



Technovations of Electrical Engineering in Green Energy System

Design and Implementation of a Surveillance Security System Based on YOLO Algorithm and IoT Technology on Mobile Data Network

Mohamadreza Masaeli^{1,2}, MSc student, Sayed Mohammadali Zanjani^{1,2}, Assistant professor

¹ Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran

² Smart Microgrid Research Center, Najafabad Branch, Islamic Azad University, Najafabad, Iran

Abstract:

The design and implementation of a surveillance security system based on the YOLO algorithm and Internet of Things (IoT) technology has significant advantages in terms of security, efficiency, scalability, rapid response, and reliability, compared to traditional methods. This paper discusses the aspects of creating a novel security system that activates an alert by detecting five categories: human, human head, gun, knife, and fall detection. The system is monitored online and can connect to the internet via a cellular data network at any location to upload images to the management panel and send a report to the user if threats are detected. The YOLOv8 algorithm is used for object training to take advantage of its user-friendly command line interface, object detection support, sample segmentation, and image classification capabilities. To increase processing speed while maintaining accuracy, the optimized model is deployed on the Raspberry Pi 4th generation board. It is clear that optimizing processing speed and using quantization techniques lead to reduced energy consumption (green energy system) and reduced operational costs of the system. To improve the speed of the model in the object detection process, the techniques of exporting, quantizing the training weights, and increasing the processor frequency (overclocking) are used. A comparison of the newly exported weights with the original training weights in terms of accuracy and speed shows that the two techniques of exporting and quantization lead to an increase in processing speed at the cost of a decrease in detection accuracy. Finally, in the training model with the proposed improvement methods, an average accuracy of $mAP \cong 0.67$ with a frame rate of $FPS \cong 4.3$ can be achieved.

Keywords: Human activity recognition system (HAR), Machine vision, Violence prevention, Exporting and quantization, YOLO algorithm.

Received: 22 December 2020

Revised: 25 February 2021

Accepted: 13 March 2021

Corresponding Author: Dr. Sayed Mohammadali Zanjani, sma_zanjani@pel.iaun.ac.ir

DOI:





مقاله پژوهشی

فناوری‌های نوین مهندسی برق در سیستم انرژی سبز

طراحی و پیاده‌سازی سامانه امنیتی نظارتی مبتنی بر الگوریتم YOLO و فناوری اینترنت اشیا بر پایه شبکه داده همراه

محمدرضا مسائلی^{۱،۲}، دانشجوی کارشناسی‌ارشد، سید محمدعلی زنجانی^{۱،۲}، استادیار

۱- دانشکده مهندسی برق، واحد نجف‌آباد، دانشگاه آزاد اسلامی، نجف‌آباد، ایران

۲- مرکز تحقیقات ریزشبکه‌های هوشمند، واحد نجف‌آباد، دانشگاه آزاد اسلامی، نجف‌آباد، ایران

چکیده: افزایش چشمگیر امنیت، بهره‌وری مقیاس‌پذیری، پاسخگویی سریع و قابلیت اطمینان از مزایای طراحی و پیاده‌سازی سامانه امنیتی نظارتی مبتنی بر الگوریتم YOLO و فناوری اینترنت اشیا، در مقایسه با روش‌های سنتی است. در این مقاله، به جنبه‌های ایجاد یک سامانه امنیتی نوین پرداخته می‌شود که با تشخیص پنج رده شامل انسان، سر انسان، تفنگ، چاقو و تشخیص سقوط، هشدار را فعال می‌کند. نظارت بر عملکرد سامانه، به صورت برخط است. این سامانه در هر نقطه به کمک شبکه داده تلفن همراه، قابلیت اتصال به اینترنت را دارد تا در صورت شناسایی تهدیدات، تصاویر را در پنل مدیریتی بارگذاری و گزارش آن را به کاربر ارسال کند. برای تعلیم اشیا از الگوریتم YOLOv8 استفاده شده است تا از مزایای مانند رابط خط فرمان کاربرپسند، پشتیبانی آن از شناسایی اشیا، تقسیم‌بندی نمونه و طبقه‌بندی تصاویر بهره‌گیرد. برای افزایش سرعت پردازش، ضمن حفظ دقت، مدل بهینه‌سازی شده در بورد رزبری پای نسل چهارم استفاده شده است. واضح است که بهینه‌سازی سرعت پردازش و استفاده از تکنیک‌های کمی‌سازی منجر به کاهش مصرف انرژی (سامانه انرژی سبز) و کاهش هزینه‌های عملیاتی سامانه می‌شود. به منظور بهبود سرعت مدل در فرایند تشخیص اشیا، از تکنیک صادر کردن، کمی‌سازی وزن‌های تعلیمی و افزایش فرکانس پردازنده (اورکلاک) استفاده می‌شود. مقایسه وزن‌های صادر شده جدید با وزن اصلی تعلیمی، در شاخص دقت و سرعت، بیانگر آن است که دو تکنیک صادر کردن و کمی‌سازی، منجر به افزایش سرعت پردازش، به‌ازای کاهش دقت در تشخیص می‌شود. در نهایت، در مدل تعلیمی با روش‌های بهبود مطرح شده می‌توان به دقت متوسط $mAP \cong 0.67$ با تعداد قاب تصویر در ثانیه $FPS \cong 4.3$ دست‌یافت.

واژه‌های کلیدی:

سامانه تشخیص فعالیت‌های انسانی HAR، بینایی ماشین، مقابله با خشونت، صادر کردن و کمی‌سازی، الگوریتم YOLO.

تاریخ ارسال مقاله: ۱۴۰۲/۱۱/۰۳

تاریخ بازنگری مقاله: ۱۴۰۲/۱۲/۲۰

تاریخ پذیرش مقاله: ۱۴۰۳/۰۲/۰۳

نویسنده‌ی مسئول: دکتر سید محمدعلی زنجانی، sma_zanjani@pel.iaun.ac.ir

DOI:



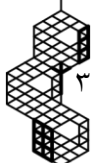
اهمیت طراحی و پیاده‌سازی سامانه امنیتی و نظارتی مبتنی بر الگوریتم YOLO و فناوری اینترنت اشیا بر پایه شبکه داده همراه، به دلیل افزایش چشمگیر امنیت، راندمان و قابلیت اطمینان در مقایسه با روش‌های سنتی است. تشخیص و هشدار خودکار تهدیدات بدون نیاز به عامل انسانی، کاهش هزینه نسبت به سامانه‌های امنیتی سنتی، مقیاس‌پذیری (قابلیت گسترش و پیاده‌سازی در سطوح مختلف)، دسترسی آسان (نظارت و کنترل از راه دور از طریق تلفن همراه)، پاسخ‌گویی سریع (سرعت بالا در تشخیص و هشدار تهدیدات) و قابلیت اطمینان (دقت بالا در تشخیص و کاهش خطاهای انسانی) از مزایای متعدد سامانه پیشنهادی است. لازم به ذکر است که الگوریتم YOLO به دلیل سرعت بالا در پردازش تصاویر و دقت قابل قبول در تشخیص اشیاء، انتخابی، برای سامانه‌های امنیتی و نظارتی مبتنی بر اینترنت اشیا، ایده‌آل است.

در دهه گذشته، پیشرفت‌های چشمگیری در حوزه هوش مصنوعی، تأثیرات عمیقی بر صنایع و سامانه‌ها در برداشته است. سامانه‌های حاضر با ترکیب قابلیت‌های تشخیص اشیاء و تعاملات برخط، این امکان را می‌دهند تا سریع و مؤثر، با تهدیدات امنیتی مقابله شود و با بهره‌گیری از دوربین‌های نظارتی و هوش مصنوعی، تحرکات و برخوردهای مخاطره‌آمیز، در محدوده دید دوربین، شناسایی شوند. بینایی ماشین در سامانه دوربین‌های مداربسته، ایده افزودن لایه‌ای از هوش مصنوعی به سامانه نظارتی است که به کاربران این امکان را می‌دهد تا از تصاویر دوربین‌های نظارتی خود، بهره بیشتری ببرند. برخی از سامانه‌های نظارتی، دارای یک تراشه هوش مصنوعی هستند که به آنها قدرت پردازشی می‌دهد تا عملکرد عناصر مختلف را بهبود بخشد؛ از جمله می‌توان به تشخیص اشیاء، شمارش افراد، تمایز بین انسان و حیوانات اشاره نمود. برای دوربین‌های مداربسته که تنها بر اساس حرکت اجسام هشدار می‌دهند، احتمال اخطارهای غیرواقعی وجود دارد، ضمن آن که نیاز به نظارت مداوم توسط پرسنل امنیتی، می‌تواند منجر به ایجاد مشکلات چشمی و علائمی همچون قرمزی و خشکی چشم، دید مات و دید دوتایی شود [۱]. همچنین مطالعات نشان داده‌اند که حتی وجود تارهای عنکبوت در مقابل دوربین‌های نظارتی می‌تواند باعث بروز هشدارهایی نادرستی شود [۲] و یا در [۳] برای تشخیص یک تهدید امنیتی در یک مکان خاص از روش شناسایی افراد مشکوک (به صورت بیکار) در محیط پرداخته است که از دقت کافی برخوردار نمی‌باشد؛ لذا می‌توان از کاربردهایی مانند سامانه‌های مبتنی بر هوش مصنوعی در سامانه‌های دوربین‌های مداربسته استفاده کرد که به وسیله تشخیص اشیاء تهدیدآمیز، راهکاری برای دوربین‌های مداربسته ارائه می‌دهند تا امنیت را به یک سطح جدید برسانند و عاملی در جهت کاهش جرم و برخوردهای خشونت‌آمیز باشند [۴-۷].

تنها ایراد این سامانه نظارتی مبتنی بر هوش مصنوعی، در محدود بودن تعداد اشیاء قابل تشخیص است که نیاز است در هنگام تعلیم، بسته به هدف موردنظر، پایگاه تصاویر ساماندهی شود [۷-۱۰]. تشخیص افراد در تصاویر دوربین‌های نظارتی یکی از مهم‌ترین عناصر از دید رایانه محسوب می‌شود و در سال‌های اخیر توجه بیشتری به خود جلب کرده است [۱۰-۱۳]. سامانه‌ای که قادر به تشخیص انسان است، نقش اساسی در برنامه‌هایی از جمله خودروهای خودران، نظارت هوش مصنوعی، رباتیک و تعامل پیشرفته انسان و ماشین ایفا می‌کند [۱۴]. با پیشرفت و گسترش شبکه‌های عصبی پیچشی CNNs [۱۵، ۱۶، ۱۷] تشخیص‌دهنده‌های مدرن انسان، عملکرد قابل توجهی در تشخیص انسان به دست آورده‌اند [۵، ۱۸، ۱۹].

سامانه‌های امنیتی، نظارتی و مراقبتی مبتنی بر هوش مصنوعی، می‌توانند به صورت خودکار و بدون نیاز به دخالت انسان، به نظارت بر محیط‌های مختلف و شناسایی موارد مشکوک بپردازند. این سامانه‌ها در اماکن تجاری و مغازه‌ها، ساختمان‌های اداری و دفاتر، مکان‌های عمومی و تفریحی، حمل نقل عمومی و محیط‌های خاص و حساس مانند بیمارستان‌ها، با نظارت بر ورود و خروج افراد، تشخیص اشیاء خطرناک و مراقبت از امنیت عمومی می‌توانند مفید باشند. از چالش‌های تشخیص اشیاء توسط دوربین‌های نظارتی به زاویه اشیاء، فاصله اشیاء تا دوربین نظارتی، انسداد و شرایط نور محیطی می‌توان اشاره کرد. شرایط نور محیطی، زاویه مختلف اشکال و فاصله از عوامل مهم در تشخیص اشیاء است. از دیگر چالش‌های اساسی در تشخیص اشیاء می‌توان به انسداد اشیاء از دید دوربین نظارتی اشاره داشت. به عبارتی تصویر کامل یک شیء، در دید دوربین نظارتی، قابل رویت نباشد [۲۰].

هدف از این مقاله، طراحی یک سامانه نظارتی و امنیتی بر پایه شبکه داده همراه به کمک الگوریتم YOLO است؛ لذا در بخش دوم، کارهای انجام شده قبلی در زمینه روش‌های تشخیص اشیاء، روش‌های رگرسیون و رده‌بندی و نیز الگوریتم YOLO معرفی شده‌اند. معیارهای ارزیابی و آماده‌سازی داده‌های تصاویر تعلیمی و سخت‌افزار پردازشی از اجزای مهم شبکه پیشنهادی می‌باشند که در بخش





سوم بررسی شده‌اند. در بخش چهارم، پیکره‌بندی پارامترها و ارزیابی عملکرد آنها همراه با پیاده‌سازی سخت‌افزاری سامانه، بررسی شده‌اند. در نهایت، بخش پنجم به نتیجه‌گیری نتایج حاصل از پیاده‌سازی سامانه نظارتی و امنیتی بر پایه شبکه داده همراه به کمک الگوریتم YOLO می‌پردازد و سامانه و تکنیک‌های استفاده شده در جهت حفظ دقت و سرعت در فرایند تعلیم شبکه ارزیابی می‌شوند.

۲- کارهای مرتبط انجام شده

توسعه سامانه‌های تشخیص خودکار سلاح در زمان واقعی بادقت و عملکرد بالا راه‌حلی امیدوارکننده برای پیشگیری از تهدیدات سلاح‌ها در مکان‌های عمومی هستند. از نتایج [۲۱] باتوجه به نتیجه دو شبکه SSD^۱ و Faster RCNN^۲ که در [۲۲،۲۳] انجام شده است، آن است که شناسایی سلاح در الگوریتم SSD سرعت بالاتری نسبت به الگوریتم Faster RCNN دارد؛ ولی الگوریتم Faster RCNN از شبکه SSD دقت پردازشی بالاتری دارد. همچنین در مرجع [۲۴] به بررسی و ارزیابی نسخه‌های ۳ و ۴ الگوریتم YOLO^۴ بر اساس [۲۵،۲۶] در سامانه تشخیص اشیاء پرداخته شده است و چالش تهیه بانک تصاویر مناسب در خصوص تعلیم در تشخیص سلاح بررسی شده است. در [۲۷] یک سامانه امنیتی، برای شناسایی سلاح مبتنی بر اینترنت اشیا پیاده‌سازی شده است که از YOLOv4 استفاده کرده است. در [۲۸] یک مدل یادگیری عمیق مبتنی بر YOLO ارائه شده است تا اسلحه‌ها را در ویدئوهای زمان واقعی CCTV^۵ شناسایی کند. در این مرجع از الگوریتم‌های مختلفی استفاده شده است. این الگوریتم‌ها شامل VGG16^۶ [۲۹]، Inception-V3 [۳۰]، Inception-ResnetV2 [۳۱،۳۲]، SSD-MobileNetV1 [۳۳،۳۴]، Faster-RCNN Inception-Resnetv2 (FRIR) [۲۳،۳۵]، YOLOv3 و YOLOv4 است و مشخص شده است که YOLOv4 بادقت بهتر و امتیاز $F1 = 91\%$ و مقدار $mAP = 91.73\%$ ، عملکرد بهتری داشته است.

در [۳۶] که به سامانه‌های تشخیص سلاح در دوربین‌های X-ray که به‌وسیله شبکه عصبی Faster RCNN انجام گرفته است می‌توان اشاره کرد. نقطه مشترک تمام آنها در چالش تهیه مجموعه بانک تصاویر موردنیاز برای تعلیم شبکه، اشاره شده است. در [۳۷] از مدل M2Det^۷ برای تحلیل داده‌ها از منابع مختلف استفاده کرده‌اند، از جمله مجموعه داده Granada که شامل ۳۰۰۰ تصویر از انواع مختلف تفنگ‌ها، مجموعه داده UCF crime که شامل ۷۲۴۷ تصویر و مجموعه داده خود مقاله که ۵۵۰۰ تصویر را شامل می‌شد. پژوهشگران همچنین شرایط مختلف محیطی را در طول تحلیل خود بررسی کردند. در این مطالعه، از دو مجموعه داده استفاده شد که مدل اول تنها بر مجموعه داده Granada تکیه کرده و مدل دوم هر دو مجموعه داده Granada و داده‌های خود مقاله را ترکیب کرد. نتایج، بیانگر دقت بالاتر مدل دوم به دلیل استفاده از تصاویر با قدرت تفکیک^۸ پایین و بالا است. با وجود این چالش‌ها و به دلیل افزایش جرم در بیشتر نقاط جهان، نظارت امنیتی به‌مراتب پر اهمیت بوده و نیاز است تا یک سامانه مبتنی بر بینایی ماشین در شناسایی تهدیدات در زمان واقعی طراحی شود.

هدف اصلی، پیاده‌سازی یک سامانه امنیتی، نظارتی، مراقبتی و محافظتی است که با استفاده از الگوریتم YOLO به تشخیص تهدید به‌منظور افزایش امنیت استفاده شود. از دیگر جنبه‌های این پژوهش در سامانه تشخیص انسان می‌باشد که یک کار مهم برای انواع کاربردهای نظارتی از جمله تشخیص اقدامات غیرعادی و نفوذ به محدوده است. به طور مثال در مرجع [۳۸] مشاهده می‌شود که هدف محققین در افزایش حدود تشخیص انسان است؛ به عبارتی تمرکز آنان یافتن راه‌حلی جهت تشخیص انسان در مکان‌های شلوغ بوده است.

از دیگر تحقیقات انجام شده در سامانه‌های تشخیص فعالیت انسان در زمان واقعی، HAR^{۱۰} می‌باشد که یک زمینه تحقیقاتی در هوش مصنوعی است [۳۹]. استفاده از فرایند تشخیص فعالیت HAR دارای کاربردهای گسترده‌ای در زمینه‌های مختلف از جمله موضوع امنیت برای شناسایی فعالیت‌های مشکوک مانند سرعت، خشونت، مراقبت بهداشتی به جهت نظارت بر سلامت افراد و شناسایی اختلالات حرکتی و تشخیص سقوط انسان و انجام فعالیت‌های ورزشی و یا ایجاد بازی‌های تعاملی می‌توان اشاره داشت [۴۰]. فرایند HAR در حوزه مراقبت پزشکی به جهت تشخیص سقوط انسان با استفاده از فنون یادگیری ماشین بر روی دوربین‌های نظارتی استفاده می‌شود. این روش می‌تواند برای نظارت و تشخیص فعالیت در خانه هوشمند نیز استفاده شود. در نهایت، فرایند یادگیری ماشین از بررسی رفتار انسان در تشخیص سقوط برای کمک به افراد مسن از نگاه فوریت‌های پزشکی سریع‌تر و پیشگیری از آسیب‌های احتمالی و جدی می‌تواند کمک شایانی به عمل آورد [۴۱]؛ لذا در راهبرد این پژوهش نیز از فرایند HAR استفاده شده است. همچنین از این فناوری می‌توان در بیمارستان‌ها و مراکز مراقبت از سالمندان بهره برد و نیروی انسانی و هزینه‌های پرستاری و مراقبتی را کاهش داد.



۱-۲- انواع روش‌های تشخیص اشیاء

هدف سامانه‌های تشخیص اشیاء، شناسایی شی موردنظر در قاب تصاویر است و در صورت وجود، سامانه تشخیصی باید رده (کلاس) و جعبه شی مربوطه را شناسایی و نمایش دهد؛ لذا سامانه‌های تشخیص اشیاء معمولاً با دو روش دومرحله‌ای و تک‌مرحله‌ای عمل می‌کنند. در روش دومرحله‌ای (مبتنی بر ناحیه)، ابتدا احتمال وجود شی در تصویر و احتمال استخراج آن بررسی می‌شود و در مرحله دوم در مورد ناحیه موردنظر تصمیم‌گیری می‌شود. در این روش ابتدا نواحی هدف، استخراج شده و سپس در ناحیه عملیات، تشخیص صورت می‌گیرد. از ایرادات این نوع از شبکه‌ها، تأخیر پردازش زمان واقعی است؛ ولی در ازای آن دقت قابل‌قبولی ارائه می‌دهند. از معروف‌ترین این الگوریتم‌های تشخیصی می‌توان به R-CNN، Fast R-CNN، Faster R-CNN اشاره کرد. این در حالی است که روش تک‌مرحله‌ای به‌طور کلی^{۱۱} به تصاویر می‌نگرد یا به عبارتی برای شناسایی اشیاء با نگاه رگرسیون^{۱۲} به موضوع می‌نگرد و از سرعت بالاتری برای تشخیص اشیاء برخوردار است. ایده تشخیص اشیاء به روش تک‌مرحله‌ای از نحوه تشخیص اشیاء توسط انسان شکل‌گرفته است که باعث به‌وجود آمدن الگوریتم YOLO شده است.

۲-۲- الگوریتم YOLO

الگوریتم YOLO باهدف ارائه روشی، مشابه کارکرد سامانه بینایی انسان طراحی شده است که از آن برای پیاده‌سازی بخش نظارتی برای تشخیص اشیاء خاص استفاده می‌شود. شبکه YOLO شامل یک شبکه عصبی کانولوشن است که بسته به نسخه و مدل از تعداد لایه‌های کانولوشنی متعددی برای استخراج و پیش‌بینی احتمال وجود شی و تعیین مختصات اشیاء استفاده می‌کند. شبکه عصبی، شبکه‌ای از نرون‌ها یا گره‌ها است که مانند مغز انسان عمل می‌کند. این شبکه با هر ورودی یاد می‌گیرد و سپس خطا را با خروجی واقعی مقایسه می‌کند و اعتبارسنجی می‌نماید [۳۷]. از مزایای الگوریتم YOLO نسبت به دیگر روش‌های تشخیص اشیاء مانند DPM^{۱۳} [۴۲] و شبکه‌های R-CNN می‌توان به این موارد اشاره کرد: YOLO بسیار سریع است [۲۵] و برای پیش‌بینی تشخیص، به‌صورت کلی به تصویر نگاه می‌کند؛ برخلاف فنون پنجره لغزان و روش پروپوزال، YOLO به کل تصویر نگاه می‌کند [۲۵]. همچنین، YOLO تصمیم‌پذیری بالایی دارد. به عبارتی زمانی که تصاویر به شبکه آموزش داده می‌شوند و سپس شبکه آموزش‌دیده روی داده‌های ورودی آزمایش می‌شود، شبکه YOLO بافاصله زیادی بهتر از شبکه‌هایی مانند DPM و R-CNN عمل می‌کند [۲۵].

۳-۲- رگرسیون و رده‌بندی^{۱۴}

رگرسیون و رده‌بندی دو روش متفاوت در آمار و یادگیری ماشین هستند که برای مدل‌سازی و پیش‌بینی اطلاعات از داده‌ها استفاده می‌شوند. هدف رگرسیون، تخمین یک تابع برای رسیدن به خروجی با مقدار واقعی از روی داده‌های ورودی است به عبارتی خروجی مدل همواره دارای یک مقدار پیوسته است؛ اما در رده‌بندی، هدف تخمین یک تابع برای رسیدن از ورودی به خروجی مطلوب به‌صورت گسسته است. الگوریتم YOLO یکی از نوآوری‌های مهم در زمینه تشخیص اشیاء است؛ این الگوریتم تمام اطلاعات لازم برای تشخیص اشیاء را به‌طور هم‌زمان از تصویر استخراج می‌کند. این شبکه عصبی به بیان اشیاء موجود در تصویر و ترسیم جعبه احتمالی اطراف اشیاء و همچنین پیش‌بینی احتمال وجود هر رده برای هر شی می‌پردازد. علت آنکه YOLO یک مسئله رگرسیون در نظر گرفته می‌شود، آن است که YOLO تنها به تعیین موقعیت اشیاء نمی‌پردازد؛ بلکه اطلاعاتی مانند اندازه جعبه مرزی^{۱۵} و احتمال وجود هر رده نیز به‌صورت مستقیم پیش‌بینی می‌شود؛ لذا این اطلاعات به‌صورت عددی و پیوسته مدل می‌شوند که با استفاده از آن می‌توان تغییرات دقیق در مختصات و ویژگی‌های مربوط به تشخیص اشیاء را پیش‌بینی کرد.

۳- روش پیشنهادی در تشخیص اشیاء اختصاصی

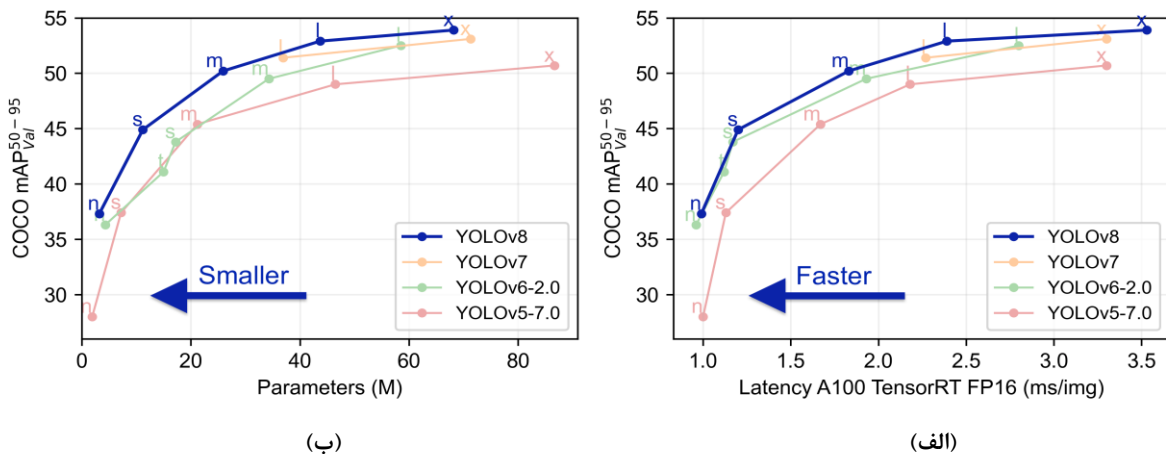
در این مقاله، تمرکز بر یک الگوریتم تشخیص اشیاء اختصاصی در جهت پیاده‌سازی یک سامانه امنیتی، نظارتی و مراقبتی برای تشخیص انسان، سلاح گرم و سرد و شناسایی سقوط انسان بروی زمین بر پایه بینایی ماشین و فناوری اینترنت اشیاء بر پایه شبکه داده همراه است که به کاربر این اجازه داده می‌شود تا سامانه را در هر زمان و مکانی بدون نیاز به شبکه WiFi فعال نماید و از آن در راستای ایجاد یک سامانه امنیتی مانند دزدگیر با نظارت دائم مانند یک نگهبان و مراقبتی در برابر انواع تهدیدات جانی و برخورد‌های خشونت‌آمیز عمل





نماید. مدل انتخابی، استفاده از YOLOv8 [۴۳] است این نسخه دقت بهتری نسبت به مدل‌های قبلی YOLO دارد و در محیط PyThorch تعلیم‌یافته است. از مزایای این نسخه در رابط خط فرمان کاربرپسند، پشتیبانی از شناسایی اشیاء، تقسیم‌بندی نمونه و طبقه‌بندی تصاویر است. همچنین از تغییرات اساسی این نسخه، تغییر در معماری پردازشی می‌باشد [۴۴]. از جمله تغییرات در این نسخه، شناسایی جعبه کادر دور اشیاء تشخیصی است که برخلاف سایر نسخه‌ها، جعبه اشیاء، بدون لنگر^{۱۶} می‌باشد. شناسایی اشیاء بدون لنگر این امکان را می‌دهد که مدل مرکز یک شی را به‌جای آفست پیش‌بینی کند. این تکنیک باعث افزایش کارایی این نسخه از YOLO و انعطاف‌پذیری بهتر می‌شود به‌نحوی که برای برنامه‌هایی که نیاز به شناسایی اشیاء به‌صورت زمان واقعی دارند، گزینه مناسبی است [۴۵].

شکل (۱) بیانگر بهبود عملکرد این نسخه از YOLO است که بر روی مجموعه داده COCO dataset نشان داده شده است [۴۳]. در این تحقیق یک چهارچوب تشخیص تهدیدات امنیتی ارائه شده است که با تجزیه و تحلیل تصاویر دوربین‌های نظارتی، امکان پایش و مراقبت از محیط موردنظر را داراست. اهداف موردنیاز ما به‌عنوان عوامل تهدید شامل: انسان، سر انسان، تفنگ، چاقو و تشخیص سقوط است. هدف از تشخیص سر انسان و انسان در جهت افزایش احتمال تشخیص روی دوربین نظارتی و چالش‌های انسداد در تصاویر و امکان در نظر گرفتن شرایط احراز هویت از روی چهره می‌باشد. هدف از تشخیص افتادگی نیز در مراقبت از سالمندان و افراد، در محیط‌های خاص به‌منظور انجام فوریت‌های پزشکی سریع‌تر، و هدف از شناسایی اشیاء خطرناک مانند سلاح گرم یا سرد در آشکارسازی اشیاء مخاطره‌آمیز جانی در محیط و در نهایت اعلام هر کدام به کاربر می‌باشد.



شکل (۱): نمایش بهبود عملکرد نسخه YOLOv8 نسبت به سایر نسخه‌ها، (الف) بهبود سرعت در پردازش. (ب) بهبود دقت در تشخیص اشیا.

۳-۱- معیارهای ارزیابی

از مهم‌ترین گام‌های پس از طراحی یک مدل یادگیری ماشین، ارزیابی خروجی آن بر روی مجموعه داده‌های دیده نشده یا آزمایشی است. در مبحث ارزیابی در روش تشخیص اشیاء بایستی مدل تشخیصی با دو عامل classification و localization مورد ارزیابی قرار گیرد. در مبحث تشخیص اشیاء معمولاً توزیع داده‌های تعلیمی هر دسته نسبت به تعداد کل داده‌ها نامتوازن است؛ بنابراین برای ارزیابی توانایی classification در یک روش تشخیص اشیاء، معیارهای Precision و Recall مورد استفاده قرار می‌گیرند [۴۶-۴۹].

الف) معیار IoU^{۱۷}: میزان دقت localization را با مقایسه میزان روی هم قرار گرفتن جعبه مرزی مرجع و جعبه پیشگویی شده مشخص می‌کند. این معیار نشان می‌دهد که در محل پیشنهادی، آیا شی وجود دارد یا خیر؟ در حقیقت این معیار به این صورت عمل می‌کند که اشتراک جعبه پیشنهادی با جعبه واقعی اطراف شی را بر اجتماع آنها تقسیم می‌کند که اگر از میزان آستانه از پیش تعیین شده بیشتر باشد، شی توسط مدل شناسایی می‌شود و اگر از حد مجاز کمتر باشد، امکان شناسایی شی در آن وجود ندارد و در نهایت در صورت وجود شی، جعبه پیشنهادی را رسم می‌کند. رابطه (۱) به بیان IoU می‌پردازد.

$$IoU = \frac{\text{area of overlap}}{\text{area of union}} \quad (1)$$



همچنین برای ارزیابی توانایی رده‌بندی یک روش تشخیص اشیاء معیارهای Precision و Recall نیز مطرح می‌شود؛ لذا برای محاسبه این دو پارامتر نیاز است تا به تعریف TP^{18} ، FP^{19} ، FN^{20} در مدل تشخیص اشیاء بپردازیم که این کار با در نظر گرفتن دقت در مبحث localization انجام می‌شود. TP بیانگر تعداد نمونه‌های مثبتی که به درستی مثبت تشخیص داده شده‌اند. FP تعداد نمونه‌های منفی که به اشتباه مثبت تشخیص داده شده‌اند. FN تعداد نمونه‌های مثبتی که به اشتباه منفی تشخیص داده شده‌اند را در مدل نمایان می‌کند.

(ب) معیار Precision: معیار (P) نشان می‌دهد که چه نسبتی از مثبت‌های پیش‌بینی شده واقعاً مثبت واقعی هستند و به صورت (۲) محاسبه می‌شود.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

(پ) معیار Recall: معیار (R) نشان می‌دهد که چه نسبتی از مثبت‌ها به درستی مثبت دسته‌بندی شده‌اند که طبق (۳) محاسبه می‌شود.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

(ت) معیار F1-Score: این معیار برای ارزیابی دقت یک آزمایش، مناسب است و حاصل دو معیار Precision و Recall را با هم در نظر می‌گیرد. معیار F1 در بهترین حالت، یک و در بدترین حالت صفر است و به صورت (۴) محاسبه می‌شود.

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

(ث) معیار AP: یک معیار ارزیابی برای مدل‌های تشخیص شیء است که میزان دقت و کارایی مدل را در تشخیص همه رده‌های شیء در یک مجموعه داده ارزیابی می‌کند. AP^{21} با محاسبه Precision در نقاط مختلف Recall محاسبه می‌شود. سپس، این نقاط به یکدیگر متصل می‌شوند تا یک منحنی Precision-Recall ایجاد شود و مقدار AP حاصل زیر منحنی این دو معیار طبق (۵) محاسبه می‌شود. لازم به یادآوری است که در هر نقطه از منحنی دقت بازخوانی، Precision(Recall) دقت در آن نقطه خاص را با توجه به مقدار بازخوانی نشان می‌دهد. در نتیجه بازه انتگرال از ۰ تا R باشد که در آن، R حداکثر مقدار بازخوانی است. این موضوع، زمانی اتفاق می‌افتد که منحنی دقت بازخوانی به طور کامل به محور X نرسیده باشد و گر نه این بازه از ۰ تا ۱ است؛ زیرا بازه بازخوانی نیز از ۰ تا ۱ می‌باشد.

$$AP = \int_0^1 (Precision(Recall)) dRecall \quad (5)$$

(ج) معیار mAP: معیار mAP یک معیار رایج برای ارزیابی عملکرد مدل‌های تشخیص شیء است. این معیار توانایی مدل در تشخیص صحیح همه اشیاء در یک مجموعه داده را اندازه‌گیری می‌کند؛ لذا ابتدا باید AP برای هر رده شیء در مجموعه داده محاسبه شود و سپس از طریق رابطه (۶) به محاسبه mAP برای کل مجموعه داده‌های تشخیص پرداخت.

$$mAP = \frac{1}{n} \sum_{i=1}^n AP_i \quad (6)$$

واضح است که اگر مقدار n به سمت اعداد بزرگ میل کند، آنگاه مقدار این سیگما برابر با انتگرال معین در بازه صفر تا یک خواهد بود؛ لذا با افزایش n تعداد نقاط نمونه‌برداری شده در منحنی دقت بازخوانی افزایش می‌یابد. البته محاسبه mAP با n بزرگ می‌تواند از نظر محاسباتی پرهزینه باشد و در برخی موارد ممکن است محاسبه انتگرال معین نیز دشوار باشد.

(چ) معیار FPS: این معیار سرعت تشخیص مدل‌ها در فریم‌ها بر ثانیه را مشخص می‌کند. شاخص FPS²² به میزان قدرت پردازنده در اجرای الگوریتم باز می‌گردد. هرچه این عدد بیشتر باشد، پردازش الگوریتم تشخیصی در اجرا، عملکردی سریع‌تری را خواهد داشت.

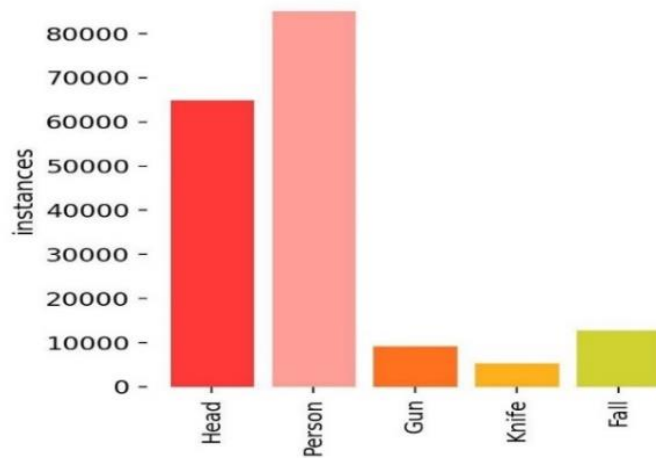
۲-۲- آماده‌سازی مجموعه داده‌های تصاویر

در این تحقیق، از مجموعه پایگاه داده تصاویر در Kaggle و Open Images و Roboflow و بخشی از پایگاه تصاویر [۳۷] استفاده نمودیم. مجموعه داده‌ها شامل تقسیم‌بندی و برچسب‌گذاری از هر رده به صورت انحصاری است. این سایت‌ها در حوزه علم داده، حاوی مجموعه داده‌های متنوع و جالب توجهی هستند که توسط مشارکت‌کنندگان گوناگون فراهم شده است و می‌توان از انواع مجموعه داده‌های آن استفاده کرد. تنوع این مجموعه داده‌ها بسیار بالا است و حوزه‌های گوناگونی از داده‌ها را شامل می‌شود. لازم به ذکر است برای کاهش میزان خطا و افزایش دقت در الگوریتم، حتماً لازم است برچسب‌گذاری تمام رده‌ها بر روی کلیه گروه‌های بانک تصاویر انجام گیرد؛ لذا در اولین قدم بر روی تمام تصاویر از هر رده تعریف شده، اقدام به برچسب‌گذاری مجدد رده‌ها می‌کنیم. این کار تأثیر به‌سزایی در میزان





خطای شبکه تعلیمی و میزان دقت شبکه دارد. نمودار شکل (۲) تمام رده‌های برچسب شده در مجموعه بانک تصاویر را نشان می‌دهد که هر دسته چند نمونه برای تعلیم دارد. تقسیم‌بندی تصاویر تعلیمی و اعتبارسنجی نیز با نسبت ۸۰٪ به ۲۰٪ در نظر گرفته شده است.



شکل (۲): نمودار تعداد رده‌های برچسب شده

از دیگر عوامل مؤثر و تأثیرگذار در دقت سامانه تشخیصی نیز می‌توان به تفکیک پذیری و کیفیت دوربین نظارتی اشاره کرد. استفاده از دوربین با کیفیت می‌تواند کارایی سامانه را در تشخیص افزایش دهد. به طور مثال، استفاده از دوربین نظارتی همراه با قابلیت دید در شب یا دوربین‌های حرارتی، می‌تواند قابلیت تشخیص را در مجموعه سامانه نظارتی مبتنی بر هوش مصنوعی افزایش دهد.

۳-۳- محیط پردازشی گوگل کولب^{۲۳}

گوگل کولب یک سرویس ابری رایگان است که به کاربران این امکان را می‌دهد که کد خود را بر روی این فضا اجرا کنند و مدل‌های یادگیری ماشین را آموزش دهند. چون تعلیم شبکه‌های عصبی پیچشی جزو سنگین‌ترین پردازش‌ها هستند، از کولب استفاده می‌شود که از انواع منابع محاسباتی، مانند GPU و CPU و TPU پشتیبانی می‌کند. این بدان معنا است که کاربران می‌توانند مدل‌های یادگیری ماشین خود را با سرعت بالا آموزش دهند. در همین راستا و با استفاده از زیرساخت کولب شرکت گوگل و پردازنده GPU مدل Tesla T4 عملیات یادگیری ماشین برای الگوریتم YOLO به جهت تشخیص اشیاء اختصاصی استفاده می‌نماییم. لازم به ذکر است که استفاده از این سرویس در طول روز دارای محدودیت بوده، اخذ پشتیبان^{۲۴} و ذخیره وزن‌های تعلیمی در هر دوره بر روی گوگل درایو^{۲۵} ضروری است [۴۵].

۳-۴- سخت‌افزار پردازشی برای اجرای سامانه بینایی

در این پروژه با استفاده از برد رزبری پای نسل چهار با هشت گیگابایت رم به تشخیص اشیاء می‌پردازیم. برد رزبری پای یک مینی کامپیوتر با قدرت محاسباتی محدود است؛ لذا مدل‌های یادگیری پیچیده و پیشرفته از شبکه‌های عصبی در این برد می‌توانند با تأخیر در زمان پردازش روبه‌رو شوند؛ به همین منظور لازم است که جهت افزایش سرعت الگوریتم تشخیص اشیاء در برد رزبری پای با صادر کردن^{۲۶} وزن تعلیمی، مدل YOLO را به سایر مدل‌ها صادر نماییم. این تکنیک به منظور کاهش زمان پردازش مدل یادگیری است. در این خصوص برای کاهش زمان پردازش در مدل تعلیمی تکنیکی‌هایی وجود دارد؛ مانند: صادر کردن مدل یادگیری ماشین از یک چارچوب^{۲۷} پردازشی به یک چارچوب دیگر. این کار می‌تواند برای اهداف مختلفی انجام شود؛ مانند: سازگاری مدل تعلیمی با سکوی^{۲۸} سخت‌افزاری، بهبود کارایی و کاهش اندازه مدل. به همین جهت باتوجه به سخت‌افزار پردازشی، مدل YOLO تعلیم‌یافته در محیط PyTorch را به فرمت TensorFlow Lite تبدیل می‌نماییم که یک چارچوب مناسب برای ادوات پردازشی با منابع محدود در مدل‌های یادگیری ماشین است. مدل TFLite به دلیل استفاده از الگوریتم فشرده‌سازی، کوچک‌تر از مدل اصلی خواهد بود. این روش، یک کار مفید برای اجرای مدل بر روی ادوات با منابع محدود است. کمی‌سازی^{۲۹}، فرایندی که به کاهش اندازه و دقت مدل و افزایش سرعت اجرای پردازش ختم



می‌شود، می‌تواند در دو سطح انجام شود: کمی‌سازی FP16 و INT8. در سطح FP16 مقادیر وزن‌ها و فاکتورهای فعال از نوع ممیز شناور ۳۲ بیتی به ممیز شناور ۱۶ بیتی کاهش می‌یابند و منجر به کاهش اندازه مدل می‌شود؛ اما در سطح INT8، مقادیر وزن‌ها و فاکتورهای فعال از نوع عدد صحیح ۱۶ بیتی به عدد صحیح ۸ بیتی کاهش می‌یابند و این کار منجر به کاهش اندازه و دقت مدل در تشخیص می‌شود؛ اما در ازای آن سرعت پردازش افزایش می‌یابد.^{۲۰}

۳-۵- بالازدگی^{۲۱} و پائین زدگی^{۲۲} در فرایند تعلیم

در فرایندهای یادگیری ماشین بالازدگی و پائین زدگی از مهم‌ترین موضوعاتی هستند که باید در نظر گرفته شود. در واقع مفهوم بالازدگی در زمانی رخ می‌دهد که مدل بسیار پیچیده شده و دقت پیش‌بینی برای داده‌های آموزشی بسیار بالا است، اما برای داده‌های جدید تست دقت پیش‌بینی کمی دارد به عبارتی مدل بیش از حد وابسته به داده‌های آموزشی شده است و برای داده‌های جدید نتایج بسیار بدی را ارائه می‌دهد. از طرفی نیز مفهوم پائین زدگی زمانی رخ می‌دهد که مدل بسیار ساده است به عبارتی مدل نتوانسته است به اندازه کافی فرصت فرایند تعلیم را داشته باشد تا رابطه‌ای میان متغیرهای ورودی و خروجی را به طور دقیق ثبت کند در نهایت هر کدام از این دو حالت باعث کاهش دقت در پیش‌بینی برای مدل تعلیمی هستند [۴۶].

۳-۶- یادگیری انتقالی^{۲۳}

یادگیری انتقالی یک تکنیک یادگیری ماشین است که از دانش یک مدل آموزش‌دیده برای آموزش یک مدل جدید استفاده می‌کند. این تکنیک می‌تواند برای بهبود عملکرد مدل جدید در یک وظیفه جدید بدون نیاز به آموزش مجدد کامل استفاده شود. تکنیک یادگیری انتقالی در بسیاری از زمینه‌های یادگیری ماشین، از جمله بینایی ماشین، پردازش زبان طبیعی و یادگیری عمیق کاربرد دارد. یادگیری انتقالی به روش‌های مختلفی انجام می‌شود. یکی از روش‌های رایج، استفاده از وزن‌های آموزش‌دیده در مدل جدید است. در این روش، وزن‌های مدل آموزش‌دیده به عنوان نقطه شروعی برای آموزش مدل جدید استفاده می‌شوند؛ لذا با استفاده از این تکنیک و ویژگی‌های استخراج شده از شبکه از قبل آموزش‌دیده، به تعلیم مدل برای شناسایی اشیاء موردنظر می‌پردازیم. تکنیک یادگیری انتقالی به بهبود مدل یادگیری در فرایند تعلیمی کمک می‌کند و منجر به صرفه‌جویی در زمان و منابع سخت‌افزاری می‌شود [۴۷].

۴- پیکربندی پارامترهای YOLOv8n در تعلیم اشیاء اختصاصی

مدل YOLOv8n یک شبکه عصبی مصنوعی با ۱۶۸ لایه و ۳۰۰۶۶۲۳ پارامتر است که کوچک‌ترین مدل ارائه شده از نسخه YOLOv8 می‌باشد. به عبارتی مدل‌های دیگر دارای تعداد لایه‌ها و پارامترهای بیشتری نسبت به YOLOv8n می‌باشند. ضمناً انتخاب مدل مناسب وابسته به توان منابع سخت‌افزاری در اجرای وزن تعلیمی است. در ادامه، تنظیمات پیکربندی YOLOv8n مورد استفاده در این مقاله، ارائه شده است. پارامتر model مسیر وزن مدل انتخابی تعلیمی را مشخص می‌کند که به عنوان نقطه شروعی برای آموزش یک شبکه اختصاصی به روش یادگیری انتقالی استفاده می‌شود. در این پژوهش، از مدل YOLOv8n.pt استفاده شده است. پارامتر epoch، تعداد دوره‌هایی را مشخص می‌کند که باید مدل برای یادگیری سپری کند. برای تعلیم اشیاء اختصاصی موردنظر، این دوره بر روی ۱۰۰ تنظیم شده است. همچنین استفاده از پارامتر patience به منظور جلوگیری از وقوع بالازدگی برای مدل تعلیمی است؛ به عبارتی مدل در فرایند تعلیم به بررسی شاخص‌های اعتبارسنجی در هر دوره یادگیری می‌پردازد و اگر در چند دوره، خطای اعتبارسنجی افزایش یابد، به منظور جلوگیری از وقوع بالازدگی در مدل، این پارامتر، فرایند تعلیم را متوقف می‌کند.

در اینجا اگر پس از سپری شدن ۱۰ دوره، خطای اعتبارسنجی، افزایشی شد، فرایند تعلیم متوقف می‌شود. batch، تعداد تصاویر آموزشی که باید به طور هم‌زمان پردازش شوند را مشخص می‌کند. انتخاب دسته بزرگ‌تر می‌تواند منجر به کاهش فرایند تعلیم شود؛ ولی از طرفی منابع سخت‌افزاری بیشتری را اشغال می‌کند و ممکن است سخت‌افزار پردازشی گیر کند؛^{۲۴} لذا انتخاب تعداد دسته‌ها بر اساس قدرت پردازنده در پردازش است که بر روی مقدار ۳۲ تنظیم شد. پارامتر imgz نیز ابعاد تصاویر ورودی برای تعلیم را مشخص می‌کند. هرچه ابعاد تصاویر بزرگ‌تر باشد، مدل تعلیمی جزئیات بیشتری از تصاویر را ثبت می‌کند؛ به همین منظور ابعاد تصاویر تعلیمی بر روی ۶۴۰*۶۴۰، تنظیم شد؛ از طرفی فرایند تعلیم در ابعاد بزرگ زمان پردازش طولانی‌تری را در پی دارد. همچنین در راستای تعلیم اشیاء اختصاصی در مدل YOLOv8 به منظور بهینه‌سازی یادگیری از پارامتر auto برای بهبود عملکرد مدل و تنظیم نرخ یادگیری و سایر

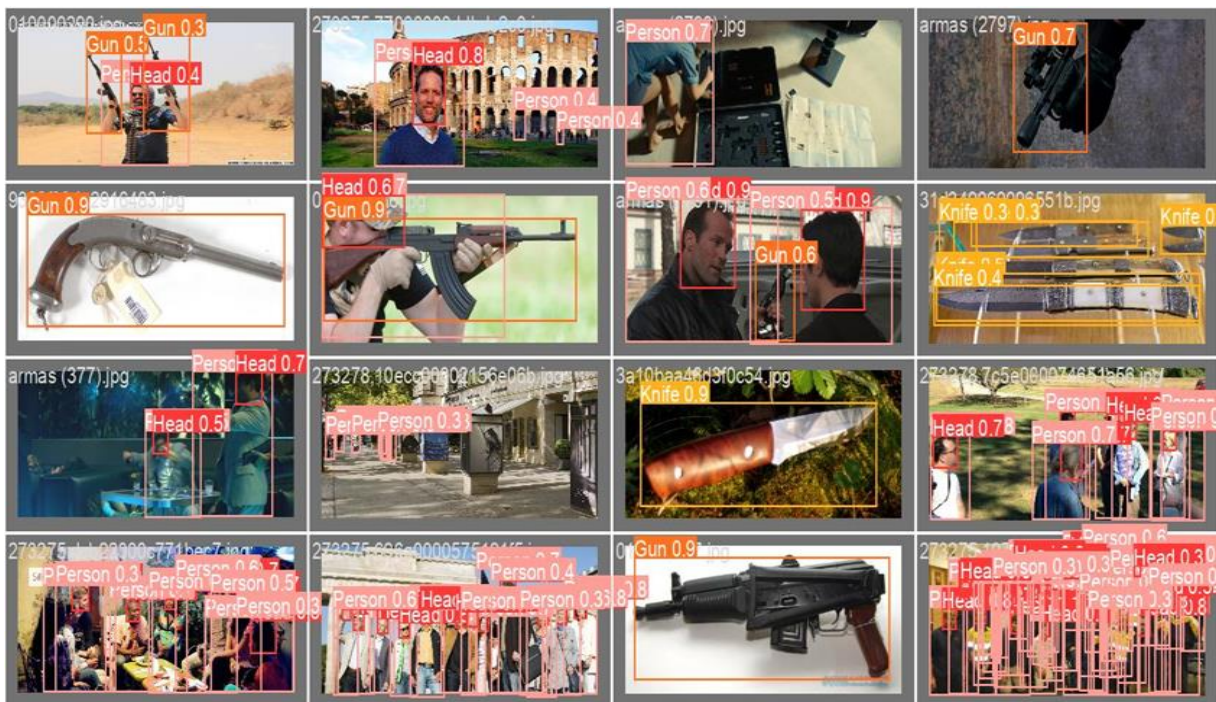




پارامترها استفاده شد. فرایند بهینه‌سازی با استفاده از پارامتر optimizer بسته به دوره‌های تعلیم، به صورت خودکار، تنظیم و بهینه می‌شود. حاصل تنظیم خودکار در انتخاب شدن مدل بهینه‌ساز SGD است که یک بهینه‌ساز رایج برای فرایندهای یادگیری عمیق است. سپس بهترین وزن YOLOv8n تعلیم‌یافته را به محیط TensorFLOW Lite صادر می‌کنیم. در ادامه، پارامترهای کمی‌سازی و وزن‌های صادر شده جدید ارزیابی و مقایسه می‌شوند.

۴-۱- ارزیابی عملکرد وزن‌های تعلیم‌یافته و صادر شده

در این بخش به ارزیابی و اعتبارسنجی وزن‌های تعلیمی خروجی YOLOv8 و TFLite به‌عنوان مدل بهینه‌ساز می‌پردازیم. شکل (۳)، بیانگر خروجی تشخیص اشیاء تعلیمی، در مدل YOLOv8n می‌باشد که از چندین زیر تصویر تشکیل شده است و هرکدام از تصاویر حاوی رده‌بندی‌های مختلفی از اشیاء می‌باشند. هر تصویر دارای برچسب‌هایی از مجموعه داده‌های تعلیمی می‌باشد که با میزان درصد اطمینان و نوع شیء به نمایش گذاشته شده است.



شکل (۳): نمونه اشیاء تشخیص‌داده‌شده در طرح پیشنهادی

ارزیابی عملکرد مدل‌ها در جداول (۱) تا (۶) در پنج کلاس آموزش‌دیده قابل مشاهده است که شامل سر، انسان، تفنگ، چاقو و تشخیص سقوط می‌باشد. در جداول (۱) تا (۶)، دو پارامتر P و R به ترتیب معرف Precision و Recall است. در جدول ۱، ارزیابی شبکه پیشنهادی در محیط PyTorch قابل رویت است. ارزیابی در این محیط، بیانگر میزان دقتی برابر ۷۰/۹٪ برای mAP در محدوده ضریب اطمینان ۰/۵ تا ۰/۹۵ و نیز دقت ۰/۹۱٪ برای ضریب اطمینان تشخیص ۰/۵ است.

در جدول ۲ اثر صادر کردن و کمی‌سازی مدل به محیط TensorFLOW Lite در محاسبه ممیز شناور ۳۲ بیتی نشان داده شده است. واضح است که ارزیابی در این محیط، بیانگر میزان دقتی برابر ۶۹/۹٪ برای mAP در محدوده ضریب اطمینان ۰/۵ تا ۰/۹۵ و نیز دقت ۹۰/۱۶٪ برای ضریب اطمینان تشخیص ۰/۵ است.

مقادیر مشابه حاصل در جداول ۳ و ۴ معرف اثر صادر کردن و کمی‌سازی مدل در محاسبه ممیز شناور ۱۶ بیتی و ۸ بیتی صحیح هستند. جدول‌های ۵ و ۶ اثر صادر کردن و کمی‌سازی کامل و جزئی شدن را در محاسبات مدل تعلیمی نشان می‌دهند. لازم به ذکر است که در کمی‌سازی کامل، تمام وزن‌های تعلیمی به فرمت محاسباتی اعداد صحیح کمی‌سازی شده تبدیل می‌شوند که این امر باعث افزایش سرعت و کاهش حجم پردازشی می‌شود، اما می‌تواند دقت را به مراتب کاهش دهد. از طرفی کمی‌سازی جزئی فقط برخی از لایه‌های مدل تعلیمی را به اعداد صحیح، کمی‌سازی می‌کند. این موضوع بین دقت، سرعت و حافظه تعادل مناسبی ایجاد می‌کند.



ارزیابی جداول (۱) تا (۶) نشان‌دهنده تأثیر استفاده از روش صادرکردن و کمی‌سازی در مدل تعلیمی می‌باشد. به عبارتی، در مدل YOLOv8n_best نسبت به مدل best_full_integer_quant، مصرف توان و نیاز به حجم پردازش بالاتری است. از طرفی، دقت تشخیص اشیا با ضریب اطمینان بالاتر در مدل YOLOv8n_best بیشتر می‌باشد؛ لذا در انتخاب مدل تشخیص اشیا، باید به دقت تشخیص و حافظه موردنیاز و توان مصرفی منابع سخت‌افزاری توجه کرد و برای کاربردهایی که به دقت تشخیص بالایی نیاز است، از مدل اصلی، و در مواردی که نیاز به کاهش مصرف حافظه و توان کم است، از مدل‌های بادقت پایین‌تر استفاده کرد و سرعت پردازش را افزایش داد.

جدول (۱): ارزیابی مدل YOLOv8n_best

Class Name	P	R	mAP 0.5	mAP 0.5:0.95
Head	۰.۸۹۲	۰.۷۵۵	۰.۸۴۳	۰.۵۵۱
Person	۰.۸۸۸	۰.۷۰۳	۰.۸۳۷	۰.۵۴۹
Gun	۰.۹۱۲	۰.۸۵۱	۰.۹۲۸	۰.۷۱۴
Knife	۰.۸۸۱	۰.۹۰۳	۰.۹۴۶	۰.۷۶۸
Fall	۰.۹۷۳	۰.۹۷۴	۰.۹۹۳	۰.۹۶۱
All	۰.۹۰۹	۰.۸۳۷	۰.۹۱	۰.۷۰۹

جدول (۲): ارزیابی مدل best_float32

Class Name	P	R	mAP 0.5	mAP 0.5:0.95
Head	۰.۸۹۸	۰.۷۴۸	۰.۸۴۱	۰.۵۵۱
Person	۰.۸۹	۰.۶۹۵	۰.۸۳۳	۰.۵۴۴
Gun	۰.۹۱۶	۰.۸۴۶	۰.۹۲۵	۰.۶۹۲
Knife	۰.۸۷۶	۰.۸۹۳	۰.۹۳۸	۰.۷۵۳
Fall	۰.۹۶۶	۰.۹۷۴	۰.۹۹۲	۰.۹۵۷
All	۰.۹۰۹	۰.۸۳۱	۰.۹۰۶	۰.۶۹۹

جدول (۳): ارزیابی مدل best_float16

Class Name	P	R	mAP 0.5	mAP 0.5:0.95
Head	۰.۸۹۸	۰.۷۴۸	۰.۸۴۱	۰.۵۴۶
Person	۰.۸۹	۰.۶۹۵	۰.۸۳۲	۰.۵۴۳
Gun	۰.۹۱۶	۰.۸۴۶	۰.۹۲۵	۰.۶۹۲
Knife	۰.۸۷۶	۰.۸۹۳	۰.۹۳۸	۰.۷۵۳
Fall	۰.۹۶۶	۰.۹۷۴	۰.۹۹۲	۰.۹۵۷
All	۰.۹۰۹	۰.۸۳۱	۰.۹۰۶	۰.۶۹۸

جدول (۴): ارزیابی مدل best_int8

Class Name	P	R	mAP 0.5	mAP 0.5:0.95
Head	۰.۸۹۳	۰.۷۴۷	۰.۸۴۲	۰.۵۵۳
Person	۰.۸۸۱	۰.۶۹۳	۰.۸۲۶	۰.۵۳۸
Gun	۰.۹۱۷	۰.۸۲۹	۰.۹۰۸	۰.۶۷۶
Knife	۰.۸۷۵	۰.۹۰۴	۰.۹۴۳	۰.۷۶۴
Fall	۰.۹۶۹	۰.۹۷۵	۰.۹۹۲	۰.۹۵۷
All	۰.۹۰۷	۰.۸۳	۰.۹۰۲	۰.۶۹۸

در نهایت در جدول (۷) با توجه به شاخص‌های ارزیابی از هر مدل، و توجه به نیاز در افزایش سرعت پردازش در ازای حفظ دقت در تشخیص اشیا تعلیمی، مدل float32 انتخاب شد و از آن به عنوان وزن اصلی برای پردازش استفاده کردیم. همچنین در مورد رزبری- پای از تکنیک افزایش فرکانس پردازنده^{۲۵} به منظور افزایش سرعت پردازشی، نیز بهره می‌بریم، هرچند این کار منجر به افزایش دمای پردازنده و مصرف توان بیشتر است. در این حالت نیاز به استفاده از یک خنک‌کننده قوی است.





جدول (۵): ارزیابی مدل best_integer_quant

Class Name	P	R	mAP 0.5	mAP 0.5:0.95
Head	۰.۷۲۵	۰.۶۸۲	۰.۷۱۵	۰.۳۸۲
Person	۰.۷۳۵	۰.۶۷۱	۰.۷۳۷	۰.۴۲۸
Gun	۰.۸۹	۰.۷۷۹	۰.۸۸۲	۰.۵۹۴
Knife	۰.۷۹۹	۰.۷۵۶	۰.۸۱۷	۰.۵۴۹
Fall	۰.۸۸۶	۰.۹۵۹	۰.۹۷۶	۰.۸۱۲
All	۰.۸۰۷	۰.۷۶۹	۰.۸۲۵	۰.۵۵۳

جدول (۶): ارزیابی مدل best_full_integer_quant

Class Name	P	R	mAP 0.5	mAP 0.5:0.95
Head	۰.۷۲۷	۰.۶۸۳	۰.۷۱۶	۰.۳۸۲
Person	۰.۷۳۴	۰.۶۷	۰.۷۳۶	۰.۴۲۶
Gun	۰.۸۹۷	۰.۷۵۴	۰.۸۷۵	۰.۵۹
Knife	۰.۷۹۲	۰.۷۵۳	۰.۸۱۱	۰.۵۴۴
Fall	۰.۸۷۹	۰.۹۵۶	۰.۹۷۴	۰.۸۱۱
All	۰.۸۰۶	۰.۷۶۳	۰.۸۲۲	۰.۵۵۱

شکل (۴) بیانگر ماتریس سردرگمی^{۲۶} نرمال شده است که عملکرد یک مدل طبقه‌بندی را نشان می‌دهد. هر خانه در ماتریس نشان‌دهنده نسبت پیش‌بینی‌های صحیح مثبت، غلط مثبت، صحیح منفی و غلط منفی است. شکل (۵) بیانگر نمودار معیار F1 است که رابطه بین امتیاز و سطح اطمینان را برای تشخیص هر رده‌بندی نشان می‌دهد. این نمودار از معیارهای ارزیابی شبکه‌های تعلیمی می‌باشد. به‌طور کلی، هر چه سطح اطمینان بالاتر باشد، امتیاز F1 بیشتر می‌شود؛ زیرا مدل، اشتباه کمتری در پیش‌بینی‌ها دارد.

جدول (۷): خلاصه‌ای از عملکرد وزن‌های تعلیم‌یافته و صادرشده در تشخیص اشیاء اختصاصی و ارزیابی سرعت پردازش در مورد رزبری پای

best وزن‌های خروجی	imgsz	Time inference (ms)	Average FPS	Average FPS with Overclock	mAP50-95 (imgsz = 640)
YOLOv8n	۳۲۰	۳۶۳.۷	≅۲	≅۲.۸	۰.۷۰۹
float32	۳۲۰	۲۴۰.۸	≅۳.۷	≅۴.۳	۰.۶۹۹
float16	۳۲۰	۲۳۷.۷	≅۳.۷	≅۴.۲	۰.۶۹۸
int8	۳۲۰	۲۳۹.۴	≅۳.۷	≅۴.۲	۰.۶۹۸
integer_quant	۳۲۰	۱۴۲.۳	≅۶.۱	≅۷.۱	۰.۵۵۳
full_integer_quant	۳۲۰	۱۳۶.۹	≅۶.۲	≅۷.۲	۰.۵۵۱

۴-۲- پیاده‌سازی سامانه امنیتی بر پایه شبکه داده همراه

ایده اساسی این بخش، نصب و راه‌اندازی سامانه امنیتی و اتصال به شبکه اینترنت از طریق شبکه‌های داده تلفن همراه است تا بتوان بدون نیاز به WiFi سامانه را مستقل از شرایط و در هر نقطه‌ای کنترل کرد. بومی‌سازی پنل مدیریتی و اتصال بر خط افزار طراحی شده و طراحی سرور اختصاصی از دیگر کارهای انجام شده است؛ به‌نحوی که تحریم یا فیلتر باعث از کار افتادن بخش نظارت از راه دور سامانه امنیتی نشود.

سامانه امنیتی، این امکان را دارد که نتایج پردازش الگوریتم تشخیص اشیاء را از طریق اینترنت به اطلاع کاربر خود برساند. در اینجا از ماژول سیم‌کارت به‌عنوان مودم سیم‌کارتی و تلفن‌کننده و سامانه پیامک استفاده می‌شود که وقوع حادثه و تهدید را به کاربر خود گزارش می‌دهد. به همین منظور در مورد رزبری پای از ماژول sim800 استفاده می‌شود. فناوری ارتباطی این ماژول بر پایه اینترنت 2G بوده و از طریق پروتکل^{۲۷} سریال به مورد رزبری پای متصل می‌شود. در شکل (۶) نحوه اتصال مورد، نمایش داده شده است.

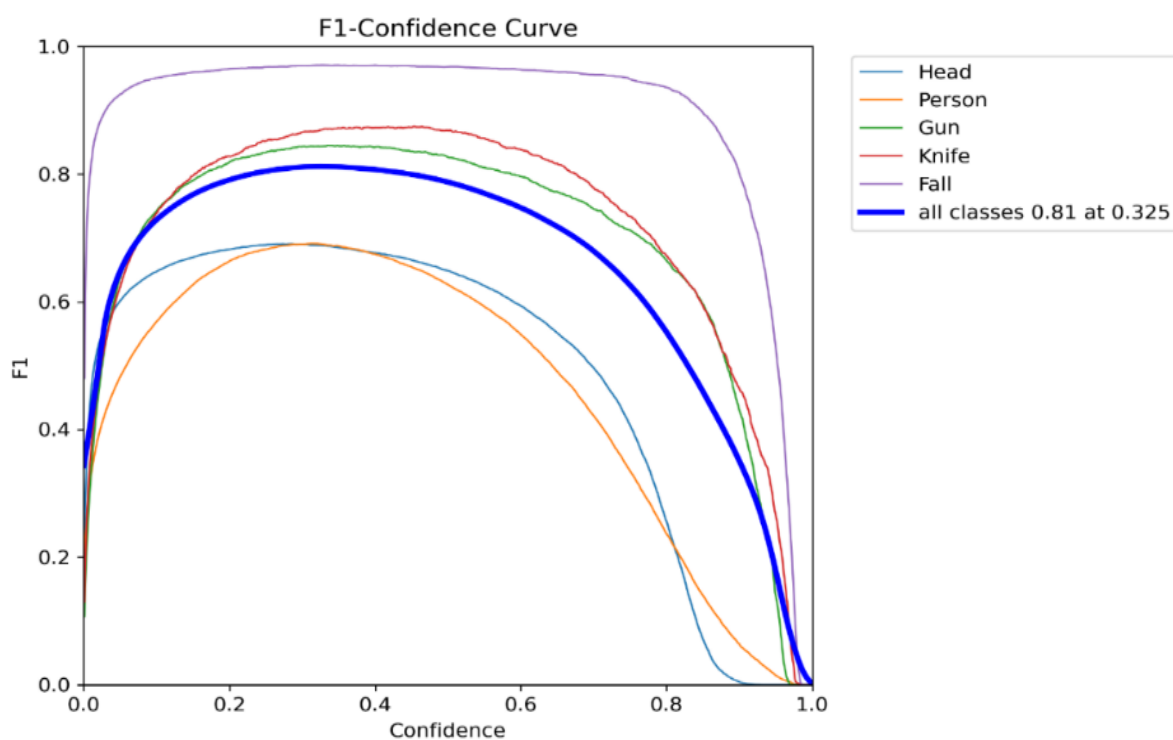
به‌منظور راه‌اندازی اینترنت سیم‌کارت در مورد رزبری پای از پروتکل نقطه‌به‌نقطه PPP^{۲۸} استفاده می‌شود؛ لذا با توجه به^{۲۹} کلیه فرایند راه‌اندازی این پروتکل برای ایجاد ارتباط با سرور و مدیریت برخط بدون نیاز به شبکه WiFi میسر شده است. به عبارتی می‌توان از طریق



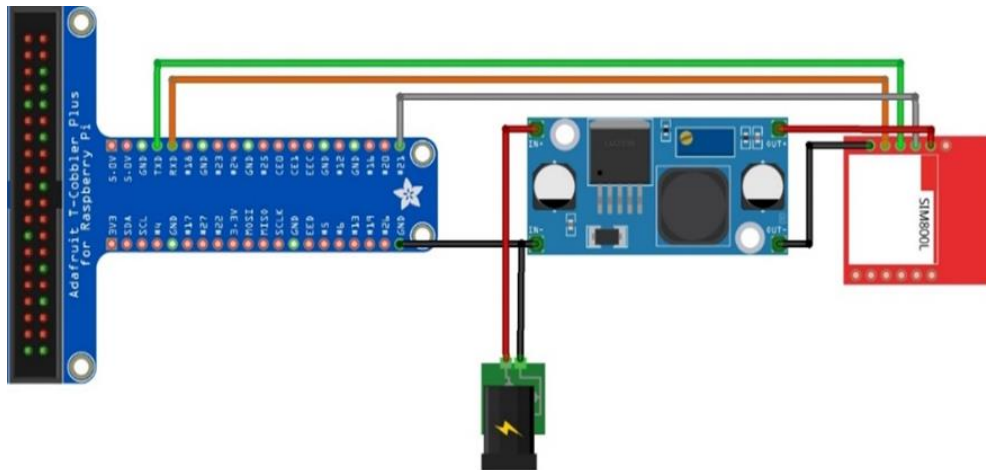
ارتباط سریال با ماژول سیم کارت، برد رزبری پای را به شبکه اینترنت و سرور اختصاصی متصل نمود. نکته مهم آن است که در فرایند برقراری تماس صوتی و ارسال پیامک هشدار، هنگامی که تهدیدات امنیتی شناسایی می‌شوند، لازم است داده همراه غیرفعال شود تا پورت سریال که توسط پروتکل PPP مشغول بوده، آزاد شود. اینک سامانه می‌تواند به برقراری تماس و ارسال اعلان هشدار بپردازد و مجدداً اینترنت داده همراه را فعال نماید و تصاویر ذخیره شده ناشی از تشخیص تهدیدات امنیتی را در سرور بارگذاری کند.



شکل (۴): نمودار بهنجارسازی ماتریس سردرگمی در ارزیابی مدل تعلیمی YOLOv8n



شکل (۵): نمودار F1-Confidence Curve در ارزیابی مدل تعلیمی YOLOv8n



شکل (۶): شماتیک اتصال sim800 به برد رزبری - پای

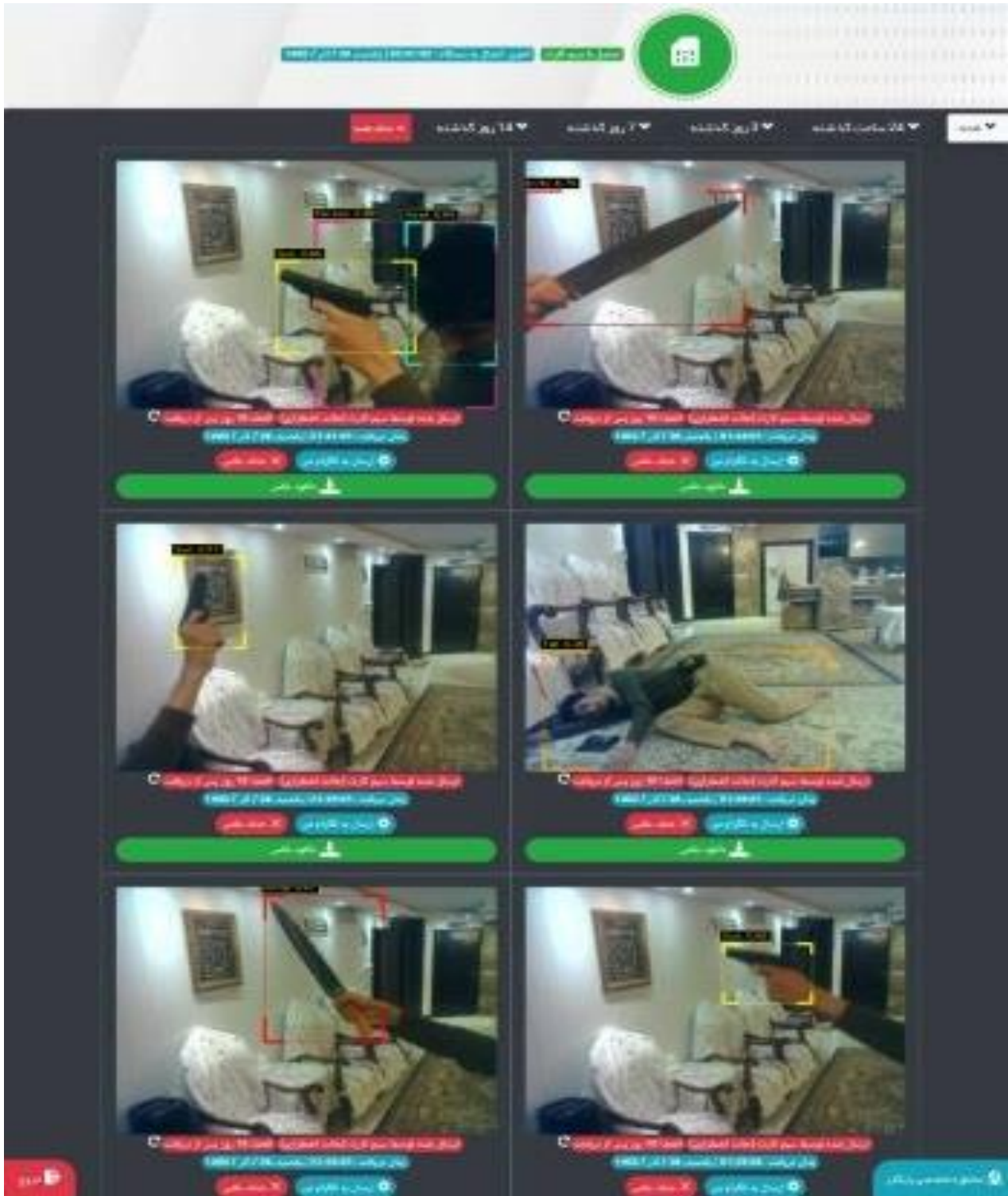
در نهایت با استفاده از پروتکل HTTP^{۴۰} و FTP^{۴۱} به مدیریت سامانه و ارسال محتوا می‌پردازیم. در پروتکل HTTP با استفاده از متدهای GET و POST به تبادل اطلاعات به منظور مدیریت آنلاین سامانه پرداخته می‌شود. این ساختار برای کارهایی مانند مدیریت فعال/ غیرفعال کردن سامانه امنیتی، درج شماره تلفن‌های اضطراری برای ارسال پیام هشدار و برقراری تماس می‌باشد. FTP یا پروتکل انتقال فایل، برای جابه‌جایی فایل‌ها بین رایانه‌های مختلف از طریق اینترنت استفاده می‌شود. به همین منظور، تصاویر ذخیره شده را از طریق اینترنت تلفن داده و به واسطه استفاده از این پروتکل بر روی سرور ارسال می‌کنیم.

لازم است اشاره شود که توابع اتصال به اینترنت و ارسال تصاویر و برقراری تماس و ارسال پیامک هشدار در سامانه به صورت پردازش‌های چندبخشی^{۴۲} نوشته شده است این امر به منظور اجرا کردن تمام زیرساخت‌های مورد نیاز پروژه سامانه امنیتی است تا در صورت شناسایی تهدید، سامانه نظارتی در یک حلقه، درگیر نباشد. در این روش بخش‌های مختلف از سامانه به صورت چند رشته هم‌زمان اجرا می‌شوند (پردازش موازی) و پردازنده با تقسیم کردن زمان پردازش بین رشته‌های مختلف به مدیریت سامانه می‌پردازد. شکل (۷) بیانگر تصاویر بارگذاری شده توسط سامانه نظارتی در پنل مدیریتی ناشی از شناسایی اشیاء تعلیمی در دوربین نظارتی است که به عنوان عوامل مخاطره‌آمیز و تهدیدات امنیتی در سامانه اطلاع‌رسانی بارگذاری شده است و به جهت نمایش رویکردها در اختیار کاربر قرار گرفته است.

۵- نتیجه‌گیری

در این مقاله سامانه امنیتی نظارتی مبتنی بر الگوریتم YOLO و فناوری اینترنت اشیا، با ارائه مزایایی مانند افزایش چشمگیر امنیت، بهره‌وری، مقیاس‌پذیری، پاسخگویی سریع و قابلیت اطمینان، در مقایسه با روش‌های سنتی معرفی شد تا گامی نوین در ارتقای امنیت اماکن و افراد برداشته شود. این سامانه با تشخیص دقیق و سریع انسان، سر انسان، تفنگ، چاقو و سقوط، و با قابلیت نظارت برخط، اتصال به اینترنت در هر نقطه، بارگذاری تصاویر و ارسال گزارش در صورت شناسایی تهدیدات، به عنوان یک راهکار ایده‌آل برای کاربردهای مختلف امنیتی شامل شناسایی انسان و اشیاء مخاطره‌آمیز، به عنوان راهبردی مناسب در حفاظت از سالمندان مطرح می‌شود. مشخصات فنی این سامانه شامل استفاده از الگوریتم تشخیص YOLOv8، دقت $mAP \cong 0.67$ ، سرعت پردازش $FPS \cong 4.3$ ، محاسبات ممیز شناور ۳۲ بیتی، مصرف انرژی کم و هزینه عملیاتی پایین، نشان‌دهنده کارایی و راندمان بالای آن است. هرچند مقایسه مدل‌های مختلف نشان می‌دهد که مدل‌های کمی‌سازی شده (ممیز شناور ۱۶ بیتی و ۸ بیتی صحیح) با افزایش سرعت پردازش، دقت را کاهش می‌دهند، در نهایت، مدل float32 به دلیل حفظ تعادل بین دقت و سرعت، برای پردازش انتخاب شده است. همچنین به منظور افزایش نرخ پردازش تصاویر نیز از تکنیک افزایش فرکانس پردازنده استفاده شد تا سرعت پردازش مدل تعلیمی در سخت‌افزار انتخابی حداکثر شود. با توجه به مزایا و مشخصات فنی ذکر شده، می‌توان نتیجه گرفت که سامانه امنیتی ارائه شده، یک راهکار نوین و کارآمد برای ارتقای امنیت اماکن و افراد است. این سامانه با اتکا به فناوری‌های پیشرفته، قادر به تشخیص دقیق و سریع تهدیدات و ارائه هشدارهای لازم است و قابلیت پیاده‌سازی در محیط‌های مختلف و ارتقا و توسعه را نیز دارا می‌باشد.





شکل (۷): تصاویر ارسال شده توسط سامانه در پنل مدیریتی

مراجع

- [1] K. Y. Loh and S. C. Reddy, "Understanding and preventing computer vision syndrome," *Malaysian Family Physician*, vol. 3, no. 3. Academy of Family Physicians of Malaysia, p. 128, 2008. Accessed: May 17, 2024. [Online]. Available: /pmc/articles/PMC4170366/
- [2] R. Hebbalaguppe, "A computer vision based approach for reducing false alarms caused by spiders and cobwebs in surveillance camera networks," 2014.
- [3] W. Aitfares, A. Kobbane, and A. Kriouile, *Suspicious behavior detection of people by monitoring*





- camera, vol. 0, pp. 113–117. doi: 10.1109/ICMCS.2016.7905601.
- [4] W. E. I. B. W. N. Afandi and N. M. Isa, "Object Detection: Harmful Weapons Detection Using YOLOv4," *IEEE Symp. Wirel. Technol. Appl. ISWTA*, vol. 2021-August, pp. 63–70, 2021, doi: 10.1109/ISWTA52208.2021.9587423.
- [5] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster R-CNN doing well for pedestrian detection?," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9906 LNCS, pp. 443–457, 2016, doi: 10.1007/978-3-319-46475-6_28/TABLES/5.
- [6] J. Li, X. Liang, S. Shen, T. Xu, J. Feng, and S. Yan, "Scale-Aware Fast R-CNN for Pedestrian Detection," *IEEE Trans. Multimed.*, vol. 20, no. 4, pp. 985–996, Apr. 2018, doi: 10.1109/TMM.2017.2759508.
- [7] S. Zhang, R. Benenson, and B. Schiele, "Filtered Channel Features for Pedestrian Detection", doi: 10.48550/arXiv.1501.05759.
- [8] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "How far are we from solving pedestrian detection?," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1259–1267. doi: 10.48550/arXiv.1602.01237.
- [9] J. Hosang, M. Omran, R. Benenson, and B. Schiele, "Taking a Deeper Look at Pedestrians." pp. 4073–4082, 2015. doi: 10.48550/arXiv.1501.05790.
- [10] P. Dollar, Z. Tu, P. Perona, and S. Belongie, "Integral Channel Features," in *Proceedings of the British Machine Vision Conference 2009*, British Machine Vision Association, 2009, pp. 91.1–91.11. doi: 10.5244/C.23.91.
- [11] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1532–1545, 2014, doi: 10.1109/TPAMI.2014.2300479.
- [12] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," pp. 304–311, Mar. 2010, doi: 10.1109/CVPR.2009.5206631.
- [13] M. Fabbri, G. Brasó, G. Maugeri, O. Cetintas, R. Gasparini, A. Ošep, S. Calderara, L. Leal-Taixé, and R. Cucchiara, "MOTSynth: How Can Synthetic Data Help Pedestrian Detection and Tracking?," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 10829–10839, Aug. 2021, doi: 10.1109/ICCV48922.2021.01067.
- [14] J. Mao, T. Xiao, Y. Jiang, and Z. Cao, "What Can Help Pedestrian Detection?" pp. 3127–3136, 2017. doi: 10.48550/arXiv.1705.02757.
- [15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.
- [16] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, Sep. 2014, doi: 10.48550/arXiv.1409.1556.
- [17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." pp. 770–778, 2016. doi: 10.48550/arXiv.1512.03385.
- [18] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9908 LNCS, pp. 354–370, 2016, doi: 10.1007/978-3-319-46493-0_22/FIGURES/8.
- [19] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen, "Repulsion Loss: Detecting Pedestrians in a Crowd." pp. 7774–7783, 2018. doi: 10.48550/arXiv.1711.07752.
- [20] R. K. Tiwari and G. K. Verma, "A Computer Vision based Framework for Visual Gun Detection Using Harris Interest Point Detector," *Procedia Comput. Sci.*, vol. 54, pp. 703–712, Jan. 2015, doi: 10.1016/J.PROCS.2015.06.083.
- [21] H. Jain, A. Vikram, Mohana, A. Kashyap, and A. Jain, "Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications," *Proc. Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2020*, pp. 193–198, Jul. 2020, doi: 10.1109/ICESC48915.2020.9155832.
- [22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 9905 LNCS, pp. 21–37, 2016, doi: 10.1007/978-3-319-46448-0_2/FIGURES/5.
- [23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. , Accessed: May 17, 2024. [Online]. Available: <https://github.com/>



- [24] T. S. S. Hashmi, N. U. Haq, M. M. Fraz, and M. Shahzad, "Application of Deep Learning for Weapons Detection in Surveillance Videos," *2021 Int. Conf. Digit. Futur. Transform. Technol. ICoDT2 2021*, May 2021, doi: 10.1109/ICODT252288.2021.9441523.
- [25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection." pp. 779–788, 2016. doi: 10.48550/arXiv.1506.02640.
- [26] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020, doi: 10.48550/arXiv.2004.10934.
- [27] A. Singh, T. Anand, S. Sharma, and P. Singh, "IoT Based Weapons Detection System for Surveillance and Security Using YOLOV4," *Proc. 6th Int. Conf. Commun. Electron. Syst. ICCES 2021*, pp. 488–493, Jul. 2021, doi: 10.1109/ICCES51350.2021.9489224.
- [28] M. T. Bhatti, M. G. Khan, M. Aslam, and M. J. Fiaz, "Weapon Detection in Real-Time CCTV Videos Using Deep Learning," *IEEE Access*, vol. 9, pp. 34366–34382, 2021, doi: 10.1109/ACCESS.2021.3059170.
- [29] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating Very Deep Convolutional Networks for Classification and Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 1943–1955, Oct. 2016, doi: 10.1109/TPAMI.2015.2502579.
- [30] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision." pp. 2818–2826, 2016. doi: 10.48550/arXiv.1512.00567.
- [31] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, pp. 4278–4284, Feb. 2017, doi: 10.1609/aaai.v31i1.11231.
- [32] S. Barratt and R. Sharma, "A Note on the Inception Score," Jan. 2018, doi: 10.48550/arXiv.1801.01973.
- [33] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," Apr. 2017, doi: 10.48550/arXiv.1704.04861.
- [34] Y. Harjoseputro, I. P. Yuda, and K. P. Danukusumo, "MobileNets: Efficient Convolutional Neural Network for Identification of Protected Birds," *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 10, no. 6, pp. 2290–2296, Dec. 2020, doi: 10.18517/ijaseit.10.6.10948.
- [35] İ. Karakaya, I. Şafak, O. Öztürk, M. Bal, and Y. E. Esin, "Gun Detection with Faster R-CNN in X-Ray Images," in *2020 28th Signal Processing and Communications Applications Conference (SIU)*, IEEE, Oct. 2020, pp. 1–4. doi: 10.1109/SIU49456.2020.9302457.
- [36] J. Lim, M. I. Al Jobayer, V. M. Baskaran, J. M. Lim, K. Wong, and J. See, "Gun detection in surveillance videos using deep neural networks," *2019 Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. APSIPA ASC 2019*, pp. 1998–2002, Nov. 2019, doi: 10.1109/APSIPAASC47483.2019.9023182.
- [37] S. Shao, Z. Zhao, B. Li, T. Xiao, G. Yu, X. Zhang, and J. Sun, "CrowdHuman: A Benchmark for Detecting Human in a Crowd," Apr. 2018, doi: 10.48550/arXiv.1805.00123.
- [38] N. Yu and J. Lv, "Human body posture recognition algorithm for still images," *J. Eng.*, vol. 2020, no. 13, pp. 322–325, 2020, doi: 10.1049/joe.2019.1146.
- [39] G. Santos, P. Endo, K. Monteiro, E. Rocha, I. Silva, and T. Lynn, "Accelerometer-Based Human Fall Detection Using Convolutional Neural Networks," *Sensors*, vol. 19, no. 7, p. 1644, Apr. 2019, doi: 10.3390/s19071644.
- [40] B. LUO and L. U. O. Bo, "Human Fall Detection for Smart Home Caring using Yolo Networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 4, p. 2023, 2023, doi: 10.14569/IJACSA.2023.0140409.
- [41] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2015, pp. 437–446. doi: 10.48550/arXiv.1409.5403.
- [42] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. Da Silva, "A Comparative Analysis of Object Detection Metrics with a Companion Open-Source Toolkit," *Electron. 2021, Vol. 10, Page 279*, vol. 10, no. 3, p. 279, Jan. 2021, doi: 10.3390/ELECTRONICS10030279.
- [43] P. Henderson and V. Ferrari, "End-to-End Training of Object Class Detectors for Mean Average Precision," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10115 LNCS, pp. 198–213, 2017, doi: 10.1007/978-3-319-54193-8_13.
- [44] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object





- classes (VOC) challenge,” *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/S11263-009-0275-4/METRICS.
- [45] E. Bisong, “Building Machine Learning and Deep Learning Models on Google Cloud Platform,” *Build. Mach. Learn. Deep Learn. Model. Google Cloud Platf.*, 2019, doi: 10.1007/978-1-4842-4470-8.
- [46] H. K. Jabbar and R. Z. Khan, “Methods to Avoid Over-Fitting and Under-Fitting in Supervised Machine Learning (Comparative Study),” in *Computer Science, Communication and Instrumentation Devices*, Singapore: Research Publishing Services, 2014, pp. 163–172. doi: 10.3850/978-981-09-5247-1_017.
- [47] K. Raza, H. Song, and S. Hong, “Fast and Accurate Fish Detection Design with Improved YOLO-v3 Model and Transfer Learning,” *Artic. Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 2, 2020, doi: 10.14569/IJACSA.2020.0110202.

زیر نویس ها

-
- ¹ Convolutional Neural Network
 - ² Single Shot Detector
 - ³ Fast Region-based Convolutional Neural Network
 - ⁴ You Only Look Once
 - ⁵ Closed-circuit television
 - ⁶ Visual Geometry Group 16-layer
 - ⁷ mean Average Precision
 - ⁸ Multi-Scale Multi-Task Detection
 - ⁹ Resolution
 - ¹⁰ Human Activity Recognition
 - ¹¹ Global
 - ¹² Regression
 - ¹³ Deformable part models
 - ¹⁴ Classification
 - ¹⁵ Bounding box
 - ¹⁶ anchor-free
 - ¹⁷ Intersection over Union
 - ¹⁸ True Positives
 - ¹⁹ False Positives
 - ²⁰ False Negatives
 - ²¹ Average Precision
 - ²² Frames Per Second
 - ²³ Google Colab
 - ²⁴ Backup
 - ²⁵ Google Drive
 - ²⁶ export
 - ²⁷ Framework
 - ²⁸ Platform
 - ²⁹ Quantization
 - ³⁰ <https://docs.ultralytics.com/modes/export/>
 - ³¹ Overfitting
 - ³² Underfitting
 - ³³ Transfer learning
 - ³⁴ Hang
 - ³⁵ Overclock
 - ³⁶ Confusion Matrix Normalize
 - ³⁷ Protocol
 - ³⁸ Point-to-Point Protocol
 - ³⁹ <https://pnatraj.medium.com/raspberry-piintegration-with-gsm-gprs-sim800a-6d0ed5b5e76b>
 - ⁴⁰ Hypertext Transfer Protocol
 - ⁴¹ File Transfer Protocol
 - ⁴² Multi-Threading

