

ارائه یک روش مبتنی بر شبکه‌های یادگیری عمیق به منظور تشخیص پهباد از داده‌های عمق

فرناز اشتري ماهيني^{۱*}، فرهاد صمدزادگان^۲، فرزانه دادرسی جوان^۳

^۱ کارشناس ارشد فتوگرامتری - گروه مهندسی نقشه‌برداری - پردیس دانشکده‌های فنی - دانشگاه تهران
f.ashtari@ut.ac.ir

^۲ استاد گروه مهندسی نقشه‌برداری - پردیس دانشکده‌های فنی - دانشگاه تهران
samadz@ut.ac.ir

^۳ استادیار گروه مهندسی نقشه‌برداری - پردیس دانشکده‌های فنی - دانشگاه تهران
fdadrasjavan@ut.ac.ir

(تاریخ تصویب اسفند ۱۴۰۰، تاریخ تصویب تیر ۱۴۰۱)

چکیده

امروزه پهپادها به دلیل مزایای فراوان و طیف گسترده‌ای از کاربردها از قبیل نقشه‌برداری، کشاورزی و مدیریت بحران بسیار مورد توجه قرار گرفته‌اند و جایگزین بسیاری از روش‌های سنتی موجود گردیده‌اند. با این وجود، گسترش پهپادها و ورود غیرمجاز آن‌ها به زیرساخت‌های مهم مانند ساختمان‌های دولتی می‌تواند زمینه تهدیدهای احتمالی را در تأمین امنیت عمومی جامعه ایجاد کند. از این رو، تدابیر متقابل نظیر به‌کارگیری سامانه‌های تشخیص، محلی‌سازی و رهگیری پهپادهای کوچک بسیار حائز اهمیت است. در سال‌های اخیر، روش‌های تشخیص مبتنی بر شبکه‌های عصبی و رویکردهای یادگیری عمیق، توانایی قابل توجهی در حوزه‌ی تشخیص پهپاد از خود نشان داده‌است. از این رو، در این تحقیق از یک روش تشخیص مبتنی بر شبکه‌های یادگیری عمیق پیچشی به منظور تشخیص پهپاد استفاده شده‌است. از سوی دیگر، استفاده از تصاویر مرئی با مشکلاتی از قبیل وجود نواحی پنهان، پیش‌زمینه شلوغ و عدم امکان جداسازی پیش‌زمینه و مشکلات نور داخل تصویر مواجه هستند. هم‌چنین، سامانه‌های مبتنی بر تصویر حرارتی، با وجود بهره‌مندی از قدرت دید در شب، دارای قدرت تفکیک مکانی کمتری نسبت به تصویر مرئی می‌باشند که روند تشخیص پهپاد را با مشکل دقت مواجه می‌کند. اخیراً، استفاده از تصاویر عمق که چالش‌های مرتبط با تصاویر مرئی را ندارند و نشان‌دهنده‌ی میزان دوری و نزدیکی شیء تا دوربین هستند، بسیار مورد استقبال قرار گرفته‌است. در این پژوهش، با استفاده از مجموعه تصاویر عمق شبیه‌سازی و واقعی و با استفاده از شبکه یادگیری عمیق YOLO (You Only Look Once) به تشخیص پهپاد، پرداخته شده‌است. تصاویر عمق واقعی در این پژوهش از طریق الگوریتم تناظر یابی شبه سراسری (SGM) به دست آمده و سرانجام اعتبارسنجی مدل آموزش‌دیده، با انواع تصاویر عمق واقعی و شبیه‌سازی با سه نوع پهپاد در محیط داخل و خارج بررسی شده‌است. در نهایت، نتایج حاصل از تشخیص پهپاد با شبکه یادگیری عمیق موردنظر در تصاویر عمق شبیه‌سازی به میانگین دقت ۸۴ درصد، میانگین زمان ۱۲۵ میلی‌ثانیه و در تصاویر عمق واقعی به میانگین دقت ۷۴ درصد، میانگین زمان ۱۳۳ میلی‌ثانیه رسید.

واژگان کلیدی: تشخیص پهپاد، تصاویر عمق، یادگیری عمیق، شبکه یادگیری عمیق YOLO، الگوریتم تناظر یابی SGM

۱- مقدمه



شکل ۱- چالش‌های مطرح در تشخیص پهپاد در تصاویر مرئی

در این پژوهش با استفاده از مجموعه تصاویر عمق و همچنین با استفاده از شبکه‌های یادگیری عمیق، به تشخیص پهپاد پرداخته شده است.

تصاویر عمق، تصاویری هستند که هر پیکسل از آن‌ها حاوی اطلاعات فاصله از دوربین و موقعیت شیء است. در اصل، یک نقشه عمق، یک تصویر دوبعدی به رنگ خاکستری است که به واسطه شدت رنگ، عمق را قابل درک می‌کند. به بیان دیگر، مناطق تاریک‌تر در نقشه‌ی عمق نشان‌دهنده شدت دور بودن شیء است که این رنگ با کاهش عمق به تدریج روشن شده و در نهایت برای اشیاء نزدیک‌تر سفید می‌شود [۱۵]. کلید اصلی تشخیص پهپاد در تصاویر عمق، اختلاف شیء با پس‌زمینه است. این اختلاف، تفاوت عمق شیء در حال حرکت با پیش‌زمینه و محیط اطراف است که به تشخیص و ردیابی کمک می‌کند؛ چرا که اطراف پهپاد در حال پرواز، شیء دیگری با عمق یکسان وجود ندارد و وجود پهپاد در پیش‌زمینه، موجب ایجاد یک ناپیوستگی در نقشه عمق می‌شود که می‌تواند در تشخیص شیء مؤثر واقع شود [۱۴].

نقشه‌های عمق به دلیل ارائه اطلاعات موقعیتیایی اشیاء در صحنه سه‌بعدی، در بسیاری از کاربردهای بصری استفاده شده و منبع اصلی اطلاعات برای حل تعداد زیادی از مسائل چالش برانگیز است [۱۶]. در گذشته، پیشرفت‌های قابل توجهی در زمینه پردازش، تجزیه، تحلیل و تفسیر داده‌های عمق انجام شده است [۱۷]. امروزه، داده‌های عمق نقش مهمی در کاربردهایی مانند تجزیه و تحلیل حرکت، جلوگیری از برخورد با مانع، حفظ میراث فرهنگی، تشخیص انسان [۱۸] و تشخیص حرکت [۱۹] دارند. همچنین، درک اطلاعات سه‌بعدی صحنه یا شیء برای انجام کارهای متعدد در بسیاری از برنامه‌های رباتیک و بینایی ماشین ضروری است [۱۷]. به دلیل مزایا، ویژگی‌های ذاتی و عدم وجود مشکلات مرتبط با تصاویر مرئی و حرارتی در تصاویر عمق، این تصاویر می‌توانند برای تحقق اهداف تشخیص شیء مورد استفاده قرار گیرند [۱۴].

در سال‌های اخیر، استفاده از پهپادها به دلیل مزایای فراوان در کاربردهایی نظیر مدیریت منابع، مدیریت بحران، جستجو و نجات، نقشه‌برداری و تأمین امنیت بسیار مورد توجه قرار گرفته است [۱]. با این وجود، این پرنده‌ها می‌توانند به عنوان تهدیدی جدی در مناطق امنیتی و نظامی محسوب شوند [۲، ۳]. بنابراین، ارائه یک رویکرد کارآمد به منظور تشخیص پهپاد از اهمیت بالایی برخوردار است. چنین فن‌آوری‌هایی می‌توانند در سامانه‌های امنیتی و نظامی مستقر شوند تا از ورود پهپادها به زیرساخت‌های مهم (ساختمان‌های دولتی، نیروگاه‌ها و پالایشگاه‌ها) جلوگیری کنند و یا امنیت را در اماکن بزرگ همانند ورزشگاه‌ها تأمین کنند. به همین دلیل، تهدیدات ناشی از این پرنده و تأمین امنیت مکان‌های حساس، لزوم پرداختن به مسئله‌ی شناسایی و تشخیص پهپاد را چندین برابر می‌کند. از روش‌های سنتی شناسایی حضور پهپاد می‌توان به استفاده از حسگرهای مختلفی مانند رادار [۴]، لایدار [۵]، حسگرهای صوتی [۶] و حرارتی [۷] اشاره کرد. اما استفاده از این حسگرها همواره با مشکلاتی از قبیل هزینه و مصرف انرژی بالا همراه بوده است [۴]. بر این اساس، بحث استفاده از روش‌های نوین تشخیص شیء مطرح می‌شود؛ در دهه‌ی گذشته، شبکه‌های یادگیری عمیق به دلیل قدرت محاسباتی و دقت بالاتر به یکی از متداول‌ترین و جدیدترین روش‌های مبتنی بر یادگیری ماشین برای کاربردهای مختلف و پردازش بصری و شناسایی شیء تبدیل شده‌اند [۸-۱۲].

با توجه به تحقیقات انجام‌شده و ابعاد کوچک طیف گسترده‌ای از این پهپادها، تشخیص و موقعیتیایی آن‌ها در محیط‌های مختلف با چالش‌هایی همراه است. مطابق شکل (۱) می‌توان به چالش‌هایی نظیر وجود نواحی پنهان، پس‌زمینه شلوغ، عدم امکان جداسازی پس‌زمینه و مشکلات نور داخل تصاویر مرئی اشاره کرد [۱۳، ۱۴]. همچنین، سامانه‌های مبتنی بر تصویر حرارتی دارای قدرت تفکیک مکانی کمتری نسبت به تصویر مرئی می‌باشند. اما نقشه‌هایی موسوم به تصاویر عمق^۱، چالش‌های مطرح در تصاویر مرئی و تصاویر حرارتی را ندارند [۱۴].

^۱ Depth image

۱-۱- تشخیص اشیاء با شبکه‌ی یادگیری عمیق

یکی از نماینده‌های اصلی در شبکه‌های یادگیری عمیق، شبکه عصبی پیچشی^۱ است که از پشت سرهم قرار گرفتن لایه‌های پیچشی و ادغام بیشینه^۲ تشکیل شده است [۲۰]. چارچوب روش‌های عمومی تشخیص شیء را می‌توان به طور عمده در دو نوع دسته‌بندی کرد: دسته اول، روش‌های تشخیص شیء مبتنی بر نواحی و دسته دوم، روش‌های مبتنی بر طبقه‌بندی و رگرسیون [۲۰].

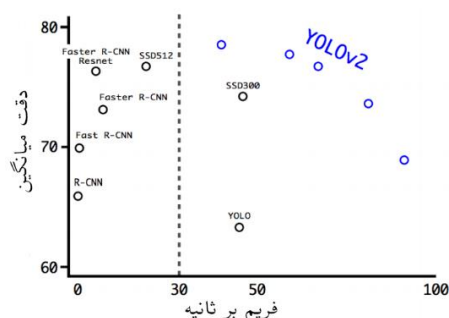
۱-۱-۱- روش تشخیص مبتنی بر نواحی

در این دسته روش‌ها، در ابتدا یک سری نواحی پیشنهادی در نظر گرفته می‌شود و هر یک از این نواحی در دسته‌های مختلف طبقه‌بندی می‌شود [۲۱]. روش‌های مبتنی بر پیشنهاد ناحیه عمدتاً شامل RCNN^۳ [۲۲]، SPP-net^۴ [۲۳]، FastRCNN^۵ [۲۴]، FasterRCNN^۶ [۲۵]، R-FCN^۷ [۲۶] و FPN^۸ [۲۷] هستند. الگوریتم‌های تشخیص شیء مبتنی بر نواحی، به‌جای در نظر گرفتن کل تصویر، تنها بخشی از نواحی جدا شده از آن تصویر که احتمال وجود شیء در آن بیشتر است را بررسی می‌کنند.

۱-۱-۲- روش تشخیص مبتنی بر طبقه‌بندی و رگرسیون

تشخیص شیء با استفاده از طبقه‌بندی و رگرسیون، اتخاذ یک چارچوب واحد برای دستیابی مستقیم به نتایج نهایی (دسته شیء و موقعیت) است. روش‌های مبتنی بر رگرسیون و طبقه‌بندی، عمدتاً شامل مدل‌های MultiBox [۲۸]، G-CNN^۹ [۲۹]، YOLO^{۱۰} [۳۰]، SSD^{۱۱} [۳۱]، YOLOv2^{۱۲} [۳۲]، YOLOv3^{۱۳} [۳۳]، DSSD^{۱۴} [۳۴]، DSOD^{۱۵} [۳۵] است. بر خلاف روش‌های مبتنی بر نواحی، در روش‌های

تشخیص مبتنی بر طبقه‌بندی و رگرسیون، در طول زمان آموزش و ارزیابی شبکه، تصویر به صورت کامل بررسی می‌شود و شبکه به صورت یکجا کل تصویر را بررسی می‌کند. با توجه به مزیت روش‌های تشخیص شیء در دسته دوم، به دلیل بررسی تصویر به‌صورت یکجا و از طرفی دیگر به دلیل سرعت و دقت بالا در شبکه یادگیری عمیق YOLO، از این شبکه به منظور تشخیص پهباد استفاده می‌شود. شبکه‌ی یادگیری عمیق YOLO^{۱۶} [۳۰] یک الگوریتم تشخیص شیء است که با بررسی کلی تصویر، محدوده حاوی شیء و احتمال تعلق شیء به کلاس مربوطه را ارائه می‌دهد. تا به امروز، نسخه‌های زیادی به منظور افزایش دقت و سرعت شبکه YOLO ارائه شده‌است. طی مطالعات انجام‌شده، نسخه دوم این شبکه [۳۲] از لحاظ سرعت، در صدر دیگر نسخه‌های این شبکه قرار داشته و در فعالیت‌هایی که سرعت تشخیص شیء دارای اهمیت بیشتری است، کاربرد دارد (شکل (۲)). به همین دلیل، از نسخه دوم شبکه یادگیری عمیق YOLO برای تشخیص پهباد در این پژوهش استفاده شده‌است.



شکل ۲- مقایسه‌ی سرعت و دقت الگوریتم‌های تشخیص شیء

هدف از این تحقیق، ارائه یک رویکرد کارآمد در زمینه تشخیص پهباد با استفاده از تصاویر عمق و روش شبکه یادگیری عمیق YOLOv2 است.

سازمان‌دهی ادامه مقاله به صورت زیر است: بخش دوم به بررسی تحقیقات انجام‌شده اختصاص دارد. در بخش سوم روش پیشنهادی و روش تولید تصاویر عمق ارائه شده است. در بخش چهارم به بررسی نتایج حاصل از آزمایش و تشخیص پهباد با شبکه YOLOv2 پرداخته شده‌است و در نهایت در بخش پنجم نتیجه‌گیری و راهکارهای آتی ارائه شده‌است.

^{۱۶} YOLO Deep Learning Network

^۱ Convolutional Neural Network (CNN)

^۲ Max-Pooling

^۳ Region Based Convolutional Neural Networks (RCNN)

^۴ Spatial Pyramid Pooling network (SPP-net)

^۵ Fast Region Based Convolutional Neural Networks (Fast-RCNN)

^۶ Faster Region Based Convolutional Neural Networks (Faster-RCNN)

^۷ Region-based Fully Convolutional Networks

^۸ Feature Pyramid Network (FPN)

^۹ Geometric Convolutional Neural Network (GCNN)

^{۱۰} You Only Look Once (YOLO)

^{۱۱} Single Shot Detector (SSD)

^{۱۲} You Only Look Once – Version 2 (YOLOv2)

^{۱۳} You Only Look Once – Version 3 (YOLOv3)

^{۱۴} Deconvolutional Single Shot Detector (DSSD)

^{۱۵} Deeply Supervised Object Detector (DSOD)

۲- مروری بر تحقیقات انجام شده

در سال‌های اخیر تحقیقات متنوعی در زمینه تشخیص پهپاد توسط محققین انجام شده است. به منظور بررسی دقیق‌تر، این تحقیقات در چهار گروه دسته‌بندی شده‌اند: (۱) تشخیص پهپاد با روش‌های مرسوم، (۲) تشخیص پهپاد با مجموعه داده عمق، (۳) تشخیص پهپاد با شبکه یادگیری عمیق و (۴) تشخیص پهپاد با داده‌های عمق و شبکه‌های یادگیری عمیق.

۲-۱- تشخیص پهپاد با روش‌های مرسوم

زدرویت^۱ و همکاران [۳۶] در سال ۲۰۱۱ میلادی با استفاده از تصاویر مرئی و استفاده از الگوریتم آشکارساز پهپاد به صورت بصری در شرایط آب و هوایی مناسب موفق به تشخیص پهپاد شدند. وو^۲ و همکاران [۳۷] در سال ۲۰۱۷، با استفاده از تصاویر مرئی به تشخیص پهپاد پرداختند. در این تحقیق، ابتدا با استفاده از تشکیل نقشه برجستگی^۳، وجود یا عدم وجود پهپاد بررسی شده و سپس با استفاده از فیلتر کالمن به پیش‌بینی وضعیت پهپاد پرداخته شد. همچنین، در سال ۲۰۱۱، لای^۴ و همکاران [۳۸] در مقاله‌ای با استفاده از فیلترهای مورفولوژیکی به منظور جلوگیری از برخورد به تشخیص پهپاد پرداختند. در این مقاله، توسعه و ارزیابی یک سیستم تشخیص مبتنی بر دید^۵، به صورت آنی و مناسب برای انواع پهپاد شرح داده شد. از دیگر تحقیقات انجام شده در این بخش، پژوهش انجام شده توسط ده‌گ^۶ و همکاران [۳۹] در سال ۲۰۱۶ است که با استفاده از حسگرهای لایدار و رادار موفق به تشخیص پهپاد شدند. در این مقاله مجموعه‌ای از آزمایش‌های زمینی و پروازی برای ارزیابی عملکرد حسگرهای کوچک لایدار و رادار انجام شد و دامنه تشخیص مانع در برابر اندازه آن برای هر دو حسگر در حالت پرواز استاتیک و دینامیک تعیین شد. در سال ۲۰۱۷ اندرسی^۷ و همکاران [۷] در مقاله‌ای به تشخیص پهپاد با استفاده از دوربین مادون قرمز حرارتی پرداختند. در این تحقیق، سه

پهپاد انتخابی در امتداد سه مسیر آزمایشی پرواز کرد و عملیات تشخیص پهپاد انجام شد. نکته مورد توجه این است که روش‌های ارائه شده، بر وجود تضاد کافی بین پس‌زمینه و شیء استوار است و تنها در این صورت، روش تشخیص بر اساس تصویر مرئی کارآمد بوده و به خوبی توانایی تشخیص پهپاد را دارد. اما صرفاً انرژی بالایی داشته و ممکن است در ادغام با پهپادهای کوچک محدودیت داشته باشد.

۲-۲- تشخیص پهپاد با مجموعه داده‌ی عمق

تصاویر عمق، تصاویری هستند که هر پیکسل در تصویر نهایی، حاوی اطلاعات موقعیت و فاصله شیء از دوربین است. این تصاویر از حسگرهای مختلفی همانند حسگر لایدار، حسگرهای مرئی همراه عمق^۸ و دوربین‌های سه‌بعدی به دست می‌آیند [۴۰، ۱۵]. در این راستا، لویز^۹ و همکاران [۴۱] در سال ۲۰۱۷ با استفاده از ابر نقطه سه‌بعدی^{۱۰} و پهپاد مجهز شده به دوربین‌های مرئی به همراه عمق، به تشخیص موانع بدون حرکت پرداختند. تصاویر عمق می‌توانند جایگزین بسیار مناسبی برای ابر نقطه باشند، چرا که اطلاعات موجود در تصاویر عمق به صورت عمده بسیاری از مشکلات را ساده‌تر کرده و چالش‌های بسیار زیادی را در بینایی کامپیوتر و تشخیص شیء [۴۲] برطرف کرده‌است. همچنین، سانگ^{۱۱} و همکاران [۴۲] در سال ۲۰۱۴ با استفاده از مدل‌های سه‌بعدی، تصاویر عمق شبیه‌سازی و روش طبقه‌بندی ماشین بردار پشتیبان^{۱۲} (SVM) به تشخیص پهپاد پرداختند. با توجه به تحقیقات انجام شده در موضوع تشخیص پهپاد با استفاده از داده‌های عمق، می‌توان گفت تشخیص پهپاد در مجموعه ابرنقطه و مدل‌های سه بعدی نیازمند استفاده از روش‌های نوین با سرعت بالاست؛ به همین دلیل، استفاده از روش‌های یادگیری عمیق می‌تواند چالش‌های مرتبط با سرعت پایین تشخیص در این دسته از مطالعات را برطرف کند.

۲-۳- تشخیص پهپاد با شبکه‌ی یادگیری عمیق

یکی از وجوه تفاوت مطالعات در زمینه تشخیص پهپاد، رویکردهای تشخیص آن است. به طوری که سقیب^{۱۳} و

^۸ RGB-D Camera

^۹ Lopez

^{۱۰} 3D point cloud

^{۱۱} Song

^{۱۲} SVM

^{۱۳} Saqib

^۱ Zsedrovits

^۲ WU

^۳ Saliency Map

^۴ Lai

^۵ Vision-based

^۶ De Haag

^۷ Andradi

تشخیص پهپاد برآمده است. در ادامه به توضیح روش پیشنهادی استفاده شده در این مقاله پرداخته شده است.

۳- روش پیشنهادی

در این تحقیق، به دلیل توانایی شبکه های یادگیری عمیق و همچنین سرعت شبکه YOLO در تشخیص اشیاء، از نسخه دوم این شبکه به منظور تشخیص پهپاد استفاده شده است. همچنین، به دلیل مزایای تصاویر عمق نسبت به دیگر تصاویر، از آن ها به عنوان ورودی روش مورد مطالعه استفاده شده است.

تصاویر عمق استفاده شده، شامل تصاویر عمق شبیه سازی و واقعی هستند. تصاویر عمق شبیه سازی از محیط مجازی ایرسیم^۶ [۱۴] و تصاویر عمق واقعی از طریق الگوریتم تناظریابی شبه سراسری^۷ (SGM) بدست آمده اند. مطابق روندنمای شکل (۳)، روش پیشنهادی شامل سه قسمت آماده سازی داده های عمق، آموزش شبکه و آزمایش مدل یادگیری عمیق پیشنهادی است.

ساختار الگوریتم تشخیص پهپاد از داده های عمق با استفاده از شبکه یادگیری عمیق به شرح زیر است:

- i. آماده سازی تصاویر عمق
 - ii. آموزش شبکه
 - عملکرد و معماری شبکه یادگیری عمیق
 - خروجی شبکه
 - تابع زیان
 - iii. آزمایش شبکه
 - بررسی معیار ارزیابی در تشخیص پهپاد
- در ابتدا تصاویر عمق شبیه سازی وارد بخش آماده سازی تصاویر عمق می شوند. سپس برای ورود به مرحله آموزش شبکه، این دسته از تصاویر به دو دسته آموزشی و آزمایشی تقسیم بندی می شوند و در نهایت پس از آماده سازی مدل با استفاده از ۸۰ درصد از داده های عمق شبیه سازی، مرحله آموزش مدل انجام می شود. سپس ۲۰ درصد باقیمانده تصاویر به همراه دیگر تصاویر عمق شبیه سازی و داده های عمق واقعی حاصل از الگوریتم تناظریابی SGM، به منظور ارزیابی دقت و بررسی تعمیم پذیری شبکه در مرحله آزمایش استفاده می شوند.

همکاران [۴۳] در سال ۲۰۱۷ در پژوهشی با استفاده از شبکه های یادگیری عمیق پیشی در تصاویر مرئی، به تشخیص پهپاد پرداختند. در این پژوهش، آزمایش هایی با معماری شبکه های مختلف مبتنی بر شبکه عصبی پیشی ZF^۱ و VGG16^۲ انجام شده است. از طرفی، به دلیل اهمیت موضوع سرعت و دقت در تشخیص پهپاد، اگر^۳ و همکاران [۴۴] در سال ۲۰۱۷ با استفاده از شبکه یادگیری عمیق YOLO در مجموعه تصاویر شبیه سازی مرئی، موفق به تشخیص پهپاد شدند. علاوه بر این، ننجو^۴ و همکاران [۴۵] در سال ۲۰۲۰ با استفاده از نسخه سوم شبکه یادگیری عمیق YOLO نیز به تشخیص پهپاد با استفاده از تصاویر حاصل از دوربین های مرئی به همراه عمق پرداختند. در این دسته مطالعات مسئله سرعت و دقت تشخیص برطرف شده است، اما همچنان چالش های مرتبط با تصاویر مرئی همانند پیش زمینه شلوغ، مشکلات نور داخل تصویر و نواحی پنهان به عنوان یک محدودیت باقی است.

۲-۴- تشخیص پهپاد با داده های عمق و شبکه یادگیری عمیق

کریو^۵ و همکاران [۱۴] در سال ۲۰۱۸ برای اولین بار به تشخیص پهپاد با استفاده از نسخه دوم شبکه یادگیری عمیق YOLO در تصاویر عمق به دست آمده از دوربین مرئی به همراه عمق پرداختند. از طرفی، همین گروه در سال ۲۰۲۰ در مقاله ای [۴۶] با استفاده از تصاویر عمق واقعی به دست آمده از دوربین سه بعدی ZED (که به طور مستقیم اطلاعات عمق صحنه را ضبط می کنند) و با استفاده از مدل یادگیری عمیق اصلاح شده YOLO به صورت On-board به تشخیص و موقعیت یابی پهپاد پرداختند. استفاده از شبکه یادگیری عمیق در این مطالعات باعث بالا رفتن سرعت و استفاده از تصاویر عمق موجب از بین رفتن چالش های مطرح در تصاویر مرئی و حرارتی شده است.

در پژوهش پیش رو سعی در ارائه رویکرد تشخیصی بر اساس تصویر عمق با استفاده از شبکه یادگیری عمیق YOLO است که ضمن بهره مندی از نکات ارائه شده در روش های پیشین، نسبت به ارائه روش جدیدی در زمینه

^۱ Zeiler and Fergus

^۲ Visual Geometry Group (VGG)

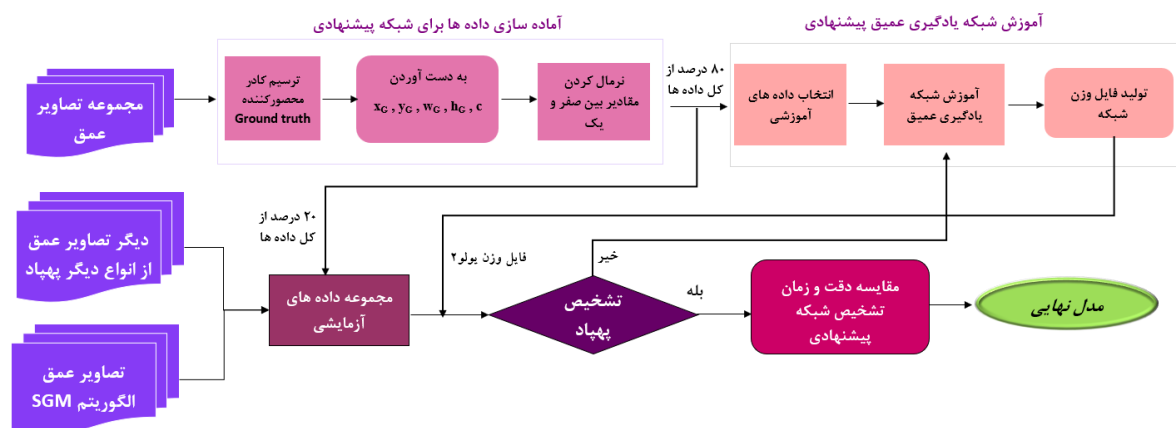
^۳ Aker

^۴ Nencho

^۵ Carrio

^۶ Microsoft Airsim

^۷ Semi Global Matching (SGM)



شکل ۳- عملکرد شبکه‌ی یادگیری عمیق پیشنهادی در تشخیص پهپاد

۳-۱- آماده سازی تصاویر عمیق

تصاویر عمیق شبیه سازی برای ورود به شبکه یادگیری عمیق YOLO، نیاز به آماده سازی دارند. به همین دلیل، با استفاده از ابزار ترسیم LabelImg [۴۷]، به منظور ترسیم کادر محصور کننده در اطراف پهپاد و ارائه اطلاعات مرکز کادر (x, y) ، طول و عرض کادر محصور کننده (w, h) و احتمال کلاس پهپاد استفاده می شود. اطلاعات حاصل از این ابزار همان پارامترهای اولیه ورودی شبکه (داده های حقیقی)^۱ هستند که به منظور آموزش وارد شبکه می شوند. برای تبدیل به فرمت ورودی شبکه، ابتدا لازم است مقادیر ورودی توسط ابزار LabelImg، بین صفر تا یک، در دو جهت x و y نرمال شوند. پس از نرمال سازی این مقادیر، اطلاعات هر کادر حاوی پهپاد به صورت $xcenter$ ، $ycenter$ ، wb و hb به عنوان ورودی شبکه یادگیری عمیق مورد نظر به همراه تصاویر عمیق شبیه سازی وارد شبکه می شوند. همچنین، به منظور ارزیابی دقت در مرحله آزمایش، از تصاویر عمیق واقعی حاصل از الگوریتم SGM استفاده شده است. الگوریتم تناظریابی SGM یک الگوریتم بینایی کامپیوتر به منظور محاسبه نقشه اختلاف متراکم^۲ از یک جفت تصویر^۳ است. با توجه به زمان اجرای قابل پیش بینی، تعادل بین کیفیت نتایج و زمان محاسبات استفاده از این روش در کاربردهای مبتنی بر دید سه بعدی مورد توجه قرار گرفته است [۴۸]. ایده الگوریتم SGM بهینه سازی در طول چند جهت و محاسبه مجموع توابع هزینه برای رسیدن به پیکسل p با میزان پراکندگی یا اختلاف d می باشد [۴۹].

۳-۲- آموزش شبکه

مهم ترین بخش از ساختار الگوریتم پیشنهادی، آموزش شبکه یادگیری عمیق است. آموزش شبکه یادگیری عمیق، شامل معماری شبکه، خروجی شبکه و تابع زیان به منظور پیش بینی کادرهای محصور کننده پیشنهادی است.

۳-۲-۱- عملکرد و معماری شبکه‌ی یادگیری عمیق

شبکه یادگیری عمیق YOLO یک شبکه عصبی منفرد است که کادر محصور کننده و احتمالات کلاس را مستقیماً از تصاویر کامل با یک مرحله ارزیابی پیش بینی می کند. از آنجا که تمامی مراحل تشخیص و شناسایی با استفاده از یک شبکه تکی انجام می شود، می توان آن را مستقیماً بر روی عملکرد تشخیص بهینه کرد. شبکه YOLO از ویژگی های کل تصویر برای پیش بینی هر محدوده استفاده می کند. برای شروع آموزش در نسخه دوم شبکه یادگیری عمیق YOLO، شبکه استخراج کننده ویژگی Darknet-19 استفاده می شود [۵۰]. این شبکه از ۱۹ لایه پیچشی و سپس لایه های Global AvgPool و لایه Softmax تشکیل شده است. این شبکه به گونه ای طراحی شده است که علاوه بر تشخیص با دقت بالا، امکان آموزش با سرعت بالا را نیز فراهم می کند. در اصل لایه های پیچشی با استفاده از فیلترهای مختلف، انواع ویژگی های داخل تصویر را استخراج می کنند. مطابق شکل (۴)، هر لایه پیچشی در شبکه یادگیری عمیق پیشنهادی شامل عملیات نرمال سازی دسته ای^۴، تابع فعال سازی Leaky Relu^۵ و لایه ادغام بیشینه^۶ است.

^۴ Batch Normalization (BN)

^۵ Leaky Rectified Linear Unit (Leaky Relu)

^۶ Max Pooling

^۱ Ground Truth

^۲ Dense Disparity Map

^۳ Stereo image

شکل (۵) معماری شبکه یادگیری عمیق YOLOv2 را نشان می‌دهد. بر اساس این شکل، تصویر عمق ورودی پس از ورود به شبکه، تحت یک سری لایه‌های پیچشی که جزئیات آن در شکل (۴) نشان داده شده‌است، قرار می‌گیرد. سپس خروجی شبکه پیچشی به منظور دریافت جزئیات بیشتر از تصاویر، از طریق لایه سازمان‌دهی مجدد^۳ با لایه‌های پیچشی دیگر ترکیب^۴ می‌شود. لایه سازمان‌دهی مجدد بعد از پنج لایه پیچشی در شبکه Darknet-19، یک لایه برای سازمان‌دهی و تنظیم شبکه است. این لایه هر پیکسل متناوب را به صورت یکی در میان گرفته و آن را در یک بعد متفاوت قرار می‌دهد. سپس یک تانسور سه‌بعدی تشکیل می‌شود که این خروجی دارای اطلاعات مختصات کادر محصورکننده و احتمالات کلاس است. در مرحله آموزش شبکه، این خروجی به همراه داده‌های حقیقی وارد تابع زیان^۵ می‌شوند.

به ازای هر سلول تشکیل‌شده در شبکه، به تعداد B کادر محصورکننده و ضریب اطمینان^۶ پیش‌بینی می‌شود. پارامتر B، تعداد کادر محصورکننده است که توسط شبکه YOLO برای یک شیء ترسیم می‌شود و توسط کاربر نیز قابل تنظیم است. این عدد در شبکه یادگیری عمیق پیشنهادی برابر با عدد پنج تنظیم می‌شود. ضریب اطمینان نیز بیانگر میزان اطمینان مدل از وجود پهنپاد در آن محدوده و همچنین میزان دقت پیش‌بینی کادر محصورکننده است. به‌طور کلی، ضریب اطمینان p_o به‌صورت رابطه (۲) تعریف می‌شود [۳۰]:

$$p_o = p_r(object) \times IOU_{pred}^{truth} \quad (2)$$

در این رابطه، $p_r(object)$ میزان احتمال کلاس شیء و IOU نشان‌دهنده اشتراک محدوده است. مفهوم IOU در رابطه (۳) بیان شده‌است. در این رابطه، Area of overlap، میزان اشتراک دو محدوده پس از تقاطع و Area of Union میزان اجتماع دو محدوده است.

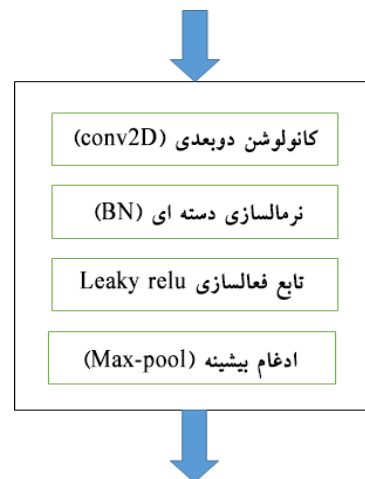
$$IOU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (3)$$

^۳ Reorganization layer (reorg)

^۴ Concatenation

^۵ Loss Function

^۶ Confidence Score



شکل ۴- جزئیات لایه‌های پیچشی در شبکه‌ی یادگیری عمیق

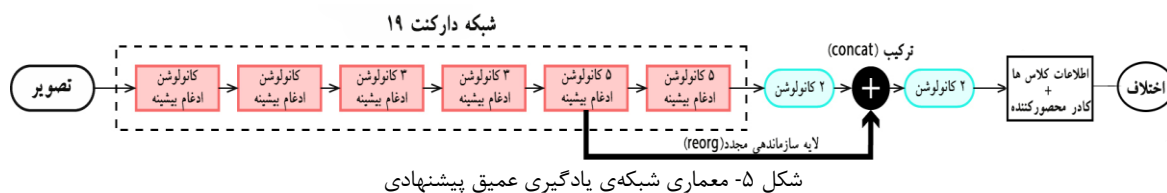
لایه ادغام بیشینه گاهی اوقات بلافاصله پس از لایه پیچشی، در اواسط شبکه پس از سه یا پنج لایه پیچشی، به منظور کاهش ابعاد و تعداد پارامترهای خروجی و استخراج بیشینه ویژگی‌ها تکرار می‌شود. عملیات نرمال‌سازی دسته‌ای، نرمال‌سازی خروجی لایه‌های پنهان را انجام می‌دهد و موجب سریع‌تر شدن عملیات یادگیری می‌شود. در شبکه یادگیری عمیق پیشنهادی، لایه‌های پیچشی با ابعاد 3×3 هستند. مهم‌ترین ورودی لایه‌های پیچشی تصویر است که به صورت یک تانسور دوبعدی وارد آن‌ها می‌شود. پس از به دست آوردن نقشه ویژگی حاصل از لایه پیچشی، عملیات نرمال‌سازی دسته‌ای انجام‌شده و وابستگی مدل حاصل به مقادیر اولیه کمتر می‌شود. سپس این خروجی وارد تابع فعالیت شبکه شده تا میزان خروجی شبکه عصبی تعیین شود.

در لایه‌های پیچشی شبکه یادگیری عمیق YOLO از تابع فعالیت Leaky Relu استفاده شده‌است. تابع فعالیت Leaky Relu، یکی از انواع توابع فعالیت Relu^۱ است که دارای شیب مثبت کمی در ناحیه منفی است؛ بنابراین، عملیات پس‌انتشار^۲ را حتی برای مقادیر ورودی منفی فعال می‌کند [۵۱]. مزیت تابع فعالیت Relu این است که مشتق این تابع برای مقادیر بزرگ‌تر از x، ثابت است و این مورد موجب سرعت بخشی به فرآیند یادگیری می‌شود. رابطه (۱) بیانگر عملکرد تابع فعالیت Leaky Relu برای مقادیر بزرگ‌تر از صفر و کوچک‌تر از صفر است.

$$f(x) = \begin{cases} 0.1x & x < 0 \\ x & x \geq 0 \end{cases} \quad (1)$$

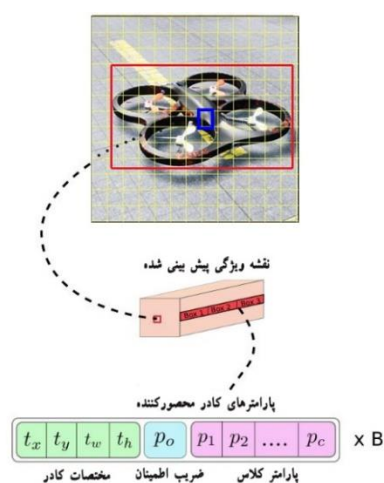
^۱ Rectified Linear Unit (Relu)

^۲ Back Propagation



شکل ۵- معماری شبکه‌ی یادگیری عمیق پیشنهادی

آموزش، B کادر محصورکننده را در هر سلول پیش‌بینی می‌کند که هر کادر دارای تعداد پارامتر $(5+C)$ است، به طوری که عدد پنج بیانگر تعداد پیش‌بینی‌ها شامل مختصات مرکز (t_x, t_y) ، طول (t_w) ، عرض (t_h) و عدد ضریب اطمینان (p_o) است. همچنین، پارامتر C بیانگر پارامتر کلاس‌ها بوده و به تعداد کلاس‌های موجود است. بنابراین، خروجی حاصل از این شبکه در مرحله آخر، احتمالات کلاس به همراه کادر محصورکننده است که به صورت یک تانسور سه‌بعدی با ابعاد $B \times (5+C) \times (S \times S)$ در شکل (۷) نشان داده شده‌است. در این شکل تقسیم‌بندی تصویر به اندازه $(S \times S)$ انجام شده‌است. سپس، نقشه ویژگی پیش‌بینی‌شده با تعداد B کادر محصورکننده تشکیل شده که هر یک از این کادرها دارای پارامترهای پیش‌بینی‌کننده وضعیت کادر می‌باشند.



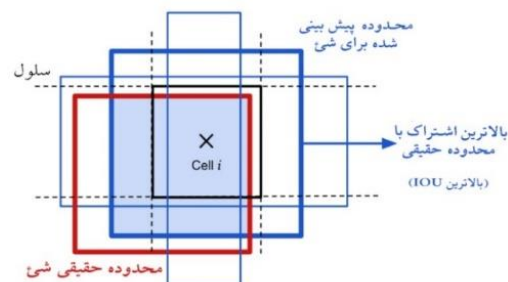
شکل ۷- خروجی شبکه‌ی یادگیری عمیق پیشنهادی

در نسخه دوم شبکه یادگیری عمیق YOLO، ابعاد خروجی شبکه $D \times 13 \times 13$ است. پارامتر D بیانگر عمق هر سلول بوده که وابسته به تعداد کلاس‌های قابل پیش‌بینی توسط شبکه است.

۳-۲-۳- تابع زبان

در مرحله آموزش، خروجی اطلاعات کادرهای پیش‌بینی‌شده به همراه کادرهای حقیقی به تابع زبان داده

بر اساس شکل (۶)، مرکز پهپاد در هر سلول شبکه^۱ قرار داشته باشد، دیگر کادرهای محصورکننده در دیگر بخش‌های تصویر حذف می‌شوند و تنها همان سلول شبکه وظیفه تشخیص شی را دارد. به بیان دیگر، کادرهای محصورکننده داخل این سلول بالاترین احتمال را برای تشخیص شی دارند. در این شکل، کادرهای محصورکننده پیش‌بینی‌شده با رنگ آبی و محدوده حقیقی شی با رنگ قرمز نمایش داده شده‌است. اگر سلول حاوی پهپاد نباشد، ضریب اطمینان صفر است. در غیر این صورت، باید میزان ضریب اطمینان با میزان اشتراک محدوده^۲ برابر باشد. در نهایت، می‌توان گفت محدوده‌ای حاوی پهپاد است که میزان IOU بالاتر یا به بیان دیگر، بالاترین اشتراک با محدوده حقیقی را داشته باشد که این محدوده در شکل (۶) با کادر آبی پررنگ نشان داده شده‌است.



شکل ۶- اشتراک محدوده حاوی شی با محدوده حقیقی شی

هر کادر محصورکننده شامل ۵ پیش‌بینی است: t_x ، t_y ، t_w ، t_h و p_o مختصات (t_x, t_y) نشان‌دهنده مرکز کادر محصورکننده نسبت به مرزهای سلول شبکه و عرض و ارتفاع (t_h, t_w) نسبت به کل تصویر پیش‌بینی می‌شود. در ادامه، به جزئیات مربوط به خروجی شبکه یادگیری عمیق YOLO پرداخته شده‌است.

۳-۲-۳- خروجی شبکه

در شبکه یادگیری عمیق YOLO نقشه ویژگی پیش‌بینی‌شده یک ماتریس سه‌بعدی است. این شبکه در روند

^۱ Grid cell

^۲ Intersection Over Union (IOU)

مقدار تابع زیان احتمالات کلاس ($loss_{ij}^p$) نیز از رابطه (۸) به دست می‌آید.

$$loss_{ij}^p = -\frac{\lambda_{class}}{N_{Lobj}} \sum_{i=0}^{s^2} \sum_{j=0}^B L_{ij}^{obj} \sum_{c \in class} p_{ij}^c \log(\hat{p}_{ij}^c) \quad (8)$$

در این رابطه، p_{ij}^c نشان‌دهنده احتمال کلاس صحیح و $\hat{p}_{ij}^c$ نشان‌دهنده احتمال کلاس پیش‌بینی شده است. مطابق رابطه (۹)، قسمت $loss_{ij}^c$ ، مربوط به ضریب اطمینان (C) است که بیانگر حضور یا عدم حضور پهپاد داخل کادر محصورکننده پیش‌بینی شده است [۵۲].

$$loss_{ij}^c = -\frac{\lambda_{obj}}{N_{conf}} \sum_{i=0}^{s^2} \sum_{j=0}^B L_{ij}^{obj} (IOU_{prediction_{ij}}^{ground truth_{ij}} - \hat{C}_{ij})^2 + \frac{\lambda_{noobj}}{N_{conf}} \sum_{i=0}^{s^2} \sum_{j=0}^B L_{ij}^{noobj} (0 - \hat{C}_{ij})^2 \quad (9)$$

در این رابطه $\hat{C}_{ij}$ بیانگر مقادیر پیش‌بینی شده برای ضریب اطمینان است. مقادیر N_{Lobj} از رابطه (۱۰) و N_{conf} از رابطه (۱۱) به دست می‌آید.

$$N_{Lobj} = \sum_{i=0}^{s^2} \sum_{j=0}^B L_{ij}^{obj} \quad (10)$$

$$N_{conf} = \sum_{i=0}^{s^2} \sum_{j=0}^B L_{ij}^{obj} + L_{ij}^{noobj} (1 - L_{ij}^{obj}) \quad (11)$$

میزان مقادیر λ_{noobj} ، λ_{coord} و λ_{class} مقادیر عددی هستند که برای وزن‌دهی به هر قسمت از تابع زیان استفاده می‌شوند. مقادیر L_{ij}^{obj} و L_{ij}^{noobj} مقادیر صفر و یک هستند و تحت شرایطی مشخص می‌باشند که در رابطه (۱۲) و (۱۳) بیان شده است [۵۲]:

$$L_{ij}^{obj} = \begin{cases} 1 & \text{if } C_{ij} = 1 \\ 0 & \text{else} \end{cases} \quad (12)$$

$$L_{ij}^{noobj} = \begin{cases} 1 & \text{if } \max_{xy} IOU_{prediction_{xy}}^{ground truth_{xy}} < 0.6 \\ 0 & \text{else} \end{cases} \quad (13)$$

در طول آموزش، تابع خطا در رابطه (۶) باید به بهترین شکل بهینه‌سازی شود. ممکن است در مراحل اولیه و کرنل‌های ابتدایی شبکه، پهپاد تشخیص داده نشود و تابع اختلاف، مقدار زیادی را به خود اختصاص دهد. اما در مراحل

می‌شود تا مقادیر احتمالات کلاس و مختصات کادر محصورکننده پیش‌بینی شده ارائه شود. یعنی می‌توان گفت یک پیش‌بینی‌کننده به عنوان نماینده تعیین می‌شود که این محدوده دارای بالاترین میزان IOU با داده‌های حقیقی است؛ بنابراین، این محدوده به عنوان محدوده نماینده وجود پهپاد انتخاب می‌شود. در شبکه یادگیری عمیق پیشنهادی، از تابع زیان میانگین مربعات^۱ استفاده می‌شود، چرا که عملیات بهینه‌سازی در این تابع اتلاف تسهیل شده و با مسائل رگرسیون سازگار است [۳۲]. میزان مقادیر پیش‌بینی و پاسخ حقیقی به صورت روابط (۴) و (۵) است.

$$prediction_{ij} = (\hat{x}_{ij}, \hat{y}_{ij}, \hat{w}_{ij}, \hat{h}_{ij}) \quad (4)$$

$$ground truth_{ij} = (x_{ij}, y_{ij}, w_{ij}, h_{ij}) \quad (5)$$

تابع اتلاف نهایی در این شبکه مجموع تابع زیان مختصات ($loss_{ij}^{xywh}$)، تابع زیان احتمال کلاس‌ها ($loss_{ij}^p$) و تابع زیان عدد ضریب اطمینان ($loss_{ij}^c$) است که به صورت رابطه (۶) نمایش داده شده است.

$$loss_{ij} = loss_{ij}^{xywh} + loss_{ij}^p + loss_{ij}^c \quad (6)$$

که در آن مقدار $loss_{ij}^{xywh}$ از رابطه (۷) پیروی می‌کند.

$$loss_{ij}^{xywh} = \frac{\lambda_{coord}}{N_{Lobj}} \sum_{i=0}^{s^2} \sum_{j=0}^B L_{ij}^{obj} \left[(x_{ij} - \hat{x}_{ij})^2 + (y_{ij} - \hat{y}_{ij})^2 + \left(\sqrt{w_{ij}} - \sqrt{\hat{w}_{ij}} \right)^2 + \left(\sqrt{h_{ij}} - \sqrt{\hat{h}_{ij}} \right)^2 \right] \quad (7)$$

رابطه (۷)، بیانگر مقایسه دو کادر محصورکننده پیش‌بینی شده و حقیقی است. به این صورت که اگر پهپاد وجود داشته باشد، مختصات مراکز این دو کادر با یکدیگر مقایسه می‌شود. در این رابطه، (x_{ij}, y_{ij}) نشان‌دهنده مختصات صحیح، $(\hat{x}_{ij}, \hat{y}_{ij})$ نشان‌دهنده مختصات پیش‌بینی شده، (w_{ij}, h_{ij}) نشان‌دهنده طول و عرض صحیح و $(\hat{w}_{ij}, \hat{h}_{ij})$ نشان‌دهنده طول و عرض پیش‌بینی شده است. همچنین، i نشان‌دهنده تعداد سلول‌های شبکه و j نشان‌دهنده تعداد کادر محصورکننده است.

^۱ Mean Squared Error (MSE)

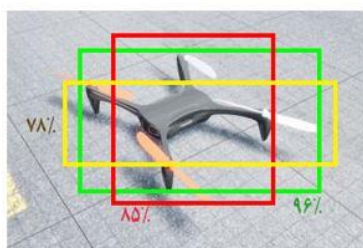
بعدی این روند ادامه یافته و وزن‌ها به روزرسانی می‌گردد. بنابراین، هرچه شبکه پیش می‌رود، مقادیر داخل کرنل به پهپاد مورد نظر شبیه‌تر می‌شود. در نتیجه، میزان تابع اختلاف کاهش یافته و میزان تفاوت بین پارامترهای خروجی و پارامترهای ورودی شبکه به حداقل میزان خود می‌رسد. سپس، شبکه وزن‌های آموزش‌دیده با بالاترین دقت در تشخیص شی را تولید می‌کند. در نهایت، بهترین کادرهای محصورکننده برای پهپاد پیش‌بینی و وزن‌های متناسب با آن تولید شده و وارد مرحله آزمایش شبکه می‌شوند.

۳-۳- آزمایش شبکه

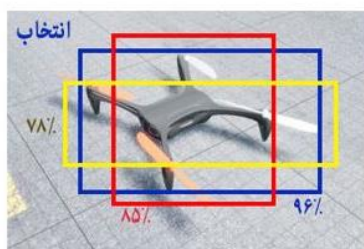
در این مرحله، وزن‌های حاصل از مرحله آموزش با انواع تصاویر آزمایشی، بررسی شدند و سپس، تشخیص پهپاد با استفاده از مدل آموزش‌دیده انجام شد. الگوریتم‌های تشخیص شی برای انتخاب بهترین کادر حاوی شی، از بین چند کادر محصورکننده پیش‌بینی‌شده، از الگوریتم حذف بیشینه‌های محلی^۱ استفاده می‌کنند. این روش برای حذف کادرهای محصورکننده محتمل و انتخاب بهترین کادر حاوی پهپاد استفاده می‌شود (شکل (۹)).

مطابق شکل (۸)، ورودی این الگوریتم، مجموعه‌ای از کادرهای محصورکننده پیش‌بینی‌شده (B)، ضریب اطمینان مرتبط با هر کادر (S) و میزان آستانه همپوشانی (N) و خروجی الگوریتم لیستی از کادرهای محصورکننده فیلتر شده (D) است.

ابتدا کادر محصورکننده با بالاترین میزان ضریب اطمینان (S) انتخاب می‌شود. سپس، کادر انتخابی از لیست کادرهای محصورکننده پیش‌بینی‌شده (B) حذف و به لیست پیشنهادی کادرهای نهایی (D) اضافه می‌شود. در مرحله بعد، کادرهای محصورکننده پیشنهادی با دیگر کادرهای محصورکننده با استفاده از میزان اشتراک محدوده (IOU) مقایسه می‌شوند. اگر این میزان از اندازه آستانه همپوشانی (N) بالاتر باشد، کادر مربوطه از لیست کادرهای (B) حذف می‌شود. در تکرار بعدی، کادر محصورکننده با بالاترین ضریب اطمینان از مجموعه کادرهای پیش‌بینی‌شده (B) انتخاب شده و به لیست پیشنهادی کادرهای نهایی (D) اضافه می‌شود. مجدداً میزان اشتراک محدوده (IOU) این کادر با تمام کادرهای باقیمانده در لیست کادرهای پیش‌بینی‌شده (B) محاسبه شده و کادرهایی که اشتراک محدوده (IOU) بیش از حد آستانه (N) دارند حذف می‌شوند. این فرآیند تا جایی تکرار می‌شود که دیگر کادر محصورکننده در مجموعه B باقی نماند و در نهایت کادر نهایی حاوی پهپاد انتخاب و پارامترهای خروجی آن ارائه می‌شوند [۵۴].



کادرهای پیش‌بینی شده



انتخاب کادر پیشنهادی با بالاترین ضریب اطمینان



حذف کادرهای پیشنهادی با کمترین همپوشانی

شکل ۹- الگوریتم حذف بیشینه‌های محلی

```

Input :  $B = \{b_1, \dots, b_N\}$ ,  $S = \{s_1, \dots, s_N\}$ ,  $N_t$ 
 $B$  is the list of initial detection boxes
 $S$  contains corresponding detection scores
 $N_t$  is the NMS threshold

begin
 $D \leftarrow \{\}$ 
while  $B \neq \text{empty}$  do
 $m \leftarrow \text{argmax } S$ 
 $\mathcal{M} \leftarrow b_m$ 
 $D \leftarrow D \cup \mathcal{M}$ ;  $B \leftarrow B - \mathcal{M}$ 
for  $b_i$  in  $B$  do
    if  $\text{iou}(\mathcal{M}, b_i) \geq N_t$  then
         $B \leftarrow B - b_i$ ;  $S \leftarrow S - s_i$ 
    end
end
 $s_i \leftarrow s_i f(\text{iou}(\mathcal{M}, b_i))$ 
end
end
return  $D, S$ 
end

```

شکل ۸- الگوریتم حذف بیشینه‌های محلی [۵۳]

^۱ Non maxima Suppression

۳-۳-۱- بررسی معیار ارزیابی در تشخیص پهپاد

معیارهای مختلفی برای ارزیابی انواع مدل‌های تشخیص شی وجود دارد [۵۵]. در این مقاله، به منظور سنجش و ارزیابی شبکه یادگیری عمیق پیشنهادی از دو معیار ارزیابی دقت^۱ و میانگین دقت متوسط^۲، استفاده شده است. معیار دقت به این معنی است که از میان ورودی‌هایی که کلاس آن‌ها مثبت پیش‌بینی می‌شود، چند درصد از آن‌ها در واقع اعضای کلاس مثبت هستند. به عنوان مثال در این مطالعه، دقت به این معنی است که از تمام ورودی‌های پیش‌بینی‌شده به عنوان پهپاد، چند درصد واقعاً پهپاد هستند. با تعیین مقادیر مثبت صحیح^۳ و مثبت غلط^۴ با استفاده از شکل (۱۰)، این معیار از طریق رابطه (۱۴) به دست می‌آید و بین اعداد صفر و یک است.

موقعیت واقعی مثبت (Positive)	موقعیت واقعی منفی (Negative)
پیش بینی مثبت (Positive)	مثبت غلط (FP)
پیش بینی منفی (Negative)	منفی غلط (FN)

شکل ۱۰- بررسی دقت تشخیص شی

$$Precision = \frac{TP}{TP + FP} \quad (14)$$

همچنین، معیار میانگین دقت متوسط با محاسبه میانگین دقت تشخیص کلاس پهپاد تعیین می‌شود، به عبارت دیگر، این معیار ارزیابی، کادر محصورکننده حقیقی را با کادر محصورکننده پیش‌بینی کننده پهپاد مقایسه می‌کند. افزایش این عدد نشان‌دهنده عملکرد دقیق‌تر شبکه پیشنهادی در تشخیص است. معیار میانگین دقت متوسط از رابطه (۱۵) حاصل می‌شود [۵۵]:

$$mAP = \frac{1}{|Classes|} \sum_{c \in Classes} \frac{TP(c)}{TP(c) + FP(c)} \quad (15)$$

در نهایت، نتیجه حاصل از تشخیص پهپاد در روند آزمایش، به همراه دقت تشخیص آن نمایش داده می‌شود.

۴- نتایج عملی

به منظور ارزیابی قابلیت مدل پیشنهادی در تشخیص و شناسایی پهپادها در محیط‌های مختلف، مراحل پیاده‌سازی، مجموعه داده استفاده شده و نتایج حاصل ارائه شده است.

۴-۱- مجموعه داده‌ی استفاده شده

برای شروع مرحله آموزش شبکه، لازم است مجموعه داده‌ای از تصاویر عمق حاوی انواع پهپاد در محیط‌های مختلف تهیه شود. برای افزایش کارایی، قابلیت اطمینان و بررسی تعمیم‌پذیری شبکه، از دو نوع مجموعه تصاویر عمق استفاده شده است: (۱) تصاویر عمق شبیه‌سازی حاصل از نرم‌افزار مایکروسافت ایرسیم. (۲) تصاویر عمق حاصل از دوربین سه‌بعدی و الگوریتم SGM.

۴-۱-۱- تصاویر عمق شبیه‌سازی حاصل از محیط مایکروسافت ایرسیم

به منظور انجام آموزش شبکه یادگیری عمیق مورد نظر، از مجموعه تصاویر عمق شبیه‌سازی از محیط شبیه‌ساز مایکروسافت ایرسیم به منظور نمایش پهپاد در محیط‌های مختلف استفاده شده است [۱۴]. مایکروسافت ایرسیم یک محیط مجازی است که برای وسایل نقلیه هوایی بدون سرنشین، بر روی موتور غیرواقعی ساخته شده است. محیط مایکروسافت ایرسیم، امکان تولید داده‌های آموزشی زیادی را برای ساخت مدل‌های یادگیری ماشین فراهم می‌کند. طراحی این محیط، امکان توسعه الگوریتم‌هایی را در محیط شبیه‌ساز ایجاد کرده و آن‌ها را در محیط واقعی قابل اجرا می‌کند [۱۴]. اجزای اصلی ایرسیم، از جمله موتور فیزیکی، نوع محیط و نوع حسگرها به گونه‌ای طراحی شده‌اند که با حداقل وابستگی در خارج از محیط نرم‌افزار به طور مستقل قابل استفاده بوده و به راحتی قابل توسعه هستند [۵۶]. در شکل (۱۱) تصویری از سه نوع پهپاد که در این نرم‌افزار استفاده شده، نشان داده شده است.



شکل ۱۱- سه نوع پهپاد استفاده شده در تولید تصاویر عمق

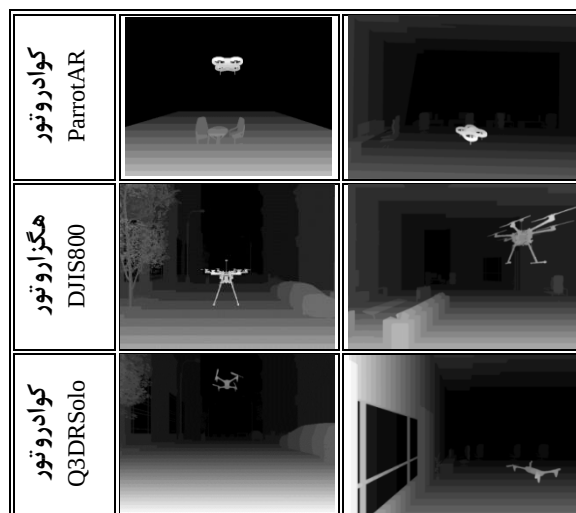
^۱ Precision

^۲ mean Average Precision

^۳ True Positive (TP)

^۴ False Negative (FN)

در تمام صحنه‌ها، یکی از پهپادها به عنوان هواپیمای «میزبان»^۱ در نظر گرفته می‌شود که نقشه‌های عمق از این پرند حاصل شده و پهپادهای دیگر که در نقشه‌های عمق قابل مشاهده هستند، به عنوان پرند «هدف»^۲ در نظر گرفته می‌شوند [۱۴]. نمونه تصاویر عمق سه پهپاد در محیط‌های مختلف مطابق شکل (۱۲) است:



شکل ۱۲- تصاویر عمق سه پرند با استفاده از نرم‌افزار ایرسیم [۱۴]

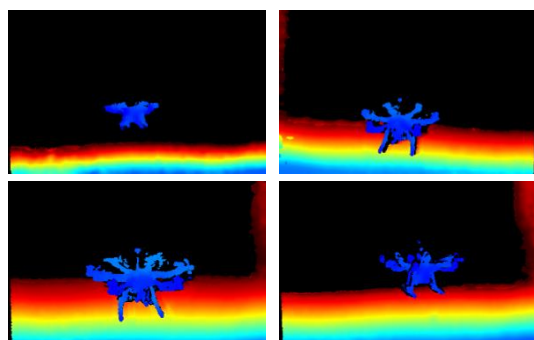
۴-۱-۲- تصاویر عمق واقعی از دوربین سه‌بعدی و الگوریتم SGM

در این پژوهش به منظور نمایش عملکرد شبکه در محیط واقعی، از جفت تصاویر حاصل از دوربین سه‌بعدی، به منظور تولید تصاویر عمق واقعی استفاده شده‌است. همانطور که در بخش آماده‌سازی تصاویر عمق گفته شد، جفت تصاویر وارد الگوریتم تناظریابی SGM شده و مطابق شکل (۱۳)، تصاویر عمق به دست می‌آید. در این مجموعه داده از یک نوع پهپاد هگزاروتور در حال حرکت در فواصل مختلف تا دوربین تصویربرداری شده‌است.

همان‌طور که گفته شد، تصاویر عمق استفاده‌شده در این پژوهش، دو نوع مجموعه‌داده از تصاویر عمق شبیه‌سازی از محیط نرم‌افزار شبیه‌ساز میکروسافت ایرسیم و همچنین تصاویر عمق واقعی به دست‌آمده از الگوریتم تناظریابی SGM است. مجموعه تصاویر عمق شبیه‌سازی استفاده‌شده شامل یک مجموعه ۶۰۰۰ تایی تصاویر عمق شامل سه نوع پهپاد در شکل (۱۲) است. به منظور آموزش شبکه، از مجموعه

۳۷۶۳ تایی از داده‌های عمق شبیه‌سازی مربوط به پهپاد کوادروتور ParrotAR در محیط میز و صندلی که نمونه آن در شکل (۱۲) نشان داده شده، استفاده شده‌است [۱۴]. این مجموعه‌داده به نسبت ۸۰ به ۲۰ تقسیم شدند و ۸۰ درصد از تصاویر (۳۰۱۰ تصویر) برای مرحله آموزش شبکه و ۲۰ درصد برای مرحله آزمایش انتخاب شدند. سپس، با استفاده از ابزار ترسیم LabelImg به منظور آماده‌سازی تصاویر عمق برای مرحله آموزش، در اطراف پهپاد کادر محصورکننده ترسیم شد و اطلاعات مربوط به کلاس شیء، مختصات مرکز و طول و عرض کادرها به عنوان پاسخ حقیقی به همراه تصاویر مربوطه، وارد شبکه شدند. همچنین، تصاویر عمق واقعی حاصل از الگوریتم SGM یک مجموعه ۵۰ تایی بوده که به منظور آزمایش شبکه در نظر گرفته شده‌است.

به طور کلی، می‌توان گفت در مرحله آزمایش مدل، از دیگر مجموعه‌داده عمق شبیه‌سازی از دو نوع پهپاد DJI800 و Q3DRSolo، انواع داده‌های عمق واقعی به دست‌آمده از الگوریتم SGM و ۲۰ درصد باقیمانده از تصاویر مربوط به پهپاد ParrotAR استفاده شده‌است.



شکل ۱۳- تصاویر عمق حاصل از الگوریتم SGM

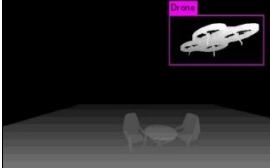
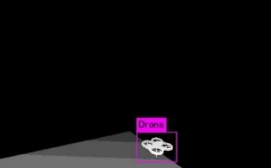
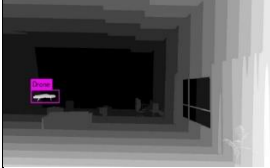
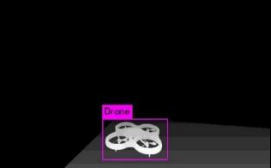
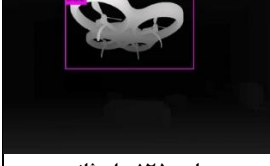
۴-۲- پیاده‌سازی و نتایج حاصل از آموزش شبکه

در شبکه پیشنهادی از مدل از پیش آموزش‌یافته Darknet-19 به منظور آموزش و طبقه‌بندی استفاده شده است [۵۰]. سخت‌افزار مورد استفاده در این پژوهش GPU با ظرفیت ۲ گیگابایت از نوع NVIDIA GEFORCE740M با قابلیت نصب کودا^۳ است و در تمامی مراحل به منظور آموزش و آزمایش شبکه، از این سخت‌افزار استفاده شده‌است. برای آموزش شبکه مورد نظر، CUDNN^۴ نسخه ۷.۴، ابزار کودا

^۳ Cuda
^۴ CUDA Deep Neural Network (cuDNN)

^۱ Host
^۲ Target

در ادامه، نتایج ارزیابی دقت و زمان در شبکه YOLOv2 با سه نوع پهپاد در تصاویر عمق شبیه سازی نمایش داده شده است.

	
زمان: ۱۲۶ میلی ثانیه دقت: ۸۵ درصد	زمان: ۱۲۳ میلی ثانیه دقت: ۸۲ درصد
	
زمان: ۱۲۵ میلی ثانیه دقت: ۸۰ درصد	زمان: ۱۲۵ میلی ثانیه دقت: ۸۲ درصد
	
زمان: ۱۲۵ میلی ثانیه دقت: ۹۲ درصد	زمان: ۱۲۴ میلی ثانیه دقت: ۹۳ درصد

شکل ۱۵- نتایج تشخیص پهپاد کوادروتور ParrotAR

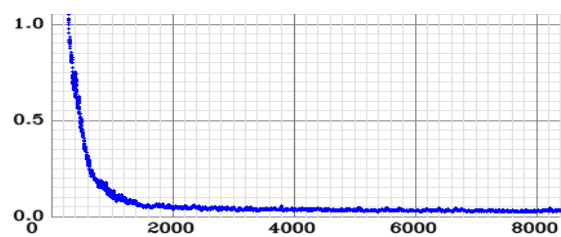
در شکل (۱۵) نتایج حاصل از تشخیص پهپاد کوادروتور ParrotAR در انواع تصاویر عمق نمایش داده شده است. بر اساس این نتایج شبکه مورد نظر به میانگین دقت ۸۶ درصد و میانگین زمان ۱۲۴ میلی ثانیه در تشخیص پهپاد رسیده است. همان طور که مشاهده می شود، این نوع پهپاد در محیط داخلی با جهت گیری های مختلف در کنار دیگر اشیاء نمایش داده شده و شبکه در تمامی این تصاویر موفق به تشخیص پهپاد شده است. در این مجموعه تصاویر، پهپاد در محیط های مختلف در پیش زمینه ها و جهت گیری های مختلف در حال پرواز است. علاوه بر این، زمان تشخیص کوتاه پهپاد نشان دهنده سرعت بالای شبکه در تشخیص است. یکی از دلایل دقت و سرعت بالا در این تصاویر، این است که شبکه با انواع تصاویر عمق از این نوع پهپاد آموزش دیده است؛ بنابراین، شبکه در مرحله آموزش با ویژگی های پهپاد کوادروتور ParrotAR آشنایی داشته و می تواند با دقت بالاتری پهپاد را تشخیص دهد.

در شکل (۱۶)، نتایج حاصل از تشخیص پهپاد کوادروتور DJI800 نمایش داده شده است که بر اساس این نتایج، شبکه

نسخه ۱۰ و کتابخانه OpenCV نسخه ۴,۰,۱ پیاده سازی شده اند. همچنین، کد اصلی چارچوب Darknet تهیه شده و فایل های پیکربندی اصلاح شدند و تعداد کلاس ها به عدد یک تغییر کرد. در فایل پیکربندی شبکه مورد نظر، سه لایه پیشی قبل از هر سه لایه YOLO وجود دارد. در این لایه ها، از فیلترهایی برای استخراج ویژگی از تصاویر ورودی پهپاد استفاده می شود. با توجه به رابطه (۱۶) تعداد فیلترها در این سه لایه پیشی، به عدد ۱۸ تغییر یافت.

$$(16) \quad 3 \times (5 + \text{تعداد کلاس ها}) = \text{تعداد فیلترها}$$

برای شروع مرحله آموزش، تعداد دسته ها و نرخ یادگیری به ترتیب روی اعداد ۱ و ۰,۰۰۰۵ تنظیم شده است. سپس، مدل با ۳۰۰۰۰ تکرار آموزش داده شد و فایل وزن پس از هر ۱۰۰۰۰ تکرار ذخیره گردید. روند نمای کلی آموزش شبکه و کاهش خطا تا ۰,۰۲ بعد از ۵ روز و ۳۰۰۰۰ تکرار در شکل (۱۴) ارائه شده است. سپس، برای آزمایش شبکه از فایل وزن نهایی استفاده و عملکرد آن با استفاده از معیار ارزیابی دقت نشان داده شده است.



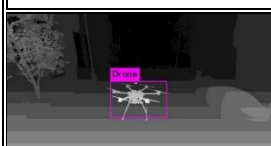
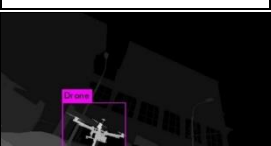
شکل ۱۴- نمودار نهایی میانگین خطای شبکه ی یادگیری عمیق

۴-۳- ارزیابی نتایج

پس از آموزش شبکه YOLOv2 و ایجاد فایل های وزن، مرحله ارزیابی و آزمایش شبکه آغاز شد. در این مرحله، از دو دسته تصویر عمق شبیه سازی و واقعی استفاده شده است. تصاویر شبیه سازی برای مرحله آزمایش مدل، شامل ۲۰ درصد تصاویر عمق باقیمانده از مرحله آموزش، تصاویر عمق مربوط به پهپاد DJI800 و Q3DRSolo و انواع داده های عمق واقعی به دست آمده از الگوریتم SGM است.

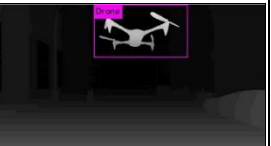
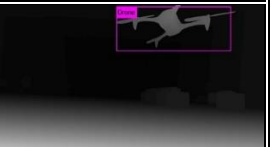
۴-۳-۱- ارزیابی نتایج در مجموعه تصاویر شبیه سازی

مورد نظر به میانگین دقت ۸۱ درصد و میانگین زمان ۱۲۵ میلی‌ثانیه رسیده‌است. تفاوت این تصاویر در وضعیت پیش‌زمینه است که نشان داده شده در تصاویری که اشیاء دیگری غیر از پهپاد در محیط وجود دارد، شبکه با دقت بالا عملیات تشخیص را انجام داده و چالش‌هایی که در خصوص استفاده از تصاویر مرئی مطرح شد، در این نتایج به چشم نمی‌خورد. میانگین دقت در تشخیص این نوع پرند ۸۱ درصد است که از نتیجه حاصل از دقت تشخیص پهپاد ParrotAR کمتر است، چرا که شبکه در طول روند آموزش، پهپاد DJI800 را مشاهده نکرده‌است و با ویژگی‌های فیزیکی این پهپاد آشنایی ندارد. بنابراین تشخیص این پهپاد با دقت بالا، نشان‌دهنده قابلیت تعمیم‌پذیری مدل می‌باشد.

	
زمان: ۱۲۶ میلی‌ثانیه دقت: ۸۲ درصد	زمان: ۱۲۶ میلی‌ثانیه دقت: ۷۸ درصد
	
زمان: ۱۲۵ میلی‌ثانیه دقت: ۸۱ درصد	زمان: ۱۲۵ میلی‌ثانیه دقت: ۸۰ درصد
	
زمان: ۱۲۵ میلی‌ثانیه دقت: ۸۰ درصد	زمان: ۱۲۴ میلی‌ثانیه دقت: ۸۷ درصد

شکل ۱۶- نتایج تشخیص پهپاد کوادروتور DJI S800

بنابراین، می‌توان گفت با انتخاب تصاویر عمق چالش‌های مرتبط با تصاویر مرئی در جداسازی پیش‌زمینه و مشکلات نور داخل تصویر برطرف شده‌است. به طور مشابه، دلیل پایین بودن دقت در این دسته از تصاویر عمق نسبت به تصاویر در روند آموزش شبکه وجود نداشته و تشخیص موفق این پهپاد در مرحله آزمایش نشان‌دهنده قدرت تعمیم‌پذیری بالای شبکه است.

	
زمان: ۱۲۴ میلی‌ثانیه دقت: ۸۷ درصد	زمان: ۱۲۶ میلی‌ثانیه دقت: ۸۰ درصد
	
زمان: ۱۲۵ میلی‌ثانیه دقت: ۸۲ درصد	زمان: ۱۲۵ میلی‌ثانیه دقت: ۸۶ درصد
	
زمان: ۱۲۴ میلی‌ثانیه دقت: ۸۹ درصد	زمان: ۱۲۶ میلی‌ثانیه دقت: ۸۰ درصد

شکل ۱۷- نتایج تشخیص پهپاد کوادروتور Q3DR Solo

یکی از مهم‌ترین ویژگی‌هایی که در تمامی نتایج به چشم می‌خورد، این است که شبکه قادر به شناسایی پهپاد در جهت‌گیری‌های مختلف در صحنه بوده‌است. رفع این چالش توسط شبکه، نشان‌دهنده آموزش مناسب با تصاویر عمق شبیه‌سازی است. چرا که شبکه تنها با تصاویر عمق پهپاد ParrotAR در محیطی ساده و بدون پیش‌زمینه شلوغ و همراه با اختلاف عمق زیاد با محیط اطراف، آموزش دیده‌است، اما توانسته انواع پهپاد با ویژگی‌های فیزیکی مختلف، در جهت‌گیری‌های متفاوت در فواصل دور و نزدیک در پیش‌زمینه‌های شلوغ را با دقت بالا و زمان کوتاه تشخیص دهد.

همچنین، شبکه یادگیری عمیق پیشنهادی توانسته است عمل تشخیص را در تصاویری چالشی که پهپاد

در شکل (۱۷)، نتایج حاصل از تشخیص پهپاد کوادروتور Q3DRSolo نمایش داده شده‌است. بر اساس این نتایج شبکه مورد نظر به میانگین دقت ۸۴ درصد و میانگین زمان ۱۲۵ میلی‌ثانیه در تشخیص پهپاد رسیده است. در این دسته از تصاویر، پهپاد Q3DRSolo در محیط داخل در پیش‌زمینه‌های مختلف پرواز کرده و همان‌طور که مشاهده می‌شود در بعضی از قسمت‌ها که اختلاف عمق پهپاد با اشیاء اطراف ناچیز است، شبکه توانسته با دقت بالا و زمان پایین، پهپاد را تشخیص دهد.

بنابراین، این نتیجه نشان‌دهنده میزان تعمیم‌پذیری و قابلیت اطمینان بالای شبکه در تشخیص با انواع پهپاد است.

	
زمان: ۱۲۴ میلی ثانیه دقت: ۷۹ درصد	زمان: ۱۲۸ میلی ثانیه دقت: ۷۸ درصد
	
زمان: ۱۲۹ میلی ثانیه دقت: ۷۰ درصد	زمان: ۱۳۵ میلی ثانیه دقت: ۸۰ درصد
	
زمان: ۱۳۸ میلی ثانیه دقت: ۶۸ درصد	زمان: ۱۴۶ میلی ثانیه دقت: ۷۰ درصد

شکل ۱۹- نتایج تشخیص پهپاد هگزاروتور

۵- جمع‌بندی

همان‌طور که گفته شد، حضور پهپاد در فواصل دور در پیش‌زمینه شلوغ، مشکلات نوری و عدم امکان جداسازی پس‌زمینه در تصاویر مرئی، تشخیص پهپادها را با مشکل روبرو می‌کند. در صورتی که استفاده از تصاویر عمق به دلیل اختلاف عمق موجود بین اشیاء این مشکل را تا حد زیادی برطرف کرده‌است. در این پژوهش، با استفاده از انواع تصاویر عمق و با استفاده از نسخه دوم شبکه یادگیری عمیق YOLO تشخیص پهپاد با موفقیت صورت گرفته‌است. همچنین، مدل نهایی بر روی انواع پهپاد در محیط‌های مختلف، در انواع پس‌زمینه با جهت‌گیری‌های گوناگون ارزیابی گردید و مشاهده شد مدل در شرایط مختلف قادر به تشخیص انواع پهپاد است؛ بنابراین، به طور کلی می‌توان گفت در میان انواع روش‌های سنتی تشخیص شیء، تشخیص با استفاده از شبکه‌های یادگیری عمیق، نتایج بسیار مناسبی را ارائه می‌دهد. در ادامه، با مقایسه رویکردهای متفاوت در انواع شبکه‌های یادگیری عمیق، شبکه یادگیری عمیق پیچشی YOLO به عنوان شبکه مطلوب از لحاظ دقت و سرعت انتخاب گردید. در نهایت، با بررسی نتایج حاصل از تشخیص

اختلاف عمق ناچیزی با اشیاء اطراف دارد، با دقت بالا و زمان کم اجرا نماید. در شکل (۱۸) نمونه‌ای از این تصاویر ارائه شده‌است. این اختلاف عمق از طریق اختلاف ناچیز در درجه خاکستری پهپاد با اشیاء اطراف قابل مشاهده است.

	
زمان: ۱۲۶ میلی ثانیه دقت: ۸۰ درصد	زمان: ۱۲۵ میلی ثانیه دقت: ۸۱ درصد
	
زمان: ۱۲۴ میلی ثانیه دقت: ۸۰ درصد	زمان: ۱۲۵ میلی ثانیه دقت: ۸۰ درصد

شکل ۱۸- نتایج حاصل از تشخیص پهپاد در تصاویر عمق با اختلاف عمق ناچیز با محیط

به طور کلی، میانگین دقت و زمان نهایی تشخیص پهپاد در تصاویر عمق شبیه‌سازی از این سه نوع پهپاد به میزان ۸۴ درصد و ۱۲۵ میلی ثانیه رسید.

۴-۳-۲- ارزیابی نتایج در مجموعه تصاویر واقعی

مطابق شکل (۱۹) نتایج حاصل از ارزیابی دقت و زمان در شبکه یادگیری عمیق موردنظر در یک نوع پهپاد در مجموعه تصاویر عمق واقعی حاصل از الگوریتم SGM نمایش داده شده‌است. پهپاد هگزاروتور، با میانگین دقت ۷۴ درصد و میانگین زمان ۱۳۳ میلی ثانیه در عمق‌های مختلف تشخیص داده شده‌است. همان‌طور که گفته شد، تصاویر عمق حاصل از این پهپاد، از طریق الگوریتم تناظریابی حاصل شده‌است و مدل پیشنهادی در تمام این تصاویر توانایی تشخیص پهپاد را داشت، اما در بعضی از تصاویر دقت حاصل از تشخیص پایین و زمان تشخیص کمی بیشتر است؛ چرا که تصاویر حاصل به اندازه تصاویر شبیه‌سازی، ویژگی‌ها و جزئیات فیزیکی پهپاد را به طور واضح نمایش نمی‌دهند و دارای نویز هستند؛ اما با این‌که شبکه در روند آموزش تنها با تصاویر عمق شبیه‌سازی از یک نوع پهپاد در یک محیط شبیه‌سازی آموزش دیده، در مرحله آزمایش توانسته این نوع پهپاد را در محیط واقعی شناسایی کند.

در شبکه YOLOv2 و برطرف کردن چالش‌ها توسط این شبکه می‌توان گفت، شبکه یادگیری عمیق پیشنهادی به دلیل بهره‌مندی از سرعت قابل قبول و دقت مناسب در تشخیص انواع پهپاد و عملکردی مناسب در تشخیص پرنده در محیط‌هایی با پیش‌زمینه پیچیده در تصاویر واقعی و شبیه‌سازی، به عنوان شبکه مناسب در تشخیص پهپاد

انتخاب می‌شود. در تحقیقات بعدی می‌توان به منظور مقایسه نتایج حاصل از این پژوهش، از انواع دیگر شبکه‌ی یادگیری عمیق YOLO، مانند شبکه YOLOv3 به منظور بهبود دقت تشخیص استفاده کرد و علاوه بر تشخیص پهپاد، به بحث تعیین موقعیت نیز پرداخت.

مراجع

- [1] Shi, X., et al., Anti-Drone System with Multiple Surveillance Technologies: Architecture, Implementation, and Challenges. IEEE Communications Magazine, 2018. 56: p. 68-74.
- [2] Sathyamoorthy, D., A Review of Security Threats of Unmanned Aerial Vehicles and Mitigation Steps. The Journal of Defence and Security, 2015. 6: p. In press.
- [3] Gomez, C. and H. Purdie, UAV- based Photogrammetry and Geocomputing for Hazards and Disaster Risk Monitoring – A Review. Geoenvironmental Disasters, 2016. 3.
- [4] Drozdowicz, J., et al. 35 GHz FMCW drone detection system. in 2016 17th International Radar Symposium (IRS).
- [5] Haag, M.U.d., C.G. Bartone, and M.S. Braasch. Flight-test evaluation of small form-factor LiDAR and radar sensors for sUAS detect-and-avoid applications. in 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC).
- [6] Mezei, J., V. Fiaska, and A. Molnár. Drone sound detection. in 2015 16th IEEE International Symposium on Computational Intelligence and Informatics (CINTI). 2015.
- [7] Andraši, P., et al., Night-time detection of uavs using thermal infrared camera. Transportation Research Procedia, 2017. 28: p. 183-190.
- [8] Seidaliyeva, U., et al. Detection of loaded and unloaded UAV using deep neural network. in 2020 Fourth IEEE International Conference on Robotic Computing (IRC). 2020.
- [9] Behera, D.K. and A.B. Raj. Drone Detection and Classification using Deep Learning. in 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS). 2020.
- [10] Ahangarha, M., et al., Crop Land Change Monitoring Based on Deep Learning Algorithm Using Multi-temporal Hyperspectral Images. ISSGE, 2020. 10(2): p. 79-89.
- [11] Learning, D., Deep learning. High-Dimensional Fuzzy Clustering, 2020.
- [12] Ji, S., et al., 3D Convolutional Neural Networks for Human Action Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013. 35(1): p. 221-231.
- [13] Samadzadegan, F., et al., Detection and Recognition of Drones Based on a Deep Convolutional Neural Network Using Visible Imagery. Aerospace, 2022. 9(1).
- [14] Carrio, A., et al., Drone Detection Using Depth Maps. 2018. 1034-1037.
- [15] Aslam, A. and M. Ansari, Depth-map generation using pixel matching in stereoscopic pair of images. arXiv preprint arXiv:1902.03471, 2019.
- [16] Lee, S., Depth camera image processing and applications. 2012. 545-548.
- [17] Jiang, X., et al., Advances in Depth Image Analysis and Applications. 2013.
- [18] Xia, L., C.-C. Chen, and J.K. Aggarwal. Human detection using depth information by kinect. in CVPR 2011 workshops. 2011. IEEE.
- [19] Suarez, J. and R.R. Murphy. Hand gesture recognition with depth images: A review. in 2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication. 2012. IEEE.
- [20] Zhao, Z.-Q., et al., Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems, 2019. 30(11): p. 3212-3232.
- [21] Girshick, R., et al. Rich feature hierarchies for accurate object detection and semantic segmentation. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
- [22] He, K., et al., Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE

- transactions on pattern analysis and machine intelligence, 2015. 37(9): p. 1904-1916.
- [23] Girshick, R. Fast r-cnn. in Proceedings of the IEEE international conference on computer vision. 2015.
- [24] Ren, S., et al., Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE transactions on pattern analysis and machine intelligence, 2016. 39(6): p. 1137-1149.
- [25] Dai, J., et al., R-fcn: Object detection via region-based fully convolutional networks. arXiv preprint arXiv:1605.06409,2016.
- [26] Lin, T.-Y., et al. Feature pyramid networks for object detection. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
- [27] Erhan, D., et al. Scalable object detection using deep neural networks. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
- [28] Najibi, M., M. Rastegari, and L.S. Davis. G-cnn: an iterative grid based object detector. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
- [29] Redmon, J., et al. You only look once: Unified, real-time object detection. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
- [30] Liu, W., et al. Ssd: Single shot multibox detector. in European conference on computer vision. 2016. Springer.
- [31] Redmon, J. and A. Farhadi. YOLO9000: better, faster, stronger. in Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.
- [32] Redmon, J. and A. Farhadi, Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
- [33] Fu, C.-Y., et al., Dssd: Deconvolutional single shot detector. arXiv preprint arXiv:1701.06659, 2017.
- [34] Shen, Z., et al. Dsod: Learning deeply supervised object detectors from scratch. in Proceedings of the IEEE international conference on computer vision. 2017.
- [35] Zsedrovits, T., et al. Collision avoidance for UAV using visual detection. in 2011 IEEE International Symposium of Circuits and Systems (ISCAS). 2011. IEEE.
- [36] Wu, Y., Y. Sui, and G. Wang, Vision-based real-time aerial object localization and tracking for UAV sensing system. IEEE Access, 2017. 5: p. 23969-23978.
- [37] Lai, J., L. Mejias, and J. Ford, Airborne Vision-Based Collision-Detection System. Journal of Field Robotics, 2011. 28: p. 137-157.
- [38] de Haag, M.U., C.G. Bartone, and M.S. Braasch. Flight-test evaluation of small form-factor LiDAR and radar sensors for sUAS detect-and-avoid applications. in 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC).2016. IEEE.
- [39] Chen, L., et al., Transforming a 3-D LiDAR Point Cloud Into a 2-D Dense Depth Map Through a Parameter Self-Adaptive Framework. IEEE Transactions on Intelligent Transportation Systems, 2016. 18: p. 1-12.
- [40] Lopez, B.T. and J.P. How. Aggressive 3-D collision avoidance for high-speed navigation. in ICRA. 2017.
- [41] Song, S. and J. Xiao. Sliding shapes for 3d object detection in depth images. in European conference on computer vision. 2014. Springer.
- [42] Saqib, M., et al. A study on detecting drones using deep convolutional neural networks. in 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). 2017.
- [43] Aker, C. and S. Kalkan. Using deep networks for drone detection. in 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). 2017.
- [44] Nenchoo, B. and S. Tantrairatn. Real-Time 3D UAV Pose Estimation by Visualization. in Multidisciplinary Digital Publishing Institute Proceedings. 2020.
- [45] Carrio, A., et al., Onboard Detection and Localization of Drones Using Depth Maps. IEEE Access, 2020. 8: p. 30480.
- [46] tzutalin, Labellmg. Git code. 2015: p. <https://github.com/tzutalin/labellmg>.
- [47] Hirschmuller, H., Stereo Processing by Semiglobal Matching and Mutual Information. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008. 30(2): p. 328-341.
- [48] Kallwies, J., et al. Triple-SGM: stereo processing using semi-global matching with cost fusion. in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2020.
- [49] Redmon, J., Darknet: Open Source Neural Networks in C. 2013--2016: p. <http://pjreddie.com/darknet/>.

- [50] Brownlee, J., A Gentle Introduction to the Rectified Linear Unit (ReLU). 2019: p. [https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/#:~:text=The%20Leaky%20ReLU%20\(LReLU%20or,Neural%20Network%20Acoustic%20Models%2C%202013.](https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/#:~:text=The%20Leaky%20ReLU%20(LReLU%20or,Neural%20Network%20Acoustic%20Models%2C%202013.)
- [51] Yumi, Object Detection using YOLOv2 on Pascal VOC2012. 2018: p. [https://fairyonice.github.io/Part 4 Object Detection with Yolo using VOC 2012 data loss.html.](https://fairyonice.github.io/Part%204%20Object%20Detection%20with%20Yolo%20using%20VOC%202012%20data%20loss.html)
- [52] K, S., Non-maximum Suppression (NMS). 019 :p. [https://towardsdatascience.com/non-maximum-suppression-nms-93ce178e177c.](https://towardsdatascience.com/non-maximum-suppression-nms-93ce178e177c)
- [53] Singh, A., Selecting the Right Bounding Box Using Non-Max Suppression (with implementation). 2020: p. [https://www.analyticsvidhya.com/blog/2020/08/selecting-the-right-bounding-box-using-non-max-suppression-with-implementation/.](https://www.analyticsvidhya.com/blog/2020/08/selecting-the-right-bounding-box-using-non-max-suppression-with-implementation/)
- [54] Susmaga, R., Confusion matrix visualization, in Intelligent information processing and web mining. 2004, Springer. p.107-116.
- [55] Shah, S., et al. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. in Field and service robotics. 2018. Springer