wandb:	best/mAP_0.5 0.84297		Images: 40
wandb:	best/mAP_0.5:0.95 0.4046		Instances: 99
wandb:	best/precision 0.89941		P: 0.939
wandb:	best/recall 0.78788		R: 0.737
wandb:	metrics/mAP_0.5 0.84325		mAP50: 0.841
wandb:	metrics/mAP_0.5:0.95 0.40411		mAP50-95: 0.405
wandb:	metrics/precision 0.89895		
wandb:	metrics/recall 0.78788		
Train	Batch= 32	Epochs= 50	Inference
wandb:	best/epoch 49		Class: all
wandb:	best/mAP_0.5 0.82343		Images: 40
wandb:	best/mAP_0.5:0.95 0.40039		Instances: 99
wandb:	best/precision 0.92538		P: 0.933
wandb:	best/recall 0.75167		R: 0.747
wandb:	metrics/mAP_0.5 0.82343		mAP50: 0.821
wandb:	metrics/mAP_0.5:0.95 0.40103		mAP50-95: 0.383
wandb:	metrics/precision 0.92539		
wandb:	metrics/recall 0.75175		
Train	Batch= 20	Epochs= 300	Inference
wandb:	best/epoch 283		Class: all
wandb:	best/mAP_0.5 0.89774		Images: 40
wandb:	best/mAP_0.5:0.95 0.46707		Instances: 99
wandb:	best/precision 0.96267		P: 0.943
wandb:	best/recall 0.83838		R: 0.836
wandb:	metrics/mAP_0.5 0.8977		mAP50: 0.891
wandb:	metrics/mAP_0.5:0.95 0.4676		mAP50-95: 0.464
wandb:	metrics/precision 0.96295		
wandb:	metrics/recall 0.83838		



شکل (۱): YoloV5s - batch: 25, epochs: 50, image-size: 640

با توجه به نمودار اول از سمت چپ (train/box_loss) و (val/box_loss) هر دو با هم روند نزولی داشته و نشان دهنده این است که overfit اتفاق نیفتاده است. همچنین از شیب تند نمودار metrics/precision صعودی است متوجه می شویم که مدل میتواند به دقت بیشتری دست یابد.



شکل (۲): YoloV5s - batch: 32, epochs: 50, image-size: 640

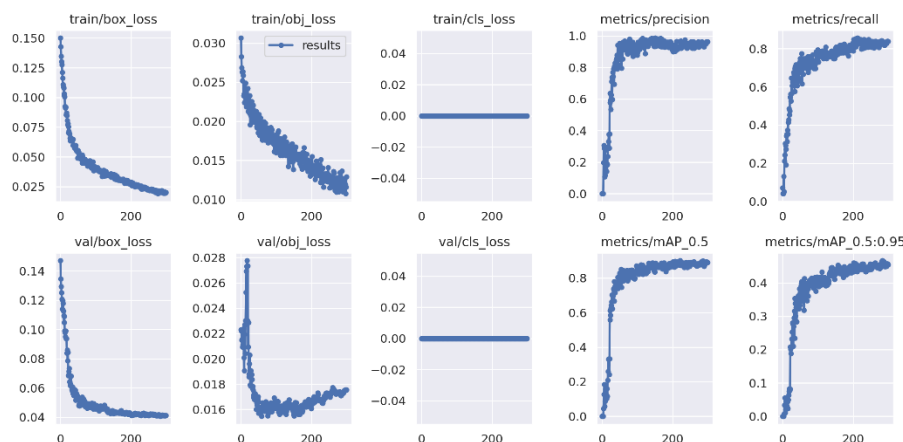
در این جا نیز بیش برآزش یا overfit رخ نداده است. همچنین با توجه به نمودار train/box_loss که شیب نزولی آن زیاد است و هنوز به یک روند ثابتی نرسیده؛ می توان این نتیجه را گرفت که مدل، قابلیت آموزش بیشتر را دارد و میتوان تعداد epochs را بیشتر کرد.

Precision حداکثر مقدار این معیار یک و یا ۱۰۰ درصد و حداقل مقدار آن صفر است و هرچه مواردی که برنامه به غلط پیش بینی کرده است که به آن False Positive می گوئیم نسبت به پیش بینی های درست یا True

Positive بیشتر باشد مقدار Precision کمتر خواهد شد. با توجه به نمودار چهارم از سمت چپ، روند صعودی آن زیاد بوده و هنوز به مقدار ثابتی نرسیده است. پس هنوز می توان شبکه را آموزش بیشتری دهیم.

Recall حداکثر مقدار این معیار یک و یا ۱۰۰ درصد و حداقل مقدار آن صفر است و هرچه مواردی که ما انتظار داشتیم پیش بینی شوند ولی برنامه پیش بینی نکرده است که به آن False Negative می گوئیم نسبت به پیش بینی های درست یا True Positive بیشتر باشد مقدار Recall کمتر خواهد شد.

با توجه به نمودار آخر که شیب تند صعودی دارد به این معناست که برنامه جای بهبود زیادی دارد.

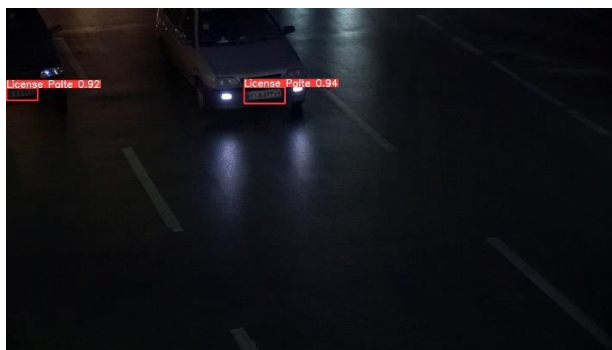


شکل (۳): YoloV5s - batch: 20, epochs: 300, image-size: 640

همانطور که در این شکل مشاهده می کنید، نمودار train/box_loss روند شیب نزولی آن تقریباً کاهش یافته و همچنین در نمودار metrics/precision و metrics/mAP_0.5 و recall روند شیب صعودی آن کاهش یافته مقدار ثابتی را طی کند و به این معنی است که مدل با آموزش بیشتر به افزایش دقت چشمگیری نخواهد رسید.

بهترین نتیجه در مدل small، به $mAP_{0.5} = 89.7\%$ داده های آموزش و $mAP_{0.5} = 89.1\%$ inference دست یافتیم. البته در هر سه آزمایشی که روی مدل small انجام شده، تفاوت بین mAP آموزش و mAP تست بسیار ناچیز است که این نشان دهنده ی آن است که شبکه به خوبی آموزش دیده است.

در شکل زیر، چندین تصویر را برای داده های تست که سعی بر این بوده که در شرایط مختلف تصویربرداری شده باشد، به مدل داده ایم.



شکل (۵)



شکل (۴)



شکل (۷)



شکل (۶)



شکل (۹)

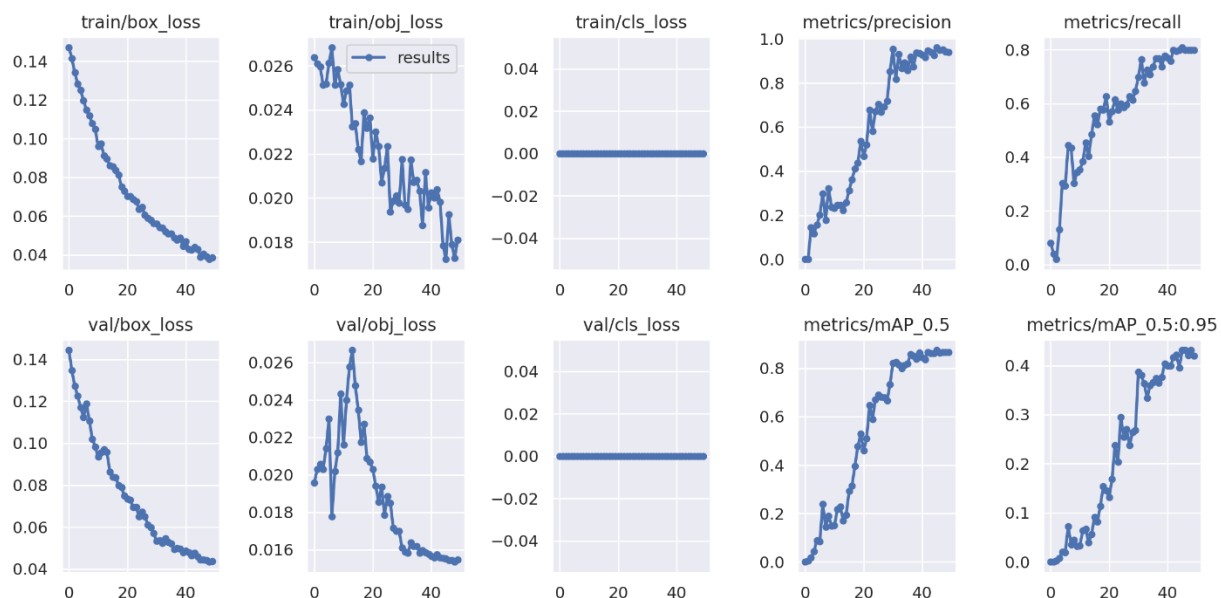


شکل (۸)

در شکل های (۴)، (۵) و (۶) با توجه به روشنایی روز و شب، مدل توانسته پلاک های موجود در تصویر را با دقت خوبی تشخیص دهد. اما در شکل های (۷)، (۸) و (۹) انتظار داشتیم که پلاک های بیشتری (که در فاصله ی دورتری از دوربین هستند) را تشخیص دهد. یا در شکل (۹) تصویری را به اشتباه پلاک تشخیص داده است. یکی از علت های ممکن این است که تصاویر به خوبی لیبیل گذاری نشده باشند. همچنین اگر تعداد تصاویر آموزش بیشتر باشد، مدل بهتر آموزش میبیند.

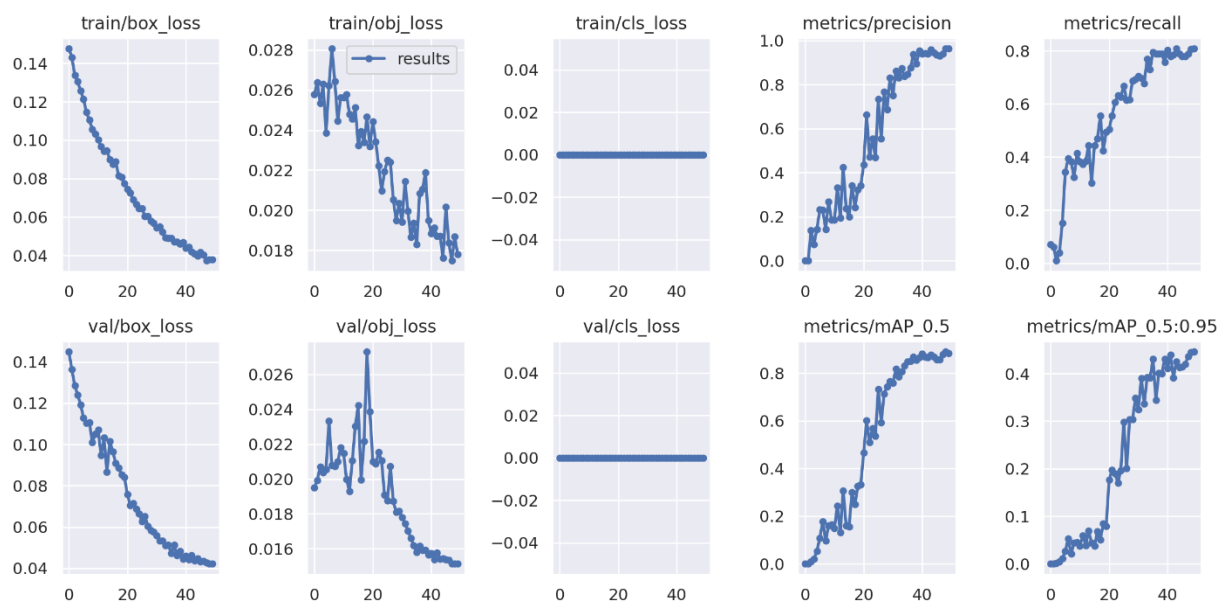
در جدول زیر مقادیر به همراه نتایج آن برای مدل medium آورده شده است.

Train	Batch= 25 Epochs= 50	Inference
wandb: best/epoch 45		Class: all
wandb: best/mAP_0.5 0.8745		Images: 40
wandb: best/mAP_0.5:0.95 0.43078		Instances: 99
wandb: best/precision 0.96034		P: 0.947
wandb: best/recall 0.80808		R: 0.788
wandb: metrics/mAP_0.5 0.87487		mAP50: 0.865
wandb: metrics/mAP_0.5:0.95 0.43094		mAP50-95: 0.417
wandb: metrics/precision 0.96029		
wandb: metrics/recall 0.80808		
Train	Batch= 30 Epochs= 50	Inference
wandb: best/epoch 48		Class: all
wandb: best/mAP_0.5 0.88968		Images: 40
wandb: best/mAP_0.5:0.95 0.44568		Instances: 99
wandb: best/precision 0.96377		P: 0.975
wandb: best/recall 0.80611		R: 0.804
wandb: metrics/mAP_0.5 0.88944		mAP50: 0.88
wandb: metrics/mAP_0.5:0.95 0.44454		mAP50-95: 0.43
wandb: metrics/precision 0.96377		
wandb: metrics/recall 0.80611		
Train	Batch= 30 Epochs= 100	Inference
wandb: best/epoch 96		Class: all
wandb: best/mAP_0.5 0.89522		Images: 40
wandb: best/mAP_0.5:0.95 0.47229		Instances: 99
wandb: best/precision 0.95487		P: 0.951
wandb: best/recall 0.85492		R: 0.848
wandb: metrics/mAP_0.5 0.89588		mAP50: 0.894
wandb: metrics/mAP_0.5:0.95 0.47534		mAP50-95: 0.458
wandb: metrics/precision 0.95488		
wandb: metrics/recall 0.85502		



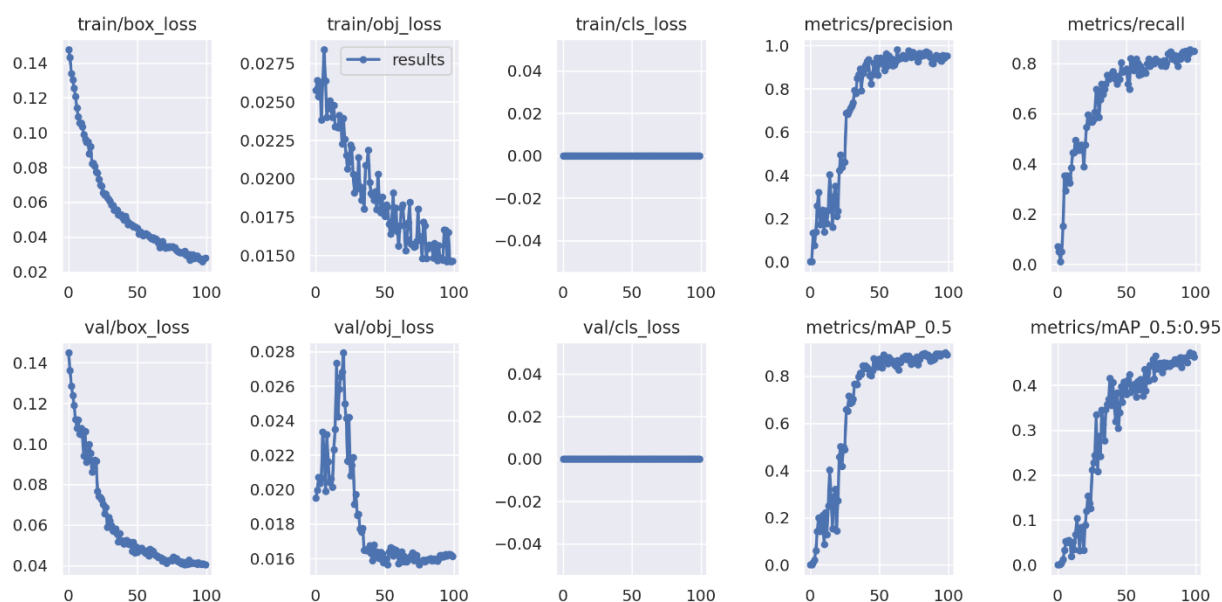
batch: 25, epochs: 50, image-size: 640 -YoloV5m : (۱۰) شکل

با توجه به نمودار اول از سمت چپ (train/box_loss) و (val/box_loss) که هر دو نمودار یک روند یکسانی را با هم به طور نزولی داشته اند؛ حاکی از این است که overfit اتفاق نیفتاده است. در نمودارهای چهارم و پنجم که نشان دهنده recall و precision هستند، میتوان گفت که شبکه را با آموزش های بیشتر به نتیجه بهتری رساند.



batch: 30, epochs: 50, image-size: 640 -YoloV5m : (۱۱) شکل

در این شکل با توجه به نمودار `metrics/precision` که تقریباً به یک شیب ثابتی رسیده است میتوان گفت که مدل به خوبی آموزش دیده است و شاید با تکرار بیشتر مقداری دقت مدل افزایش یابد.



شکل (۱۲): YoloV5m - batch: 30, epochs: 100, image-size: 640

بنابراین در این قسمت تعداد `epochs` را ۱۰۰ در نظر گرفته تا نتیجه را مشاهده کنیم. همانطور که میبینید نتیجه حدوداً ۱ درصد افزایش یافته است. ممکن بود اگر تعداد `epochs` را بیشتر از ۱۰۰ در نظر گرفته بودیم، نتیجه افزایش بیشتری می داشت.

بهترین نتیجه در مدل `medium`، به `mAP_0.5 = 89.5%` داده‌های آموزش و `mAP_0.5 = 89.4%` `inference` دست یافتیم. البته در هر سه آزمایشی که روی مدل `small` انجام شده، تفاوت بین `mAP` آموزش و `mAP` تست بسیار ناچیز است که این نشان دهنده ی آن است که شبکه به خوبی آموزش دیده است.

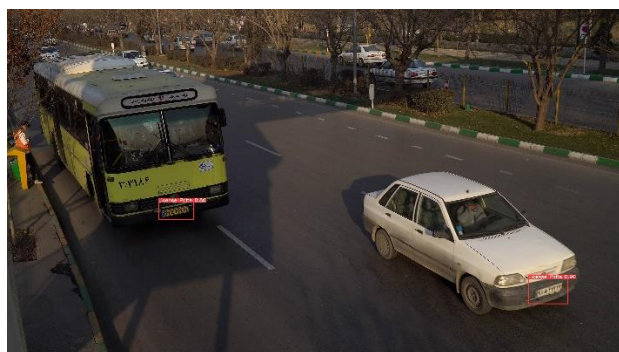
همان تصاویر تست که برای مدل Small را داده ایم به مدل medium می دهیم و نتایج در زیر آمده است.



شکل (۱۴)



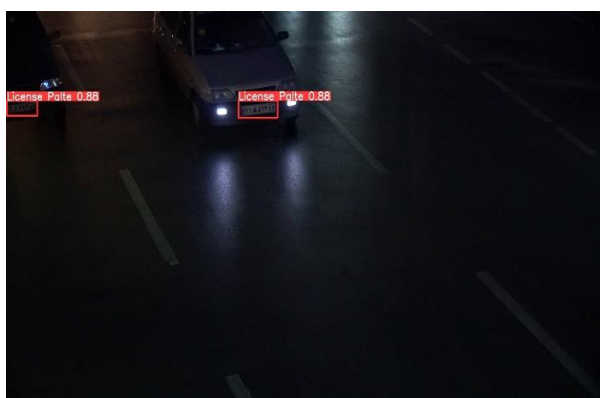
شکل (۱۳)



شکل (۱۶)



شکل (۱۵)



شکل (۱۸)

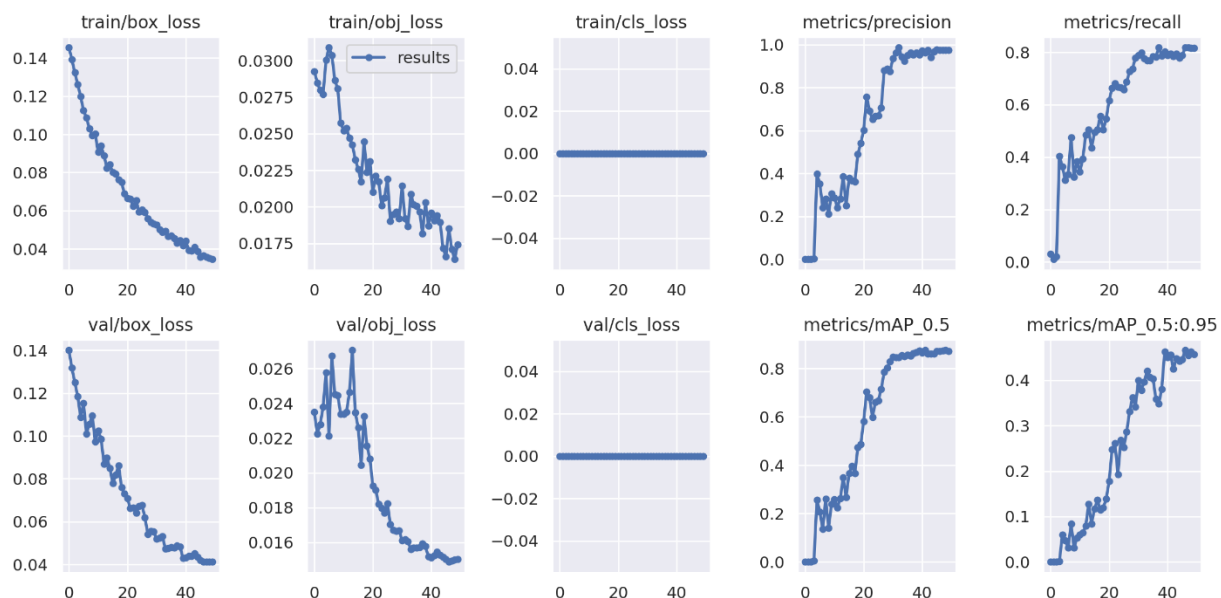


شکل (۱۷)

نتایج تقریباً مشابه با مدل Small بوده و همانطور که گفته شد، یکی از علت های اینکه نتوانسته همه ی پلاک های دور را به خوبی تشخیص دهد، لیبل گذاری نامناسب بوده است.

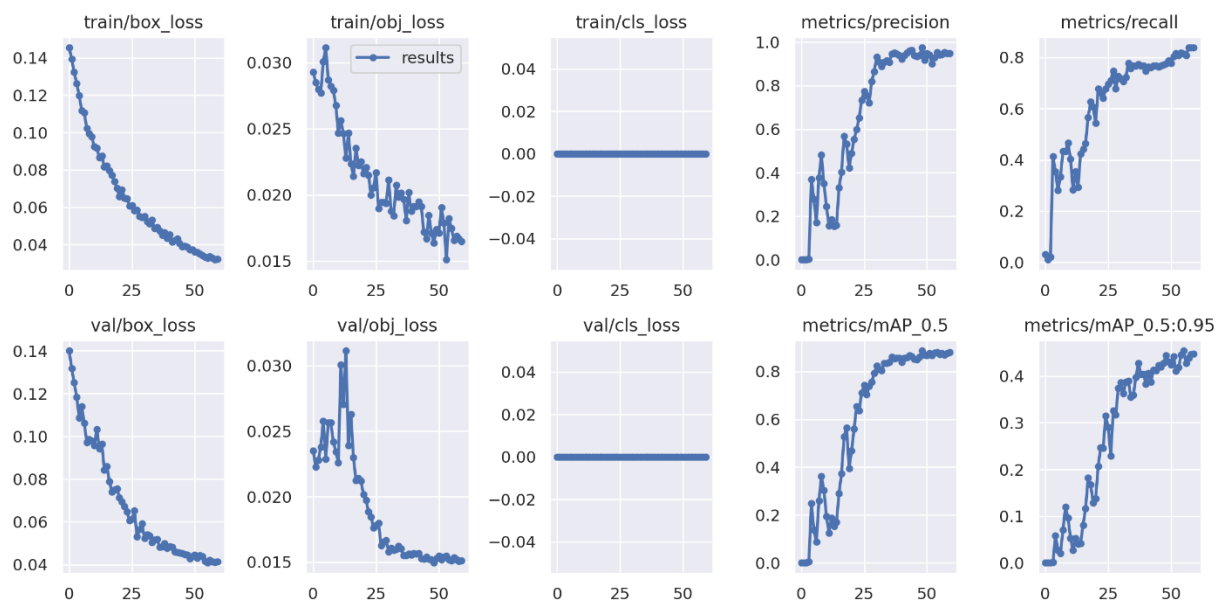
در جدول زیر مقادیر به همراه نتایج آن برای مدل Large آورده شده است.

Train	Batch= 25	Epochs= 50	Inference
wandb:	best/epoch 46		Class: all
wandb:	best/mAP_0.5 0.87		Images: 40
wandb:	best/mAP_0.5:0.95 0.46626		Instances: 99
wandb:	best/precision 0.9759		P: 0.955
wandb:	best/recall 0.81799		R: 0.818
wandb:	metrics/mAP_0.5 0.86983		mAP50: 0.874
wandb:	metrics/mAP_0.5:0.95 0.4673		mAP50-95: 0.457
wandb:	metrics/precision 0.97589		
wandb:	metrics/recall 0.81786		
Train	Batch= 25	Epochs= 60	Inference
wandb:	best/epoch 55		Class: all
wandb:	best/mAP_0.5 0.87149		Images: 40
wandb:	best/mAP_0.5:0.95 0.45448		Instances: 99
wandb:	best/precision 0.94163		P: 0.941
wandb:	best/recall 0.81478		R: 0.808
wandb:	metrics/mAP_0.5 0.87156		mAP50: 0.867
wandb:	metrics/mAP_0.5:0.95 0.45391		mAP50-95: 0.464
wandb:	metrics/precision 0.94162		
wandb:	metrics/recall 0.81466		
Train	Batch= 20	Epochs= 100	Inference
wandb:	best/epoch 88		Class: all
wandb:	best/mAP_0.5 0.89581		Images: 40
wandb:	best/mAP_0.5:0.95 0.48018		Instances: 99
wandb:	best/precision 0.93738		P: 0.963
wandb:	best/recall 0.81818		R: 0.785
wandb:	metrics/mAP_0.5 0.89577		mAP50: 0.883
wandb:	metrics/mAP_0.5:0.95 0.48081		mAP50-95: 0.477
wandb:	metrics/precision 0.9369		
wandb:	metrics/recall 0.81818		



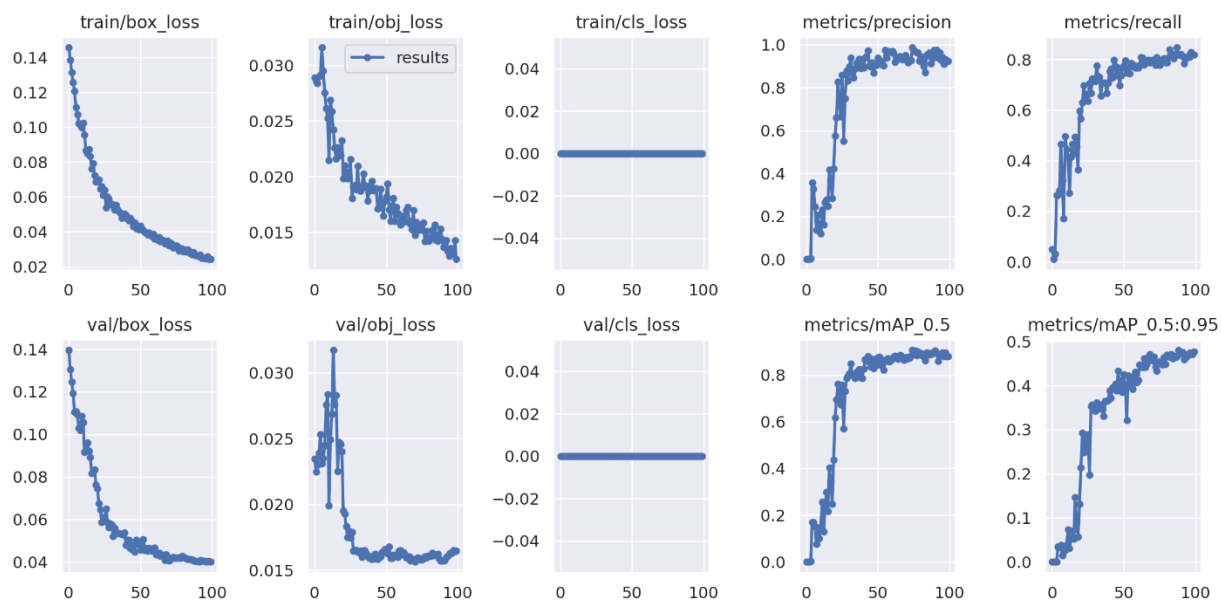
شکل (۱۹): YoloV5l - batch: 25, epochs: 50, image-size: 640

در این جا نیز، **overfit** اتفاق نیفتاده است زیرا همان طور که در نمودار اول از سمت چپ (**train/box_loss**) و (**val/box_loss**) مشاهده می کنید، هر دو نمودار یک روند یکسانی را با هم به طور نزولی داشته اند.



شکل (۲۰): YoloV5l - batch: 25, epochs: 60, image-size: 640

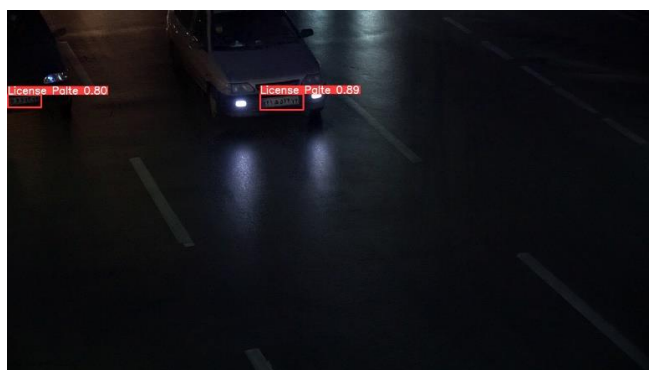
در این جا تعداد باری که شبکه آموزش ببیند را به مقدار ۶۰ رسانده ایم. اما همانطور که مشاهده میکنید تغییر محسوسی نداشته ایم.



شکل (۲۱): YOLOv5l - image-size: 640, epochs: 100, batch: 20

در اینجا هم تعداد epochs را بیشتر کرده و مقدار batch را کاهش داده ایم تا نتیجه را بررسی کنیم. در نمودار metrics/precision و metrics/mAP_0.5 و recall روند ثابتی پیدا کرده که نشانگر آن است که اگر تعداد epochs را باز هم بیشتر کنیم، افزایش چشم گیری نخواهیم داشت.

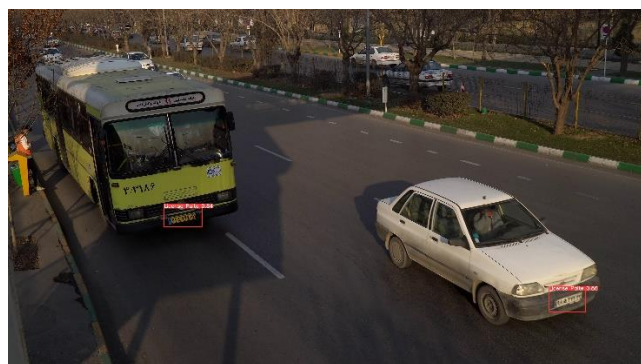
همان تصاویر تست به مدل large داده شده و نتایج در زیر آمده است.



شکل (۲۳)



شکل (۲۲)



شکل (۲۵)



شکل (۲۴)



شکل (۲۷)



شکل (۲۶)

همانطور که مشاهده می کنید، پلاک ها به درستی تشخیص داده شده اند، و شی دیگری به عنوان پلاک تشخیص داده نشده است. اما انتظاری که داشتیم این است پلاک ماشین هایی که از دوربین دور هستند، نیز تشخیص داده شود که در این مدل نیز این اتفاق نیفتاده است.

یک آزمایش دیگری را که بررسی کرده ایم تغییر سایز تصاویر بوده است. این آزمایش را در دو مدل medium و large انجام داده ایم. همان طور که مشاهده می کنید اندازه تصاویر در نتایج بسیار اهمیت داشته و با اندازه ۳۲۰، مقادیر mAP آموزش و تست کاهش چشمگیری داشته اند.

Train Model: Large Batch= 32 Epochs= 50 size= 320	Inference
wandb: best/epoch 49	Class: all
wandb: best/mAP_0.5 0.70703	Images: 40
wandb: best/mAP_0.5:0.95 0.3218	Instances: 99
wandb: best/precision 0.87963	P: 0.914
wandb: best/recall 0.62626	R: 0.626
wandb: metrics/mAP_0.5 0.70617	mAP50: 0.7
wandb: metrics/mAP_0.5:0.95 0.3234	mAP50-95: 0.322
wandb: metrics/precision 0.87914	
wandb: metrics/recall 0.62626	
Train Model: Medium Batch= 20 Epochs= 50 size= 320	Inference
wandb: best/epoch 46	Class: all
wandb: best/mAP_0.5 0.67194	Images: 40
wandb: best/mAP_0.5:0.95 0.30242	Instances: 99
wandb: best/precision 0.86994	P: 0.889
wandb: best/recall 0.59596	R: 0.565
wandb: metrics/mAP_0.5 0.67221	mAP50: 0.669
wandb: metrics/mAP_0.5:0.95 0.30343	mAP50-95: 0.298
wandb: metrics/precision 0.86994	
wandb: metrics/recall 0.59596	

با توجه به نتایج هر سه مدل که در بالا آورده شده اند، مقدار Best Precision هم هایلایت شده است. این معیار به معنای تعداد درست هایی که درست تشخیص داده شده اند، است و برای این مسئله؛ تشخیص پلاک، معیار مهمی است. البته mAP یعنی میانگین precision در تعداد باری که مدل آموزش یا استنتاج می شود، است که به همین سبب این معیار را برای نتیجه گیری استفاده کرده ایم.

بهترین نتیجه نیز به مدل large با $batch = 20$ و $epochs = 100$ بوده که مقدار mAP آموزش برابر با ۸۹,۵٪ و مقدار mAP استنتاج برابر با ۸۸,۳٪ بوده است. این که تفاوت بین mAP آموزش و تست ناچیز است به این معنی است که شبکه به خوبی آموزش دیده است. همچنین در آزمایش های انجام شده، بیش برآزش یا overfit اتفاق نیفتاده است.

اما برای اینکه به نتایج بهتری برسیم، چند راهکار را میتوانیم آزمایش کنیم. (۱) با توجه به نتایجی که به دست آوردیم و مشاهده کردیم که در هر سه مدل متاسفانه پلاک هایی که در فاصله دورتر بودند، شبکه آن ها را تشخیص نداده و این را میتوان نتیجه گرفت که لیبیل گذاری به خوبی انجام نشده است. یعنی اگر پلاک های دورتر را نیز در لیبیل گذاری قرار داده بودیم ممکن بود که نتایج بهتری را به دست بیاوریم. (۲) در یادگیری عمیق، هرچه تعداد تصاویر و شرایط مختلف را در داده های آموزش بگنجانیم، میتوانیم به نتایج بسیار بهتر و دقیق تری دست یابیم. برای این راهکار یا میتوانیم تعداد تصاویر را بیشتر در نظر بگیریم یا اینکه از Augmentation استفاده کنیم. (۳) همچنین برای post-processing نیز می توانیم از دو فاکتور $conf-thres$ و $iou-thres$ که به ترتیب confidence threshold و NMS IoU threshold هستند را تغییر دهیم تا مدل سختگیرانه تر آموزش ببیند.

لینک وزن ها به همراه نتایج حاصل از مدل large، $batch=20$ ، $epochs=100$ و $image-size=640$ (بهترین نتیجه در مقایسه با هر سه مدل) در زیر آمده است:

https://drive.google.com/drive/folders/148wyVodCdgtg-zC_XFTbjvCe2bYbtBuk?usp=share_link