نشریه علمی علوم و فنون نقشه‌برداری، دوره ۱۴، شماره ۴، صفحه ۵۹ تا ۷۸، خرداد ماه ۱۴۰۴ - مقاله پژوهشی

تلفیق یادگیری عمیق و اصول فتوگرامتری برای شناسایی و دسته‌بندی هندسی چاله‌ها در یک سیستم تصویرمبنای زمینی

مهدی امیردهی[۱]، علی حسینی نوه*[۲]

[۱] دانشجوی ارشد فتوگرامتری-دانشکده نقشه‌برداری-دانشگاه صنعتی خواجه‌نصیرالدین‌طوسی
m.amirdehi@email.kntu.ac.ir

[۲] دانشیار دانشکده نقشه‌برداری-دانشگاه صنعتی خواجه‌نصیرالدین‌طوسی
hosseininaveh@kntu.ac.ir

(دریافت: آذر ۱۴۰۳، تصویب: بهمن ۱۴۰۳)

چکیده

با افزایش تردد، سلامت جاده‌ها پراهمیت‌تر شده است و با پیشرفت هوش مصنوعی و الگوریتم‌های رباتیک، می‌توان بهتر و دقیق‌تر به پایش سلامت آن‌ها پرداخت. در این مقاله، هدف شناسایی و دسته‌بندی هندسی چاله‌ها با گوشی همراه است و به دلیل اینکه گوشی همراه دارای یک دوربین و فاقد طول خط پایه می‌باشد، مقادیر به‌صورت نسبی محاسبه شدند. نوآوری این تحقیق بر این است که از شبکه عصبی YOLO برای شناسایی چاله‌ها به همراه تعیین موقعیت مختصات دوبعدی تصویری آن‌ها و از الگوریتم ORB SLAM 3 به‌منظور تعیین موقعیت و وضعیت دوربین فیلم‌برداری استفاده شد. برای تعیین موقعیت سه‌بعدی چاله‌ها، ابتدا فیلم مسیر موردنظر همراه با زمان اخذ داده در هرلحظه توسط نرم‌افزار VIRec نصب‌شده بر روی گوشی ذخیره گردید و به‌عنوان ورودی به الگوریتم ORB SLAM 3 داده شد. موقعیت و وضعیت دوربین در هر فریم توسط این الگوریتم به دست آمد و منجر به ایجاد ماتریس نگاشت برای هر فریم از فیلم شد. جهت شناسایی چاله‌ها نیز، شبکه عصبی YOLO V8m، مدل دنبال کننده با دقت ۹۲ ٪ پس از آموزش با داده‌های مربوط به سطح خیابان‌های شهری کشورهای مختلف مورداستفاده قرار گرفت. پس از شناسایی نیز، چاله‌ها با بیضی برازش داده‌شده بر آن‌ها در فریم‌های مختلف دنبال شدند تا مختصات تصویری آن‌ها در موقعیت‌های مختلف فیلم به دست آید. درنهایت با استفاده از موقعیت و وضعیت دوربین حاصل از الگوریتم ORB SLAM 3 و موقعیت تصویری دوبعدی چاله‌های حاصل از YOLO، مثلث‌بندی و سرشکنی دسته اشعه صورت گرفت و مختصات دو سوی محور بزرگ و کوچک هر بیضی به‌صورت نسبی تخمین زده شد. جهت ارزیابی دقت نقاط تبدیل‌شده از پارامتر خطای نگاشت مجدد استفاده گردید که بیانگر میزان خطای زیر ۱ تا ۳/۵ پیکسل بوده است. به‌منظور بررسی بیشتر نتایج ازنظر کمّی، به مقایسه مساحت‌های نسبی پرداخته شد. در ابتدا مساحت نرمال‌شده حاصل از نقاط تبدیل‌شده هر یک از چاله‌ها محاسبه گردید و با مساحت متناظر نرمال‌شده خود که توسط گیرنده مولتی فرکانس G1 Plus Sout با دقت ۰/۰۱۲ متر در اختیار بود مقایسه شدند. نتایج گویای اختلاف ۲ تا ۵ درصدی بین چاله‌های محاسبه‌شده با واقعیت زمینی بوده است.

واژگان کلیدی: YOLO، مثلث‌بندی، چاله، Visual SLAM، Visual Odometry، یادگیری عمیق، رباتیک

* نویسنده رابط

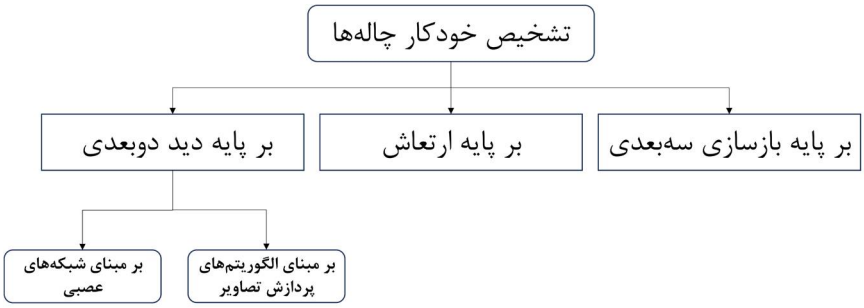
۱- مقدمه

همیشه سلامت جاده‌ها جزو مسائل بااهمیت به شمار می‌آمد که امروزه با افزایش وسایل نقلیه و حجم عبور و مرور، به اهمیت آن افزوده‌شده است. یکی از اصلی‌ترین معضلات در بحث پایش جاده‌ای، وجود چاله‌ها می‌باشد که منجر به تخریب جاده‌ها می‌شود و می‌تواند خسارات مالی و گاهاً جانی زیادی از قبیل آسیب به‌وسیله نقلیه و یا انحراف آن‌ها را به همراه داشته باشد[۱-۳]. به همین دلیل شناسایی و دسته‌بندی چاله‌ها به جهت ارزیابی جاده‌ها می‌تواند از جنبه‌های مختلفی حائز اهمیت باشد. ازجمله فواید آن می‌توان به موضوع ارجاع به شهرداری جهت ترمیم و بازسازی و همچنین هشدار به راننده در هنگام رانندگی اشاره کرد. در سال ۲۰۲۱، محققین به ارائه روشی برای تشخیص چاله‌ها در ماشین‌های خودران پرداختند اما صرف تشخیص و نمایش چاله‌ها مدنظر بود[۴]. در ادامه این پژوهش، در سال ۲۰۲۳ آقای سن به تولید الگوریتمی پرداختند که علاوه بر تشخیص چاله، به‌عنوان یک سیستم هشدار به راننده نیز عمل کند به این‌گونه که در فاصله معینی از ماشین، وجود چاله اعلام شود[۵]. در راستای همین تحقیق، ساثویک و همکاران به توسعه سیستم مشابه هشدار راننده پرداختند با این تفاوت که برای کاهش هزینه، از سنسورهای گوشی هوشمند جهت شناسایی و تعیین فاصله چاله استفاده شد. همچنین در این کار انجام‌گرفته، کاربر می‌توانست گزارش وجود چاله را ثبت کند که برای مراکز عمومی و مربوطه نیز کارآمد باشد[۶].

همان‌طور که بیان شد، در کارهای انجام‌شده، اطلاعاتی از وضعیت چاله‌ها در اختیار قرار نمی‌گیرد و الگوریتم‌های ایجادشده صرفاً به تشخیص آن‌ها و یا ثبت وجود داشتنشان بر روی نقشه می‌پردازند. در این تحقیق، به‌منظور تشخیص چاله و دسته‌بندی هندسی آن‌ها، به استخراج اطلاعات کمّی دیگری مانند شکل، موقعیت و مساحتشان به‌صورت نسبی با استفاده از سنسورهای گوشی هوشمند پرداخته شد. اطلاعات بیان‌شده می‌تواند جهت مدیریت بهتر به کمک سازمان‌های مربوطه برای بررسی بهتر چاله‌ها و پایش دقیق‌تر جاده‌ای به کار گرفته شود .

در ادامه، کارهای مشابه انجام‌شده در این حوزه مرور شده و سپس روش پیشنهادی و نحوه پیاده‌سازی آن آورده شده است. درنهایت نیز نتایج به‌دست‌آمده در بخش ۵ بیان گردید.

۲- مرور کارهای گذشته در زمینه تشخیص خودکار چاله



شکل ۱- روش‌های تشخیص خودکار چاله‌ها

تحقیقات در مورد روش تشخیص چاله‌ها به‌طور عمده به سه دسته تقسیم می‌شود: ۱- بر پایه حسگر ارتعاش[۱]، ۲- بر پایه بازسازی سه‌بعدی به‌وسیله لیزر اسکنرها و ۳- بر پایه دید دوبعدی [۱-۳]. هریک دارای معایب و مزایایی هستند که به بیان هریک در ادامه پرداخته خواهد شد.

۲-۱- روش مبتنی بر حسگر ارتعاش

این روش از داده‌های جمع‌آوری‌شده از حسگرهای شتاب‌سنج داخل خودرو استفاده می‌کند[۳] و سپس به آنالیز سیگنال[۲] می‌پردازد. در این روش چاله به‌عنوان یک تکانه مشخص و قابل‌توجه در نظر گرفته می‌شود. درعین‌حال که این روش مقرون به‌صرفه‌ترین در بین دو روش دیگر است و

۱ Vibration base

۲ Signal processing

نیاز به ذخیره‌سازی اطلاعات کمی دارد و باعث عملکرد بهتر سیستم در زمان لحظه‌ای[۱] می‌شود اما نمی‌تواند تمامی سطوح جاده را پایش کند و فقط قابلیت شناسایی چاله‌های موجود در مسیر چرخ‌ها را دارد. همچنین تمایز بین علت ارتعاش که می‌تواند چاله، دست‌انداز و یا موانع دیگر باشد نیز از موارد چالش‌برانگیز این روش می‌باشد. صحت و دقت این روش نیز تحت تأثیر فاکتورهایی همچون محل قرارگیری سنسور و شرایط جاده‌ای است[۷, ۸]. علاوه بر ایرادات ذکرشده، یکی از اهداف این پژوهش، طبقه‌بندی چاله‌ها بر اساس مساحت آن‌ها و استخراج شکل آن‌ها می‌باشد که این روش به دلیل اینکه فقط اطلاعات ارتعاش سنسورهای شتاب را تجزیه‌وتحلیل می‌کند در ارائه شکل و مساحت چاله‌ها دارای محدودیت است[۲].

۲-۲- روش مبتنی بر بازسازی سه‌بعدی

این روش به ایجاد و نمایش سه‌بعدی چاله‌ها با استفاده از چندین تصویر از یک صحنه در حالت‌های مدل ابر نقطه، مدل مِش[۲] و مدل‌های هندسی می‌پردازد[۳]. با استفاده از روش‌های مبتنی بر دید استریو[۳] ازجمله فتوگرامتری و تکنیک‌های ساختار حاصل از حرکت (SFM) [۴] می‌توان اندازه، شکل و عمق چاله را ارزیابی کرد و حجم آن را با دقت به دست آورد. مسئله تعیین موقعیت نسبی دوربین و بازسازی ساختار سه‌بعدی عوارض در جوامع بینایی ماشین با نام SFM شناخته می‌شود[۹, ۱۰]. برخی دیگر از روش‌های متداول مبتنی بر بازسازی سه‌بعدی چاله‌ها شامل استفاده از لیزر اسکنر سه‌بعدی، دید استریو و سنسور کینکت[۵] نیز می‌باشد[۱]. سنسور کینکت یک دستگاه حسگر حرکتی است که ابتدا توسط مایکروسافت برای کنسول‌های بازی ایکس باکس توسعه یافت و بعداً برای کاربردهای مختلف دیگر ازجمله رباتیک، مراقبت‌های بهداشتی و اسکن سه‌بعدی گسترش یافت. این روش علی‌رغم دقت خوبی که دارد هزینه‌بر و زمان‌بر می‌باشد. برای مثال لیزر اسکنرها، تجهیزات گران‌قیمتی هستند و استفاده از دوربین‌های استریو نیز مشکلات مربوط به خود را دارند همچون اینکه باید هردو دوربین به‌طور دقیق تنظیم شوند زیرا ممکن است در صورت وجود لرزش در حین حرکت خودرو دوربین‌ها ناهماهنگ شوند و درنتیجه بر کیفیت نتیجه تأثیر بگذارند. همچنین استفاده از سنسور کینکت نیاز به تحقیقات بیشتر جهت رفع خطاها برای استفاده در کارهای دقیق دارد. در کنار موارد ذکرشده، کیفیت و دقت مدل سه‌بعدی ایجادشده با استفاده از روش بازسازی سه‌بعدی به عواملی مانند تعداد، توزیع و وضوح تصاویر، کالیبراسیون دوربین و اثربخشی الگوریتم‌های تطبیق ویژگی[۶] بستگی دارد که می‌تواند تأثیر منفی بگذارند[۱۱, ۱۲].

۲-۳- روش مبتنی بر دید دوبعدی

این روش با توجه به اینکه از تصاویر و یا ویدیو به‌عنوان ورودی استفاده کند به دو دسته برمبنای تصویر و برمبنای ویدئو تقسیم می‌شود و در ادامه با پردازش آن‌ها و یادگیری عمیق به نمایش چاله‌ها می‌پردازد[۱, ۳]. نسبت به روش بازسازی سه‌بعدی این روش دارای هزینه کمتری می‌باشد و قابلیت اجرا در لحظه را دارد اما چون برمبنای بینایی است، بنابراین قابل‌تصور می‌باشد که تا چه میزان متأثر از نور و شرایط آب و هوایی محیط می‌باشد که درنتیجه کار اثرگذار خواهد بود. همچنین به دلیل اینکه از اطلاعات دوبعدی در این روش استفاده می‌شود، برای رسیدن به مشخصات هندسی همچون اندازه و مساحت مشکلاتی وجود دارد که نیاز به تحقیقات بیشتر در این زمینه دارد [۲, ۱۳].

جدول ۱- محدودیت‌های هریک از روش‌های تشخیص چاله

رویکردها	محدودیت‌ها
ارتعاش	- سنسورها ممکن است به دلیل شرایط جاده آسیب ببینند - پایش نکردن تمامیت مسیر - فاقد اطلاعات مساحت و شکل چاله‌ها
بازسازی سه‌بعدی	- اسکنر لیزری سه‌بعدی گران‌قیمت - تلاش محاسباتی بالا برای بازسازی سطح - امکان به هم خوردن پارامترهای دوربین در صورت استفاده از دوربین استریو
دید دوبعدی	- امکان به هم خوردن پارامتر دوربین در شرایط مختلف جاده - وابسته به شرایط آب‌وهوایی و نوری

۱ Real time

۲ Mesh model

۳ Stereo Vision

۵ Structure from motion

۵ kinect

۶ Feature matching

در جدول ۱ به بررسی و بیان محدودیت‌هایی که اجرای هر روشی خواهد داشت، پرداخته‌شده است.

به‌طور خلاصه، تکنیک‌های مختلفی برای تشخیص چاله‌ها استفاده‌شده است که هرکدام نقاط قوت و ضعف خاص خود را دارند. عواملی که جزو فاکتورهای مهم این مقاله هستند شامل، نیاز به داشتن دقت در عین سرعت مناسب، مقرون‌به‌صرفه بودن و نیز قابلیت اجرایی داشتن جهت برآورده ساختن هدف‌های این پژوهش علمی می‌باشد که روش مبتنی بر دید دوبعدی این نیازها را برآورده می‌کند. عیب‌های دو روش اول شامل ارزیابی منحصر به مسیر عبوری چرخ‌ها در روش مبتنی بر حسگر ارتعاش در کنار عدم امکان استخراج شکل چاله و نیز هزینه زیاد لیزر اسکنرها و زمان‌برتر بودن در روش بازسازی سه‌بعدی ازجمله عواملی بودند که شرایط لازم را برای اجرای این پروژه فراهم نکردند. در روش مبتنی بر دید دوبعدی نیز چالش‌هایی در خصوص به‌کارگیری دوربین تک‌چشمی[۱] وجود دارد که در این مقاله به آن پرداخته خواهد شد.

در تشخیص عوارض بر پایه دید دوبعدی، پیش‌تر متدهای گوناگونی پیاده‌سازی شده‌اند که موردبررسی قرارگرفته‌اند، هرکدام دارای معایب و فوایدی بودند. در ادامه، پژوهش‌های انجام‌شده در چند سال اخیر مربوط به تشخیص خودکار چاله بر مبنای تصویر دوبعدی بررسی‌شده است.

۲-۳-۱- بر مبنای الگوریتم‌های پردازش تصاویر

همان‌طور که بیان شد تشخیص چاله بر پایه دید دوبعدی با روش‌های مختلفی تا به الان انجام‌شده است یکی از این روش‌ها، مبتنی بر الگوریتم‌های پردازش تصویر می‌باشد که تحقیقاتی نیز در این زمینه صورت گرفته است. برای مثال گائو و همکاران به تشخیص چاله‌ها با استفاده از ادغام ویژگی‌های بافتی و مقیاس خاکستری[۲] پرداختند، همچنین از مدل یادگیری ماشین LIBSVM نیز استفاده کردند. LIBSVM یک کتابخانه پرکاربرد برای SVM[۳] که یک الگوریتم یادگیری ماشین برای کارهای طبقه‌بندی و رگرسیون است می‌باشد. در این پژوهش ابتدا چاله از طریق فیلتر بافت، سطح خاکستری تصویر و عملیات مورفولوژیکی تقسیم می‌شود. سپس در مرحله دوم چاله‌ها، با کمّی کردن ویژگی‌های تصویر و استفاده از طبقه‌بندی کننده LIBSVM، جهت شناسایی مقادیر ویژگی به دو دسته آموزش و تست تقسیم می‌شوند. روش پیشنهادی با چهار روش دیگر OTSU، K-means، الگوریتم Watershed و تشخیص لبه مقایسه شد. روش پیشنهادی نتایج با دقت بیشتری را ارائه داده بود[۱۴].

تکنیک‌های پردازش تصاویر عموماً ساده و مقرون‌به‌صرفه هستند؛ اما حجم بالای تصاویر و اعمال تکنیک‌های این حوزه بر آن‌ها جزو چالش‌های این روش می‌باشد که بار محاسباتی قابل‌توجهی را ایجاد می‌کند و منجر به اتلاف وقت زیادی جهت پردازش‌ها صرف می‌شود[۱۵]. یکی از نیازهای این مقاله نیز تشخیص عارضه در کمترین زمان ممکن است که این روش از این لحاظ دارای محدودیت است.

۲-۳-۲- بر مبنای شبکه‌های عصبی

استفاده از شبکه‌های عصبی در تشخیص عوارض دارای تنوع زیادی می‌باشد. در این روش، تصاویر جمع‌آوری‌شده پس از طی مرحله پیش‌پردازش به‌عنوان ورودی وارد شبکه عصبی می‌شوند. در آموزش شبکه عصبی حتی نوع تصویر استفاده‌شده در آن می‌تواند از دسته‌های گوناگونی باشد. برای مثال، بهاتیا و همکاران از تصاویر حرارتی[۴] برای آموزش شبکه CNN[۵] استفاده کردند[۱۶]. تفاوت بین تصاویر نوری[۶] و حرارتی در شرایط شکل‌گیری تصویر است که تصویر نوری برمبنای نور و تصویر حرارتی برمبنای حرارت ساطع‌شده از شئ است. از مزیت‌های این روش دقت بالا در شرایط آب و هوایی مختلف، عملکرد یکسان در شب و روز، قیمت کمتر نسبت به دیگر سنسورهای دید در شب، دقت بالا و پیچیدگی کم می‌باشد، همچنین ریسک عدم شناسایی برخی چاله‌ها نیز در این روش وجود ندارد. اما اجرای این روش نیازمند دوربین‌های حرارتی می‌باشد که هزینه‌برتر نسبت به دوربین‌های معمولی هستند و برای کارهایی همچون این پروژه توجیه اقتصادی ندارد.

در برخی مقالات نیز به دلیل وجود شبکه‌های عصبی گوناگون و تفاوت عملکرد هریک، تشخیص چاله‌ها توسط

---

۱ Monocular

۲ Gray scale

۳ Support Vector Machines

۴ Thermal

۵ Convolutional Neural Network

۶ Optical

یادگیری ماشین و یادگیری عمیق صورت گرفته است. این شبکه‌ها می‌توانند جهت بهبود در عملکرد دستخوش تغییرات قرار گیرند و یا از شبکه‌های موجود استفاده شود[۱۷, ۱۸].

برای مثال در مقاله منتشرشده توسط فن و همکاران، تشخیص چاله با استفاده از شبکه‌های YOLO V3[۱]، SSD، HOG[۲] با SVM و Faster R-CNN انجام شد. در این مقاله Faster R-CNN در مقابل YOLO V3 دارای سرعت بیشتری بوده است، دقت این الگوریتم نیز در داده‌های کم، بهتر از YOLO بود اما با افزایش حجم داده‌ها از این دقت کاسته شد، همچنین زمانی که ابعاد چاله‌ها کاهش می‌یافت، YOLO عملکرد بهتر خودش را در مقایسه با بقیه نشان می‌داد. YOLO V3 در این پژوهش به‌دقت ٪۸۲ دست‌یافت و عملکرد بهتری را نسبت به بقیه الگوریتم‌ها نشان داد[۱۹]. YOLO به شبکه عصبی تشخیص اشیا مشهور است که در سال ۲۰۱۵، ردمون و همکاران اولین نسخه را معرفی کردند[۲۰]. به‌مرورزمان این شبکه عصبی بهبود یافت و از نسخه‌های جدیدتر در نسخه‌های بالاتر رونمایی شد و از رقبای خود پیشی گرفت. با توجه به عملکرد بهتر این شبکه در مقایسه با دیگر شبکه‌های عصبی موجود، عموماً از این شبکه جهت تشخیص اشیا یا کارهای مرتبط به آن در حوزه بینایی ماشین استفاده می‌شود. نمونه استفاده از این الگوریتم در مقاله کاویتا و همکاران می‌باشد. در این مقاله، ورژن V3 الگوریتم تشخیص عوارض ازجمله چاله، بر روی برد جداگانه Raspberry Pi پیاده‌سازی شد تا بتواند به‌صورت خودکار و بر روی مانیتور در ماشین‌های خودران قابل‌اجرا باشد و به‌صورت آنی نتیجه تشخیص الگوریتم نسبت به عوارض بر روی صفحه‌نمایش داده شود[۴]. Raspberry Pi یک کامپیوتر تک برد کوچک، مقرون‌به‌صرفه و همه‌کاره است که توسط بنیاد Raspberry Pi توسعه‌یافته است. به دلیل اندازه جمع‌وجور، هزینه کم و قابلیت‌های قوی، به‌طور گسترده در آموزش، پروژه‌های سرگرمی و برنامه‌های حرفه‌ای استفاده می‌شود.

در استفاده دیگر از YOLO که مرتبط با تشخیص چاله نیز می‌باشد، می‌توان به کار انجام‌شده در سال ۲۰۲۳ که توسط آقای سن انجام‌شده اشاره کرد. در این مقاله، هدف ایجاد یک سیستم اعلام‌خطر به راننده بود که برای رسیدن به این سیستم دقیق و سریع از DashCam استفاده شد. دو ورژن V3و V4 در این مقاله مورداستفاده قرار گرفت که ورژن V4 نتایج بهتری را نشان داده است. الگوریتم نزدیک به ۷۰ درصد موفق به شناسایی شد و چاله‌ها در فاصله ۱۰ متری در حالتی که سرعت وسیله نقلیه ۳۰km/hr بود صورت گرفت[۵].

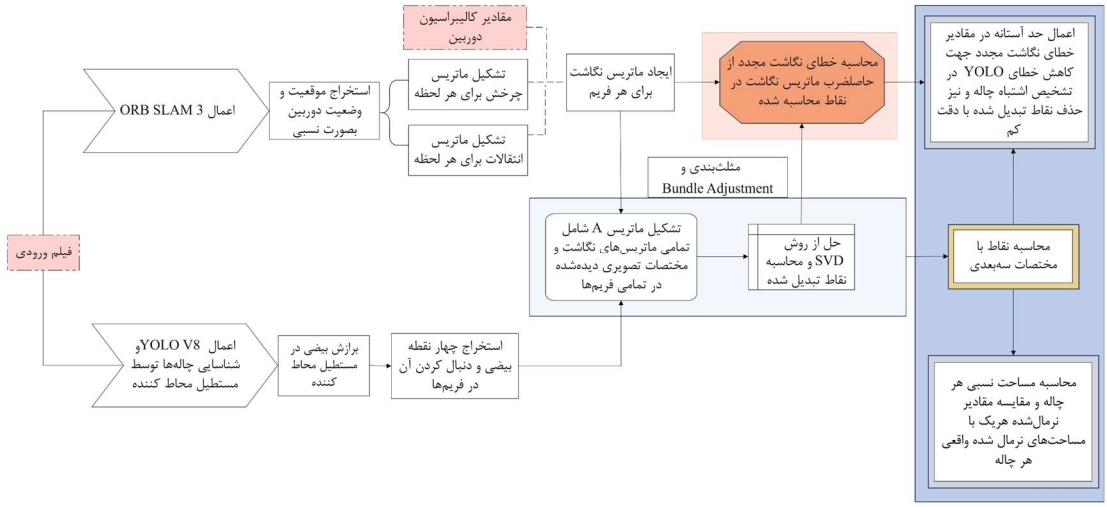
در سال ۲۰۲۲ نیز کاری مرتبط با مطالب بیان‌شده با استفاده از شبکه عصبی YOLO صورت گرفت [۶]. محققان در این پروژه به ساخت نرم‌افزاری بر مبنای گوشی هوشمند پرداختند که نیازی به نصب سنسورهای مجزا ندارد و از سنسورهای خود گوشی هوشمند و دوربین آن کمک می‌گیرد تا هزینه ساخت را کاهش دهند. این اپلیکیشن در دو حالت عمومی و دولتی ایجاد گردید که در حالت عمومی، هم کاربر می‌تواند چاله‌های جدید را معرفی کند و هم از هشداری که نرم‌افزار در فاصله ۵۰ متری از چاله به کاربر می‌دهد، استفاده کند. در حالت دولتی هم به‌منظور گزارش‌نویسی و اقدامات لازم به سازمان مربوطه اطلاع‌رسانی می‌کند. در این پژوهش از YOLO V3 استفاده شد که با استفاده از ۶۳۴ تصویر و ۴۰۰۰ تکرار، آموزش دید. علی‌رغم کارهای خوبی که در این مقاله صورت گرفته است مانند استفاده از سنسورهای گوشی هوشمند، قابلت تعامل با کاربر و هشدار در فاصله مناسب ۵۰ متری، دارای ایراداتی نیز است، از قبیل استفاده از نسخه V3 که دقت و سرعت کمتری نسبت به نسخه‌های بالاتر دارد و استفاده از عکس‌هایی باکیفیت بالا که باعث می‌شود در صورت ورود تصویر باکیفیت کم، دقت شبکه کاهش یابد. همچنین اطلاعاتی در مورد چاله نیز در دسترس قرار نمی‌گیرد.

به‌طورکلی در بین شبکه‌های عصبی بیان‌شده، شبکه عصبی YOLO داری سرعت خوب در عین داشتن دقت بالا است. همچنین پارامترهایی مانند سرعت تشخیص و مدت‌زمان آن، تا حد زیادی در نسخه‌های بالاتر با توجه به مطالب جمع‌آوری‌شده بهبودیافته‌اند که در زمان نگارش این مقاله، نسخه V8 آن موجود بوده است و طبق بررسی‌ها، نتایج مطلوب‌تری را ارائه داد. به‌طورکلی در کارهای انجام‌شده و مقالات اشاره‌شده، همان‌طور که بیان شد فقط بحث شناسایی چاله‌ها مطرح گردید و در برخی موارد تحقیقاتی برای کمک به راننده انجام‌گرفته است درصورتی‌که می‌توان اطلاعات بیشتری مانند موقعیت سه‌بعدی چاله‌ها و مساحتشان را به دست آورد. دستیابی

۴You Only Look Once

۲ Histogram of Oriented Gradients

به موقعیت سه‌بعدی چاله‌ها، نیازمند ایجاد ارتباط بین فضای دوبعدی تصاویر متوالی و موقعیت سه‌بعدی چاله‌ها است که در این تحقیق به ارائه راهکاری برای توسعه چنین ارتباطی پرداخته‌شده است.



شکل ۲- مراحل کلی انجام روش پیشنهاد شده

۳- روش پیشنهادی

این تحقیق شامل چهار بخش تشخیص چاله به‌صورت دوبعدی، برازش بیضی در محیط دوبعدی بر آن، تعیین موقعیت و وضعیت فریم‌های متوالی و انجام مثلث‌بندی[۱] همراه با سرشکنی دسته اشعه[۲] جهت تخمین موقعیت سه‌بعدی بیضی برازش داده‌شده بر چاله‌ها می‌باشد. در بخش مربوط به شناسایی چاله‌ها از شبکه عصبی YOLO V8m استفاده گردید و برای برازش بیضی بر چاله‌های شناسایی‌شده از محیط پایتون کمک گرفته شد. برای تعیین موقعیت و وضعیت دوربین نیز الگوریتم ORB SLAM 3 بکار گرفته شد. درنهایت با استفاده از موقعیت‌های تصویری به‌دست‌آمده از بیضی‌های برازش داده‌شده حاصل از YOLO و موقعیت و وضعیت فریم‌ها توسط ORB SLAM 3 به مثلث‌بندی و اعمال سرشکنی دسته اشعه جهت دستیابی موقعیت‌های سه‌بعدی پرداخته شد.

در شکل ۲، جایگاه انجام هریک از الگوریتم‌ها و نیز عملیات‌های موردنیاز دیگر ذکرشده است. در ادامه نیز مراحل انجام کار بیان خواهد شد و توضیحات مربوط به هر بخش بیان می‌شود.

طبق شکل ۲، این کار در دو قسمت که در راستای یکدیگر می‌باشند صورت می‌گیرد که یک قسمت آن مربوط به بحث شناسایی و بحث دیگر، تعیین موقعیت دوربین است.

در ادامه، بخش‌ها و زیر بخش‌های هریک شرح داده‌شده‌اند.

۳-۱- شناسایی و دنبال کردن چاله با استفاده از شبکه عصبی YOLO

در ابتدا، فیلم ورودی، به شبکه عصبی YOLO V8 آموزش داده‌شده وارد می‌شود تا عملیات شناسایی چاله‌ها در آن صورت پذیرد. YOLO برخلاف الگوریتم‌های تشخیص دیگر، تمام اشیاء موجود در تصویر را در یک مرحله پیش‌بینی می‌کند به همین دلیل دارای سرعت ارائه بالایی نیز می‌باشد. این الگوریتم مسئله تشخیص اشیاء را به‌عنوان یک مسئله رگرسیون بجای مسئله طبقه‌بندی می‌بیند و با تکنیک جداسازی فضایی بین جعبه‌های محاط کننده به الصاق احتمال به هریک از تصاویر تشخیص داده‌شده توسط یک شبکه CNN می‌پردازد[۲۱]. این الگوریتم تصویر را به بخش‌های کوچک‌تر تقسیم کرده و هر بخش را به‌صورت مستقل پردازش می‌کند. سپس آن‌ها را با مدل‌های پیش پردازشگر مقایسه می‌کند تا بهترین پیش پردازشگر را برای هر بخش انتخاب کند. این فرآیند شامل مراحل تقسیم تصویر به سلول‌های کوچک‌تر، پردازش هر سلول با مدل‌های پیش پردازشگر، انتخاب بهترین پیش پردازشگر برای هر سلول و ترکیب نتایج برای تشخیص اشیاء است. YOLO V8 یکی از

۱ Triangulation

۲ Bundle adjustment

نسخه‌های پیشرفته‌تر و دقیق‌تر الگوریتم YOLO برای تشخیص اشیاء است. این نسخه توسط گروه تحقیقاتی Meta AI توسعه‌یافته و در سال ۲۰۲۳ معرفی شد[۱۳]. YOLO V8 در مقایسه با معماری‌های پیشین خود مانند YOLO V7تکامل‌یافته است و با بهبودهایی در عملکرد و سرعت عمل ارائه‌شده است. برخی از ویژگی‌های کلیدی این نسخه عبارت‌اند از استفاده از مدل‌های پیش پردازشگر برای افزایش دقت تشخیص، بهینه‌سازی بیشتر برای کاهش زمان پردازش و پشتیبانی از تشخیص اشیای بزرگ‌تر و جزئیات بیشتر است. در مطالعات مقایسه‌ای صورت گرفته، YOLO V8توانسته بود دقت تشخیص اشیاء را به نسبت به نسخه‌های قبلی خود افزایش دهد. همچنین این شبکه عصبی قابلیت دنبال کردن عارضه موردنظر را در طول فیلم دارد که در این مقاله برای دنبال شدن یک چاله در طول فیلم از این مدل دنبال کننده[۱] YOLO استفاده شد. YOLO در این حالت به این صورت عمل می‌کند که در هر فریم به‌صورت انحصاری به شناسایی عارضه می‌پردازد و به آن یک شماره مختص به خود اختصاص می‌یابد و جعبه محاط کننده بر آن قرار می‌گیرد. در فریم‌های بعدی، دنبال کننده YOLO، اشیاء شناسایی‌شده را بر اساس معیارهای مختلف با موارد شناسایی‌شده قبلی مطابقت می‌دهد[۲۲]. معیارهایی که برای تطبیق اشیاء شناسایی‌شده در فریم‌ها در این سیستم دنبال کننده استفاده می‌شوند، شامل ارزیابی جنبه‌های متعدد اشیاء برای تعیین اینکه آیا آن‌ها با یک موجودیت مطابقت دارند یا خیر می‌باشد. از این معیارها می‌توان به مجاورت فضایی[۲] اشاره کرد که بر دو پارامتر همپوشانی جعبه مرزیIOU [۳]و فاصله مرکزی اشاره کرد. پارامتر IOU اندازه‌گیری می‌کند که جعبه‌های مرزی دو تشخیص چقدر باهم همپوشانی دارند. مقادیر بالا IOU نشان می‌دهد که تشخیص جدید همان شئ موجود در فریم قبلی می‌باشد. معیار بعدی می‌توان به فاصله مرکزی اشاره کرد که نشان‌دهنده‌ی فاصله اقلیدسی بین مرکز جعبه‌های مرزی در فریم‌های متوالی است و فاصله‌های کوچک‌تر نشان‌دهنده احتمال بالاتری از تشخیص همان شئ است. از دیگر معیارها می‌توان ویژگی‌های ظاهری که شامل شباهت بصری مانند رنگ، بافت و شکل عارضه موجود در جعبه محاط کننده و یا تطبیق هیستوگرام است را نام برد.

هرچند YOLO از معیارهای متفاوت دیگری نیز استفاده می‌کند که به بیان برخی از آن‌ها خواهیم پرداخت. یکی از این معیارها سازگاری زمانی و موقعیتی در حالت فریم به فریم است. به این صورت که عارضه‌هایی که در فریم‌های متوالی و نزدیک به موقعیت قبلی شناسایی می‌شوند، احتمال مطابقت را بیشتر می‌کنند و مواردی مانند پرش‌های ناگهانی در موقعیت جعبه محاط کننده یا تغییر اندازه آن، احتمال یکسان بودن تشخیص را کاهش می‌دهد[۲۳]. معیار دیگری که می‌توان به آن اشاره کرد، پیش‌بینی موقعیت عارضه در فریم بعدی می‌باشد. YOLO با استفاده از الگوریتمی مانند فیلتر کالمن با توجه به‌سرعت و موقعیت شئ موردنظر در فریم قبل، موقعیت شئ را در فریم بعدی پیش‌بینی می‌کند که منجر به کاهش وابستگی الگوریتم به پارامتر IOU می‌شود[۲۴]. قابل‌ذکر است YOLO در نظر می‌گیرد که اگر یک شئ به‌طور موقت ناپدید شود (مثلاً به دلیل انسداد)، ردیاب موقعیت آن را پیش‌بینی می‌کند و سعی می‌کند وقتی دوباره ظاهر شد همان شناسه را دوباره اختصاص دهد. همچنین دنبال کننده از معیارهای تطبیق قوی برای به حداقل رساندن الصاق شناسه‌های جدید، زمانی که اشیاء از مسیرهای متقابل یا همپوشانی یکدیگر عبور می‌کنند، استفاده می‌کند[۲۵]. به این صورت YOLO چاله‌ها را شناسایی و دنبال می‌کند. که نهایتاً در فایل حاصل‌شده، هر بیضی که بیانگر یک چاله است، دارای ID خاص خود و نیز مختصات تصویری دو سوی محور بزرگ و کوچک آن در فریم‌های مختلف می‌باشد. چون مختصات تصویری هر چاله در طول فیلم متغیر است، بنابراین IDها با توجه به اینکه چند بار در فریم‌های متفاوت دیده‌شده باشند، تکرار شده‌اند. برای رسیدن از این مختصات متفاوت تصویری به یک مختصات واحد برای هر چاله، از مثلث‌بندی استفاده گردید.

۳-۲- برازش بیضی بر چاله‌های شناسایی‌شده

جعبه محاط شده حاصل از YOLO، فضای بیشتری از چاله را پوشش می‌دهد، برای کاهش فضای اضافه و نزدیک‌تر شدن به شکل چاله، از شیوه برازش بیضی درون جعبه محاط کننده استفاده گردید. جهت برازش بیضی بر چاله‌های شناسایی‌شده در ابتدا مرکز مستطیل‌های محاط شده با

۱ Tracker

۲ Spatial Proximity

۳ Intersection over Union

استفاده از نصف طول و عرض آن محاسبه شد که به‌عنوان مرکز بیضی انتخاب گردید. همچنین قطر اطول بیضی برابر با طول مستطیل و قطر افضل آن نیز برابر با عرض مستطیل قرار گرفت.

۳-۳- تعیین موقعیت دوربین

۳-۳-۱- اعمال ORB SLAM 3 و استخراج موقعیت دوربین

برای محاسبه موقعیت سه‌بعدی چاله‌ها ابتدا نیاز است تا موقعیت و وضعیت دوربین در هرلحظه به دست آید. به این منظور فیلم گرفته‌شده از مسیر، به‌عنوان ورودی به الگوریتم ORB SLAM 3 داده می‌شود.

ORB SLAM 3 ادامه پروژه ORB-SLAM و یک SLAM بصری همه‌کاره برای کار با طیف گسترده‌ای از سنسورها (دوربین‌های تک‌چشمی، استریو، RGB-D) است. درمجموع SLAM بصری راهکاری است که با استفاده از آن، یک عامل (ربات) می‌تواند در محیطی ناشناخته و با موقعیتی نامعلوم، نقشه محیط اطراف را تهیه و موقعیت خود را در آن به دست آورد[۱۷]. ورودی این الگوریتم تصاویر اخذشده توسط دوربین می‌باشد و خروجی آن، فایلی است شامل انتقالات و چرخش دوربین در طول مسیر که می‌توان با این داده‌ها برای هر فریم ویدیو، ماتریس نگاشت[۱] ایجاد شود.

ماتریس‌های نگاشت، لازمه مثلث‌بندی می‌باشد. این ماتریس‌ها شامل پارامترهای داخلی و خارجی دوربین هستند. پارامترهای داخلی شامل فاصله کانونی و نقطه اصلی تصویر بوده که با کالیبراسیون دوربین به دست می‌آید و پارامترهای خارجی شامل موقعیت و چرخش دوربین است.

۳-۳-۲- ایجاد ماتریس نگاشت برای هر فریم

فایل خروجی ORB SLAM کمک می‌کند تا بتوان در هر فریم با استفاده از موقعیت نسبی دوربین (X،Y و Z) و مقادیر کواترنیون[۲] (q_x، q_y، q_z و q_w)، ماتریس نگاشت را تشکیل داد. پارامترهای کواترنیون جهت‌گیری دوربین را نشان می‌دهند و برای توصیف چرخش‌ها در فضای سه‌بعدی بدون وجود مشکلاتی مانند تکینگی‌ها[۳]، استفاده می‌شود.

برای تشکیل ماتریس نگاشت ابتدا ماتریس چرخش در هر فریم با استفاده از مقادیر کواترنیون تشکیل می‌شود که به شکل زیر می‌باشد:

$$R=\begin{bmatrix} 1-2(q_y^2+q_z^2) & 2(q_xq_y-q_zq_w) & 2(q_xq_z+q_yq_w) \\ 2(q_xq_y+q_zq_w) & 1-2(q_x^2+q_z^2) & 2(q_yq_z-q_xq_w) \\ 2(q_xq_z-q_yq_w) & 2(q_yq_z+q_xq_w) & 1-2(q_x^2+q_y^2) \end{bmatrix} \quad (۱)$$

سپس ماتریس انتقال در هر فریم با استفاده از موقعیت دوربین ایجاد می‌شود:

$$T=\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (۲)$$

برای تشکیل ماتریس خارجی[۴]، این دو ماتریس باید با یکدیگر ادغام شوند:

$$(R\,|\,\mathrm{T})=\begin{bmatrix} a_{11} & a_{12} & a_{13} & X \\ a_{21} & a_{22} & a_{23} & Y \\ a_{31} & a_{32} & a_{33} & Z \end{bmatrix} \quad (۳)$$

در مرحله بعدی این ماتریس در ماتریس داخلی[۵] ضرب می‌شود:

$$K=\begin{bmatrix} F_x & 0 & C_x \\ 0 & F_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \quad (۴)$$

که در آن F_x و F_y فاصله کانونی در راستای X و Y و C_x و C_y مختصات نقطه اصلی هستند.

درنهایت ماتریس نگاشت حاصل‌ضرب ماتریس داخلی در ماتریس خارجی می‌باشد:

$$P=\begin{bmatrix} F_x & 0 & C_x \\ 0 & F_y & C_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \end{bmatrix} \quad (۵)$$

۱ Projection Matrix
۲ Quaternion
۳ Singularities
۴ Extrinsic Matrix
۵ Intrinsic Matrix

$$P=\begin{bmatrix} F_xR_{11}+C_xR_{31} & F_xR_{12}+C_xR_{32} & F_xR_{13}+C_xR_{33} & F_xT_x+C_xT_z \\ F_yR_{21}+C_yR_{31} & F_yR_{22}+C_yR_{32} & F_yR_{23}+C_yR_{33} & F_yT_y+C_yT_z \\ R_{31} & R_{32} & R_{33} & T_z \end{bmatrix} \quad (۶)$$

این ماتریس برای هر فریم از دوربین به‌صورت جدا به همراه زمان اخذ آن فریم در واحد نانوثانیه ذخیره و تشکیل می‌گردد.

۳-۴- اعمال مثلث‌بندی و سرشکنی دسته اشعه

حال برای محاسبه مختصات سه‌بعدی نقاطی که در چندین فریم مشترک هستند، می‌توان از تکنیکی به نام مثلث‌بندی استفاده کرد.

مثلث‌بندی در فتوگرامتری یک تکنیک اساسی است که برای تعیین مختصات سه‌بعدی نقاط با تجزیه‌وتحلیل هندسه عکس‌های دارای همپوشانی استفاده می‌شود. این فرآیند شامل گرفتن تصاویر از زوایای مختلف و استفاده از اصول هندسه برای محاسبه موقعیت نقاط در فضا است[۲۶]. در این روش نقاط تصویر به فضای سه‌بعدی بازتاب داده می‌شود و بهترین نقطه در راستای خطوط اپی‌پولار که پرتوها در آن تلاقی می‌کنند (یا نزدیک‌تر می‌شوند) پیدا می‌شوند. درنهایت نیز سرشکنی دسته اشعه (Bundle Adjustment) باعث کاهش خطای مثلث‌بندی جهت تخمین نقاط سه‌بعدی می‌شود. درواقع Bundle Adjustment یک تکنیک برای بهبود ساختار سه‌بعدی و پارامترهای دوربین است که به‌طور گسترده در فتوگرامتری و بینایی ماشین استفاده می‌شود. اصطلاح "Bundle" به دسته‌ای از پرتوهای نوری اطلاق می‌شود که از یک صحنه سه‌بعدی از طریق لنزهای دوربین پخش می‌شود و "Adjustment " شامل بهینه‌سازی این Bundleها برای به حداقل رساندن خطاهای نگاشت مجدد[۱] است.

همان‌طور که بیان شد ماتریس P یک ماتریس ۴×۳ است که با توجه به تعداد فریم‌های دوربین از $P_1, P_2, ..., P_n$ وجود دارد. همچنین مختصات تصویری دوبعدی نقاط به فرم (u_i, v_i) در دسترس هستند که هدف یافتن بهترین تخمین مختصات سه‌بعدی (X, Y, Z) این نقاط در فضای مدل است. برای این کار در ابتدا از راه‌حل خطی کمترین مربعات[۲] استفاده می‌شود و معادلات برای هر فریم به‌صورت زیر نوشته می‌شوند:

$$\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = P_i \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (۷)$$

با گسترش این معادله، دو رابطه زیر به دست خواهد آمد:

$$u_i.(P_i[2].[X,Y,Z,1]^T) = P_i[0].[X,Y,Z,1]^T \quad (۸)$$

$$v_i.(P_i[2].[X,Y,Z,1]^T) = P_i[1].[X,Y,Z,1]^T \quad (۹)$$

که $P_i[0]$، $P_i[1]$ و $P_i[2]$ سطرهای ماتریس نگاشت P_i هستند.

قابل‌ذکر است که ماتریس نگاشت هر فریم و نقاط حاصل از YOLO با استفاده از واحد زمانی که با یکدیگر مشترک هستند و در واحد نانوثانیه می‌باشد با یکدیگر ادغام می‌شوند.

درنهایت با مرتب کردن قیود در فرم ماتریس، برای هر فریم می‌توان دو معادله را بر اساس بسطهای فوق استخراج کرد و برای n فریم، این منجر به سیستمی شامل $2n$ معادله که تشکیل‌دهنده‌ی سطرهای ماتریس هستند به شکل زیر می‌شود:

$$A=\begin{bmatrix} u_1.P_{31} & u_1.P_{32} & u_1.P_{33} & u_1.P_{34}-P_{11} \\ u_1.P_{31} & u_1.P_{32} & u_1.P_{33} & u_1.P_{34}-P_{12} \\ v_1.P_{31} & v_1.P_{32} & v_1.P_{33} & v_1.P_{34}-P_{21} \\ v_1.P_{31} & v_1.P_{32} & v_1.P_{33} & v_1.P_{34}-P_{22} \\ \vdots & \vdots & \vdots & \vdots \\ u_n.P_{31} & u_n.P_{32} & u_n.P_{33} & u_n.P_{34}-P_{n1} \\ v_n.P_{31} & v_n.P_{32} & v_n.P_{33} & v_n.P_{34}-P_{n2} \end{bmatrix} \quad (۱۰)$$

حال برای حل، این ماتریس باید به فرم $A.X_{\hom}=0$ تبدیل شود که A، ماتریس ۴ × $2n$ حاصل ضرایب معادله است و $X_{\hom}$ برابر با $[X,Y,Z,1]^T$ می‌باشد که بردار

۱ Reprojection Error

۲ Linear Least Squares Solution

مختصات همگن[1] نقطه سه‌بعدی است. حال با استفاده از روش SVD[2] می‌توان به حل این دست پیدا کرد. SVD روشی برای حل سیستم‌های خطی است، به‌ویژه آن‌هایی که دارای قیود هستند.

معادله $A.X_{\text{hom}} = 0$ با محاسبه SVD مقدار A و گرفتن آخرین ردیف V (بردارهای منفرد سمت راست) حل می‌شود. بردار X_{hom} که مقدار $\| A.X_{\text{hom}} \|$ را به حداقل می‌رساند، با مختصات نقطه سه‌بعدی به شکل همگن مطابقت دارد. به‌طور خاص، بردار مفرد سمت راست برابر با کوچک‌ترین مقدار مفرد A است. نتیجه این مرحله مختصات همگن به شکل $X_{\text{hom}} = [X, Y, Z, W]^T$ است بنابراین مختصات سه‌بعدی واقعی به شکل X، Y و Z از تقسیم‌بر مقدار W به دست می‌آیند:

$$X = \frac{X}{W} \quad Y = \frac{Y}{W} \quad Z = \frac{Z}{W} \qquad (۱۱)$$

به این صورت، با انجام مثلث‌بندی، در انتها مختصات سه‌بعدی نقاط چاله‌ها، به دست می‌آید و هر ID دارای چهار نقطه که متعلق به چهارسوی بیضی برازش داده‌شده می‌باشند، است. مختصات به‌دست‌آمده شده در محیط مدل محاسبه‌شده‌اند و حاصل دیده شدن یک چاله در فریم‌های متوالی می‌باشند. به این صورت که هر چه تعداد مشاهدات بیشتر باشد، دقت تبدیل نیز افزایش خواهد یافت.

۳-۵- مقایسه مساحت نرمال‌شده چاله‌های به‌دست‌آمده با مقادیر واقعی و محاسبه میزان خطای نگاشت مجدد هریک

در انتهای کار، برای بررسی دقت تبدیل مختصات و نیز تأثیر تعداد نقاط بر دقت تبدیل، پارامترهایی همچون تعداد دفعات دیده شدن هر ID، مساحت نسبی هر چاله و خطای نگاشت مجدد[3] استخراج گردیدند.

با توجه به اینکه دو سوی محور بزرگ و کوچک بیضی تبدیل می‌شوند بنابراین با استفاده از فرمول محاسبه بیضی می‌توان مساحت هریک را به دست آورد.

$$\text{مساحت بیضی} = \pi \times a \times b \qquad (۱۲)$$

که در آن π معرف عدد ۳/۱۴، a نصف محور بزرگ و b نصف محور کوچک می‌باشد.

چون مساحت‌های به‌دست‌آمده به‌صورت نسبی می‌باشند، با تقسیم تمام مقادیر محاسبه‌شده بر بزرگ‌ترین مساحت، اعداد، نرمال‌شده و تمامی مساحت‌ها بین صفر و یک قرار می‌گیرند. برای مقایسه با مساحت‌های واقعی، مقادیر آن‌ها نیز باید نرمال شوند. در آخر اختلاف بین مساحت واقعی و نسبی هر چاله به درصد در دسترس قرار می‌گیرد.

پارامتر بعدی که محاسبه گردید میانگین خطای نگاشت است. همانطورکه بیان شد، ماتریس نگاشت که نقاط سه‌بعدی در فضا را به نقاط تصویر دوبعدی نگاشت می‌کند به شکل $P = K.(R \mid T)$ می‌باشد یعنی مانند ضریبی عمل می‌کند که مختصات تصویری دوبعدی را به سه‌بعدی تبدیل می‌کند. برای محاسبه خطای نگاشت مجدد نیز از این ماتریس استفاده می‌شود تا از نقاط سه‌بعدی محاسبه‌شده به نقاط دوبعدی تصویری رسید. حال اگر X را مختصات سه‌بعدی محاسبه‌شده در نظر بگیریم، مختصات دوبعدی برابر با $x' = P.X$ می‌باشد که در آن x نقطه دوبعدی پیش‌بینی‌شده در مختصات همگن است:

$$x' = [u', v', w']^T \qquad (۱۳)$$

و با تقسیم‌بر x به مختصات دکارتی تبدیل می‌شود:

$$(u, v) = \left(\frac{u'}{w'}, \frac{v'}{w'}\right) \qquad (۱۴)$$

خطای نگاشت مجدد نشان می‌دهد که نقطه دوبعدی دوباره محاسبه‌شده حاصل از نقاط سه‌بعدی (u,v)، چقدر به نقطه دوبعدی مشاهده‌شده اصلی (x, y) نزدیک است:

$$e = \sqrt{(u - x)^2 + (v - y)^2} \qquad (۱۵)$$

این فاصله اقلیدسی بین نقطه دوبعدی مشاهده‌شده (x,y) و نقطه دوبعدی محاسبه‌شده (u,v) است که در واحد

۱ Homogeneous coordinate vector

۲ Singular Value Decomposition

۳ Reprojection Error

پیکسل می‌باشد. هنگامی‌که چندین فریم تصویر وجود دارد که نقطه سه‌بعدی یکسانی را مشاهده می‌کنند، خطای نگاشت مجدد برای هر نما محاسبه می‌شود و سپس خطای میانگین به دست می‌آید:

$$\text{میانگین خطای نگاشت مجدد} = \frac{1}{N}\sqrt{e_i} \qquad (۱۶)$$

که در آن N تعداد فریم‌های دوربین است.

۴- پیاده‌سازی

همانطورکه در بخش‌های قبلی این مقاله نیز مطرح گردید، این کار دارای دو بخش است که در هر بخش دارای زیرشاخه‌هایی می‌باشند، اما ورودی هردو بخش، فیلمی است که از چاله‌های موجود در مسیری مشخص گرفته‌شده است.

شکل ۳- نمایی از منطقه فیلم‌برداری شده

مطابق با مطالب ذکرشده، برای برداشت داده اولیه موردنیاز الگوریتم، از نرم‌افزار VIRec نصب‌شده بر روی گوشی هوشمند Samsung S7 Edge استفاده شد. این نرم‌افزار به فیلم‌برداری و نیز ذخیره زمان اخذ هر داده می‌پردازد که برای تعیین موقعیت دوربین توسط الگوریتم ORB SLAM 3 مورداستفاده قرار می‌گیرد. به‌این‌ترتیب با استفاده از نرم‌افزار مذکور، از مسیر مشخصی که دارای چاله بود فیلم‌برداری صورت گرفت.

اطلاعات دیگری که موردنیاز الگوریتم می‌باشد فایل کالیبراسیون دوربین مورداستفاده است. این فایل حاوی مقادیر فاصله کانونی در راستای X و Y (F_x و F_y) و مختصات نقطه اصلی در راستای X و Y (C_x و C_y) که به پارامترهای درونی شناخته می‌شوند و نیز ضرایب اعوجاج دوربین (K_2 ، K_1، P_1 و P_2) است. برای رسیدن به این مقادیر، از پکیج Camera Calibration که مربوط به ROS [۱] است، کمک گرفته شد. این پکیج بر روی Ubuntu نسخه ۱۸/۰۴ نصب گردید که با استفاده از یک صفحه شطرنجی با حرکت دوربین در جهات مختلف به محاسبه پارامترهای کالیبراسیون دوربین می‌پردازد.

همچنین برای در دست داشتن شکل چاله‌ها و موقعیتشان، با استفاده از گیرنده مولتی فرکانس G1 Plus-Sout با دقت ۰/۰۱۲ متر به برداشت نقاط لبه چاله‌ها پرداخته شد.

۴-۱- شبکه عصبی YOLO V8m

برای استفاده از شبکه‌های عصبی، در مرحله اول باید به آموزش آن‌ها پرداخت. در این مقاله نیز ابتدا به آموزش YOLO پرداخته شد و سپس مورداستفاده قرار گرفت.

۴-۱-۱- آموزش الگوریتم YOLO V8m

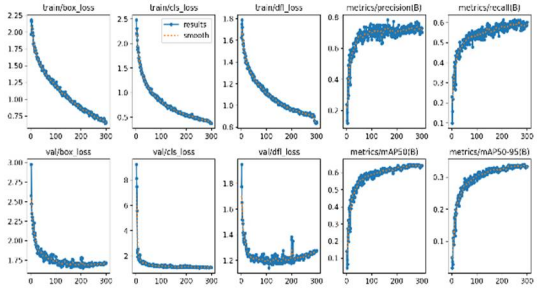
برای آموزش شبکه‌های عصبی نیاز است تا مجموعه تصاویری تحت عنوان داده آموزشی آماده شوند. در این تصاویر شئ یا اشیاء موردنظر توسط جعبه محاط کننده مشخص می‌شوند و مختصات تصویری هر یک از اشیاء به همراه برچسب و شماره تصویر موردنظر در یک فایل با فرمت ".txt" ذخیره می‌گردند. تصاویر جهت آموزش، به شبکه داده می‌شود و شبکه عصبی به استخراج پارامترهایی جهت تشابه سنجی بین اشیاء در تصاویر مختلف می‌پردازد تا درنهایت به یک وزن مشخص برای نورون‌های موجود در شبکه برسد. درنهایت برای بررسی عملکرد شبکه آموزش‌دیده مجموعه تصاویری با نام داده تست به آن داده می‌شود که در آن شئ موردنظر مشخص نشده است و شبکه باید به وجود یا عدم وجود آن در تصاویر پی ببرد. درنهایت با توجه به تشخیص شبکه، پارامترهایی همچون Percision، Recall، mAP ۵۰ و mAP ۵۰-۹۰ وجود دارد. Percision میزان صحت پیش‌بینی‌های مثبت انجام‌شده توسط شبکه را نشان می‌دهد، Recall توانایی مدل برای یافتن تمام موارد مثبت در مجموعه

۱ Robot Operation System

داده را اندازه‌گیری می‌کند، mAP ۵۰ تعداد تشخیص‌های درست با میانگین خطای ۵۰ درصد را نمایش می‌دهد و mAP ۵۰-۹۰ نیز بیان‌کننده تعداد تشخیص‌های درست با میانگین خطای بین ۵۰ تا ۹۰ درصد است. بدیعی است که هرچه حجم داده آموزشی بیشتر و در حالت‌ها و زوایای متفاوت‌تری از شئ موردنظر باشد، شبکه نیز عملکرد بهتری از خود را نشان خواهد داد. در این کار داده آموزشی از سایت روبوفلو[۱] [۲۷] که تهیه‌کننده داده برای عوارض متفاوت است تهیه گردید و پس از بهبود مجموعه داده، شامل حذف داده‌های نامناسب و نیز جایگذاری دقیق جعبه‌های محاط کننده بر روی چاله‌ها، به‌عنوان داده آموزشی به شبکه عصبی داده شد.

شکل ۴- نمونه داده استفاده‌شده برای آموزش شبکه عصبی YOLO V8

همان‌طور که بیان شد در این مقاله برای تشخیص چاله‌ها، با توجه به تحقیقات صورت گرفته، از شبکه عصبی YOLO- V8 استفاده گردید. نسخه V8 از سری YOLO مانند نسخه‌های قبلی، خود دارای چندین مدل است: YOLO V8n، YOLO V8l، YOLO V8x، YOLO V8m و YOLO V8s[۱۳]. هریک از این مدل‌ها دارای تفاوت در ویژگی‌هایی مانند سرعت و تعداد پارامترهایشان هستند که برخی از این شبکه‌ها مانند YOLO V8n و YOLO V8s آموزش دیدند و بررسی شدند. درنهایت مدل YOLO V8m با ۱۲۶۵ عکس جهت آموزش شبکه، ۱۱۸ تصویر برای ارزیابی شبکه ، ۴۰۱ تصویر به‌منظور اعتبارسنجی[۲]، ۳۰۰ تکرار[۳] در ابعاد تصاویر ۶۴۰ و اندازه دسته[۴] ۸، ازلحاظ صحت، دقت و سرعت بهترین نتیجه را به خود اختصاص داد.



شکل ۵- نتایج حاصل از آموزش شبکه YOLO v8m

جدول ۲- نتایج به‌دست‌آمده از آموزش شبکه YOLO

نتایج نهایی YOLO	پارامترها
٪ ۹۲	Precision
٪ ۶۸	Recall
٪ ۸۲	۵۰mAP
٪ ۶۶	۵۰-۹۰ mAP

۴-۱-۲- اعمال الگوریتم YOLO V8

پس از آموزش شبکه، فیلم ورودی به آن داده تا شبکه عصبی با توجه به سازوکار بیان‌شده در بخش‌های قبل به تشخیص چاله‌ها بپردازد. یکی دیگر از ویژگی‌های شبکه عصبی YOLO دنبال کردن شئ موردنظر در طول فیلم است. با توجه به این‌که موقعیت عارضه موردنظر به دلیل حرکت دوربین، در طول فیلم متغیر است، YOLO با استفاده از قابلیت دنبال کردن خود، تشخیص می‌دهد که عارضه موجود در فریم جدید همان عارضه قبلی است که موقعیت آن تغییر کرده است. این قابلیت کمک می‌کند تا یک چاله در فریم‌های متفاوت، از زمان ظاهر شدن تا خروجش در فیلم دنبال شود. خروجی این مرحله فایلی است شامل شماره هر چاله که با ID نمایش داده می‌شود، زمان شناسایی چاله که با سربرگ Timestamp در واحد نانوثانیه مشخص می‌گردد و نیز مختصات تصویری گوشه سمت چپ بالای جعبه محاط کننده. در مرحله بعدی برای دنبال کردن چاله‌ها در فریم‌های مختلف و نیز نزدیک‌تر شدن به شکل آن، روش برازش بیضی انجام گرفت.

۱ robofllow

۲ Validation

۳ Epoch

۴ Batch Size

۴-۱-۳- برازش بیضی درون جعبه محاط کننده

برای نزدیک شدن به شکل چاله‌ها، به برازش بیضی در محیط پایتون و استفاده از پکیج Open CV[1] نسخه ۴.۱۰ درون مستطیل حاصل از YOLO پرداخته شد. در این روش، به کمک پکیج Open CV، بیضی درون مستطیل حاصل از YOLOبرازش داده شد.Open CV یک کتابخانه نرم‌افزاری بینایی کامپیوتر و یادگیری ماشین منبع باز است که طیف گسترده‌ای از ابزارها و عملکردها را برای برنامه‌های کاربردی در این زمینه به‌صورت در لحظه فراهم می‌کند. اگرچه با برازش بیضی نیز همچنان فضای اضافی نسبت به چاله واقعی دارد، اما فضای پِرت کمتری نسبت به مستطیل ارائه می‌دهد.

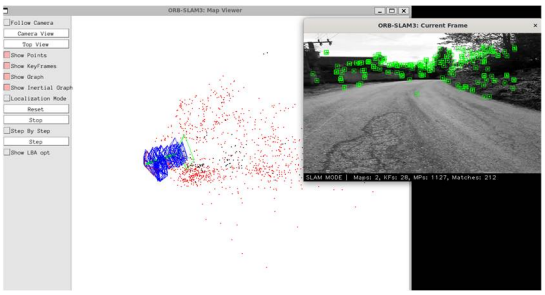
الف ب

شکل ۶- مقایسه خروجی الگوریتم همراه با واقعیت زمینی برداشت‌شده. الف) یکی از چاله‌های مشاهده‌شده. ب) همان چاله پس از تشخیص توسط شبکه عصبی YOLO با مستطیل قرمز، محاط بیضی بر آن با رنگ سبز و شکل واقعی چاله برداشت‌شده با گیرنده مولتی فرکانس با رنگ سفید.

حال بیضی‌های برازش داده‌شده بر روی چاله‌ها در فریم‌های مختلف دنبال می‌شوند و چهار مختصات دوبعدی که مربوط به دو سوی محورهای بزرگ و کوچک آن‌ها هستند، ذخیره می‌شوند اما چون با حرکت دوربین موقعیت تصویری چاله‌ها و به دنبال آن موقعیت تصویری محورها در هرلحظه تغییر می‌کند، در فایل خروجی، شماره IDهای تکراری با مختصات‌های گوناگون موجود می‌باشد.

۴-۲- اجرای الگوریتم ORB SLAM 3

این الگوریتم که یکی از الگوریتم‌های ناوبری رباتیک می‌باشد، با استفاده از تطابق دادن نقاط مشابه در طول مسیر فیلم‌برداری به تعیین موقعیت دوربین در هرلحظه می‌پردازد. در دوربین‌های دوچشمی، این تطابق بین تصاویر برداشتی بین دو چشم دوربین انجام می‌شود اما در این مقاله چون از دوربین گوشی استفاده‌شده است و اصطلاحاً تک‌چشمی است، این تطابق بین فریم‌های متوالی انجام می‌گیرد و با توجه به تغییر موقعیت این نقاط کلیدی در فریم جدید نسبت به فریم قبل، میزان تغییر مکان و چرخش دوربین مشخص می‌شود. این الگوریتم در Ubuntu ورژن ۲۰/۰۴ مورداستفاده قرار گرفت.



شکل ۷- اجرای ORB SLAM 3 بر روی Ubuntu ورژن ۲۰/۰۴

برای رسیدن به‌دقت خوب و عملکرد درست الگوریتم، باید مقادیر کالیبراسیون دوربین محاسبه شود و جایگذاری گردند. برای این کار از پکیج Camera Calibration که مربوط به ROS است، کمک گرفته شد. این پکیج بر روی Ubuntu ورژن ۱۸/۰۴ نصب گردید که با استفاده از یک صفحه شطرنجی با حرکت دوربین در جهات متفاوت به محاسبه پارامترهای کالیبراسیون دوربین می‌پردازد.

خروجی این مرحله شامل Timestamp در واحد نانوثانیه، موقعیت X، Y و Z به‌صورت نسبی و دوران به‌صورت کواترنیون (ربعی) qx، qy، qz و qw است. این موقعیت‌ها و دوران‌ها در سیستم مختصات محلی‌اند که داده‌های موردنیاز برای مرحله بعد هستند.

۴-۳- اجرای روش مثلث‌بندی

حال مجموعه نقاط چاله‌ها در سیستم مختصات تصویری دوبعدی با استفاده از YOLO به دست آمدند و موقعیت و وضعیت دوربین نیز در هرلحظه توسط ORB SLAM 3 در دسترس قرار گرفت. هر گروه نقطه، دارای یک ID بود که بیانگر شماره چاله است و چون یک چاله در فریم‌های مختلف دیده و دنبال شد، متناسب با تعداد دفعاتی که در فریم‌های مختلف دیده‌شده باشد تکرار گردیده بود. هدف از این قسمت، رسیدن به یک مختصات واحد و سه‌بعدی برای هر شماره

۱ Open Source Computer Vision Library

چاله با استفاده از چندین مختصات دوبعدی بود که با استفاده از مبانی فتوگرامتری و روش مثلث‌بندی صورت گرفت.

در این مرحله، با استفاده از مقادیری که در هرلحظه از وضعیت دوربین توسط ORB SLAM به دست آمد به تشکیل ماتریس چرخش و انتقال پرداخته شد. به این صورت که با مقادیر کواترنیون، ماتریس چرخش و با مقادیر X، Y و Z ماتریس انتقال تشکیل گرفت. این دو ماتریس برای تشکیل ماتریس بیرونی با یکدیگر ادغام شدند که حاصل، یک ماتریس ۴×۳ بود. همچنین با استفاده از مقادیر کالیبراسیون شامل فاصله کانونی در راستای X و Y و نیز مختصات نقطه اصلی، ماتریس داخلی K تشکیل شد. درنهایت ماتریس داخلی و بیرونی در یکدیگر ضرب شدند و ماتریس نگاشت به دست آمد.

در ادامه برای محاسبه مختصات سه‌بعدی برای هر ID، از روش مثلث‌بندی همراه با سرشکنی دسته اشعه برای به حداقل رساندن میزان خطای نگاشت مجدد استفاده گردید. به این صورت که با توجه به تعداد دفعاتی که یک ID در فریم‌های مختلف دیده‌شده و ماتریس‌های نگاشت مربوط به آن، ماتریس A به دست می‌آید. این ماتریس به تعداد دو برابر فریم‌های دیده شدن یک چاله دارای سطر است و نیز تعداد ستون‌های آن ۴ می‌باشد. برای حل، این ماتریس به فرم $A.X_{\hom}=0$ تبدیل شد که درنهایت با حل به روش SVD به مختصات همگن رسیدیم که برابر با $[X, Y, Z, W]^T$ است. درنهایت نیز با تقسیم X، Y و Z بر مؤلفه آخر ینی W، مختصات سه‌بعدی محاسبه شدند.

۵- نتایج و ارزیابی

در این مقاله، برای بررسی عملکرد روش پیشنهادی، فیلم‌برداری از یک مسیر مشخص شامل چاله‌ها، با استفاده از نرم‌افزار VIRerc و گوشی Samsung مدل S7 Edge صورت گرفت. همچنین برای برداشت نقاط لبه چاله‌ها با مختصات UTM، از گیرنده مولتی فرکانس G1 Plus Sout با دقت ۰/۰۱۲ متر استفاده شد تا مقادیر واقعی در دسترس قرار گیرند و به‌عنوان واقعیت زمینی، معیار مقایسه با نتایج به‌دست‌آمده باشند.

ازآنجایی‌که شروع کار روش پیشنهادی ذکرشده وابسته به تشخیص YOLO در تشخیص چاله‌ها می‌باشد، این مرحله نقش کلیدی دارد. درواقع الگوریتم به محاسبه مختصات شئی که در این مرحله شناسایی‌شده می‌پردازد، حال ممکن است شبکه نتواند یک چاله را تشخیص دهد یا اینکه تشخیص اشتباه بر چاله بودن بدهد.

در فیلمی که به‌عنوان ورودی به الگوریتم YOLO V8m داده شد، ۷ چاله وجود دارد. شبکه عصبی YOLO V8m در فیلم ورودی ۲۵۹ شئ را تحت عنوان چاله‌های گوناگون شناسایی کرد. طبق بررسی‌های انجام‌شده در نتایج به‌دست‌آمده بر روی منطقه تست، تمامی چاله‌ها شناسایی شدند اما تعداد IDهای الصاق شده بسیار بیشتر از تعداد واقعی چاله‌ها می‌باشد که دلیل عمده آن شکست شبکه در دنبال کردن چاله‌های یکسان بوده است. به این صورت که الگوریتم در نمای اول چاله را در فاصله دور شناسایی می‌کند اما با نزدیک شدن دوربین به همان چاله، آن را به‌عنوان چاله جدید شناسایی کرده و شماره جدید به آن اختصاص می‌دهد و این چرخه چندین بار تا رسیدن و عبور کردن از همان چاله اتفاق می‌افتد. دلیل دیگری که مربوط به بالا بودن تعداد شماره‌ها تحت عنوان چاله می‌باشد، عملکرد نادرست شبکه در تشخیص می‌باشد به این صورت که در مواردی، عوارضی را به‌اشتباه چاله تشخیص داده و به آن شماره اختصاص داده است. خطاهای ذکرشده عموماً در تعداد تکرار کم فریم‌ها اتفاق افتاده‌اند به همین دلیل برای کم کردن این قبیل خطاها، حداقل تکرار ۵ فریم به‌عنوان حد آستانه انتخاب گردید. هرچند عدد انتخاب‌شده می‌توانست بزرگ‌تر نیز باشد اما به‌منظور مقایسه، شماره‌هایی با بیشتر از ۵ تکرار فریم باقی ماندند.

در ادامه صرف‌نظر از اینکه شبکه عصبی به‌درستی چاله‌های موردنظر را تشخیص داده است یا نه، به تخمین موقعیت سه‌بعدی تمامی ۲۳ مورد ذکرشده با استفاده از روش مثلث‌بندی همراه با سرشکنی دسته اشعه پرداخته شد. برای بررسی دقت نقاط تبدیل‌شده، تعداد تکرار هریک به همراه میانگین خطای نگاشت مجددشان محاسبه گردید. به این صورت که در ابتدا از نقاط دوبعدی با استفاده از ماتریس نگاشت و روش مثلث‌بندی همراه با سرشکنی دسته اشعه، موقعیت سه‌بعدی هر نقطه تخمین زده شد. در مرحله بعد برای بررسی دقت تبدیل ، این نقاط که حالا با مختصات سه‌بعدی بوده‌اند با استفاده مجدد از ماتریس نگاشت به مختصات دوبعدی تصویری تبدیل شدند. درنهایت با محاسبه فاصله اقلیدسی بین نقاط تصویری اصلی و محاسبه‌شده به واحد پیکسل، دقت تبدیل نقاط به دست آمد.

جدول ۳- نتایج به‌دست‌آمده حاصل تبدیل نقاط شناسایی‌شده توسط YOLO به همراه تعداد تکرار و خطای نگاشت مجدد هر گروه از نقاط

شماره ID	تعداد تکرار	میانگین خطای نگاشت مجدد (Pix)
۱	۱۳	۲/۹۶
۱۸	۱۹	۲/۶۱
۲۰	۱۷	۲/۷۳
۳۰	۱۰	۳/۳۵
۷۰	۷	۳/۵۷
۸۰	۱۶	۲/۵۹
۱۰۰	۱۰	۳/۳۳
۱۳۲	۱۵	۲/۴۶
۱۴۵	۷	۳/۷۸
۱۵۳	۴۷	۱/۶۲
۱۶۹	۳۰	۱/۷۱
۱۹۱	۳۹	۱/۷۳
۱۹۷	۱۰	۴/۶۲
۲۰۹	۸	۳/۴۹
۲۱۴	۴۱	۱/۷۶
۲۲۲	۳۴	۱/۷۴
۲۲۶	۵۲	۰/۹۹
۲۴۰	۷۷	۰/۹۷
۲۴۴	۱۷	۲/۶۴
۲۴۹	۱۴	۳/۲۷
۲۵۴	۶	۳/۸۲
۲۵۸	۵۲	۱/۶۴
۲۵۹	۸	۳/۵۲

همان‌طور در جدول ۳ بیان‌شده است، شبکه عصبی YOLO، ۲۳ شئ را به‌عنوان چاله شناسایی کرده است درصورتی‌که تنها ۷ چاله در فیلم وجود دارد که در جدول نیز با رنگ خاکستری مشخص‌شده‌اند. در این جدول ستون شماره یک، شماره گروه نقاطی است که به‌عنوان چاله شناسایی شدند، ستون دوم تعداد دفعات دیده شدن چاله در فریم‌های مختلف است و ستون سوم میانگین خطای نگاشت مجدد هستند. با توجه به میزان خطاهای به‌دست‌آمده حاصل از تبدیل نقاط در جدول فوق می‌توان به‌دقت خوب و قابل‌قبول مثلث‌بندی در تخمین نقاط سه‌بعدی از دوبعدی پی برد. مطلب دیگری که از این جدول قابل‌تأمل است، تشخیص‌های متفرقه‌ی شبکه است که مدنظر نمی‌باشند. با توجه به نتایج به‌دست‌آمده این عوارض عموماً در تعداد فریم‌های کمی دیده‌شده‌اند. به بیانی دیگر احتمال اینکه تعداد عارضه‌هایی که به تعداد بالا دیده‌شده‌اند، عوارض موردنظر باشند بیشتر است به همین جهت با استفاده از این پارامتر می‌توان میزان تشخیص‌های متفرقه‌ی شبکه را کمتر کرد و به حداقل رساند. برای این کار می‌توان با اعمال حد آستانه و مشخص کردن حداقل تعداد تکرار فریم برای هر نقطه به این هدف رسید اما اعمال یک عدد به‌عنوان حداقل تکرار برای حذف موارد اشتباه تنها به‌صورت تجربی امکان‌پذیر است و نیز برای هر داده‌ای، این عدد متفاوت خواهد بود. برای بهتر انجام دادن این فیلترینگ، از پارامتری که در اینجا برای ارزیابی دقت تبدیل مورداستفاده قرارگرفته شد جهت دسترسی سیستماتیک به عدد موردنظر نیز استفاده گردید، زیرا مقادیر خطای نگاشت مجدد، رابطه غیرمستقیم با تعداد دفعات دیده شدن چاله‌ها در فریم‌های مختلف دارند. به‌طور غالب، با افزایش تعداد مشاهدات، روند این خطا نزولی می‌باشد. این نزول در نقاط با تعداد تکرار پایین‌تر بیشتر به چشم می‌خورد و از یک حدی به بعد، افزایش مشاهدات، تغییرات چشمگیری بر نتایج نخواهد داشت.

با استفاده از خطاهای نگاشت می‌توان تعداد نتایج به‌دست‌آمده را فیلتر کرد. با این کار هم تعداد شناسایی‌هایی که مدنظر نیستند کاهش می‌یابد و نیز نقاط باقی‌مانده دارای دقت بالاتری از تبدیل هستند. در صنایع و با استفاده از تجهیزات دقیق، میزان خطای نگاشت مدنظر برای کارهایی بااهمیت بالا ،زیر یک پیکسل[1] می‌باشد[۲۸]. اما در اکثریت کارهای پردازش تصاویر میزان خطای تا ۲ پیکسل نیز کاربرد دارند[۲۹]. هرچند میزان مجاز این خطا متناسب به کار موردنظر می‌تواند متغیر باشد که در جدول زیر به‌طورکلی بیان‌شده است.

جدول ۴- دسته‌بندی خطاهای نگاشت مجدد

وضعیت	میزان خطا
عالی	کمتر از ۱ پیکسل
مناسب برای بیشتر کارهای بینایی ماشین	بین ۱ تا ۲ پیکسل
قابل‌قبول متناسب با برخی کاربردها	بین ۲ تا ۵ پیکسل
دارای مشکل	بیشتر از ۵ پیکسل

طبق جدول ۴، تمامی نقاط تبدیل‌شده دارای دقت قابل قبولی می‌باشند اما جهت اعمال فیلترینگ، در اینجا با توجه به این‌که میانگین خطای نگاشت مجدد بین یک تا دو پیکسل مناسب کارهای مرتبط با بینایی ماشین می‌باشند؛ به همین

۱ SubPixel

منظور حداکثر ۲ پیکسل به‌عنوان حد آستانه برای میزان خطای نگاشت مجدد اعمال شد.

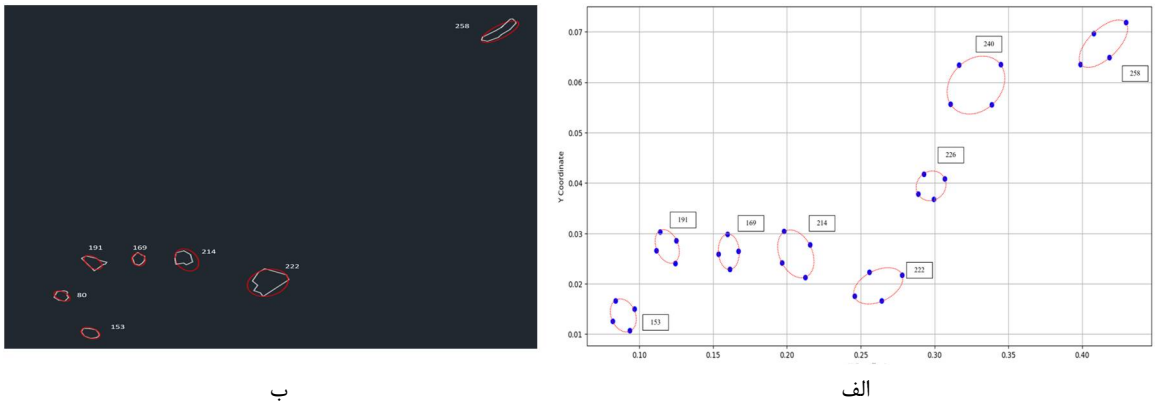
جدول ۵ - چاله‌های باقیمانده پس از اعمال فیلتر حداکثر خطای مجاز نگاشت مجدد

شماره ID	تعداد تکرار	میانگین خطای نگاشت مجدد (Pix)
۱۵۳	۴۷	۱/۶۲
۱۶۹	۳۰	۱/۷۱
۱۹۱	۳۹	۱/۷۳
۲۱۴	۴۱	۱/۷۶
۲۲۲	۳۴	۱/۷۴
۲۲۶	۵۲	۰/۹۹
۲۴۰	۷۷	۰/۹۷
۲۵۸	۵۲	۱/۶۴

قابل‌ذکر است این حد آستانه می‌تواند برای امور دیگر متفاوت باشد، برای مثال اگر هدف صرفاً تعیین موقعیت چاله در مسیر باشد، می‌توان از نقاطی با میانگین خطای بالاتری نیز استفاده کرد.

در جدول ۵ نقاطی که با دقت بالایی تبدیل‌شده‌اند قرار دارند. با اعمال این حد آستانه، تعداد ID های شناسایی‌شده از ۲۳ عدد به ۸ عدد کاهش‌یافته‌اند. این IDها علاوه بر تعداد تکرار بالا، میانگین خطای نگاشت مجدد پایینی نیز دارند. با این روش به‌صورت علمی حد آستانه حداقل تعداد تکرار برای هر ID هم به دست می‌آید.

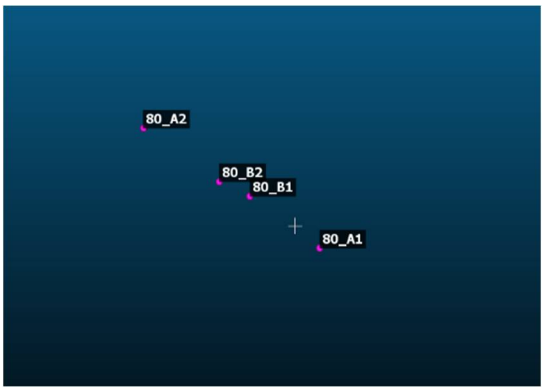
برای بررسی بصری شکل چاله‌های محاسبه‌شده، پس از اعمال حد آستانه بر روی خطای نگاشت مجدد، نقاط در محیط پایتون رسم شدند و بر هریک از گروه نقاط بیضی برازش داده‌شده و همچنین نقاط برداشت‌شده توسط گیرنده مولتی فرکانس که به‌عنوان مرجع و واقعیت زمینی می‌باشند در محیط AutoCAD ترسیم شدند و برای دید بهتر، بر هریک از آن‌ها نیز بیضی برازش داده شد.



ب الف

شکل ۸- مقایسه بصری چاله‌ها: الف) نمایش بیضی برازش داده‌شده بر روی نقاط تبدیل‌شده چاله‌ها ب) نمایش چاله‌های برداشت‌شده با استفاده از گیرنده مولتی فرکانس که بیضی حاصل از YOLO بر آن‌ها برازش داده شد در محیط AutoCAD.

همان‌طور که از شکل ۸ برمی‌آید، علی‌رغم شباهت بصری چاله‌های متناظر با یکدیگر، تفاوت‌هایی بین تصویر الف و ب وجود دارد. تفاوت اول عدم وجود چاله با ID شماره ۸۰ در نقاط تبدیل‌شده می‌باشد. دلیل این عدم وجود، حد آستانه گذاری انجام بر روی میزان خطای نگاشت مجدد می‌باشد. حذف شدن این گروه از نقطه به دلیل تعداد مشاهدات کم و بالطبع دقت پایین در تبدیل بوده است و در صورت عدم حذف، با توجه به تبدیل نادرست نقاط، نتایج اشتباهی را نیز حاصل می‌کرد.



شکل ۹- نمایش چاله با ID شماره ۸۰ در مختصات تبدیل‌شده در نرم‌افزار Cloud Compare

با توجه به شکل ۹، نقاط چهارگوشه بیضی برازش داده‌شده بر چاله، به دلیل تعداد مشاهدات کم و بیشتر بودن میزان خطای نگاشت مجددشان از حد آستانه، تبدیل آن‌ها به‌صورت اشتباه صورت گرفته و این چهار نقطه بجای تشکیل یک بیضی، در امتداد یکدیگر قرار گرفتند؛ بنابراین با استفاده از حد آستانه، این ID حذف گردید. همان‌طور که بیان شد، این میزان خطا برای بسیاری از کار ها می‌تواند مناسب باشد اما با توجه به هدف این کار تحقیقاتی، باعث ایجاد خطا در نتایج می‌شود.

تفاوت دیگری که بین تصاویر شکل ۸ وجود دارد، اضافه شدن دو چاله با شماره ۲۲۶ و ۲۴۰ است. این دو عارضه که با اعمال فیلتر نیز از میان تشخیص‌های متفرقه شبکه YOLO حذف نشدند، مربوط به خرابی سطح آسفالت بودند که ازنظر ظاهری شباهت زیادی با چاله داشتند و در شکل ۹ نمایش داده شدند.

شکل ۱۰- ID های به‌اشتباه باقی‌مانده به‌عنوان چاله بعد از اعمال حد آستانه در پارامتر نگاشت مجدد

درنهایت نیز به‌منظور ارزیابی دقت تبدیل نقاط چاله‌ها، مقایسه‌ای بین مساحت چاله‌های محاسبه‌شده و مقدار واقعی آن‌ها صورت گرفت. مساحت‌های به‌دست‌آمده توسط الگوریتم در محیط مدل بوده، به همین منظور جهت مقایسه با مساحت‌های واقعی، برای مقادیر هر دو گروه از مساحت‌ها، نرمال‌سازی انجام گرفت.

جدول ۶ - اختلاف بین مساحت‌های نرمال‌شده واقعی و محاسبه‌شده

شماره ID	مقادیر نرمال شده مساحت‌های واقعی	مقادیر نرمال شده مساحت‌های به‌دست‌آمده	درصد اختلاف بین مقادیر نرمال شده واقعی و به‌دست‌آمده
۱۵۳	۰/۵۲	۰/۵۰	۲
۱۶۹	۰/۴۶۳	۰/۴۲۴	۳/۹
۱۹۱	۰/۴۵۹	۰/۴۲۱	۳/۸
۲۱۴	۰/۸۷	۰/۹۱	-۴
۲۲۲	۰/۸۶	۰/۹۱	-۵
۲۵۸	۱	۱	۰

به این صورت که تمامی مساحت‌ها بر بزرگ‌ترین مساحت موجود در گروه خود تقسیم شدند و مقداری بین صفر و یک گرفتند. اختلاف میان مساحت واقعی و محاسبه‌شده هر چاله به دست آمد و به درصد تبدیل شد که در جدول ۶ آورده شده‌اند.

طبق جدول ۶ و نتایج به‌دست‌آمده، اختلاف بین مساحت‌ها محاسبه‌شده و واقعی بین ۲ و ۵ درصد می‌باشد که بیانگر دقت بالای تبدیلات است.

به‌طورکلی روش اجراشده، هم دقت بالایی را در تبدیل نقاط از فضای دوربین به مدل حاصل کرد و نیز با اعمال حد آستانه، میزان خطای YOLO در تشخیص‌های متفرقه را به حداقل رساند و باعث شد نقاطی که با دقت بالا تبدیل شدند باقی بمانند.

۶- نتیجه‌گیری و پیشنهادات

هدف از این تحقیق شناسایی و طبقه‌بندی چاله‌ها در خیابان‌های شهری بوده است. محققان در کارهای صورت گرفته صرفاً به تشخیص چاله و تائید وجود آن‌ها پرداختند. در این تحقیق، با رویکردی جدید و تلفیق داده‌های دوبعدی تصویری چاله‌ها و موقعیت سه‌بعدی فریم‌ها به موقعیت سه‌بعدی بیضی برازش داده‌شده بر روی چاله‌ها در فضای سه‌بعدی دست پیدا کردیم. به این منظور از الگوریتم ORB-SLAM 3 در تخمین موقعیت فریم‌ها و از YOLO V8m به‌منظور شناسایی چاله‌ها در تصویر دوبعدی استفاده شد. برای کاهش فضای اضافی حاصل از مستطیل محاط کننده حاصل از YOLO بر روی چاله‌ها، بیضی دوبعدی بر آن‌ها برازش داده شد و موقعیت سه‌بعدی دوسوی محورهای بیضی با استفاده از مثلث‌بندی و سرشکنی دسته اشعه محاسبه گردید. قبل از پیاده‌سازی، جهت آموزش شبکه عصبی YOLO، پایگاه داده‌ای آماده‌سازی شد. این داده شامل ۱۲۶۵ تصویر آموزشی، ۱۱۸ تصویر ارزیابی و ۴۰۱ تصویر اعتبارسنجی بوده است که حاصل این آموزش، دقت ۹۲ ٪ شبکه شد. پس از آموزش YOLO، از این الگوریتم به‌منظور شناسایی چاله‌ها در یک خیابان شهری استفاده شد. الگوریتم پیشنهادی برای یک مسیر حدوداً ۱۳ متری مورد ارزیابی قرار گرفت.

مساحت حاصل از روش پیشنهادی با مساحت محاسبه‌شده با استفاده از گیرنده G1 Plus Sout مقایسه گردید. نتایج نشان داد که الگوریتم پیشنهادی با دقت ۲ تا ۵

درصد می‌تواند به‌صورت نسبی مساحت چاله‌ها را تخمین بزند. برای ارزیابی دقت نقاط تبدیل‌شده نیز پارامتر میانگین خطای نگاشت مجدد برای هر شماره از چاله استخراج گردید. میزان خطای این پارامتر برای اکثریت نقاط به‌صورت تقریبی زیر ۱ تا ۳/۵ پیکسل بوده است که دقت قابل‌قبول تبدیل را نشان می‌دهد. قابل‌ذکر است الگوریتم YOLO در تشخیص و دنبال کردن چاله‌ها دارای ضعف بود و موردی که در خروجی این شبکه عصبی به چشم می‌خورد شناسایی شماره چاله‌هایی با تعداد خیلی بیشتر از تعداد چاله‌های موجود در فیلم ورودی بود. دلیل این امر نیز، اکثراً به دلیل قطع دنبال شدن چاله‌های یکسان توسط شبکه عصبی بوده است که با هر بار قطع شدن یک شماره جدید به چاله قبلی الصاق می‌گردد. همچنین در موارد کمی نیز شبکه به‌اشتباه، عوارضی که چاله نبودند را به‌عنوان چاله تشخیص داده است. در تمامی موارد ذکرشده، این تشخیص‌ها در تعداد کمی از فریم‌ها دیده شدند که در مرحله اول برای حذف این دسته از اشتباهات با انتساب عدد ۵ به‌عنوان حداقل تکرار دیده شدن عارضه، تعداد تشخیص‌های شبکه از ۲۵۹ به ۲۳ مورد کاهش یافت. با استخراج تعداد دفعات دیده شدن در فریم‌های متوالی و مقایسه با میانگین خطای نگاشت مجدد هر مورد، رابطه معکوس بین این دو پارامتر مشهود بود، به این صورت که به‌طور غالب، با افزایش تعداد مشاهدات، روند این خطا نزولی بوده است. این نزول در نقاط با تعداد تکرار پایین‌تر بیشتر به چشم می‌خورد و از یک حدی به بعد، افزایش مشاهدات، تغییرات چشمگیری بر نتایج نداشت. به همین منظور جهت کاهش تشخیص‌های متفرقه شبکه عصبی و نیز پرهیز از انتساب عددی به‌صورت تجربی به‌عنوان حداقل تعداد تکرار هر عارضه، حد آستانه‌ای برای پارامتر میانگین خطای نگاشت مجدد در نظر گرفته شد. به این صورت که با انتخاب چاله‌هایی با حداکثر میانگین خطای ۲ پیکسل، تعداد تشخیص‌های شبکه از ۲۳ به ۸ مورد کاهش یافت. خروجی این فیلترینگ، انتخاب نقاط با دقت تبدیل بالا می‌باشد که رابطه با مستقیم با تعداد دفعات دیده شدن عارضه موردنظر در فریم‌های متوالی دارد. از طرفی چون شبکه عصبی YOLO عموماً عارضه‌های درست را در تعداد تکرار بالا شناسایی می‌کند، با این روش می‌توان تشخیص‌های نامرتبط را کاهش داد.

در این کار پژوهشی شبکه عصبی نقش اساسی را ایفا می‌کند؛ زیرا اگر شبکه عصبی با دقت فقط عوارض مدنظر را شناسایی کند، تشخیص‌های متفرقه کاهش می‌یابد و نیز در صورت قوی بودن استحکام شبکه عصبی و آموزش کافی، عارضه مدنظر در طول فیلم و فریم‌های متوالی بدون شکست دنبال می‌شود. به همین ترتیب هرچه تعداد دیده شدن یک عارضه در طول فیلم بیشتر باشد، دقت تبدیل آن نقاط نیز بالاتر می‌رود. به این منظور، در کارهای آتی جهت بهبود خروجی بهتر و دقیق‌تر می‌توان از شبکه عصبی با استحکام بالاتر همراه با مجموعه داده آموزشی جامع‌تر جهت آموزش شبکه استفاده کرد. مورد بعدی که می‌تواند منجر به نتایج بهتری شود، استفاده از دوربین‌های استریو و ترجیحاً صنعتی است. در این تحقیق به دلیل استفاده از دوربین تک‌چشمی تلفن همراه هوشمند، ابهام مقیاس وجود دارد که با استفاده از دوربین استریو این مشکل قابل‌حل است. دوربین‌های صنعتی نیز دارای استحکام بالا هستند که در طول پروژه پارامترهای کالیبراسیون آن‌ها ثابت می‌ماند اما در دوربین‌های معمولی امکان تغییر این مقادیر خصوصاً در مسیرهای طولانی بسیار بالا می‌باشد.

مراجع

[۱] P. Riya, K. Nakulraj, and A. Anusha, "Pothole Detection Methods," in *2018 3rd International Conference on Inventive Computation Technologies (ICICT)*, 2018: IEEE, pp. 120-123 .

[۲] Y.-M. Kim, Y.-G. Kim, S.-Y. Son, S.-Y. Lim, B.-Y. Choi, and D.-H. Choi" ,Review of Recent Automated Pothole-Detection Methods," *Applied Sciences,* vol. 12, no. 11, 2022, doi: 10.3390/app12115320.

[۳] W. S. Qureshi *et al.*, "An Exploration of Recent Intelligent Image Analysis Techniques for Visual Pavement Surface Condition Assessment," *Sensors (Basel),* vol. 22, no. 22, Nov 21 2022, doi: 10.3390/s22229019.

[٤] K. R and N. S, "Pothole and Object Detection for an Autonomous Vehicle Using YOLO," presented at the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021.

[٥] S. Sen *et al.*, "Pothole Detection System Using Object Detection through Dash Cam Video Feed," presented at the 2023 International Conference for Advancement in Technology (ICONAT), 2023.

[٦] M. Sathvik, G. Saranya, and S. Karpagaselvi, "An Intelligent Convolutional Neural Network based Potholes Detection using Yolo-V7," presented at the 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), 2022.

[٧] D. Chen, N. Chen, X. Zhang, and Y. Guan, "Real-Time Road Pothole Mapping Based on Vibration Analysis in Smart City," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* vol. 15, pp. 6972-6984, 2022, doi: 10.1109/jstars.2022.3200147.

[٨] P. M. Harikrishnan and V. P. Gopi, "Vehicle Vibration Signal Processing for Road Surface Monitoring," *IEEE Sensors Journal,* vol. 17, no. 16, pp. 5192-5197, 2017, doi: 10.1109/jsen.2017.2719865.

[٩] C. G. Harris and J. Pike, "3D positional integration from image sequences," *Image and Vision Computing,* vol. 6, no. 2, pp. 87-90, 1988.

[١٠] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature,* vol. 293, no. 5828, pp. 133-135, 1981.

[١١] Aparna, Y. Bhatia, R. Rai, V. Gupta, N. Aggarwal, and A .Akula, "Convolutional neural networks based potholes detection using thermal imaging," *Journal of King Saud University - Computer and Information Sciences,* vol. 34, no. 3, pp. 578-588, 2022, doi: 10.1016/j.jksuci.2019.02.004.

[١٢] S. Gupta, P. Sharma, D .Sharma, V. Gupta, and N. Sambyal, "Detection and localization of potholes in thermal images using deep neural networks," *Multimedia Tools and Applications,* vol. 79, no. 35-36, pp. 26265-26284, 2020, doi: 10.1007/s11042-020-09293-8.

[١٣] J. Terven, D.-M .Córdova-Esparza, and J.-A. Romero-González, "A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas," *Machine Learning and Knowledge Extraction,* vol. 5, no. 4, pp. 1680-1716, 2023.

[١٤] M. Gao, X. Wang, S. Zhu ,and P. Guan, "Detection and segmentation of cement concrete pavement pothole based on image processing technology," *Mathematical Problems in Engineering,* vol. 2020, no. 1, p. 1360832, 2020.

[١٥] I. Scholl, T. Aach, T. M. Deserno, and T. Kuhlen, "Challenges of medical image processing," *Computer science-Research and development,* vol. 26, pp. 5-13, 2011.

[١٦] Y. Bhatia, R. Rai, V. Gupta, N. Aggarwal, and A. Akula, "Convolutional neural networks based potholes detection using thermal imaging," *Journal of King Saud University-Computer and Information Sciences,* vol. 34, no. 3, pp. 578-588, 2022.

[١٧] S. Patra, A. I. Middya, and S. Roy, "PotSpot: Participatory sensing based monitoring system for pothole detection using deep learning," *Multimedia Tools and Applications,* vol. 80, no. 16, pp. 25171-25195, 2021.

[١٨] R. Sathya and B. Saleena, "A framework for designing unsupervised pothole detection by integrating feature extraction using deep recurrent neural network," *Wireless Personal Communications,* vol. 126 ,no. 2, pp. 1241-1271, 2022.

[١٩] R. Fan, U. Ozgunalp, Y. Wang, M. Liu, and I. Pitas, "Rethinking road surface 3-d reconstruction and pothole detection: From perspective transformation to disparity map segmentation," *IEEE Transactions on Cybernetics,* vol. ,٥٢no. 7, pp. 5799-5808, 2021.

[٢٠] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A Review of Yolo algorithm developments," *Procedia computer science,* vol. 199, pp. 1066-1073, 2022.

[٢١] C. Wan, Y. Pang, and S. Lan, "Overview of YOLO Object Detection Algorithm," *International Journal of Computing and Information Technology,* vol. 2, no. 1, pp. 11-11, 2022.

[٢٢] M. Nazarkevych and N. Oleksiv, "Object recognition system based on the Yolo model and database formation," *Ukrainian Journal of Information Technology,* vol. 6, pp. 120-126, 01/01 2024, doi: 10.23939/ujit2024.01.120.

[٢٣] B. Karthika, M. Dharssinee, V. Reshma, R. Venkatesan, and R. Sujarani, "Object Detection Using YOLO-V8," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2024: IEEE, pp. 1-4 .

[٢٤] B. P, P. P, D. M, and V. S, "Unauthorized Drone Detection Using Deep Learning," *International Journal for Research in Applied Science and Engineering Technology,* vol. 12, no. 5, pp. 282-287, 2024, doi: 10.22214/ijraset.2024.61496.

[٢٥] F. Imanuel, S. K. Waruwu, A. Linardy, and A. M. Husein, "Literature review application of yolo algorithm for detection and tracking," *Journal of Computer Networks, Architecture and High Performance Computing,* vol. 6, no. 3, pp. 1378-1383, 2024.

[٢٦] C. Altuntas, "Triangulation and time-of-flight based 3D digitisation techniques of cultural heritage structures," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* vol. 43, pp. 825-830, 2021.

[۲۷] "pothoe dataset." https://roboflow.com/ (accessed.

[۲۸] W. Song, X. Xie, G. Li, and Z. Wang, "Flexible method to calibrate projector–camera systems with high accuracy," *Electronics Letters,* vol. 50, no. 23, pp. 1685-1687, 2014, doi/۱۰,۱۰٤۹ :el.2014.1383.

[۲۹] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on pattern analysis and machine intelligence,* vol. 22, no. 11, pp. 1330-1334, 2000.