

بسمه تعالی



دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

آزمایشگاه سخت افزار

گزارش پایانی پروژه شماره ۳

سامانه کمک نگهدارنده خط

استاد: جناب آقای دکتر اجلالی

گروه ۵: ترلان بهادری - همیلا میلی

فهرست مطالب

۱	مقدمه	۴
۲	نمونه‌های سخت‌افزاری	۵
۳	پیاده‌سازی نرم‌افزاری	۵
۱-۳	معرفی LKA در MATLAB	۶
۱-۱-۳	Lane Keeping Assist	۶
۲-۱-۳	Vehicle and Environment	۷
۴	متغیرها و خروجی‌های LKA	۱۱
۱-۴	متغیرهای مورد آزمایش	۱۱
۱-۱-۴	موقعیت Y سنسور	۱۲
۲-۱-۴	موقعیت X سنسور	۱۲
۳-۱-۴	ارتفاع سنسور	۱۲
۴-۱-۴	فاصله کانونی دوربین	۱۲
۲-۴	خروجی‌های قابل بررسی	۱۳
۳-۴	عوامل تعیین‌کننده کیفیت سیستم نگاه‌دارنده خط	۱۴
۱-۳-۴	تشخیص سریع خط	۱۴
۲-۳-۴	سرعت پایین در هنگام چرخش	۱۴
۳-۳-۴	تغییر کم زاویه فرمان	۱۴
۵	آزمایشات و نتایج	۱۵
۱-۵	موقعیت Y سنسور	۱۵
۲-۵	موقعیت X سنسور	۱۶
۳-۵	ارتفاع سنسور	۱۸
۴-۵	فاصله کانونی دوربین	۱۹
۶	نتایج	۲۲
۷	منابع	۲۳

شکل ۱ - انواع سیستم‌های اطلاعاتی استفاده شده در خودروهای مجهز به سیستم کمک نگهدارنده خط. همان طور که از شکل مشخص است، در برخی مناطق تنها سنسورهای کوتاه‌برد مورد استفاده هستند. این در حالی است که سنسورهای دوربرد و حتی دوربین در برخی مکان‌ها مورد استفاده قرار می‌گیرند.	۵
شکل ۲ - ساختار سیستم LKA در MATLAB	۶
شکل ۳ - اجزای داخلی Lane Keeping Assist	۷
شکل ۴ - اجزای داخلی Vehicle and Environment	۸
شکل ۵ - تنظیمات سنسور	۱۰
شکل ۶ - تنظیمات دوربین	۱۱
شکل ۷ - موقعیت خودرو در صفحه مختصات	۱۲
شکل ۸ - نمایش متغیرهای نام برده شده بر روی شکل	۱۳
شکل ۹ - اهمیت شناسایی نقاط کور ماشین برای یافتن بهترین مکان برای سنسور و دوربین	۱۴
شکل ۱۰ - نمودارهای مقایسه حالت پیش‌فرض و حالت $y = -0.90m$	۱۶
شکل ۱۱ - میزان دید در حالت پیش‌فرض	۱۷
شکل ۱۲ - میزان دید در حالتی که سنسور در جلوترین بخش ماشین قرار گرفته است	۱۷
شکل ۱۳ - نمودارهای مقایسه حالت پیش‌فرض و حالت $x = 3.70m$	۱۸
شکل ۱۴ - نمودارهای مقایسه حالت پیش‌فرض و حالت $height = 0.77m$	۱۹
شکل ۱۵ - میزان دید در حالت پیش‌فرض	۲۰
شکل ۱۶ - میزان دید در حالتی که فاصله کانونی دوربین برابر با $[1500, 1500]$ pixels است	۲۰
شکل ۱۷ - نمودار مقایسه حالت پیش‌فرض و حالت $focal\ length = [1500, 1500]$ pixels	۲۱
شکل ۱۸ - بزرگنمایی نمودار شکل ۱۷ به منظور نمایش جزئیات	۲۱

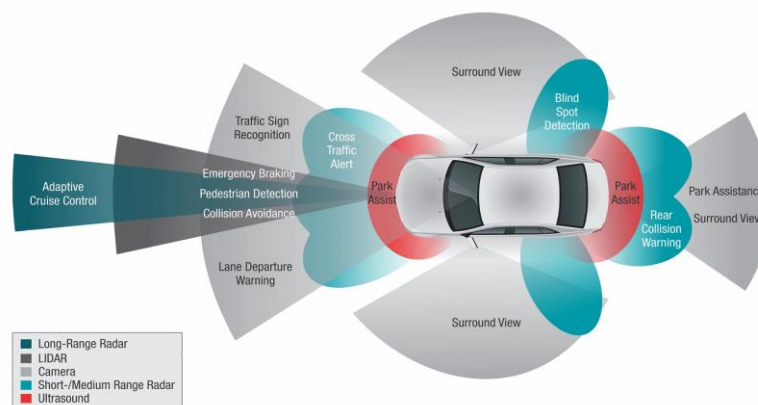
۱ مقدمه

سامانه‌های کمک نگه‌دارنده خط با هدف کمک به راننده برای جلوگیری از انحراف از خط و مسیر طراحی می‌شوند. آنچه در این گونه سامانه‌ها اهمیت دارد، تشخیص به موقع انحراف از مسیر و اعمال زاویه فرمان مناسب برای برگرداندن راننده به مسیر اولیه است. در طراحی این گونه سامانه‌ها تنظیمات و روش‌های متنوعی مورد استفاده قرار گرفته‌اند. برخی شرکت‌ها با نصب سنسورهای مادون قرمز ترافیک اطراف ماشین را تشخیص می‌دهند و از دوربین برای تشخیص ترافیک و خط در قسمت جلویی خودرو استفاده می‌کنند. تنوع انواع سنسورها و دوربین‌ها و همچنین امکان تغییر تعداد سنسورهای مورد استفاده برای طراحی سامانه به پیچیدگی طراحی این سامانه‌ها اضافه کرده‌است. گستردگی تنظیمات سخت‌افزاری و پیشرفته بودن نمونه‌های فعلی مورد استفاده در شرکت‌های برتر ماشین‌سازی ما را بر آن داشت تا در این آزمایش با بهره‌گیری از امکانات نرم‌افزاری و شبیه‌سازی، تاثیر ویژگی‌های جانبی نظیر موقعیت سنسورها و ویژگی‌های کانونی دوربین را بررسی کنیم و نتایج را در قالب گزارش پیش‌رو ارائه دهیم.

کلیدواژه‌گان: سامانه کمک‌نگه‌دارنده خط، شبیه‌سازی MATLAB Lane Keeping Assist

۲ نمونه‌های سخت‌افزاری

برای شبیه‌سازی بهتر لازم است مشخصات و موقعیت سنسورها و دوربین‌ها را در نمونه‌های سخت‌افزاری فعلی در نظر بگیریم تا بتوانیم با توجه به آنها به حالتی بهتر دست یابیم یا نشان دهیم سخت‌افزارهای فعلی، بهترین نمونه‌های ممکن هستند. در برخی خودروهای مجهز به سیستم کمک‌نگهدارنده از ۶ سنسور مادون قرمز برای تشخیص ترافیک استفاده می‌شود که ۴ سنسور در قسمت عقب خودرو و ۲ سنسور در قسمت جلو تعبیه می‌شوند. در کنار این سنسورها از یک دوربین نیز برای تشخیص خطوط استفاده می‌شود. همان طور که از نحوه نصب سنسورها برمی‌آید، نصب سنسور در قسمت عقبی ماشین ارتباطی با بخش کمک نگه‌دارنده خط نداشته و برای سیستم‌های دیگری نظیر جلوگیری از تصادف از بخش عقبی ماشین مورد استفاده قرار می‌گیرند. با توجه به این موضوع می‌توان دریافت که ورودی‌های مورد نیاز برای شبیه‌سازی ما حداکثر دو سنسور در قسمت جلویی ماشین و یک دوربین خواهد بود. علاوه بر تعداد سنسورها، نوع آنها نیز در طراحی سیستم اهمیت دارد. در سیستم‌های فعلی از سنسورهای کوتاه/متوسط برد، سنسورهای دوربرد و سنسورهای فراصوت استفاده می‌شود. نوع و تعداد سنسورهای مورد استفاده در یک نمونه سامانه در شکل ۱ قابل مشاهده است.



شکل ۱ - انواع سیستم‌های اطلاعاتی استفاده شده در خودروهای مجهز به سیستم کمک نگهدارنده خط. همان طور که از شکل مشخص است، در برخی مناطق تنها سنسورهای کوتاه‌برد مورد استفاده هستند. این در حالی است که سنسورهای دوربرد و حتی دوربین در برخی مکان‌ها مورد استفاده قرار می‌گیرند.

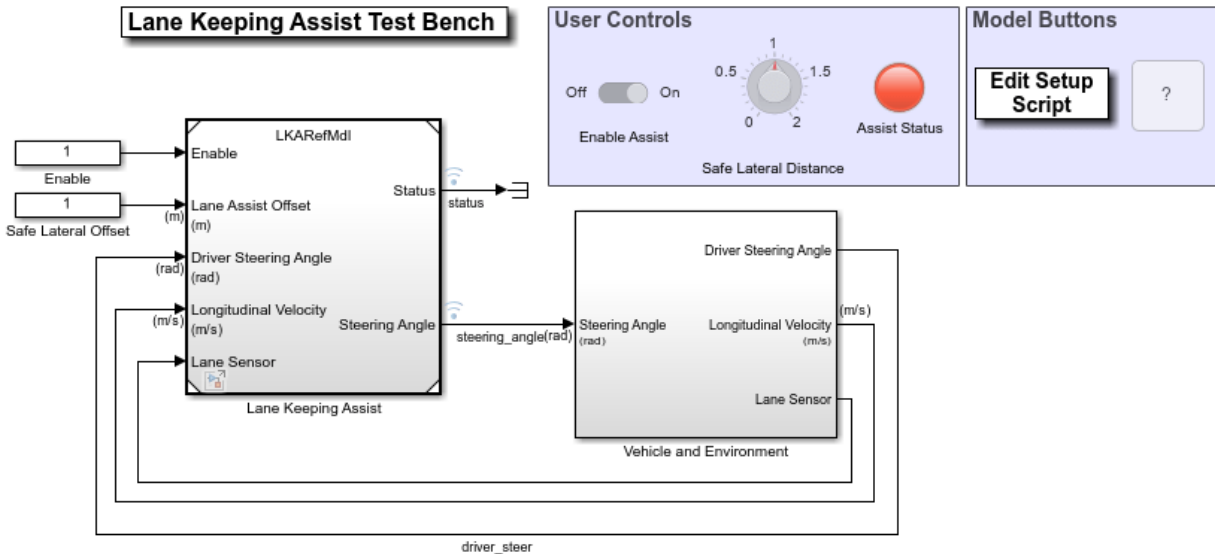
۳ پیاده‌سازی نرم‌افزاری

با توجه به توضیحات ارائه شده در بخش پیشین، پروژه به صورت نرم‌افزاری شبیه‌سازی شده و متغیرهای در نظر گرفته شده برای خودرو مورد آزمایش قرار گرفته‌اند. یکی از چالش‌هایی که در این نوع پیاده‌سازی با آن مواجه هستیم، واحد درک تصویر و تشخیص خط است که پیاده‌سازی آن نیازمند استفاده از سامانه‌های پردازش تصویر است. برای جلوگیری از پیچیدگی این بخش، از سامانه Lane Keeping Assist with Lane Detection (یا به اختصار LKA) پیاده‌سازی شده در MATLAB استفاده می‌کنیم. این سامانه با در نظر گرفتن دوربین و سنسورها، مسیر راننده منحرف شده را شبیه‌سازی کرده و عملکرد سیستم کمک نگه‌دارنده خط را در اصلاح این مسیر نمایش می‌دهد.

۱-۳ معرفی LKA در MATLAB

نمای کلی این سیستم در شکل ۲ نمایش داده شده است. این سیستم از دو بخش اصلی تشکیل شده است:

۱. سامانه کمک نگهدارنده خط که زاویه فرمان خودرو را تنظیم می‌کند. (Lane keeping Assist)
۲. زیرمجموعه خودرو و محیط که برای شبیه‌سازی موقعیت خودرو و مشخص نمودن فاصله خودرو از خطوط طراحی می‌شود. (Vehicle and Environment)



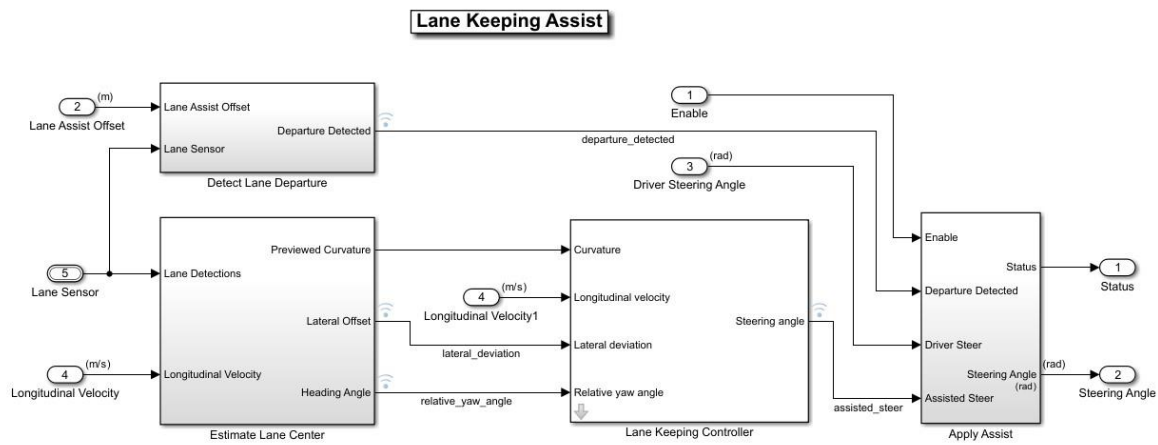
شکل ۲ - ساختار سیستم LKA در MATLAB

۱-۱-۳ Lane Keeping Assist

شمای اجزای داخلی این بخش در شکل ۳ آورده شده است.

- بلوک Detect Lane Departure به محض تشخیص خط و نزدیک شدن خودرو به خط، سیگنال Departure Detected را با مقدار True به بخش بعدی ارسال می‌کند.
- Estimate Lane Center وظیفه دارد اطلاعات دریافت شده از سنسور را به صورت مناسب به بخش بعدی انتقال دهد. این بخش با در نظر گرفتن پهنای ماشین و فاصله تشخیص داده شده توسط سنسور از سمت راست و چپ جاده، مرکز جاده را به صورت تقریبی با احتمال خطای مناسب محاسبه کرده و به بخش بعد منتقل می‌کند.
- محاسبات اصلی مربوط به تعیین زاویه نهایی فرمان در زیرسیستم Lane Keeping Controller صورت می‌گیرند. با دریافت تمامی ورودی‌های مورد نیاز از جمله سرعت و انحنای جاده، زاویه مناسب فرمان محاسبه شده و برای بخش نهایی آماده می‌شود.
- Apply Assist آخرین بخش از سیستم Lane Keeping Assist است و کار آن اعمال زاویه جدید به فرمان است. در صورتی که خط توسط خط نگهدارنده تشخیص داده شود و همچنین سیستم به طور کلی فعال باشد، زاویه اعمال شده توسط خود

راننده نیز در نظر گرفته شده و زاویه جدیدی برای فرمان به عنوان خروجی نهایی محاسبه می‌شود. در صورت فعال نبودن سیستم، دستوری مبنی بر تغییر زاویه فرمان ارسال نخواهد شد.

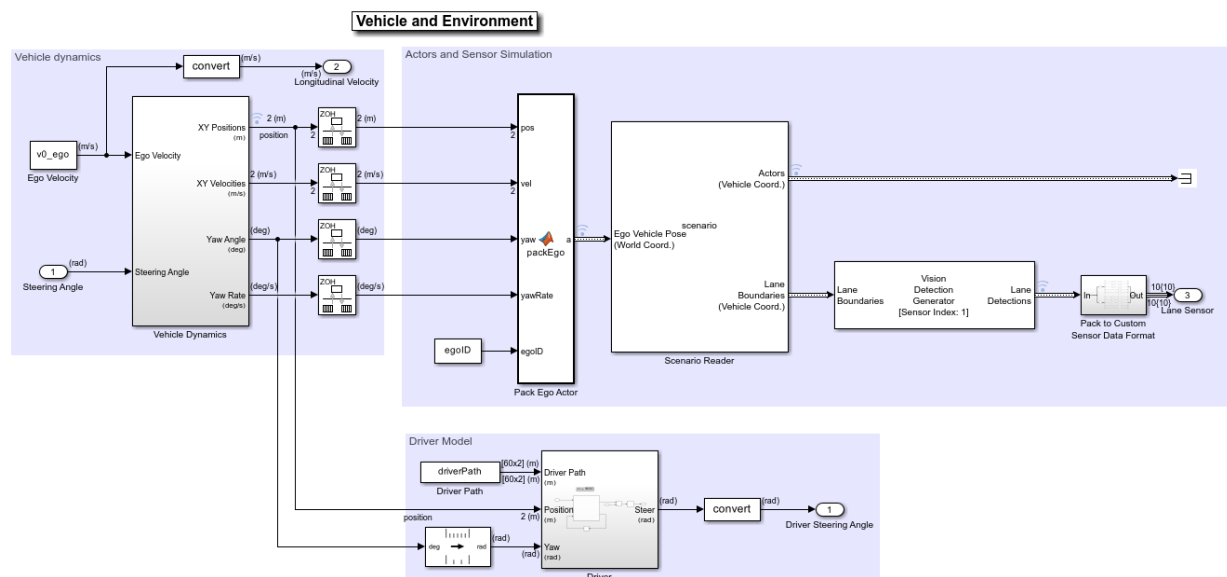


شکل ۳- اجزای داخلی Lane Keeping Assist

۲-۱-۳ Vehicle and Environment

شمای اجزای داخلی این بخش در شکل ۴ آورده شده است.

- زیرسیستم Vehicle Dynamics با استفاده از Vehicle Dynamics Blockset (مدل ساز و شبیه ساز دینامیک خودرو در محیط‌های سه بعدی مجازی) مسئولیت شبیه‌سازی خودرو را بر عهده دارد.
- بلوک Scenario Reader بر اساس موقعیت خودرو و با توجه به اطلاعات خوانده شده از فایل سناریو با نام LKATestBenchScenario.mat، محدوده دقیق خطوط چپ و راست جاده را مشخص می‌کند.
- بلوک Vision Detection Generator محدوده خطوط جاده را از بلوک Scenario Reader می‌گیرد. بخش Detection Generator، میدان دید یک دوربین تک‌چشمی را مدل‌سازی می‌کند و انحنا، مشتق انحنا، و طول موثر محدوده دیده شده از جاده را تعیین می‌کند و هر گونه مانع دیگری را نیز در نظر می‌گیرد.
- بخش Driver زاویه فرمان راننده را بر اساس مسیر طی شده توسط راننده که با دستور helperLKASetUp ساخته می‌شود، تعیین می‌کند.



شکل ۴- اجزای داخلی Vehicle and Environment

در میان تمامی این بخش‌ها، Vision Detection Generator بیشترین اهمیت را دارد، چرا که تمامی آزمایش‌های لازم برای روند پروژه از جمله تعیین موقعیت سنسورها و تغییر ویژگی‌های دوربین از این بخش امکان پذیر است.

۳-۱-۲- Vision Detection Generator

دو بخش اصلی این زیرسیستم برای تشخیص بینایی، دوربین و سنسور هستند که توضیحات هر یک در ادامه آورده شده است.

۳-۱-۲-۱-۳ مشخصات دوربین

Vision Detection Generator تشخیص‌هایش را بر اساس داده‌های دریافتی از یک دوربین حسگر تک‌چشمی نصب شده بر روی خودرو ایجاد می‌کند. یکی از مقادیر قابل تغییر در دوربین، فاصله زمانی بین خواندن داده‌ها از سنسور است (Update Interval). این مقدار به صورت پیش‌فرض برابر با ۰.۱ ثانیه است. همچنین می‌توانیم پارامترهای دیگری شامل موقعیت مکانی این دوربین، فاصله کانونی، زاویه انحراف و... را تغییر دهیم تا میزان تأثیرات آن بر تشخیص خط را بسنجیم. ویژگی اصلی دوربین استفاده شده در این مازول تک‌چشمی بودن آن است و تفاوت آن با دوربین‌هایی که دو چشم دارند (از دو زاویه تصویر را تشخیص می‌دهند) در این است که دوربین تک‌چشمی، تصاویر دو بعدی تولید می‌کند و از این نظر با بینایی انسان متفاوت است. یکی از چالش‌هایی که در مورد دوربین‌های تک‌چشمی مطرح شده، عدم توانایی تشخیص "عمق" یا فاصله اشیا است، زیرا در واقع این دوربین‌ها جهان سه بعدی اطراف را به دو بعد تصویر می‌کنند. مزیت آن‌ها نیز در ساده بودن پردازش مورد نیاز برای تصویرهایی است که تولید می‌کنند. علاوه بر این، هزینه استفاده از این دوربین‌ها کمتر است و به علت سبک بودن پردازش، می‌توانند سریع‌تر عمل کرده و تعداد تصاویر بیشتری در یک زمان مشخص تولید کنند. بنابراین این دوربین‌ها برای کاربردهایی که صرفاً تشخیص یک جسم مد نظر است و نه تشخیص فاصله از آن — مانند سیستم نگه‌دارنده خط یا تشخیص تابلوهای شامل علائم — به کار می‌روند.

علاوه بر دوربین، ماژول Vision Detection Generator از سنسور نیز برای تشخیص خطوط استفاده می‌کند. نوع سنسور مورد استفاده در این ماژول سنسور بینایی (Vision Sensor) است.

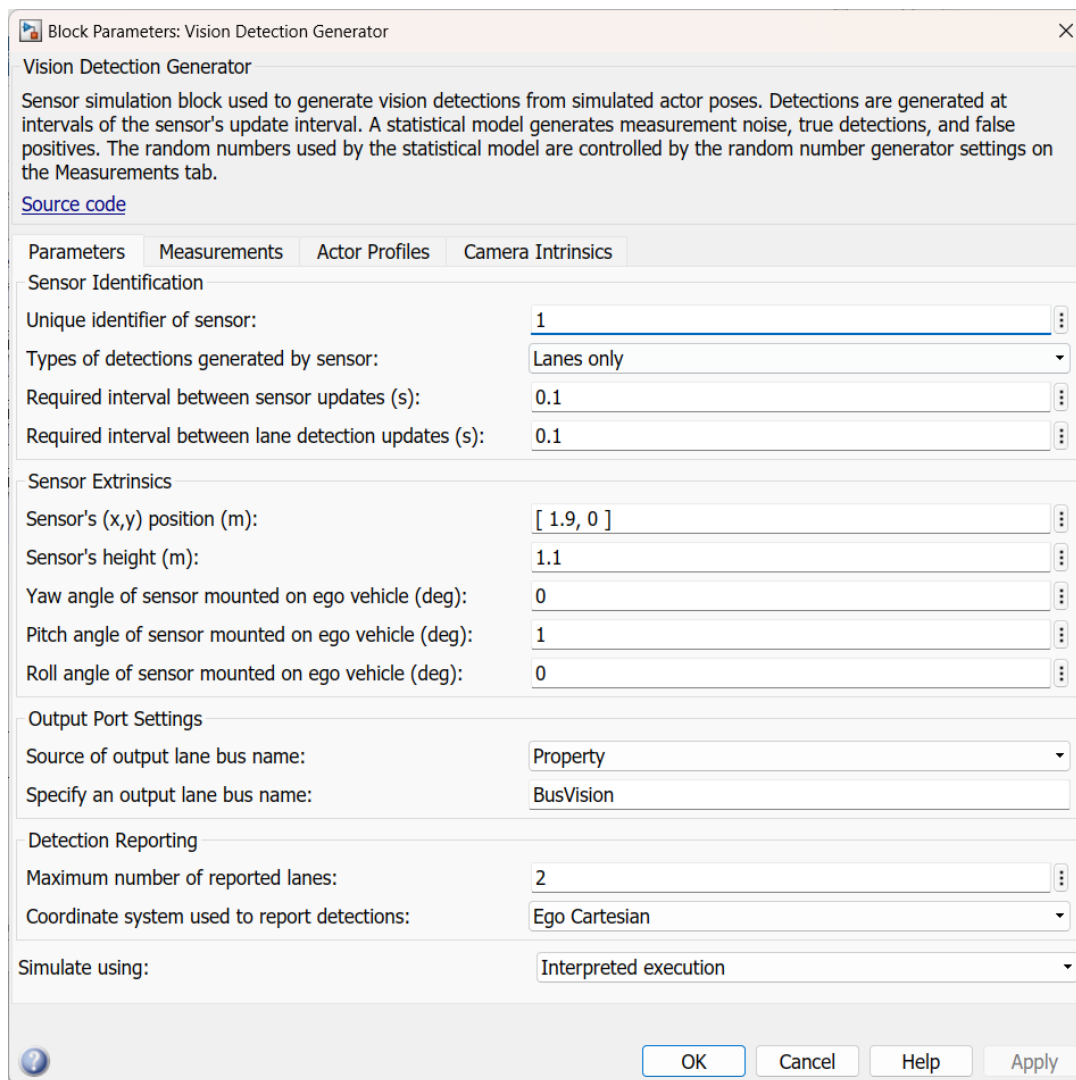
سنسورهای بینایی دارای یک دوربین نهفته هستند تا حضور اجسام را تشخیص دهند یا بین شکل و رنگ آن‌ها تمایز قائل شوند. این سنسورها به صورت فشرده شامل دوربین، چراغ و کنترل‌کننده هستند، هر چند که در بین انواع مختلف سنسورهای بینایی اجزای دیگری نیز می‌تواند به سنسور اضافه شود.

سنسورهای بینایی به دو دسته اصلی تقسیم می‌شوند: سنسورهای بینایی تک‌رنگ (Monochrome Vision Sensor) و سنسورهای بینایی رنگی (Colour Vision Sensors). تفاوت این دو دسته سنسورها در آن است که سنسورهای تک‌رنگ فیلتری با نام فیلتر بیر (Bayer filter) ندارند و بنابراین تصویر شناسایی شده توسط آنها تنها از رنگ‌های سیاه و سفید ساخته شده است. این کار باعث می‌شود که استفاده از این سنسورها برای کارهایی که نیاز به تشخیص رنگ ندارند، سریعتر باشد. از آنجایی که برای تشخیص خطوط در جاده نیز کافیت خطی سفید در زمینه سیاه/خاکستری تشخیص داده شود، استفاده از این سنسور به جای نسخه رنگی آن توصیه می‌شود.

۳-۱-۲-۱-۳-۳ مشخصات قابل تنظیم در Vision Detection Generator

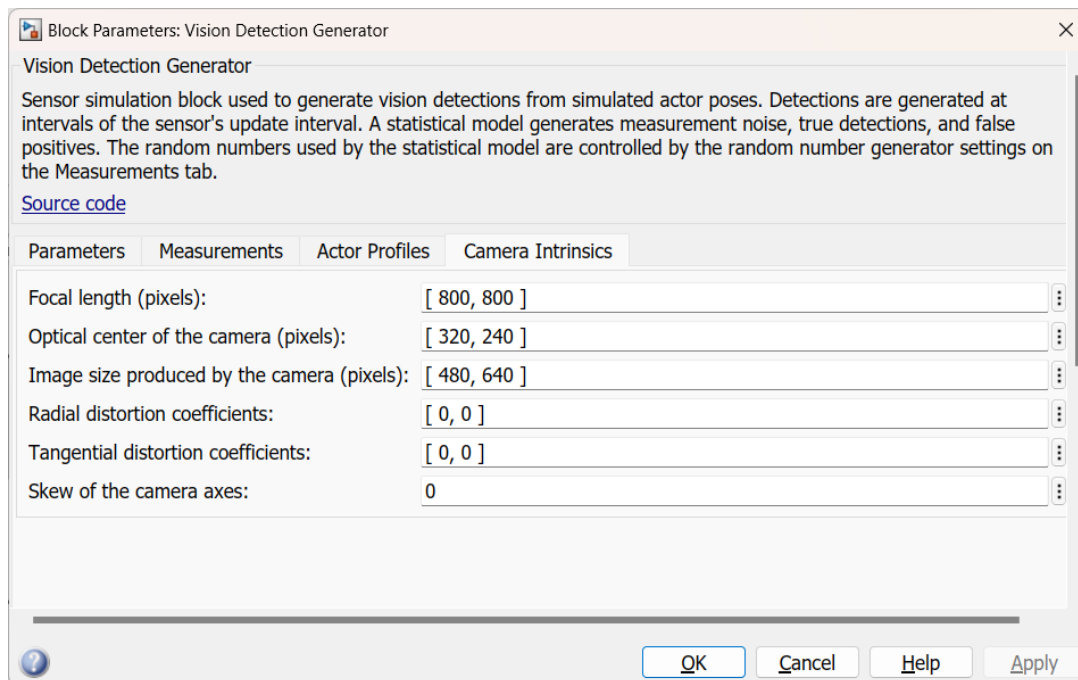
همان‌طور که در شکل‌های ۵ و ۶ قابل مشاهده است، این ماژول امکان تغییر تنظیمات دوربین و سنسورها را در اختیار ما قرار می‌دهد. در شکل ۵ تنظیمات مربوط به سنسور و در شکل ۶ تنظیمات دوربین قابل مشاهده است.

- Unique identifier of sensor: این عدد شماره سنسور مورد بررسی را نشان می‌دهد. از آنجایی که در این سیستم یک سنسور استفاده شده است، این مقدار قابل تغییر نمی‌باشد.
- Types of detections generated by sensor: این متغیر نوع سنسور را مشخص می‌کند. سنسور می‌تواند فقط خط را تشخیص دهد، خط و برخورد را تشخیص دهد و یا خط و اجسام را تشخیص دهد. از آنجایی که بررسی برخورد با اشیاء از اهداف این پروژه نمی‌باشد، تغییری در این پارامتر ایجاد نشده است.
- Required interval for updates: این متغیر که به صورت پیش‌فرض برابر با ۰.۱ است، نشان می‌دهد که فاصله بین زمانی دریافت یک داده از سنسور تا دریافت داده بعدی چقدر باشد.
- مقادیر قرار گرفته در بخش Sensor Extrinsic، مقادیری هستند که به لحاظ آزمایشی اهمیت دارند. از آنجایی که هدف نهایی این پروژه رسیدن به بهترین تنظیمات برای دوربین و سنسورهاست، متغیرهای این بخش اهمیت می‌یابند. در طی آزمایشات به این نتیجه رسیدیم که موقعیت سنسور (Sensor's position) و ارتفاع سنسور (Sensor's height) از عوامل تاثیرگذار بر سامانه کمک نگه‌دارنده خط هستند.



شکل ۵ - تنظیمات سنسور

- پارامترهای نشان داده شده در بخش Camera Intrinsics برای تعیین مشخصات دوربین اهمیت دارند. از بین تمامی این مشخصات، پس از آزمایش متوجه شدیم تنها عامل دارای تاثیر چشم گیر بر روی عملکرد سیستم فاصله کانونی دوربین (Focal length) است. تاثیر عواملی چون موقعیت مرکز نوری دوربین (Optical center of the camera) و انحراف آن (Skew of the camera axes) در برابر تاثیر فاصله کانونی قابل چشم پوشی بوده و بنابراین در این گزارش به آن پرداخته نشده است.



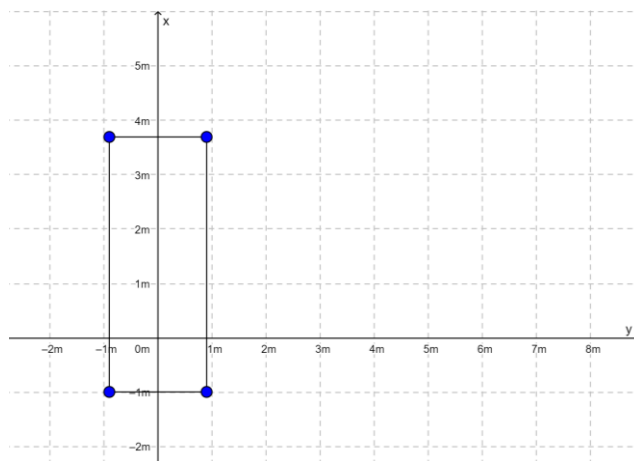
شکل ۶ - تنظیمات دوربین

۴ متغیرها و خروجی‌های LKA

با توجه به معرفی‌های انجام شده در بخش‌های قبلی، در این قسمت متغیرهای مورد آزمایش را معرفی کرده و بررسی می‌کنیم چه عواملی بر روی کیفیت سیستم تاثیرگذار هستند. با در دست داشتن این اطلاعات وارد بخش آزمایشات می‌شویم و در نهایت درباره بهترین مقادیر نتیجه‌گیری می‌کنیم.

۴-۱ متغیرهای مورد آزمایش

به منظور بررسی متغیرهای مورد آزمایش، ابتدا باید مشخصات اولیه خودرو در سامانه شبیه‌سازی شده را در نظر گرفت. خودروی مورد استفاده دارای ارتفاع ۱.۴ متر، طول ۴.۷ متر و عرض ۱.۸ متر است. موقعیت x در آزمایش‌ها معادل طول ماشین و موقعیت y نشان دهنده حرکت در عرض ماشین است. مرکز ماشین در راستای محور x بر روی مرکز مختصات قرار نگرفته است، به همین دلیل نقطه جلویی ماشین در طول $x = 3.7m$ و نقطه عقبی ماشین در طول $x = -1.0m$ قرار گرفته است. در رابطه با عرض ماشین، مقدار y می‌تواند از $y = -0.9m$ (چپ‌ترین نقطه ماشین) تا $y = 0.9m$ متغیر باشد (راست‌ترین نقطه ماشین). ارتفاع ماشین نیز از $0.0m$ که معادل سطح زمین شروع می‌شود و تا $1.4m$ که بالاترین نقطه ماشین است افزایش می‌یابد. موقعیت خودرو نسبت به مرکز مختصات در شکل ۷ نشان داده شده است.



شکل ۷ - موقعیت خودرو در صفحه مختصات

از میان تمامی متغیرهای مطرح شده برای دوربین و سنسورها، عوامل زیر مورد آزمایش قرار گرفته‌اند:

۴-۱-۱ موقعیت y سنسور

حالت پیش‌فرض: $0.0m$

بازه تغییر بین $-0.9m$ الی $0.9m$ بوده و ۱۰ آزمایش بر روی این متغیر انجام شده است، بنابراین نقاط با فاصله $0.18m$ از یکدیگر قرار گرفته‌اند.

۴-۱-۲ موقعیت x سنسور

حالت پیش‌فرض: $1.9m$

بازه تغییر بین $-1.0m$ الی $3.7m$ بوده و ۱۰ آزمایش بر روی این متغیر انجام شده است، بنابراین نقاط با فاصله $0.47m$ از یکدیگر قرار گرفته‌اند.

۴-۱-۳ ارتفاع سنسور

حالت پیش‌فرض: $1.1m$

بازه تغییر بین $0.0m$ الی $1.4m$ بوده و ۲۰ آزمایش بر روی این متغیر انجام شده است، بنابراین نقاط با فاصله $0.07m$ از یکدیگر قرار گرفته‌اند.

۴-۱-۴ فاصله کانونی دوربین

حالت پیش‌فرض: $[800, 800]$ pixels

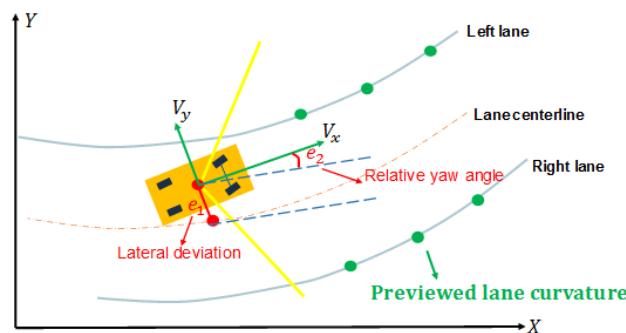
مؤلفه اول در فاصله کانونی بیانگر مختصات X کانون و مؤلفه دوم بیانگر مختصات Y کانون است و اعداد بر حسب پیکسل هستند. در آزمایش‌های انجام شده، این مقدار را از $[100, 100]$ تا $[2000, 2000]$ با فاصله‌های $[100, 100]$ تغییر دادیم و تاثیر آن بر عملکرد سامانه نگه‌دارنده خط را بررسی کردیم.

۲-۴ خروجی‌های قابل بررسی

خروجی‌های سامانه کمک نگه‌دارنده خط در MATLAB به صورت سیگنال نمایش داده می‌شوند. برخی از خروجی‌ها و ویژگی‌های آنها در ادامه معرفی می‌شوند:

- `departure_detected (true/false)`: ماژول Detect Lane Departure تولیدکننده این سیگنال است. زمانی که وسیله نقلیه در نزدیکی خط شناسایی شود، مقدار این سیگنال ۱ (true) می‌شود.
- `lateral_deviation (scalar)`: انحراف جانبی وسیله نقلیه بر حسب متر نسبت به خط مرکزی جاده محاسبه می‌شود. انحراف جانبی e_1 هنگامی که وسیله نقلیه در سمت راست خط مرکزی قرار دارد مثبت و زمانی که در سمت چپ قرار دارد منفی است.
- `relative_yaw_angle (scalar)`: زاویه محور طولی خودرو نسبت به خط مرکزی جاده بر حسب رادیان است که از فرمول $e_2 = \theta_e - \theta_c$ که θ_e زاویه خودرو و θ_c زاویه خط مرکزی جاده است و هر دو زاویه‌ها نسبت به مختصات جهانی محاسبه شده‌اند.
- `steering_angle (scalar)`: زاویه فرمان جلو بر حسب رادیان. منظور از زاویه فرمان، زاویه بین لاستیک‌های جلو و محور طولی وسیله نقلیه است. زاویه فرمان زمانی که به سمت بخش مثبت محور ماشین باشد، مثبت است.
- `assisted_steer (scalar)`: مشابه زاویه فرمان و بر حسب رادیان است. این زاویه، زاویه‌ای است که توسط سیستم کمک نگه‌دارنده محاسبه می‌شود و زاویه فرمان اصلی را تغییر می‌دهد.
- `Velocity (non-negative scalar)`: سرعت ماشین بر حسب m/s.

حال از بین خروجی‌های ممکن، لازم است بررسی کنیم تغییر کدام خروجی‌ها و به چه صورت به معنای عملکرد بهتر سیستم خواهند بود. برای مشخص شدن بهتر زوایا و ویژگی‌های مطرح شده در سیگنال‌های بالا، شکل ۸ در ادامه آورده شده است.

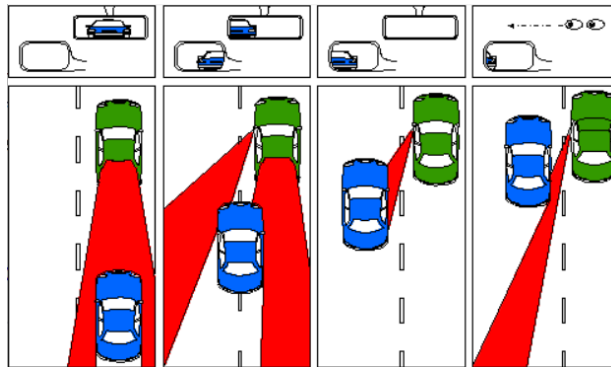


شکل ۸ - نمایش متغیرهای نام برده شده بر روی شکل

۳-۴ عوامل تعیین کننده کیفیت سیستم نگاهدارنده خط

۱-۳-۴ تشخیص سریع خط

مهم ترین عامل تعیین کننده کیفیت سیستم، توانایی آن در تشخیص سریع خط است. به محض آن که سنسورها و دوربین تشخیص دهند که اتومبیل در حال نزدیک شدن به خط است، باید سیگنالی مبنی بر فعال سازی سیستم کمک نگاهدارنده خط ارسال شود تا زاویه فرمان کمکی اعمال شده و از نزدیک شدن به خط جلوگیری شود. اگر تاخیری در ارسال این سیگنال وجود داشته باشد زاویه فرمان اعمال شده توسط سیستم هوشمند تندتر خواهد بود چرا که باید در زمان کمتری میزان انحراف راننده از مسیر را جبران کند. در برخی موارد نادر نیز ممکن است انتخاب محل نامناسب برای سنسور یا دوربین باعث شود خط در ناحیه کور ماشین قرار گرفته و هیچ شناسایی صورت نگیرد. در مدل شبیه سازی شده در MATLAB این سیگنال معادل با «departure_detected» است.



شکل ۹ - اهمیت شناسایی نقاط کور ماشین برای یافتن بهترین مکان برای سنسور و دوربین

۲-۳-۴ سرعت پایین در هنگام چرخش

زمانی که اتومبیل در حالت چرخش قرار می گیرد، وارد وضعیت نسبتاً ناپایداری می شود چرا که هر فشار خارجی اعمال شده بیش از حد می تواند در نهایت منجر به چپ شدن ماشین شده و سلامت راننده را در خطر بیندازد. به همین جهت هر چه سرعت ماشین در زمان چرخش کمتر باشد عملکرد سیستم بهتر است. همان طور که در بالا نیز به آن اشاره شد، اغلب سیگنال ها تحت تاثیر زمان تشخیص خط قرار دارند. اگر خط دیر تشخیص داده شود ممکن است برای حفظ ماشین در مسیر لازم باشد سرعت بیشتری در زمان چرخش فرمان به آن اعمال شود. سرعت ماشین در حین چرخش در شبیه سازی MATLAB همان سیگنال «Velocity» است.

۳-۳-۴ تغییر کم زاویه فرمان

مشابه سرعت کم در زمان چرخش، زاویه فرمان نیز باید به صورت تدریجی تغییر کند. تغییر ناگهانی و زیاد در زاویه فرمان نشان دهنده اشکال در سیستم نگاهدارنده خط است. دو سیگنال «steering_angle» و «assisted_steer» به منظور بررسی این ویژگی مورد استفاده قرار می گیرند.

۵ آزمایشات و نتایج

در این بخش، به ازای هر متغیر ابتدا پیش‌بینی خود در رابطه با نحوه تاثیر متغیر بر روی کیفیت سامانه را بررسی می‌کنیم. سپس بهترین نتیجه به دست آمده در مقایسه با حالت پیش‌فرض را بررسی کرده و درباره آن توضیحاتی می‌دهیم. پیش از توضیح هر یک از متغیرها به صورت جداگانه، جدول زیر دامنه تغییر هر متغیر، تعداد آزمایش‌ها، فاصله بین متغیرها و سیگنال‌های مورد بررسی به ازای هر آزمایش را نشان می‌دهد.

متغیر (واحد)	دامنه تغییر	تعداد آزمایش‌ها	فاصله بین متغیرها	سیگنال‌های تغییر یافته	مقدار پیش‌فرض
موقعیت y سنسور	$[-0.90m, 0.90m]$	۱۰	$0.18m$	departure_detected	$0.0m$
موقعیت x سنسور	$[-1.0m, 3.7m]$	۱۰	$0.47m$	departure_detected	$1.9m$
ارتفاع سنسور	$[0.0m, 1.4m]$	۲۰	$0.07m$	assisted_steer, Velocity	$1.1m$
فاصله کانونی دوربین	$[100, 100] px$ تا $[2000, 2000] px$	۲۰	$[100, 100] px$	departure_detected	$[800, 800] px$

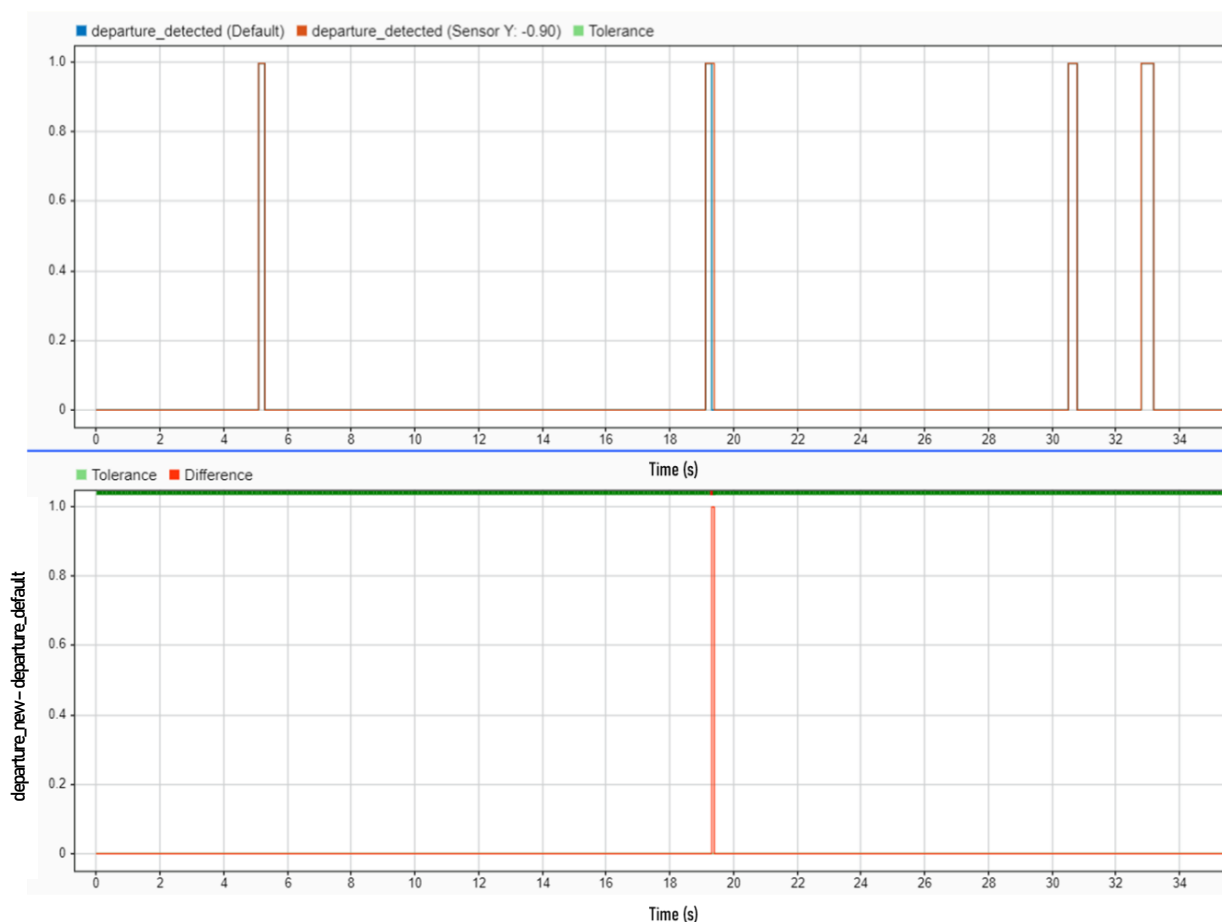
#px/تختصار یافته pixels/ست.

۵-۱ موقعیت y سنسور

با توجه به قرارگیری اولیه سنسور y در وسط عرض ماشین و احتمال به وجود آمدن نقطه کور با راست و چپ کردن ماشین، حدس اولیه بر آن بود که حفظ موقعیت پیش‌فرض برای y بهترین نتیجه را تولید کند. آزمایش‌های انجام شده بر روی متغیر y تاییدی بر روی حدس اولیه ما بودند، از این نظر که در حالت‌های راست‌ترین و چپ‌ترین سنسور تقریباً عملکرد خود را از دست داد و در حالت‌های غیر از حالت مرکزی ($y=0.0m$) نیز بهبودی نسبت به حالت پیش‌فرض نیافت.

شکل ۱۰ دو نمودار مربوط به حالت $y=-0.90m$ را نشان می‌دهد. نمودار بالایی نشان دهنده زمانی است که سیگنال departure_detected برابر با ۱ شده است. از آنجایی که نوع این متغیر آنالوگ است، واحدی ندارد. خطوط آبی تشخیص خط در حالت پیش‌فرض را نشان می‌دهند و خطوط نارنجی تشخیص خط در حالت $y=-0.90m$ هستند. نمودار پایینی نشان‌دهنده تفاوت دو سیگنال بر حسب زمان است. همان طور که مشخص است، پس از پایان یافتن زمان تشخیص سیگنال در حالت پیش‌فرض، در حالت $-0.90m$ همچنان خط تشخیص داده می‌شود. این موضوع نکته منفی است، چرا که تشخیص اشتباه نزدیک شدن به خط (در

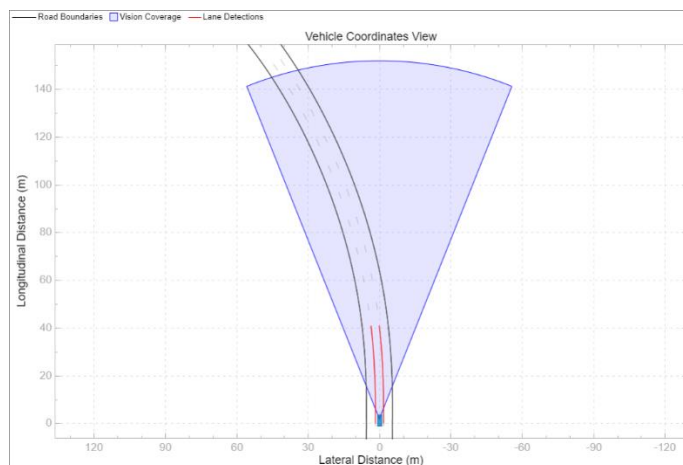
شرایطی که خطی وجود ندارد) ممکن است باعث شود خودرو در جهت مخالف از مسیر منحرف شود. بنابراین حالت ایده‌آل برای متغیر y همان حالت $y=0.0m$ است و به هیچ عنوان نباید سنسور را در راست‌ترین یا چپ‌ترین سمت ماشین قرار داد.



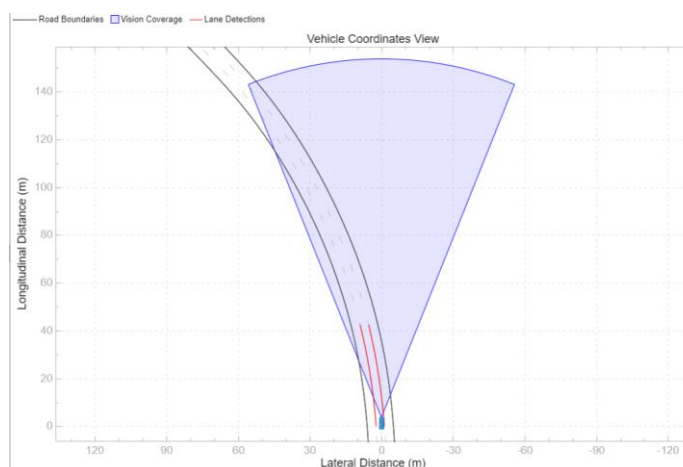
شکل ۱۰ - نمودارهای مقایسه حالت پیش‌فرض و حالت $y=-0.90m$

۲-۵ موقعیت X سنسور

در رابطه با مقدار x، انتظار می‌رفت مقادیر بیشتر x باعث ایجاد نتایج بهتری شوند (جلوتر بردن سنسور باعث تشخیص سریع‌تر و همچنین افزایش دامنه دید شود). برخلاف انتظار، مقادیر جلویی نتایج بدتری به همراه داشتند چرا که باعث به وجود آمدن نقطه کور می‌شدند. شکل زیر تغییر موقعیت سنسور از حالت پیش‌فرض آن به جلوترین بخش خودرو و اثر آن در میزان دید را نشان می‌دهد. همان‌طور که در شکل‌های ۱۱ و ۱۲ نیز مشخص است، جلو بردن سنسور باعث ایجاد ناحیه کور می‌شود و در برخی موارد خط تشخیص داده نمی‌شود.

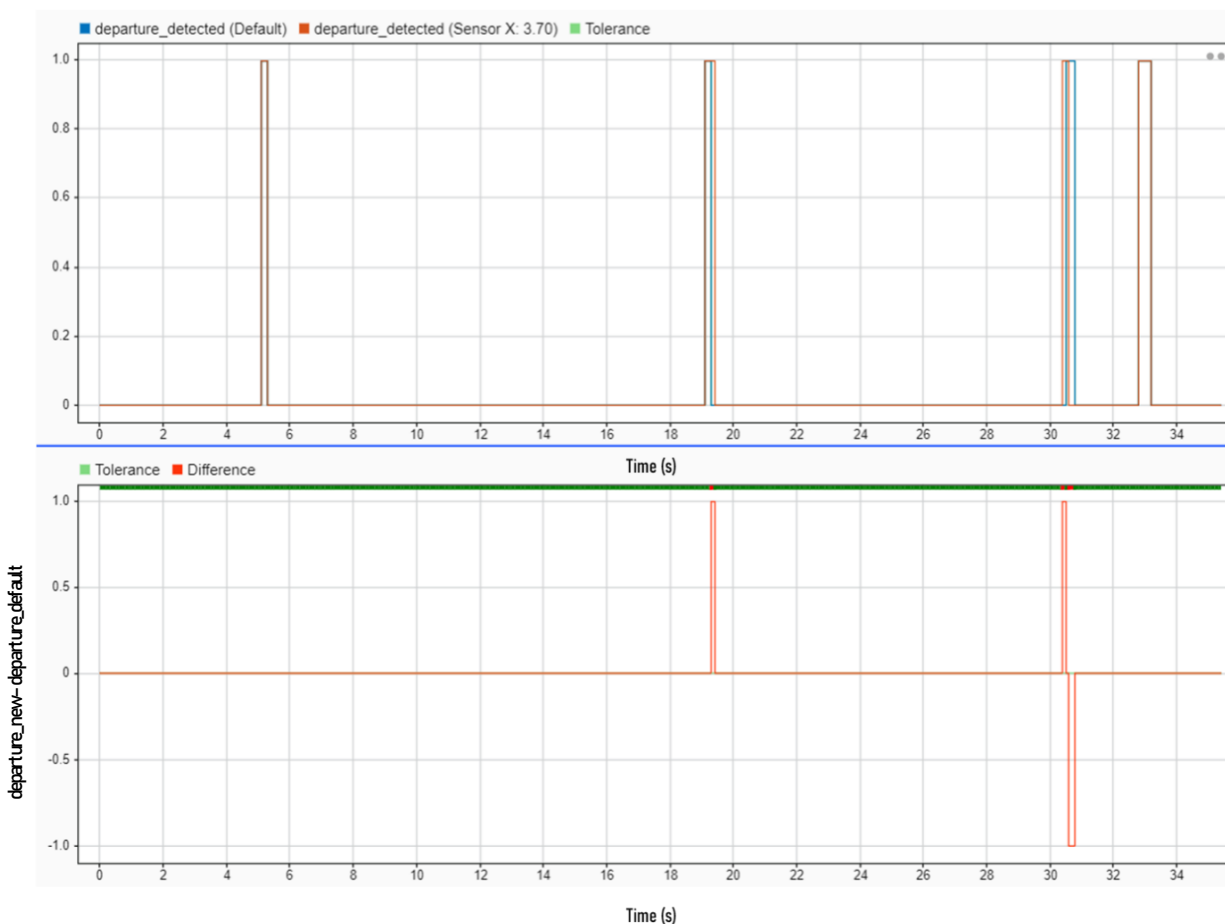


شکل ۱۱ - میزان دید در حالت پیش فرض



شکل ۱۲ - میزان دید در حالتی که سنسور در جلوترین بخش ماشین قرار گرفته است

برای تعیین محدوده‌ای از ماشین که در صورت نصب سنسور خطا ایجاد می‌شود، آزمایشات را با نمودارهایی مشابه نمودار شکل ۱۳ مقایسه کردیم. در این نمودارها نیز مشابه نمودارهای موقعیت y ، محور افقی نشان‌دهنده زمان آزمایش بر حسب ثانیه است و نمودار عمودی تنها مقادیر ۰ یا ۱ را قبول خواهد کرد که به ترتیب به معنای عدم تشخیص خط و تشخیص خط می‌باشند. همان‌طور که در نمودار قابل مشاهده است، در لحظه ۱۹ هر دو حالت خط را تشخیص داده‌اند، اما در حالت $x=3.70m$ برای مدت طولانی‌تری مقدار سیگنال برابر با ۱ باقی مانده است. از سویی دیگر در لحظات ۳۰ تا ۳۱، با توجه به نمودار اختلافات در بخش پایینی متوجه می‌شویم که در حالت جدید با وجود آنکه تشخیص خط سریعتر اتفاق افتاده است، اما برای مدت کوتاه‌تری در زاویه دید ماشین قرار داشته‌اند. با توجه به چنین نتایجی و انجام آزمایش‌های مشابه، نتیجه گرفتیم که نصب سنسور در پشت ماشین تا واسط ماشین با وجود کاستن از میزان زاویه دید (بخش کمتری از محوطه روبه‌روی ماشین تحت پوشش قرار می‌گیرد)، مشکلی در تشخیص خطوط ایجاد نمی‌کند. این در حالی است که در یک چهارم جلویی خودرو، نصب سنسور باعث ایجاد خطا می‌شود (به ویژه بازه $2.29m$ تا $3.70m$).



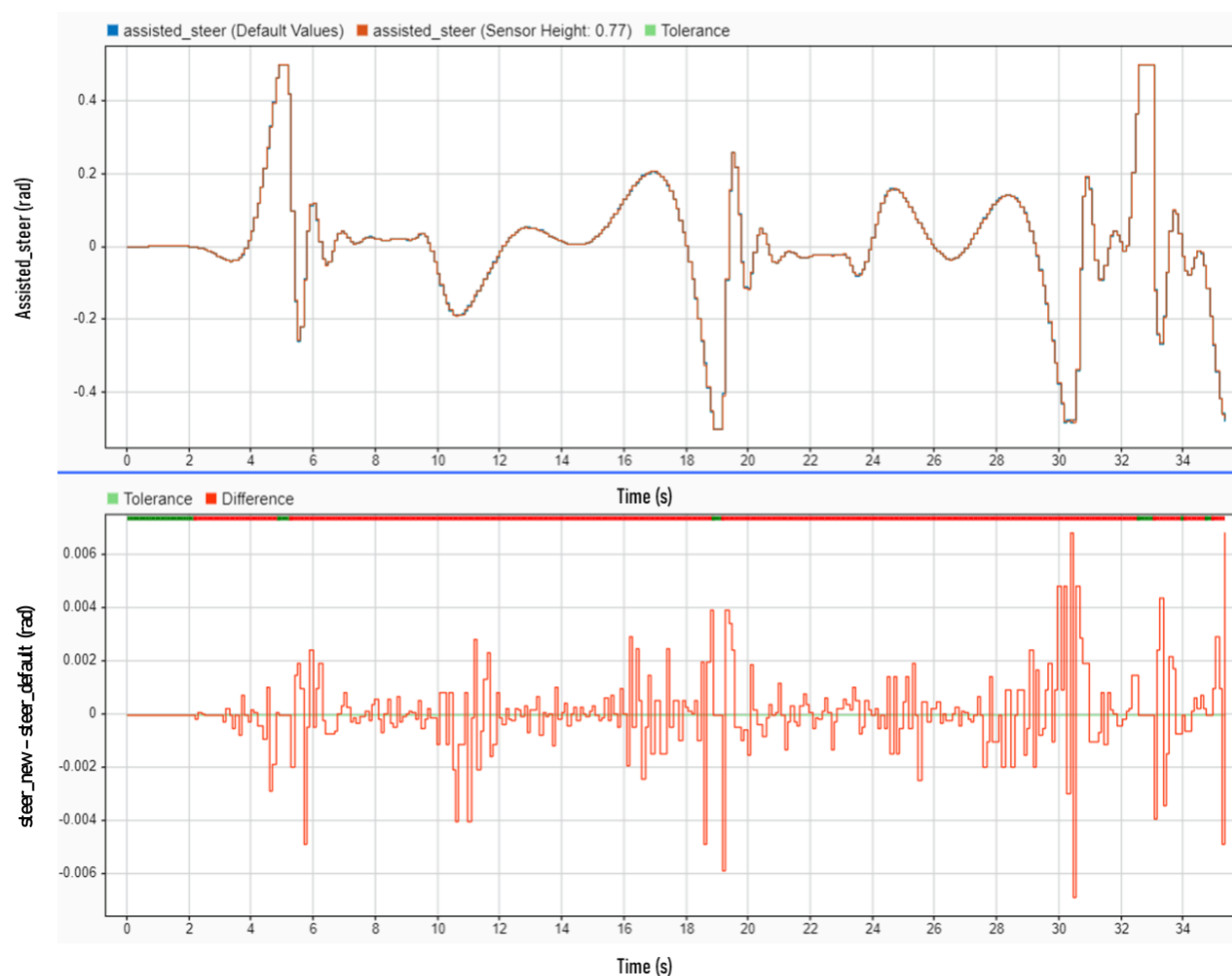
شکل ۱۳ - نمودارهای مقایسه حالت پیش فرض و حالت $x = 3.70m$

۳-۵ ارتفاع سنسور

در حالت پیش فرض، سنسور در ارتفاع $1.10m$ از سطح زمین قرار گرفته است که 0.30 متر با سقف ماشین فاصله دارد. فرض اولیه در رابطه با ارتفاع نیز این بوده است که ارتفاع وسط ماشین بهترین مکان برای نصب سنسور باشد. به همین منظور سنسور را از کف زمین تا سقف ماشین جابه جا کردیم تا نتایج مربوط به ارتفاع را نیز با یکدیگر مقایسه کنیم. در زمان تغییر ارتفاع سنسور سیگنال `departure_detected` دچار تغییر نشد و بنابراین به سراغ سیگنال `assisted_steer` رفتیم. با توجه به ادعای MATLAB مبنی بر انتخاب بهترین مقادیر برای هر متغیر، سیگنالی با کمترین تفاوت نسبت به حالت پیش فرض مورد بررسی قرار گرفت (یعنی سیگنال تولید شده در ارتفاع $0.77m$). از آنجایی که میزان تفاوت این دو آزمایش در سیگنال `assisted_steer` بسیار کم بود ($0.00687rad$) سیگنال نهایی سرعت را نیز برای مقایسه دو حالت مورد بررسی قرار دادیم. با بررسی سرعت ماشین در دو آزمایش متوجه شدیم ارتفاع $0.77m$ سرعت کمتری در زمان چرخش دارد پس ارتفاع بهتری نسبت به ارتفاع پیش فرض است و در نتیجه **ارتفاع $0.77m$** (نزدیک به وسط ماشین) بهترین ارتفاع برای نصب سنسور خواهد بود.

شکل ۱۴ نمودارهای مقایسه حالت پیش فرض با بهترین ارتفاعی که به آن رسیده ایم را نشان می دهد. از روی شکل و انطباق نسبتاً کامل دو نمودار بر روی هم در بخش بالایی شکل نیز می توان فهمید که میزان تفاوت بین دو آزمایش در حالت مقایسه `assisted_steer`

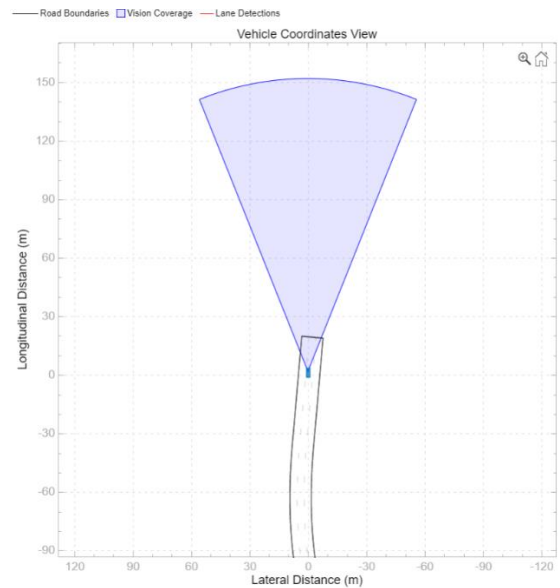
بسیار کم است. با این حال در ارتفاع ۰.۷۷ متر سرعت ماشین ۰.۷۷m/s و در ارتفاع پیش فرض این سرعت برابر با ۰.۸۰m/s بوده است.



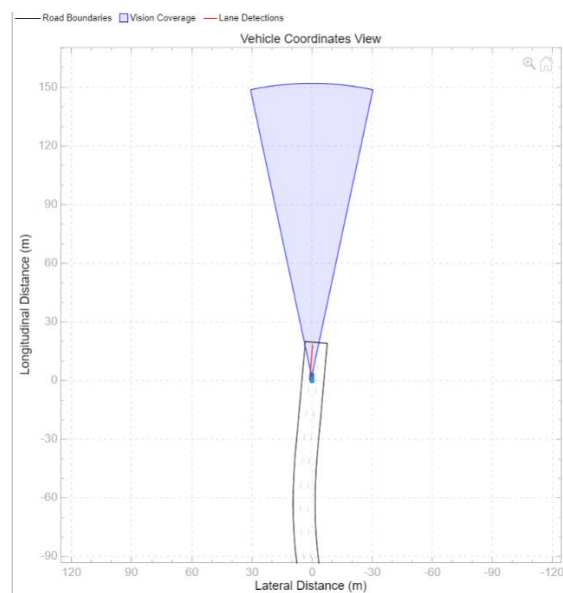
شکل ۱۴ - نمودارهای مقایسه حالت پیش فرض و حالت $height=0.77m$

۴-۵ فاصله کانونی دوربین

در آزمایش‌های مربوط به فاصله کانونی دوربین، مقدار متغیر را از $[100, 100]$ pixels تا $[2000, 2000]$ pixels با فاصله‌های $[100, 100]$ تغییر دادیم و تاثیر آن بر عملکرد سامانه نگه‌دارنده خط را بررسی کردیم. مشاهده شد که در حالتی که مقدار فاصله کانونی برابر با $[1500, 1500]$ pixels بود، به علت اینکه در این حالت نسبت به حالت پیش فرض، زاویه دید دوربین باریک‌تر و دورتر است، انحراف از خط زودتر تشخیص داده شد. چون در این زمان میزان انحراف کمتر بود، به تغییر زاویه کمتری برای باقی ماندن در خط نیاز بود و این تغییر زاویه هم زودتر و نرم‌تر انجام شد. در شکل‌های ۱۵ و ۱۶، میدان دید دوربین در حالتی که فاصله کانونی به ترتیب برابر با $[800, 800]$ pixels و $[1500, 1500]$ pixels باشد آمده است.

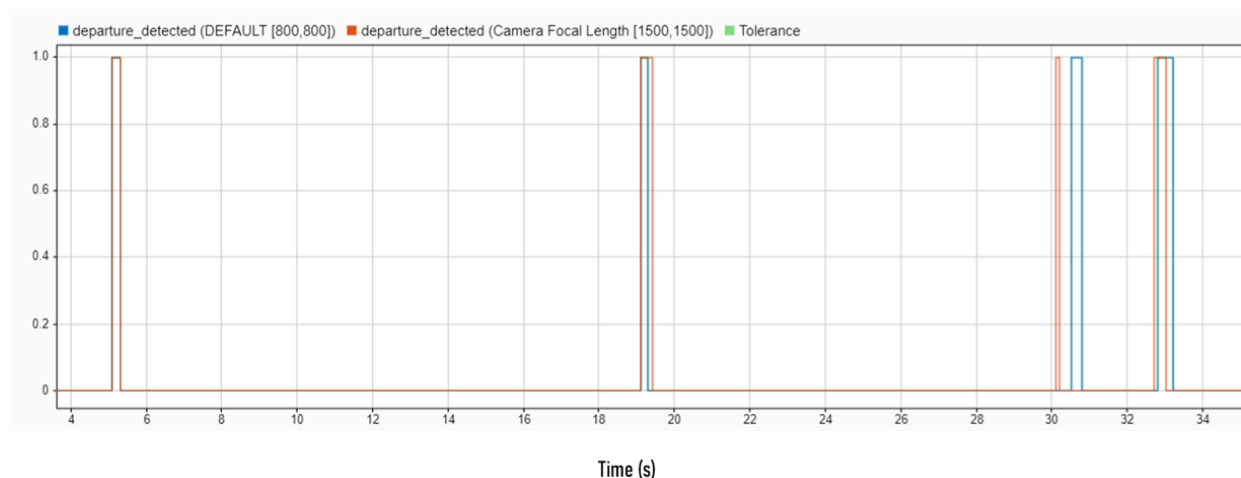


شکل ۱۵ - میزان دید در حالت پیش فرض

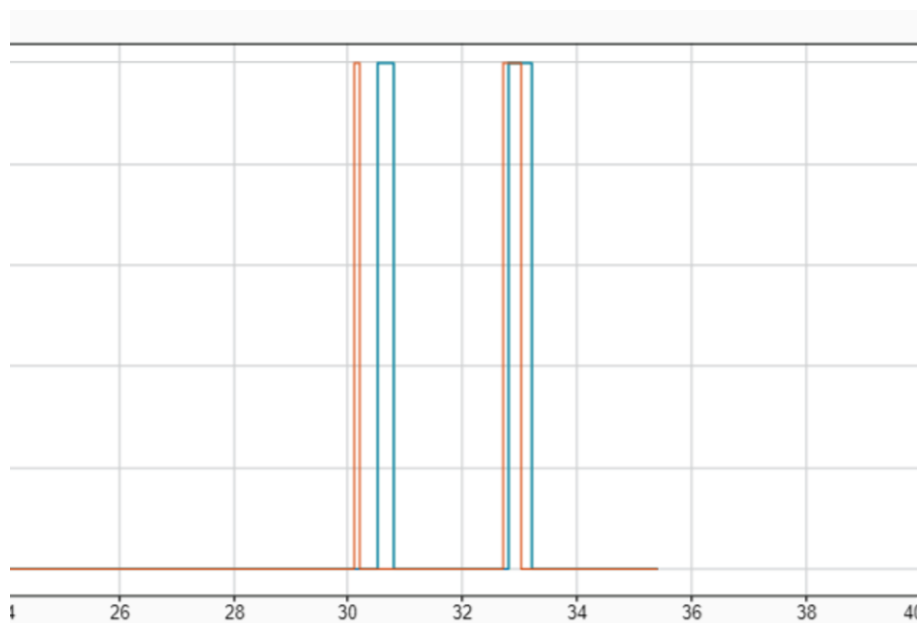


شکل ۱۶ - میزان دید در حالتی که فاصله کانونی دوربین برابر با $1500, 1500/pixels$ است

در ادامه تصویر نمودار مربوط به سیگنال departure_detected بر حسب زمان آمده است. هر زمان که این سیگنال برابر با ۱ شود نشان دهنده این است که سامانه انحراف از خط را تشخیص داده است. مشاهده می شود که در حالتی که فاصله کانونی برابر با $1500, 1500/pixels$ بوده انحراف زودتر تشخیص داده شده و زودتر برطرف شده است (شکل ۱۷ نمودار اصلی است و شکل ۱۸ برای نمایش بهتر جزئیات در بازه زمانی ۳۰ تا ۳۴ ثانیه بزرگنمایی شده است).



شکل ۱۷ - نمودار مقایسه حالت پیش فرض و حالت $focal\ length = [1500, 1500] / pixels$



شکل ۱۸ - بزرگنمایی نمودار شکل ۱۷ به منظور نمایش جزئیات

با توجه به نتایجی که از آزمایش‌ها بدست آمد، می‌شود نتیجه گرفت که شاید در سناریوهای مختلف، مقادیر بهینه پارامترها متفاوت باشند و یک مجموعه مقادیر ثابت همیشه بهترین عملکرد را نداشته باشد. برای بررسی دقیق‌تر این موضوع روش پیشنهادی این است که از تعداد بیشتر از یک دوربین که در مکانهای مختلف خودرو نصب شده‌اند استفاده شود. همچنین با توجه به تاثیر فاصله کانونی در عملکرد دوربین، می‌توانیم از دو یا چند دوربین مختلف که در یک مکان قرار دارند اما فاصله کانونی‌شان متفاوت است استفاده کنیم و هر بار تصمیم برای تغییر زاویه فرمان به صورت میانگین وزنی از داده‌های مجموعه دوربین‌ها اخذ شود. برای اینکه به هر دوربین چه وزنی داده شود می‌شود با بررسی آماری داده‌های قبلی تصمیم گرفت (اینکه تشخیص هر دوربین در مجموع در چه درصدی از موارد بهتر بوده). همچنین می‌شود معیاری برای محاسبه میزان اطمینان تشخیص هر دوربین در موقعیت‌های مختلف به دست آورد.

۶ نتایج

در این پروژه، تلاش کردیم با در نظر گرفتن عوامل متعدد تاثیرگذار مرتبط با سنسور و دوربین، تاثیر هر یک بر عملکرد سامانه کمک نگه‌دارنده خط را بسنجیم. در اکثر آزمایش‌ها، توجه به سیگنال تشخیص خط (departure_detected) برای تعیین بهترین مقدار متغیر کافی بود، و همین عامل می‌تواند نشان‌دهنده میزان تاثیرگذاری دوربین و سنسور در تشخیص خط باشد.

در این مجموعه آزمایش‌ها، با در نظر گرفتن ۴ متغیر و سیگنال departure_detected در اولویت، توانستیم بهترین مقدار را برای ۳ متغیر پیدا کنیم و این مقادیر پیدا شده با مقادیر پیش‌فرض تفاوت معناداری داشته‌اند. در یک مورد که موفق به پیدا کردن مقدار دقیق بهینه نشده‌ایم، بازه‌ای ارائه شده است که نشان‌دهنده مقادیر نامناسب برای متغیر است (موقعیت x سنسور).

در ادامه این آزمایشات و در راستای بهبود هر چه بهتر شبیه‌سازی، میتوان تاثیر عواملی چون افزایش تعداد دوربین و سنسور را بررسی کرد و حتی نوع این موارد را نیز تغییر داد. جدول زیر خلاصه‌ای از نتایج فعلی را در اختیار قرار می‌دهد:

متغیر (واحد)	سیگنال‌های تغییر یافته	مقدار پیش‌فرض	بهترین مقدار
موقعیت y سنسور	departure_detected	۰.۰m	۰.۰m
موقعیت x سنسور	departure_detected	۱.۹m	در یک چهارم جلویی خودرو، نصب سنسور باعث ایجاد خطا می‌شود
ارتفاع سنسور	assisted_steer, Velocity	۱.۱m	۰.۷۷m
فاصله کانونی دوربین	departure_detected	[۸۰۰, ۸۰۰] px	[۱۵۰۰, ۱۵۰۰] px

از توجه شما متشکریم

- [١] <https://www.aparat.com/v/KdzC6>
- [٢] <https://digiato.com/article/2016/07/18/%D8%AA%DA%A9%D9%86%D9%88%D9%84%D9%88%DA%98%DB%8C-lane-keeping-assist%D8%9B-%D8%B1%D9%88%D8%B4-%D9%85%D8%B1%D8%B3%D8%AF%D8%B3-%D8%A8%D9%86%D8%B2-%D8%A8%D8%B1%D8%A7%DB%8C-%D9%BE%DB%8C%D8%B4%DA%AF%DB%8C>
- https://www.researchgate.net/publication/316681953_Hardware_Simulation_of_Active_Lane_Keeping_Assist_Based_on_Fuzzy_Logic
- [٣] <https://uk.mathworks.com/help/mpc/ug/lane-keeping-assist-system-using-model-predictive-control.html>
- [٤] <https://uk.mathworks.com/help/mpc/ug/lane-keeping-assist-with-lane-detection.html>
- [٥] <https://www.youtube.com/watch?v=hSiTNyfp3T0>
- [٦] <https://www.mathworks.com/help/driving/ref/visiondetectiongenerator-system-object.html>
- [٧] <https://www.mathworks.com/help/driving/ug/calibrate-a-monocular-camera.html>
- [٨] <https://www.foresightauto.com/if-humans-have-two-eyes-shouldnt-cars-as-well>
- [٩] <https://www.foresightauto.com/cameras-radar-and-lidar-which-is-the-right-choice-for-autonomous-vehicles/>
- [١٠] <https://www.keyence.com/products/vision/vision-sensor/>
- [١١] <https://www.mathworks.com/help/mpc/ref/lanekeepingassistsystem.html>
- [١٢] <https://www.mathworks.com/help/vision/ref/cameraintrinsics.html>
- [١٣] <https://www.mathworks.com/help/driving/ref/monocamera.html>