

مروری بر سیستم‌های ردیابی تماس با استفاده از تصاویر دوربین‌های هوشمند

مجتبی مینایی^۱، محمدرضا ملک^{۲*}

^۱ دانشجوی کارشناسی ارشد سیستم‌های اطلاعات مکانی - دانشکده مهندسی نقشه‌برداری - دانشگاه صنعتی

خواجه‌نصیرالدین طوسی

mojtaba.minaei@email.kntu.ac.ir

^۲ استاد دانشکده مهندسی نقشه‌برداری - دانشگاه صنعتی خواجه‌نصیرالدین طوسی

mrmalek@kntu.ac.ir

(تاریخ دریافت شهریور ۱۴۰۰، تاریخ تصویب مهر ۱۴۰۰)

چکیده

با گسترش ویروس کرونا و تبدیل آن به یک همه‌گیری، به توصیه سازمان بهداشت جهانی قرنطینه و ایجاد محدودیت‌های اجتماعی یک راه اساسی برای کنترل بیماری است. علی‌رغم کارایی مناسب، اجرا و مدیریت این روش‌ها پرهزینه بوده و به شکل عمومی و برای طولانی مدت قابل اجرا نمی‌باشند. سیستم‌های ردیابی تماس می‌توانند به عنوان جایگزینی برای قرنطینه عمومی استفاده شوند. دوربین‌های هوشمند به دلیل گستردگی خود، ساختار مناسبی برای ایجاد یک سیستم هوشمند ردیابی تماس هستند. طراحی و اجرای سیستم مذکور چالش‌هایی دارد. ساختار مدل تشخیص شیء کارا، مجموعه داده کافی و نحوه موقعیت‌یابی و ردیابی افراد چالش‌هایی هستند که در این مقاله با بررسی انواع سیستم‌های ردیابی تماس موجود و ارزیابی نتایج آن‌ها، بررسی شده‌اند.

واژگان کلیدی: ردیابی تماس، کووید ۱۹، دوربین‌های هوشمند، الگوریتم تشخیص شیء، پردازش تصویر

* نویسنده رابط

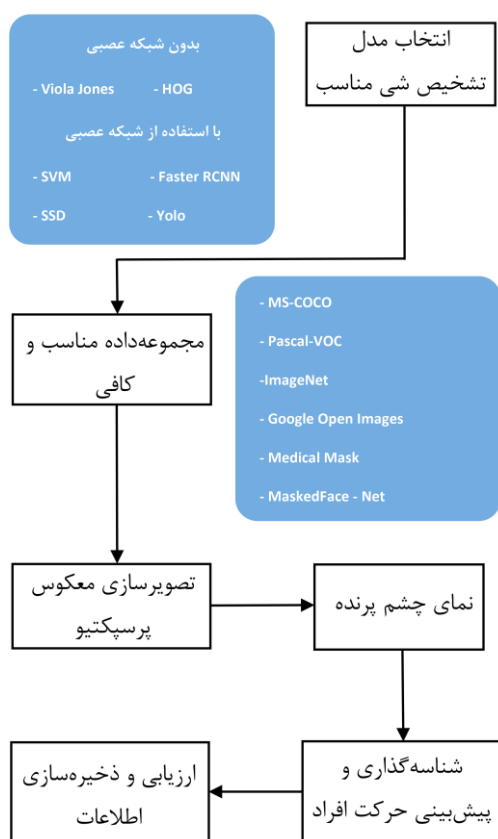
۱- مقدمه

با گسترش ویروس کرونا و تبدیل شدن آن به یک همه‌گیری، به توصیه سازمان بهداشت جهانی، قرنطینه و ایجاد محدودیت‌های اجتماعی یک راه اساسی برای کنترل بیماری است [۱]. وجود تراکم جمعیتی بالا و امکان‌پذیر نبودن تعطیلی برخی از مشاغل و ادارات، منجر به ایجاد سیستم‌های هوشمندی شد که بدون نیاز به تعطیلی و قرنطینه عمومی از شیوع ویروس کرونا جلوگیری کنند [۲]. یکی از تجهیزاتی که می‌تواند بدین منظور استفاده گردد، دوربین‌های هوشمند هستند. امروزه کم‌تر نقطه‌ای را می‌توان یافت که تحت نظارت دوربین‌های هوشمند نباشد. این دوربین‌ها در ابتدا به طور محدود با اهداف امنیتی مثل نظارت بر فرودگاه‌ها و اماکن حساس ایجاد شدند. اما با رشد تکنولوژی، افزایش کیفیت تصاویر، فراهم شدن سیستم‌های ذخیره‌سازی اطلاعات و کاهش قیمت دوربین‌های هوشمند کاربردهای دیگری نیز پیدا کرده و برای اهدافی از قبیل کنترل و نظارت بر معابر و بزرگراه‌های شهری استفاده گردیدند. این دوربین‌ها به دلیل توانایی دریافت اطلاعات در هر ساعتی از روز و با هر میزان روشنایی و همچنین ذخیره‌سازی اطلاعات، هوشمند نامیده می‌شوند. با پوشش مناسبی که این سیستم‌ها در فضاهای باز و مسقف دارند، می‌توانند به‌عنوان ساختاری برای تولید داده‌های موردنیاز در پردازش‌های تصویری استفاده شوند. سیستم‌های هوشمند کنترل ترافیک، خودروهای خودران^۱، سیستم‌های نظارتی - امنیتی خودکار در فرودگاه‌ها و پایانه‌های مسافرتی از جمله کاربردهای دوربین‌های هوشمند در حوزه پردازش تصویر و تشخیص شیء هستند [۳].

پس از گسترش کرونا و افزایش نیاز به راهکارهای متفاوت از قرنطینه عمومی، توجه پژوهشگران معطوف به سیستم‌های ردیابی تماس گردید. این سیستم‌ها از ابزارهایی مثل بلوتوث و امواج رادیویی بهره می‌گیرند اما پوشش بالا نداشته و نیازمند همکاری بالایی از سوی کاربران هستند [۴-۶]. ساختار دیگری که می‌تواند با پوشش بالا و بدون نیاز به تجهیزات جدید برای ردیابی بیماران استفاده گردد دوربین‌های هوشمند هستند.

سیستم‌هایی که از تلفن همراه برای ردیابی بیماران استفاده می‌کنند صرفاً برای تماس‌های فرد - فرد مفید هستند. اما دوربین‌های هوشمند به دلیل ثابت بودن موقعیت خود می‌توانند تماس‌های فرد - محیط را نیز شناسایی کنند [۷]. ساخت این سیستم‌ها ساده نبوده و محققین با چالش‌های متعددی روبه‌رو بوده‌اند که برخی به‌خوبی حل شده و برخی به قوت خود باقی مانده‌اند.

اولین چالش، مدل تشخیص شیء مناسبی است که بتواند افراد را شناسایی و ردیابی کند. به‌منظور آموزش مدل انتخاب‌شده، مجموعه داده‌ای نیاز است که با پوشش کافی دقت مدل را بهبود بخشد. برای ردیابی افراد باید محیط از دوبردی عکسی به سه‌بعدی واقعی برود. سپس با شناسه گذاری و پیش‌بینی حرکت افراد، عملیات ردیابی صورت پذیرد. در نهایت اطلاعات به‌دست‌آمده برحسب نیاز ذخیره‌سازی می‌شوند. شکل ۱ نشان‌دهنده مراحل و چالش‌های پیشرو ساخت یک سیستم ردیابی افراد است.



شکل ۱- مراحل ساخت مدل‌های ردیابی افراد

در این پژوهش، هرکدام از مراحل مطرح شده در شکل ۱ بررسی شده و با روشن‌سازی موضوعات کمتر کار شده، زمینه تحقیق بیشتر محققان فراهم شده است. در

^۱ Self-Driving Cars

تشکیل مستطیلی به دور اشیاء و سپس محاسبه احتمال عضویت و یا عدم عضویت آن شیء در طبقه هدف، اشیاء را تشخیص می‌دهد [۱۰]. از نظر شرایط اجرا این روش قوی^۷ بوده و می‌تواند ما را به سرعت آنی^۸ برساند اما دقت آن وابسته به پوشش و گستردگی فایل طبقه ساز است [۱۱]. در حال حاضر طبقه سازهای آبخاری زیادی تولید و تحت کتابخانه‌ی OpenCV در دسترس عموم قرار گرفته‌اند. از چهارچوب ویولا-جونز برای شناسایی چهره، رفتار شناسی و تشخیص اتومبیل‌ها استفاده شده است [۱۲-۱۴].

۲-۲- هیستوگرام گرادیان جهت‌دار

در این روش با نحوه تغییرات شدت رنگ پیکسل‌ها، عکس به مناطق کوچکی تقسیم می‌شود که به هرکدام یک سلول گفته می‌شود. یک سلول بخشی از یک شیء است. با در کنار هم قرارگرفتن سلول‌ها در برحسب تغییرات شدت رنگشان، یک بلوک تشکیل می‌شود. در نهایت باتوجه به جهت تغییرات شدت رنگ، اشیاء شناسایی می‌شوند. این روش نیز همانند چهارچوب ویولا-جونز نیازمند اطلاعات ویژگی‌ها بوده و از شبکه‌ی عصبی استفاده نمی‌کند. در مقایسه با روش قبلی نیاز به داده‌های آموزشی کمتری دارد اما دقت آن وابسته به شرایط تشخیص بوده و قوی نیست [۱۳].

۲-۳- ماشین بردار پشتیبانی (SVM)

این روش برای دسته‌بندی به روش راهنمایی شده استفاده می‌شود. از هسته‌ها^۹ برای دسته‌بندی استفاده شده و با روشی غیر احتمالاتی اشیاء را طبقه‌بندی می‌کند. هسته‌ها داده‌های ویژه‌ای هستند که می‌توانند به‌عنوان نماینده طبقه خود انتخاب شوند و بردارهای پشتیبانی با استفاده از آن‌ها ساخته شوند. یک بردار پشتیبانی عبارت است از تابعی که دو طبقه را از هم تفکیک می‌کند. از ماشین بردار پشتیبانی برای طبقه‌بندی خطی و غیرخطی استفاده و بدین منظور از فوق صفحه‌ها^{۱۰} استفاده می‌شود. این روش برای داده‌های کوچکی که امکان دسته بندی

بخش ۲ به مدل‌های تشخیص شیء مرتبط پرداخته و با یکدیگر مقایسه می‌شوند. در بخش ۳ مجموعه‌داده‌هایی که برای آموزش هرکدام از این مدل‌ها استفاده شده‌اند مطرح می‌شوند. نحوه تبدیل فضای دوبعدی تصاویر دوربین‌های نظارتی به سه‌بعدی واقعی در بخش ۴ مطرح و در بخش ۵ نیز به روش‌های ردیابی و پیش‌بینی حرکت افراد پرداخته می‌شود. خروجی بخش ۵ اطلاعات فرد - فرد یا فرد - محیط است، نحوه ذخیره‌سازی و استفاده از این اطلاعات در بخش ۶ مطرح می‌شود.

۲- مدل‌های تشخیص شیء مرتبط

شناسایی^۱ و تشخیص^۲ خودکار اشیاء جزو اهداف اصلی پردازش تصویر است. ابزارهای مختلفی می‌توانند برای پردازش تصویر استفاده شوند که یادگیری ماشین^۳ و یادگیری عمیق^۴ جزو آن‌ها هستند [۸]. در این بخش ساختار مدل‌های مختلفی که برای تشخیص افراد استفاده شده‌اند مطرح و با یکدیگر مقایسه می‌شوند. دو رویکرد در الگوریتم‌های تشخیص شیء وجود دارد. اولی این است که بدون استفاده از شبکه عصبی با به‌کارگیری از روش‌هایی که در بخش ۲-۲ و ۳-۲ آمده است، ابتدا ویژگی‌ها استخراج شوند و سپس با استفاده از روش‌های طبقه‌بندی، اشیاء را به هر طبقه اختصاص دهیم. رویکرد دوم استفاده از تکنیک‌های مبتنی بر شبکه عصبی است. در این رویکرد نیازی به استخراج ویژگی‌ها از قبل نبوده و هر دو مرحله با هم انجام می‌شود [۹].

۲-۱- چارچوب Viola - Jones

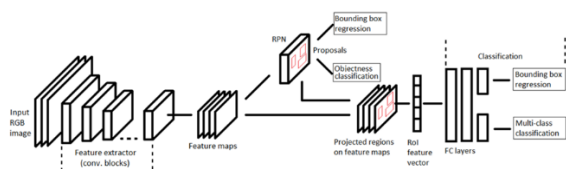
در این روش، ابتدا تفاوت ویژگی‌ها با استفاده از هارگونه‌ها^۵ پیدا می‌شود. هارگونه بیان‌کننده تفاوت مقدار میانگین شدت رنگ در یک همسایگی عکسی است [۹]. سپس از کنار هم قرار گرفتن مجموعه‌ای از این اطلاعات یک طبقه ساز آبخاری^۶ ساخته می‌شود. این طبقه ساز که در غالب فایل XML ساخته می‌شود شامل اطلاعات مثبت و منفی از شیء هدف است. چهارچوب ویولا-جونز با

^۱ Recognition
^۲ Detection
^۳ Machine Learning
^۴ Deep Learning
^۵ Haar-Like
^۶ Cascade Classifiers

^۷ Robust
^۸ Real-Time
^۹ Kernels
^{۱۰} Hyperplanes

۴-۲- Faster R-CNN

در ابتدا روش R-CNN^۲ مطرح شد. این مدل با استفاده از نواحی انتخاب شده توسط کاربر اشیاء را استخراج می‌کند [۱۶]. از معایب آن سرعت بسیار پایین آن است که آن را برای اهداف آنی نامناسب می‌سازد [۲۰]. سپس مدل Fast R-CNN ارائه شد که با تولید نقشه ویژگی‌ها برای هر عکس حجم محاسبات را بشدت کاهش داده و در نتیجه سرعت آن به مراتب بالاتر است. این مدل همچنان دقت بسیار بالایی دارد و سرعتی نامناسب برای اهداف آنی. در این مدل نیز مناطق مورد نظر^۳ به صورت انتخابی توسط کاربر وارد می‌شوند. هنگامی که این مناطق نیز با استفاده از نقشه ویژگی‌ها بدست آمدند، سرعت بالاتری بدست آمد که به عنوان Faster R-CNN نام گذاری شد [۲۱]. در شکل ۳ می‌توان معماری روش ناحیه محور را مشاهده نمود. همان طور که در این شکل مشاهده می‌شود، از سمت چپ، پس از ورود عکس‌های رنگی به مدل، ابتدا ویژگی‌ها استخراج شده و سپس در مرحله بعدی نقشه‌های ویژگی و طبقه‌بندی به صورت همزمان بدست می‌آیند. در هر مرحله از لایه‌های تکاملی استفاده شده و در صورتی که مدل پیچیده باشد می‌توان از لایه‌های مختلف استفاده نمود [۲۲].



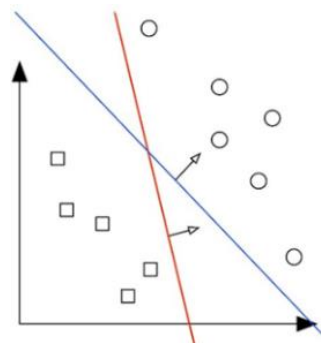
شکل ۳- معماری روش Faster R-CNN [۲۲]

۴-۵- تشخیص‌دهنده تک‌برداشتی (SSD)

این مدل از مدل‌های سریع و دقیق تشخیص شیء است که به صورت دو مرحله، فرایند تشخیص شیء را به ازاء هر یک از تصاویر طی می‌کند و به این ترتیب سریع‌تر از روشی مثل Faster R-CNN است. مدل SSD از دو بخش ستون فقرات و سر تشکیل شده است. قسمت اول به صورت پیش‌آمخته از مجموعه داده شبکه عکسی بدست می‌آید و قسمت دوم نیز چندلایه تکاملی است که طبقه‌بندی

آنها به طور مشخص وجود داشته باشد مناسب است و دقت آن در مقابل طبقه‌های وابسته و داده‌های دارای اعوجاج^۱ بشدت کاهش می‌یابد. در این روش از شبکه‌ی عصبی استفاده شده و نیاز به مراحل پیش محاسبات به منظور استخراج ویژگی‌ها ندارد.

کارکرد این روش وابسته به توانایی ما در ایجاد استقلال بین طبقه‌ها است. به عنوان مثال در [۱۵] مدلی با عنوان Deep Robust One-Class Classification طراحی شده است که تک طبقه‌ای بوده و فقط به صورت ۰ و ۱ عضویت و عدم عضویت را تعیین می‌کند. در [۱۶] با ترکیب مدل SVM و R-CNN با میانگین دقت بالای ۵۰٪ توانسته‌اند اشیاء موجود در مجموعه داده‌ی Pascal VOC را تشخیص بدهند. در کاری که شباهت بیشتری با موضوع این مقاله دارد، با روش SVM مدلی طراحی شده است که با تصاویر دوربین‌های تکی می‌تواند علاوه بر تشخیص اشیاء، زاویه قرارگیری آن‌ها نسبت به دوربین را شناسایی کرده و در نتیجه در طراحی اتوموبیل‌های خودران مفید باشد [۱۷]. از همین روش در مدل DeepSocial که افراد را در سطح خیابان‌ها شناسایی کرده و فاصله بین آن‌ها را اندازه‌گیری می‌کند و به این ترتیب در مدیریت شهری مربوط به کرونا کمک می‌کند نیز استفاده شده است [۱۸]. شکل ۲، دو مثال از بردار پشتیبانی به رنگ‌های آبی و قرمز در فضای جواب را نشان می‌دهد. مشاهده می‌شود که با توجه به استقلال کامل بین دو طبقه مستطیل و دایره می‌توان چندین بردار پشتیبان دقیق تعریف نمود. روش SVM ثبات بسیار زیادی دارد اما سرعت آن وابسته به حجم داده‌های آموزشی و پیچیدگی هسته‌ها است بنابراین نمی‌تواند در مسائل تشخیص شیء آنی استفاده شود.

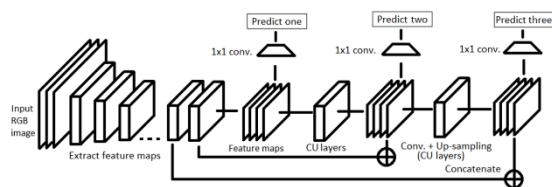


شکل ۲- نمایش دو بردار پشتیبانی [۱۹]

^۲ Region Based Convolutional Neural Networks

^۳ Region Of Interest (ROI)

^۱ Noisy Data



شکل ۵- معماری روش Yolo [۲۲]

۷-۲- مقایسه و ارزیابی

عملکرد یک مدل علاوه بر روشی که استفاده می‌شود، وابسته به پارامترهای مسئله است؛ بنابراین نمی‌توان نتایج مدل‌های مختلف را با هم مقایسه کرد. اما در برخی از آثار از چند روش استفاده و در انتها با یکدیگر مقایسه شده‌اند. نتایج مدل DeepSocial با روش‌های دیگر مقایسه شده و در جدول ۱ آورده شده است:

جدول ۱- مقایسه روش‌های تشخیص شیء [۱۸]

روش	دقت	سرعت
DeepSocial	۹۹/۸٪	۲۴/۱ fps
Yolo(V3)	۸۴/۶٪	۲۳ fps
SSD	۶۹/۱٪	۱۰ fps
Faster R-CNN	۹۶/۹٪	۳ fps

در مدل دیگری سه روش به‌منظور ردیابی افراد استفاده و در جدول ۲ با یکدیگر مقایسه شده‌اند:

جدول ۲- مقایسه روش‌های تشخیص شیء [۲۰]

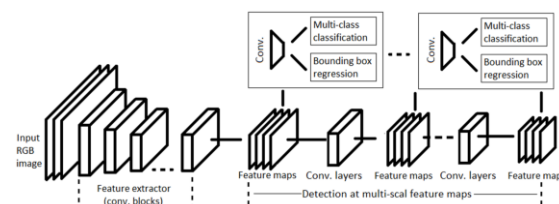
روش	mAP ^۱	IoU ^۲	سرعت
Faster R-CNN	۰/۸۶۸	۰/۹۰۷	۴ fps
Yolo(V4)	۰/۸۴۷	۰/۹۰۶	۹ fps
SSD	۰/۴۴۵	۰/۹۱۸	۲۵ fps

در جدول ۲ شاخص IoU دقت مدل در تشخیص موقعیت اشیاء و قرار دهی قلاب دور آنهاست.

۳- مجموعه داده‌های مرتبط

در بخش ۲ در خصوص روش‌های مختلف تشخیص شیء صحبت شد. پارامترهای بسیاری در عملکرد یک مدل تشخیص شیء تأثیرگذار هستند که باتوجه به آن‌ها می‌توان مدل را بهینه ساخت. اما حتی یک مدل بهینه نیز

موردنظر را انجام می‌دهد [۲۳]. در شکل ۴ می‌توان معماری روش SSD را مشاهده نمود. از سمت چپ، پس از ورود عکس‌های رنگی به مدل، ابتدا ویژگی‌ها استخراج شده و سپس در مرحله بعدی نقشه‌های ویژگی و طبقه‌بندی به‌صورت هم‌زمان بدست می‌آیند. در هر مرحله از لایه‌های تکاملی استفاده شده و در صورتی که مدل پیچیده باشد می‌توان از لایه‌های مختلف استفاده نمود [۲۲].



شکل ۴- معماری روش SSD [۲۲]

روش SSD برای کاربردهای متعددی استفاده شده است. به‌عنوان مثال برای ردیابی انسان‌ها در تصاویر دوربین‌های هوشمند، اتوموبیل‌های خودران، کنترل ترافیک خودکار و فروشگاه‌های هوشمند [۱۸، ۲۰، ۲۲].

۶-۲- You Only Look Once (Yolo)

روش بهبودیافته SSD است که با معماری ساده می‌تواند با سرعتی بالاتر از بقیه روش‌ها عمل تشخیص شیء را انجام دهد. برخلاف روش SSD که برای تشخیص به چندین بار اجرای مدل نیاز دارد، در این روش طبقه‌بندی تنها با یک‌بار اجرا بدست می‌آید [۲۴].

نسخه‌های متعددی برای این روش ساخته شده‌اند که هرکدام روش حل متفاوتی دارند. برای هر نسخه نیز معماری‌های متفاوتی در دسترس است که با نام‌های DarkNet شناخته می‌شوند. این روش سریع‌تر از روش‌های قبلی بوده اما دقت آن وابسته به معماری و مجموعه داده استفاده شده است [۱۸].

از این روش در کاربردهای ردیابی و تشخیص انسان‌ها، شناسایی خودکار زباله‌ها در سطح شهر، شناسایی خودکار تصادفات رانندگی و اتوموبیل‌های خودران استفاده شده است [۱۸، ۲۰، ۲۲]. در شکل ۵ می‌توان معماری روش Yolo را مشاهده نمود. از سمت چپ، ابتدا تصاویر رنگی وارد مدل می‌شوند، سپس ویژگی‌های عکس استخراج شده و در مرحله آخر با اجرای لایه‌های تکاملی طبقه‌بندی صورت می‌گیرد [۲۲].

^۱ mean Average Precision (mAP)

^۲ Intersection Of Union (IOU)

بدون مجموعه داده مناسب و کافی نمی‌تواند خوب آموزش دیده و عمل کند. مجموعه داده عبارت است از داده‌هایی که در چند گروه مختلف اما مرتبط با هم، ذخیره می‌شوند [۲۵].

ساخت مجموعه داده، برحسب کاربرد آن، می‌تواند فرایندی کوتاه و کم‌خرج یا بسیار زمان‌بر و پرهزینه باشد. مجموعه داده‌هایی که برای پردازش تصویر استفاده می‌شوند از نظر اندازه حجیم و حاوی تعداد بسیار زیادی عکس هستند، به همین خاطر اولویت برنامه‌نویسان، استفاده از مجموعه داده‌های موجود است. یک مجموعه داده عکسی دارای ویژگی‌هایی است که آن را برای یک هدف مناسب می‌سازد؛ این ویژگی‌ها عبارتند از [۲۵]:

۱- تعداد طبقه‌ها

۲- تعداد عکس موجود در هر طبقه

۳- تعداد عکس‌های برچسب‌گذاری شده در هر طبقه

۴- تفکیک‌پذیری^۱ عکسی

۵- حجم ذخیره‌سازی مجموعه داده

برای دستیابی به دقت بالا در طراحی مدل ردیابی و تشخیص انسان‌ها، اولویت اول این است که از مجموعه داده‌ای بسیار کامل از انسان‌ها استفاده کنیم که شامل تمامی حالات مختلف قرارگیری، وضعیت نور متفاوت، دید غیر کامل در اندازه‌های مختلف باشد. در صورتی که بخواهیم طبقه‌های بیشتری را تشخیص دهیم، باید پیش از شناسایی این طبقه‌ها از مجموعه داده‌ای استفاده کنیم که جامع بوده و توانایی آموزش تشخیص برای همه آن طبقه‌ها را داشته باشد [۲۶]. در ادامه مجموعه داده‌های مرتبط توضیح داده شده و آن‌ها را بررسی خواهیم کرد.

۳-۱- MS-COCO

یکی از مجموعه داده‌های رایگان و در دسترس عموم است که توسط شرکت مایکروسافت تولید شده است. بیش از ۳۳۰ هزار عکس که حدود ۲۰۰ هزار عدد از آن برچسب‌گذاری شده هستند و در مجموع حدود ۱/۵ میلیون شیء در این تصاویر قابل رویت هستند. از این تعداد ۲۵۰ هزار عدد مربوط به افراد است و در مجموع حدود ۵۰ طبقه مختلف دارد. حجم این مجموعه داده حدود ۲۵ گیگابایت است و در فرمت JSON ارائه می‌شود.

شود [۲۷]. این مجموعه داده در تمامی مدل‌های ذکر شده در بخش ۲ با اهداف متفاوتی مثل تشخیص افراد و تشخیص اتوموبیل استفاده شده است [۱۸، ۲۰، ۲۸].

۳-۲- Pascal VOC

این مجموعه داده نیز به صورت رایگان در دسترس عموم بوده و دارای ۲۰ طبقه شامل ۱۵۰۰ عکس برای آموزش و ۱۵۰۰ عکس برای ارزیابی می‌باشد. از این تعداد ۸۵۱ عکس مربوط به انسان‌ها هستند. حجم آن تنها ۲ گیگابایت بوده و در فرمت XML در دسترس است [۲۷]. این مجموعه داده در کنار MS-COCO برای تشخیص و ردیابی افراد استفاده شده است [۱۸، ۲۰].

۳-۳- ImageNet

نسخه تکمیل شده Pascal VOC توسط دانشگاه استنفورد می‌باشد که علاوه بر اضافه شدن تصاویر، به صورت درختی برچسب‌گذاری شده‌اند. داده‌های این مجموعه داده به گونه‌ای برچسب‌گذاری شده‌اند که بتوانند به سؤالی پاسخ بدهند مثل "آیا در این تصویر اتوموبیل وجود دارد؟" و در صورت مثبت بودن جواب سراغ سؤال بعدی مرتبط می‌رود. حدود ۱۵ میلیون عکس، در ۱۰۰۰ طبقه در این مجموعه داده وجود دارد که حدود ۳ میلیون آن برچسب‌گذاری شده‌اند. حجم آن بیش از ۱۵۰ گیگابایت است [۲۷].

از آنجاکه حق کپی‌رایت عکس‌ها محفوظ است، تنها تصاویر بندانگشتی^۲ آنها در دسترس هستند و در صورت نیاز باید از لینک مرجع هر عکس ابتدا بارگیری شوند؛ بنابراین نمی‌توان کل مجموعه داده را به صورت یکجا دریافت نمود. این مجموعه داده را می‌توان در نسخه‌های محدودتری که هرکدام زیرمجموعه‌ای از داده‌های اصلی هستند دریافت نمود مثل شبکه عکسی کوچک^۳ که تنها ۲۰۰ طبقه دارد [۲۷].

اگرچه این مجموعه داده دارای تصاویر بسیار زیادی از انسان‌ها است اما به دلیل آنکه به صورت درختی برچسب‌گذاری شده‌اند نمی‌تواند مستقیماً برای فرایند طبقه‌بندی استفاده شود. روش‌های تک برداشتی از وزن‌های حاصل از محاسبات صورت‌گرفته روی این مجموعه داده استفاده می‌کنند [۲۷].

^۲ Thumbnail
^۳ Tiny ImageNet

^۱ Resolution

جدول ۳- مقایسه مجموعه داده‌های مرتبط با شناسایی و ردیابی افراد [۲۷]

Size(GB)	Labeled	Images	Classes	Dataset
25	200K	330K	50	MS COCO
2	1.5K	3K	20	Pascal VOC
150	3M	15M	1000	ImageNet
456	1.9M	9M	600	Google Open Images
2.5	3K	6K	20	Medical Mask
38	70K	134K	2	MaskedFace-Net

۴- موقعیت‌یابی با دوربین‌های هوشمند

به‌منظور ردیابی و بررسی فاصله اجتماعی بین افراد، نیازمند موقعیت‌یابی در تصاویر دوربین‌های هوشمند هستیم. سال‌هاست در رشته‌های سنجش‌ازدور و فتوگرامتری از تصاویر استریو برای توجیه عکسی، کالیبره سازی دوربین‌ها و محاسبه فاصله استفاده می‌شود. دوربین‌های هوشمند علی‌رغم گستردگی خود هنوز به پوشش مشترک کافی در تصاویر نرسیده و بنابراین نمی‌توان از تکنیک‌های استریو برای موقعیت‌یابی استفاده نمود [۳۱]. مهم‌ترین مانع برای موقعیت‌یابی با استفاده از دوربین‌های هوشمند خطای پرسپکتیو است. خطای پرسپکتیو، عبارت است از وجود مقیاس متفاوت در نقاط مختلف تصویر. در دید سه‌بعدی بین جسمی که بزرگ‌تر است با جسمی که نزدیک‌تر است تفاوت وجود دارد اما در تصاویر دوربین‌های هوشمند این تفاوت آشکار نیست. این خطا وابسته به زاویه دوربین و جسم است [۳۲]. در ادامه تبدیل خنثی کننده این اثر و سپس نمای دید پرنده توضیح داده می‌شوند.

۴-۱- تصویرسازی معکوس پرسپکتیو

یکی از راهکارهای برگرداندن فضای دو بعد به سه‌بعدی است. در تصاویر سه‌بعدی می‌توانیم مرکز چهارچوب دور افراد را با سه پارامتر (x, y, z) نمایش دهیم. اما در تصاویر دوبعدی تنها (x, y) وجود دارد و به عبارتی محوری z با دو محور دیگر ترکیب شده است؛ بنابراین دیگر نمی‌توان از فواصل اقلیدسی استفاده نمود. برای جداکردن محور z از دو محور دیگر، ابتدا باید دوربین را توجیه و کالیبره کنیم. بدین منظور چند گام باید برداشته شود [۱۸]:

- محاسبه پارامترهای توجیه داخلی دوربین
- محاسبه موقعیت، ارتفاع و زاویه دید دوربین
- تبدیل ارتفاع اشیاء به ارتفاع صفر ($z = 0$)

۳-۴- Google Open Images

این مجموعه داده توسط گوگل جمع‌آوری شده است و به صورت رایگان در اختیار عموم می‌باشد. تصاویر عمومی گوگل دارای ۹ میلیون عکس در ۶۰۰ طبقه مختلف است که ۱/۹ میلیون آن علاوه بر برچسب، دارای اطلاعات روابط بین طبقه نیز هستند و به این ترتیب می‌توان مشکل عدم استقلال طبقه‌ها را برطرف نمود [۲۷].

۳-۵- Medical Mask

این مجموعه داده حاوی ۶۰۰۰ عکس از افراد است که تنوع گسترده‌ای شامل سن، قومیت و مناطق دارد. این مجموعه شامل ۲۰ طبقه است که انواع مختلف ماسک و نحوه استفاده صحیح و غیر صحیح را پوشش می‌دهد و برای مدل‌های رفتارشناسی کرونا مناسب می‌باشد. این مجموعه داده را می‌توان به صورت رایگان در حجم ۲/۵ گیگابایت بارگیری نمود [۲۹].

۳-۶- MaskedFace - Net

این مجموعه داده نیز همانند مجموعه داده ماسک پزشکی برای کنترل ویروس کرونا ساخته و در دسترس عموم قرار گرفته است. این مجموعه حاوی ۱۳۴ هزار عکس است که در حجم ۳۸ گیگابایت قابل بارگیری هستند [۲۹]. با استفاده از این داده‌ها مدلی طراحی شده است که وضعیت ماسک افراد را شناسایی کرده و در صورت تناقض با استانداردهای مربوط به کرونا به افراد هشدار می‌دهد [۳۰].

۳-۷- مقایسه مجموعه داده‌ها

انتخاب مجموعه داده وابسته به اهداف مدل و رویکرد اتخاذ شده در روش تشخیص شیء است. افزایش طبقه و تعداد عکس، تشخیص‌های منفی را کاهش داده اما منجر به افزایش حجم محاسبات می‌شود. مجموعه داده‌هایی که فرمت یکسان دارند را می‌توان به سادگی ترکیب نمود. در جدول ۳ خلاصه‌ای از آنچه در خصوص مجموعه داده‌های مختلف بیان شده است.

^۱ Perspective Error

پارامترهای توجیه داخلی دوربین‌های هوشمند را می‌توان با استفاده از مشخصات مندرج در دفترچه راهنمای این دوربین‌ها به دست آورد. با فرض اینکه در طول محاسبات موقعیت و زاویه دوربین تغییر نکند، می‌توان با یک‌بار اندازه‌گیری به ارتفاع، زاویه دید و موقعیت آن رسید. این اندازه‌ها را می‌توان به صورت صحیح و یا وزن‌دار وارد محاسبات نمود. اگر فرض کنیم که پای افراد همواره روی زمین ($z = 0$) قرار دارد، صفر نمودن ارتفاع ساده بوده و می‌توان موقعیت آن‌ها را با استفاده از مستطیل بدست آمده از تشخیص شیء تعریف نمود. برای تبدیل موقعیت دوبعدی (u, v) به سه‌بعدی (X_m, Y_m, Z_m) از رابطه زیر استفاده می‌کنیم [۱۸] (رابطه ۱):

$$[u \ v \ 1]^T = KRT[X_\omega \ Y_\omega \ Z_\omega \ 1]^T \quad (۱)$$

در رابطه ۱، R ماتریس دوران است (رابطه ۲):

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (۲)$$

در رابطه ۱، T ماتریس تفسیر است (رابطه ۳):

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\frac{h}{\sin \theta} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (۳)$$

همچنین K ماتریس پارامترهای توجیه داخلی دوربین است (رابطه ۴):

$$K = \begin{bmatrix} f * ku & s & C_x & 0 \\ 0 & f * kv & C_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (۴)$$

در روابط ۱ تا ۴ ارتفاع دوربین، f فاصله کانونی دوربین، ku و kv پارامترهای کالیبراسیون دوربین در جهت‌های افقی و عمودی هستند و C_x و C_y جابه‌جایی مرکز تصویر دوربین است. می‌توان فرمول ۴ را بدین صورت بازنویسی کرد [۱۸] (رابطه ۵):

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} X_\omega \\ Y_\omega \\ Z_\omega \\ 1 \end{bmatrix} \quad (۵)$$

در رابطه ۵، $X_\omega, Y_\omega, Z_\omega$ مختصات همگن شده هستند.

پس از مرحله قطعه‌بندی^۱ چهارچوب‌های مستطیلی شکل به دور افراد کشیده می‌شود. این مستطیل‌ها در تصاویر مونو بدون دوران بوده و زوایا همگی ۹۰ درجه است. همان‌طور که اشاره شد مسئله ما محاسبه قد افراد نیست، بلکه می‌خواهیم به موقعیت افراد برسیم، بنابراین اگر برای چهارچوب دور هر فرد، نقطه سمت چپ پایین را به‌عنوان نماینده در نظر بگیریم، می‌توانیم تمامی محاسبات را انجام دهیم و نیازی به اطلاعات نقاط دیگر نیست. ویژگی مشترک همه افراد این است که روی سطح زمین قرار دارند و در نتیجه ($z = 0$) است. پس می‌توان Z_ω را از رابطه ۵ حذف و ماتریس مجهولات را نیز کوچک‌تر نمود [۱۸] (رابطه ۶):

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} X_\omega \\ Y_\omega \\ 1 \end{bmatrix} \quad (۶)$$

۴-۲- دید پرنده

پس از تبدیل عکس پرسپکتیو، محیط به نمای دید پرنده تبدیل می‌شود؛ بدین منظور، کاربر باید برای هر دوربین و در هر وضعیت دو خط موازی انتخاب کند. در واقع با این کار یک مستطیل در فضای سه‌بعدی به سیستم معرفی می‌شود. حال با یک تبدیل کشیدگی و دوران، فضا سه‌بعدی می‌شود [۳۳]. تصویر ناشی از این دوران بسیار خطا داشته و عملکرد تشخیص شیء به‌شدت مختل می‌شود. حتی پس از تشخیص شیء نیز تصاویر بسیار به‌هم‌ریخته شده و اشیاء قابل تشخیص نیستند [۳۴]. اما اگر به‌جای چندضلعی و خط، تنها نقاط را تبدیل کنیم می‌توان به‌دقت مناسبی رسید. بدین منظور نقطه وسط روی ضلع پایین چارچوب ($z = 0$) انتخاب شده و مابقی پیکسل‌ها حذف می‌شوند. تبدیل اعمال شده و سپس روابط مکانی محاسبه می‌شود [۲۰]. این امر را می‌توان در شکل ۶ به‌خوبی مشاهده نمود. در شکل ۷ نمودار تبدیل چشم پرنده آورده شده است.



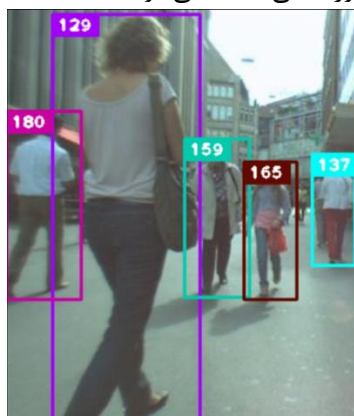
شکل ۶- تبدیل دید پرنده

^۱ Segmentation

۵-۱- شناسه گذاری افراد

برای آن که افراد ردیابی شوند ابتدا باید شناخته و از همدیگر تمییز داده شوند. بدین منظور باید به هر فرد یک شناسه منحصر به فرد اختصاص داده شده و اطلاعات دیگر برحسب این شناسه ذخیره شوند. در چارچوب SORT ابتدا ویژگی‌های افراد توسط الگوریتم‌های تشخیص شیء استخراج می‌شود و در صورتی که این ویژگی‌ها برای فرد دیگری در پایگاه داده وجود نداشته باشد به او یک شناسه اختصاص می‌یابد. در صورتی که این ویژگی‌ها قبلاً در پایگاه داده ذخیره شده باشند، شناسه قبلی فرد فراخوانی می‌شود [۳۷].

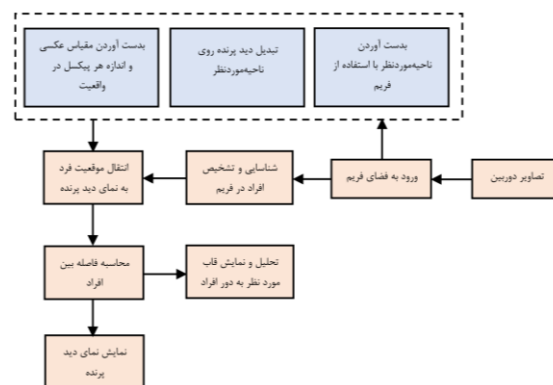
اطلاعات مراحل بعدی نیز برحسب شناسه افراد ذخیره می‌شود؛ بنابراین حجم اطلاعات موجود در پایگاه داده مسئله مهمی است. برحسب هدف ردیابی می‌توان اطلاعات قدیمی را از پایگاه داده حذف نمود. به عنوان مثال در سیستم‌های خودران می‌توان با در نظر گرفتن شروط خروج شیء از کادر و ظاهر نشدن آن تا ۱۰ ثانیه بعد اطلاعات آن را حذف نمود. در ردیابی افراد به منظور حفظ فاصله گذاری اجتماعی می‌توان اطلاعات را در ابتدای هر روز یا پس از هر بار اطلاع رسانی حذف نمود [۳۸]. در مدل DeepSocial تنها اطلاعات ثانویه و تحلیلی باقی مانده و مابقی به صورت آنی حذف می‌شوند [۱۸].



شکل ۸- نحوه شناسه گذاری در چهارچوب SORT [۲۰]

۵-۲- پیش بینی حرکت افراد

در چهارچوب SORT، از پالایه کالمن^۳ برای پیش بینی حرکات افراد و از روش بهینه سازی مجارستانی^۴ برای



شکل ۷- فلوچارت مراحل طی شده برای رسیدن به دید پرنده در مدل DeepSocial [۱۸]

همان طور که در بخش ۴-۱ ذکر شد برای رسیدن به دید پرنده باید دوربین‌ها توجیه شوند. این کار می‌تواند به صورت اتوماتیک با استفاده از داده‌های سنسورهای اندازه گیری اینرشیال^۱ باشد [۳۵].

۵- تکنیک‌های ردیابی افراد

منظور از ردگیری، ایجاد ارتباط بین اشیاء در چند فریم متفاوت است. با تحلیل روابط توپولوژیک می‌توانیم نحوه رفتار افراد در یک فیلم را بررسی کنیم؛ در صورتی که فاصله اجتماعی مناسب بین افراد برای جلوگیری از سرایت کرونا ۲/۸ متر باشد، اگر برای هر فرد یک شعاع ۱/۴ متری در نظر بگیریم، اگر در لحظه اول رابطه توپولوژیک جدا، در لحظه دوم ملاقات و در لحظه سوم متصل باشد می‌توان نتیجه گرفت که این افراد فاصله اجتماعی را رعایت نکرده‌اند. حال اگر یکی از افراد مبتلا باشد فرد دیگر را نیز در صورت ارتباط زمانی کافی می‌توانیم مبتلا بدانیم [۱۸، ۳۶].

برای ردیابی افراد سه گام باید برداشته شود:

۱- شناسه^۲ گذاری افراد

۲- پیش بینی حرکت افراد

۳- تحلیل و ذخیره سازی اطلاعات

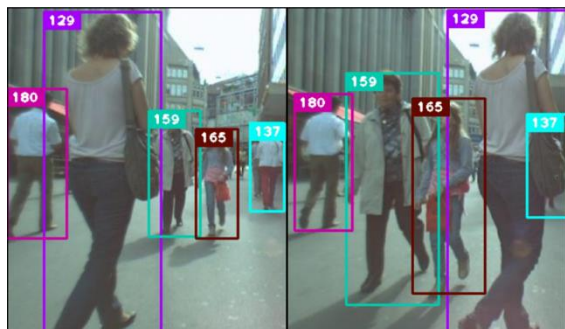
Simple and Online Real-Time Tracking (SORT) نام

چارچوبی است که امکان انجام هر سه گام را به صورت آنی و با حجم محاسبات کم می‌دهد [۳۷]. در این بخش مراحل و نحوه کار این چهارچوب شرح داده می‌شوند.

^۳ Kalman Filter
^۴ Hungarian Optimization

^۱ Inertial Measurement Unit(IMU)
^۲ Identity(ID)

شیء، s مساحت و r نیز جهت آن است [۳۹]. نتایج پیش-بینی حرکت افراد را می‌توان در شکل ۹ مشاهده نمود.



شکل ۹- مقایسه‌ی دو فریم بدست آمده از پالایه کالمن و الگوریتم مجارستانی

الگوریتم مجارستانی در دسته الگوریتم‌های بهینه‌سازی ترکیبی^۱ قرار می‌گیرد که مسئله تخصیص وظایف را حل می‌کند. این الگوریتم همچنین برای پیداکردن شباهت در مجموعه داده‌ها استفاده می‌شود [۴۰]. به دلایل مختلف ممکن است تصاویر حاصل از دوربین‌های نظارتی پیوسته نباشد مثل وجود مانع یا نور نامناسب. اولویت اول استفاده از تصاویر پیوسته و تحلیل حاصل از روش مجارستانی است تا فرد در فریم‌های مختلف شناسایی شود. همچنین پس از شناسایی فرد در دو فریم مختلف با فاصله زمانی t در دو فریم و محاسبه فاصله طی شده او می‌توان از فرمول ۱۴ سرعت او را محاسبه کرد [۲۰]:

$$\Delta x = v \cdot t \quad (14)$$

در نهایت برای هر فریم نتایج حاصل از الگوریتم مجارستانی و پالایه کالمن مقایسه می‌شود. اگر نتایج شباهت زیادی با هم داشته باشند از تحلیل‌های حاصل از پالایه کالمن استفاده می‌شود اما اگر تفاوت داشتند از الگوریتم مجارستانی استفاده می‌شود [۱۸]. تصاویری از مدل ردیابی افراد DeepSocial در شکل ۱۰ آورده شده است. از سمت چپ، ابتدا افراد شناسه گذاری می‌شوند. سپس حرکت آن‌ها با استفاده از پالایه کالمن و الگوریتم مجارستانی پیش‌بینی می‌شود. این کار برای فریم‌های بعدی نیز تکرار می‌شود.



شکل ۱۰- اجرای SORT در مدل DeepSocial [۱۸]

پیداکردن شباهت افراد و شناسایی آن‌ها در فریم‌های مختلف استفاده می‌شود.

معادلات ساده شده پالایه کالمن را در می‌توان در ۴ گام نوشت:

۱- معادله پردازش (رابطه ۷):

$$X_k = Ax_{k-1} + W_{k-1} \quad (7)$$

در رابطه ۷، A ماتریس تبدیل و x_k وضعیت در لحظه‌ی k نسبت به لحظه‌ی $k-1$ است. W نیز بردار خطای گاوسی با توزیع نرمال است.

۲- معادله اندازه‌گیری شده (رابطه ۸):

$$Z_k = Hx_k + V_k \quad (8)$$

در رابطه ۸، H ماتریس مشاهدات و Z_k وضعیت مشاهده شده در لحظه k است. v_k نیز بردار خطای گاوسی با توزیع نرمال است.

۳- معادله به‌روزرسانی زمان:

از دو معادله ۷ و ۸ برای ایجاد مدل خطی در لحظه k استفاده می‌شود. از آنجایی که x_k مستقیماً از مشاهدات و اندازه‌گیری‌ها بدست نمی‌آید پس از اطلاعات Z_k برای به‌روزرسانی وضعیت x_k استفاده می‌شود. پس وضعیت قبلی $\hat{x}_k^-$ و خطای آن $\hat{x}_k$ از وضعیت قبلی بدست می‌آید (روابط ۹ و ۱۰):

$$\hat{x}_k^- = A\hat{x}_{k-1} + W_k \quad (9)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (10)$$

۴- معادله به‌روزرسانی اندازه‌ها (روابط ۱۱ تا ۱۳):

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (11)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (12)$$

$$P_k = (1 - K_k H)P_k^- \quad (13)$$

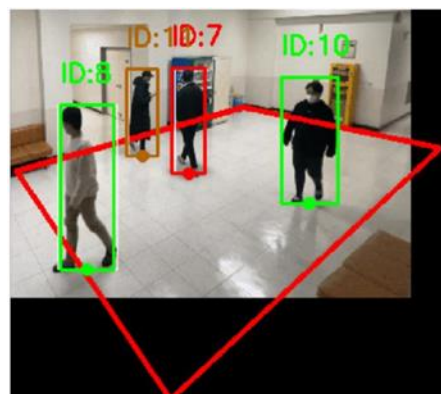
در نهایت با استفاده از معادلات بالا وضعیت به‌روزرسانی شده شیء به دست می‌آید. نتایج این تکنیک از دو جهت مفید است؛ اولاً می‌تواند اطلاعات کمکی برای روش بهینه‌سازی مجارستانی باشد و ثانیاً در صورت پیوسته نبودن پوشش تصاویر برای تحلیل‌های بعدی استفاده شود. خروجی این تکنیک ماتریس مقابل است: $x = [u, v, s, r, u', v', s']$ که u و v موقعیت مرکز قاب دور

^۱ Combinatorial

۵-۳- ارزیابی و ذخیره سازی اطلاعات

پس از اتمام ردیابی و به دست آمدن نتایج، تحلیل های لازم انجام شده و نتایج مورد نظر در پایگاه داده ذخیره می شوند. این تحلیل ها را می توان با دو رویکرد فرد - فرد و فرد - محیط انجام داد.

برای بررسی تماس های فرد - فرد کنترل فاصله افراد و پوشش ماسک نیاز است. با ایجاد یک حائل^۱ به دور افراد و بررسی روابط توپولوژیک در نمای دید پرنده می توان فاصله بین افراد را بررسی کرد. باتوجه به وجود مجموعه داده های ماسک پزشکی و شبکه چهره ماسک دار می توان مدل تشخیص شیء را به گونه ای آموزش داد که نحوه پوشش ماسک را شناسایی و سپس برحسب فاصله افراد شانس انتقال بیماری را محاسبه کند [۳۶]. نمونه ای از ردیابی تماس فرد - فرد در شکل ۱۱ آورده شده است. ابتدا تمامی مراحل ردیابی آنی و برخط اجرا شده و سپس فاصله بین افراد محاسبه شده است. فاصله کافی در این مدل ۱/۸ متر است.



شکل ۱۱- بررسی تماس فرد - فرد [۳۶]

در بررسی تماس های فرد - محیط، نتایج باید برحسب محیط و افراد ذخیره شوند. بدین منظور محیط باید شناسه گذاری، اشتراک بین دوربین ها مشخص و روابط مکانی بین محیط برقرار شود. تماس های فرد - محیط عبارتند از عطسه، سرفه و هر نوع رفتار از فردی که ماسک ندارد. با شناسایی وضعیت ماسک افراد، شانس بیماری زایی محاسبه و نتایج برحسب محیطی که فرد در آن حضور داشته ذخیره می شود.

۶- نتیجه گیری

استفاده از دوربین های هوشمند به عنوان زیرساختی موجود و در دسترس راه مناسبی برای کنترل و حفظ فاصله گذاری اجتماعی است. هرچند ایجاد سیستمی که بتواند مبتلایان را شناسایی و افراد مشکوک را ردیابی کند ممکن می باشد اما در این راه چالش های مهمی وجود دارد. دقت و سرعت مهم ترین پارامترهای یک مدل تشخیص شیء هستند. با توجه به مباحثی که در بخش ۲ مطرح شد، مدل های YoloV3، YoloV4 و SSD بالاترین سرعت و دقت را نسبت به مدل های دیگر دارند. مدل های SVM و R-CNN اگرچه ثبات عملکرد بیشتری نسبت به مدل های دیگر دارند اما سرعت پایین، آن ها را برای اهداف آنی پردازش تصویر نامناسب ساخته است. مدل های HOG و Viola-Jones نیز علی رغم آنکه سرعت خوبی دارند اما دقت پایینی داشته و نمی توانند برای ردیابی افراد مناسب باشند. مدل های تشخیص شیء برای افزایش دقت، به گونه ای طراحی می شوند که بیشترین تعداد پارامتر را داشته باشند. اما افزایش تعداد پارامتر شانس overfit شدن مدل را افزایش می دهد. پوشش طبقه های ممکن در مسئله و افزایش حجم داده های آموزشی از overfit شدن مدل جلوگیری می کند. مجموعه داده های Pascal VOC و MS-COCO از تعداد طبقه های کمتری نسبت به Google Open Images و ImageNet برخوردار بوده و دقت کمتری دارند. اما از طرفی حجم محاسبات را کاهش داده و به سرعت بیشتری می رسند. بنابراین شناخت مسئله و دادوستد بین دقت و سرعت، تعیین کننده مجموعه داده مناسب است.

دربین مقالات و آثار موجود، مدلی که بتواند با استفاده از رفتارشناسی، افراد آلوده و سپس محیط های آلوده را تشخیص بدهد وجود نداشته و لذا نیازمند نوآوری و پیشرفت است. چالش های موجود الگوریتم رفتارشناسی آنی و مجموعه داده های کافی برای انجام این کار هستند.

^۱ Buffer

مراجع

- [1] F. Aslam, "COVID-19 and importance of social distancing," 2020.
- [2] L. Nanotkar, S. Dhanvij, and A. Joshi, "COVID-19 and importance of social distancing," *Journal of Critical Reviews*, vol. 7, no. 8, pp. 1103-1104, 2020.
- [3] M. Bramberger, A. Doblander, A. Maier, B. Rinner, and H. Schwabach, "Distributed embedded smart cameras for surveillance applications," *Computer*, vol. 39, no. 2, pp. 68-75, 2006.
- [4] G. F. Hatke et al., "Using Bluetooth Low Energy (BLE) signal strength estimation to facilitate contact tracing for COVID-19," *arXiv preprint arXiv:2006.15711*, 2020.
- [5] D. J. Leith and S. Farrell, "Coronavirus contact tracing: Evaluating the potential of using bluetooth received signal strength for proximity detection," *ACM SIGCOMM Computer Communication Review*, vol. 50, no. 4, pp. 66-74, 2020.
- [6] I. Nakamoto et al., "Evaluation of the design and implementation of a peer-to-peer COVID-19 contact tracing mobile app (COCOA) in Japan," *JMIR mHealth and uHealth*, vol. 8, no. 12, p. e22098, 2020.
- [7] S. Ojagh, S. Saeedi, and S. H. Liang, "A Person-to-Person and Person-to-Place COVID-19 Contact Tracing System Based on OGC IndoorGML," *ISPRS International Journal of Geo-Information*, vol. 10, no. 1, p. 2, 2021.
- [8] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.
- [9] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *Proceedings. international conference on image processing*, 2002, vol. 1: IEEE, pp. I-I .
- [10] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137-154, 2004.
- [11] D. Chatterjee and S. Chandran, "Comparative study of camshift and KLT algorithms for real time face detection and tracking applications," in *2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, 2016: IEEE, pp. 62-65 .
- [12] D. Hefenbrock, J. Oberg, N. T. N. Thanh, R. Kastner, and S. B. Baden, "Accelerating Viola-Jones face detection to FPGA-level using GPUs," in *2010 18th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines*, 2010: IEEE, pp. 11-18 .
- [13] Y. Xu, G. Yu, Y. Wang, X. Wu, and Y. Ma, "A hybrid vehicle detection method based on viola-jones and HOG+ SVM from UAV images," *Sensors*, vol. 16, no. 8, p. 1325, 2016.
- [14] L. Yun and Z. Peng, "An automatic hand gesture recognition system based on Viola-Jones method and SVMs," in *2009 Second International Workshop on Computer Science and Engineering*, 2009, vol. 2: IEEE, pp. 72-76 .
- [15] S. Goyal, A. Raghunathan, M. Jain, H. V. Simhadri, and P. Jain, "DROCC: Deep robust one-class classification," in *International Conference on Machine Learning*, 2020: PMLR, pp. 3711-3721 .
- [16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580-587 .
- [17] Z. Liu, Z. Wu, and R. Tóth, "Smoke: Single-stage monocular 3d object detection via keypoint estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 996-997 .
- [18] M. Rezaei and M. Azarmi, "Deepsocial: Social distancing monitoring and infection risk assessment in covid-19 pandemic," *Applied Sciences*, vol. 10, no. 21, p. 7514, 2020.
- [19] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. , pp. 273-297, 1995.
- [20] M. Shorfuzzaman, M. S. Hossain, and M. F. Alhamid, "Towards the sustainable development of smart cities through mass video surveillance: A response to the COVID-19 pandemic," *Sustainable cities and society*, vol. 64, p. 102582, 2021.
- [21] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91-99, 2015.
- [22] N. S. Pun, S. K. Sonbhadra, S. Agarwal, and G. Rai, "Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques," *arXiv preprint arXiv:2005.01385*, 2020.

- [23] W. Liu et al., "Ssd: Single shot multibox detector," in European conference on computer vision, 2016: Springer, pp. 21-37 .
- [24] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7263-7271 .
- [25] P. Marshall, N. Rajguru, and A. Slosar, "Bayesian evidence as a tool for comparing datasets," Physical Review D, vol. 73, no. 6, p. 067302, 2006.
- [26] F. Abramovich and M. Pensky, "Classification with many classes: challenges and pluses," Journal of Multivariate Analysis, vol. 174 ,p. 104536, 2019.
- [27] K. Gauen et al., "Comparison of visual datasets for machine learning," in 2017 IEEE International Conference on Information Reuse and Integration (IRI), 2017: IEEE, pp. 346-355 .
- [28] S. Jamiya S and E. Rani P, "LittleYOLO-SPP: A delicate real-time vehicle detection algorithm," arXiv e-prints, p. arXiv: 2011.05940, 2020.
- [29] M. Loey, G. Manogaran, M. H. N. Taha, and N. E. M. Khalifa, "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection," Sustainable cities and society, vol. 65, p. 102600, 2021.
- [30] A. Cabani, K. Hammoudi, H. Benhabiles, and M. Melkemi, "MaskedFace-Net–A dataset of correctly/incorrectly masked face images in the context of COVID-19," Smart Health, vol. 19, p. 100144, 2021.
- [31] B. Rinner and W. Wolf, "An introduction to distributed smart cameras," Proceedings of the IEEE, vol. 96, no. 10, pp. 1565-1575, 2008.
- [32] J. Valente and S. Soatto, "Perspective distortion modeling, learning and compensation ",in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015, pp. 9-16 .
- [33] A. Loukkal, Y. Grandvalet, T. Drummond, and Y. Li, "Driving among Flatmobiles: Bird-Eye-View occupancy grids from a monocular camera for holistic trajectory planning," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 51-60 .
- [34] Y. Lu, J. Huang, Y.-T. Chen, and B. Heisele, "Monocular localization in urban environments using road markings," in 20th IEEE Intelligent Vehicles Symposium (IV), 2017: IEEE, pp. 468-474 .
- [35] S. Wang, S. Fidler, and R. Urtasun, "Lost shopping! monocular localization in large indoor spaces," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2695-2703 .
- [36] M. Shin and N. Moon, "Indoor Distance Measurement System COPS (COVID-19 Prevention System)," Sustainability, vol. 13, no. 9, p. 4738, 2021.
- [37] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking ",in 2016 IEEE international conference on image processing (ICIP), 2016: IEEE, pp. 3464-3468 .
- [38] Z. Qiu et al., "Vision-based moving obstacle detection and tracking in paddy field using improved yolov3 and deep SORT," Sensors, vol. 20, no. 15, p. 4082.۲۰۲۰ ,
- [39] S. S. Pathan, A. Al-Hamadi, and B. Michaelis, "Intelligent feature-guided multi-object tracking using Kalman filter," in 2009 2nd International Conference on Computer, Control and Communication, 2009: IEEE, pp. 1-6 .
- [40] A. Frank, "On Kuhn's Hungarian method—a tribute from Hungary," Naval Research Logistics (NRL), vol. 52, no. 1, pp. 2-5, 2005.