



به نام خدا

نام دانشکده: دانشکده برق و کامپیوتر

ترم: 4021

نام طراح: علی بزرگ‌زاد

تاریخ تحویل: 26 آذر 1402

نام درس: یادگیری عمیق

تمرین عملی: مبحث حوزه CNN

استاد: دکتر سمانه حسینی

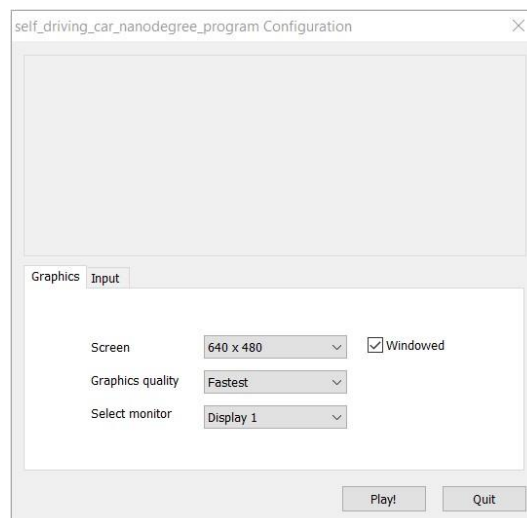
شبیه‌سازی ماشین خودران به کمک CNN

۱- معرفی تمرین

در این تمرین سعی بر آن است تا بتوان به کمک شبکه‌های عصبی Convolutional هدایت یک ماشین را انجام داد. برای این کار لازم است که ورودی و خروجی شبکه مشخص شود. برای اینکار شما باید با استفاده از تصویر دریافتی از دوربین خودرو زاویه فرمان را تعیین کنید تا ماشین همواره در جاده بماند. این تمرین در نهایت به این صورت تست خواهد شد که مدل آموزش دیده، تصویر دوربین جلوی ماشین را دریافت می‌کند و زاویه مناسب برای فرمان را پیشبینی خواهد کرد. برای انجام این کار از شبیه‌ساز توسعه داده شده توسط Udacity و مدل پیشنهادی ارائه شده توسط Nvidia استفاده خواهد شد.

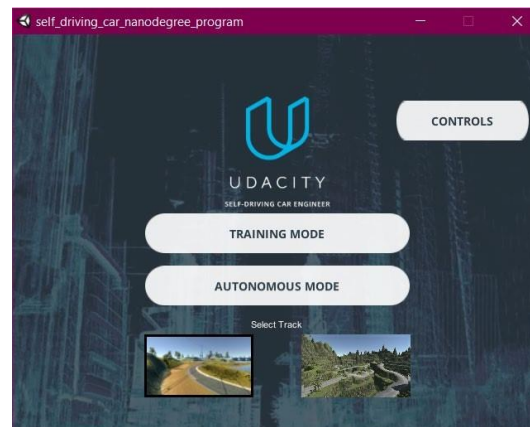
۲- جمع‌آوری داده‌ها و ایجاد Dataset

برای جمع‌آوری داده مورد نیاز برای ماشین خودران ابتدا شبیه‌ساز را دانلود نمایید ([آپلود شده بر روی IUTBox](#)). بعد از extract کردن فایل شبیه‌ساز با نام beta_simulator.exe (پسته به سیستم‌عامل شما، در نکات پایانی به آن اشاره شده است) را اجرا کنید. پنجره مشابه شکل 1 باز خواهد شد. تنظیمات دلخواه خود را انتخاب کنید و Play را انتخاب کنید.



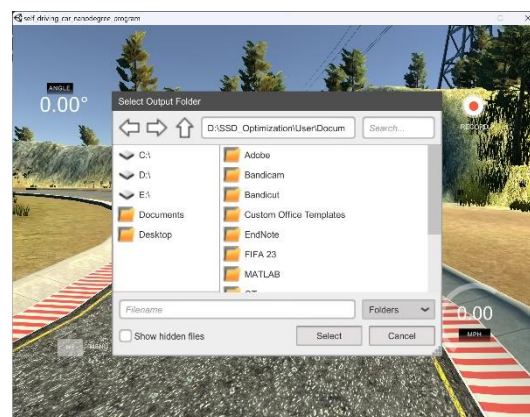
شکل 1- شبیه‌ساز ماشین خودران

پس از آن صفحه انتخاب حالت شبیه‌ساز باز خواهد شد. در این قسمت برای جمع‌آوری داده باید حالت Training mode را انتخاب کنید تا با حرکت دادن ماشین در مسیر درست داده‌های مورد نیاز تولید و ذخیره شوند. این صفحه در شکل 2 نشان داده شده است.



شکل 2- صفحه انتخاب حالت اجرای شبیه‌ساز

با استفاده از کلیدهای معرفی شده در قسمت Controls می‌توانید ماشین را هدایت کنید. بعد از آماده‌سازی شبیه‌ساز برای جمع‌آوری داده‌ها باید ماشین را در کل مسیر هدایت کنید. این کار باید در هر دو جهت رفت و برگشت انجام گیرد که بالانس داده‌ها حفظ شود (نکته کلیدی 😊). برای ذخیره داده‌ها گزینه Record در بالای صفحه که با رنگ قرمز مشخص شده است را بزنید و از شما می‌خواهد که مسیر ذخیره داده‌ها را مشخص کنید (شکل 3). پس از انتخاب مسیر دلخواه گزینه Select را بزنید و مجدد روی گزینه Record کلیک کنید، حال شروع به هدایت ماشین در مسیر کنید. از آنجایی که زاویه فرمان یک مقدار پیوسته است بهتر با کمک Mouse هدایت را انجام دهید تا داده‌ها یکنواخت و پیوسته‌تر باشند. برای اینکه تعداد مناسبی داده جمع‌آوری کنید، پیشنهاد می‌شود حدوداً 5 بار در مسیر رفت و 5 بار در مسیر برگشت حرکت کنید.



شکل 3- انتخاب مسیر ذخیره داده‌ها

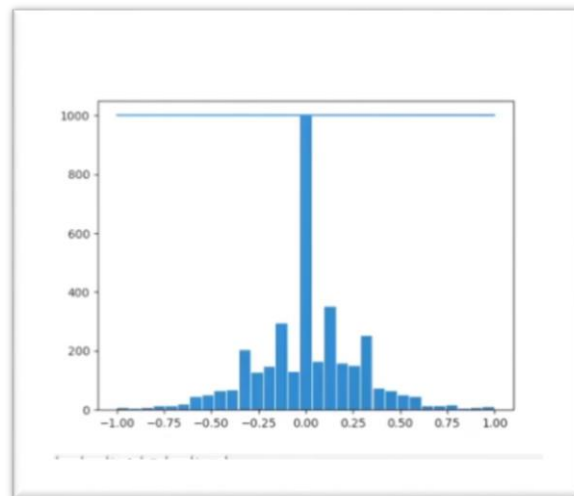
پس از اتمام حرکت ماشین در تمام مسیر باز زدن گزینه Recording شبیه‌ساز شروع به ذخیره داده می‌کند که در شکل 4 نشان داده شده است. در پایان پروسه در پوشه انتخابی، یک پوشه با نام IMG شامل تمام تصاویر ضبط شده از دوربین‌های ماشین و یک فایل به نام driving_log.csv خواهید داشت. در فایل csv تمامی اطلاعات حرکت ماشین در مسیر ذخیره شده است. ستون‌های این فایل به ترتیب Left, Right, Steering, Throttle, Brake و Speed هستند. سه ستون اول نام تصاویر دریافتی از دوربین‌های وسط، چپ و راست هستند. شما برای انجام تمرین فقط از تصاویر دوربین و مقدار Steering که همان زاویه فرمان است استفاده کنید.



شکل 4- ذخیره داده‌های جمع‌آوری شده توسط شبیه‌ساز

۳- بررسی و بالانس کردن Dataset

برای اینکه ماشین همواره بین خطوط جاده بماند باید شبکه طوری آموزش ببینید که مدام به چپ یا راست انحراف نداشته باشد. برای بررسی این که آیا داده‌های جمع‌آوری شده توزیع مناسبی برای آموزش شبکه دارند یا خیر باید Histogram داده‌ها (طبیعتاً برای زاویه فرمان این بررسی انجام خواهد شد) رسم شود. برای دیتاستی که بالانس داده‌ها حفظ شده باشد، Histogram حدوداً به صورت نمایش داده شده در شکل 5 خواهد بود.



شکل 5- Histogram مطلوب برای داده‌ها

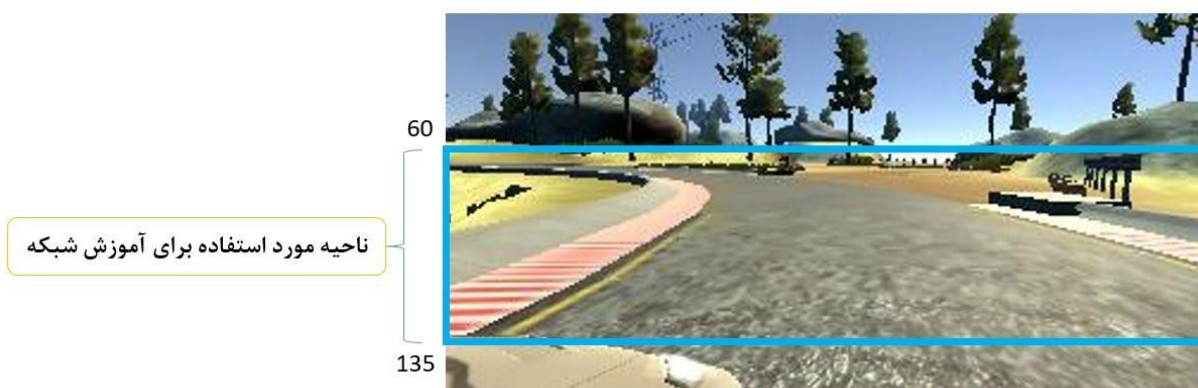
نکته قابل ملاحظه دیگر هم تعداد رسم Histogram به صورتی است که مرکز نمودار 0 باشد. به دلیل اینکه ماشین عموماً مستقیم حرکت می‌کند مطلوب است که تعداد زاویه فرمان 0 از همه بیشتر باشد اما این نکته را هم در نظر بگیرید که اگر تعداد زیادی از داده‌ها 0 باشد تاثیر سایر زوایا بسیار کم خواهد شد و شبکه قادر به یادگیری نخواهد بود. پس به صورت تصادفی و کنترل نشده تعداد داده‌های 0 را هم کاهش دهید. (شاید زیادی راهنمایی کردم 😊). اگر بعد از اعمال این عملیات باز هم Histogram شما تقریباً شبیه شکل 5 نشد به این معناست که داده‌های جمع‌آوری شده از توزیع مناسبی برخوردار نبودند و باید مجدداً فرآیند جمع‌آوری داده را انجام دهید تا خلا داده‌ها برطرف شود (البته این احتمال بسیار پایین است).

۴- Data Augmentation

برای اینکه مدل ما از Generalization مناسب برخوردار باشد، لازم است تا تصاویر دریافتی کمی متنوع باشند. برای اینکار از حالت‌های Flip، Zoom، pan، rotate و تغییر سطح روشنایی استفاده کنید (اعمال تمام موارد الزامی می‌باشد). برای اعمال Flip حتما در نظر داشته باشید که زاویه متناظر با تصویر هم باید در جهت مخالف باشد (منفی شود). توجه کنید که این تغییرات باید به صورت تصادفی و نه به صورت یکنواخت، بر روی تعدادی از داده‌ها فقط اعمال شود. لازم به ذکر است که این تغییرات فقط برای داده‌های آموزشی اعمال می‌شود. (شاید بعضی از موارد برای شما توضیح واضح‌تر باشد ولی جهت این گفته شده که ابهامی در انتها باقی نماند)

۵- پیش‌پردازش داده‌ها

پیش‌پردازش لازم برای انجام این تمرین باید شامل برش قسمت جاده از کل تصویر، استفاده از فضای رنگی YUV و تغییر سایز تصویر باشد. مدل ارائه شده توسط Nvidia برای این کار تصویر با سایز عرض 200 و ارتفاع Pixel 66 را استفاده می‌کند. ناحیه مختص به جاده باید برای آموزش شبکه استفاده شود که در شکل 6 نمایش داده شده است. برای پیش‌پردازش، Normalization و استفاده از فیلتر GaussianBlur هم پیشنهاد می‌شود. (در صورت اعمال تغییری در پیش‌پردازش می‌بایست این تغییرات را در هنگام تست مدل نیز اعمال کنید، در ادامه نحوه تست مدل گفته شده است)



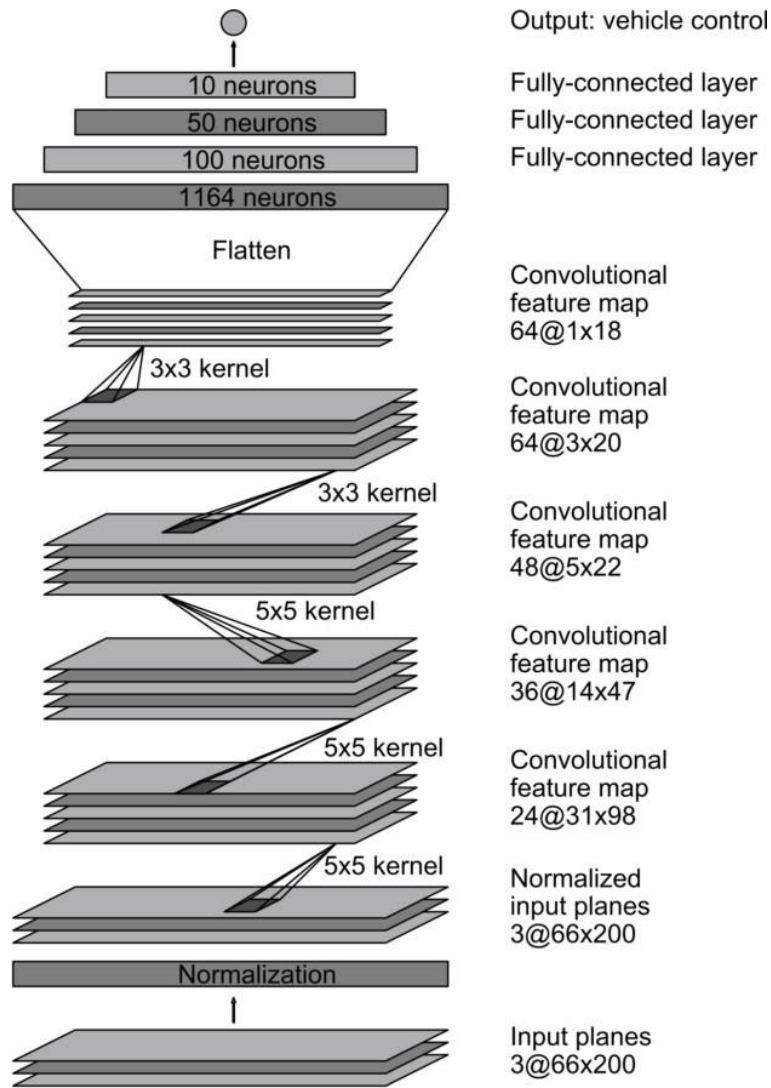
شکل 6- ناحیه پیشنهادی مورد استفاده برای آموزش شبکه

۶- Batch کردن دیتاست

پیشنهاد می‌شود تابعی تعریف کنید که در نهایت Dataset شما را به صورت Batch به سایز دلخواه شما تقسیم‌بندی کند. از این تابع در آموزش شبکه استفاده خواهید کرد.

۷- آموزش شبکه

در این مرحله پیشنهاد می‌شود با استفاده از شکل 7 شبکه خود را بسازید. نمودارهای آموزش شبکه را رسم و بررسی کنید. حتما صحت آموزش شبکه خود را از طریق این نمودارها بسنجید. در پایان مدل نهایی خود را ذخیره کنید.



شکل 7- معماری پیشنهادی شبکه برای ماشین خودران

۸- تست مدل

پیشنهاد می‌شود برای انجام این تمرین ابتدا از طریق فایل `package_list.yaml` یک محیط مجازی مختص این تمرین بسازید تا از صحت عملکرد ماژول‌ها اطمینان حاصل کنید. این محیط برای `Tensorflow_GPU` و سیستم‌عامل 64 بیتی در نظر گرفته شده و کاملاً تست شده است. نحوه ایجاد محیط در `package_list.yaml` توضیح داده شده است. دلیل اینکار این است که برخی از نسخه‌های کتابخانه‌های مورد نیاز باهم ناسازگارند و ممکن است منجر به عدم اجرای صحیح شبیه‌ساز شود. البته این صرفاً یک پیشنهاد می‌باشد و به طبع با `Package`های دیگر نیز امکان اجرا وجود خواهد داشت.

بعد از اجرای شبیه‌ساز و انتخاب نقشه مورد نظر آن را با حالت `Autonomous` اجرا کنید، سپس فایل `TestSimulation.py` که در اختیارتان قرار گرفته است را با توجه به تعیین آدرس `Model` آموزش دیده به اجرا درآورید. ماشین باید با دقت خوبی در مسیر شروع به حرکت کند.

- بسیاری از موارد مطرح شده، صرفاً یک پیشنهاد است و شما می‌توانید از روش‌های دیگر نیز استفاده نمایید ولی بخش Data Augmentation را می‌بایست تمام مواردی که در آن مطرح شده را به صورت کامل پیاده‌سازی کنید، چرا که نمره‌دهی بقیه بخش‌ها با این قسمت رابطه مستقیم خواهد داشت.
- از مرحله تست خود یک **فیلم** کوتاه تهیه نمایید و روی آن موارد مورد نیاز را توضیح دهید. شبیه‌ساز باید در خود فیلم تازه اجرا شود (بعد از شروع به ضبط فیلم تازه شبیه‌ساز را اجرا نمایید)، پیست مورد نظر را انتخاب و فایل TestSimulation.py را اجرا نمایید تا ماشین شروع به حرکت در پیست و یک دور چرخش در پیست نماید. در پایان **فیلم** کد مورد نظر خود را به صورت خلاصه بررسی کرده و در مورد آن توضیح دهید.
- مرحله جمع‌آوری داده از روی شبیه‌ساز و تست مدل ذخیره شده شما بر روی شبیه‌ساز نیاز به قدرت پردازش نداشته و به راحتی بر روی هر لپتاپی می‌توانید اجرا کنید (با توجه به تنظیمات گفته شده در مرحله دوم). اما مرحله آموزش مدل بر روی تصاویر جمع‌آوری شده که نیاز به پردازش دارد را می‌توانید از GoogleColab استفاده نمایید (به طبع باید تصاویر جمع‌آوری شده را به روش‌های که قبلاً بیان شده به GoogleColab یا GoogleDrive خود انتقال دهید).
- شبیه‌ساز Udacity برای تمام سیستم‌عامل‌ها وجود دارد (**آپلود شده بر روی IUTBox**) که می‌توانید از آنها بسته به نیاز خود استفاده نمایید ولی به طبع نسخه ویندوز آن پیشنهاد می‌شود و با توجه به مورد قبل که نیاز به پردازش در مرحله جمع‌آوری داده و تست وجود ندارد، می‌توانید از شبیه‌ساز ویندوز استفاده نمایید. (این انتخاب دیگر با خود شماست)
- در صورتی که تمام شما دوستان بتوانید، مدل را به بهترین نحو آموزش و تست نمایید برای تعیین **بهترین** مدل آموزش دیده و تست شده، غیر از چرخش یک دور ماشین به دور پیست، میزان وسط بودن ماشین، تعداد ترمزهای ماشین در طول مسیر و زمان چرخش یک دور پیست نیز در نظر گرفته خواهد شد.
- پیست سمت چپ به مراتب ساده‌تر است و از آن مسیر برای این تمرین استفاده نمایید. در صورتی که بتوانید مدل را روی پیست سمت راست آموزش بدهید، **نمره اضافه** خواهد داشت.

نکات تحویل تمرین:

- همانطور که قبلاً هم اطلاع داده شد، شما مجاز هستید در طول ترم تا 8 روز تاخیر در تحویل کل تکالیف داشته باشید.
- دانشجویان می‌توانند در حل تکالیف با دوستان خود مشورت نمایند اما در نهایت هرکس موظف است تکلیف را به صورت فردی، انجام و تحویل دهد. **لذا، در صورت مشاهده تکالیف کپی بین دانشجویان، نمره تمامی افراد شرکت کننده در آن صفر خواهد بود.**
- توضیحات شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است. لطفاً تمامی نکات و فرض‌هایی که برای پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید را در گزارش ذکر کنید.
- در صورت داشتن هرگونه سوال می‌توانید از طریق ایمیل یا اکانت تلگرام زیر با دستیار آموزشی مربوطه در ارتباط باشید.

Email: a.bozorgzad@ec.iut.ac.ir / Telegram: [@Ali_Ai_Dev](https://t.me/Ali_Ai_Dev)