دانشکده مهندسی کامپیوتر

شمارش تعداد افراد در اتوبوس به کمک بینایی کامپیوتر

پایان‌نامه برای دریافت درجه کارشناسی

در رشته مهندسی کامپیوتر گرایش هوش مصنوعی

بردیا کریمی زندی

استاد راهنما: دکتر محمدرضا محمدی

بهار ۹۹

تأییدیه‌ی هیأت داوران جلسه‌ی دفاع از پایان‌نامه/رساله

نام دانشکده: دانشکده مهندسی کامپیوتر

نام دانشجو: بردیا کریمی زندی

عنوان پایان‌نامه یا رساله: شمارش تعداد افراد در اتوبوس به کمک بینایی کامپیوتر

تاریخ دفاع:

رشته: مهندسی کامپیوتر

ردیف	سمت	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا مؤسسه	امضا
۱	استاد راهنما	محمدرضا محمدی	استادیار	دانشگاه علم و صنعت ایران	
۲	استاد راهنما				
۳	استاد مشاور				
۴	استاد مشاور				
۵	استاد مدعو خارجی				
۶	استاد مدعو خارجی				
۷	استاد مدعو داخلی			دانشگاه علم و صنعت ایران	
۸	استاد مدعو داخلی				

تأییدیه‌ی صحت و اصالت نتایج

باسمه تعالی

اینجانب بردیا کریمی زندی به شماره دانشجویی ۹۵۵۲۱۳۸۷ دانشجوی رشته مهندسی کامپیوتر گرایش هوش مصنوعی مقطع تحصیلی کارشناسی تأیید می‌نمایم که کلیه‌ی نتایج این پایان‌نامه/رساله حاصل کار اینجانب و بدون هرگونه دخل و تصرف است و موارد نسخه‌برداری‌شده از آثار دیگران را با ذکر کامل مشخصات منبع ذکر کرده‌ام. درصورت اثبات خلاف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم (قانون حمایت از حقوق مؤلفان و مصنفان و قانون ترجمه و تکثیر کتب و نشریات و آثار صوتی، ضوابط و مقررات آموزشی، پژوهشی و انضباطی ...) با اینجانب رفتار خواهد شد و حق هرگونه اعتراض درخصوص احقاق حقوق مکتسب و تشخیص و تعیین تخلف و مجازات را از خویش سلب می‌نمایم. در ضمن، مسئولیت هرگونه پاسخگویی به اشخاص اعم از حقیقی و حقوقی و مراجع ذی‌صلاح (اعم از اداری و قضایی) به عهده‌ی اینجانب خواهد بود و دانشگاه هیچ‌گونه مسئولیتی در این خصوص نخواهد داشت.

نام و نام خانوادگی:

امضا و تاریخ:

مجوز بهره‌برداری از پایان‌نامه

بهره‌برداری از این پایان‌نامه در چهارچوب مقررات کتابخانه و با توجه به محدودیتی که توسط استاد راهنما به شرح زیر تعیین می‌شود، بلامانع است:

☐ بهره‌برداری از این پایان‌نامه/ رساله برای همگان بلامانع است.

☐ بهره‌برداری از این پایان‌نامه/ رساله با اخذ مجوز از استاد راهنما، بلامانع است.

☐ بهره‌برداری از این پایان‌نامه/ رساله تا تاریخ ..... ممنوع است.

نام استاد یا اساتید راهنما:

تاریخ:

امضا:

چکیده

در دنیای امروز با توجه به رشد فناوری و به وجود آمدن شاخه‌ای جدید در علم هوش مصنوعی به نام یادگیری ماشین[۱] بسیاری از کارها به کمک ماشین‌ها در حال انجام هستند. یکی از زیرشاخه‌های یادگیری ماشین , بینایی ماشین[۲] نام دارد که به طور کلی برای پردازش عکس و فیلم می‌باشد.

شمارش افراد در حال حاضر به کمک زیر ساخت‌های سخت افزاری انجام می‌شود، برای مثال استفاده از کارت جهت ورود به یک ساختمان یا استفاده از اثر انگشت؛ حال اگر ما بخواهیم که از زیر ساخت‌های سخت افزاری کمتر استفاده کنیم آن هم به علت هزینه نگهداری و بعضا زمانگیر بودن آن‌ها؛ می‌توانیم از دوربین‌ها برای تشخیص افراد استفاده کنیم و به کمک بینایی ماشین این کار قابل انجام می‌باشد. این روش هم نیاز کمتری به سخت افزار دارد هم سرعت بالاتری دارد و مهم تر از همه انسان در آن‌ها دخیل نیست و کاملا اتوماتیک این کار انجام می‌شود.

در این پروژه سعی بر آن کردم روشی مناسب برای تشخیص افراد، هنگام داخل شدن و یا خارج شدن از اتوبوس پیدا کنم. شبکه‌های عمیق مختلف بر روی دیتاستی که خودم آن را تهیه کردم امتحان شده اند و نتایج آن‌ها مانند سرعت و دقت با یکدیگر مقایسه شده اند.

[۱] Machine learning

[۲] Computer vision

فهرست مطالب

فهرست شکل‌ها

فهرست جدول‌ها

فصل ۱ : مقدمه

۱-۱- مقدمه

با توجه به اهمیت فراهم بودن اطلاعات دقیق، مناسب و به روز برای کمک به تصمیم گیری مدیران و همچنین تهیه گزارشات مدیریتی برای سازمان‌های بالاسری ، وجود راهکارهای سیستمی برای جمع آوری و تحلیل اطلاعات ناوگان اتوبوس‌رانی مهم می‌شود. در حال حاضر مسافرانی که کارت بلیط خود را ثبت کنند به عنوان مسافر در سیستم درنظر گرفته می‌شوند ; اما در واقعیت در شهری مانند مشهد تمام مردم کارت خود را به همراه ندارند و سیستم به طور دقیق آمار افراد را نمی‌تواند گزارش کند.

به طور کلی برای شمارش افراد داخل اتوبوس راه حل‌هایی وجود دارد که هر کدام خوبی‌ها و بدی‌های خود را دارند. دو راه اصلی برای اینکار تشخیص چهره[۱] و تشخیص انسان[۲](بدن انسان) است. در سیستم مبتنی بر تشخیص چهره، باید دوربین در موقعیتی قرار گیرد که بتواند چهره ی فرد را به خوبی ثبت کند. این راه‌حل معمولا هزینه‌ی محاسباتی زیادی دارد و برای شمارش افراد یکتا باید به همراه سیستم مقایسه چهره استفاده شود. همچنین برای تشخیص نوع حرکت (ورود یا خروج) دو راه‌حل وجود دارد. راه‌حل اول استفاده از دو دوربین و راه حل دوم استفاده از سیستم رهگیری[۳] است . به کمک این سیستم و تخمین جهت حرکت فرد، ورود یا خروج شخص مشخص می‌گردد.

حال در این پروژه وظیفه بنده پیدا کردن بهترین راه‌حل برای شناسایی افراد در اتوبوس به کمک شبکه‌های عمیق[۴] مختلف بود که این مقاله چکیده‌ای بر پژوهش‌های انجام شده بنده می‌باشد.

---

۱ Face detection

۲ Object detection

۳ Object tracking

۴ Deep neural networks

فصل ۲ : مروری بر تحقیقات پیشین

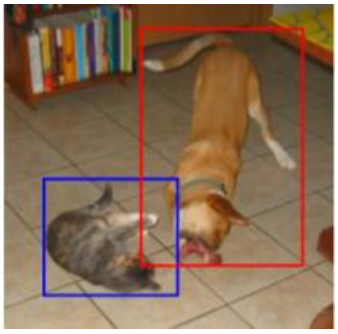
۲-۱- پژوهش‌های انجام شده در این زمینه

برای این کار بنده از شبکه‌های عصبی عمیق از قبل یاد گرفته[۱] استفاده کردم. شبکه‌های مختلفی برای تشخیص سریع انسان وجود دارند مانند یولو[۲] [۱]و فست آر سی ان ان [۲][۳] و همینطور اس اس دی[۴][۳] (شناساگر چند پنجره‌ای یک بار مشاهده) که هر کدام ویژگی‌های خاص خود را دارا هستند. بنده به بررسی چندین شبکه که به گفته‌ی مقالات آن‌ها برای شناسایی مناسب تر هستند پرداختم.

۲-۲- شبکه شناساگر اس اس دی

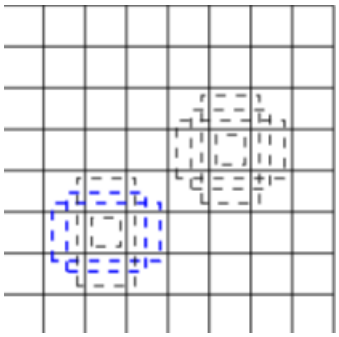
این شبکه سعی بر آن داشته‌است که بتواند سرعت پردازش و شناسایی را بر روی تصاویر افزایش دهد و در همین حال دقت آن را تا اندازه‌ی مناسب بالا ببرد.

نکته مهم آن است که این شبکه خود به تنهایی عمل شناسایی را انجام نمی‌دهد و فرایندی دارد که دانستن آن ضروری می‌باشد. خیلی از شبکه‌های سنتی بدین شکل بودند که تصویر از یک شبکه کانولوشنال[۵](پنجره لغزان) عبور می‌کند و سپس لایه آخر آن به یک شبکه تمام درگیر[۶] چسبانده می‌شد و جواب را بدست می‌آورد. در شبکه‌های شناساگر به جای اتصال به شبکه تمام درگیر در آخر شبکه، سعی می‌شود که نوع جسم شناخته شده و محل قرارگیری آن گزارش شود. پس در واقع بخش بزرگی از شناسایی به وسیله شبکه دیگری انجام می‌شود که آن را به اسم کلاسیفایر[۷] می‌شناسند.

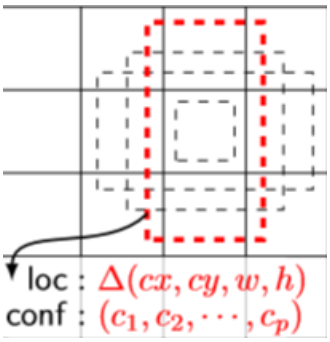


شکل ۱ عکس و پنجره‌های بدست آمده

[۱] Pre-trained networks
[۲] Yolo
[۳] Fast r-CNN
[۴] SSD (single shot multibox detector)
[۵] Convolutional
[۶] Fully-connected
[۷] Classifier

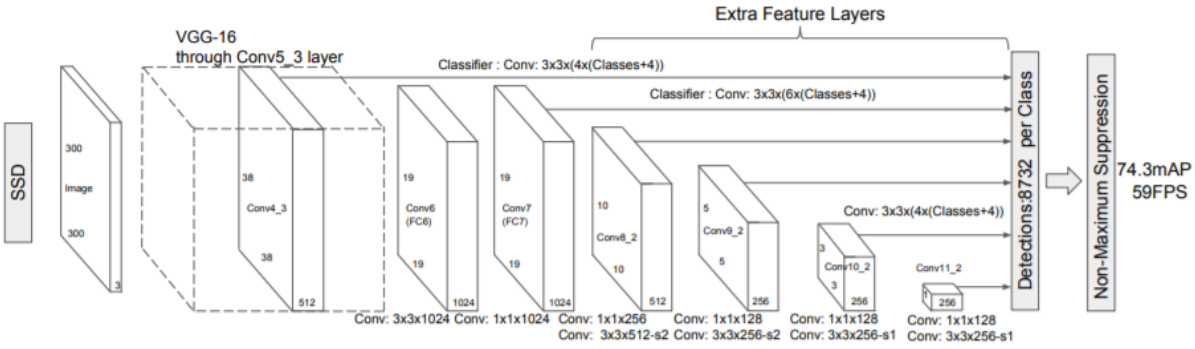


شکل ۲ نقشه ویژگی ۴*۴



شکل ۳ نقشه ویژگی ۸*۸

حال فرض کنید که یک شبکه کانولوشنال به اس اس دی چسبانده‌ایم و تصویر شماره‌ی ۱ نشان دهنده‌ی ورودی و جواب شناسایی‌ها می‌باشد. حال این شبکه نقشه ویژگی[۱] های ۴*۴ و ۸*۸(تصویرهای ۲ و ۳) را بعد از دیدن عکس تولید می‌کند که هر کدام جنبه‌های مختلفی را دیده‌اند و درصد اطمینان متفاوتی دارند و مقدار خطا را به دست می‌آورد و شبکه را آموزش می‌دهد. حال باید بدانیم این پنجره‌های بدست آمده از کجا تولید شده‌اند.



شکل ۴ معماری اس اس دی

[۱] Feature map

مطابق شکل یک شبکه کانولوشنال را در نظر می‌گیریم، این شبکه دارای چندین لایه‌ی پنجره‌لغزان می‌باشد که اس اس دی از این لایه‌ها خروجی می‌گیرد و به کمک خروجی آن‌ها نقشه‌های ویژگی را ایجاد می‌کند و به کمک آن‌ها جسم‌ها را شناسایی می‌کند. دقت اس اس دی نسبت به دیگر شبکه‌های شناسایی بالا نیست اما سرعت آن بسیار بالاتر می‌باشد برای همین از آن استفاده کردم.

شبکه کانولوشنال استفاده شده نیز بسیار مهم و تاثیر گذار در این امر می‌باشد که در قسمت‌های بعد به آن رسیدگی می‌کنیم.

۳-۲- شبکه موبایل‌نت اس اس دی

همانطور که پیشتر گفته‌شد شبکه اس اس دی [۳] نیاز به یک شبکه کانولوشنال برای تشخیص دارد، حال یک شبکه که بسیار سریع و با دقت خوب برای محاسبات سریع و همزمان می‌باشد شبکه موبایل‌نت[۱] [۴] می‌باشد. این شبکه هدف اصلی آن برای تلفن‌های همراه بوده است به همین خاطر وزن‌های آن زیاد نیست و سریع می‌باشد.

این شبکه نوع دیگری از لایه‌های کانولوشنال را به اسم کانولوشنال عمق نگر جداشدنی[۲] استفاده می‌کند. در حالات عادی فیلترهای کانولوشنال تمام عمق تصویر را با هم می‌بینند و خروجی ها را با هم میدهد اما در این مدل این دو مرحله از هم جدا میشوند و ابتدا به طور جدا فیلتر صورت می‌گیرد و در مرحله بعد خروجی‌ها با هم ترکیب می‌شوند. این باعث افزایش سرعت می‌شود.

کانولوشن معمولی :

$$K*K*M*N*F*F$$

کانولوشن عمق بین :

$$K*K*M*F*F + M*N*F*F$$

(K: سایز کرنل -M : عمق ورودی -N : سایز ورودی –F : سایز خروجی)

این شبکه نسخه‌های مختلفی دارد که من از نسخه شماره ۲ که در ۲۰۱۸ و توسط دیتاست کوکو[۳] آموزش‌داده شده‌است استفاده کردم.

---

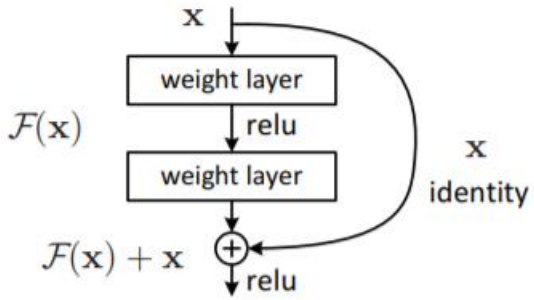
[۱] Mobile nets

[۲] Depth wise Separable Convolutional

[۳] COCO

۴-۲- شبکه رزنت اس اس دی

دیگر شبکه‌ای که برای شناسایی اجسام کمک کننده می‌باشد رزنت[۱] [۵] می‌باشد، با توجه به مطالعاتی که داشته‌ام این شبکه دقت بالاتری نسبت به دیگر شبکه‌ها دارد و دلیل آن هم وزن‌های بسار زیاد در این شبکه می‌باشد اما همین وزن‌ها باعث می‌شود که شبکه نسبت به دیگر شبکه‌های کوچکتر کندتر باشد.



شکل ۵ یادگیری باقی‌مانده‌ای

از دیگر نکات جالب و موثر درباره این شبکه استفاده از لایه‌های عقب تر برای ساختن لایه‌های رو به جلو می‌باشد که دقت را خیلی بالا برده است اما به تبع آن حجم شبکه نیز بسیار زیاد می‌باشد.شکل ۴ این مطلب را به وضوح بیان می‌کند.

۵-۲- شبکه افیشنت نت اس اس دی

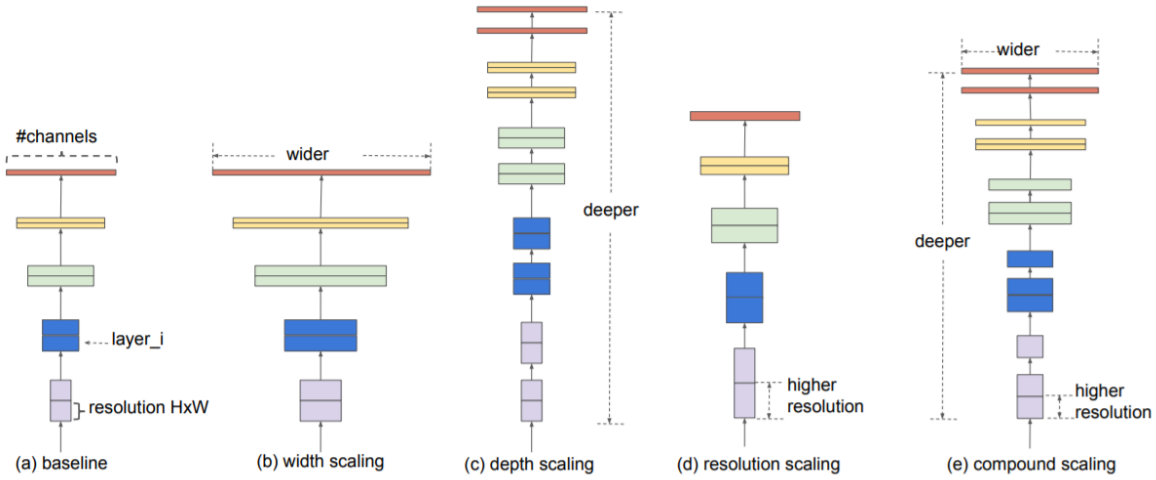
در قسمت بعد سعی بر امتحان کردن نوعی دیگر از شبکه‌های شناسایی کردم. شبکه‌ی افیشنت‌نت[۲][۶] یک شبکه‌ی جدید است که در سال ۲۰۱۹ معرفی شد این با شبکه ادعا می‌کند با اینکه سرعت بالایی دارد، دقت بالایی نیز دارد و سعی بر آن داشتم که این شبکه را هم امتحان کنم.

ساختار این شبکه بدین شکل است که ابتدا سعی بر افزایش موثر وزن‌های شبکه دارد، بدین شکل که یک شبکه پایه انتخاب می‌کند و بعد از آن به صورت موثر وزن‌ها در ۳ جهت عمق، عرض و دقت زیاد می‌کند، در نتیجه برای تقویت یک شبکه سعی بر اضافه کردن لایه‌ی جدید نمی‌کند برای همین دقت بالا می‌رود ولی سرعت خیلی کاهش نمی‌یابد.شکل پایین به همین مورد دلالت دارد.

---

[۱] Resnet

[۲] Efficient net



شکل ۶ بزرگ کردن شبکه به صورت موثر

حال آن‌ها یک شبکه پایه درست کردند و سعی بر افزایش موثر آن داشتند و به ۸ نوع شبکه مختلف رسیدند که از B0 تا B7 نام دارد که هر چه وزن‌ها زیاد شوند، دقت بالا می‌رود. بنده چون تصاویر ورودی ام ۳۰۰ * ۳۰۰ بود از B3 برای یادگیری استفاده کردم و این شبکه را به عنوان شبکه کانولوشنال برای اس اس دی قرار دادم.

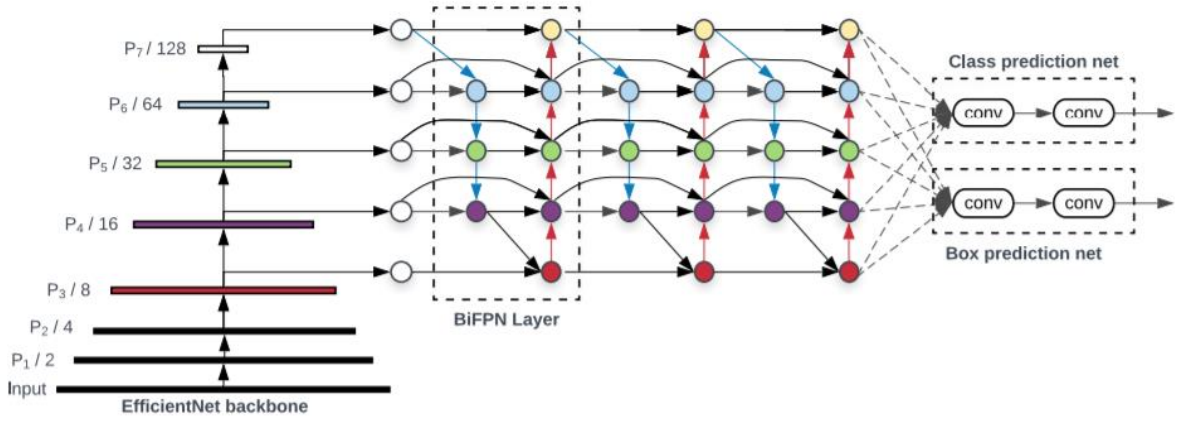
۶-۲- شبکه افیشنت‌دت

شبکه افیشنت‌دت[۱][۷] مانند شبکه موبایل‌نت اس اس دی دارای دو بخش می‌باشد و به طور کلی برای شناسایی ۲ کار مهم را انجام می‌دهد. ۱: نشان دادن ویژگی ها در چند مقیاس ۲: افزایش مقیاس شبکه به صورت بهینه.

مورد دوم که مربوط به شبکه اصلی و کلاسیفایر می‌باشد که این شبکه شبکه افیشنت‌نت را به عنوان بک‌بون[۲] استفاده می‌کند. اما مورد اول مبحثی جدید می‌باشد.

[۱] Efficientdet

[۲] Backbone



شکل ۷ تصویر معماری افیشنت‌نت

مطابق شکل ۷ کاری که این شبکه انجام می‌دهد این است که خروجی لایه‌های کانولوشن شبکه را از افیشنت‌دت میگیرد و در مرحله بعد از آن‌ها استخراج ویژگی انجام می‌دهد. لایه‌های BiFPN بدین صورت می‌باشد که به صورت وزن دار و همچنین به صورت ۲ طرفه در هر مرحله ویژگی لایه ها با هم ترکیب می‌شوند و به صورت لایه‌های کانولوشنال با هم ترکیب می‌شوند. این ترکیب نیز به ۳ عامل اصلی وابسته می‌باشد : ۱: حتما از ورودی اصلی خروجی داریم ۲: در لایه‌های میانی هم رو به پایین استخراج ویژگی میکنیم هم رو به بالا ۳: این استخراج ویژگی‌ها به صورت دو طرفه می‌باشد(در مدل های قدیمی مانند FPN همه به صورت تماما متصل و رو به جلو بودند یا در PANet یک دفعه رو به پایین و دفعه دیگر رو به بالا).طبق ادعای این شبکه تا سال ۲۰۱۹ از دیگر شبکه‌های سریع دقت بالاتری داشت.

۷-۲- شبکه یولو

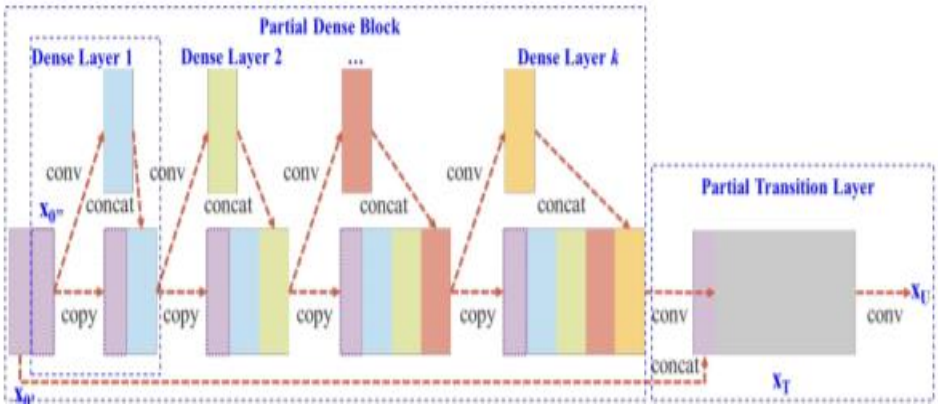
شبکه یولو از دیگر شبکه‌های سریع برای تشخیص می‌باشد که ورژن‌های مختلفی دارد. سال ۲۰۱۹ افیشنت‌نت پیشتاز شبکه‌های شناسایی بود اما در ماه جوت در سال ۲۰۲۰ ورژن شماره ۵ یولو از همه‌ی شبکه‌ها پیشی گرفت و در حال حاضر بهترین شبکه در این زمینه می‌باشد.

این شبکه شامل ۳ قسمت اصلی می‌باشد: ۱: بک‌بون[۱] ۲: نک[۲] ۳:هد[۳]

قسمت اصلی آن که کلاسیفایر آن می‌باشد شبکه‌ای به اسم سی اس پی نت[۴][۸] می‌باشد. این شبکه باعث بالا بردن سرعت در یک شبکه‌ی عمیق شده است و این شبکه نیز در سال ۲۰۲۰ عرضه شده است.

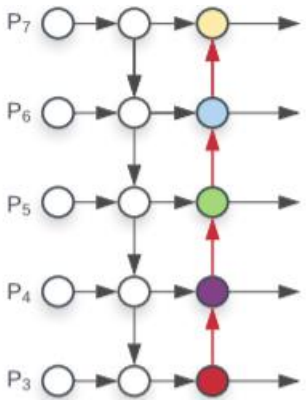
---

[۱] Backbone
[۲] Neck
[۳] Head
[۴] CSPNet



شکل ۸ معماری سی اس پی نت

همانور که در شکل ۸ قابل ملاحظه می‌باشد در هر قسمت نقشه ویژگی یک مرحله به مرحله بعد منتقل می‌شود و یک دفعه هم در لایه دنس مشارکت می‌کند. این کار باعث کاهش ممووری و بالانس شدن هزینه محاسبه و همچنین باعث افزایش مسیر گرادیان(افزایش دقت سی ان ان) می‌شود. در آخر نیز ویژگی‌ها استخراج می‌شود که در آن از ورودی اولیه کمک گرفته می‌شود، در مدل‌های قبلی مانند Dense Net از ورودی برای استخراج ویژگی استفاده نمی‌شد.



شکل ۹ مدل PANet

در قسمت میانی آن مانند افیشنت‌نت باید از یک الگوریتمی برای استخراج ویژگی استفاده کرد که در ورژن ۵ از PANet استفاده می‌شود که در شکل ۹ قابل دیدن می‌باشد.

قسمت آخر مربوط به باکس‌های شناسایی می‌باشد که در واقع قسمت پایانی شناسایی می‌باشد.

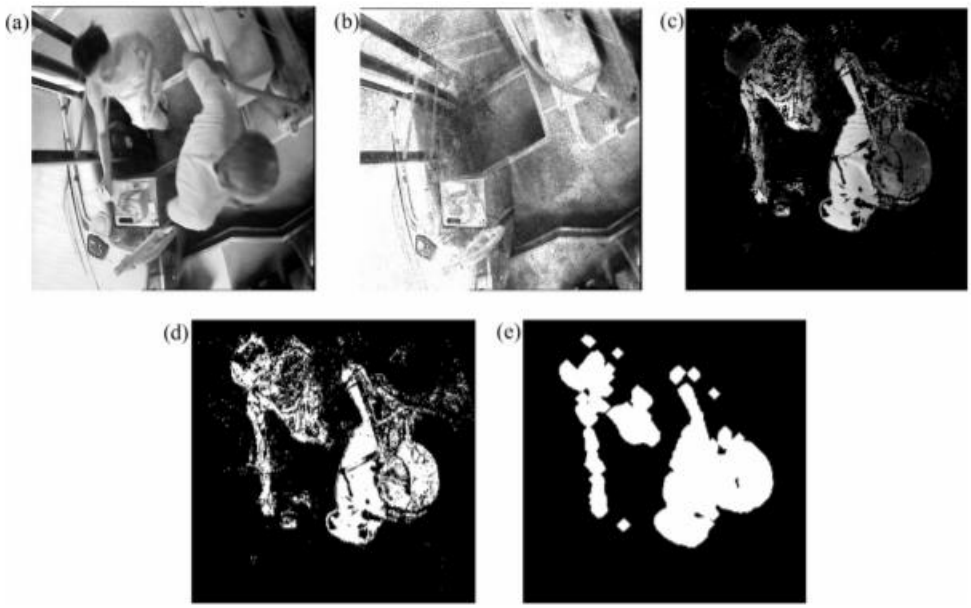
۸-۲- کارهای انجام شده توسط دیگران

بعد از شناسایی انواع شبکه‌ها نگاهی به کارهای دیگر افراد متخصص برای حل مسئله‌ای شبیه به این کردم و از آن‌ها به اطلاعات گوناگونی دست پیدا کردم. طبق [۹] شبکه‌های اس اس دی نسبت به شبکه‌های دیگر مانند فستر آر سی ان ان[۲] بهتر هستند برای شناسایی همزمان و در این مقاله محل قرارگیری دوربین فرق داشت و با توجه به پوشش مردم، آن‌ها از شناسایی صورت استفاده کردند. مورد دیگر در این مقاله این بود که با توجه به دیتاست بزرگی که داشتند توانستند حالات در روز، شب و حتی هوای بارانی را امتحان کنند و به نتایج مطلوبی برسند.

شکل ۱۰ نمونه قرار گیری دوربین در یکی از مقالات

همانطور که در شکل شماره ۱۰ پیدا می‌باشد، دوربین کاملا از بالا قرار ندارد و در این مقاله علاوه بر صورت، از تشخیص کلاه نیز برای انجام این کار استفاده کردند.

در [۱۰] راه‌حل دیگری به کار برده شد و آن نیز به علت ثابت ماندن دوربین در اتوبوس بود. در این روش آن‌ها عکسی را که در آن مسافری نیست را در نظر گرفتند و فضای اتوبوس را مشخص کردند؛ در مرحله ی بعد هرگاه مسافر وارد تصویر می‌شد، تصویر آن را جدا می‌کردند و یک تصویر سیاه و سفید که فقط شامل مسافر هست را به شبکه می‌دادند. این روش به خوبی از روش‌های سنتی در بینایی کامپیوتر مانند لبه‌یابی و تشخیص رنگ نیز استفاده کرد.



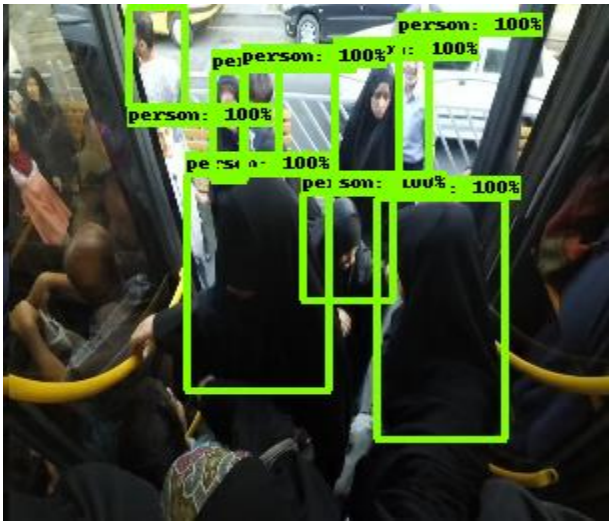
شکل ۱۱ نمونه‌ای از مکان‌یابی مسافر در اتوبوس

همانطور که در شکل ۱۱ پیدا می‌باشد، در این روش تنها پیکسل‌های سفید انسان‌ها می‌باشند که تصویر خروجی را به شبکه کانولوشال می‌دهند. این‌کار باعث کاهش محاسبات و به طبع آن افزایش سرعت می‌شود.

فصل ۳ : روش تحقیق

۱-۳- ایجاد دیتاست

واضح است که برای یادگیری ماشین یکی از مهم‌ترین نکات استفاده از یک دیتاست مناسب می‌باشد که با آن شناسایی را به درستی انجام بدهیم. لازم به ذکر است که این کار به هیچ وجه آسان نبود چون ابتدا باید فریم ها را از فیلم جدا می‌کردم و سپس آن‌ها که مناسب برای یادگیری بودند را برچسب‌گذاری می‌کردم. متسفانه این‌کار را مجبور شدم ۲ بار انجام بدهم زیرا مرتبه اول به دلیل عدم تجربه اندازه فریم ها درست نبودند و شبکه نمی‌توانست درست کار کند. در کل برای آموزش ۲۶۰ عکس که البته نزدیک به ۴۰ تای آن‌ها نیز عکس انسان بودند ولی مربوط به شناسایی افراد در اتوبوس نبودند، دلیل آن هم کم بودن حجم دیتاست بود. همچنین ۴۷ تا عکس که با عکس‌های یادگیری متفاوت بودند برای ارزیابی استفاده‌شد.



شکل ۱۲ تصویر برچسب‌گذاری شده

به کمک نرم افزار لیبل‌می[۱] موقعیت تک‌تک مسافران را پیدا کردم و آن را به فرمت سی اس وی[۲] در آوردم.

حال برای شبکه‌هایی که اس اس دی به عنوان استخراج ویژگی در آن‌ها استفاده می‌شد از فرمت تی اف رکورد[۳] که مخصوص پردازش در فریمورک تنسورفلو می‌باشد استفاده کردم .

برای شبکه‌ی یولو، چون انحصارا در فریم‌ورک پای‌تورچ[۴] بود باید آن‌ها در وبسایت رب‌فلو[۵] بارگذاری می‌کردم و آنها در در نوتبوک گوگل استفاده می‌کردم.

---

[۱] LabelMe
[۲] csv
[۳] TFRecord
[۴] PyTorch
[۵] Roboflow

۲-۳- پیدا کردن راه‌حل اولیه

برای یافتن بهترین راه ابتدا باید به داده‌های جمع‌آوری شده توسط دوربین‌ها توجه کرد. ویژگی‌های زیادی در هر تصویر وجود دارد که باید به آن‌ها با دقت توجه کرد.

شکل ۱۳ درب زنانه ساعت شلوغی

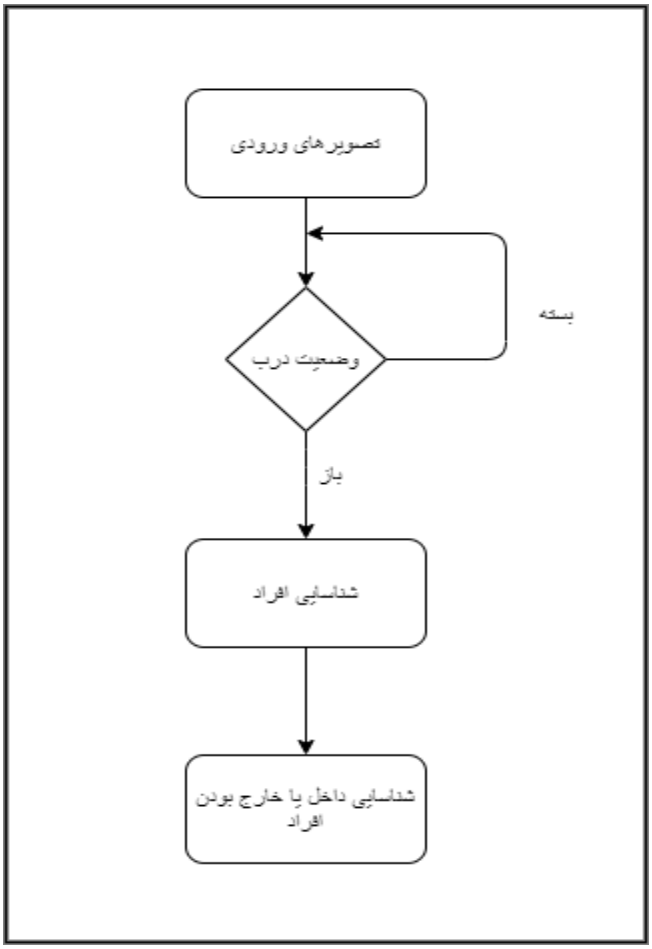
شکل ۱۴ درب زنانه ساعت خلوت

شکل ۱۵ درب مردانه

شکل ۱۳ نشان دهنده تصویر ثبت شده در ساعات شلوغی می‌باشد. همچنین شکل ۱۴ تصویر ثبت شده در هنگام خلوت بودن اتوبوس می‌باشد. همانطور که قابل ملاحظه می‌باشد چه در هنگام شلوغی و چه در هنگام خلوت بودن در قسمت زنانه تشخیص چهره خانم‌ها ناممکن می‌باشد و دلیل آن‌هم پوشش آن‌ها می‌باشد. همچنین با اینکه در تصویر شماره ۱۵ چهره واضح‌تر می‌باشد اما بازهم اگر سر شخص رو به پایین باشد، چهره‌ی او برای دوربین مشخص نمی‌باشد.

۳-۳- مسیر کلی حل مسئله

یکی از اصلی ترین سرچشمه‌های من یک پژوهش در همین زمینه بود. به کمک[۹] یک مسیر کلی برای حل این پژوهش پیدا کردم.



شکل ۱۶ الگوریتم پیشنهادی برای این پژوهش

طبق شکل شماره ۱۶ یک مسیر با ۳ قسمت اصلی پیشرو داریم که یک به یک آن‌ها را به انجام رساندم.

۴-۳- چگونگی درب

وقتی این سامانه بخواهد در بلند مدت به کار برود، مصرف انرژی را باید بتواند کنترل کند. به همین منظور دلیلی وجود ندارد که وقتی درب بسته می‌باشد، سامانه شناسایی را انجام دهد. همچنین برای این که توان محاسباتی بالایی نیاز داریم و این که در بیشتر زمان‌ها درب بسته می‌باشد، می‌توان تصاویر را ذخیره کرد و هنگامی که اتوبوس در حال حرکت می‌باشد شمارش افراد را انجام داد.

برای اینکار بهترین مورد، استفاده از قسمت بالای درب به جهت می‌باشد؛ به این علت اینکه در هنگام باز و بسته شدن درب، مسافری آن قسمت از درب را اشغال نمی‌کند. نکته‌ی اصلی ماجرا این است که باید از تاثیرگذاری نور خورشید غافل نشویم، چون اگر بخواهیم بررسی کنیم که آن قسمت تاریک می‌باشد یا خیر، نور خورشید ممکن است الگوریتم را دچار خطا کند.

شکل ۱۷ لبه‌های پیدا شده توسط لبه‌یاب کنی

شکل ۱۸خط‌های پیدا شده بعد از اعمال تبدیل هاف

من هم از الگوریتم خودم که صرفا چک کردن چندین پیکسل می‌باشد استفاده کردم که در این روش از هیچ‌کدام از الگوریتم‌های بینایی کامپیوتر استفاده نکردم و این روش تنها برای این دو دوربین می‌باشد و به صورت اتوماتیک عمل نمی‌کند.

روش دوم استفاده از روش‌های بینایی کامپیوتر می‌باشد. بدین صورت که ابتدا به کمک لبه‌یاب کنی[۱] سعی در پیدا کردن لبه‌ها کردم که در شکل ۱۷ نشان داده شده‌است و سپس به کمک تبدیل هاف[۲] سعی در پیدا کردن خط‌های عمودی درب‌ها کردم. در این الگوریتم چند نکته وجود دارد، ابتدا اینکه نور خورشید بسیار بر دقت الگوریتم موثر می‌باشد، به فرض مثال همانطور که در شکل ۱۸ قابل ملاحظه می‌باشد به علت انرژی متفاوت پیکسل‌ها قسمت‌های بالایی تشخیص داده نشده‌اند. مورد بعد اینکه بر خلاف الگوریتم قبلی این الگوریتم به صورت کاملا خودکار عمل می‌کند، یعنی با جابه‌جایی دوربین نیازی به تغییری در کد یا به اصطلاح کالیبره‌کردن الگوریتم نمی‌باشد. هر چند این الگوریتم به علت پیچیدگی‌هایی که دارد از الگوریتم قبلی کندتر می‌باشد.

۳-۵- شناسایی انسان

قسمت اصلی بخش شناسایی انسان بود که من باید شبکه‌هایی را که درباره آن‌ها تحقیق کردم را پیدا سازی می‌کردم. مشکلات و زحمات زیادی در هر بخش بود که به آن‌ها می‌پردازیم. برای این بخش کارها را به دو فریم‌ورک تقسیم می‌کنیم.

۳-۵-۱ تنسورفلو

به کمک این فریم‌ورک شبکه‌های موبایل‌نت اس اس دی و رزنت اس اس دی را آموزش دادم. هر دو این شبکه‌ها در این فریم‌ورک پیاده سازی شده‌اند و آموزش آنها به شرح زیر می‌باشد. ابتدا باید یک شبکه از پیش آموزش داده شده را داشته باشیم که فرمت ذخیره آنها مخصوص تنسورفلو باشند، این مدل گراف منجمد شده نام دارد. در داخل این گراف یک فایل داریم که می‌توان تعداد کلاس‌ها(که برای ما یک کلاس آدم بود) ، هایپر پارامترها، مسیر دیتای آموزش و آزمون، تعداد سمپل‌ها و حتی اضافه کردن دیتا[۳] را در آن‌ها مشخص کرد.

---

[۱] Canny

[۲] Hough

[۳] Data augmentation

شبکه از پیش آموزش دیده نیز در صفحه گیت‌هاب[۱] تنسورفلو وجود دارد که آن‌ها در گوگل می‌توان بارگذاری کرد. آموزش شبکه‌ها نسبتا طولانی می‌باشد اما به کمک تنسوربرد[۲] روند انجام کار و دقت‌ها قابل مشاهده می‌باشد.

اما برای شبکه افیشنت‌نت اس اس دی مراحل کار دشوارتر بود زیرا افیشنت‌نت در این فریم‌ورک پیاده سازی نشده است و مجبور بودم از ایتدا خودم تمامی فاز پیاده‌سازی را انجام می‌دادم. بعد از نزدیک به ۲ هفته تلاش و در حالی که نزدیک به پیاده‌سازی بودم، در یک صفحه گیت‌هاب پیاده سازی نصفه نیمه‌ای از آن بود و با هر زحمتی از آن استفاده کردم. اما مشکل دیگری نیز وجود داشت و آن نبود یک چک‌پوینت مناسب که بر روی دیتاست کوکو آموزش دیده باشد بود و تنها یک شبکه بود که بر روی ایمیج‌نت[۳] آموزش دیده شده بود که دیتاستی مناسب برای شناسایی نیست.

۲-۵-۳ پای تورچ

از این فریم ورک برای یولو[۱] ورژن ۵ استفاده کردم که بسیار از تنسورفلو مفیدتر بود و تقریبا تمامی کارها حتی امتحان کردن شبکه بر روی فیلم را خودش انجام می‌دهد. این شبکه نیز یک نوت‌بوک جدا دارد و همه‌ی کارها توضیح داده شده‌است. همچنین به کمک رب‌فلو که پیش از این اشاره کرده بودم، عملیات افزایش دیتا را بر روی آن وبسایت انجام دادم.

شبکه ی یولو[۱] ۴ نوع مختلف دارد که تفاوت آن‌ها در تعداد وزن‌ها می‌باشد که همه‌ی آن‌ها نیز مورد بررسی قرار گرفتند.

لازم به ذکر است که تمامی کارها بر روی سرویس گوگل‌کولب[۴] و به کمک جی پی یو[۵] انجام شده‌اند و تمامی گزارش‌ها آن به فرمت تی اف ایونت[۶] قابل دسترس می‌باشند که بر روی صفحه گیت‌هاب بنده وجود دارند.

---

[۱] GitHub
[۲] Tensor board
[۳] ImageNet
[۴] Google colab
[۵] GPU
[۶] TFEvents

۶-۳- دیگر کارهای انجام شده

یکی دیگر از کارهای انجام شده ارائه الگوریتمی بود که تشخیص بدهد مسافر شناسایی شده داخل اتوبوس قرار دارد یا خارج آن. این کار بدین منظور بود که به مرحله ردیابی کمک کند.

برای انجام این کار از روش‌های یادگیری نمی‌شد استفاده کرد به علت پیچیدگی و عدم داشتن دیتاست پس تصمیم بر آن شد که از روش‌های سنتی استفاده شود یعنی با در نظر گرفتن مختصات فرد و ایجاد ارتباط میان اندازه‌ی فرد و نزدیک بودن یا دور بودن نسبت به درب‌ها و دوربین، داخل یا خارج بودن آن را مشخص کنیم.

به علت آنکه معیاری در فضای پیوسته قابل دسترس نبود نمی‌توان دقت را محاسبه کرد اما می‌توان اندازه‌گیری بصری متوجه شد که این الگوریتم در زمان هایی که شناساگر به درستی ابعاد شخص را پیدا کند، جواب درست را مشخص می‌کند. پس یعنی دقت شبکه و به خصوص دقت مکان‌یابی آن تاثیر مستقیم بر این الگوریتم دارد.در پایین نمونه‌هایی از این کار را می‌توانید مشاهده کنید.(آبی برای داخل و سبز برای قرارگیری در خارج)

شکل ۱۹ نمونه اول از تشخیص مکان مسافر

شکل ۲۰ نمونه دوم از تشخیص مکان مسافر

شکل ۲۱ نمونه سوم از تشخیص مکان مسافر

فصل ۴ : نتایج آزمایش‌ها و تفسیر آنها

۱-۴- نتایج چگونگی درب

نخست به علت قرارگیری متفاوت دوربین در دو درب زنانه و مردانه به دو راه‌حل مجزا نیاز داشتیم. همچنین بنده از هر دو الگوریتم پیشنهادی در قسمت قبل استفاده کردم و نتایج آن‌ها را بدست آوردم.

جدول ۱ دقت اندازه گیری برای چگونگی درب به کمک الگوریتم پیشنهادی

نوع درب	تعداد تصاویر	درصد درستی
زنانه	۳۲۱	۹۸.۴۴
مردانه	۱۰۹۸	۹۶.۵۳

جدول ۲ دقت اندازه گیری برای چگونگی درب توسط بینایی ماشین

نوع درب	تعداد تصاویر	درصد درستی
زنانه	۳۲۱	۹۰.۵
مردانه	۱۰۹۸	۹۱.۶۲

با توجه به مقدارهای بدست آمده در جدول‌های شماره ۱و۲ می‌توان به این نتیجه رسید که تقریبا هر دو راه حل مسئله را حل می‌کنند اما الگوریتمی که بنده برای صرفا همین مسئله ارائه کردم جواب بهتری می‌دهد اما بدی آن خودکار نبودن آن می‌باشد.

۲-۴- نتایج شناسایی انسان

در مبحث شناسایی اشیا عموما برای شناسایی دقت الگوریتم از مکان‌یابی اشتراک به اجتماع[۱] استفاده می‌کنند بدین معنی که محدوده‌ای که شبکه برای عملیات شناسایی پیدا کرده چقدر با جواب مسئله تطابق دارد و طبق آن درستی یا غلط بودن جواب را در نظر می‌گیریم. مکان‌یابی اشتراک به اجتماع انواع مختلفی دارد مثلا ۵۰٪ درستی، یعنی اگر بیش از ۵۰٪ شناسایی با جواب تطابق داشت جواب را درست ارزیابی می‌کنیم.

حال برای ارزیابی از دقت[۲] و یادآوری[۳] استفاده می‌شود.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

بدین صورت که برای هر کلاس دقت را ارزیابی می‌کنیم (در این مسئله تنها یک کلاس داریم) و سپس دقت را بر اساس یادآوری بدست می‌آوریم و انتگرال این نمودار به عنوان میانگین دقت ارزیابی می‌شود و این عدد را برای تمامی کلاس‌ها مقایسه می‌کنیم و سپس میانگین می‌گیریم.

$$AveragePrecision = \int_0^1 p(r)dr$$

$$MeanAp = \frac{\sum Ap}{classes}$$

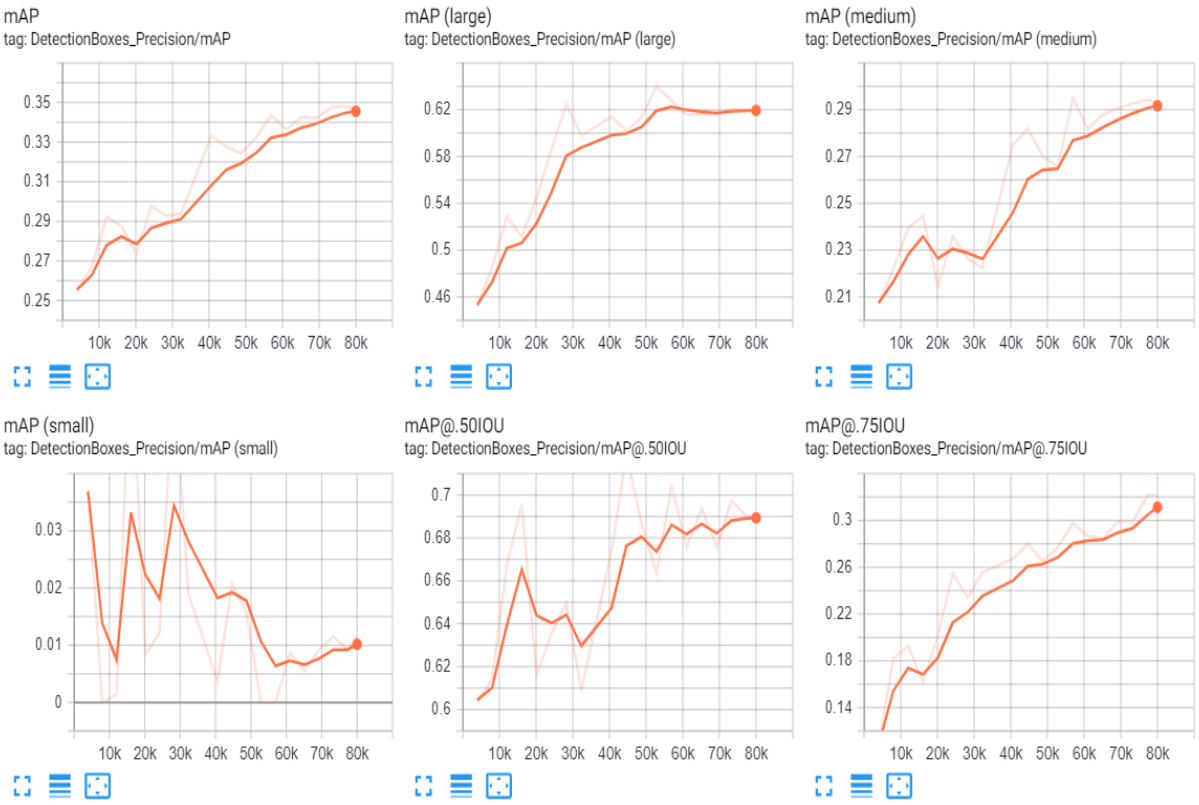
[۱] IoU

[۲] precision

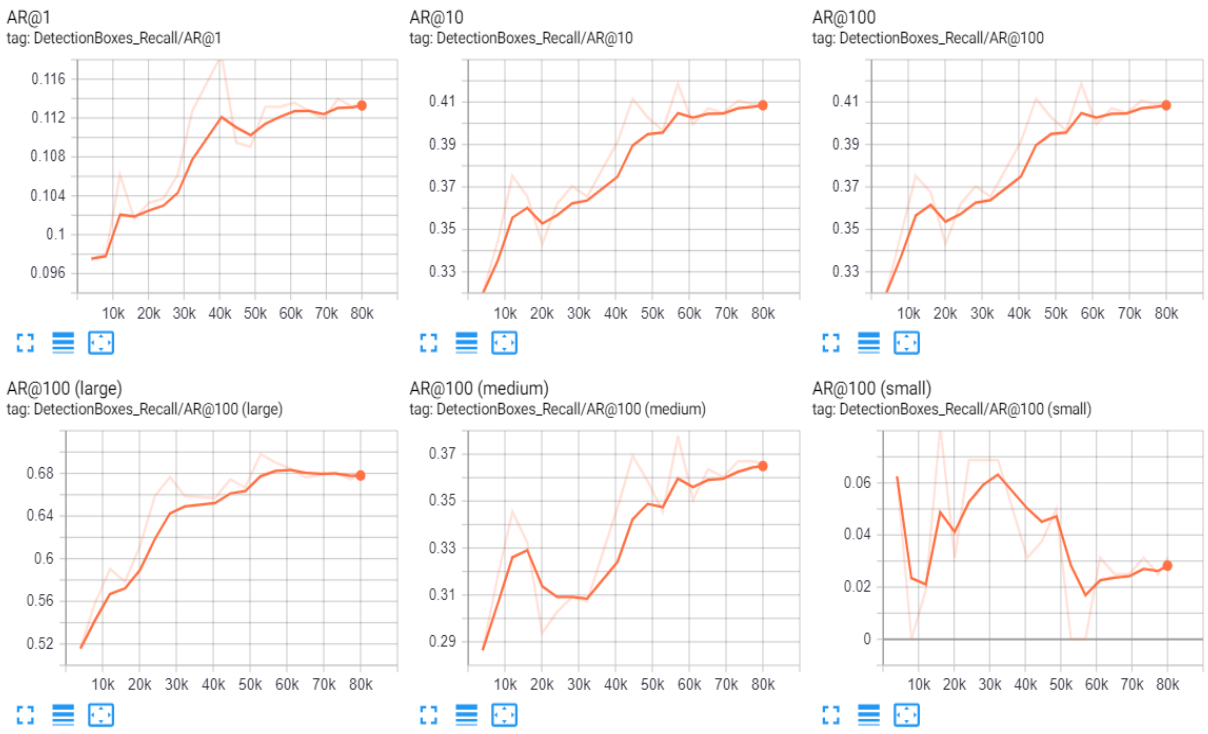
[۳] recall

۱-۲-۴- نتایج روی شبکه موبایل‌نت اس اس دی

این شبکه چون نسبت به شبکه‌های دیگر سبک‌تر می‌باشد، راحت تر آموزش دیده است و زمان آموزش آن از بقیه نیز کمتر می‌باشد.



شکل ۲۲میانگین درستی برای موبایل‌نت اس اس دی



شکل ۲۳ میانگین یادآوری در موبایل‌نت اس اس دی

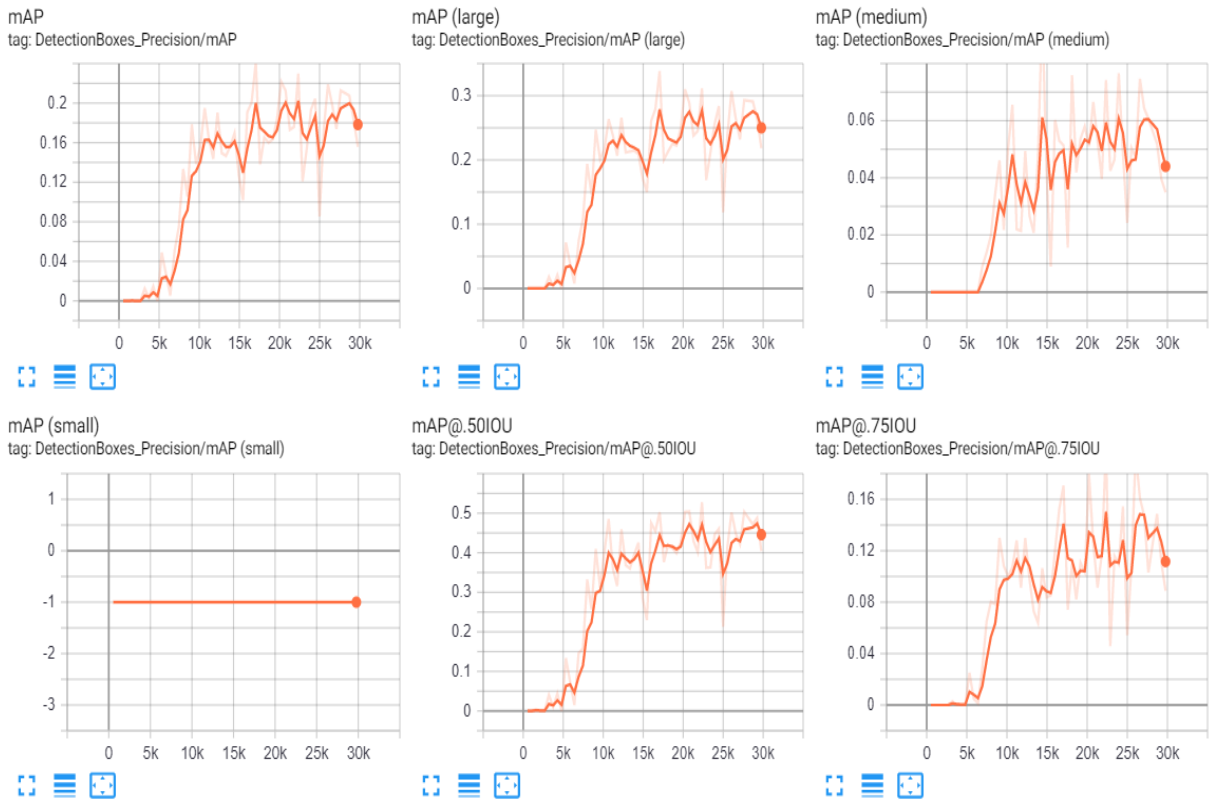
شکل ۲۴ نمونه شماره یک از موبایل‌نت اس اس دی

شکل ۲۵ نمونه شماره دو از موبایل‌نت اس اس دی

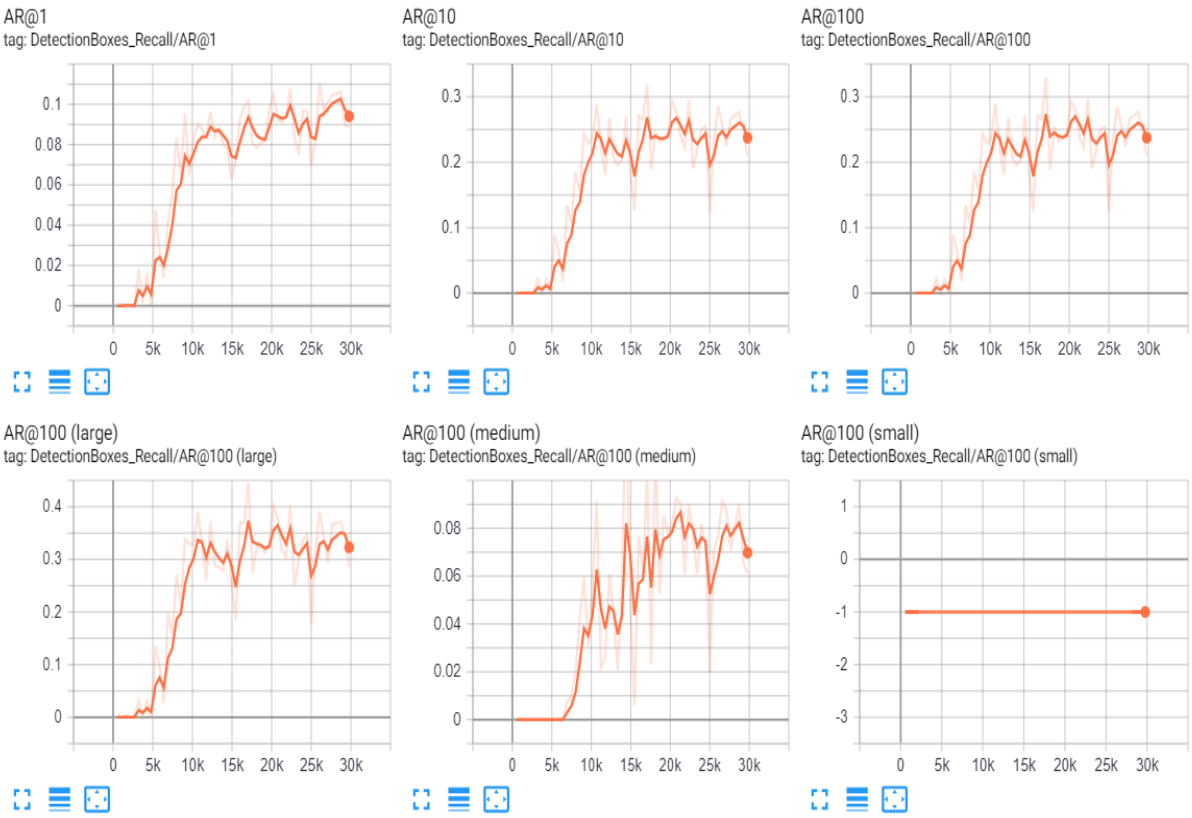
همچنین نمونه‌ای از دقت این شبکه در بالا نشان داده شده است. تصاویر سمت راست جواب درست و تصاویر سمت چپ جواب شبکه می‌باشد.

۲-۲-۴- نتایج بر روی رزنت اس اس دی

این شبکه علیرغم این که دقت بالاتری بر روی دیتاست کوکو نسبت به اس اس دی دارد به هیچ عنوان نتایج مطلوبی نمی‌دهد، دلیل آن هم به نظر من کم بودن دیتای برای آموزش می‌باشد. همچنین آموزش بر روی این شبکه بسیار زمانگیر می‌باشد و نشان می‌دهد این شبکه به هیچ عنوان مناسب برای مسئله ما نیست.



شکل ۲۶ میانگین درستی برای رزنت اس اس دی



شکل ۲۷میانگین یادآوری برای رزنت اس اس دی

شکل ۲۸ نمونه شماره یک از رزنت اس اس دی

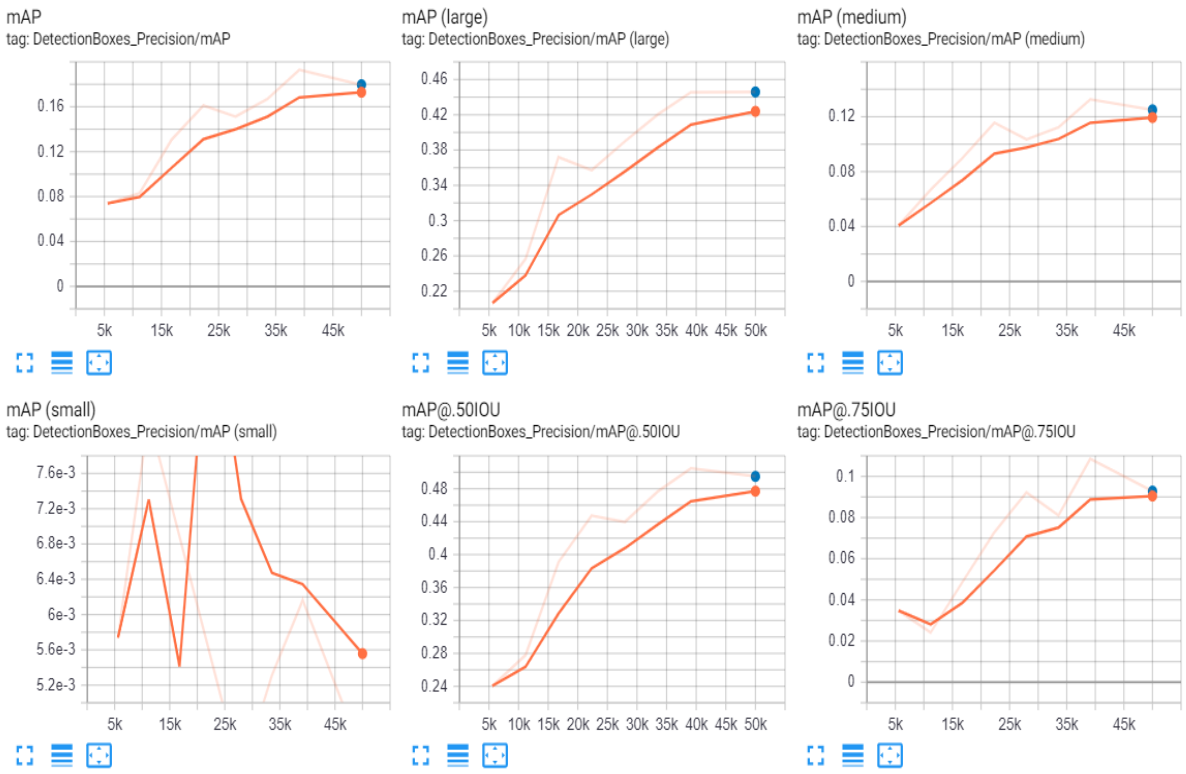
شکل ۲۹ نمونه شماره دو از رزنت اس اس دی

همانطور که در نمونه‌های بالا نمایان می‌باشد این شبکه نه تنها تمامی مسافران را تشخیص نداده بلکه آن‌هایی که دقیق هستند نیز به درستی اندازه‌های آن‌ها را مشخص نکرده است.

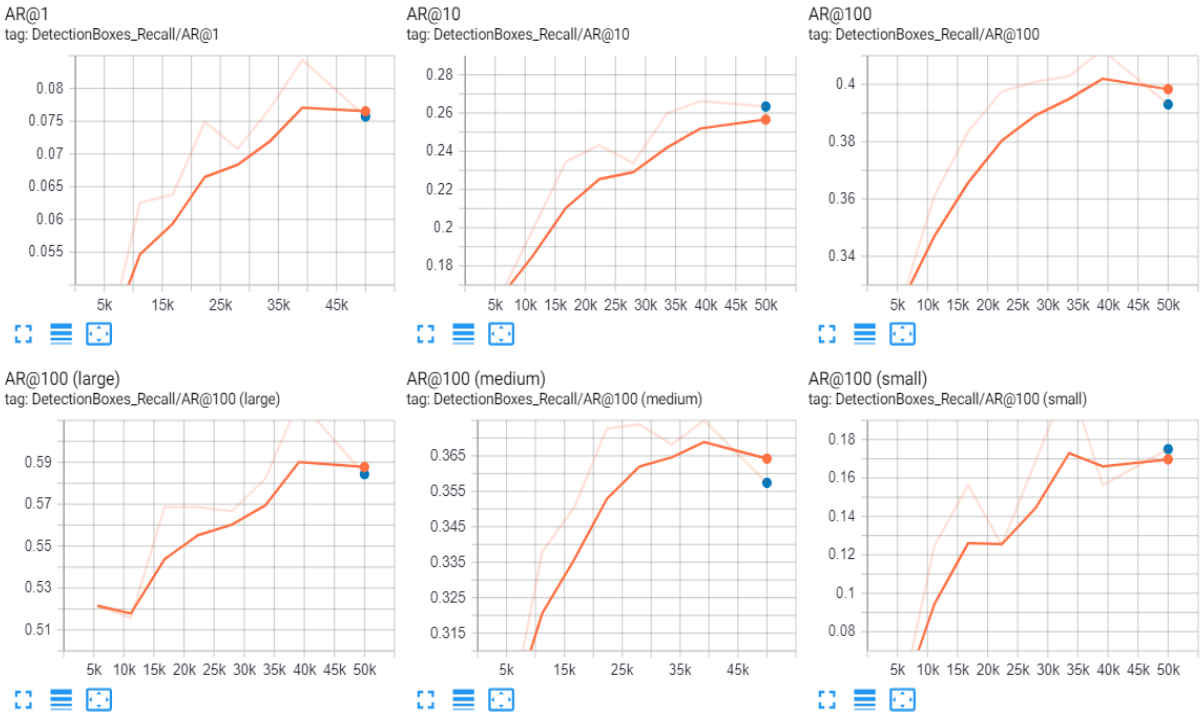
۳-۲-۴- نتایج بر روی افیشنت‌نت اس اس دی

این شبکه نیز سرعت خوبی هنگام آموزش داشت و کمی از موبایل‌نت اس اس دی بیشتر بود اما دقت آن بالا نبود و دلیل اصلی آن نبود یک چک‌پوینت مناسب برای تشخیص اشیا بود که باعث شد دقت آن بالا نرود.

با اینکه دقت‌ها نسبت به شبکه‌ی موبایل‌نت اس اس دی پایین هستند اما بهتر از رزنت اس اس دی کار کرده است. نمونه‌های زیر بیان گر دقت هستند.



شکل ۳۰میانگین درستی برای افیشنت نت



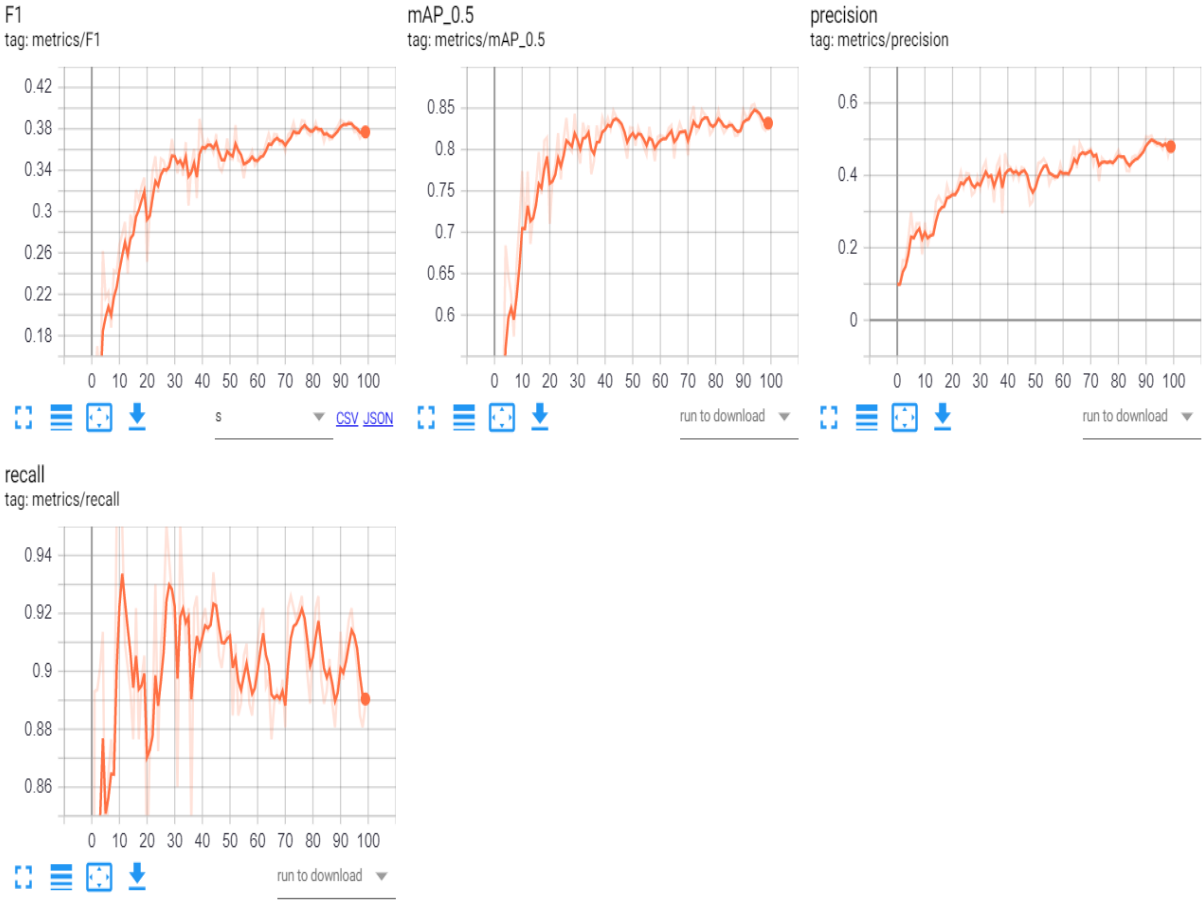
شکل ۳۱ میانگین یادآوری برای افیشنت نت

شکل ۳۲ نمونه شماره یک از افیشنت نت

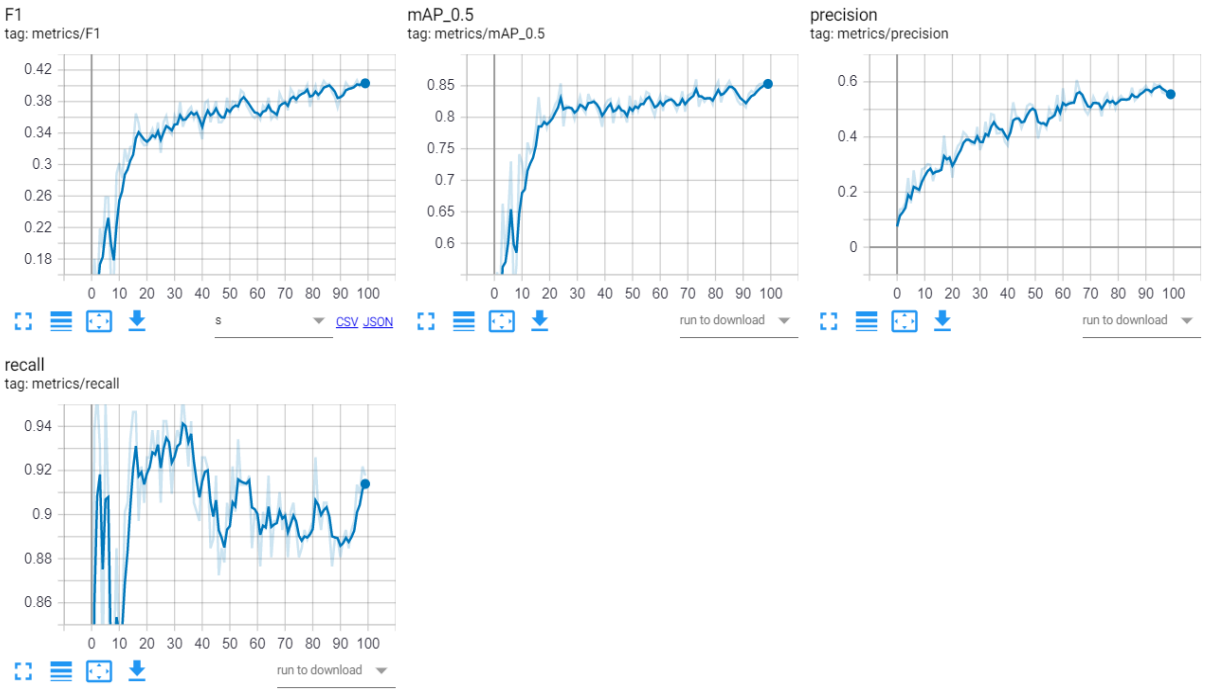
شکل ۳۳ نمونه شماره دو از افیشنت نت

۴-۲-۴- نتایج بر روی شبکه یولو

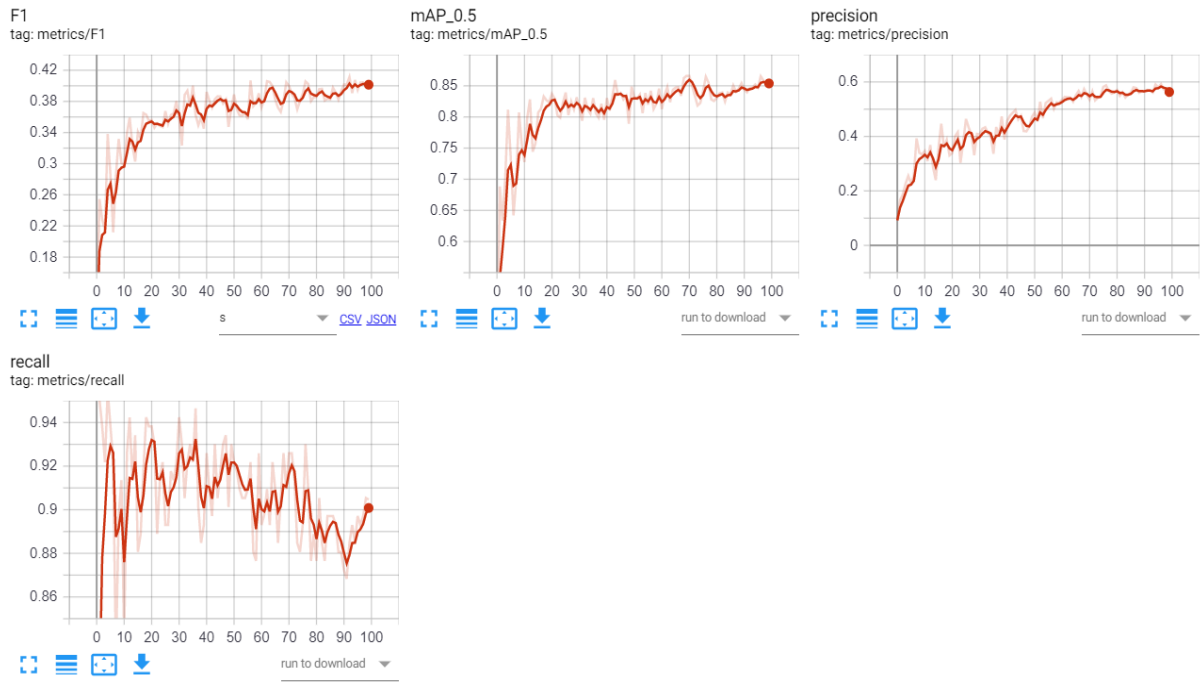
شبکه‌ی یولو طبق ادعایی که داشت، می‌بایست از تمامی شبکه‌ها بهتر عمل می‌کرد و همین اتفاق نیز افتاد و نتایج آن در هر ۴ نوع بسیار بالاتر از دیگر شبکه‌ها بود.



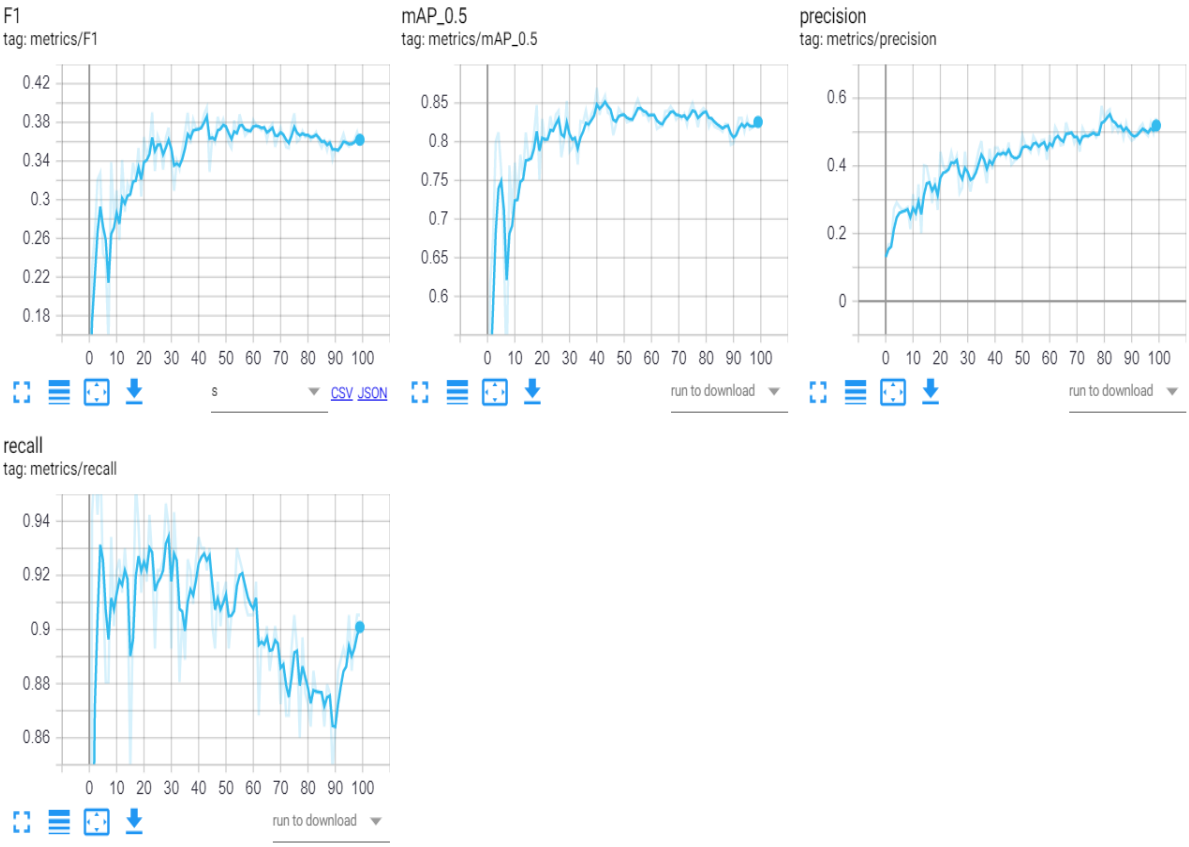
شکل ۳۴ نتایج برای یولو S



شکل ۳۵ نتایج برای یولو M



شکل ۳۶ نتایج برای یولو L



شکل ۳۷ نتایج برای یولو X

۳-۴- نتایج کلی و قابل مقایسه برای همه‌ی شبکه‌ها

جدول ۳ نتایج بدست آمده برای شناسایی

نوع شبکه	میانگین درستی ۵۰ به بالا
MOBILENET-SSD	۶۸.۹۳
RESNET-SSD	۴۰.۴
EFFICEINTNET-SSD	۴۷.۷
YOLOV5-S	۸۵.۴۷
YOLOV5-M	۸۵.۷۳
YOLOV5-L	۸۶.۴۶
YOLOV5-X	**۸۶.۹۳**

همانطور که در جدول شماره ۳ می‌توان مشاهده کرد، شبکه‌های یولو با اختلاف فراوان نسبت به دیگر شبکه‌ها بهترین دقت را دارا هستند، هرچند که ورژن X که بزرگترین ورژن از این شبکه می‌باشد، تفاوت چندانی با ورژن S ندارد.

۴-۴- نتایج سرعت شناسایی انسان

جدول ۴ مقایسه سرعت شبکه‌ها در شرایط یکسان

نوع شبکه	زمان کل(۱۵۰۰ فریم)	میانگین زمان
MOBILENET-SSD	**۶۸.۳۸ ثانیه**	**۴۵.۵۸ میلی‌ثانیه**
RESNET-SSD	۶۴۰.۷۸ ثانیه	۴۲۷.۱۸میلی‌ثانیه
EFFICEINTNET-SSD	۶۶.۳۶ ثانیه	۴۴.۲۴ میلی‌ثانیه
PRE-TRAINED MOBILENET-SSD	۹۷.۳۸ ثانیه	۶۴.۶۸ میلی‌ثانیه
YOLOV5-S	۳۴۸.۰۲ ثانیه	۲۳۲.۰۱ میلی‌ثانیه
YOLOV5-M	۷۱۹.۸۴ ثانیه	۴۷۹.۸۹ میلی‌ثانیه
YOLOV5-L	۱۲۱۰.۰۷ ثانیه	۸۰۶.۷۱ میلی‌ثانیه
YOLOV5-X	۲۲۰۰.۱ ثانیه	۱۴۶۶.۶ میلی‌ثانیه

همانطور که در جدول شماره ۴ مشاهده می‌شود، شبکه موبایل‌نت اس اس دی سرعت بالاتری نسبت به دیگر شبکه‌ها دارد. نکته جالب‌تر اینکه شبکه‌ای که بنده با تنها یک کلاس آموزش دادم از شبکه موبایل‌نت اس اس دی سبک‌تر و سریع تر است به علت اینکه شبکه اصلی ۹۰ کلاس دارد پس یعنی حتی اگر در شرایطی خود چک‌پونت شبکه دقت کافی را داشت بهتر از تعداد کلاس‌ها را به کمک آموزش بر روی دیتا ست خود کاهش دهیم تا سرعت بالا برود.

نتیجه‌گیری

با توجه به پژوهش‌های بنده با این مجموعه‌ی تصاویر محدود و زمان‌بر بودن برچسب گذاری بهترین شبکه برای تشخیص مسافران که هم دقت مطلوب و هم سرعت مناسبی داشته باشد شبکه یولو ورژن S می‌باشد این شبکه هم فضای نسبتا کوچکی را اشغال می‌کند و هم دقت مناسبی در تشخیص انسان دارد. به این نکته باید توجه داشت که در مقاله موبایل‌نت این شبکه از یولو ورژن ۳ دقت بالاتری دارد اما ورژن جدید یولو فوق‌العاده قدرتمند در تشخیص می‌باشد.

علاوه بر مشکل دیتاست مشکل دیگری نیز وجود داشت و آن نداشتن پردازنده‌ی قدرتمند برای آموزش شبکه بود هر چند سرویس گوگل بسیار کمک کننده و مفید می‌باشد اما دارای محدودیت‌هایی نیز می‌باشد که کار را برای مقیاس‌های بزرگتر سخت و پیچیده می‌کند. هر چند که در سرعت یولو ورژن S تقریبا ۵ برابر زمان بیشتری نسبت به موبایل‌نت اس اس دی استفاده می‌کند اما باید توجه داشت که در واقعیت ما همه‌ی فریم‌ها را بررسی نمی‌کنیم و تقریبا هر ۱ ثانیه عملیات تشخیص را انجام می‌دهیم، همچنین اگر از جی پی یو مناسب استفاده کنیم سرعت تشخیص هر فریم به شدت کاهش می‌یابد.

سرچشمه

[۱] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779-788 .

[۲] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91-99 .

[۳] W. Liu *et al.*, "SSD: Single Shot MultiBox Detector," *Lecture Notes in Computer Science,* pp. 21-37, 2016, doi: 10.1007/978-3-319-46448-0_2.

[٤] A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," ed, 2017.

[٥] S. Targ, D. Almeida, and K. Lyman, "Resnet in resnet: Generalizing residual architectures," *arXiv preprint arXiv:1603.08029,* 2016.

[٦] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *arXiv preprint arXiv:1905.11946,* 2019.

[۷] M. Tan, R. Pang, and Q. Le, *EfficientDet: Scalable and Efficient Object Detection*. 2019.

[۸] C.-Y. Wang, H.-Y. Mark Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, *CSPNet: A new backbone that can enhance learning capability of cnn* (Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops). 2020.

[۹] Y.-W. Hsu, T.-Y. Wang, and J.-W. Perng, "Passenger flow counting in buses based on deep learning using surveillance video," *Optik,* vol. 202, p. 163675, 2020/02/01/ 2020, doi: https://doi.org/10.1016/j.ijleo.2019.163675.

[۱۰] G .Liu, Z. Yin, Y. Jia, and Y. Xie, "Passenger flow estimation based on convolutional neural network in public transportation system," *Knowledge-Based Systems,* vol. 123, pp. 102-115, 2017/05/01/ 2017, doi: https://doi.org/10.1016/j.knosys.2017.02.016.