مقاله پژوهشی

مجله ماشین‌بینایی و پردازش تصویر

دوره ۱۳ ، شماره ۱ ، تابستان ۱۴۰۵

افزایش ایمنی رانندگی با تشخیص خودکار عدم تمرکز راننده مبتنی‌بر یادگیری عمیق

سمیرا کریمی چقاکبودی[۱] و عبداله چاله‌چاله[۲]

چکیده

با افزایش تصادفات جاده‌ای ناشی از حواس‌پرتی و رفتارهای ناهنجار رانندگان، استفاده از سامانه‌های هوشمند مبتنی‌بر بینایی ماشین به یکی از راهکارهای مؤثر در ارتقای ایمنی حمل‌ونقل تبدیل شده است. در این مقاله، یک روش نوین مبتنی‌بر یادگیری عمیق برای شناسایی خودکار عوامل عدم تمرکز راننده ارائه می‌شود. در روش پیشنهادی، شبکه عصبی پیچشی ConvNeXt-Base به‌همراه مکانیزم توجه SE-Net به‌منظور استخراج ویژگی‌های مؤثر به‌کار گرفته شده و خروجی آن به شبکه عصبی KAN برای انجام طبقه‌بندی نهایی ارسال می‌شود. مدل ارائه‌شده بر روی مجموعه‌دادگان AUCDD-V2 و SFDDD که شامل ۹ کلاس از رفتارهای مرتبط با عدم تمرکز راننده و یک کلاس تمرکز راننده است، مورد ارزیابی قرار گرفت و به‌ترتیب به صحت ۹۵/۰۱٪ و ۹۵/۶۶٪ دست یافت. نتایج تجربی نشان می‌دهند که روش پیشنهادی در مقایسه با روش‌های موجود عملکرد بهتری داشته و می‌تواند به‌عنوان راهکاری مؤثر در افزایش ایمنی رانندگی مورد استفاده قرار گیرد.

کلید واژه ها

بینایی ماشین، پردازش تصویر، یادگیری عمیق، تشخیص خودکار عدم تمرکز راننده

۱- مقدمه

با افزایش تعداد خودروها در دهه‌های اخیر، ترافیک و تصادفات جاده‌ای به یکی از چالش‌های اساسی جوامع تبدیل شده است. براساس آمار سازمان بهداشت جهانی[۱]، سالانه بیش از ۱/۱۹ میلیون نفر در تصادفات جاده‌ای جان خود را از دست می‌دهند و این نوع حوادث همچنان از مهم‌ترین علل مرگ‌ومیر در گروه سنی ۵ تا ۲۹ سال به شمار می‌روند.

این مقاله در ۳ دی ۱۴۰۴ دریافت و در ۲۲ تیر ۱۴۰۵ پذیرفته شد

[۱] سمیرا کریمی چقاکبودی، کارشناس ارشد مهندسی کامپیوتر، گرایش هوش مصنوعی و رباتیکز، گروه مهندسی کامپیوتر، دانشگاه رازی

رایانامه: karimisamira1993@gmail.com

[۲] عبداله چاله‌چاله، دانشیار گروه مهندسی کامپیوتر، دانشگاه رازی (نویسنده مسئول)

رایانامه: chalechale@razi.ac.ir

بخش قابل توجهی از این تلفات در کشورهای با درآمد کم و متوسط رخ می‌دهد؛ کشورهایی که با وجود در اختیار داشتن تنها ۶۰ درصد وسایل نقلیه جهان، حدود ۹۲ درصد از مرگ‌ومیرهای جاده‌ای را به خود اختصاص داده‌اند [۱].

یکی از مهم‌ترین عامل‌های بروز این حوادث، کاهش سطح هوشیاری و تمرکز راننده است. مطالعات نشان می‌دهند که خواب‌آلودگی نقش چشمگیری در کاهش هوشیاری دارد و حدود ۲۰ تا ۳۰ درصد تصادفات جاده‌ای به این عامل مربوط می‌شود [۲]. از سوی دیگر، استفاده از تلفن همراه هنگام رانندگی نیز یکی از اصلی‌ترین علل کاهش تمرکز است. با توجه به گسترش وابستگی به تلفن همراه، بسیاری از رانندگان در حین رانندگی از این وسیله استفاده می‌کنند و این امر احتمال وقوع تصادف را تا چهار برابر افزایش می‌دهد [۳]. همچنین فعالیت‌هایی مانند خوردن و آشامیدن، مصرف مواد مخدر، استفاده از داروهای خواب‌آور و هر عاملی که توانایی تصمیم‌گیری یا واکنش به شرایط جاده را کاهش دهد، خطر تصادف را تشدید می‌کند.

[۱] World Health Organization (WHO)

برای کاهش چنین حوادثی، سامانه‌های هوشمند تشخیص عدم تمرکز راننده توسعه یافته‌اند که به‌طور کلی در سه گروه طبقه‌بندی می‌شوند. روش اول، روش‌های مبتنی‌بر وسیله نقلیه [۴] است که با تحلیل الگوهای حرکتی خودرو مانند انحراف مسیر یا تغییرات ناگهانی سرعت، میزان توجه راننده برآورد می‌شود؛ اما این روش‌ها به نوع خودرو، سبک رانندگی و شرایط جاده وابسته‌اند و هزینه اجرایی بالایی دارند. روش دوم، روش‌های مبتنی‌بر علائم فیزیولوژیکی [۵] است که از سیگنال‌هایی مانند ضربان قلب، حرکت چشم یا فعالیت مغزی برای ارزیابی هوشیاری استفاده می‌کنند. با وجود دقت بالا، نیاز به اتصال حسگر به بدن راننده کارایی این روش‌ها را در شرایط واقعی محدود کرده و حین رانندگی برای راننده مزاحمت ایجاد می‌کند.

در مقابل، روش سوم، روش‌های مبتنی‌بر وضعیت ظاهری راننده [۶] است که بر تحلیل داده‌های تصویری تکیه دارند. این روش‌ها از طریق پردازش تصاویر[۱] دریافتی از دوربین داخل خودرو، رفتارهایی مانند استفاده از تلفن همراه، خوردن و آشامیدن، صحبت کردن با دیگران و حرکات دست را به‌طور مؤثر شناسایی می‌کنند. این دسته از روش‌ها هزینه مناسب، پیاده‌سازی ساده، عدم وابستگی به نوع خودرو و مهم‌تر از همه عدم ایجاد مزاحمت برای راننده را به همراه دارند. همین مزایا باعث شده است که این روش‌ها به یکی از کاربردی‌ترین و مؤثرترین رویکردها در سامانه‌های کمک‌راننده تبدیل شوند.

با توجه به این مزایا، هدف این پژوهش نیز بهره‌گیری از روش‌های مبتنی‌بر وضعیت ظاهری راننده و توسعه یک سامانه کارآمد مبتنی‌بر بینایی ماشین[۲] و یادگیری عمیق[۳] است که بتواند با تحلیل تصاویر راننده، نشانه‌های حواس‌پرتی را شناسایی کرده و در صورت لزوم هشدارهای لازم را ارائه دهد.

با وجود پیشرفت‌های اخیر، بررسی مطالعات گذشته نشان می‌دهد که چالش‌هایی همچنان وجود دارند. یکی از مشکلات رایج، تقسیم‌بندی نادرست داده‌های آموزش و آزمون است. در بسیاری از پژوهش‌ها داده‌های مربوط به یک راننده به‌طور تصادفی در هر دو بخش آموزش و آزمون قرار گرفته‌اند [۷] که موجب ارزیابی غیرواقعی عملکرد مدل و دور شدن آن از شرایط واقعی می‌شود. در برخی مطالعات پس از استخراج ویژگی[۴] از طبقه‌بندهای ساده استفاده شده است ([۸] تا [۱۰].) اگرچه این مدل‌ها برای کاربردهای بلادرنگ و محیط‌های با منابع محدود مناسب هستند، اما به‌طور معمول توانایی محدودتری در تحلیل روابط غیرخطی پیچیده میان ویژگی‌ها دارند.در همین راستا، دراین پژوهش، با هدف رفع چالش‌های موجود در مطالعات پیشین، یک چارچوب مبتنی‌بر یادگیری عمیق ارائه شده است که ضمن استفاده از روش‌های مبتنی‌بر وضعیت ظاهری راننده، با طراحی سازوکار مناسب برای تقسیم‌بندی داده‌ها بر اساس راننده و بهره‌گیری از یک معماری ترکیبی برای استخراج ویژگی و طبقه‌بندی، امکان ارزیابی دقیق‌تر و نزدیک‌تر به شرایط واقعی را فراهم می‌کند.

در ادامه سازماندهی مقاله به این صورت است: در بخش دوم، مطالعات پیشین مرور می‌شود؛ بخش سوم روش‌شناسی پژوهش را شرح می‌دهد؛ بخش چهارم شامل آزمایش‌ها و نتایج است؛ و در نهایت در بخش پنجم جمع‌بندی و پیشنهادهایی برای کارهای آینده ارائه می‌شود.

۲- کارهای مرتبط

رانندگی یک فرایند پیچیده است که نیازمند توجه مستمر، پردازش سریع اطلاعات محیطی و واکنش به‌موقع در شرایط متغیر جاده است. کاهش سطح تمرکز راننده می‌تواند زمان واکنش او را افزایش داده و احتمال وقوع تصادف را به‌طور قابل‌توجهی بالا ببرد. با پیشرفت روش‌های مبتنی‌بر بینایی ماشین و یادگیری عمیق، پژوهش‌های متعددی به شناسایی عدم تمرکز راننده پرداخته‌اند. در این بخش، مطالعات پیشین بررسی می‌شوند که با بهره‌گیری از داده‌های تصویری و مدل‌های یادگیری عمیق، انواع رفتارهای حواس‌پرتی رانندگان را تحلیل و شناسایی کرده‌اند.

در مقاله [۸]، مدلی ترکیبی معرفی شده است که از چندین شبکه عصبی پیچشی[۵] شامل AlexNet [۱۱]، Inception-V3 [۱۲]، ResNet-50 [۱۳] و VGG-16 [۱۴] استفاده می‌کند. در این روش، هر شبکه به‌طور جداگانه بر روی داده‌های مختلف شامل تصاویر خام، تصاویر بخش‌بندی‌شده‌ی پوست، چهره و دست‌ها آموزش دیده و خروجی مدل‌ها به‌صورت وزنی ترکیب شده است. برای دستیابی به بهترین عملکرد، وزن‌های ترکیبی با استفاده از الگوریتم ژنتیک بهینه‌سازی شده‌اند.

در مقاله [۹]، مدل ADNet[۶] معرفی شده است که مبتنی‌بر معماری ResNet-18 بوده و با افزودن مکانیزم توجه، امکان تقویت ویژگی‌های مهم و حذف اطلاعات غیرضروری را فراهم می‌کند. برای بهبود عملکرد طبقه‌بندی نیز از تکنیک‌های افزایش داده[۷] استفاده شده و این امر منجر به بهبود دقت در شناسایی رفتارهای مختلف رانندگی شده است.

مدل ViT-DD[۸] [۱۰] یک روش نوآورانه مبتنی‌بر Vision Transformer [۱۵] است که به صورت چندوظیفه‌ای [۱۶] طراحی شده و قادر است همزمان حواس‌پرتی راننده و حالات احساسی او را تشخیص دهد. علاوه بر این، با بهره‌گیری از یادگیری شبه‌نظارت‌شده [۱۷]، مدل توانایی استفاده از داده‌هایی

---

۱ Image processing
۲ Computer Vision
۳ Deep Learning (DL)
۴ Feature Extraction
۵ Convolutional Neural Networks (CNN)
۶ Attention-based Deep Neural Network (ADNet)
۷ Data Augmentation
۸ Vision Transformer for Driver Distraction Detection (ViT-DD)

که برچسب احساسی ندارند را نیز دارد، که این امر به بهبود دقت و عملکرد کلی مدل کمک می‌کند.

در چارچوب پیشنهادی مقاله [۱۸]، از ترکیب معماری Inception-V3 و شبکه BiLSTM [۱۹] استفاده شده است. در این رویکرد، ابتدا ویژگی‌های مکانی تصاویر توسط Inception-V3 استخراج و سپس وابستگی‌های زمانی میان فریم‌ها توسط BiLSTM تحلیل می‌شود. ترکیب این دو بخش موجب افزایش دقت در تشخیص وضعیت‌های مختلف حواس‌پرتی راننده شده است.

در مقاله [۲۰]، به‌منظور بهبود دقت مدل‌های تشخیص حواس‌پرتی، مکانیزم توجه با الهام از نقشه‌های فعال‌سازی کلاس[۱] [۲۱] بازطراحی شده است. این روش سه محدودیت شامل محدودیت متمرکز، محدودیت متعامد و محدودیت بین‌نمونه‌ای معرفی می‌کند که موجب تمرکز بهتر مدل روی نواحی کلیدی تصویر و کاهش هم‌پوشانی توجه بین کلاس‌ها شده و بدون افزایش هزینه محاسباتی، دقت را به‌طور قابل‌ملاحظه‌ای افزایش می‌دهد.

در پژوهش [۲۲]، مدلی مبتنی‌بر EfficientNet-B0 [۲۳] همراه با مکانیزم توجه کانالی طراحی شده است. هدف این مدل افزایش دقت و کاهش هشدارهای اشتباه بوده و با استفاده از روش‌های پیش‌پردازش مانند افزایش داده، نرمال‌سازی و چرخش تصاویر، کیفیت داده‌های ورودی بهبود یافته است.

مدل CAT-CapsNet پیشنهادشده در مقاله [۲٤] ترکیبی از شبکه‌های عصبی پیچشی، مکانیزم توجه و شبکه‌های کپسولی [۲۵] است. در این مدل، ابتدا ویژگی‌های اولیه استخراج شده، سپس مکانیزم توجه برای تمرکز بر بخش‌های کلیدی به‌کار می‌رود و در نهایت شبکه‌های کپسولی با حفظ روابط مکانی بین ویژگی‌ها، درکی ساختاری‌تر از وضعیت راننده ارائه می‌کنند.

روش‌های سنتی تشخیص حواس‌پرتی راننده که بر پایه‌ی طبقه‌بندی عمل می‌کنند، در مواجهه با مجموعه‌داده‌های جدید معمولاً عملکرد ضعیفی از خود نشان می‌دهند. در مقاله‌ی [۲٦]، برای رفع این مشکل از مدل CLIP[۲] [۲۷] استفاده شده است؛ مدلی که با یادگیری ارتباط میان تصویر و متن عمل می‌کند. در این روش، به‌جای استفاده از برچسب‌های عددی، برای هر رفتار راننده یک جمله‌ی توصیفی کوتاه تولید می‌شود. سپس مدل می‌آموزد که تصویر مربوط به هر رفتار را به جمله‌ی متناظر آن نزدیک کند. در مرحله‌ی تشخیص نیز، تصویر جدید با تمامی جملات توصیفی مقایسه می‌شود و مناسب‌ترین گزینه انتخاب می‌گردد. نتایج نشان می‌دهد که CLIP در مقایسه با شبکه‌های سنتی، دقت را بیش از دو برابر افزایش داده است؛ با این حال، تشخیص برخی رفتارها، مانند رانندگی ایمن، همچنان چالش‌برانگیز باقی مانده است.

در پژوهش [۲۸]، رویکردی برای آموزش شبکه‌های عصبی که از تمرکز اشتباه مدل روی بخش‌های نامربوط تصویر جلوگیری می‌کند. در این روش، به جای استفاده از برچسب‌های قطعی که یک یا صفر هستند، یک سیستم نمره‌دهی چندسطحی جایگزین طبقه‌بند سنتی می‌شود. مهم‌ترین نوآوری این پژوهش، استفاده از ماتریس نظارتی پویا است که بر اساس توزیع گاوسی طراحی شده و در طول فرآیند آموزش، به صورت تصادفی در یک بازه مجاز نوسان می‌کند. این نوسان باعث می‌شود شبکه عصبی نتواند مسیرهای میانبر مانند توجه به پس‌زمینه یا جزئیات بی‌اهمیت خودرو را برای کاهش خطا پیدا کند و در عوض، به نواحی مرتبط با رفتار راننده تمرکز کند. نتیجه این فرایند، کاهش خطاهای تعمیم‌پذیری و بهبود تشخیص وضعیت‌های حواس‌پرتی در شرایط متنوع رانندگی است.

در مقاله [۲۹] مدلی ارائه شده که برای استخراج ویژگی‌های تصویری از معماری Swin Transformer [۳۰] استفاده می‌کند. این مدل برای اینکه کلاس‌های مختلف رفتار راننده بهتر از یکدیگر قابل تشخیص باشند، سه مرحله کلیدی انجام می‌دهد:ابتدا بردارهای مرکزی هر کلاس مقداردهی اولیه می‌شوند؛ سپس با استفاده از تابع زیان SC loss فاصله نمونه‌های یک کلاس از مرکز همان کلاس کاهش داده می‌شود؛ و در پایان، بردارهای مرکزی به‌گونه‌ای جابه‌جا می‌شوند که مرزبندی بین کلاس‌ها واضح‌تر شود. این سه مرحله در کنار هم باعث افزایش تفکیک‌پذیری و دقت مدل در تشخیص وضعیت‌های مختلف حواس‌پرتی راننده می‌شود.

پژوهش [۳۱] یک مدل دومرحله‌ای ارائه کرده است. در مرحله اول، شبکه DeepLabv3+ [۳۲] بخش راننده را از پس‌زمینه جدا می‌کند و در مرحله دوم، تصویر به یک شبکه پیچشی الهام‌گرفته از معماری VGG وارد می‌شود تا رفتار راننده تحلیل و طبقه‌بندی شود.

در پژوهش [۳۳]، یک رویکرد نوآورانه برای تشخیص حواس‌پرتی و خواب‌آلودگی راننده ارائه شده است که شامل طراحی سه شبکه عصبی پیچشی می‌باشد. یکی از شبکه‌ها با نام FDNN به‌طور کامل از ابتدا طراحی شده و دو شبکه دیگر مبتنی‌بر انتقال یادگیری از معماری‌های VGG16 و VGG19 با لایه‌های اضافی TLVGG هستند. این طراحی امکان پردازش بلادرنگ با دقت و سرعت بالا را فراهم می‌کند و نشان‌دهنده بهبود قابل توجهی نسبت به مدل‌های پیشین در توانایی تشخیص حالات حواس‌پرتی راننده است.

در مقاله [۳٤]، یک سیستم نظارت چهره راننده برای تشخیص خواب‌آلودگی و عدم تمرکز حواس طراحی شده است. این سیستم با پردازش تصاویر دوربین، نشانه‌های خستگی و کاهش هوشیاری راننده را از ناحیه چشم استخراج می‌کند و سپس با الگوریتم KNN میزان عدم تمرکز راننده تخمین زده می‌شود. نتایج آزمایش‌ها در محیط واقعی نشان می‌دهد که روش پیشنهادی دارای دقت بالایی است و امکان استفاده در سیستم‌های بلادرنگ را فراهم می‌کند.

---

[۱] Class Activation Maps

[۲] Contrastive Language–Image Pretraining (CLIP)

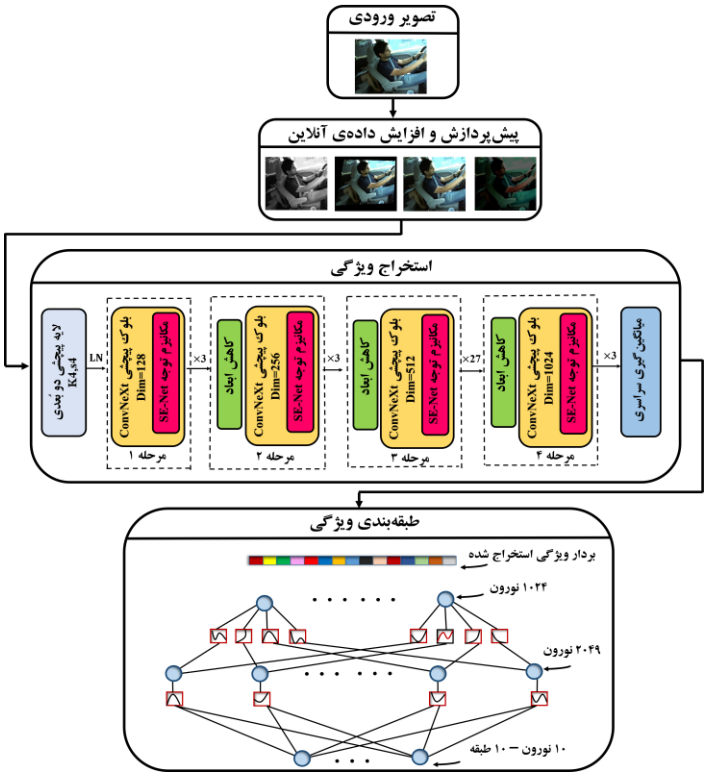
در پژوهش [۳۵]، یک سیستم تشخیص راننده حواس‌پرت با استفاده از شبکه‌های عصبی پیچشی طراحی شده است. این سیستم از معماری‌های شناخته‌شده‌ای مانند VGG16، ResNet-50، YOLO - v7، Faster R-CNN، RetinaNet و Detectron2 استفاده می‌کند و قابلیت پردازش در زمان واقعی دارد. همه مدل‌ها پیاده‌سازی و با یکدیگر مقایسه شده‌اند. نتایج نشان می‌دهد که الگوریتم‌های ارائه‌شده می‌توانند به کاهش تصادفات ناشی از حواس‌پرتی راننده کمک کنند.

در مجموع، پیشرفت‌های اخیر در حوزه بینایی ماشین نقش مهمی در توسعه سامانه‌های نظارت بر وضعیت راننده داشته است. هدف این سامانه‌ها کاهش خطرات جاده‌ای از طریق شناسایی شرایط خطرناک و ارائه هشدارهای مناسب به راننده است. استفاده از هشدارهای صوتی، بصری، لرزشی یا حتی فعال‌سازی سامانه‌های ایمنی خودرو می‌تواند احتمال وقوع تصادف را به‌طور چشمگیری کاهش دهد.

با توجه به عملکرد موفق روش‌های مبتنی‌بر تصویر در مطالعات پیشین، این پژوهش نیز همین رویکرد را دنبال می‌کند. در این کار، از تصاویر راننده برگرفته از یک پایگاه‌داده عمومی استفاده شده که پیشتر از طریق دوربین داخلی خودرو ثبت گردیده‌اند. این تصاویر سپس با یک چارچوب مبتنی‌بر یادگیری عمیق پردازش و تحلیل می‌شوند. هدف اصلی، شناسایی عوامل مرتبط با عدم تمرکز راننده بر اساس الگوهای رفتاری و بصری است تا سامانه‌ای دقیق و قابل‌اعتماد برای کاهش خطرات ناشی از حواس‌پرتی ارائه شود.

۳- روش شناسی

در این بخش، روش پیشنهادی پژوهش تشریح می‌شود که در قالب سه مرحله اصلی طراحی شده است. در مرحله اول، پیش‌پردازش و افزایش داده‌ها با هدف بهبود کیفیت تصاویر ورودی و افزایش تنوع نمونه‌های آموزشی انجام می‌گیرد تا داده‌های مناسب‌تری برای آموزش مدل فراهم شود و شبکه عصبی بتواند الگوهای مهم را بهتر بیاموزد. مرحله دوم به استخراج ویژگی‌ها اختصاص دارد که این فرآیند با بهره‌گیری از یک شبکه عصبی پیچشی مجهز به مکانیزم توجه صورت می‌پذیرد؛ شبکه پایه در این مرحله پس از بررسی و ارزیابی چندین شبکه عصبی پیچشی مختلف انتخاب شده است تا مؤثرترین و متمایزترین ویژگی‌ها از تصاویر استخراج گردد و اطلاعات کلیدی برای تصمیم‌گیری دقیق در اختیار طبقه‌بند قرار گیرد. در نهایت، در مرحله سوم، ویژگی‌های استخراج‌شده به‌منظور تعیین وضعیت تمرکز یا عدم تمرکز راننده مورد طبقه‌بندی قرار می‌گیرند و نتایج حاصل از این مرحله به‌عنوان خروجی نهایی روش پیشنهادی ارائه می‌شوند. نمای کلی مراحل پژوهش در شکل ۱ ارائه شده و در ادامه، هر یک از مراحل با جزئیات بیشتری شرح داده می‌شود.



شکل (۱) : روند کلی روش پیشنهادی

۳-۱- پیش‌پردازش و افزایش داده

پیش‌پردازش و افزایش داده‌ها از مراحل اساسی در آماده‌سازی داده‌های تصویری برای مدل‌های یادگیری عمیق محسوب می‌شوند. پیش‌پردازش با هدف بهبود کیفیت تصاویر، کاهش نویز، یکنواخت‌سازی ابعاد ورودی و نرمال‌سازی داده‌ها انجام می‌گیرد، در حالی‌که افزایش داده‌ها با هدف افزایش تنوع ظاهری نمونه‌های آموزشی و جلوگیری از بیش‌برازش[۱] مدل به‌کار گرفته می‌شود. ترکیب این دو فرایند نقش مهمی در افزایش توان تعمیم‌پذیری مدل و بهبود عملکرد آن در مواجهه با داده‌های جدید و شرایط واقعی ایفا می‌کند.

در این پژوهش برای آماده‌سازی داده‌ها، ابتدا اندازه تصاویر یکسان و پیکسل‌ها نرمال‌سازی شدند تا مقیاس ویژگی‌ها یکنواخت شود. سپس چندین روش افزایش داده به کار گرفته شد تا مدل نسبت به شرایط واقعی مقاوم شود و تعمیم‌پذیری آن افزایش یابد: تغییر روشنایی و رنگ برای کاهش اثر تغییر نور، تغییر پرسپکتیو و هندسه برای مقابله با زاویه‌های دید متفاوت، تبدیل تصادفی تصاویر به خاکستری برای تمرکز روی ویژگی‌های ساختاری، چرخش تصاویر، محوسازی و حذف تصادفی نواحی برای مقاوم‌سازی در برابر نویز و بخش‌های پوشیده شده.

فرایند افزایش داده‌ها به‌صورت آنلاین و در حین آموزش مدل انجام شد؛ به‌گونه‌ای که تصاویر اصلی ذخیره‌شده بدون هیچ تغییری باقی مانده و در هر مرحله از آموزش، نسخه‌ای تصادفی و متفاوت

---

[۱] Overfitting

از تصویر ورودی تولید و به مدل ارائه می‌شد[۳۶]. این رویکرد علاوه بر افزایش مؤثر تنوع داده‌های آموزشی، از افزایش حجم فیزیکی مجموعه‌داده جلوگیری کرده و در عین حال موجب بهبود پایداری و تعمیم‌پذیری مدل در برابر تغییرات نوری، هندسی و نویزهای محیطی می‌گردد. لازم به ذکر است که تعداد فریم‌ها برای هر کلاس و هر راننده در مجموعه دادگان AUCDD-V2 [۸] و SFDDD [۳۷] استفاده شده در این پژوهش متفاوت است و توزیع کلاس‌ها به صورت طبیعی حفظ شده و به‌طور مصنوعی متوازن نشده است. این رویکرد موجب می‌شود که مدل روی داده‌ای مشابه شرایط واقعی رانندگی آموزش دیده و نتایج آن واقع‌گرایانه و معتبر باقی بماند، و همچنین مقایسه عملکرد مدل با سایر پژوهش‌های پیشین که توزیع طبیعی داده‌ها را حفظ کرده‌اند منصفانه باشد. شکل ۲ نمونه‌هایی از تصاویر حاصل از اعمال تکنیک‌های افزایش داده را نشان می‌دهد.

شکل (۲) : نمونه‌هایی از تصاویر آموزشی پس از اعمال تکنیک‌های پیش‌پردازش و افزایش داده

۲-۳- استخراج ویژگی

در یادگیری عمیق، استخراج ویژگی یکی از مراحل کلیدی در پردازش و تحلیل داده‌های ورودی محسوب می‌شود. در شبکه‌های عصبی پیچشی، ویژگی‌های تصاویر از طریق چندین لایه پردازشی شناسایی شده و به بردارهای عددی تبدیل می‌شوند که اطلاعات مهم و معنادار تصویر را نشان می‌دهند. هدف این فرآیند، کاهش حجم داده‌ها و تمرکز بر ویژگی‌هایی است که نقش بیشتری در شناسایی و طبقه‌بندی ایفا می‌کنند.

در این مطالعه، ابتدا چند شبکه عصبی پیچشی مورد ارزیابی قرار گرفتند و یکی از آن‌ها به‌عنوان شبکه پایه برای استخراج ویژگی‌ها انتخاب شد؛ شبکه‌ای که بهترین عملکرد را در تشخیص رفتارهای عدم تمرکز راننده داشت. برای افزایش کیفیت و بهبود ویژگی‌های استخراج شده، این شبکه با مکانیزم توجه تجهیز شد تا تمرکز بیشتری روی ویژگی‌های مؤثر داشته باشد. در نهایت، ویژگی‌های به‌دست آمده آماده مرحله طبقه‌بندی شدند. جزئیات هر یک از این مراحل در ادامه ارائه شده است.

۱-۲-۳- انتخاب مدل پایه برای استخراج ویژگی

در این پژوهش، به‌منظور انتخاب مدل پایه برای استخراج ویژگی، از شبکه‌های عصبی پیچشی از پیش‌آموزش‌دیده استفاده شده است. این رویکرد به‌ویژه در شرایطی که حجم داده‌های آموزشی محدود است، با بهره‌گیری از انتقال دانش[۱] از مجموعه‌داده‌های بزرگ، موجب بهبود عملکرد مدل می‌شود. شبکه‌های پیش‌آموزش‌دیده قادرند ویژگی‌های عمومی و سطح‌بالا را از تصاویر استخراج کنند و در نتیجه دقت تشخیص الگوها افزایش یابد. از این‌رو، با توجه به محدود بودن داده‌های موجود در این پژوهش، استفاده از این شبکه‌ها مورد توجه قرار گرفته است.

به‌منظور شناسایی مناسب‌ترین مدل برای مجموعه‌داده موردنظر، ده شبکه عصبی پیچشی از پیش‌آموزش‌دیده، شامل SqueezeNet1-1 [۳۸]، MobileNet-v3-Large [۳۹]، EfficientNet-B0، GoogLeNet [۴۰]، ShuffleNet-v2-x2-0 [۴۱]، DenseNet-161 [۴۲]، ResNet-152، AlexNet، ConvNeXt-Base [۴۳] و VGG-19 مورد ارزیابی قرار گرفته‌اند. تمامی این مدل‌ها بر روی مجموعه‌داده ImageNet آموزش دیده‌اند که شامل بیش از ۱۴ میلیون تصویر در ۱۰۰۰ کلاس مختلف بوده و یکی از معتبرترین منابع در حوزه بینایی ماشین به شمار می‌رود [۴۴].

در مرحله اولیه، تمامی شبکه‌ها با ساختار پیش‌فرض و لایه طبقه‌بندی نهایی اصلی خود آموزش و ارزیابی شدند تا مناسب‌ترین معماری از نظر توان استخراج ویژگی برای تشخیص رفتارهای عدم تمرکز راننده مشخص شود. نتایج تجربی نشان داد که شبکه ConvNeXt-Base نسبت به سایر مدل‌ها عملکرد بهتری دارد، بنابراین به‌عنوان شبکه پایه برای استخراج ویژگی انتخاب شد.

این عملکرد برتر ناشی از ویژگی‌های طراحی این مدل است که باعث توانایی بالاتر شبکه در مدل‌سازی وابستگی‌های فضایی تصاویر و کسب نتایج بهتر در ارزیابی‌های تجربی شده است.

---

[۱] Transfer Learning

معماری ConvNeXt در سال ۲۰۲۲ با هدف بازنگری در اصول طراحی شبکه‌های عصبی پیچشی معرفی گردید. ایده اصلی این معماری آن بود که با به‌روزرسانی و اصلاح ساختار شبکه‌های پیچشی کلاسیک، می‌توان عملکرد آن‌ها را به سطح مدل‌های مبتنی‌بر Transformer [۴۵] نزدیک کرد. در سال‌های اخیر، معماری‌های مبتنی‌بر Transformer نظیر Vision Transformer و Swin Transformer با توانایی مدل‌سازی وابستگی‌های بلندمدت و ارتباطات غیرمحلی، پیشرفت قابل‌توجهی در بینایی ماشین ایجاد کرده‌اند. ConvNeXt با حفظ ماهیت پیچشی خود، از برخی اصول طراحی مدرن این مدل‌ها الهام گرفته و تلاش کرده است محدودیت‌های شبکه‌های پیچشی سنتی را کاهش دهد.

در طراحی ConvNeXt اصلاحاتی نظیر استفاده از هسته‌های[۱] پیچشی بزرگ‌تر (۷×۷)، جایگزینی نرمال‌سازی دسته‌ای[۲] با نرمال‌سازی لایه‌ای[۳]، بازطراحی ساختار بلوک‌های باقی‌مانده[۴]، به‌کارگیری تابع فعال‌سازی GELU[5] و بهینه‌سازی ساختار مرحله‌ای شبکه اعمال شده است. این تغییرات منجر به بهبود توانایی مدل در استخراج ویژگی‌های فضایی و افزایش کارایی آن شده‌اند.

در پژوهش ارائه‌شده در منبع [۴۳]، معماری ConvNeXt در سه وظیفه اصلی بینایی ماشین مورد ارزیابی قرار گرفته و نتایج قابل‌توجهی کسب کرده است. این مدل در طبقه‌بندی مجموعه داده ImageNet، با دستیابی به دقت ۸۵/۱ درصد، عملکردی برتر نسبت به Swin-B نشان داده است. همچنین در حوزه تشخیص اشیا بر روی مجموعه داده COCO، مدل مذکور با کسب امتیاز ۵۴/۰ در شاخص box AP، از مدل‌های Transformer پیشی گرفته و در قطعه‌بندی معنایی مجموعه داده ADE20K نیز عملکردی رقابتی از خود به نمایش گذاشته است. افزون بر دقت، این معماری ضمن حفظ سادگی شبکه‌های پیچشی سنتی، به سرعت پردازش بالاتری (۹۵/۷ تصویر در ثانیه) در مقایسه با Swin Transformer (۸۵/۱ تصویر در ثانیه) دست یافته است.

در این پژوهش، از نسخه ConvNeXt-Base که یکی از پیکربندی‌های متداول این معماری محسوب می‌شود، به‌عنوان مدل پایه استفاده شده است؛ شکل ۳ نیز این مدل را نمایش می‌دهد.

معماری ConvNeXt-Base شامل چهار مرحله اصلی است که هر مرحله از چند بلوک ConvNeXt تشکیل شده است. در ابتدای شبکه، یک لایه پیچشی با هسته ۴×۴ و گام[۶] ۴ برای کاهش ابعاد تصویر و استخراج ویژگی‌های اولیه به کار گرفته شده است. سپس ویژگی‌ها وارد چهار مرحله با تعداد کانال‌های ۱۲۸، ۲۵۶، ۵۱۲ و ۱۰۲۴ می‌شوند.

هر بلوک ConvNeXt شامل یک لایه پیچشی عمقی[۷] با هسته ۷×۷ است که ضمن کاهش تعداد پارامترها، توان مدل در استخراج ویژگی‌های فضایی گسترده را افزایش می‌دهد. در ادامه، نرمال‌سازی لایه‌ای، یک لایه پیچشی ۱×۱ و تابع فعال‌سازی GELU اعمال می‌شود. استفاده از مقیاس‌دهی لایه‌ای[۸] و حذف تصادفی مسیر[۹] نیز به پایداری فرآیند آموزش کمک می‌کند. علاوه بر این، میان مراحل متوالی از لایه‌های Downsample برای کاهش ابعاد استفاده شده است. لازم به ذکر است که در معماری اصلی ConvNeXt پس از، لایه میانگین‌گیری سراسری[۱۰] ، یک لایه کاملاً متصل[۱۱] برای انجام طبقه‌بندی نهایی در نظر گرفته شده است.

در مرحله انتخاب مدل پایه، تمامی شبکه‌ها با ساختار پیش‌فرض خود و بدون اعمال تغییر در لایه طبقه‌بندی نهایی مورد ارزیابی قرار گرفته‌اند. اصلاحات معماری شامل حذف لایه طبقه‌بندی، افزودن مکانیزم توجه و جایگزینی طبقه‌بند با شبکه عصبی KAN در مرحله طراحی معماری پیشنهادی انجام شده است.

۲-۲-۳- ادغام مکانیزم توجه SE-Net[۱۲]

شبکه‌های عصبی پیچشی به‌عنوان یکی از پرکاربردترین معماری‌های یادگیری عمیق، توانایی بالایی در استخراج ویژگی‌های محلی از تصاویر دارند. این شبکه‌ها با عملیات پیچشی، ویژگی‌های مهم نواحی کوچک تصویر را شناسایی می‌کنند؛ با این حال، تمرکز بر ویژگی‌های محلی باعث محدودیت در درک ساختار کلی تصویر و وابستگی‌های بلندمدت بین بخش‌های مختلف آن می‌شود. برای غلبه بر این محدودیت، مکانیزم‌های توجه پیشنهاد شده‌اند که امکان تمرکز بر نواحی مهم و ارزیابی اهمیت کانال‌ها را فراهم می‌کنند.

در این پژوهش، مکانیزم توجه SE-Net [۴۷] به شبکه پایه ConvNeXt-Base افزوده شده است تا فرآیند استخراج ویژگی بهینه‌تر شود و دقت مدل در تصمیم‌گیری نهایی افزایش یابد.

این معماری توسط Hu و همکاران در سال ۲۰۱۸ معرفی شد و برای تقویت ویژگی‌های کانالی در شبکه‌های عصبی پیچشی و بهبود عملکرد در وظایف طبقه‌بندی تصاویر به‌کار می‌رود. SE-Net با یادگیری اهمیت نسبی هر کانال، امکان تمرکز شبکه بر ویژگی‌های کلیدی را فراهم می‌کند و نقش کانال‌های کم‌اهمیت را کاهش می‌دهد.

پیاده‌سازی SE-Net در این تحقیق پس از لایه‌های پیچشی عمقی در تمام بلاک‌های ConvNeXt-Base انجام شده است.

---

[۱] Kernel
[۲] Batch Normalization
[۳] Layer Norm
[۴] Residual blocks
[5] Gaussian Error Linear Unit (GELU)
[۶] Stride
[۷] Depthwise Convolution
[۸] Layer scale
[۹] Drop path
[۱۰] Global Average Pooling (GAP)
[۱۱] Dense Layer
[۱۲] Squeeze-and-Excitation Networks (SE-Net)

ساختار کلی این پیاده‌سازی در شکل ۴ نمایش داده شده است. دلیل این انتخاب آن است که لایه‌های پیچشی عمقی هر کانال را به‌صورت مستقل پردازش می‌کنند، اما توانایی محدودی در ارزیابی اهمیت نسبی کانال‌ها و تعامل بین ویژگی‌ها دارند. افزودن SE-Net به شبکه باعث می‌شود اهمیت نسبی کانال‌ها به‌صورت پویا یاد گرفته شود. این کار موجب استخراج ویژگی‌های مؤثرتر و بهبود دقت مدل می‌گردد.

مکانیزم SE-Net شامل سه مرحله اصلی است: فشرده‌سازی[۱]، برانگیختگی[۲] و مقیاس‌دهی[۳] این مراحل و فرمول‌های مرتبط با آن‌ها اقتباس شده از معماری SE-Net معرفی شده توسط Hu و همکاران هستند و در ادامه برای روشن شدن فرآیند هر مرحله، توضیحات تفصیلی آن‌ها ارائه شده است [۴۷].

الف) فشرده سازی: در این مرحله، ابعاد مکانی ویژگی‌ها کاهش یافته و اطلاعات هر کانال به یک مقدار عددی خلاصه می‌شود.

فرض کنید ورودی شبکه نقشه ویژگی[۴] سه‌بُعدی X با ابعاد $H' \times W' \times X'$ باشد. ابتدا با یک عملیات پیچشی (F_{tr}) ویژگی‌های سطح بالاتر استخراج شده و نقشه ویژگی $F_{tr} : X \to U$ با ابعاد $H \times W \times C$ به‌دست می‌آید. سپس، میانگین‌گیری سراسری روی هر کانال انجام می‌شود تا بردار فشرده Z_C حاصل شود:

$$Z_c = F_{sq}(U_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i, j) \qquad (۱)$$

که در آن $u_c(i, j)$ مقدار ویژگی در مختصات (i, j) کانال C است. خروجی این مرحله یک بردار یک‌بُعدی Z_C با ابعاد $1 \times 1 \times C$ خواهد بود که پایه تولید وزن‌های توجه در مرحله بعدی می‌باشد.

ب) برانگیختگی: بردار فشرده‌شده Z وارد شبکه‌ای کوچک شامل دو لایه کاملاً متصل می‌شود تا اهمیت هر کانال به‌صورت پویا یاد گرفته شود.

$$S = F_{ex}(z, W) = \sigma(g(z, W)) = \sigma(W_2 \delta(W_1 z)) \qquad (۲)$$

که در آن:

- $W_1 \in \mathbb{R}^{c/r \times c}, W_2 \in \mathbb{R}^{c \times c/r}$ ماتریس‌های وزنی لایه‌های کاملاً متصل هستند که در مرحله برانگیختگی برای یادگیری وابستگی‌های میان کانال‌ها به‌کار می‌روند. این دو ماتریس یک گلوگاه محاسباتی ایجاد می‌کنند، به‌گونه‌ای که ابتدا W_2 ابعاد بردار ویژگی را کاهش می‌دهد و سپس W_1 آن را به مقدار اولیه بازمی‌گرداند.
- r نرخ کاهش بُعد[۵] است، که به‌صورت ضمنی از طریق ابعاد ماتریس‌های وزنی $W_1 \in \mathbb{R}^{c/r \times c}, W_2 \in \mathbb{R}^{c \times c/r}$ تعیین می‌شود و مشخص می‌کند تعداد نورون‌های لایه میانی تا چه میزان کاهش یابد. این پارامتر به‌طور مستقیم در رابطه ظاهر نمی‌شود، بلکه از طریق تعیین ابعاد وزن‌ها در معماری اعمال می‌گردد. معمولاً $r = 16$ انتخاب می‌شود، به‌این‌معنا که تعداد نورون‌ها در این مرحله به 1/16 تعداد کانال‌ها کاهش یابد و سپس مجدداً به مقدار اولیه بازگردد، که تعادل مناسبی بین کاهش پیچیدگی و حفظ اطلاعات مهم برقرار کند. مقادیر کمتر از ۱۶ باعث افزایش بار محاسباتی و پیچیدگی مدل می‌شوند، در حالی که مقادیر بیشتر از ۱۶ ممکن است منجر به از دست رفتن اطلاعات حیاتی و کاهش دقت شبکه شود، تعیین این مقدار براساس آزمایش‌های تجربی انجام شده به‌دست آمده است [۴۷].
- δ تابع فعال‌سازی Relu و σ تابع فعال‌سازیSigmoid است.
- خروجی S شدت و اهمیت نسبی هر کانال را مشخص می‌کند. هرچه مقدار S به ۱ نزدیک‌تر باشد، نقش آن کانال در تصمیم‌گیری نهایی شبکه پررنگ‌تر بوده و ویژگی‌های آن تقویت می‌شوند؛ در مقابل، مقادیر نزدیک به صفر موجب تضعیف کانال و کاهش تأثیر آن در خروجی مدل می‌گردند.

پ) مقیاس‌دهی: در این مرحله، وزن‌های تولیدشده S_c بر روی نقشه ویژگی‌های ورودی u_c اعمال می‌شوند تا ویژگی‌های کلیدی تقویت و ویژگی‌های کم‌اهمیت تضعیف شوند:

$$\tilde{X}_c = F_{scale}(u_c, s_c) = s_c \cdot u_c \qquad (۳)$$

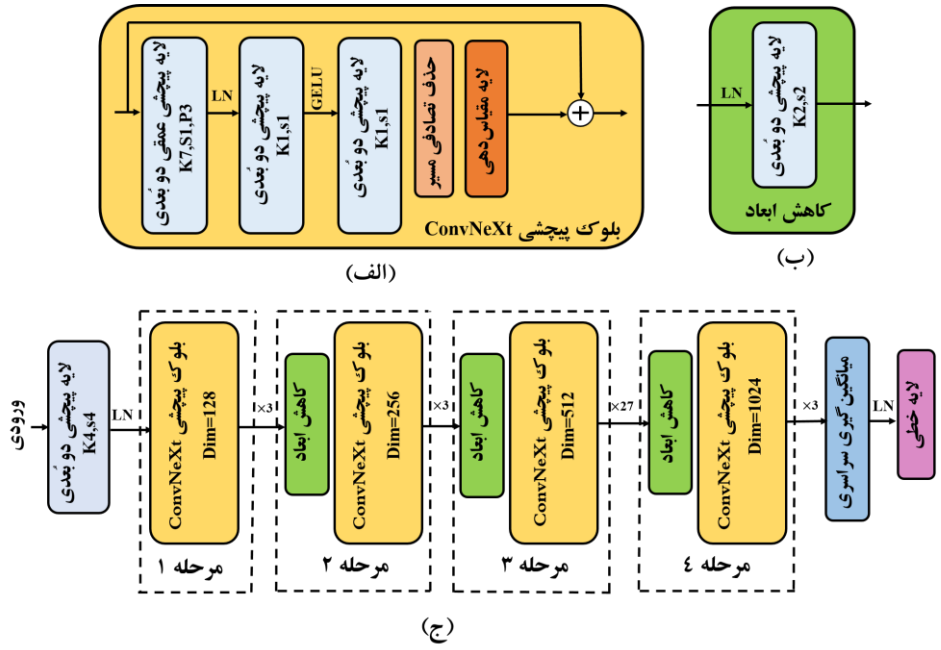
- که در آن $(\cdot)$ ضرب عنصر به عنصر است. نتیجه این فرآیند، نسخه‌ای بهبود یافته از نقشه ویژگی‌های اولیه است که وارد مراحل بعدی شبکه می‌شود و در نهایت موجب بهبود دقت و پایداری مدل در وظیفه طبقه‌بندی تصاویر مجموعه‌داده موردنظر می‌گردد.

۳-۲-۳- استخراج ویژگی با شبکه ConvNeXt-Base

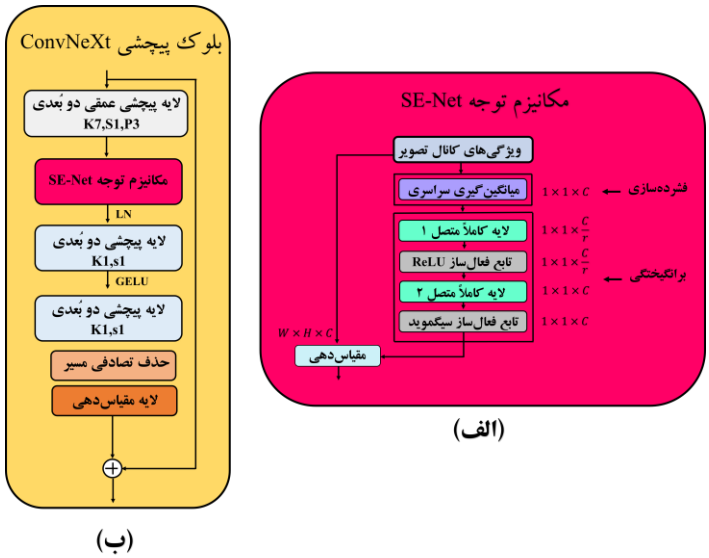
در این پژوهش، برای استخراج ویژگی از داده‌های تصویری، از شبکه عصبی پیچشی ConvNeXt-Base مجهز به مکانیزم توجه SE-Net استفاده شده است.

---

۱ Squeeze
۲ Excitation
۳ Scaling
۴ Feature Map
۵ Reduction ratio



شکل (۳) : معماری شبکه ConvNeXt-Base [۴۶] شامل؛ (الف) بلوک‌های پیچشی ConvNeXt، (ب) بخش کاهش ابعاد، و (ج) ساختار کلی شبکه.



شکل (۴): (الف) ساختار مکانیزم توجه SE-Net، (ب) ادغام مکانیزم توجه SE-Net به یک بلاک از شبکه عصبی پیچشی ConvNeXt-Base

این مکانیزم با یادگیری خودکار، نواحی مهم تصویر را برجسته کرده و تأثیر ویژگی‌های کم‌اهمیت را کاهش می‌دهد، که منجر به استخراج دقیق‌تر و معنادارتر ویژگی‌ها می‌شود.

به منظور بهره‌گیری از دانش از پیش‌آموخته‌شده، روش تنظیم دقیق[۱] [۴۸] روی مدل اعمال شد. وزن‌های اولیه شبکه که پیش‌تر بر روی مجموعه‌داده ImageNet آموزش دیده‌اند، به عنوان نقطه شروع در نظر گرفته شده و سپس تمامی لایه‌های شبکه با داده‌های مجموعه‌داده موردنظر بازآموزی شده‌اند. این رویکرد امکان بهره‌گیری همزمان از دانش عمومی شبکه و سازگاری آن با داده‌های جدید را فراهم می‌کند.

فرآیند استخراج ویژگی به این صورت انجام می‌شود: ابتدا تصویر ورودی پس از پیش‌پردازش و افزایش داده، وارد لایه ابتدایی ConvNeXt-Base شده و ویژگی‌های اولیه آن استخراج می‌شود. سپس داده به بلاک‌های اصلی شبکه شامل چندین لایه پیچشی عمیق منتقل می‌شود. در این مرحله، مکانیزم توجه SE-Net فعال شده و بر روی نواحی مهم تصویر تمرکز می‌کند تا ویژگی‌های کلیدی تقویت و ویژگی‌های کم‌اهمیت کاهش یابند.

ویژگی‌های حاصل از بلاک‌های پردازشی پس از میانگین‌گیری سراسری فشرده شده و با عبور از لایه نرمال‌سازی، مقیاس و توزیع آن‌ها استاندارد می‌شود. در نهایت، ویژگی‌ها به یک بردار یک‌بُعدی با ابعاد ۱۰۲۴ تبدیل می‌شوند و از طریق عملیات مسطح‌سازی[۲] برای مرحله‌ی طبقه‌بندی آماده می‌گردند. این بردار نمایانگر کلیه اطلاعات مهم تصویر است و نسخه‌ای فشرده و قابل پردازش از ویژگی‌های استخراج‌شده در اختیار مرحله‌ی طبقه‌بندی قرار می‌گیرد.

۳-۳- طبقه‌بندی ویژگی با شبکه عصبی KAN

در سال ۲۰۲۴ شبکه‌ای نوین با عنوان KAN[۳] معرفی شد که ساختار آن با الهام از قضیه نمایش کولموگروف آرنولد شکل گرفته

---

[۱] Fine-Tuning

[۲] Flatten

[۳] Kolmogorov-Arnold Networks (KAN)

است [۴۹]. برخلاف پرسپترون‌های چندلایه[1] که از وزن‌های خطی ثابت و توابع فعال‌سازی از پیش تعیین‌شده استفاده می‌کنند، در KAN توابع فعال‌سازی به صورت تک‌متغیره و قابل یادگیری روی یال‌ها تعریف شده‌اند. این طراحی باعث افزایش دقت در تقریب توابع پیچیده و بهبود تفسیرپذیری نسبت به MLP می‌شود. با توجه به این قضیه که هر تابع چندمتغیره را می‌توان با ترکیبی از توابع تک‌متغیره بازنویسی کرد، KAN به جای وزن‌های خطی از توابع تک‌متغیره قابل آموزش استفاده می‌کند، که آن را به جایگزینی توانمند برای MLP تبدیل می‌کند. در شبکه‌های MLP، تابع موردنظر با ترکیبی از وزن‌های خطی و یک تابع فعال‌سازی ثابت تقریب زده می‌شود:

$$f(X) \approx \sum_{i=1}^{N(\epsilon)} a_i \sigma(W_i \cdot X + b_i) \qquad (۴)$$

که در آن X ورودی شبکه، W_i وزن‌های قابل یادگیری، b_i بایاس، σ تابع فعال‌سازی ثابت (مانند ReLU) و a_i وزن‌های خروجی هستند. همچنین، $N(\epsilon)$ حداقل تعداد نورون‌های لایه‌ی پنهان برای دستیابی به خطایی کوچک‌تر از ϵ است. در KAN این ساختار با جایگزینی وزن‌ها توسط توابع تک‌متغیره تغییر یافته است [۴۹]:

$$f(X) = \sum_{q=1}^{2n+1} \phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right) \qquad (۵)$$

در این مدل، توابع $\phi_{q,p}$ و ϕ_q تک‌متغیره و قابل یادگیری هستند. به این معنی که هر ویژگی به جای ضرب در یک وزن ثابت، ابتدا از یک تابع تک‌متغیره عبور می‌کند و ضرایب آن توسط الگوریتم پس‌انتشار خطا[2] به‌روزرسانی می‌شود. این طراحی موجب افزایش انعطاف‌پذیری، دقت بیشتر در تقریب تابع و بهبود تفسیرپذیری نسبت به MLP می‌شود. تابع تک‌متغیره استفاده‌شده در KAN ترکیبی از تابع پایه SiLU[3] و یک تابع Spline [۵۰] است و به‌صورت زیر نمایش داده می‌شود:

$$\phi(x) = w_b b(x) + w_s spline(x) \qquad (۶)$$

در این رابطه، تابع $b(x)$ که در اینجا به‌صورت تابع SiLU تعریف شده است، نقش مشابه اتصالات باقی‌مانده[4] در شبکه‌های عمیق دارد و با حفظ اطلاعات پایه‌ای ورودی، پایداری یادگیری را افزایش می‌دهد. این تابع به‌شکل زیر تعریف می‌شود:

$$b(x) = silu(x) = x / (1 + e^{-x}) \qquad (۷)$$

تابع SiLU پیوسته و مشتق‌پذیر است و در برخی مسائل عملکرد بهتری نسبت به ReLU ارائه می‌دهد. تابع Spline نیز قطعه‌قطعه‌ای و هموار بوده و برای تقریب روابط غیرخطی و پیچیده مورد استفاده قرار می‌گیرد. دو پارامتر کلیدی، نقش مهمی در شکل‌دهی Spline ایفا می‌کنند:

- تعداد گره‌ها (G): نقاط مشخصی که دامنه تابع را به زیرناحیه‌های کوچک تقسیم می‌کنند و در هر ناحیه یک چندجمله‌ای مستقل تعریف می‌شود. مقدار G تعیین‌کننده انعطاف تابع و میزان تطابق آن با داده‌ها است؛ مقادیر کوچک تابعی ساده و روان ایجاد می‌کنند و مقادیر بزرگ انعطاف بیشتر اما با ریسک بیش‌برازش همراه هستند.
- مرتبه (K): درجه چندجمله‌ای هر بخش که همواری و پیچیدگی تابع را کنترل می‌کند. در شبکه‌های KAN معمولاً از Spline مرتبه ۳ استفاده می‌شود.

ساختار کلی Spline به صورت زیر است:

$$spline(x) = \sum_i c_i B_i(x) \qquad (۸)$$

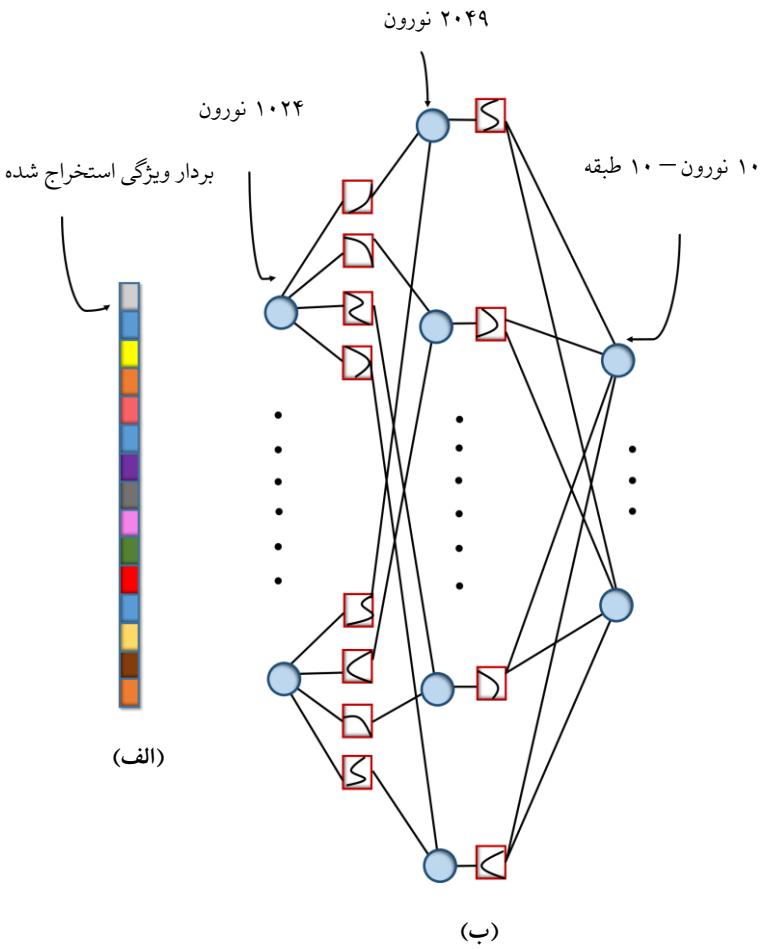
که در آن c_i ضرایب قابل یادگیری و $B_i(x)$ توابع پایه B-Spline هستند. ضرایب W_b و W_s نیز میزان تأثیر تابع پایه و Spline را تعیین می‌کنند. مقداردهی اولیه آنها به‌گونه‌ای انجام می‌شود که پایداری گرادیان در طول آموزش حفظ شود از جمله با استفاده از مقدارسازی وزن‌ها به روش ژاویر.

طبق فرم اصلی قضیه کولموگروف آرنولد، در معماری KAN معمولاً از n نورون در لایه ورودی و $2n+1$ نورون در لایه میانی استفاده می‌شود. این انتخاب کمک می‌کند تا شبکه بتواند تابع هدف را دقیق‌تر تقریب بزند و روابط غیرخطی پیچیده را بهتر یاد بگیرد. با این حال، ساختار شبکه انعطاف‌پذیر است و می‌توان آن را براساس نیاز مسئله تغییر داد. در این پژوهش نیز از KAN به‌عنوان طبقه‌بند نهایی استفاده شد.

پس از استخراج ویژگی‌ها توسط شبکه ConvNeXt-Base مجهز به مکانیزم توجه، بردار ویژگی ۱۰۲۴ تایی حاصل از لایه میانگین‌گیری سراسری، به‌عنوان ورودی شبکه KAN در نظر گرفته شد. بر این اساس، لایه ورودی شامل ۱۰۲۴ نورون، لایه میانی ۲۰۴۹ نورون (مطابق رابطه $2n+1$) و لایه خروجی ۱۰ نورون (تعداد کلاس‌ها) طراحی شد. همچنین تعداد گره‌های Spline (پارامتر G) با روش تجربی برابر ۲۰ انتخاب شد تا دقت و تعمیم‌پذیری مدل به شکل مناسبی حفظ شود. در شکل ۵ نمایی از شبکه عصبی KAN پیاده‌سازی شده در این پژوهش نشان داده شده است.

---

[1] Multilayer Perceptron (MLP)
[2] Backpropagation of Error
[3] Sigmoid Linear Unit
[4] Residual connections



شکل (۵): نمای کلی شبکه عصبی KAN پیاده‌سازی‌شده [۴۹]؛ (الف) بردار ویژگی استخراج‌شده، (ب) ساختار شبکه عصبی KAN.

به‌منظور ارائه‌ی نمایی ساختاریافته و الگوریتمی از مراحل اجرای روش پیشنهادی، در ادامه شبه‌کد کلی چارچوب ارائه‌شده در شکل ٦ آورده شده است. این شبه‌کد، روند کامل پردازش از مرحله دریافت تصاویر ورودی تا استخراج ویژگی، اعمال مکانیزم توجه و انجام طبقه‌بندی نهایی توسط شبکه عصبی KAN را به‌صورت گام‌به‌گام تشریح می‌کند و دیدی منسجم از فرآیند آموزش و ارزیابی مدل فراهم می‌سازد.

۴- آزمایش و تحلیل نتایج

در این بخش، مجموعه‌داده مورد استفاده، رویکردهای آموزشی، تنظیمات پیاده‌سازی و معیارهای ارزیابی معرفی شده و نتایج حاصل از مدل‌های از پیش‌آموزش‌دیده با روش پیشنهادی مقایسه و تحلیل می‌گردد.

```
Input:
    Image datasets {I1, I2, ..., In}
    Number of epochs E
    Batch size B
    Learning rate η

Output:
    Optimal model parameters θ*
    Predicted class labels C = {c1, c2, …, ck}
    Classification metrics: Accuracy, Precision, Recall, F1-score
------------------------------------------------------------
Phase 1: Data Preparation
------------------------------------------------------------
Load image frames
Apply preprocessing operations:
  - Resizing
  - Normalization
  - Data augmentation
Construct mini-batches with batch size B
------------------------------------------------------------
Phase 2: Feature extraction using ConvNeXt-Base network equipped
with SE-Net attention mechanism
------------------------------------------------------------
Load ConvNeXt-Base backbone pre-trained on ImageNet

For each stage s in ConvNeXt:
    For each block b in stage s:
        For each depthwise convolution layer dw in block b:
            Extract intermediate feature map F_dw ∈ ℝ^{C×H×W}
            Compute channel-wise attention weights using SE-Net module:
                w = σ(FC(GAP(F_dw)))
            Recalibrate feature map:
                F_dw_hat = F_dw ⊙ w

Apply global average pooling and flatten features:
    f ∈ ℝ^{1024}
------------------------------------------------------------
Phase 3: Classification Using Kolmogorov–Arnold Network
------------------------------------------------------------
Initialize KAN layers:
    KAN1: ℝ^{1024} → ℝ^{2049}
    KAN2: ℝ^{2049} → ℝ^{k}
Construct hybrid architecture:
    ConvNeXt-SE → KAN1 → KAN2
------------------------------------------------------------
Phase 4: Training and Optimization
------------------------------------------------------------
Initialize loss function, optimizer, and early-stopping strategy
Train the model for E epochs
Select optimal parameters θ* based on validation performance
------------------------------------------------------------
Phase 5: Evaluation
------------------------------------------------------------
Evaluate θ* on test data
Compute Accuracy, Precision, Recall, and F1-score
Generate confusion matrix

Return:
    θ* and classification results
```

شکل (۶): شبه‌کد از روش پیشنهادی

۱-۴- معرفی مجموعه دادگان و رویکردهای آموزشی و ارزیابی

در این پژوهش، برای آموزش و ارزیابی مدل پیشنهادی از مجموعه‌داده AUCDD-V2 [۸] استفاده شده است. این مجموعه‌داده یکی از منابع عمومی معتبر در حوزه تشخیص حالات عدم تمرکز راننده به شمار می‌رود و داده‌های آن بر اساس راننده تفکیک شده‌اند؛ بدین معنا که رانندگانی که در مجموعه آموزش حضور دارند، در مجموعه آزمون هیچ تصویری ندارند. رعایت این اصل موجب می‌شود مدل به‌جای تکیه بر ویژگی‌های ظاهری افراد، الگوهای رفتاری کلی رانندگی را بیاموزد. در نتیجه، توانایی تعمیم مدل به رانندگان جدید افزایش یافته و ارزیابی آن به شرایط واقعی نزدیک‌تر می‌شود. مطالعات پیشین [۱۰] و [۲۰] نیز به‌طور صریح بر اهمیت این نوع تقسیم‌بندی برای دستیابی به ارزیابی واقع‌گرایانه تأکید کرده‌اند.

با این حال، در برخی پژوهش‌ها داده‌های این مجموعه به‌صورت تصادفی تقسیم شده‌اند که در نتیجه، بخشی از داده‌های آموزش در مجموعه آزمون تکرار می‌شود و بدین ترتیب نتایج ارزیابی قابل اتکا نبوده و با شرایط عملی رانندگی سازگار نیست. افزون بر این، در بعضی مطالعات ([۸] تا [۱۰]) مشخص نیست که از مجموعه اعتبارسنجی استفاده شده است یا خیر، در حالی‌که در برخی دیگر [۱۸] و [۲۰] از اعتبارسنجی بهره گرفته شده است. در این پژوهش به تمام این موارد توجه شده و دو رویکرد آموزشی و ارزیابی مجزا طراحی گردیده است.

- رویکرد اول: تقسیم‌بندی دو بخشی (آموزش و آزمون).

در این رویکرد، داده‌ها به دو مجموعه آموزش و آزمون تفکیک شده‌اند؛ به‌گونه‌ای که تصاویر مربوط به هر راننده صرفاً در یکی از این دو مجموعه قرار دارند و هیچ‌گونه همپوشانی میان رانندگان وجود ندارد. این تقسیم‌بندی امکان ارزیابی دقیق توانایی مدل در شناسایی رفتارهای عدم تمرکز رانندگان جدید را فراهم می‌سازد.

- رویکرد دوم: تقسیم‌بندی سه بخشی (آموزش، اعتبارسنجی و آزمون).

در این رویکرد، داده‌ها به سه مجموعه مجزا تقسیم شده‌اند و هر راننده تنها در یکی از این مجموعه‌ها قرار گرفته است. این ساختار علاوه بر ارزیابی نهایی مدل، امکان بهینه‌سازی و تنظیم پارامترها را در مرحله اعتبارسنجی فراهم می‌کند. رانندگان مجموعه اعتبارسنجی به‌صورت تصادفی و بر اساس افراد، از بین داده‌های آموزشی انتخاب شده‌اند.

مجموعه‌داده AUCDD-V2 شامل تصاویر استخراج‌شده از ویدئوهایی است که با مشارکت ۴۴ نفر، (۲۹ مرد و ۱۵ زن) از ۷ کشور مختلف، جمع‌آوری شده است. پس از پردازش، این ویدئوها به فریم‌هایی با وضوح ۱۰۸۰×۱۹۲۰ و ۴۸۰×۶۴۰ پیکسل تبدیل شده‌اند و در مجموع ۱۴۴۷۸ فریم تصویری به‌دست آمده است. این تصاویر شامل ۱۰ کلاس (C0 تا C9) هستند که رفتارهای مختلف رانندگی را نمایش می‌دهند. توزیع فریم‌ها در هر کلاس برای هر دو رویکرد در جدول ۱ و ۲ ارائه شده است.

جدول (۱): توزیع تعداد فریم‌های رویکرد اول برای مجموعه داده AUCDD-V2

آزمون	آموزش	تعداد فریم‌ها
۶	۳۸	تعداد رانندگان
۳۴۶	۲۶۴۰	رانندگی ایمن
۲۱۳	۱۵۰۵	ارسال پیامک با دست راست
۱۹۴	۱۰۶۲	استفاده از موبایل با دست راست
۱۸۰	۹۴۴	ارسال پیامک با دست چپ
۱۷۰	۱۱۵۰	استفاده از موبایل با دست چپ
۱۷۰	۹۵۳	تنظیم رادیو
۱۴۳	۹۳۳	نوشیدن
۱۴۳	۸۹۱	برداشتن چیزی از صندلی عقب
۱۴۶	۸۹۸	مرتب کردن مو و آرایش
۲۱۸	۱۵۷۹	صحبت با سرنشین

جدول (۲): توزیع تعداد فریم‌های رویکرد دوم برای مجموعه داده AUCDD-V2

اعتبار سنجی	آزمون	آموزش	تعداد فریم‌ها
۶	۶	۳۲	تعداد رانندگان
۵۵۴	۳۴۶	۲۰۸۶	رانندگی ایمن
۱۹۲	۲۱۳	۱۳۱۳	ارسال پیامک با دست راست
۱۵۹	۱۹۴	۹۰۳	استفاده از موبایل با دست راست
۱۳۷	۱۸۰	۸۰۷	ارسال پیامک با دست چپ
۱۶۷	۱۷۰	۹۸۳	استفاده از موبایل با دست چپ
۱۴۴	۱۷۰	۸۰۹	تنظیم رادیو
۱۳۶	۱۴۳	۷۹۷	نوشیدن
۱۱۱	۱۴۳	۷۸۰	برداشتن چیزی از صندلی عقب
۱۴۳	۱۴۶	۷۵۵	مرتب کردن مو و آرایش
۲۲۸	۲۱۸	۱۳۵۱	صحبت با سرنشین

علاوه بر مجموعه‌داده AUCDD-V2، به منظور ارزیابی قابلیت تعمیم و جلوگیری از وابستگی مدل به یک پایگاه داده خاص، آزمایش‌های تکمیلی بر روی مجموعه‌داده عمومی SFDDD [۳۷] نیز انجام شده است. مجموعه‌داده SFDDD یکی از مجموعه‌داده‌های پرکاربرد در زمینه تشخیص عدم تمرکز راننده محسوب می‌شود که به‌عنوان یک چالش در پلتفرم کگل معرفی گردیده است. این مجموعه‌داده شامل ۲۲۴۲۴ فریم می‌باشد که با مشارکت ۲۶ راننده (۱۴ مرد و ۱۲ زن) با وضوح ۴۸۰×۶۴۰ جمع‌آوری گردیده است.

مجموعه‌داده SFDDD همانند مجموعه‌داده AUCDD-V2 شامل ۱۰ کلاس مختلف از رفتارهای رانندگی می‌باشد که به ترتیب از C0 تا C9 برچسب‌گذاری شده‌اند. همچنین داده‌های این مجموعه بر اساس افراد، برای هر یک از بخش‌های آموزش، آزمون

و اعتبارسنجی تفکیک شده‌اند. توزیع فریم‌ها در هر کلاس، برای هر دو رویکرد، در جداول ۳ و ۴ گزارش شده است.

جدول (۳): توزیع تعداد فریم‌های رویکرد اول برای مجموعه داده SFDDD

آزمون	آموزش	تعداد فریم‌ها
۶	۲۰	تعداد رانندگان
۵۶۴	۱۹۲۵	رانندگی ایمن
۴۷۴	۱۷۹۳	ارسال پیامک با دست راست
۵۰۱	۱۸۱۶	استفاده از موبایل با دست راست
۵۳۶	۱۸۱۰	ارسال پیامک با دست چپ
۵۳۱	۱۷۹۵	استفاده از موبایل با دست چپ
۵۲۶	۱۷۸۶	تنظیم رادیو
۵۳۰	۱۷۹۵	نوشیدن
۴۸۰	۱۵۲۲	برداشتن چیزی از صندلی عقب
۴۰۰	۱۵۱۱	مرتب کردن مو و آرایش
۴۳۶	۱۶۹۳	صحبت با سرنشین

جدول (۴): توزیع تعداد فریم‌های رویکرد دوم برای مجموعه داده SFDDD

اعتبار سنجی	آزمون	آموزش	تعداد فریم‌ها
۶	۵	۱۵	تعداد رانندگان
۵۶۴	۴۰۸	۱۵۱۷	رانندگی ایمن
۴۷۴	۴۲۱	۱۳۷۲	ارسال پیامک با دست راست
۵۰۱	۴۵۶	۱۳۶۰	استفاده از موبایل با دست راست
۵۳۶	۴۳۹	۱۳۷۱	ارسال پیامک با دست چپ
۵۳۱	۴۴۶	۱۳۴۹	استفاده از موبایل با دست چپ
۵۲۶	۴۵۴	۱۳۳۲	تنظیم رادیو
۵۳۰	۴۲۹	۱۳۶۶	نوشیدن
۴۸۰	۳۵۴	۱۱۶۸	برداشتن چیزی از صندلی عقب
۴۰۰	۳۱۵	۱۱۹۶	مرتب کردن مو و آرایش
۴۳۶	۳۶۳	۱۳۳۰	صحبت با سرنشین

شایان ذکر است که در مجموعه داده AUCDDD-V2، تقسیم‌بندی داده‌ها به مجموعه‌های آموزش و آزمون به صورت پیش‌فرض توسط گردآورندگان مجموعه داده انجام شده بود. در مقابل، برای مجموعه داده SFDDD، این تقسیم‌بندی به‌صورت دستی در این پژوهش صورت گرفت. نکته مهم این است که در هر دو مجموعه داده، فرآیند افزایش داده پس از انجام تقسیم‌بندی داده‌ها و منحصراً بر روی داده‌های آموزشی اعمال شد. داده‌های اعتبارسنجی و آزمون در هیچ مرحله‌ای از فرآیند افزایش داده یا آموزش مدل مورد استفاده قرار نگرفتند.

شکل ۷ نمونه‌هایی از تصاویر این مجموعه‌دادگان را نمایش می‌دهد. رعایت این ساختار و به‌کارگیری دو رویکرد متفاوت، امکان تحلیل دقیق‌تر و مقایسه عملکرد مدل در شرایط گوناگون را فراهم می‌کند.

شکل (۷): نمونه تصاویر تمامی کلاس‌ها از دو مجموعه‌داده‌ی AUCDD-V2 و SFDDD

۴-۲- جزئیات پیاده‌سازی و تنظیمات آموزش

در این پژوهش، مدل با استفاده از زبان برنامه‌نویسی Python و چارچوب PyTorch پیاده‌سازی شد. این چارچوب به دلیل انعطاف‌پذیری بالا، قابلیت پردازش موازی و پشتیبانی از پردازنده‌های گرافیکی[۱]، بستر مناسبی برای توسعه و آموزش مدل‌های یادگیری عمیق فراهم می‌سازد. آزمایش‌های اولیه در محیط برخط Kaggle انجام گرفت که شامل یک پردازنده مرکزی[۲] چهار هسته‌ای، حدود ۳۱ گیگابایت حافظه موقت[۳] و واحد پردازش گرافیکی NVIDIA Tesla T4 با حدود ۱۴ گیگابایت حافظه اختصاصی بود.

با این حال، به‌منظور انجام آموزش و ارزیابی نهایی، از یک سامانه محاسباتی مستقل با توان سخت‌افزاری بالاتر استفاده شد که به پردازنده Intel Core i9-7900X ده هسته‌ای، ۱۲۸ گیگابایت حافظه موقت و کارت گرافیک NVIDIA GeForce RTX 3080 با ۱۰ گیگابایت حافظه اختصاصی مجهز است. همچنین برای پیاده‌سازی و اجرای کدها، از محیط Visual Studio Code به‌عنوان بستر برنامه‌نویسی استفاده شد.

---

۱ Graphics Processing Unit (GPU)

۲ Central Processing Unit (CPU)

۳ Random Access Memory

در بخش تنظیمات آموزشی، در رویکرد اول که داده‌ها به دو بخش آموزش و آزمون تقسیم شدند، در طول فرآیند آموزش از تکنیک زمان‌بندی نرخ یادگیری StepLR با پارامترهای step_size=3 و gamma=0.5 استفاده شد. در این روش، نرخ یادگیری پس از هر سه دوره آموزش با ضریب ۰/۵ کاهش می‌یابد تا مدل در مراحل ابتدایی با سرعت بیشتری به سمت نقاط بهینه همگرا شده و سپس با کاهش تدریجی نرخ یادگیری، فرآیند بهینه‌سازی با دقت بالاتری ادامه پیدا کند.

در رویکرد دوم، که داده‌ها به سه بخش آموزش، اعتبارسنجی و آزمون تقسیم شدند. در پایان هر دوره آموزش، صحت مدل بر روی مجموعه‌ی اعتبارسنجی محاسبه شده و در صورت بهبود عملکرد، وزن‌های مدل ذخیره می‌شوند. ارزیابی نهایی با استفاده از وزن‌هایی انجام می‌شود که بالاترین صحت را بر روی داده‌های اعتبارسنجی داشته‌اند.

به‌منظور تنظیم نرخ یادگیری در این رویکرد، از زمان‌بندی ReduceLROnPlateau استفاده شده است. بر اساس این راهبرد، اگر صحت مجموعه‌ی اعتبارسنجی طی دو دوره آموزش متوالی بهبود نیابد patience=2، نرخ یادگیری با ضریب factor=0.1 کاهش پیدا می‌کند. علاوه بر این، برای جلوگیری از ادامه‌ی آموزش در شرایط عدم بهبود و کاهش بیش‌برازش، از روش توقف زودهنگام[۱] استفاده شده است؛ به‌گونه‌ای که اگر صحت اعتبارسنجی طی پنج دوره آموزش پیاپی بهبود نیابد (patience=5)، فرآیند آموزش متوقف می‌شود.

به‌کارگیری راهبردهای فوق با هدف کاهش بیش‌برازش و انتخاب وزن‌های بهینه مبتنی‌بر عملکرد مجموعه‌ی اعتبارسنجی انجام شده است.

همچنین ابرپارامترهای مدل، شامل تعداد تکرارهای آموزش، اندازه دسته، اندازه تصویر ورودی، نرخ یادگیری و روش بهینه‌سازی، مشابه مطالعات قبلی بر اساس نیازهای معماری ارائه‌شده و آزمایش‌های اولیه روی مجموعه اعتبارسنجی بررسی شدند. برای هر ابرپارامتر، مقادیری که بهترین عملکرد، بیشترین پایداری و همگرایی مناسب مدل را فراهم کردند، انتخاب شدند. مدل با این مقادیر آموزش داده شد و عملکرد نهایی آن روی داده‌های آزمون ارزیابی شد تا دقت و اعتبار نتایج تضمین شود. مقادیر نهایی این ابرپارامترها در جدول ۵ ارائه شده‌اند.

جدول (۵): ابرپارامترهای مورد استفاده در آموزش مدل

ابرپارامتر	مقدار
تعداد تکرارهای آموزش	۲۰ تکرار
اندازه دسته‌های آموزش	۸
روش بهینه سازی	Adam
نرخ یادگیری	lr=1e-4
اندازه تصویر ورودی	۳۸۴×۳۸۴

[۱] Early Stopping

۳-۴- معیارهای ارزیابی

در این پژوهش، عملکرد مدل‌های تشخیص عدم تمرکز راننده با استفاده از معیارهای رایج طبقه‌بندی ارزیابی شده است. معیار اصلی ارزیابی، صحت[۲] بوده و در کنار آن، از ماتریس درهم‌ریختگی[۳] و معیارهای دقت[۴]، حساسیت[۵] و امتیاز F1 برای تحلیل دقیق‌تر عملکرد مدل استفاده شده است [۵۱].

ماتریس درهم‌ریختگی عملکرد مدل را با استفاده از چهار مؤلفه مثبت واقعی[۶]، منفی واقعی[۷]، مثبت کاذب[۸] و منفی کاذب[۹] نمایش می‌دهد. در ادامه، معیارهای به‌کار رفته تشریح می‌شوند:

- صحت: این معیار نشان می‌دهد چه نسبتی از پیش‌بینی‌های مدل صحیح بوده‌اند. به عبارت دیگر، صحت توانایی کلی مدل را در تشخیص درست تمامی نمونه‌ها، چه مثبت و چه منفی، اندازه‌گیری می‌کند و نشان می‌دهد مدل تا چه حد توانسته رفتار یا حالت‌های رانندگی را به درستی شناسایی کند.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (۹)$$

- حساسیت: حساسیت میزان توانایی مدل در شناسایی صحیح نمونه‌های مثبت را نشان می‌دهد. به عبارت دیگر، این معیار درصد نمونه‌های واقعی مثبت که مدل توانسته به درستی شناسایی کند را اندازه‌گیری می‌کند و اهمیت آن در کاربردهای ایمنی رانندگی بسیار بالاست، زیرا از دست دادن نمونه‌های مثبت می‌تواند منجر به خطرات جدی شود.

$$Sensitivity / \mathrm{Re}\,call = \frac{TP}{TP + FN} \qquad (۱۰)$$

- دقت: دقت بیان می‌کند که از میان نمونه‌هایی که مدل به عنوان مثبت پیش‌بینی کرده است، چه نسبتی واقعاً مثبت بوده‌اند. این معیار نشان‌دهنده توانایی مدل در کاهش هشدارهای نادرست است و اهمیت آن زمانی مشخص می‌شود که پیش‌بینی‌های نادرست ممکن است منجر به اقدامات غیرضروری یا اشتباه شود.

$$\mathrm{Pr}\,ecision = \frac{TP}{TP + FP} \qquad (۱۱)$$

[۲] Accuracy
[۳] Confusion matrix
[۴] Precision
[۵] Sensitivity
[۶] True Positive (TP)
[۷] True Negative (TN)
[۸] False Positive (FP)
[۹] False Negative (FN)

- امتیاز F1: میانگین هارمونیک دقت و حساسیت را محاسبه می‌کند و عملکرد متوازن مدل در شناسایی صحیح حالات مثبت و کاهش خطاها را ارزیابی می‌کند.

$$F1_{Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (۱۲)$$

این معیارها به‌طور جامع عملکرد مدل را ارزیابی می‌کنند و معیار صحت به‌عنوان شاخص اصلی برای مقایسه مدل‌ها در نظر گرفته شده است.

۴-۴- نتایج حاصل از شبکه‌های از پیش آموزش دیده

با توجه به محدودیت نمونه‌ها در مجموعه‌داده‌ی AUCDD-V2 و SFDDD از نظر تنوع افراد و شرایط ظاهری، استفاده از شبکه‌های عصبی از پیش آموزش‌دیده به‌منظور بهبود عملکرد مدل‌ها ضروری بود. بدین منظور، عملکرد ده شبکه‌ی آموزش‌دیده بر روی ImageNet مورد ارزیابی قرار گرفت. برای انطباق بهتر مدل‌ها با داده‌ها، تمامی لایه‌ها به‌طور کامل بر روی هر دو مجموعه‌داده به صورت مجزا بازآموزی شدند و فرآیند آموزش در دو رویکرد آموزش-آزمون و آموزش-آزمون-اعتبارسنجی انجام گرفت.

نتایج حاصل از این دو رویکرد که در جداول ۶، ۷، ۸ و ۹ ارائه شده‌اند، بیانگر آن است که در میان معماری‌های مورد بررسی، شبکه‌ی ConvNeXt-Base عملکردی برتر نسبت به سایر مدل‌ها از خود نشان داده است. این شبکه با دستیابی به بیشترین مقادیر در شاخص‌های ارزیابی متعدد و ارائه‌ی نتایجی پایدارتر، به‌عنوان مدل پایه جهت استخراج ویژگی انتخاب گردید. عملکرد بهتر این مدل به دلیل توانمندی بالای آن در یادگیری الگوهای پیچیده و دقت مناسب در شناسایی ویژگی‌های نهفته در داده‌ها بوده است.

جدول (۶): نتایج شبکه‌های از پیش آموزش دیده با رویکرد اول روی مجموعه داده AUCDD-V2

مدل	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
SqueezeNet1-1	۶۱/۶۶	۶۳/۴۵	۶۱/۷۲	۶۲/۸۵
Mobilenet-v3-large	۶۴/۷۰	۶۶/۸۲	۶۴/۶۳	۶۴/۶۸
EfficientNet-B0	۷۹/۸۸	۸۱/۴۰	۷۹/۵۹	۷۹/۸۰
GoogLeNet	۶۹/۵۰	۷۰/۰۶	۶۹/۵۱	۶۹/۴۸
Shufflenet-v2-x2-0	۷۲/۳۳	۷۳/۱۱	۷۲/۲۸	۷۲/۳۳
DenseNet-161	۸۱/۶۸	۸۱/۷۸	۸۱/۶۶	۸۱/۶۷
ResNet-152	۷۸/۲۹	۸۰/۵۴	۷۸/۴۲	۷۸/۳۰
Alexnet	۵۹/۶۶	۶۲/۷۴	۵۹/۶۳	۵۹/۶۶
ConvNeXt-Base	**۸۵/۵۰**	**۸۶/۹۰**	**۸۵/۴۸**	**۸۵/۴۴**
VGG-19	۸۰/۳۳	۸۲/۶۷	۸۰/۲۹	۸۰/۳۴

جدول (۷): نتایج شبکه‌های از پیش آموزش دیده با رویکرد دوم روی مجموعه داده AUCDD-V2

مدل	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
SqueezeNet1-1	۵۹/۲۶	۶۰/۴۱	۵۹/۳۵	۵۹/۲۳
Mobilenet-v3-large	۶۳/۷۵	۶۵/۶۱	۶۳/۶۹	۶۰/۷۰
EfficientNet-B0	۷۸/۶۳	۸۰/۰۲	۷۸/۴۵	۷۸/۸۵
GoogLeNet	۶۸/۲۵	۶۹/۴۰	۶۸/۳۶	۶۷/۳۹
Shufflenet-v2-x2-0	۷۰/۴۹	۷۲/۶۱	۷۰/۶۸	۷۰/۵۵
DenseNet-161	۷۹/۲۷	۸۰/۳۲	۷۹/۲۵	۷۸/۲۳
ResNet-152	۷۷/۵۴	۷۹/۵۸	۷۷/۵۲	۷۷/۵۱
Alexnet	۵۶/۶۶	۵۸/۱۹	۵۶/۱۷	۵۶/۲۶
ConvNeXt-Base	**۸۴/۸۹**	**۸۵/۴۹**	**۸۴/۷۷**	**۸۴/۶۵**
VGG-19	۷۹/۴۳	۸۰/۳۲	۷۹/۴۰	۷۹/۴۴

جدول (۸): نتایج شبکه‌های از پیش آموزش دیده با رویکرد اول روی مجموعه داده SFDDD

مدل	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
SqueezeNet1-1	۶۲/۸۷	۶۴/۲۹	۶۲/۸۶	۶۲/۵۳
Mobilenet-v3-large	۶۶/۵۰	۶۸/۶۵	۶۶/۴۸	۶۶/۷۵
EfficientNet-B0	۸۲/۹۹	۸۴/۰۳	۸۲/۹۲	۸۲/۹۸
GoogLeNet	۷۰/۵۲	۷۳/۹۰	۷۰/۵۱	۷۰/۴۶
Shufflenet-v2-x2-0	۷۴/۸۰	۷۵/۷۲	۷۴/۷۹	۷۴/۶۰
DenseNet-161	۸۲/۶۳	۸۳/۵۰	۸۲/۴۸	۸۲/۳۹
ResNet-152	۷۹/۹۰	۸۰/۱۸	۷۹/۸۷	۷۹/۲۷
Alexnet	۶۲/۷۵	۶۴/۲۲	۶۲/۷۲	۶۲/۷۴
ConvNeXt-Base	**۸۶/۹۹**	**۸۷/۵۰**	**۸۶/۹۶**	**۸۶/۹۹**
VGG-19	۸۲/۷۹	۸۲/۹۳	۸۲/۵۷	۸۲/۶۶

جدول (۹): نتایج شبکه‌های از پیش آموزش دیده با رویکرد دوم روی مجموعه داده SFDDD

مدل	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
SqueezeNet1-1	۶۱/۸۸	۶۳/۷۳	۶۱/۶۹	۶۱/۸۰
Mobilenet-v3-large	۶۵/۵۵	۶۶/۶۱	۶۵/۴۵	۶۵/۳۴
EfficientNet-B0	۸۱/۴۶	۸۳/۳۴	۸۱/۴۰	۸۱/۴۲
GoogLeNet	۶۸/۳۳	۷۱/۵۷	۶۸/۲۰	۶۸/۲۹
Shufflenet-v2-x2-0	۷۳/۷۲	۷۴/۴۱	۷۳/۶۹	۷۳/۷۰
DenseNet-161	۸۱/۴۹	۸۲/۱۴	۸۱/۳۷	۸۱/۴۵
ResNet-152	۷۷/۵۶	۷۹/۲۷	۷۷/۴۵	۷۷/۳۸
Alexnet	۶۰/۴۵	۶۱/۳۹	۶۰/۳۲	۶۰/۴۴
ConvNeXt-Base	**۸۵/۶۷**	**۸۶/۲۵**	**۸۵/۶۰**	**۸۵/۶۴**
VGG-19	۸۰/۴۷	۸۱/۵۵	۸۰/۴۳	۸۱/۴۴

۵-۴- نتایج حاصل از روش پیشنهادی

نتایج ارزیابی روش پیشنهادی روی مجموعه‌دادگان AUCDD-V2 و SFDDD با چهار معیار صحت، دقت، حساسیت و F1 در جداول ۱۰ ارائه شده‌اند.

جدول (۱۰): نتایج کلی روش پیشنهادی با رویکرد اول و دوم

مجموعه داده	رویکرد	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
AUCDD-V2	رویکرد اول	۹۵/۰۱	۹۵/۱۹	۹۴/۷۴	۹۴/۸۳
	رویکرد دوم	۹۴/۳۳	۹۴/۲۵	۹۴/۵۰	۹۴/۲۶
SFDDD	رویکرد اول	۹۵/۶۶	۹۵/۷۲	۹۵/۳۵	۹۵/۲۱
	رویکرد دوم	۹۴/۱۳	۹۴/۰۳	۹۳/۷۱	۹۳/۴۳

با توجه به ناموزون بودن توزیع نمونه‌ها در کلاس‌های مختلف مجموعه‌دادگان AUCDD-V2 و SFDDD، اتکا به معیار صحت به‌تنهایی نمی‌تواند بیانگر عملکرد واقعی مدل در تمامی کلاس‌ها باشد. ازاین‌رو، علاوه بر صحت، معیارهای دقت، حساسیت و امتیاز F1 نیز به‌صورت میانگین کلان[۱] محاسبه و گزارش شده‌اند تا ارزیابی عملکرد مدل با درنظرگرفتن سهم برابر تمامی کلاس‌ها انجام شود. در این رویکرد، عملکرد مدل ابتدا برای هر کلاس به‌صورت مستقل محاسبه شده و سپس میانگین ساده‌ی آن‌ها به‌عنوان شاخص نهایی گزارش می‌شود. نتایج ارائه‌شده در جدول ۱۰ نشان می‌دهد روش پیشنهادی در معیارهای دقت، حساسیت و F1 نیز عملکرد بهتری داشته است.

همچنین بر اساس نتایج ارائه‌شده در جدول ۱۱ و ۱۲ مدل پیشنهادی این پژوهش در مقایسه با روش‌های پیشین عملکرد برتری بر روی مجموعه‌دادگان AUCDD-V2 و SFDDD نشان داده است. به‌منظور مقایسه‌ی منصفانه با مطالعات پیشین، تمرکز اصلی بر معیار «صحت» قرار داده شد؛ چراکه اغلب پژوهش‌های پیشین صرفاً این معیار را گزارش کرده‌اند. اگرچه ممکن است جزئیات مربوط به پیش‌پردازش داده‌ها، تنظیمات پیاده‌سازی و معماری مدل‌ها در مطالعات مختلف متفاوت باشد، مقایسه بر مبنای مجموعه‌داده یکسان و معیار ارزیابی مشترک انجام شده است. لازم به ذکر است که این مقایسه با جدیدترین مقالاتی انجام شده که شرط (تقسیم‌بندی بر اساس راننده) را رعایت کرده‌اند. اگرچه مقالات جدیدتری، مانند [۵۲] به نتایج بهتری دست یافته‌اند، اما به‌دلیل عدم رعایت تقسیم‌بندی بر اساس راننده، در این مقایسه لحاظ نشده‌اند.

در رویکرد اول، مدل پیشنهادی به صحت ۹۵/۰۱٪ و در رویکرد دوم به صحت ۹۴/۳۳٪ روی مجموعه داده AUCDDD-V2 دست یافته است. همچنین، مدل پیشنهادی به صحت ۹۵/۶۶٪ در رویکرد اول و ۹۴/۱۳٪ در رویکرد دوم روی مجموعه داده SFDDD رسیده است که نسبت به روش‌های پیشین، بهبود قابل توجهی را نشان می‌دهد. این نتایج حاکی از آن است که ساختار ترکیبی ارائه‌شده، افزون بر دستیابی به صحت بالا، از پایداری و کارایی مناسبی در طبقه‌بندی رفتار رانندگان برخوردار است.

نمودار زیان مربوط به داده‌های آموزشی در رویکرد اول، در شکل الف-۸ و الف-۹ ارائه شده است که کاهش تدریجی و یکنواخت مقدار خطا در طول فرآیند آموزش را نشان می‌دهد و بیانگر روند همگرایی مناسب مدل است. همچنین، در بخش ب-۸ و ب-۹ تغییرات زیان را در طول مراحل آموزش با در نظر گرفتن روش توقف زودهنگام در رویکرد دوم را نمایش می‌دهد. در این فرآیند، آموزش در نقطه‌ای متوقف شده است که عملکرد مدل بر روی داده‌های اعتبارسنجی همچنان در سطح مطلوبی قرار داشته و از افت کارایی جلوگیری شده است.

به‌کارگیری روش توقف زودهنگام موجب حفظ تعادل میان فرآیند یادگیری و جلوگیری از بیش‌برازش، بهبود قابلیت تعمیم‌پذیری مدل و در عین حال کاهش هزینه‌های محاسباتی شده است.

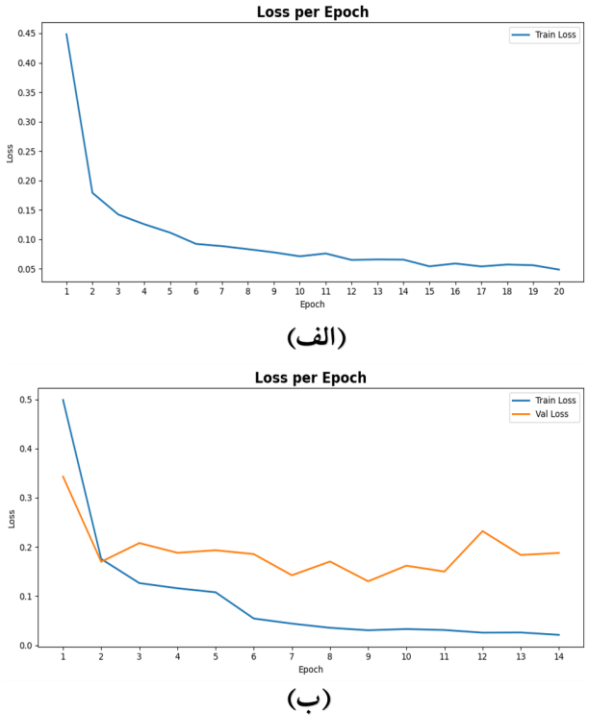
جدول (۱۱): مقایسه روش پیشنهادی با کارهای پیشین با مجموعه داده AUCDD-V2

مرجع	مدل	تقسیم‌بندی داده بر اساس راننده			صحت (%)	سال انتشار
		Val	Test	Train		
[۸]	GA-Weighted Ensemble	۰	۶	۳۸	۹۰/۰۶	۲۰۱۹
[۱۸]	C-SLSTM	۸	۶	۳۰	۹۲/۷۰	۲۰۲۰
[۹]	ADNet	۰	۶	۳۸	۹۰/۲۲	۲۰۲۲
[۱۰]	Vit-DD	۰	۶	۳۸	۹۳/۵۹	۲۰۲٤
[۲۰]	CA mechanism	۶	۶	۳۲	۹۳/۷۷	۲۰۲٤
رویکرد اول	**ConvNeXt-Base+ SE-Net +KAN**	**۰**	**۶**	**۳۸**	**۹۵/۰۱**	**اکنون**
رویکرد دوم	**ConvNeXt-Base+ SE-Net +KAN**	**۶**	**۶**	**۳۲**	**۹۴/۳۳**	**اکنون**

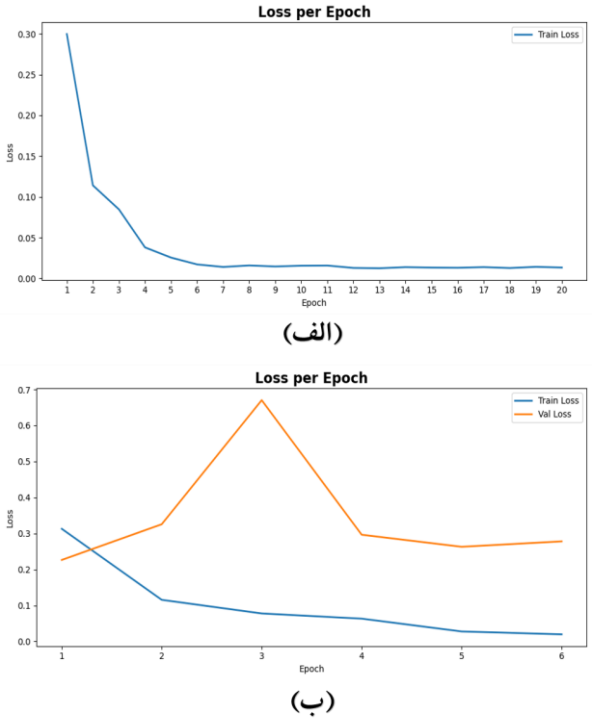
جدول (۱۲): مقایسه روش پیشنهادی با کارهای پیشین با مجموعه داده SFDDD

مرجع	مدل	تقسیم‌بندی داده بر اساس راننده			صحت (%)	سال انتشار
		Val	Test	Train		
[۱۰]	Vit-DD	۰	۶	۲۰	۹۲/۵۱	۲۰۲٤
[۲۰]	CA mechanism	۵	۵	۱۵	۹۳/۵۳	۲۰۲٤
رویکرد اول	**ConvNeXt-Base+ SE-Net +KAN**	**۰**	**۶**	**۲۰**	**۹۵/۶۶**	**اکنون**
رویکرد دوم	**ConvNeXt-Base+ SE-Net +KAN**	**۵**	**۵**	**۱۵**	**۹۴/۱۳**	**اکنون**

[۱] Macro-Average



شکل (۸): نمودار تغییرات مقدار زیان در فرآیند آموزش برای مجموعه‌داده AUCDD-V2؛ (الف) رویکرد اول و (ب) رویکرد دوم.

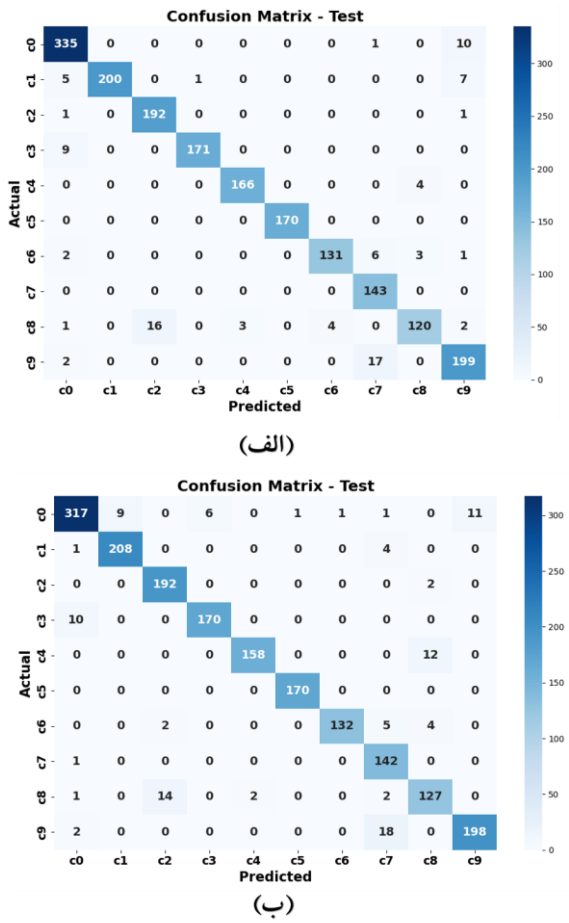


شکل (۹): نمودار تغییرات مقدار زیان در فرآیند آموزش برای مجموعه‌داده SFDDD؛ (الف) رویکرد اول و (ب) رویکرد دوم.

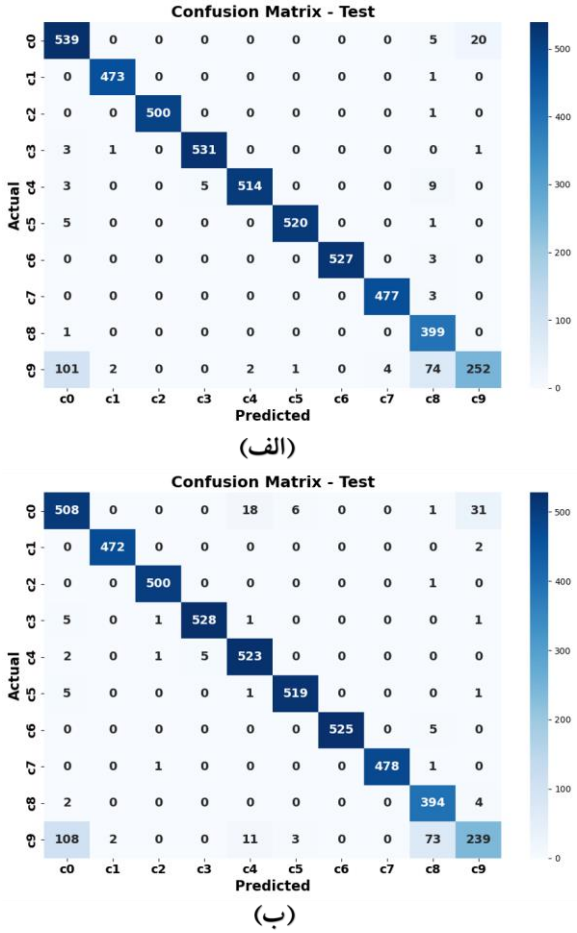
افزون بر این، به‌منظور تحلیل دقیق‌تر نحوه‌ی تفکیک کلاس‌ها و درک بهتر نقاط ضعف و قوت مدل، ماتریس‌های سردرگمی ارائه شده‌اند که امکان بررسی عملکرد مدل در هر کلاس به‌صورت جداگانه را فراهم می‌کنند. همان‌طور که در شکل ۱۰ مشاهده می‌شود، بیشترین میزان خطا در تشخیص کلاس‌هایی رخ داده است که الگوهای حرکتی مشابهی دارند؛ برای مثال، کلاس C8 (مرتب کردن مو و آرایش) با کلاس C2 (استفاده از موبایل با دست راست) و کلاس C9 (صحبت با سرنشین) با کلاس C7 (برداشتن چیزی از صندلی عقب) اشتباه گرفته شده‌اند. در مقابل، برخی از کلاس‌ها مانند C5 («تنظیم رادیو»)، C7 و C2 تقریباً بدون هیچ خطایی شناسایی شده‌اند. این موضوع بیانگر توانایی بالای مدل در تشخیص رفتارهایی است که الگوهای حرکتی متمایز و مشخصی دارند و نشان می‌دهد که مدل قادر است ویژگی‌های کلیدی و متمایز رفتارها را به‌خوبی استخراج کند.

همچنین در شکل ۱۱ مشاهده می‌شود که کلاس C9 علاوه بر اشتباه با کلاس C8، گاهی با کلاس C0 («رانندگی ایمن») نیز اشتباه گرفته شده است. به‌طور مشابه، کلاس C0 نیز گاهی با کلاس C9 همپوشانی دارد و این موضوع نشان‌دهنده پیچیدگی تفکیک برخی رفتارهای نزدیک از نظر حرکتی است.

به‌طور کلی، نتایج به‌دست آمده نشان می‌دهند که مدل در تمایز رفتارهایی با حرکات منحصر به فرد و مشخص عملکرد بسیار مطلوبی دارد؛ اما در تفکیک رفتارهایی که از نظر حرکتی شباهت بالایی با یکدیگر دارند، با چالش مواجه است. این چالش عمدتاً در کلاس‌هایی مشاهده می‌شود که الگوی حرکتی مشابه یا همپوشان دارند.



شکل (۱۰): ماتریس درهم‌ریختگی برای مجموعه‌داده AUCDD-V2؛ (الف) رویکرد اول و (ب) رویکرد دوم.



شکل (۱۱): ماتریس درهم‌ریختگی برای مجموعه‌داده SFDDD؛ (الف) رویکرد اول و (ب) رویکرد دوم.

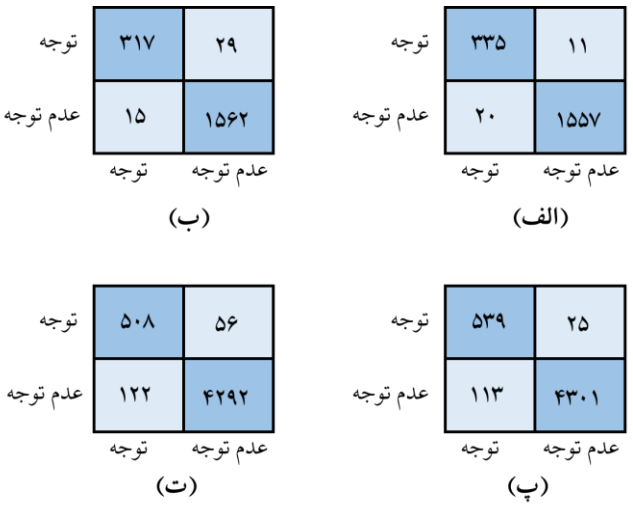
علاوه بر تحلیل‌های ۱۰ کلاسه، ارزیابی عملکرد مدل بر اساس رویکرد دوکلاسه (تفکیک وضعیت توجه از عدم توجه) نیز برای هر دو مجموعه داده صورت گرفته است. جزئیات ماتریس درهم‌ریختگی در شکل ۱۲ و نتایج حاصل از معیارهای ارزیابی در جدول ۱۳ قابل مشاهده است.

جدول (۱۳): نتایج حاصل از عملکرد مدل در حالت دوکلاسه

مجموعه‌داده	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
AUCDD-V2 در رویکرد اول	۹۸/۳۹	۹۶/۸۲	۹۴/۳۷	۹۵/۵۸
AUCDD-V2 در رویکرد دوم	۹۷/۷۱	۹۱/۶۲	۹۵/۴۸	۹۳/۵۱
SFDDD در رویکرد اول	۹۷/۲۳	۹۵/۵۷	۸۲/۶۷	۸۸/۶۵
SFDDD در رویکرد دوم	۹۶/۴۲	۹۰/۰۷	۸۰/۶۳	۸۵/۰۹

نتایج نشان می‌دهد که تفکیک داده‌ها به دو کلاس از نظر کاربردی نیز می‌تواند مفید باشد. اما، در این پژوهش با توجه به مطالعات پیشین همان رویکرد استاندارد رعایت شده و تحلیل بر اساس ۱۰ کلاس اصلی انجام شده است.

لازم به ذکر است که برخی از خطاهای مدل، مربوط به جابجایی بین خود کلاس‌های عدم توجه c1 تا c9 است. به عبارت دیگر، مدل در تشخیص کلاس عدم توجه موفق عمل کرده است، اما در تعیین نوع دقیق آن عدم توجه دچار خطا شده است. به همین دلیل، در تحلیل دوکلاسه توجه در مقابل عدم توجه، این موارد به عنوان تشخیص درست در نظر گرفته شده‌اند، زیرا ماهیت اصلی وضعیت عدم توجه به درستی توسط مدل تشخیص داده شده است.



شکل (۱۲): ماتریس درهم‌ریختگی برای طبقه‌بندی دو کلاسه؛ (الف،ب) نتایج مجموعه‌داده AUCDD-V2 در رویکرد اول و دوم، (پ، ت) نتایج مجموعه‌داده SFDDD در رویکرد اول و دوم

۴-۶- مطالعه فرسایشی جهت تحلیل تأثیر مؤلفه‌های پیشنهادی

به‌منظور بررسی دقیق تأثیر مؤلفه‌های پیشنهادی، مطالعه‌ای فرسایشی بر روی ۱۰ درصد از تصاویر هر یک از بخش‌های آموزش، اعتبارسنجی و آزمون مجموعه‌داده‌ی AUCDD-V2 انجام شد. این انتخاب به‌دلیل محدودیت‌های سخت‌افزاری و عدم دسترسی به پردازنده‌ی گرافیکی مناسب برای پردازش هم‌زمان کل مجموعه‌داده صورت گرفت. در تمامی سناریوها، ساختار پایه مدل شامل شبکه ConvNeXt-Base به‌عنوان استخراج‌کننده ویژگی ثابت در نظر گرفته شد و تنها اجزای الحاقی تغییر داده شدند. ویژگی‌های استخراج‌شده به یک طبقه‌بند سه‌لایه با ۱۰۲۴، ۲۰۴۹ و ۱۰ نورون منتقل شدند.

به‌منظور تضمین مقایسه‌ای منصفانه، همه سناریوها تحت شرایط آموزشی و ارزیابی یکسان اجرا شدند و مقدار بذر تصادفی (Seed=42) به‌منظور بازتولیدپذیری نتایج ثابت نگه داشته شد. در این چارچوب، با حذف یا جایگزینی مؤلفه‌های کلیدی، تأثیر یادگیری انتقالی، مکانیزم توجه و شبکه KAN بر عملکرد نهایی مدل مورد ارزیابی قرار گرفت. در ادامه چهار سناریو بررسی شده تشریح میگردد:

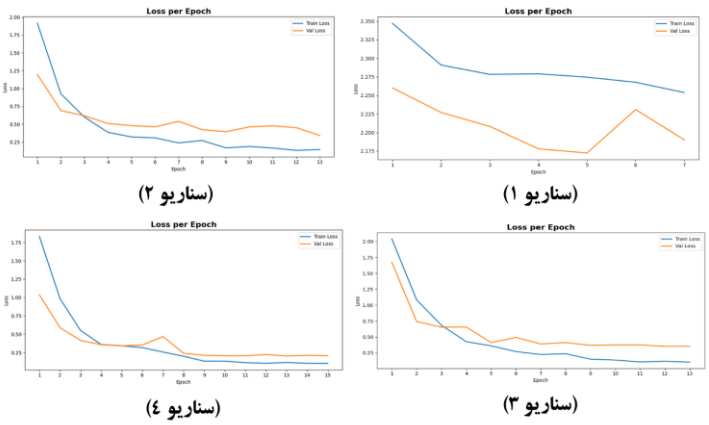
- سناریو۱: شبکه ConvNeXt-Base به عنوان استخراج کننده ویژگی با لایه طبقه‌بندی MLP بدون وزن‌های اولیه ImageNet
- سناریو۲: شبکه ConvNeXt-Base به عنوان استخراج کننده ویژگی با لایه طبقه‌بندی MLP و وزن‌های اولیه ImageNet
- سناریو۳: شبکه ConvNeXt-Base مجهز به مکانیزم توجه SE-Net به عنوان استخراج کننده ویژگی با لایه طبقه‌بندی MLP و وزن‌های اولیه ImageNet
- سناریو٤: شبکه ConvNeXt-Base مجهز به مکانیزم توجه SE-Net به عنوان استخراج کننده ویژگی با لایه طبقه‌بندی KAN و وزن‌های اولیه ImageNet

نتایج این تحلیل در ادامه در جدول ۱۴ ارائه شده است.

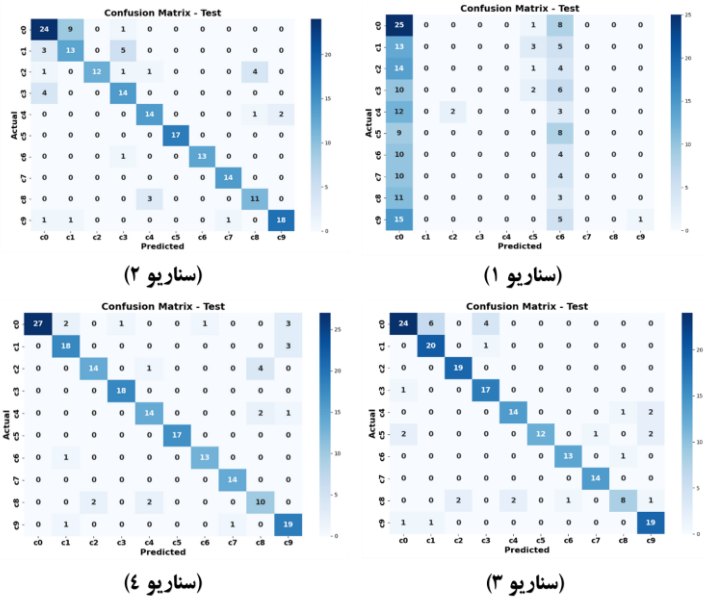
جدول (۱۴): نتایج حاصل از مطالعه فرسایشی

سناریو	صحت (%)	دقت (%)	حساسیت (%)	امتیاز F1 (%)
سناریو ۱	۱۵/۸۷	۱۲/۷۴	۱۰/۶۹	۰/۰۵۲۳
سناریو ۲	۷۹/۳۷	۸۲/۲۷	۸۱/۲۹	۸۱/۲۱
سناریو ۳	۸۴/۶۶	۸۶/۰۴	۸۵/۳۷	۸۴/۸۹
سناریو ۴	**۸۶/۷۷**	**۸۶/۸۲**	**۸۷/۵۹**	**۸۶/۸۸**

نتایج نشان داد که استفاده از وزن‌های اولیه ImageNet موجب بهبود فرآیند یادگیری مدل شده است و افزودن مکانیزم توجه SE-Net نیز توانایی مدل در استخراج ویژگی‌های معنادارتر را افزایش داده است. علاوه بر این، جایگزینی طبقه‌بند MLP با KAN در این چارچوب به بهبود عملکرد نهایی مدل منجر شده است. به‌طورکلی، ترکیب ConvNeXt-Base با SE-Net و KAN بهترین نتایج را ارائه کرده و بیانگر تأثیر مثبت تمامی مؤلفه‌های پیشنهادی است. همچنین در شکل ۱۳ و ۱٤ به ترتیب نمودار زیان و ماتریس درهم‌ریختگی مربوط به هر سناریو در این مطالعه فرسایشی نشان داده شده است.



شکل (۱۳): نمودار زیان سناریوهای مربوط به مطالعه فرسایشی



شکل (۱٤): ماتریس درهم‌ریختگی سناریوهای مربوط به مطالعه فرسایشی

۵- نتیجه‌گیری و کارهای آینده

در این پژوهش، روشی برای تشخیص خودکار عدم تمرکز راننده با استفاده از بینایی ماشین و یادگیری عمیق ارائه شد. نتایج نشان داد که به دلیل محدودیت تعداد نمونه‌های موجود در مجموعه‌دادگان، بهره‌گیری از شبکه‌های از پیش‌آموزش‌دیده برای ارتقای عملکرد مدل ضروری است. بر این اساس، پس از ارزیابی‌های دقیق، شبکه ConvNeXt-Base به‌عنوان مدل پایه برای استخراج ویژگی‌ها انتخاب شد. این معماری مبتنی‌بر بلوک‌های پیچشی عمقی با هسته ۷×۷ است که در مراحل ابتدایی اطلاعات مکانی را استخراج کرده و سپس تمرکز پردازش را بر ویژگی‌های کانالی معطوف می‌کند؛ این ویژگی موجب تحلیل دقیق‌تر و مؤثرتر داده‌ها می‌شود. برای افزایش دقت، مکانیزم توجه SE-Net به معماری ConvNeXt-Base افزوده شد و در مرحله طبقه‌بندی، شبکه عصبی KAN مورد استفاده قرار گرفت. قابلیت یادگیری توابع فعال‌سازی توسط KAN انعطاف‌پذیری مدل را افزایش داده و امکان تقریب توابع پیچیده با دقت بالا را فراهم می‌کند.

نتایج آزمایش‌ها حاکی از آن است که روش پیشنهادی در هر دو رویکرد مورد بررسی عملکرد قابل‌توجهی از خود نشان داده است. بر روی مجموعه‌داده AUCDD-V2، در رویکرد اول که تقسیم‌بندی داده‌ها به‌صورت آموزش و آزمون انجام شد، مدل به صحت ۹۵/۰۱٪ دست یافت. در رویکرد دوم، با تقسیم‌بندی داده‌ها به آموزش، اعتبارسنجی و آزمون، صحت ۹۴/۳۳٪ حاصل شد. به‌طور مشابه، در مجموعه‌داده SFDDD، مدل در رویکرد اول به صحت ۹۵/۶۶٪ و در رویکرد دوم به صحت ۹۴/۱۳٪ دست یافت. کاهش نسبی صحت در رویکرد دوم ناشی از تخصیص بخشی از داده‌ها به مرحله اعتبارسنجی و کاهش داده‌های آموزشی بود، که این موضوع اهمیت افزایش تنوع و گستردگی داده‌های آموزشی را برجسته می‌کند.

با توجه به نتایج به‌دست‌آمده، مسیرهایی برای تحقیقات آینده مطرح می‌شوند که می‌توانند دقت و کارایی مدل تشخیص عدم تمرکز راننده را بهبود بخشند. پس از استخراج ویژگی‌ها از شبکه ConvNeXt-Base مجهز به SE-Net، می‌توان پیش از ارسال ویژگی‌ها به مرحله طبقه‌بندی از الگوریتم‌های انتخاب ویژگی[۱] استفاده کرد تا ویژگی‌های کم‌اهمیت حذف شده و دقت مدل افزایش یابد [۵۳]؛ همچنین، می‌توان از شبکه‌های عصبی پیچشی دیگر مانند EfficientNet، VGG و DenseNet برای استخراج ویژگی‌های مکمل بهره برد و سپس این ویژگی‌ها را با ویژگی‌های ConvNeXt-Base ترکیب کرد تا عملکرد مدل در شرایط واقعی بهبود یابد.

علاوه بر این، به‌کارگیری روش چکانش دانش[۲] امکان انتقال دانش از یک مدل بزرگ‌تر و پیچیده‌تر به مدل کوچک‌تر و با پارامتر کمتر را فراهم می‌کند، به گونه‌ای که مدل کوچک بتواند عملکرد مشابه مدل بزرگ را با هزینه محاسباتی کمتر ارائه دهد [۵۴].

از منظر مجموعه‌داده‌ها، گسترش داده‌های آموزشی با تعداد بیشتر افراد، تصاویر ثبت‌شده از زوایای مختلف، تنوع در پوشش و ظاهر، و شرایط نوری متفاوت می‌تواند تعمیم‌پذیری مدل را افزایش داده و عملکرد آن را در شرایط واقعی بهبود بخشد. با پیاده‌سازی این مسیرها در تحقیقات آینده، می‌توان انتظار داشت که مدل ترکیبی ConvNeXt-Base مجهز به SE-Net و طبقه‌بند KAN با دقت و قابلیت تعمیم بالاتر، برای تشخیص رفتار رانندگان در محیط‌های واقعی آماده شود.

مراجع

[1] Road Traffic Injuries. https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries, Accessed April 21, 2025.

[2] Sinha, A., Aneesh, R., and Gopal, S. K., "Drowsiness detection system using deep learning," International Conference on Bio Signals, Images, and Instrumentation, 2021.

[۳] پ. عمرانی ,س. داودی, **"بررسی تمایل رانندگان به استفاده از تلفن همراه در هنگام رانندگی در دو حالت تلفن در دست و هندزفری, "** *اولین کنفرانس ملی مهندسی عمران، توسعه هوشمند و سیستم‌های پایدار*, ۱۴۰۰.

[4] Doudou, M., Bouabdallah, A., and Berge-Cherfaoui, V., "Driver drowsiness measurement technologies: current research, market solutions, and challenges," *International Journal of Intelligent Transportation Systems Research*, Vol. 18, pp. 297-319, 2020.

[5] Saleem, A. A., Siddiqui, H. U. R., Raza, M. A., Rustam, F., Dudley, S., and Ashraf, I., "A systematic review of physiological signals based driver drowsiness detection systems," *Cognitive Neurodynamics*, Vol. 17, No. 5, pp. 1229–1259, 2023.

[6] Pal, A., Kar, S., and Bharti, M., "Algorithm for distracted driver detection and alert using deep learning," *Optical Memory and Neural Networks*, Vol. 30, pp. 257–265, 2021.

[7] Alsrehin, N. O., Gupta, M., Alsmadi, I., and Alrababah, S. A., "U2-net: a very deep convolutional neural network for detecting distracted drivers," *Applied Sciences*, Vol. 13, No. 21, p. 11898, 2023.

[8] Eraqi, H. M., Abouelnaga, Y., Saad, M. H., and Moustafa, M. N., "Driver distraction identification with an ensemble of convolutional neural networks," *Journal of Advanced Transportation*, Vol. 2019, p. 4125865, 2019.

[9] Xiao, W., Liu, H., Ma, Z., and Chen, W., "Attention-based deep neural network for driver behavior recognition," *Future Generation Computer Systems*, Vol. 132, pp. 152–161, 2022.

[10] Ma, Y. and Wang, Z., "Vit-dd: multi-task vision transformer for semi-supervised driver distraction detection," Intelligent Vehicles Symposium, pp. 417–423, 2024.

[11] Alom, M. Z. et al., "The history began from alexnet: a comprehensive survey on deep learning approaches," arXiv, 2018.

[12] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z., "Rethinking the inception architecture for computer vision," IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826, 2016.

[13] He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition," IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778, 2016.

[14] Simonyan, K. and Zisserman, A., "Very deep convolutional networks for large-scale image recognition," arXiv, 2014.

---

[۱] Feature selection

[۲] Knowledge Distillation

[15] Dosovitskiy, A. et al., "An image is worth 16x16 words: transformers for image recognition at scale," arXiv, 2020.

[16] Zhang, Z. and Yang, Y., "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 34, No. 12, pp. 5586–5609, 2022.

[17] Ghiasi, G., Zoph, B., Cubuk, E. D., Le, Q. V., and Lin, T.-Y., "Multi-task self-training for learning general representations," IEEE International Conference on Computer Vision, pp. 8856–8865, 2021.

[18] Mase, J. M., Chapman, P., Figueredo, G. P., and Torres, M. T., "A hybrid deep learning approach for driver distraction detection," International Conference on ICT Convergence, 2020.

[19] Graves, A. and Schmidhuber, J., "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, Vol. 18, pp. 602–610, 2005.

[20] Gao, H. and Liu, Y., "Improving real-time driver distraction detection via constrained attention mechanism," *Engineering Applications of Artificial Intelligence*, Vol. 128, p. 107408, 2024.

[21] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A., "Learning deep features for discriminative localization," IEEE Conference on Computer Vision and Pattern Recognition, pp. 2921–2929, 2016.

[22] Khan, T., Choi, G., and Lee, S., "Effnet-ca: an efficient driver distraction detection based on multiscale features extractions and channel attention mechanism," *Sensors*, Vol. 23, No. 8, p. 3835, 2023.

[23] Tan, M. and Le, Q., "Efficientnet: rethinking model scaling for convolutional neural networks," International Conference on Machine Learning, pp. 6105–6114, 2019.

[24] Mittal, H. and Verma, B., "Cat-capsnet: a convolutional and attention based capsule network to detect the driver's distraction," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 24, No. 9, pp. 9561–9570, 2023.

[25] LaLonde, R. and Bagci, U., "Capsules for object segmentation," arXiv, 2018.

[26] Duan, C., Liao, J., Ding, N., and Cao, L., "Enabling robust distracted driving performance across datasets with clip," International Conference on SMC-IoT, 2023.

[27] Radford, A. et al., "Learning transferable visual models from natural language supervision," International Conference on Machine Learning, pp. 8748–8763, 2021.

[28] Duan, C., Liu, Z., Xia, J., Zhang, M., Liao, J., and Cao, L., "Enhancing cross-dataset performance of distracted driving detection with score-softmax classifier," arXiv, 2023.

[29] Zhang, Z., Yang, L., and Lv, C., "Highly discriminative driver distraction detection method based on swin transformer," *Vehicles*, Vol. 6, No. 1, pp. 140–156, 2024.

[30] Liu, Z. et al., "Swin transformer: hierarchical vision transformer using shifted windows," IEEE International Conference on Computer Vision, pp. 10012–10022, 2021.

[31] Khellal, A., Boulahmar, M., Bahi, A., and Nemra, A., "Distracted driver detection using convolutional neural networks based segmentation model," International Conference on Electrical Engineering and Automatic Control, 2024.

[32] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H., "Encoder-decoder with atrous separable convolution for semantic image segmentation," European Conference on Computer Vision, pp. 801–818, 2018.

[۳۳] م. هاشمی، ع. میررشید، س. ع. بهشتی شیرازی، **"تشخیص حواس پرتی و خواب آلودگی راننده از طریق روشهای مبتنی بر پردازش تصویر و یادگیری عمیق,"** *فناوری اطلاعات و ارتباطات انتظامی*, ۱۳۹۹.

[۳٤] ف. نساء، ع. ا. خواصی، **"تشخیص حواس پرتی راننده مبتنی بر الگوریتمهای بینایی ماشین با استفاده از ویژگی های چشمی,"** *چهارمین کنفرانس ملی علوم و مهندسی کامپیوتر و فناوری اطلاعات*, ۱۳۹۷.

[۳۵] م.ع. چاروسائی، ی. کبیری، ح.ر. چاروسائی، **"طراحی رویکردهای یادگیری عمیق سیستم های تشخیص حواس پرتی راننده,"** *ششمین کنفرانس بین المللی مطالعات جهانی در علوم تکنولوژی و مهندسی*, ۱٤۰۱.

[36] Tang, Z., Gao, Y., Karlinsky, L., Sattigeri, P., Feris, R., and Metaxas, D., "OnlineAugment: Online Data

Augmentation with Less Domain Knowledge," European Conference on Computer Vision (ECCV), pp. 313–329, 2020.

[37] State Farm. "State Farm Distracted Driver Detection.", https://www.kaggle.com/c/state-farm-distracted-driver-detection/overview, Accessed April 21, 2025.

[38] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K., "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 mb model size," arXiv, 2016.

[39] Howard, A. et al., "Searching for mobilenetv3," IEEE International Conference on Computer Vision, pp. 1314–1324, 2019.

[40] Szegedy, C. et al., "Going deeper with convolutions," IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[41] Ma, N., Zhang, X., Zheng, H.-T., and Sun, J., "Shufflenet v2: practical guidelines for efficient cnn architecture design," European Conference on Computer Vision, pp. 116–131, 2018.

[42] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q., "Densely connected convolutional networks," IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708, 2017.

[43] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S., "A convnet for the 2020s," IEEE Conference on Computer Vision and Pattern Recognition, pp. 11976–11986, 2022.

[44] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L., "Imagenet: a large-scale hierarchical image database," IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255, 2009.

[45] Vaswani, A. et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, Vol. 30, 2017.

[46] Chen, S., Ogawa, Y., Zhao, C., and Sekimoto, Y., "Large-scale individual building extraction from open-source satellite imagery via super-resolution-based instance segmentation approach," *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 195, pp. 129–152, 2023.

[47] Hu, J., Shen, L., and Sun, G., "Squeeze-and-excitation networks," IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141, 2018.

[48] Tajbakhsh, N. et al., "Convolutional neural networks for medical image analysis: full training or fine tuning?," *IEEE Transactions on Medical Imaging*, Vol. 35, No. 5, pp. 1299–1312, 2016.

[49] Liu, Z. et al., "Kan: kolmogorov-arnold networks," arXiv, 2024.

[50] Hasan, Md. S., Alam, Md. N., Asad, Md. F.-A., Muhammad, N., and Tunç, C., "B-spline curve theory: an overview and applications in real life," *Nonlinear Engineering*, Vol. 13, p. 20240054, 2024.

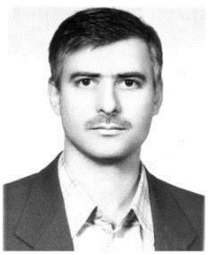
[51] Alzubaidi, L. et al., "Review of deep learning: concepts, cnn architectures, challenges, applications, future directions," *Journal of Big Data*, Vol. 8, pp. 1–74, 2021.

[52] Qin, B., Zhang, B., and Qian, J., "ICK-PANet: A Multiscale Driver Distraction Detection Network Based on Attention and Pyramid Convolution," Vehicles, Vol. 8, No. 4, p. 83, 2026.

[53] Dokeroglu, T., Deniz, A., and Kiziloz, H. E., "A comprehensive survey on recent metaheuristics for feature selection," *Neurocomputing*, Vol. 494, pp. 269–296, 2022.

[54] Hinton, G., Vinyals, O., and Dean, J., "Distilling the knowledge in a neural network," arXiv, 2015.

سمیرا کریمی چقاکبودی، دارای مدرک کارشناسی ارشد مهندسی کامپیوتر گرایش هوش مصنوعی و رباتیکز از دانشگاه رازی، کرمانشاه، ایران می‌باشد. زمینه‌های علمی مورد علاقه ایشان شامل هوش مصنوعی، یادگیری عمیق، بینایی ماشین و پردازش تصویر است.



عبداله چاله‌چاله، مدرک کارشناسی (مهندسی برق) و کارشناسی ارشد (مهندسی کامپیوتر - نرم‌افزار) خود را از دانشگاه صنعتی شریف، تهران، ایران دریافت کرده است. وی مدرک دکتری مهندسی کامپیوتر خود را از دانشگاه ولونگونگ استرالیا در سال ۱۳۸۳ دریافت نموده است. ایشان هم‌اکنون به‌عنوان عضو هیأت علمی در دانشگاه رازی کرمانشاه مشغول به کار می‌باشد. زمینه‌های علمی مورد علاقه ایشان شامل هوش مصنوعی، پردازش تصویر و ویدئو، سیستم‌های توزیع شده و اینترنت اشیا است.