مجله تحقیق در عملیات در کاربردهای آن، سال بیست و یکم، شماره اول (پیاپی ۸۰)، بهار ۰۳، ص ص ۳۶-۱۷
شاپا ۷۲۸۶-۲۲۵۱

آموزش بهینه رفتار راننده با استفاده از یادگیری عمیق در قالب رویکرد آموزشی End-to-End برای هدایت خودروی خودران

مرتضی آقامحمدی[۱]، علی جمالی[۲*]، کامراد خوشحال رودپشتی[۳]

۱- دانشجوی دکتری مکانیک طراحی کاربردی، گروه مکانیک، واحد بندر انزلی، دانشگاه آزاد اسلامی، بندر انزلی، ایران

۲- دانشیار، گروه مکانیک، دانشگاه گیلان، رشت، ایران

۳- استادیار، گروه کامپیوتر، واحد لاهیجان، دانشگاه آزاد اسلامی، لاهیجان، ایران

رسید مقاله: ۲۲ خرداد ۱۴۰۲

پذیرش مقاله: ۸ آبان ۱۴۰۲

چکیده

یکی از مهم‌ترین شاخص‌های اصلی در معیار عملکرد خودروهای خودران، سیاست اتخاذشده توسط سیستم خودران در خصوص تعیین سرعت خودرو و زاویه فرمان می‌باشد. برای تعیین این سیاست همواره محققان با چالش انتخاب روش آموزش بهینه ما بین دو رویکرد سنتی مدولار و مدرن End-to-End مواجه بوده‌اند. اخیرا تحقیقات زیادی در راستای معرفی رویکرد End-to-End وکاربرد آن در این حوزه انجام شده است. در این پژوهش مدلی بهینه برای پیش‌بینی رفتار راننده با به‌کارگیری این رویکرد مدرن در قالب یادگیری عمیق برای آموزش شبکه‌های عصبی مصنوعی ارایه‌شده است. به عبارتی دستیابی به مدلی با دقت قابل قبول نسبت به کارهای مشابه در هدایت خودروی خودران مدنظر بوده است. برای این منظور بر اساس بررسی‌های انجام‌شده بر روی معماری شبکه‌های موجود، دو معماری که دارای پتانسیل‌های لازم برای دست‌یابی به این مهم بوده‌اند انتخاب گردید. همچنین برای نادیده نگرفتن رابطه زمانی بین اسلایدها و نشان دادن وابستگی‌های زمانی بصری و بررسی تاثیر آن در نتیجه، در آموزش مدل از ترکیب شبکه‌های عصبی پیچشی (کانولوشنال) با یک نوع شبکه بازگشتی با عنوان حافظه کوتاه مدت بلند LSTM استفاده شده است. در این پژوهش از یک مجموعه داده کامل که در شرایط رانندگی واقعی جمع‌آوری شده و دارای برچسب بوده و شامل تصاویر و اطلاعات عمق می‌باشد، استفاده شد و با طراحی الگوریتم‌های آموزشی و بهینه‌سازی پارامترهای آموزش با استفاده از الگوریتم بهینه‌سازی آدام چندین مدل آموزش‌دیده ارایه گردید که از بین نتایج به‌دست آمده برخی از پیش‌بینی‌ها بهینه‌تر از کارهای مشابه بودند و این امر نشان از تاثیر بی‌بدیل وابستگی‌های زمانی در آموزش و اثرگذاری شبکه‌های بازگشتی در کنار پردازش قوی شبکه‌های پیچشی را دارد.

کلمات کلیدی: خودروی خودران، رویکرد آموزشی End-to-End ، یادگیری عمیق، شبکه‌های عصبی مصنوعی عمیق، شبکه‌های عصبی پیچشی، شبکه بازگشتی با حافظه کوتاه مدت بلند LSTM .

* عهده‌دار مکاتبات

آدرس الکترونیکی: ali.jamali@guilan.ac.ir

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

۱ مقدمه

افزایش مداوم تعداد وسایل نقلیه در جاده‌ها منجر به افزایش فشار برای حل مسایلی مانند تراکم ترافیک، آلودگی و ایمنی جاده شده است. وسایل نقلیه خودران این پتانسیل را دارند که سیستم های حمل و نقل ما را از نظر ایمنی و کارایی متحول کنند. پاسخ اصلی برای حل این مسایل در میان جامعه تحقیقاتی اتومبیل‌های خودران است. به عنوان مثال طبق گزارش سازمان جهانی بهداشت، سالانه ۱/۳ میلیون نفر در تصادفات جاده ای جان خود را از دست می‌دهند. در همین حال، تخمین زده می شود که تا ۹۰٪ از تمام تصادفات رانندگی ناشی از خطاهای انسانی است. بنابراین وسایل نقلیه خودران[1] می‌توانند با حذف خطاهای راننده، پیشرفت‌های ایمنی قابل توجهی را ارایه دهند. مزایای بیشتر ارایه‌شده توسط وسایل نقلیه خودران شامل مصرف سوخت بهتر، کاهش آلودگی، اشتراک‌گذاری خودرو، افزایش بهره‌وری و بهبود جریان ترافیک است. همچنین از زمانی که هوش مصنوعی به‌طور فراگیر به عموم معرفی گردید، یکی از تکنولوژی‌هایی که مردم هر روز انتظار آن را می‌کشند تا به‌طور گسترده در چرخه سیستم حمل و نقل قرار گیرد، خودروهای خودران است[۱]. از سوی دیگر، توسعه علم یادگیری ماشین[2]، و به ویژه یادگیری عمیق[3]، کاوش و پیشرفت تحقیقات در زمینه خودروهای خودران را به طور قابل توجهی تسریع کرده است و همین امر بستر وسیعی جهت انجام تحقیقات را پیش روی محققان فراهم آورده است [۲]. یکی از فضاهای تحقیقاتی، شیوه‌های متنوع به‌کارگیری از شبکه‌های عصبی می‌باشد. به‌طور معمول کاربرد شبکه‌های عصبی عمیق در برازش مدل‌ها در خصوص هدایت وسایل نقلیه خودران با دو رویکرد، یادگیری ترتیبی[4] و یادگیری End-to-End ارایه می‌گردد. وسایل نقلیه خودران در روش ترتیبی معمولا به طور کلی از پنج جزء عملکردی، ادراک، محلی‌سازی، برنامه‌ریزی، کنترل و مدیریت سیستم تشکیل شده‌اند. این رویکرد ادراک واسطای به طور گسترده در صنعت خودرو نیز مورد استفاده قرار می‌گیرد که به یک سیستم کمکی بزرگ برای تشخیص، پیش بینی و تصمیم گیری نیاز دارد. اما در مقابل، رویکرد یادگیری End-to-End از نظر پیاده‌سازی سبک‌تربوده و از یک نظریه متمایز برای راندن ماشین استفاده می‌کند. این روش به دنبال ساخت مدل‌های یادگیری ماشین بر اساس تقلید از رانندگی انسان است[۱، ۳، ۴].

در رویکردEnd-to-End پیکسل‌های تصاویر به‌عنوان ورودی همراه با رفتار راننده یا همان برچسب‌های زاویه فرمان و سرعت خودرو که توسط انسان کنترل می‌شود، مولفه‌های آموزشی را تشکیل می‌دهند[۵]. هدف اصلی در این رویکرد انتخاب سیاست رانندگی، مستقل از شناسایی ویژگی‌های خاص انسانی در ادراک است. سیاست های آموخته‌شده کاملاً فاقد قابلیت تفسیر هستند؛ زیرا شبکه عصبی مانند یک جعبه سیاه است. وقتی یک شبکه عصبی عمیق مستقیماً از مشاهدات خام برای کنترل استفاده می‌کند، نمی‌توانیم نحوه عملکرد آن را توضیح دهیم که البته برخی از محققین این ویژگی را یک اشکال برای این رویکرد می‌دانند در حالی‌که این ویژگی مدل

[1] Autonomous vehicles
[2] Machine learning
[3] Deep learning
[4] Modular learning

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

را قادر می‌سازد تا از اطلاعات خام سنسورهای بسیار مهمی مانند لیدار[1] که نقاط ابری[2] را در اختیار ما قرار می‌دهند، ویژگی‌های کاربردی غیر قابل فهم را برای بهبود عملکرد مدل استفاده نماید [۶، ۸، ۹].

در این تحقیق سعی شده است با پیاده سازی معماری‌های مختلف شبکه‌های عصبی عمیق مبتنی بر رویکرد End-to-End و به‌کارگیری روش بهینه‌سازی متناسب با شبکه‌ها، بر روی یک مجموعه داده واقعی که مشمول تصاویر و اطلاعات عمق می‌شود، مدل‌هایی را برازش کنیم که با کاهش حداکثری خطا نتایج قابل قبولی در خصوص پیش‌بینی رفتار راننده نسبت به کارهای مشابه ارایه نماید. در این تحقیق، الگوریتم‌های یادگیری عمیق مدنظر را بر روی یک مجموعه داده متناسب با اهداف تحقیق که مجموعه ای بزرگ و کامل بوده و در شرایط رانندگی واقعی جمع‌آوری شده است، پیاده‌سازی می‌کنیم. برازش مدل در قالب الگوهای متفاوت انجام‌شده است تا سیاست‌های رانندگی خودکار در چارچوب یادگیری End-to-End تعیین و سپس نتایج را نسبت به الگوریتم‌های دیگر در کارهای مشابه مقایسه نماییم.

در اولین گام، بررسی‌های انجام‌شده بر روی پیشینه تحقیق ارایه شده است. سپس در بخش سوم، تمامی فرایند مدلسازی در این تحقیق شامل، مجموعه داده، الگوریتم‌های پیشنهادی و مبانی نظری آن و روش ارزیابی به تفسیر بیان شده است. پیاده‌سازی آزمایش و ارایه نتایج به‌دست آمده و تحلیل‌های آن در بخش چهارم این تحقیق ارایه‌شده و در نهایت در بخش پنجم جمع‌بندی نتایج و پتانسیل‌های موجود برای تحقیقات آینده را بیان کرده‌ایم.

۲ پیشینه تحقیق

چنی چن و همکاران سه رویکرد آموزشی در خودرو های خودران را بررسی و مقایسه کردند. دو رویکرد اصلی که بیشترین کاربرد را درتحقیقات این حوزه دارد یعنی ادراک واسطه‌ای[3] یا همان روش ترتیبی و رویکرد End-to-End که بر اساس بازتابی از تقلید رفتار راننده است. این محققان روش سومی را با نام رویکرد ادراک مستقیم[4] معرفی می‌کنند که از نظر عملکردی بین دو رویکرد فوق قرار می‌گیرد. در این تحقیق برای آموزش شبکه عصبی از داده‌های ضبط‌شده توسط یک بازی رایانه‌ای استفاده‌شده و سپس مدل برازش‌شده را بر روی داده های واقعی KITI[5] آزمایش کردند و مدعی شدند که نتایج مطلوبی حاصل گردید[۳]. سپس بوجارسکی و همکاران یک شبکه عصبی کانولوشنی را مبتنی بر رویکرد تقلیدی End-to-End آموزش دادند که تمام خطوط خط پردازش مورد نیاز برای رانندگی یک وسیله نقلیه خودران را بدون نیاز به تشخیص الگو آموزش می‌دهد. انگیزه اصلی این گروه از بین بردن شناخت ویژگی های مورد نیاز انسان بود. در واقع آنها می‌خواستند مجموعه‌ای از قوانین «اگر، آنگاه، دیگری» را کنار بگذارند. آنها ابتدا از طریق سخت افزار NVIDIA DRIVETM PX اقدام به جمع‌آوری داده‌های آموزش کردند. سپس از طریق NVIDIA DevBox اقدام به آموزش شبکه عصبی مدل خود کردند[۷]. همچنین شیوو و همکاران، یک معماری ترکیبی جدیدی را با نام FCM-LSTM

[1] Lidar sensor
[2] Point cloud
[3] Mediated perception
[4] Direct Perception
[5] Karlsruhe Institute of Technology and Toyota Technological Institute (KITTI)

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

معرفی کردند که در این مدل علاوه بر در نظر گرفتن تاثیر زمان و ترکیب آن با شبکه های کاملاً کانولوشنال که توسط مجموعه داده‌های ویدیویی نسبتاً بزرگ تحت آموزش قرار می‌گرفتند، از تکنیک بخش‌بندی صحنه‌ها برای بهبود عملکرد تحت یک الگوی یادگیری ممتاز استفاده می‌کردند که این روش با عنوان رویکرد آموزش ممتاز معرفی شد[۴]. پس از آن نظر به رشد روزافزون توجه به روش یادگیری بازتاب رفتاری در خودروهای خودران، فقدان یک مجموعه داده بزرگ و کامل که هم شامل تصاویر و هم اطلاعات عمق تصاویر باشد و در عین حال رفتار راننده را به عنوان برچسب داشته باشد، به شدت احساس می‌شد. پیینگ چن و همکاران، در یک مطالعه نسبتا کامل، مجموعه داده بزرگی را ارایه کردند که شامل تمامی موارد فوق می‌باشد و همچنین از این مجموعه داده برای آموزش چندین مدل و نمایش نتایج به صورت مقایسه‌ای استفاده کردند[۱۰]. یکی دیگر از تحقیقات انجام‌شده در این حوزه، کدویلا و همکاران بود که در آن می‌گویند، یادگیری تقلیدی به تنهایی کارایی لازم را در شرایط پیچیده شهری ندارد و برای افزایش دقت و بهبود این روش پیشنهاد می‌شود که یادگیری تقلیدی مبتنی بر شروطی باشد. همچنین آنها این روش را در شبیه‌ساز سه بعدی CARLA[1] و با یک کامیون روباتیک در یک منطقه مسکونی آزمایش کردند[۱۱]. همچنین وانگ و همکاران، که به دنبال کاهش مداخلات انسانی تا حد ممکن وکاهش مشکلات احتمالی در خصوص برازش بیش از حد و ناپدید شدن گرادیان بودند، توانستند، با ارایه یک مدل CNN در قالب یک رویکرد یادگیری End-to-End و با پیاده‌سازی تکنیک Dropout به نتایج بهتری نسبت به تحقیق بوجارسکی و همکاران دست پیدا کنند[۱۲]. در تحقیقی دیگر، ای سیاوو و همکاران در راستای بهبود نتایج برای عملکرد خودروی خودران با رویکرد End-to-End پنج مدل را برازش کردند که تفاوت آنها در داده های ورودی و نحوه تلفیق داده های تصویر و لیدار درمعماری شبکه بود[۱۳]. همچنین کیشی ایشیهارا و همکارن یادگیری تقلیدی مشروط را مبتنی بر رویکرد End-to-End ارایه دادند که با نام اختصاری CIL[2] معرفی شد. در این روش علاوه بر این‌که شبکه به‌طور تقلیدی تحت آموزش قرار می‌گیرد برخی از ویژگی های بصری را به‌عنوان شرط عملکرد یادگیری تقلیدی نیز به‌صورت فرعی آموزش می‌بیند و به‌عنوان مثال تفسیر چراغ راهنمایی را آزمایش کردند[۱۴]. آدیتیا پراکاش و همکاران نیز درتحقیقی که در آن درچهارچوب رویکرد پایان به پایان مدلی را در محیط شبیه‌ساز کارلا معرفی کرده بودند، مدعی شدند که با نوآوری ارایه‌شده درروش تلفیق داده‌های ورودی، تصاویر و نقاط ابری لیدار با نام ترانس فیوزر[3]، نتایج مطلوب‌تری نسبت به روش های فیوژن درعملکرد مدل برای هدایت خودروبه‌دست آوردند[۱۵]. پارک مینگیو و همکارانش در دانشگاه سونچون هیانگ تحقیقی را در زمینه مشابه ارایه کردند که در آن به‌جای استفاده از داده‌های شبیه‌سازی‌شده، با استفاده از داده‌های رانندگی واقعی الگوریتم رانندگی مستقل E2E[4] را ارایه کردند که دو مزیت را در آماده‌سازی داده‌ها به‌همراه داشت. اول این‌که داده‌های بسیار مشابه را که در سرعت‌های پایین و زمان‌های توقف خودرو جمع‌آوری شده بودند را با حفظ صحیح ساختار داده‌ها، حذف کردند و دوم این‌که

[1] California Amateur Radio Linking Association (CARLA)
[2] Conditional imitation learning (CIL)
[3] TransFuser
[4] End-to-End

[Downloaded from jorlu.ir on 2026-11-29]
[DOI: 10.71773/jamlu-2024-1-2167]

داده‌های لیدار را برای استفاده در شبکه عصبی بر پایه معادله معرفی‌شده به تصاویر دو بعدی با سه کانال رنگی که هر یک بیانگر فواصلی خاص بود ارایه دادند. همچنین دیگر ابتکار ارایه‌شده در این مقاله ترکیب دو معماری به‌صورت موازی برای استخراج ویژگی‌ها برای دو نوع ورودی ارایه شده بود[۱۶]. و در نهایت دیان چن و همکاران تحقیقی را مبتنی بر یادگیری End-to-End ارایه کردند که در آن برای افزایش دقت عملکرد در سه آیتم تکمیل مسیر، هدایت خودرو و اطلاعات، روشی را با نام LAV[1] معرفی کردند که نشان می‌دهد با استفاده از جمع‌آوری اطلاعات از خودروهای اطراف آموزش مدل غنی‌تر و عملکردی بهتر خواهد داشت. این تست در محیط شبیه‌ساز انجام شده است[۱۷].

در بررسی پیشینه تحقیق و کارهای مرتبط مشخص گردیدکه هدف‌گیری اصلی در این تحقیقات، دستیابی به دقت قابل قبول با رویکرد آموزشی مدرن مدنظر بوده است. اما نکته مشترک و قابل تامل در این تحقیقات، چالش انتخاب الگوریتم و استفاده از الگوریتم‌های تکراری می‌باشد. ما نیز تصمیم گرفتیم بر خلاف کارهای مشابه از الگوریتم‌های به‌روزشده و غیرتکراری استفاده نموده و روش‌های ترکیبی را نیز آزمایش نماییم.

۳ روش‌شناسی تحقیق

در این بخش از تحقیق به تفسیر قسمت‌هایی می‌پردازیم که در کنار یکدیگر فرایند آموزش مدلی را تشکیل می‌دهند که توانایی پیش‌بینی رفتار بهینه راننده را محقق می‌نماید. ابتدا مجموعه داده و خصوصیات آن معرفی شده است و سپس اطلاعات مورد نیاز در خصوص آماده‌سازی مدل جهت پیاده‌سازی بر روی مجموعه داده ارایه گردید. و بعد از آن الگوهای متفاوت جهت آموزش مدل بررسی شده است و در نهایت جهت دستیابی به نتایج مطلوب، ارزیابی مدل و روش بهینه‌سازی تشریح گردید.

۳-۱مجموعه داده

بدون شک داشتن مجموعه داده با کیفیت می‌تواند در آموزش و اجرای الگوریتم‌های خودروهای خودران بسیار موثر باشد. مجموعه داده‌ها اغلب از طریق شبیه‌سازها و یا با رانندگی وسیله نقلیه در جاده‌های عمومی تولید و جمع‌آوری می‌شوند. تهیه و تولید مجموعه داده در حوزه خودرو های خودران همواره یکی از موضوعات اصلی در تحقیقات این حوزه بوده است. در بسیاری از تحقیقات مشابه به‌دلیل سادگی کار از مجموعه داده‌های شبیه‌سازی‌شده استفاده شده است که البته هم دارای معایب و هم مزایایی می‌باشد. ولی در این تحقیق از یک مجموعه داده واقعی استفاده می‌کنیم. تولید چنین مجموعه داده‌های بسیار سخت و زمان بر است، به این صورت است که وسیله نقلیه به سنسورهای مختلفی مانند دوربین، لیدار، رادار، جی پی اس و ... مجهز می‌شود و با رانندگی و هدایت دستی خودرو توسط راننده در جاده‌های مختلف، که نظر به استراتژی تحقیق ممکن است شرایط ترافیکی و آب و هوایی متفاوتی مد نظر باشد، اطلاعات ضبط‌شده توسط حسگرها به عنوان یک مجموعه داده خام جمع‌آوری می‌شود که قطعا برای استفاده در مدل‌ها نیاز به پیش پردازش‌های زیادی دارد. ولی اگر

[1] Learning from All Vehicles (LAV)

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

مجموعه داده کامل‌تری مدنظر باشد و به‌دنبال مدلسازی با نظارت هستیم، رفتار راننده نیز باید به‌عنوان برچسب داده‌ها هم‌زمان ذخیره‌سازی گردد که در این حالت مرحله همگام‌سازی داده‌ها نقش حیاتی در ارایه این مجموعه داده خواهد داشت. می‌توان گفت از سال ۲۰۰۹ به بعد، چند سال پس از چالش بزرگ DARPA[1]، موضوع جمع‌آوری اطلاعات در جاده‌های عمومی اهمیت بیشتری پیدا کرد. نظر به این‌که در این تحقیق، آموزش End-to-End برای مدل در نظر گرفته شده بود و همچنین از آنجایی که به‌دنبال ارتقای دقت در پیش بینی رفتار راننده با به‌کارگیری اطلاعات عمق بودیم، می‌بایست مجموعه داده‌ای را انتخاب می‌کردیم که هم دارای برچسب می‌شد و هم از طریق سنسور لیدار اطلاعات عمق را نیز در اختیار ما قرار می‌داد در جدول زیر برخی از محبوب‌ترین مجموعه داده‌هایی که قابلیت دسترسی برای آن فراهم بود و در تحقیقات مشابه استفاده‌شده نشان داده شده است [۱۰، ۱۸].

جدول ۱. تعدادی از محبوب‌ترین و کامل‌ترین مجموعه داده‌های جمع‌آوری‌شده در حوزه خودروهای خودران [۱۸].

نام مجموعه داده	زمان و مکان	حجم داده	شرایط ترافیکی	حسگرها	فرمت داده
Daimler pedestrian	۲۰۰۶ – ۲۰۱۶ چین	۸۸/۳ گیگا بایت	ترافیک شهری	دوربین خاکستری تک چشمی: دید استریو، رنگی یا خاکستری	**تصویر(عکس)**
Caltech	۲۰۰۹ آمریکا	۱۱ گیگا بایت	ترافیک شهری	دوربین رنگی تک چشمی	**ویدئو**
Cheddar Gorge	۲۰۱۰ انگلستان	۳۲۹ گیگا بایت	خشک، آفتابی، روشن، سرد	دوربین رنگی دید استریو حسگر لیدار، حسگر چندگانه	**تصویر(عکس)** **نقاط ابری**
KITTI	۲۰۱۱ آلمان	۱۸۰ گیگا بایت	شهری، روستایی، بزرگراه	۲ عدد دوربین سیاه وسفید تک چشمی، حسگر لیدار، حسگر چندگانه	**تصویر(عکس)** **نقاط ابری** **برچسب**
Oxford	۲۰۱۴ – ۲۰۱۵ انگلستان	۲۳/۱۵ ترا بایت	شرایط مختلف: نوری و آب و هوایی	۳ دوربین رنگی تک چشمی دید استریو، حسگر لیدار، GPS/INS	**تصویر(عکس)** **نقاط ابری** **داده موقعیت مکانی**
comma.ai	۲۰۱۶ آمریکا	۸۰ گیگا بایت	نور روز، بیشتر بزرگراه	دوربین رنگی تک چشمی، ژیروسکوپ GPS+IMU	**تصویر(عکس)** **داده موقعیت مکانی**
Cityscapes	۲۰۱۶ آلمان – سوئیس - فرانسه	۶۳/۱۴ گیگا بایت	نورروز، فاقد شرایط آب و هوایی نامطلوب	دوربین رنگی دید استریو	**تصویر(عکس)** **برچسب**
Udacity	۲۰۱۶ آمریکا	۲۲۳ گیگا بایت	آفتابی، ابری، نور روز	دوربین رنگی تک چشمی، GPS+IMUحسگر لیدار،	**تصویر(عکس)** **نقاط ابری، برچسب** **داده موقعیت مکانی**
LiVi-Set	۲۱۰۸ چین	۱۵ گیگا بایت ۱ ترا بایت	شهری، روستایی، شرایط مختلف: نوری و آب و هوایی	دوربین، حسگر لیدار، حسگر چندگانه	**تصویر(عکس)** **نقاط ابری** **برچسب**
CMU	۲۰۱۰ – ۲۰۱۱ آمریکا	۲۲ گیگا بایت	شرایط مختلف: نوری و آب و هوایی	۳ عدد دوربین رنگی تک چشمی، حسگر لیدار، GPS+IMU	**تصویر(عکس)** **نقاط ابری، برچسب** **داده موقعیت مکانی**
Ford	۲۰۰۹ آمریکا	۱۰۰ گیگا بایت	مرکز شهر، حلقه بسته، محوطه دانشگاه	دو نوع حسگر لیدار، دوربین همه جانبه، GPS+IMU	**تصویر(عکس)** **نقاط ابری** **داده موقعیت مکانی**
Stanford	۲۰۰۹ – ۲۰۱۰ آمریکا	۵/۷۲ گیگا بایت	شهری محوطه دانشگاه	حسگر لیدار، GPS+IMU	**نقاط ابری** **داده موقعیت مکانی**

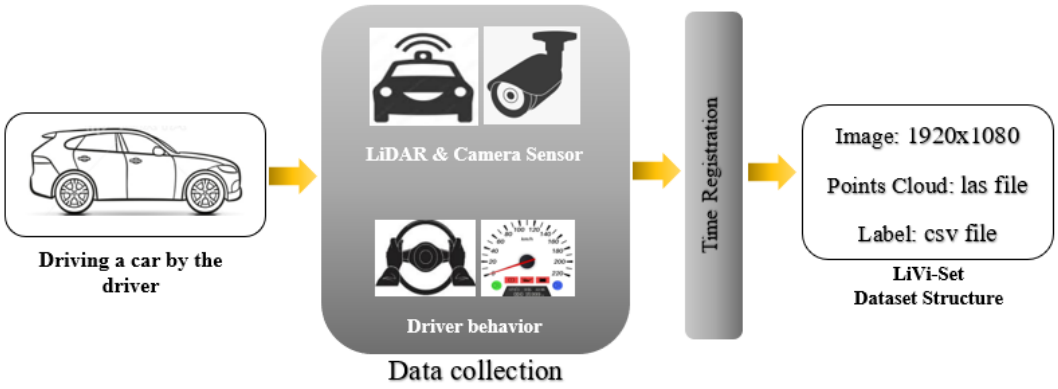
[1] Defense Advanced Research Projects Agency

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

AMUSE	۲۰۱۳ سوئد	۱/۱۶ ترا بایت	شهری، خارج از شهر، شرایط مختلف: نوری و آب و هوایی	دوربین همه جانبه، سنسور سرعت، ۳ عدد سنسور ارتفاع، GPS+IMU	**تصویر(عکس)** **داده موقعیت مکانی**
Elektra	۲۰۱۶ اسپانیا	۱۰/۳۱ گیگا بایت	شهری، نور رو، شب در یک مجموعه داده جدا	دوربین تک چشمی دید استریو، سنسور مادون قرمز دور	**تصویر(عکس)** **نقاط ابری**
JAAD	۲۰۱۶ کانادا – آمریکا	-	شهری، روستایی، روز، کمی شب، شرایط آب و هوایی مختلف	دوربین رنگی تک چشمی	**ویدئو** **بر چسب**

بر این اساس مجموعه داده Livi-Set که از نظر خواسته‌های مدنظر با اهداف این تحقیق هم راستا بود در نظر گرفته شد. از مواردی که می‌توان به آن اشاره کرد، شرایط ترافیکی وحجم کافی داده ها می‌باشدکه با بیش از ۱۰ هزار فریم صحنه واقعی خیابان و با مقدار داده در کل بیش از ۱ ترابایت در گروه بزرگ‌ترین مجموعه داده‌ها قرار دارد. این مجموعه داده از نظر شرایط رانندگی دارای تنوع بسیار کاملی است. این تنوع شامل مسیرهای محلی، بلوارها، جاده‌های اصلی، مسیرهای کوهستانی، محدوده مدارس، تعداد زیادی تقاطع، روگذر، پل های زیرگذر و ... می‌باشد. همچنین این مجموعه داده صحنه‌هایی با تعداد عابر پیاده متفاوت را نشان می‌دهد. این تنوع صحنه‌ها به خوبی می‌تواند رانندگی در شرایط واقعی را فراهم کند. پس از همگام‌سازی داده‌ها، این مجموعه داده در مجموع ۳۸۸۸۰ فرم تصویر و همین تعداد فایل Las و برچسب‌های مربوطه را در اختیار ما قرار می‌دهد که نشان‌دهنده رفتار راننده است. این مجموعه در قالب ۲۸۴ پوشه شامل هر سه مورد فوق برای آموزش و ۲۰ پوشه برای ارزیابی و در نهایت ۲۰ پوشه برای تست در دسترس بوده است [۱۰]. ساختار مجموعه داده مورد استفاده در تحقیق به شرح زیر است.



شکل ۱. ساختار مجموعه داده Livi-set

این مجموعه داده توسط یک وسیله نقلیه جمع‌آوری اطلاعات جاده چند منظوره به‌دست آمده است. وسیله نقلیه مورد استفاده یک دستگاه خودروی سواری بیوک GL8 است که دارای چندین اسکنر و سنسور بوده و سه نوع سیگنال، یعنی ابرهای نقطه‌ای، ویدئوها و رفتارهای رانندگی را جمع‌آوری می‌کند.

ابرهای نقطه‌ای: دو اسکنر Velo-dyne، یکی HDL-32E، و دیگری VLP-16، برای جمع‌آوری اطلاعات عمق در خودرو مورد نظر استفاده شده است. اما بیشتر ابرهای نقطه‌ای توسط اسکنر HDL-32E جمع‌آوری‌شده

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

است که می‌تواند طیف وسیعی از داده‌ها را با دقت بالا جمع‌آوری کند. مشخصات فنی مجموعه داده‌های آن شامل فرکانس ۱۰ هرتز و ۳۲ پرتو لیزر در یک پرتو با عمق ۷۰-۱ متر و وضوح ۲ سانتی‌متر است. همچنین زاویه اسکنر در نمای افقی ۳۶۰ درجه و در نمای عمودی از" ۱۰/۶۷+ تا ۳۰/۶۷- "درجه، چگالی نقاط حدود ۷۰۰۰۰۰ نقطه در ثانیه است. محل نصب این اسکنرها در قسمت جلو و بالای خودرو نصب می‌شود.

ویدئو: برای ضبط تصاویر از دوربین رنگی استفاده می‌شود که با به‌روزرسانی لحظه‌ای جلوی خودر بر روی داشبورد نصب می‌شود. این دوربین توانایی فیلم برداری با سرعت ۳۰ فریم بر ثانیه با رزولوشن ۱۹۲۰ × ۱۰۸۰ را دارد و همچنین با حافظه ۱۲۸ گیگ می‌تواند ۲۰ ساعت فیلم با کیفیتP ۱۰۸۰به صورت مداوم ضبط کند.

رفتار رانندگی: برای ثبت رفتار راننده که شامل سرعت خودرو و زاویه فرمان است، از نرم افزاری استفاده می‌شود که به صورت بی‌سیم به سیستم کنترل خودرو متصل می‌شود. اطلاعات مورد نظر در مورد سرعت را از طریق سنسورهایی دریافت می‌کند که دقت آنها تا ۰/۱ کیلومتر در ساعت است. اما برای زاویه فرمان از سنسور نقاله الکترونیکی استفاده شده که دقت آن ۱ درجه است. این نقاله برای یک سمت از اعداد مثبت و برای طرف دیگر از اعداد منفی استفاده می‌کند تا تفاوت بین زاویه فرمان راست و چپ را نشان دهد.

همان‌طور که در شکل ۱ نشان داده شد، در نهایت این مجموعه داده شامل دو نوع داده می‌باشد. تصاویر ویدئویی ضبط‌شده که عکس‌هایی با زمان‌بندی مشخص از آن استخراج‌شده و همچنین نقاط ابری مربوط به اطلاعات عمق که از طریق سنسور لیدار ۳۶۰ درجه جمع‌آوری شده است و این مجموعه داده دارای یک فایل csv[1] می‌باشد که در واقع به عنوان برچسب مجموعه داده بوده و دو پارامتر کنترلی سرعت خودرو و زاویه فرمان که از سوی راننده اعمال شده است را جمع‌آوری کرده است.

۳-۲ آماده‌سازی مدل

در این تحقیق مدل‌ها در قالب شبکه‌های عصبی عمیق مبتنی بر یادگیری عمیق و با رویکرد آموزشی End-to-End ارایه شده است. شبکه‌های عصبی عمیق[2] استفاده‌شده در این مدل‌ها، الگوریتم پیچشی عمیق و الگوریتم حافظه کوتاه مدت بلند[3] که از زیرمجموعه شبکه‌های بازگشتی[4] است می‌باشد.

۳-۲-۱ رویکرد آموزشی End-to-End

یکی از مهم‌ترین چالش‌های حوزه یادگیری ماشین انتخاب روش مناسب جهت حل یک مشکل خاص است. همچنین وجود الگوریتم‌های بیشمار نیز خود چالشی جدی است. ولی آنچه کار را سخت‌تر می‌کند، نیاز ضروری به معماری‌هایی با اجزای آموزشی زیاد در حل مسایل پیچیده نظیر خودروهای خودران است (استخراج ویژگی، بهینه‌سازی، پیش‌بینی، تصمیم‌گیری) که می‌توان الگوریتم‌های متفاوتی را برای هر یک در نظر گرفت. حال مساله

[1] Comma-separated values
[2] Deep neural networks (DNNs)
[3] Long short–term memory (LSTM)
[4] Recurrent Neural Networks (RNNs)

[Downloaded from jorlu.ir on 2026-11-29]
[DOI: 10.71773/jamlu-2024-1-2167]

مهم این است که قطعا برای به‌دست آوردن نتیجه بهینه نیاز به اعمال تغییرات در لایه‌ها و الگوریتم‌ها می‌باشد، اما از آنجایی که هر یک از این لایه‌ها وظیفه خاصی را دنبال می‌کنند تشخیص این‌که در نتیجه‌گیری کل سیستم تاثیر مثبت داشته باشد یا نه، بسیار دشوار است. با این تفسیر رویکرد جدیدی با عنوان آموزش پایان به پایان E2E ارایه گردید. این روش مبتنی بر یادگیری عمیق می‌باشد که از زیرمجموعه‌های یادگیری ماشین و هوش مصنوعی بوده و اشاره به یک یادگیری پیچیده دارد که هدف آن دور زدن لایه های میانی است. این روش با استفاده از ساختار شبکه‌های عمیق DNN می‌تواند محاسبات پیچیده ای را بر روی مجموعه داده‌های بزرگ انجام دهد و با تکنیکی به‌ظاهر ساده با ایجاد یک خط پردازش پیچیده، لایه‌های میانی آموزش را محو می‌کند [۱، ۴، ۱۳]. مدل ارایه‌شده در این تحقیق نیز در چارچوب E2E آموزش داده شده است.

۳-۲-۲ شبکه‌های عصبی عمیق

بخش آموزش و یادگیری در ساخت مدل‌ها باعث توسعه شبکه‌های عصبی مصنوعی گردید. سپس این شبکه‌ها برای ذخیره و ارزیابی میزان اهمیت هر یک از ورودی‌ها نسبت به خروجی نیاز به مکانی جهت ذخیره‌سازی این اطلاعات داشتند که موجب تشکیل لایه‌های پنهان در دل این شبکه‌ها گردید. شبکه‌های عصبی عمیق با استفاده از مدل‌سازی پیچیده ریاضی، داده‌ها را به روش‌های پیچیده پردازش می‌کنند. در واقع این شبکه‌ها نتیجه تکامل این مسیر در حوزه هوش مصنوعی می‌باشند[۱۹].

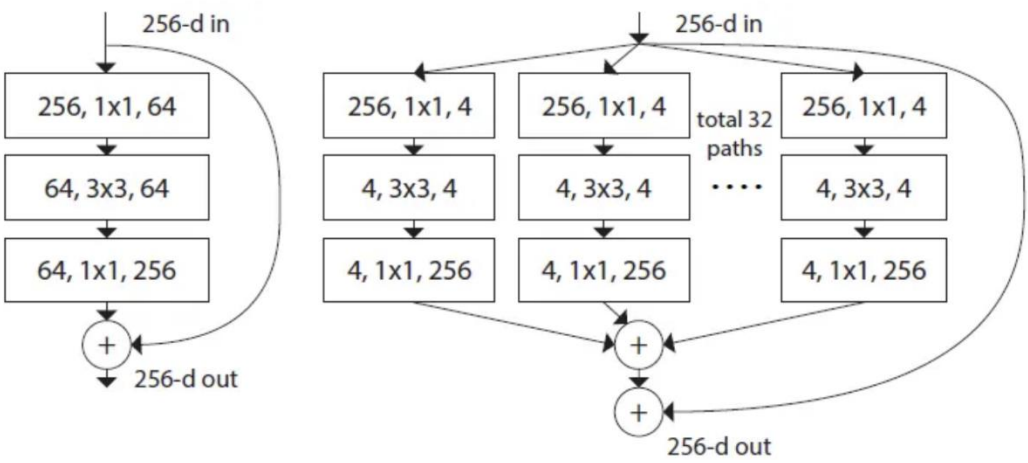
در این تحقیق از میان الگوریتم‌های متعدد شبکه‌های عصبی، دو نوع الگوریتم پیچشی و بازگشتی را برای مدل ها در نظر گرفتیم و از این بین معماری‌هایی را برگزیدیم که تا این زمان در هیچ یک از تحقیقات مشابه از آنها استفاده نگردیده است. در ادامه معماری‌هایی را که در قالب دو نوع شبکه فوق برای مدلسازی در این تحقیق برازش شده است را شرح خواهیم داد.

معماری ResNext: یکی از معماری‌های انتخاب‌شده برای این تحقیق، شبکه ResNext بود که از شبکه‌های نسبتا جدید و مبتکرانه‌ای است که در سال ۲۰۱۷ توسط تحقیقات هوش مصنوعی فیس بوک معرفی شد. این شبکه بر اساس نقاط قوت و ضعف شبکه های VGG و ResNet است. در واقع ResNext نتیجه ترکیب یک استراتژی تکرار ResNet با یک استراتژی تقسیم-تبدیل-ادغام شبکه Inception است. به عبارت دیگر یک بلوک شبکه ورودی را تقسیم کرده و پس از تبدیل به فرمت مورد نیاز، آن را ادغام می‌کند تا خروجی را در جایی که هر بلوک از توپولوژی یکسانی پیروی می‌کند به دست آورد. یعنی شبکه ResNext در مقایسه با ResNet، یک فراپارامتر به نام کاردینالیته[1] را برای تنظیم ظرفیت مدل معرفی می‌کند که از الگوی تقسیم-تبدیل-ادغام مدل Inception استفاده می‌کند. شکل ۲ نمای کلی از معماری ResNext را در مقایسه با ResNet، به ویژه در گلوگاه‌ها و تبدیل‌های جمع‌آوری‌شده، ارایه می‌دهد[۲۰].

[1] cardinality

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]



شکل ۲. بلوک باقیمانده در ResNet (سمت چپ)، بلوک باقیمانده در ResNext با Cardinality = 32 (سمت راست) [۲۰].

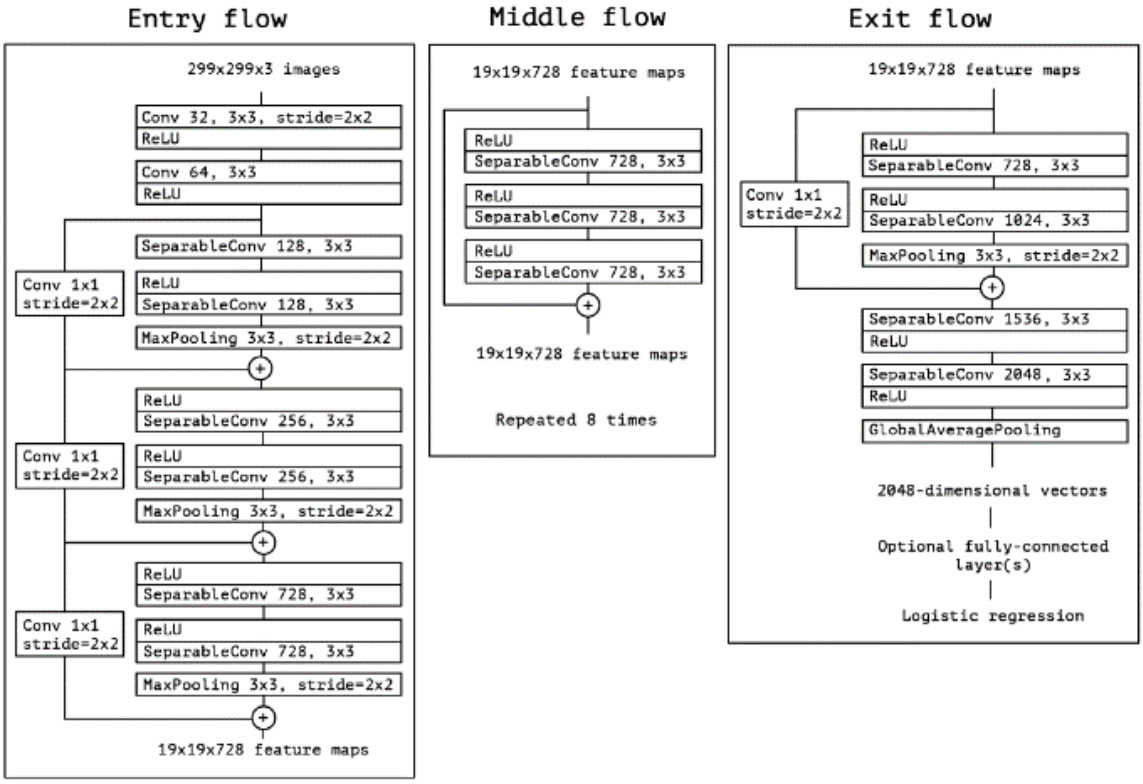
معماری Xception: دومین معماری انتخاب‌شده برای این تحقیق، شبکه Xception بود. این شبکه که مخفف Extreme version of Inception، نسخه بهبودیافته Inception V3 است، در سال ۲۰۱۷ توسط Chollet، خالق کتابخانه Keras Deep Learning معرفی شد. این یک معماری شبکه عصبی پیچشی اصلاح‌شده با پیچیدگی عمیق قابل تفکیک، بر اساس لایه‌های عمیق پیچیده قابل تشخیص است. در واقع، این فرضیه بیان می‌کند که نگاشت همبستگی بین کانالی را می‌توان به طور کامل از همبستگی‌های فضایی در نقشه‌های ویژگی شبکه عصبی کانولوشن جدا کرد. در مقایسه با کانولوشن معمولی، ما نیازی به کانوالو در همه کانال‌ها نداریم. این به معنای اتصالات کمتر و مدل سبک‌تر است. در این شبکه پردازش داده‌ها در سه مرحله انجام می‌شود. ابتدا داده از جریان ورودی عبور می‌کند و سپس با عبور از جریان میانی هشت بار خود را تکرار می‌کند و در نهایت از جریان خروجی خارج می‌شود. همان‌طور که در شکل زیر مشاهده می‌شود، SeparableConv یک پیچیدگی قابل تشخیص عمیق است. می‌بینیم که SeparableConv ها به عنوان ماژول‌های اولیه در نظر گرفته می‌شوند و در کل معماری یادگیری عمیق تعبیه شده‌اند[۲۱]. شرح کامل عملکرد و تفسیر شبکه در مقاله [۲۱] ارایه‌شده است.

شبکه LSTM: بدون شک وقتی از رانندگی خودکار صحبت می‌کنیم، پارامتر زمان در همه ابعاد بسیار مهم است. به خصوص تشخیص شرایط محیطی و پیش‌بینی تغییرات احتمالی در واحد زمان تاثیر زیادی بر تصمیم مدل در رانندگی خودران خواهد داشت. به زبان ساده‌تر، بررسی تصاویر و ویژگی های استخراج‌شده با در نظر گرفتن توالی زمانی آنها توسط مدل می‌تواند در افزایش دقت مدل موثر باشد. بنابراین، یک روش مبتنی بر گرادیان کارآمد به نام حافظه کوتاه مدت بلند (LSTM) می‌تواند این تقاضا را برآورده کند[۲۲]. در شکل ۴ نشان داده شده است که این نوع شبکه بازگشتی دارای واحدی می‌باشد که با یک ساختار زنجیره‌ای اطلاعات را در طول زمان به خاطر می‌سپارد در واقع ضعفی که در شبکه‌های برگشتی در قالب محوشدگی گرادیان[1] شناخته می‌شود با تکنیک حافظه کوتاه مدت بلند پوشش داده شده است. بر اساس الگوهای آموزشی تعریف‌شده در این تحقیق اثر به‌کارگیری این تکنیک مشهود است[۲۳].

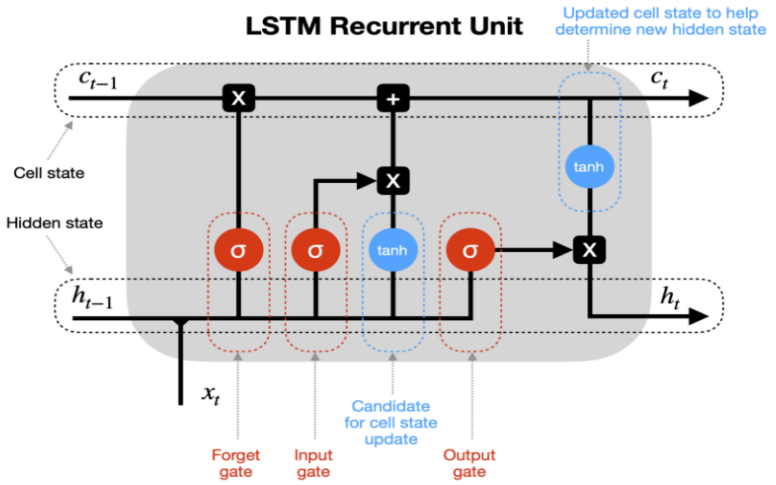
[1] Vanishing Gradient

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]



شکل ۳. معماری شبکه Xception [۲۱].



شکل ٤. واحد برگشتی حافظه کوتاه مدت بلند LSTM. [Image by Dobilas, S. https://towardsdatascience.com]

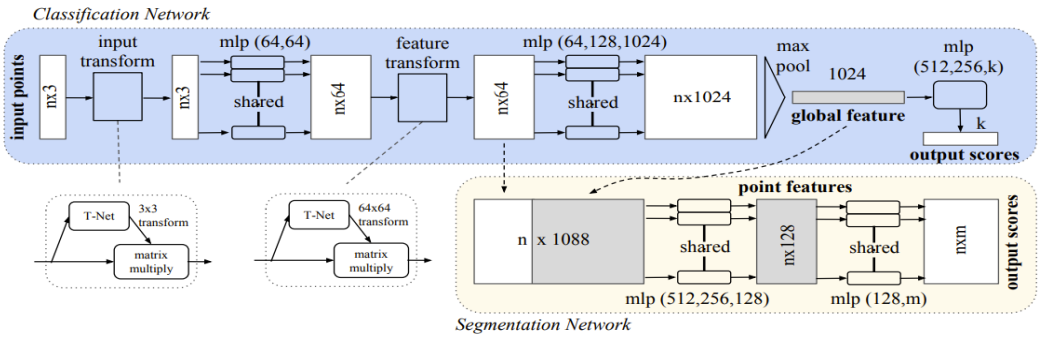
۳-۲-۳ آماده‌سازی اطلاعات عمق

روش‌های مختلفی برای استفاده از داده‌های خام جمع‌آوری‌شده از حسگرهای لیدار یا نقاط ابری در الگوریتم‌های آموزشی وجود دارد. اما برای این‌که بتوانیم عملکرد معماری‌های مختلف را به طور دقیق با آثار مشابه مقایسه کنیم، تصمیم گرفتیم از همان تبدیل‌ها به صورت زیر استفاده کنیم. همچنین استفاده از روش‌های بهینه در تبدیل نقاط ابری می‌تواند پیشنهاد خوبی برای کارهای آینده باشد.

[Downloaded from jorlu.ir on 2026-11-29]

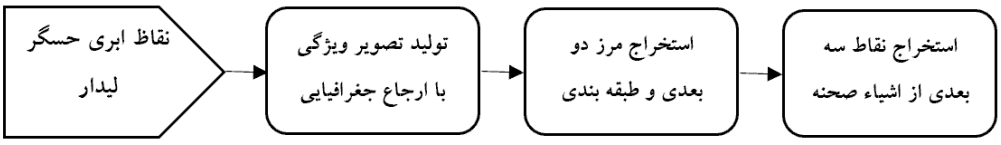
[DOI: 10.71773/jamlu-2024-1-2167]

POINTNET: این شبکه که توسط چارلز و همکاران در سال ۲۰۱۷ معرفی شد، توانست نقاط ابری پراکنده را به عنوان ورودی مستقیم و با دقت قابل قبولی دریافت کند و ویژگی‌های خاصی را در قالب تصاویر معمولی به عنوان خروجی ارایه دهد. قابل ذکر است که اگر بخواهیم از روش آموزش متوالی استفاده کنیم، باید فاصله بین اجسام تا خودرو را با اتخاذ روشی از طریق داده‌های خام محاسبه و در اختیار مدل قرار دهیم. اما در رویکرد End to End هر خروجی که از شبکه PointNet در قالب یک ویژگی خاص و به صورت جعبه سیاه غیرقابل درک برای ما در اختیار مدل قرار می‌گیرد می‌تواند تعریفی از فواصل اشیاء برای مدل باشد[۲۴].



شکل ۵. ساختار معماری شبکه PointNet . [۲۴]

PCM[1]: این روش که در تحقیق بیشنگ یانگ و همکاران ارایه شده است می‌تواند در سه مرحله زیر اطلاعات لازم را از نقاط ابری استخراج کند که عبارتند از: (۱) تعیین وزن نقاط، (۲) تولید تصویر ویژگی جغرافیایی مرجع (تصاویر خاکستری) و (۳) تقسیم‌بندی تصویر به سمت استخراج شی از تصویر ویژگی. تصویر مشخصه ارجاع‌شده جغرافیایی ایجادشده از نقاط ابری متحرک می‌تواند توزیع فضایی نقاط اسکن‌شده را ارایه دهد و ویژگی‌های هندسی محلی مهم اشیاء صحنه خیابان را که انتظارات اصلی ما از نقاط ابری هستند حفظ کند[۲۵]. برای به‌دست آوردن اطلاعات عمق، شن و همکاران در [۱۰] از این الگوریتم پیشنهادی برای ترسیم نقاط ابری روی داده های لیدار و ارایه مدل اصلی استفاده کردند.



شکل ٦. نمودار جریان روش پیشنهادی در الگوریتم PCM. [۲۵]

۳-۲-٤ ارزیابی مدل

ارزیابی مدل‌ها بر خلاف خیلی از تحقیقات این حوزه که به‌صورت گسسته و در بهترین حالت برای چهار فرمان حرکت رو به جلو و یا توقف و گردش به راست یا چپ انجام می‌شود، در این تحقیق پیش‌بینی از نوع پیوسته

[1] Point Clouds Mapping

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

است. یعنی برای میزان سرعت خودرو و چرخش زاویه فرمان عدد دقیق پیش‌بینی می‌گردد که مبتنی بر مقایسه خط مشی پیش‌بینی‌شده در خصوص تعیین سرعت و زاویه فرمان خودرو با برچسب مربوطه در داده‌های آزمایشی خواهد بود. از آنجا که در گزارش عملکرد و مهارت یک مدل رگرسیون باید میزان خطا در پیش‌بینی‌ها ارایه گردد و نمی‌توان صحت را محاسبه کرد، امکان محاسبه مستقیم صحت در خصوص مقایسه اعداد پیش‌بینی‌شده با برچسب‌های مورد نظر میسر نمی‌باشد. لذا جلوتر توضیح داده شده است چگونه با تعریف آستانه حد یا به عبارتی با تعریف تلرانسی برای پذیرفته‌شدن اعداد پیش‌بینی‌شده در خصوص دو پارامتر مورد نظر این امکان را میسر می‌کنیم. در واقع ما به‌دنبال آن هستیم که ببینیم پیش‌بینی‌ها چقدر به واقعیت نزدیک است و پیش‌بینی رفتار راننده برای رانندگی در شرایط واقعی مناسب و قابل قبول خواهد بودیا خیر. به همین دلیل یکی از سه معیار پر کاربرد در محاسبه خطا در مدل‌های رگرسیون، که ریشه میانگین مربعات خطا[1] است در این تحقیق نیز مورد استفاده قرار گرفته شده است چرا که استدلال می‌شود با بزرگ‌نمایی مقدار خطا به موازات تعریف آستانه حد حداکثر سختگیری در محاسبه خطا و دقت مدل انجام می‌شود و از این‌رو پیش بینی‌ها قابل اعتمادتر خواهد بود به عبارت دیگر پیش‌بینی قرار گرفته‌شده در بازه آستانه حد با توجه به بزرگ‌نمایی محاسبه خطا بوده و این استدلال قابلیت اعتماد به مدل را افزایش می‌دهد[۲۶].

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{obs,i} - X_{model,i})^2}{n}} \quad (1)$$

برای محاسبه ریشه‌ی خطای میانگین مربعات در رابطه فوق، X_{obs} مقدار واقعی و X_{model} مقدار پیش‌بینی‌شده و n تعداد نقاط داده غیر از دست‌رفته (تعداد پیش بینی) می‌باشد.

لذا از آنجا که این نوع ارزیابی معیار بصری مشخصی در خصوص عملکرد مدل جهت مقایسه القا نمی‌کند، و ما به‌دنبال بستری مناسب جهت مقایسه نتایج با کارهای مشابه بودیم در کنار محاسبه خطا، با استفاده از تعریف آستانه حد شرایط را برای تعیین صحت مدل نیز فراهم کردیم. بر این اساس باید میزان انحراف قابل قبول از نتایج مطلوب را تعیین می‌نمودیم. لذا جهت ایجاد شرایط یکسان برای ارزیابی و مقایسه با کارهای مشابه، میزان انحراف قابل قبول برای زاویه فرمان را تا ۶ درجه و برای سرعت خودرو تا ۵ کیلومتر در ساعت در نظر گرفتیم. یعنی در واقع کلیه اعداد پیش‌بینی‌شده در بازه آستانه حد به‌عنوان پیش بینی صحیح در نظر گرفته می‌شود که بر این اساس از تقسیم تعداد پیش‌بینی‌های درست برکل پیش‌بینی‌ها می‌توان به صحت[2]مدل دست یافت[۱۰].

۳-۲-۵ بهینه‌سازی مدل

در این تحقیق از الگوریتم بهینه‌سازی آدام استفاده شده است. در این الگوریتم برای به‌روزرسانی وزن‌های شبکه به‌جای استفاده از روش کلاسیک گرادیان نزولی تصادفی[3]، شبکه بر اساس داده‌های ورودی آموزش فرایند

[1] Root Mean Square Error (RMSE)
[2] Accuracy
[3] Stochastic gradient descent (SGD)

[Downloaded from jorlu.ir on 2026-11-29]
[DOI: 10.71773/jamlu-2024-1-2167]

به‌روزرسانی را انجام می‌دهد. نام این الگوریتم برگرفته از اصطلاح تخمین گشتاور تطبیقی[1] است که در آن برای به‌روزرسانی وزن‌های آموزش شبکه از روش تکرار در داده‌های آموزشی استفاده می‌شود. این الگوریتم در واقع ترکیبی از مزایای دو الگوریتم محبوب Adagrad و RMSprop می‌باشد.

الگوریتم Adagrad با ثابت نگه داشتن نرخ یادگیری عملکرد قابل قبولی با گرادیان‌های پراکنده (گرادیان‌های نامتراکم) دارد. این پراکندگی به‌شدت در مسایل بینایی کامپیوتر و پردازش زبان طبیعی قابل مشاهده است.

AdaGrad (۲)

گام اول:
$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{v_t + \epsilon}} \cdot \frac{\partial L}{\partial w_t}$$

گام دوم:
$$v_t = v_{t-1} + \left[\frac{\partial L}{\partial w_t}\right]^2$$

در رابطه فوق در گام اول، که به‌روزرسانی پارامتر ها انجام می شود، α نرخ یادگیری، v_t تخمین گرادیان مربعی دوم، و ϵ اپسیلون را خواهیم داشت. پس از آن در گام دوم نوبت به تکرار همگرایی خواهد بود که در آن $\partial L/\partial w_t$ نمایانگر گرادیان نسبت به پارامتر w_t است.

الگوریتم RMSprop نیز با حفظ نرخ یادگیری پارامترها که از طریق تطبیق میانگین گرادیان‌های مرتبط با وزن‌ها به‌دست می‌آید که در تنظیمات برخط و مسایلی که نویز زیادی دارند عملکرد بهینه‌ای دارد.

RMS Prop (۳)

گام اول:
$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{v_t + \epsilon}} \cdot \frac{\partial L}{\partial w_t}$$

گام دوم:
$$v_t = \beta v_{t-1} + (1+\beta)\left[\frac{\partial L}{\partial w_t}\right]^2$$

در رابطه فوق در گام اول، مجدد به‌روزرسانی پارامترها انجام می‌شود، α نرخ یادگیری، v_t تخمین گرادیان مربعی دوم، و ϵ اپسیلون را خواهیم داشت. پس از آن در گام دوم نوبت به تکرار همگرایی خواهد بود که در آن $\partial L/\partial w_t$ نمایانگر گرادیان نسبت به پارامتر w_t بوده و β یک پارامتر تطابق است به آن اضافه شده است.

یکی از دلایل اصلی انتخاب الگوریتم آدام، متناسب بودن مزایای این الگوریتم با چالش‌های پیش رو در آموزش مدل این تحقیق بوده است. حجم بالای داده‌ها و پیاده‌سازی ساده و آسان با حداقل اشغال حافظه و توان محاسباتی کارآمد و متناسب بودن با مسایل دارای گرادیان های با نویز زیاد. همه و همه از مزایای کاربردی این الگوریتم است. همچنین آدام یک تکنیک بهینه‌سازی تصادفی کارآمد است که فقط به گرادیان‌های مرتبه اول نیاز دارد و حافظه بسیار کمی را اشغال می‌کند. در این روش برای پارامترهای مختلف، نرخ یادگیری تطبیقی

[1] Adaptive moment estimation

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

فردی[1] با تخمین لحظه‌های اول و دوم گرادیان محاسبه می‌شود. مزیت دیگر این تکنیک این است که مقدار به‌روزرسانی پارامتر در مقایسه با مقیاس مجدد گرادیان ثابت است. چهار پارامتر اصلی وظیفه پیکربندی الگوریتم بهینه‌سازی آدام را بر عهده دارند. اولین پارامتر نسبتی است که به‌روزرسانی وزن‌ها بر اساس آن انجام می‌پذیرد، که به آن نرخ یادگیری[2] و یا طول گام گفته می‌شود. دومین و سومین پارامتر که برای تخمین‌های گشتاور[3] اول و لحظه دوم کاربرد دارد، نرخ‌های فروپاشی نمایی[4] هستند. و آخرین پارامتر هم عددی بسیار کوچک است که برای جلوگیری از تقسیم بر صفر استفاده می‌شود. عملکرد این الگوریتم در روابط (۴) معرفی شده است [۲۷].

(۴)

Adam

$g_t \leftarrow \nabla_\theta f_t(\theta_t - ۱)$ ← گام اول

$m_t \leftarrow B_۱.M_{t-۱} + (۱ - B_۱).g_t$ ← گام دوم

$u_t \leftarrow B_۲.u_{t-۱} + (۱ - B_۲).g_t^{۲}$ ← گام سوم

$\hat{m}_t \leftarrow m_t / (۱ - \beta_۱^t)$ ← گام چهارم

$\hat{u}_t \leftarrow u_t / (۱ - \beta_۲^t)$ ← گام پنجم

$\theta_t \leftarrow \theta_{t-۱} - \alpha.\hat{m}_t / (\sqrt{\hat{u}_t} + \in)$ ← گام ششم

t= شمارنده زمان

m_t = تخمین گرادیان مربعی اول

= تخمین گرادیان مربعی دوم

θ_t = مقدار اولیه پارامترها

$B_۱$ و $B_۲$ = پارامترهای تطابق هستند که نشان‌دهنده نقش میزان تطابق گرادیان‌های گذشته در به‌روزرسانی وزن‌ها است .

ϵ = یک مقدار بسیار کوچک است که به جلوگیری از تقسیم بر صفر در مخرج عبارت تقسیم کننده در روابط کمک می‌کند

در روابط فوق گام اول نشان‌دهنده گرادیان تابع هدف نسبت به پارامترهای مدل در زمان $\theta_t - ۱$ است. این گرادیان نقش اصلی را در فرآیند به‌روزرسانی پارامترها با توجه به اطلاعات جاری داده‌ها ایفا می‌کند. در این مرحله گرادیان ها بدون هدف و کاملا تصادفی در بازه زمانی t دریافت می‌شوند. گام دوم تخمین گرادیان مربعی اول، گام سوم تخمین گرادیان مربعی دوم، گام چهارم تصحیح گرادیان مربعی اول، گام پنجم تصحیح گرادیان مربعی دوم و در نهایت در گام آخر به‌روزرسانی پارامترها را خواهیم داشت.

[1] Individual adaptive learning rates
[2] learning rate
[3] moment estimate
[4] Exponential decay rates

[Downloaded from jorlu.ir on 2026-11-29]

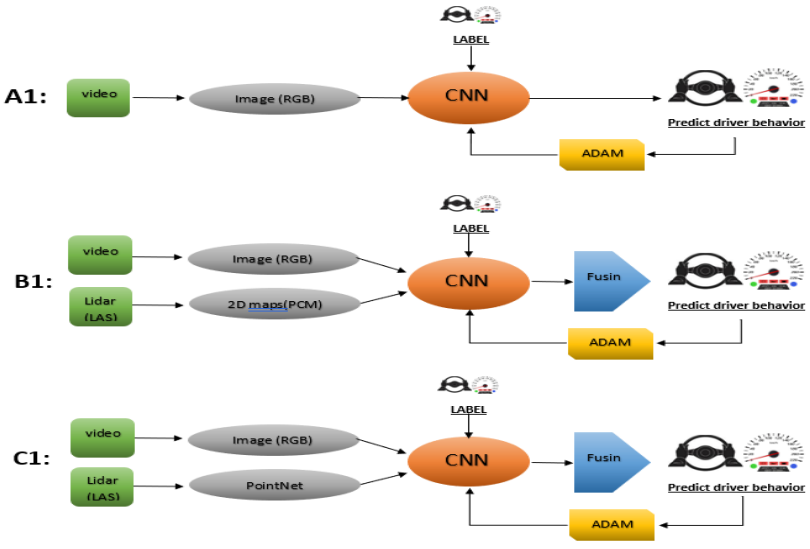
[DOI: 10.71773/jamlu-2024-1-2167]

در معرفی عملکرد الگوریتم آدام در رابطه فوق، در مرحله اول گرادیان ها بدون هدف و کاملا تصادفی در بازه زمانی t دریافت می‌شود. در مرحله دوم به‌روزرسانی تخمین گشتاور اول انجام می‌شود و در مرحله سوم بروزرسانی تخمین گشتاور دوم و مرحله چهارم، محاسبه بایاس برای تخمین گشتاور اول و محاسبه بایاس برای تخمین گشتاور دوم در مرحله پنجم انجام می‌شود. در نهایت در مرحله ششم به‌روزرسانی پارامتر ها انجام می‌پذیرد.

۳-۲-٦ تعیین الگوی آموزش

همان‌طور که اشاره شد، رویکرد انتخاب‌شده برای یادگیری مدل در خصوص پیش‌بینی رفتار راننده در این تحقیق، روش End to End است. به این ترتیب، مدل به‌طور خودکار بازنمایی‌های داخلی مراحل پردازش، مانند شناسایی ویژگی‌های مفید جاده را می‌آموزد. نکته ای که در این قسمت توضیح داده خواهد شد الگوریتم انتخابی برای آموزش بر اساس ورودی های مختلف و در نظر گرفتن وابستگی‌های زمانی در داده های آموزشی است. بر این اساس دو الگو در نظر گرفته شده است. و برای هر یک از الگوها سه نوع ترکیب ورودی، به شرح زیر تعیین شده است و در نهایت برای هر یک از معماری‌های معرفی‌شده ۶ بار آموزش انجام شده است که بر اساس آن ۱۲ نتیجه مختلف برای معرفی صحت مدل‌ها در بخش ارایه نتایج خواهیم داشت.

الگوی اول: در این الگو آموزش تنها از طریق الگوریتم‌های شبکه‌های عصبی عمیق پیچشی بدون در نظر گرفتن وابستگی زمانی انجام می‌شود. سه نوع ترکیب ورودی برای این مدل در نظر گرفته شده است. در حالت اول A1، فقط تصاویر RGB که از داده‌های ویدئویی استخراج شده‌اند. در حالت دوم B1، تصاویر RGB با اطلاعات عمقی که از طریق شبکه PointNet پردازش می‌شوند و در حالت سوم C1، تصاویر RGB با اطلاعات عمقی که با نگاشت داده‌های LIDAR دریک شبکه عصبی عمیق با عنوان PCM پردازش می‌شوند. هر یک از این ترکیب ها به‌عنوان بردار ورودی با برچسب زاویه فرمان و سرعت خودرو همراه می‌باشد.

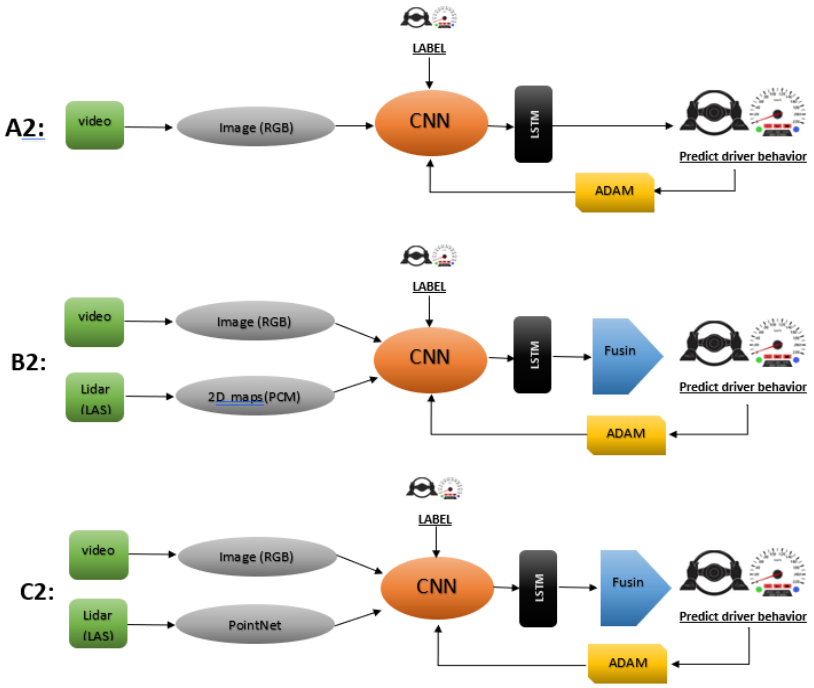


شکل۷. الگوی اول در الگوریتم آموزش مدل

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

الگوی دوم: در الگوی دوم ترکیبی از شبکه‌های کانولوشنی و بازگشتی را خواهیم داشت و بر این اساس همان ترکیب ورودی‌های مدل اول حفظ شده است، با این تفاوت که وابستگی زمانی برای آموزش مدل در قالب الگوریتم شبکه LSTM برای هر سه حالت در نظر گرفته شده است. سه ترکیب ورودی برای این الگو با عنوان A2, B2, C2 ارایه شده است.



شکل۸. الگوی دوم در الگوریتم آموزش مدل

٤ پیاده‌سازی و تحلیل نتایج

کلیه فرایند تحقیق شامل آماده‌سازی مجموعه داده و مدلسازی در بستر زبان برنامه‌نویسی پایتون صورت گرفته و همچنین آموزش مدل‌ها و انجام تست‌ها در فضای رایانش ابری انجام شده است. در این فرایند پس از آماده‌سازی مجموعه داده معرفی‌شده در بخش ۲-۲ و همچنین ساخت انواع مدل‌های انتخاب‌شده در بخش ۳-۲ و نیز تعیین سیاست مدنظر برای بردارهای ورودی در قالب الگوهای آموزش، اعتبار مدل‌های پیشنهادی در پیش‌بینی رفتار راننده برازش گردید. در این مرحله هر یک از معماری‌ها را در قالب دو مدل (شبکه عصبی عمیق و شبکه عصبی عمیق به‌همراه حافظه کوتاه مدت بلند) و با سه نوع بردار ورودی که در بخش الگوی آموزش شرح داده شده است آموزش داده و پس از آن با ۲۰ درصد از داده‌ها باقیمانده که برای تست در نظر گرفته شده بود، مدل‌های آموزش دیده را در گام آزمایش قرار دادیم که نتایج به‌دست آمده در خصوص صحت مدل‌ها در چهارچوب آستانه حد تعریف‌شده، به‌شرح جدول زیر می‌باشد:

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

جدول ۲. ارزیابی عملکرد الگوریتم‌های پیشنهادی با ورودی‌های مختلف

شبکه‌ها	ارزیابی مدل‌ها از نظر صحت	DNN			DNN+LSTM		
		A1	B1	C1	A2	B2	C2
Xception	سرعت خودرو	۶۹/۷	**۷۹/٤**	۶۸/۸	۷۴/۲	۷۷/۶	۷۹/۱
	زاویه فرمان	۷۰/۱	۸۱/۲	۷۶/۴	۷۹/۳	۸۰/۸	**۸۳/٤**
ResNext	سرعت خودرو	۷۲/۶	۷۸/۵	۷۱/۷	۷۵/۸	۷۵/۹	**۸۱/٦**
	زاویه فرمان	۶۸/۲	۷۹/۹	۶۹/۳	۷۸/۷	۷۹/۲	**۸٤/۳**

نتایج حاصله در ستون‌های A برای هر دو نوع معماری به معنای عملکرد مدل‌های فاقد اطلاعات عمق می‌باشد و همان‌طور که انتظار داشتیم قطعا باید دقت کمتری نسبت به دیگر مدل‌ها را ارایه می‌نمود. البته مقایسه ستون‌های A1 و A2 این استدلال را نیز به ما می‌دهد که هر زمان از تصاویر به تنهایی به‌عنوان ورودی برای مدل استفاده می‌نماییم، اطلاعات توالی تصاویر که حاصل به‌کارگیری حافظه کوتاه مدت بلند است نتایج بهینه‌تری را در اختیار ما قرار می‌دهد. اما نتایج ستون‌های B و C مشمول اطلاعات عمق یا در واقع شناسایی فواصل اشیاء می‌باشد و ضمن اینکه اهمیت وجود اطلاعات عمق را در مقایسه با نتایج ستون های A آشکار می‌کند، بیان‌کننده چالش به‌کارگیری این اطلاعات نیز می‌باشد. تفاوت نتایج ما بین ستون‌های B و C نشان می‌دهد که تکنیک‌های به‌کارگیری اطلاعات عمق می‌تواند تاثیر فراوانی بر نتایج داشته باشد. به‌عنوان مثال وقتی از تکنیک PCM برای استخراج اطلاعات عمق در مدل‌های DNN+LSTM استفاده کردیم، بر خلاف انتظار باعث کاهش دقت نسبت به مدل‌های DNN گردید. ولی در مقابل تکنیک PointNet در مدل‌های DNN+LSTM به‌خوبی جواب داده و بهینه‌ترین نتایج را منتج شده است. این در حالی می‌باشد که به‌کار گیری تکنیک PCM در مدل‌های DNN کارایی بهتری نسبت به تکنیک PointNet نشان داده بود. بنابراین نتیجه می‌گیریم استخراج اطلاعات عمق بشیوه الگوریتم PCM زمانی که می‌خواهیم از الگوریتم LSTM استفاده نماییم نمی‌تواند نتایج مطلوبی را در پی داشته باشد. لذا به‌کارگیری روش‌های استفاده از اطلاعات عمق خود به تنهایی نیز می‌تواند بستر فراوانی جهت انجام تحقیقات در آینده باشد. اما نکته برجسته در بررسی نتایج عملکرد بهتر مدل‌های زیرستون DNN+LSTM نسبت به مدل های DNN بوده است. بی شک این نتایج گویای اثر بخشی توالی‌های زمانی در تصمیم‌گیری مدل می‌باشد. هر چند روش آموزش E2E خارج از قدرت تحلیل انسانی می‌باشد؛ ولی مشاهده می‌شود هر چه اطلاعات بیشتری در اختیار این جعبه سیاه آموزشی قرار گیردکه امکان استخراج ویژگی‌هایی نظیر فواصل اشیاء و همچنین وابستگی‌های زمانی را بتواند فراهم نماید، قطعا نتایج مطلوب‌تری حاصل خواهد شد.

۵ نتیجه‌گیری و پیشنهادات

در این مطالعه برای پیش‌بینی رفتار راننده در تعیین میزان سرعت خودرو و زاویه فرمان، مدل‌های متفاوتی ارایه گردید که نتایج حاصل هر یک از مدل‌ها گویای تاثیرگذاری برخی از عوامل بسیار مهم در دستیابی به این هدف بودند. ابتدا دو مورد از معماری‌های جدید و موفق در حوزه یادگیری E2E که در هیچ یک از کار های مشابه

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

در خودرو های خودران استفاده نشده بود با استفاده از یک مجموعه داده بسیار کامل مورد آزمایش قرار گرفت که نتایج مطلوب‌تری نسبت به کارهای مشابه حاصل گردید. همچنین نتایج این مطالعه به‌وضوح نشان می‌دهد دو عامل اصلی، یکی اطلاعات عمق که در قالب دو نوع الگوریتم به‌کار گرفته شده و دیگری اثر بخشی وابستگی زمانی که در ترکیب الگوریتم LSTM با مدل‌های پایه اعمال گردید اثر مطلوبی در آموزش مدل‌ها داشته است. بنابراین نتایج این تحقیق و کارهای مشابه نشان می‌دهد با افزایش روز افزون مجموعه داده‌های این حوزه و به‌کارگیری تکنیک‌های متفاوت در پیاده‌سازی آموزش E2E می‌تواند زمینه‌ی اعتمادسازی در خصوص به‌کارگیری این روش در خودروهای خودران را فراهم آورد. همچنین بر اساس تجربیات به‌دست آمده در این تحقیق می‌توان سه محور تحقیقاتی برای کارهای آینده پیشنهاد داد. ابتدا در خصوص جمع‌آوری مجموعه داده بومی و متناسب با شرایط ترافیکی کشور و دوم تمرکز بروی الگوریتم‌های استخراج ویژگی برای داده‌های خام لیدار که می‌توانند تاثیرات قابل توجهی بروی نتایج مدل داشته باشند. و در نهایت برای بهینه‌سازی مدل، ارایه ترکیبی از ژنتیک الگوریتم و الگوریتم آدام در تعیین پارامترهای اصلی پیکربندی می‌تواند نوآوری قابل توجهی باشد.

قدردانی

با سپاس از خانم دکتر ییپینگ چن از تیم DBNet که در تهیه مجموعه داده ما را یاری نمودند.

منابع

[1] Kuutti.S., Fallah.S., Bowden.R., Barber.P. (2019). Deep Learning for Autonomous Vehicle Control, Algorithms, State-of-the-Art and Future Prospects. University of Waterloo. Copyright©by Morgan & Claypool

[2] Kocić. J., Jovičić. N., Drndarević.V. (2019). An End-to-End Deep Neural Network for Autonomous Driving Designed for Embedded Automotive Platforms. MDPI, Sensors 2019, 19(9), 2064; https://doi.org/10.3390/s19092064

[3] Chen. C., Seff. A., Kornhauser. A., and Xiao. J.(2015). Deepdriving: Learning affordance for direct perception in autonomous driving. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), ICCV '15, pages 2722– 2730, Washington, DC, USA, 2015. IEEE Computer Society.

[4] Xu. H., Gao. Y., Yu. F., and Darrell. T.,(2016). End-to-end learning of driving models from large-scale video datasets. arXiv:1612.01079 [cs.CV]. CoRR, abs/1612.01079, 2016.

[5] Pfeiffer. M., Schaeuble. M., Nieto. J., Siegwart. R., and Cadena. C., (2018) From Perception to Decision: A Data-driven Approach to End-to-end Motion Planning for Autonomous Ground Robots. arXiv:1609.07910v3 [cs.RO] .

[6] Codevilla. F., Lopez. A.M., Koltun. V., and Dosovitskiy. A., (2018) On Offline Evaluation of Vision-based Driving Models. arXiv:1809.04843v1 [cs.CV].

[7] Bojarski. M., Testa. D. D., Dworakowski. D., Firner. B., Flepp. B., Goyal. P., Jackel. L.D., Monfort. M., Muller. U., Zhang. J., Zhang. X., Zhao. J., and Zieba. K., (2016) End to end learning for self-driving cars. CoRR, abs/1604.07316, 2016.

[8] Eraqi. H.M., Moustafa. M.N., and Honer. J., (2017) End-to-End Deep Learning for Steering Autonomous Vehicles Considering Temporal Dependencies. arXiv:1710.03804v3 [cs.LG].

[9] Chen. J., Eben. S., Tomizuka. Li. M., (2020) Interpretable End-to-end Urban Autonomous Driving with Latent Deep Reinforcement Learning. arXiv:2001.08726v3 [cs.RO].

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]

[10] Chen. Y., Wang. J., Li. J., Lu. C., Luo. Z., Xue. H., Wang.C., (2018) Lidar-video driving dataset: Learning driving policies effectively. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 5870–5878
[11] Codevilla. F., Muller. M., Lopez. A., Koltun. V., Dosovitskiy. A., (2018) End-to-end Driving via Conditional Imitation Learning. arXiv:1710.02410v2 [cs.RO].
[12] Wang. Y., Liu. D., Jeon. H., Chu. Z., and Matson. E., (2019). End-to-end Learning Approach for Autonomous Driving: A Convolutional Neural Network Model. n Proceedings of the 11th International Conference on Agents and Artificial Intelligence (ICAART 2019), pages 833-839 ISBN: 978-989-758-350-6
[13] Xiao. Y., Codevilla. F., Gurram. A., Urfalioglu. O., Lopez. A. M., (2020). Multimodal End-to-End Autonomous Driving. arXiv:1906.03199v2 [cs.CV].
[14] Ishihara. K., Kanervisto. A., Miura. J., Hautamaki. V., (2021). Multi-task Learning with Attention for End-to-end Autonomous Driving. arXiv:2104.10753v1 [cs.RO].
[15] Prakash. A., Chitta. K., Geiger. A., (2021). Multi-Modal Fusion Transformer for End-to-End Autonomous Driving. arXiv:2104.09224v1 [cs.CV].
[16] Park. M., Kim and S. Park,: A Convolutional Neural Network-Based End-to-End Self-Driving Using LiDAR and Camera Fusion: Analysis Perspectives in a Real-World Environment. MDPI. Electronics 2021, 10(21), 2608; https://doi.org/10.3390/electronics10212608.
[17] Chen. D., Krahenbuhl. P., (2022). Learning from All Vehicles. arXiv:2203.11934v2 [cs.RO].
[18] Yin. H., and Berger. C., (2017). When to use what data set for your self-driving car algorithm: An overview of publicly available driving datasets. IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1-8, doi: 10.1109/ITSC.2017.8317828
[19] Johnson. J., (2020). A hands-on introduction to math, stats, and machine learning / What's a Deep Neural Network?. e-book; https://www.bmc.com/blogs/deep-neural-network.
[20] Xie. S., Girshick. R., Dollar. P., Tu. Z., He. K., (2017). Aggregated Residual Transformations for Deep Neural Networks. arXiv:1611.05431v2 [cs.CV].
[21] Chollet. F., (2017). Xception: Deep Learning with Depthwise Separable Convolutions. arXiv:1610.02357v3 [cs.CV].
[22] Hochreiter. S., Schmidhuber. J.,(1997). LONG SHORT-TERM MEMORY. Neural Computation 9(8):1735- 1780, 1997 - ieeexplore.ieee.org
[23] Donahue. J., Hendricks. L. A., Rohrbach. M., Venugopalan. S., Guadarrama. S., Saenko. K., Darrell. T., (2016). Long-term Recurrent Convolutional Networks for Visual Recognition and Description. arXiv:1411.4389v4 [cs.CV].
[24] Qi. C. R., Su. H., Mo. K., Guibas. L. J., (2017). PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. arXiv:1612.00593v2 [cs.CV].
[25] Yang. B., Wei. Z., Li. Q., and Li. J., (2012). Automated extraction of street-scene objects from mobile lidar point clouds. International Journal of Remote Sensing - INT J REMOTE SENS.
[26] Acharya. S., (2021). What are RMSE and MAE? Published in Towards Data Science. https://towardsdatascience.com/what-are-rmse-and-mae-e405ce230383.
[27] Kingma. D. P., Lei Ba. J., (2017). ADAM: A method for stochastic optimization. arXiv:1412.6980v9 [cs.LG].

[Downloaded from jorlu.ir on 2026-11-29]

[DOI: 10.71773/jamlu-2024-1-2167]