

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه صنعتی نوشیروانی بابل

دانشکده مهندسی برق و کامپیوتر

پایان نامه کارشناسی رشته مهندسی کامپیوتر گرایش نرم افزار

تشخیص خواب آلودگی با شبکه های عصبی

نگارش

جمال الدین دمیرچی

استاد راهنما

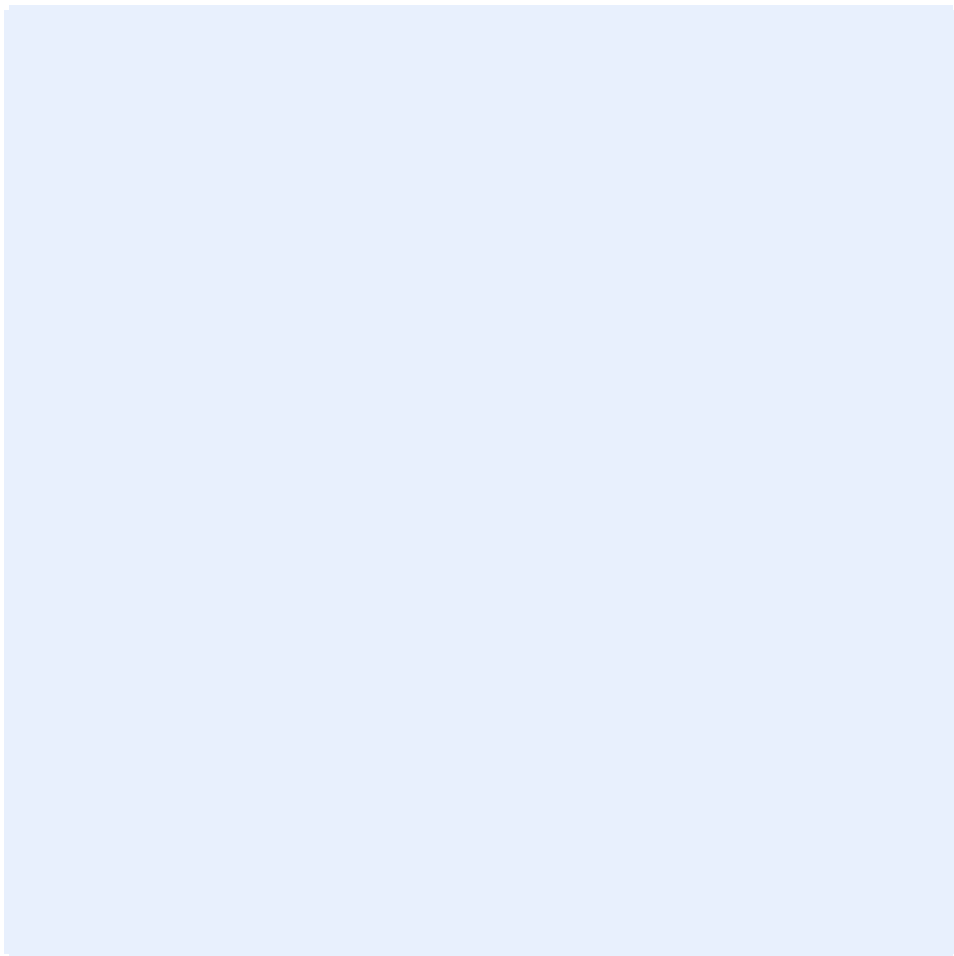
دکتر فریده ابراهیمی خاله سری

استاد مشاور

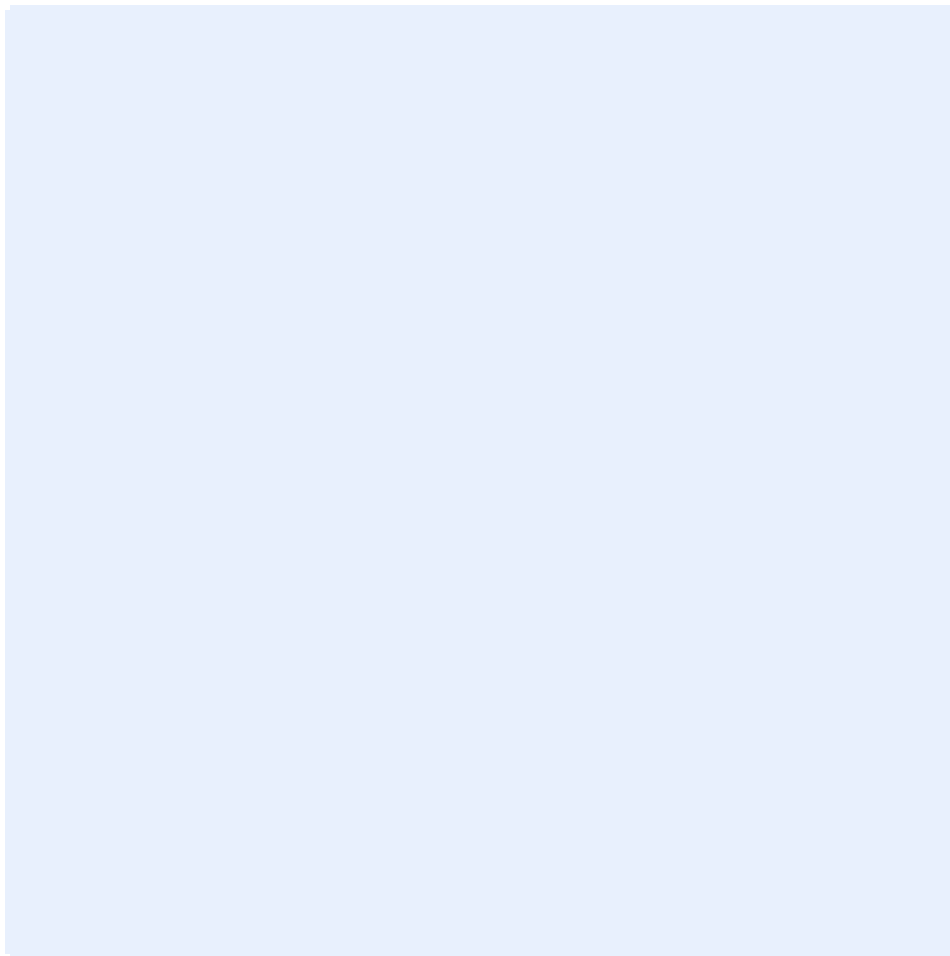
نام و نام خانوادگی استاد / استادان مشاور را اینجا وارد کنید

(خرداد 1402)

برگ اصالت و مالکیت اثر



برگ تأیید هیئت داوران / صورت جلسه دفاع (به زبان فارسی)



صفحه تقدیم

تقدیم به مادر مهربان و پدر گرامی و برادر عزیزم که با محبت‌ها و زحمات بی‌دریغشان، زمینه‌ی رشد و پیشرفت مرا فراهم آورده‌اند. سپاسگزارم از وجود ارزشمندتان در زندگی‌ام.

صفحه سپاسگزاری

بر خود لازم می‌دانم از زحمات و تلاش‌های بی‌دریغ استاد ارجمند خانم دکتر ابراهیمی که در طول این مدت، حامی و راهنمای من در مسیر کسب علم و دانش بوده‌اند، صمیمانه قدردانی نمایم.

چکیده

هدف: هدف این پژوهش، ارائه یک روش تشخیص زودهنگام خواب‌آلودگی رانندگان با استفاده از شبکه‌های عصبی کانولوشن (CNN) برای کاهش تصادفات جاده‌ای ناشی از خستگی و کمبود خواب است.

روش‌شناسی پژوهش: در این مطالعه از مجموعه داده (CEW) Closed Eye in the Wild برای آموزش مدل شبکه عصبی کاوشگر جهت تجزیه و تحلیل ناحیه صورت و چشم رانندگان استفاده شده است.

یافته‌ها: نتایج نشان می‌دهد که مدل پیشنهادی با استفاده از CNN می‌تواند با دقت 99.15 درصد خواب‌آلودگی رانندگان را تشخیص دهد.

نتیجه‌گیری: خواب‌آلودگی رانندگان یکی از عوامل عمده تصادفات جاده‌ای در سراسر جهان است که اغلب ناشی از خستگی و کمبود خواب می‌باشد.

با توجه به نتایج به دست آمده، استفاده از روش پیشنهادی می‌تواند به عنوان یک سیستم هشدار زودهنگام برای جلوگیری از تصادفات جاده‌ای ناشی از خواب‌آلودگی رانندگان مورد استفاده قرار گیرد. این روش می‌تواند در خودروها و ناوگان حمل و نقل پیاده‌سازی شود تا از بروز حوادث جاده‌ای ناشی از خواب‌آلودگی جلوگیری کند.

کلیدواژه‌ها: خواب‌آلودگی رانندگان، تشخیص زودهنگام، شبکه‌های عصبی کانولوشن، پردازش تصویر، ایمنی جاده‌ای، تشخیص خواب‌آلودگی

فهرست نوشتار

| | |
|----|---|
| ۱۷ | ۱. آشنایی با مفاهیم اولیه |
| ۱۷ | ۱-۱. مقدمه |
| ۱۸ | ۱-۲. مفاهیم پایه |
| ۱۸ | ۱-۲-۱. یادگیری ماشین |
| ۲۰ | ۱-۲-۲. شبکه‌های عصبی |
| ۲۳ | ۱-۲-۳. ارزیابی مدل |
| ۲۷ | ۲. مروری بر منابع |
| ۲۷ | ۲-۱. مقدمه |
| ۲۸ | ۲-۲. انتخاب دیتاست |
| ۳۲ | ۲-۳. پیش پردازش |
| ۳۳ | ۲-۳-۱. افزایش داده ها |
| ۳۴ | ۲-۴. پیاده‌سازی مدل |
| ۳۷ | ۲-۵. عملکرد و نتایج |
| ۴۱ | ۳. راه حل پیشنهادی |
| ۴۱ | ۳-۱. دیاگرام |
| ۴۱ | ۳-۲. توضیح مراحل دیاگرام |
| ۴۱ | ۳-۲-۱. پیش پردازش داده‌ها |
| ۴۱ | ۳-۲-۱-۱. وارد کردن کتابخانه‌های مورد نیاز |
| ۴۲ | ۳-۲-۱-۲. بارگذاری و بررسی داده‌ها |
| ۴۲ | ۳-۲-۱-۳. پیش پردازش داده‌ها |
| ۴۲ | ۳-۲-۲. پردازش داده‌ها |
| ۴۲ | ۳-۲-۲-۱. تعریف معماری مدل |
| ۴۲ | ۳-۲-۲-۲. تقسیم داده‌ها |
| ۴۳ | ۳-۲-۲-۳. آموزش و ارزیابی مدل |
| ۴۳ | ۳-۲-۲-۴. ذخیره و بارگذاری مدل |

| | |
|----|-----------------------------------|
| ۴۳ | ۳-۲-۲-۵. پیش‌بینی |
| ۴۳ | ۳-۲-۳. ارزشیابی |
| ۴۳ | ۳-۲-۳-۱. مصورسازی نتایج |
| ۴۴ | ۳-۲-۳-۲. ارزیابی عملکرد |
| ۴۴ | ۳-۳. موبایل‌نت |
| ۴۴ | ۳-۳-۱. مقدمه |
| ۴۴ | ۳-۳-۲. معرفی |
| ۴۵ | ۳-۳-۳. معماری |
| ۴۶ | ۳-۳-۴. دلایل انتخاب مدل |
| ۴۷ | ۳-۴. مدل طراحی شده |
| ۵۱ | ۴. پیاده‌سازی و تحلیل نتایج |
| ۵۱ | ۴-۱. مقدمه |
| ۵۱ | ۴-۲. پیش‌پردازش |
| ۵۲ | ۴-۳. پردازش |
| ۵۲ | ۴-۳. تحلیل نتایج |
| ۵۷ | ۵. خلاصه و نتیجه‌گیری |
| ۵۷ | ۵-۱. خلاصه و نتیجه‌گیری |
| ۵۹ | ۶. فهرست منابع |

فهرست کوتاه‌نوشت‌ها

فهرست جدول‌ها

| | |
|--|----|
| جدول 1-2. مشخصات مدل مقاله..... | ۳۵ |
| جدول 1-2. نتایج و ارزیابی مدل مقاله..... | ۳۸ |
| جدول 1-1. نمونه جدول در فصل یک..... | 18 |

فهرست تصویرها

| | |
|---|----|
| تصویر 1-1. ساختار یک شبکه عصبی..... | ۲۰ |
| تصویر 2-1. انواع شبکه‌های عصبی..... | ۲۲ |
| تصویر 3-1. ساختار CNN..... | ۲۳ |
| تصویر 4-1. معیارهای ارزیابی مدل..... | ۲۵ |
| تصویر 1-2. نمونه تصویر از چشم بسته در دیتاست ۱..... | ۲۹ |
| تصویر 2-2. نمونه تصویر از چشم باز در دیتاست ۱..... | ۲۹ |
| تصویر 3-2. نمونه تصویر از دیتاست ۲..... | ۳۰ |
| تصویر 4-2. نمونه تصویر از چشم بسته در دیتاست CEW..... | ۳۱ |
| تصویر 5-2. نمونه تصویر از چشم باز در دیتاست CEW..... | ۳۲ |
| تصویر 6-2. مدل موجود در مقاله به صورت ویژال..... | ۳۶ |
| تصویر 1-3. دیاگرام راه حل پیشنهادی..... | ۴۱ |
| تصویر 2-3. نمایی از Mobilenet..... | ۴۵ |
| تصویر 3-3. کانولوشن قابل تفکیک عمقی vs استاندارد..... | ۴۶ |
| تصویر 4-3. نمایی از مدل طراحی شده به صورت ویژال..... | ۴۸ |
| تصویر 5-3. ارزیابی مدل طراحی شده..... | ۴۹ |
| تصویر 6-3. پیش‌بینی با استفاده از مدل طراحی شده..... | ۴۹ |
| تصویر 1-4. گزارش از عملکرد مجموعه آموزش مدل نهایی..... | ۵۳ |
| تصویر 2-4. گزارش از عملکرد مجموعه Validation مدل نهایی..... | ۵۳ |
| تصویر 3-4. دقت مدل نهایی برای مجموعه آزمایش..... | ۵۳ |
| تصویر 4-4. پیش‌بینی تصاویر از مجموعه تست..... | ۵۵ |
| تصویر 5-4. پیش‌بینی تصاویر خارج از دیتاست..... | ۵۶ |

فهرست نمودارها

| | |
|---------|--|
| ۳۹..... | نمودار 1-2. Training and Testing Accuracy |
| ۳۹..... | نمودار 2-2. Training and Testing Loss |
| ۵۴..... | نمودار 1-4. Training Loss |
| ۵۴..... | نمودار 2-4. Training and Validation Loss |
| ۵۴..... | نمودار 3-4. Training and Validation Accuracy |

فهرست پیوست‌ها

پیوست الف: اصالت و مالکیت یک پایان‌نامه 28

۱. آشنایی با مفاهیم اولیه

۱-۱. مقدمه

اکثر مرگ و میرها و جراحات انسانی ناشی از تصادفات ترافیکی هستند. براساس گزارش سازمان بهداشت جهانی، هر سال یک میلیون نفر در سراسر جهان در اثر جراحات ناشی از تصادفات ترافیکی جان خود را از دست می‌دهند. رانندگانی که خواب کافی نداشته، استراحت نکرده‌اند یا احساس خستگی می‌کنند، ممکن است پشت فرمان به خواب بروند و جان خود و دیگر عابران جاده را به خطر بیندازند.

تحقیقات در مورد تصادفات جاده‌ای نشان داد که تصادفات جاده‌ای عمده به دلیل خواب‌آلودگی در حین رانندگی رخ می‌دهد. امروزه مشاهده شده که رانندگی با خستگی اصلی‌ترین دلیل بروز خواب‌آلودگی است. اکنون خواب‌آلودگی به اصل اصلی برای افزایش تعداد تصادفات جاده‌ای تبدیل شده است.

این مسئله جهانی که بسیار مهم است در اسرع وقت حل شود، به یک مشکل عمده تبدیل شده است. هدف غالب همه دستگاه‌ها بهبود عملکرد در تشخیص خواب‌آلودگی در زمان واقعی است. دستگاه‌های زیادی برای تشخیص خواب‌آلودگی توسعه یافته‌اند که بر الگوریتم‌های مختلف هوش مصنوعی متکی هستند. بنابراین، تحقیق ما نیز مربوط به تشخیص خواب‌آلودگی راننده است که می‌تواند خواب‌آلودگی راننده را با شناسایی چهره و سپس ردیابی چشم شناسایی کند.

۱-۲. مفاهیم پایه

۱-۲-۱. یادگیری ماشین

اگر بخواهیم خیلی ساده یادگیری ماشین را تعریف کنیم، یادگیری ماشین وظیفه‌ی هوشمندتر کردن کامپیوترها بدون آموزش صریح به آن‌ها در مورد چگونگی رفتارشان است. ماشین این کار را با شناسایی الگوها در داده‌ها انجام می‌دهد، به‌ویژه برای داده‌های متنوع و با ابعاد بالا، مانند تصاویر و سوابق سلامتی بیمار.

فرایند ماشین لرنینگ (Machine Learning) با وارد کردن داده‌های آموزشی به الگوریتم انتخاب‌شده آغاز می‌شود. الگوریتم با استفاده از این داده‌ها یادگیری و استخراج الگوهایی را شروع می‌کند تا بتواند وظیفه‌اش را که برای مثال، می‌تواند طبقه‌بندی یا شناسایی باشد انجام دهد. بعد از اینکه الگوریتم مرحله‌ی یادگیری را گذارند، لازم است ببینیم برای داده‌های جدیدی هم که واردش می‌شوند عملکرد خوبی را دارد یا نه.

برای آزمایش اینکه آیا این الگوریتم به‌درستی کار می‌کند یا نه، داده‌های ورودی جدیدی به الگوریتم ماشین لرنینگ وارد می‌شوند؛ سپس پیش‌بینی‌های الگوریتم بررسی می‌شود.

اگر پیش‌بینی الگوریتم مطابق انتظار نباشد، الگوریتم چندین بار دوباره آموزش می‌بیند تا زمانی که خروجی مدنظر به دست آید. این موضوع الگوریتم یادگیری ماشین را قادر می‌کند تا به‌طور مداوم به‌تنهایی از داده‌ها یاد بگیرد و بهینه‌ترین جواب ممکن را ارائه کند و با گذشت زمان به‌تدریج دقت آن افزایش یابد.

ماشین لرنینگ به دو حوزه‌ی اصلی یادگیری باناظر (Supervised Learning) و یادگیری بدون ناظر (Unsupervised Learning) تقسیم می‌شود. البته دو نوع دیگر نیز وجود دارد: یادگیری نیمه‌نظارتی (Semi-supervised Learning) و یادگیری تقویتی (Reinforcement Learning) که امروزه محل توجه است.

هر یک از آن‌ها در ماشین لرنینگ هدف و وظایف خاصی دارند، نتایج خاصی را ارائه می‌کنند و از فرم‌های مختلف داده استفاده می‌کنند. تقریباً ۷۰ درصد یادگیری ماشین یادگیری باناظر (Supervised Learning) است، درحالی‌که یادگیری بدون ناظر

(Unsupervised Learning) میان 10 تا 20 درصد است. در ادامه هر یک از آن‌ها را توضیح خواهیم داد.

یادگیری با ناظر:

در این نوع یادگیری، به الگوریتم داده‌های آموزشی همراه با برچسب‌ها یا نتایج مورد انتظار داده می‌شود. هدف آن پیدا کردن رابطه بین ورودی‌ها و خروجی‌های مطلوب است. انواع آن عبارتند از:

- رگرسیون (Regression)

- طبقه‌بندی (Classification)

- پیش‌بینی (Forecasting)

یادگیری بدون ناظر:

در این نوع یادگیری، به الگوریتم تنها داده‌های آموزشی بدون برچسب داده می‌شود و الگوریتم باید الگوها و ساختارهای پنهان در داده‌ها را کشف کند. انواع آن عبارتند از:

- خوشه‌بندی (Clustering)

- کاهش ابعاد (Dimensionality Reduction)

- انجمن قوانین (Association Rule Learning)

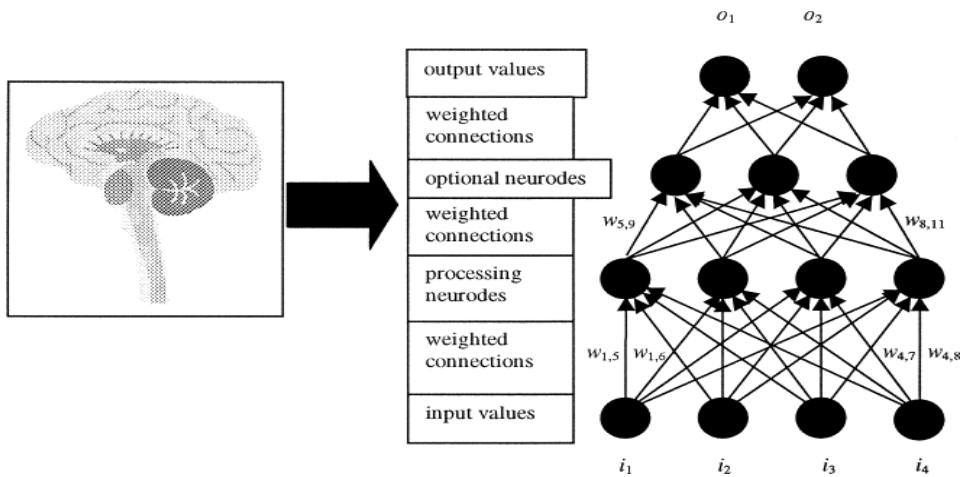
تفاوت اصلی این دو نوع یادگیری این است که در یادگیری با ناظر، نتایج مورد انتظار از قبل موجود است و الگوریتم باید رابطه بین ورودی و خروجی را یاد بگیرد. در حالی که در یادگیری بدون ناظر، هیچ نتیجه از پیش تعیین شده‌ای وجود ندارد و الگوریتم باید الگوها و ساختارهای داده‌ها را کشف کند.

در این پروژه برای تشخیص خواب آلودگی عملاً ما یک مسئله طبقه‌بندی شامل دو کلاس را حل می‌کنیم که در ادامه توضیح خواهیم داد.

۲-۲-۱. شبکه‌های عصبی

شبکه‌های عصبی مصنوعی یا ANNs، شاخه‌ای از مدل‌های مربوط به یادگیری ماشین هستند که با استفاده از اصول سازماندهی نورون‌های موجودات زنده ایجاد شده‌اند. یک ANN با مجموعه‌ای از گره‌های متصل به یکدیگر به نام نورون‌های مصنوعی ایجاد می‌شود. اگر بخواهیم مدل مغز را در نظر بگیریم، رشته‌های عصبی و گره‌های پردازشی که روند بررسی اطلاعات را دنبال می‌کنند، شبکه‌های عصبی مصنوعی نامیده می‌شوند.

دریافت، پردازش و انتقال سیگنال از جمله مهم‌ترین کاربردهای شبکه‌های عصبی مصنوعی به حساب می‌آید. هر رشته در این روش پردازش اطلاعات همچون سیناپس‌های مغزی عمل می‌کند. این رشته‌ها، سیگنال‌های مربوط به خود را دریافت می‌کنند. در مرحله بعد، آن‌ها با استفاده از گره یا نود، داده‌های ورودی را پردازش کرده و در نهایت آن را در اختیار سایر رشته‌ها قرار می‌دهند. معمولاً یک شبکه عصبی مصنوعی چندین لایه مختلف دارد و هر کدام از این لایه‌ها، یک سری اهداف خاص را نیز دنبال می‌کنند.



تصویر 1-1. ساختار یک شبکه عصبی

انواع مختلفی از شبکه‌های عصبی مصنوعی (ANN) وجود دارد که هر کدام دارای ویژگی‌ها و کاربردهای منحصر به فرد خود هستند.

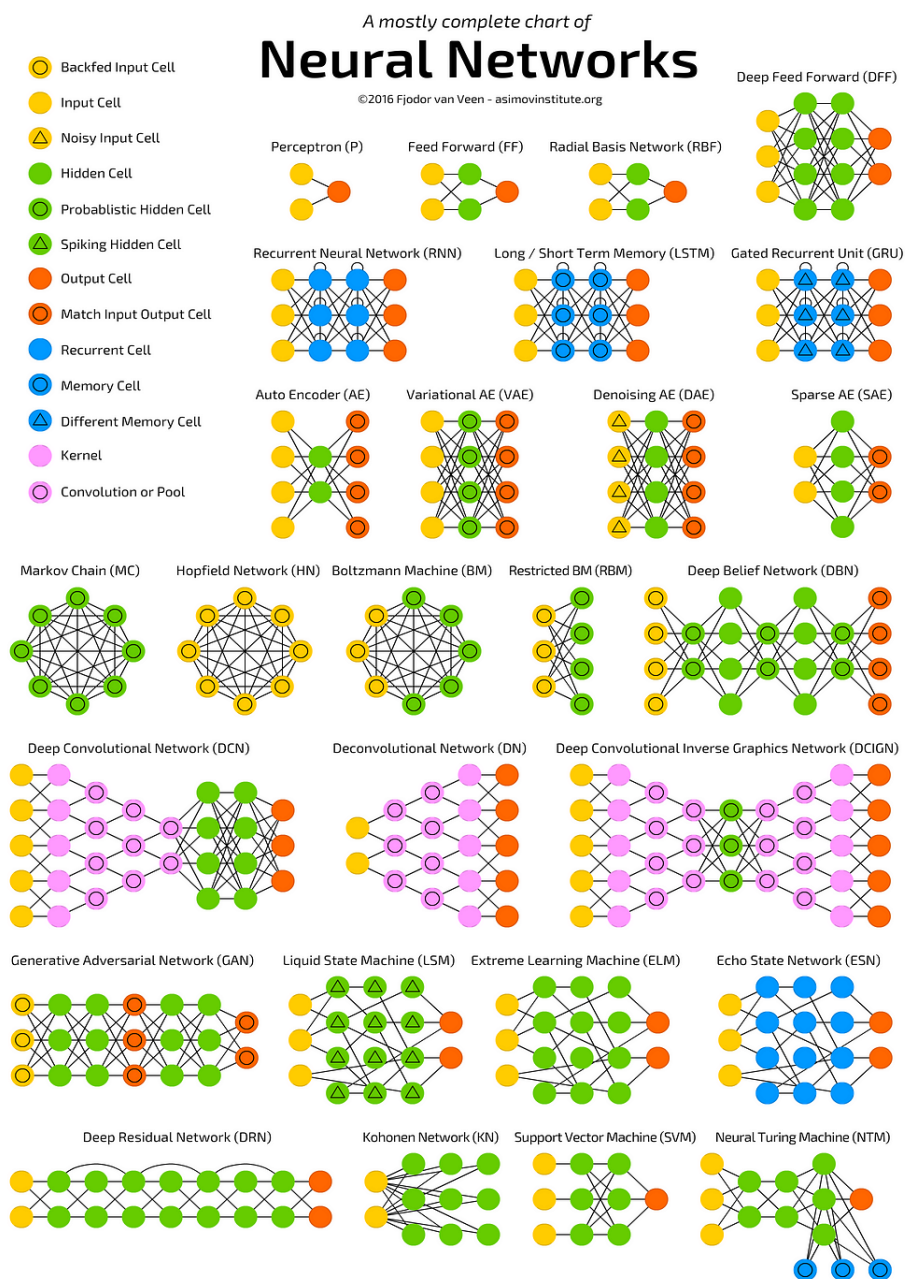
شبکه‌های عصبی پیشخور (FFNN): این نوع شبکه‌ها ساده‌ترین نوع شبکه عصبی مصنوعی هستند که اطلاعات تنها در یک جهت، از گره‌های ورودی به گره‌های خروجی، جریان می‌یابند. آن‌ها برای کارهایی مانند طبقه‌بندی، رگرسیون و تشخیص الگو استفاده می‌شوند.

شبکه‌های عصبی کانولوشنی (CNN): این شبکه‌ها نوعی از شبکه‌های عصبی هستند که معمولاً برای کارهای تشخیص تصویر و ویدیو استفاده می‌شوند. آن‌ها از لایه‌های کانولوشن برای استخراج ویژگی‌ها از تصاویر استفاده می‌کنند و در کارهایی مانند تشخیص اشیا (Object Detection) و بخش‌بندی تصویر (Image Segmentation) موفق بوده‌اند.

شبکه‌های عصبی بازگشتی (RNN): این شبکه‌ها، شبکه‌های عصبی هستند که برای پردازش داده‌های متوالی مانند سری‌های زمانی یا زبان طبیعی طراحی شده‌اند. آن‌ها از حلقه‌های بازخورد استفاده می‌کنند تا به اطلاعات اجازه دهند در طول زمان باقی بماند و در کارهایی مانند تشخیص گفتار (Speech Recognition) و ترجمه زبان موفق بوده‌اند.

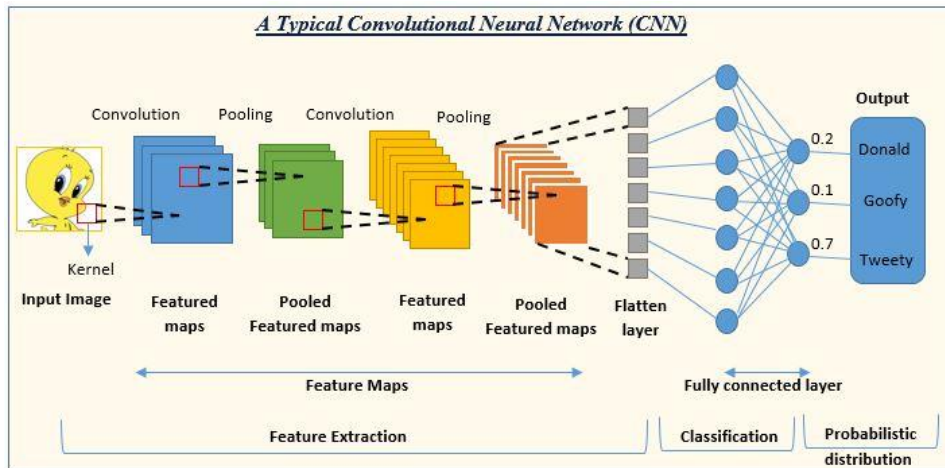
شبکه‌های عصبی رمزگذار خودکار (Autoencoder): این‌ها نوعی از شبکه‌های عصبی هستند که برای یادگیری بدون نظارت استفاده می‌شوند، جایی که داده‌های ورودی و خروجی یکسان هستند. رمزگذارهای خودکار معمولاً برای استخراج ویژگی، فشرده سازی تصویر و گفتار و تشخیص ناهنجاری استفاده می‌شوند.

شبکه‌های مولد متخاصم (GAN): این‌ها نوعی از شبکه‌های عصبی هستند که می‌توانند داده‌های جدیدی شبیه به داده‌های آموزشی تولید کنند. GAN‌ها معمولاً در سنتز تصویر و ویدئو استفاده می‌شوند. [11]



تصویر 1-2. انواع شبکه‌های عصبی

با توجه به ماهیت پروژه برای حل مسئله از شبکه‌های عصبی کانولوشنی استفاده شد. در ادامه تصویر مرتبط با این ساختار کلی این مدل از شبکه‌های عصبی را مشاهده می‌کنید:



تصویر 1-3. ساختار CNN

در فصل‌های آینده ساختار دقیق مدلی که استفاده و طراحی شد به طول کامل مورد بررسی قرار می‌گیرد.

۳-۲-۱. ارزیابی مدل

برای ارزیابی مدل تشخیص خواب‌آلودگی با استفاده از شبکه‌های عصبی کانولوشنی، می‌توان از معیارهای مختلفی استفاده کرد. این معیارها به ما کمک می‌کنند تا عملکرد مدل

را به طور کمی ارزیابی کرده و نقاط قوت و ضعف آن را شناسایی کنیم. برخی از معیارهای متداول برای ارزیابی مدل‌های طبقه‌بندی تصویر مانند تشخیص خواب‌آلودگی عبارتند از:

۱. دقت (Accuracy): این معیار نسبت پیش‌بینی‌های درست مدل را به کل پیش‌بینی‌ها نشان می‌دهد. در تشخیص خواب‌آلودگی، دقت نشان می‌دهد که چه درصدی از تصاویر به درستی به عنوان خواب‌آلود یا غیرخواب‌آلود طبقه‌بندی شده‌اند.

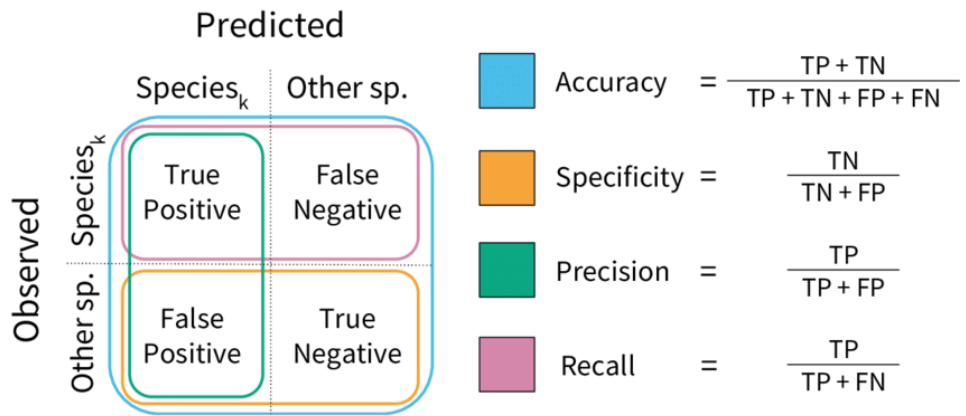
۲. صحت (Precision): صحت نسبت پیش‌بینی‌های درست مثبت (خواب‌آلود) را به کل پیش‌بینی‌های مثبت نشان می‌دهد. این معیار نشان می‌دهد که از بین تصاویری که مدل آن‌ها را خواب‌آلود پیش‌بینی کرده، چه تعدادی واقعاً خواب‌آلود بوده‌اند.

۳. بازیابی (Recall) یا حساسیت (Sensitivity): این معیار نسبت پیش‌بینی‌های درست مثبت را به کل موارد مثبت واقعی نشان می‌دهد. در تشخیص خواب‌آلودگی، بازیابی نشان می‌دهد که چه درصدی از تصاویر خواب‌آلود واقعی، توسط مدل به درستی شناسایی شده‌اند.

۴. F1-Score: این معیار ترکیبی از صحت و بازیابی است و میانگین هارمونیک آن‌ها را محاسبه می‌کند. هرچه F1-Score بالاتر باشد، نشان‌دهنده تعادل بهتر بین صحت و بازیابی است.

۵. ماتریس درهم‌ریختگی (Confusion Matrix): این ماتریس نشان می‌دهد که مدل چگونه داده‌ها را در هر طبقه طبقه‌بندی کرده است. در تشخیص خواب‌آلودگی، این ماتریس نشان می‌دهد که چه تعداد از تصاویر خواب‌آلود و غیرخواب‌آلود به درستی یا به اشتباه طبقه‌بندی شده‌اند.

با استفاده از این معیارها، می‌توانیم عملکرد مدل CNN خود را در تشخیص خواب‌آلودگی ارزیابی کرده و در صورت لزوم، تغییرات لازم را برای بهبود آن اعمال کنیم.



تصویر 4-1. معیارهای ارزیابی مدل

۲. مروری بر منابع

۲-۱. مقدمه

جاده ها شاهد افزایش تعداد وسایل نقلیه هستند، که منجر به افزایش تصادفات رانندگی شده است. این تصادفات به عنوان علت اصلی مرگ در بسیاری از نقاط جهان شناخته می شوند. راننده به عنوان مسئول، ایمنی خود و سرنشینانش را در دست دارد.

متاسفانه خواب آلودگی که اغلب به عنوان یک نگرانی ایمنی نادیده گرفته می شود، می تواند منجر به تصادفات و تلفات شود اگر مورد توجه قرار نگیرد. برای بهبود ایمنی جاده‌ای، ضروری است که مسئله خواب آلودگی رانندگان مورد توجه قرار گیرد و سیستم‌های تشخیص موثر پیاده‌سازی شوند. با توجه به تعداد زیادی از تصادفات در سراسر جهان که به دلیل خواب آلودگی رانندگان رخ می دهد، تشخیص و رسیدگی به این مسئله به یک جزء ضروری در سیستم های نظارت مدرن بر رانندگان تبدیل شده است. [7]

تشخیص خواب آلودگی راننده یک زمینه تحقیقاتی مهم در حوزه ایمنی حمل و نقل است. برای رسیدگی به این مسئله، بسیاری از محققان سیستم‌های تشخیص خواب آلودگی راننده را با استفاده از تکنیک‌های مختلف توسعه داده‌اند. گواک و همکاران [2] ویژگی‌های تشخیص خواب آلودگی را با استفاده از دوربین‌ها ارزیابی کردند. آن‌ها ویژگی‌ها را به ویژگی‌های دستی و ویژگی‌هایی که به طور خودکار با استفاده از شبکه‌های عصبی کانولوشنال (CNN) آموزش دیده بودند، تقسیم کردند. آن‌ها دقتی معادل 65.2 درصد را بر روی داده‌های ساختگی به دست آوردند. کپسیوا و همکاران [3] از ترکیبی از شبکه عصبی کانولوشنال (CNN)، شبکه عصبی کانولوشنال مبتنی بر دروازه کنترل بازگشتی (Conv GRNN) و یک لایه رای گیری برای تشخیص خواب آلودگی راننده استفاده کردند. آن‌ها به دقت متوسط 84.41 درصد دست یافتند. یو و همکاران [4] یک شبکه عصبی کانولوشنال

پشته شده عمیق را پیشنهاد دادند و به دقت متوسط 94.80 درصد دست یافتند. مهتا و همکاران [5] از نسبت چشم و نسبت بسته شدن چشم به همراه طبقه بندی کننده جنگل تصادفی برای دستیابی به دقت 84 درصد استفاده کردند. در نهایت، ساتاسیوام و همکاران [6] استفاده از نسبت چشم (EAR) و طبقه بندی کننده ماشین بردار پشتیبان (SVM) را پیشنهاد دادند که به دقتی نزدیک به 94 درصد دست یافت. در مجموع، این مطالعات نشان دادند که استفاده از CNN ها و ویژگی های دستی، همراه با طبقه بندی کننده هایی مانند جنگل تصادفی و SVM، می تواند دقت خوبی در تشخیص خواب آلودگی راننده داشته باشد.

۲-۲. انتخاب دیتاست

چندین مجموعه داده استاندارد وجود دارد که محققان از آنها برای تشخیص خواب آلودگی استفاده کرده اند. در ادامه نمونه ای از داده های این دیتاست ها را مشاهده می کنید:

دیتاست ۱:



تصویر 1-2. نمونه تصویر از چشم بسته در دیتاست ۱



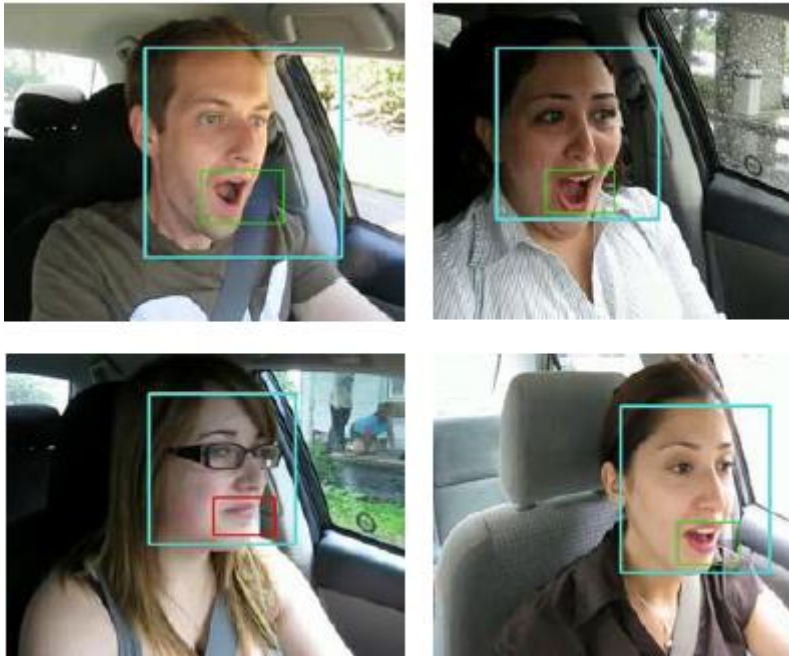
تصویر 2-2. نمونه تصویر از چشم باز در دیتاست ۱

دیتاست ۲:

YAWNING DETECTION DATASET (YAWDD) یک مجموعه داده از ویدیوهای ضبط شده توسط یک دوربین داخل خودرو از رانندگان در یک خودروی واقعی با ویژگی‌های چهره‌ای مختلف (مرد و زن، با و بدون عینک/عینک آفتابی، نژادهای مختلف) در حال صحبت کردن، آواز خواندن، ساکت بودن و خمیازه کشیدن است. می‌تواند بیشتر برای توسعه و آزمون الگوریتم‌ها و مدل‌ها برای تشخیص خمیازه، اما همچنین شناسایی و ردیابی چهره و دهان استفاده شود. ویدیوها در شرایط نوری طبیعی و متغیر ضبط شده‌اند. ویدیوها در دو مجموعه ارائه می‌شوند، که در ادامه توضیح داده می‌شود:

در مجموعه اول، دوربین در زیر آینه جلوی خودرو نصب شده است. این مجموعه 322 ویدیو را ارائه می‌دهد، هر کدام برای یک موقعیت متفاوت: 1- رانندگی عادی (بدون صحبت)، 2- صحبت کردن یا آواز خواندن در حین رانندگی، و 3- خمیازه کشیدن در حین رانندگی. هر شخص 3 یا 4 ویدیو دارد.

در مجموعه دوم، دوربین روی داشبورد راننده نصب شده است. این مجموعه 29 ویدیو را ارائه می‌دهد، یکی برای هر شخص، و هر ویدیو شامل رانندگی در سکوت، رانندگی در حین صحبت کردن و رانندگی در حین خمیازه کشیدن است. [8]



تصویر 2-3. نمونه تصویر از دیتاست ۲

پس از بررسی این مجموعه‌ها و مقایسه آن‌ها با توجه به این که دیتاست اول فقط شامل تصویر چشم بود دیتاستی که در ادامه معرفی می‌شود برای ادامه کار انتخاب شد.

مجموعه داده مورد استفاده، مجموعه داده Closed Eyes in the Wild (CEW) [6] است. این مجموعه داده شامل 2423 نفر است، که 1192 نفر چشمان بسته و 1231 نفر چشمان باز دارند.

در ادامه نمونه‌ای از داده‌های این دیتاست‌ها را مشاهده می‌کنید:



تصویر 2-4. نمونه تصاویر از چشم بسته در دیتاست CEW



تصویر 2-5. نمونه تصاویر از چشم باز در دیتاست CEW

۳-۲. پیش پردازش

طبق مقاله [1] پیش پردازش تصاویر ورودی برای تغذیه به مدل یادگیری عمیق است. تابع `get_data()` تصاویر را می خواند، آن ها را به اندازه 145×145 پیکسل تغییر اندازه می دهد و به آرایه ها تبدیل می کند. به همین ترتیب، تابع `face_for_yawn()` تصاویر را می خواند، چهره ها را در تصاویر با استفاده از فایل `haarcascade_frontalface_default.xml` تشخیص می دهد، چهره های تشخیص داده شده را به اندازه 145×145 پیکسل تغییر اندازه می دهد و آن ها را همراه با برچسب مربوطه به یک لیست اضافه می کند. سپس داده های پیش پردازش شده با استفاده از تابع `append_data()` ترکیب می شوند که تصاویر چهره را با تصاویر چشم ادغام می کند و آنها را به صورت یک آرایه Numpy باز می گرداند. در نهایت، داده های ورودی با استفاده از تابع `train_test_split()` به مجموعه های آموزشی و آزمایشی تقسیم می شوند و با استفاده از کلاس `ImageDataGenerator()` تبدیل می شوند که پردازش های تصویری

اضافی مانند بزرگنمایی، چرخش و واژگونی تصاویر را انجام می‌دهد تا حجم داده‌های آموزشی را افزایش دهد و عملکرد مدل را بهبود بخشد.

آنچه در این راهنما می‌آید، دربرگیرنده چگونگی نگارش پارسا است. مخاطبان اصلی این راهنما، دانشجویانی هستند که پارسا را به زبان فارسی می‌نویسند؛ بنابراین، این راهنما برای زبان فارسی است؛ ولی توصیه‌ها و رهنمودهایی نیز برای دانشجویانی که آن را به زبان‌هایی همچون عربی، انگلیسی، فرانسه و ... می‌نویسند، در بخش‌های جداگانه آمده است.

۱-۳-۲. افزایش داده‌ها

افزایش داده‌ها یک فرایند است که با ایجاد تنوع در داده‌های تصویری اصلی، تعمیم‌پذیری مدل را بهبود می‌بخشد. در این مورد [1]، دو تکنیک افزایش داده‌ها پیاده‌سازی شده است:

برش تصادفی برای برش تصادفی یک قسمت از تصویر و تغییر اندازه آن به اندازه اصلی تصویر استفاده می‌شود. این تکنیک به مدل کمک می‌کند تا ویژگی‌ها را از قسمت‌های مختلف تصویر یاد بگیرد و در نتیجه آن را در برابر تغییرات در موقعیت چشم‌ها یا چهره در تصویر مقاوم‌تر می‌کند. در [1]، این کار با استفاده از تابع `tf.image.random_crop()` انجام می‌شود.

واژگونی افقی برای واژگون کردن افقی تصویر استفاده می‌شود. این تکنیک به مدل کمک می‌کند تا ویژگی‌ها را از جهت‌های مختلف یاد بگیرد و در نتیجه آن را در برابر تغییرات در جهت‌گیری سر یا چهره در تصویر مقاوم‌تر می‌سازد. در [1] این کار با استفاده از تابع `tf.image.random_flip_left_right()` انجام می‌شود.

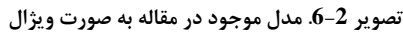
این تکنیک‌های افزایش داده‌ها در طول هر دوره آموزش با استفاده از API `tf.data.Dataset` بر روی داده‌های آموزشی اعمال می‌شوند. این امر اطمینان می‌دهد که مدل بر روی تنوع‌های مختلفی از تصاویر آموزش می‌بیند و در نتیجه عملکرد آن بر روی داده‌های ناشناخته بهبود می‌یابد. با استفاده از افزایش داده‌ها، مدل می‌تواند بیاموزد که خواب‌آلودگی را در شرایط مختلف، مانند شرایط نوری متفاوت، موقعیت‌های مختلف سر و چشم و ظاهرهای متفاوت راننده، شناسایی کند.

۴-۲. پیاده‌سازی مدل

مدل از یک سری لایه‌های کانولوشن و مکس پولینگ تشکیل شده است که در ادامه چند لایه کاملاً متصل قرار دارد. شکل ورودی CNN بر اساس شکل داده‌های آموزشی که در این مورد یک تنسور ۴ بعدی نمایانگر تصاویر با ابعاد (ارتفاع، عرض، کانال‌ها) و یک اندازه دسته تعیین می‌شود. اولین لایه کانولوشنی دارای 512 فیلتر، با اندازه هسته 3×3 و تابع فعال‌سازی ReLU است. پس از آن یک لایه مکس پولینگ با اندازه پولینگ 2×2 قرار دارد که ابعاد مکانی نقشه ویژگی را کاهش می‌دهد. لایه‌های بعدی همین الگو را دنبال می‌کنند و به تدریج تعداد فیلترها و ابعاد نقشه ویژگی را کاهش می‌دهند. لایه Flatten خروجی آخرین لایه کانولوشنی را به یک تنسور یک بعدی تغییر شکل می‌دهد که سپس از طریق یک لایه Dropout برای کاهش بیش‌برازش عبور می‌کند. در نهایت، لایه‌های کاملاً متصل شامل دو لایه پنهان با فعال‌سازی ReLU و یک لایه خروجی با تابع فعال‌سازی نرمال شده است که یک توزیع احتمال بر روی چهار کلاس ممکن تولید می‌کند. مدل با استفاده از تابع زیان categorical_crossentropy، معیار دقت و بهینه‌ساز Adam کامپایل می‌شود. این مدل CNN می‌تواند بر روی یک مجموعه داده طبقه‌بندی تصویر آموزش و ارزیابی شود. [1]

جدول 2-1. مشخصات مدل مقاله

| Layer Type | Configuration |
|-------------------------------------|------------------------------------|
| Convolutional Layers | 5 |
| Filters in Convolutional Layers | 512, 512, 256, 256, 256 |
| Kernel Size in Convolutional Layers | (3, 3) |
| Max Pooling Layers | 5 |
| Pooling Size in Max Pooling Layers | (2, 2) |
| Dense Layers | 3 |
| Units in Dense Layers | 128, 64, 4 |
| Loss Function | Categorical Crossentropy (SoftMax) |
| Optimizer | Adam |
| Learning Rate | 0.001 (default of Adam) |
| Number of Parameters | 4,873,924 |



CNN شامل چندین لایه کانولوشنی با اندازه‌های فیلتر افزایشی است که پس از آن لایه‌های مکس پولینگ قرار دارند که ابعاد مکانی نقشه‌های ویژگی را کاهش می‌دهند.

خروجی آخرین لایه مکس پولینگ صاف شده و از طریق یک لایه کاملاً متصل (متراکم) که دارای regularization حذف موقت برای کاهش بیش برآزش است، عبور می‌کند. در نهایت، لایه خروجی دارای چهار گره است که نمایانگر چهار کلاس (خمیازه، بدون خمیازه، چشم بسته و چشم باز) است و از تابع فعال‌سازی نرمال شده برای تولید احتمالات کلاس استفاده می‌کند.

CNN با استفاده از بهینه‌ساز Adam و تابع زیان آنتروپی متقاطع دسته‌ای آموزش داده می‌شود. مدل برای 50 دوره، با اندازه دسته 32 آموزش دیده می‌شود. داده‌های اعتبارسنجی برای ارزیابی مدل پس از هر دوره استفاده می‌شود و بهترین مدل ذخیره می‌شود. پس از اتمام آموزش، مدل بر روی مجموعه آزمون برای تعیین عملکرد آن بر روی داده‌های ناشناخته ارزیابی می‌شود. بنابراین، خواب‌آلودگی با استفاده از CNN آموزش دیده برای طبقه‌بندی تصاویر چهره و چشم‌های افراد به عنوان هوشیار یا خواب‌آلود بر اساس پیش‌بینی‌های مدل تشخیص داده می‌شود.

مدل آموزش دیده در یک سیستم تشخیص خواب‌آلودگی راننده که از فیلم‌برداری لحظه‌ای دوربین استفاده می‌کند، ادغام خواهد شد. اگر مدل خواب‌آلودگی را تشخیص دهد، یک هشدار به راننده برای استراحت یا توقف رانندگی هشدار می‌دهد که می‌تواند تصادفات ناشی از خواب‌آلودگی راننده را کاهش دهد.

توجه داشته باشید در مدلی که ما در فصل آینده استفاده خواهیم برخلاف این مدل فقط وضعیت چشم‌ها مورد بررسی قرار گرفته می‌شود و عملاً یک دست‌بندی باینری را انجام می‌دهیم.

۵-۲. عملکرد و نتایج

سیستم پیشنهادی تشخیص خواب‌آلودگی راننده [1] بر روی دو مجموعه داده YawDD و CEW آموزش دیده و ارزیابی شده است. حداکثر دقت آزمون بر روی مجموعه داده YawDD به 97.06٪ و حداکثر دقت آموزش به 97.62٪ رسید. حداقل زیان آموزش 0.0628 و

حداقل زیان آزمون 0.0755 بود. مدل آموزش دیده همچنین بر روی تصاویر آزمون ارزیابی شد و دقت آن 96.02٪ و زیان آن 0.1072 بود.

سیستم پیشنهادی تشخیص خواب‌آلودگی راننده [1]، مبتنی بر الگوریتم‌های بینایی کامپیوتر و یادگیری عمیق، دقت بالایی در تشخیص خواب‌آلودگی راننده بر روی دو مجموعه داده YawDD و CEW نشان داده است. مجموعه داده YawDD به طور خاص به دلیل تنوع در قاب‌های خمیازه و بدون خمیازه، چالش برانگیز است. با این حال، سیستم نتایج عالی را بر روی این مجموعه داده به دست آورد که نشان دهنده اثربخشی آن در سناریوهای واقعی است.

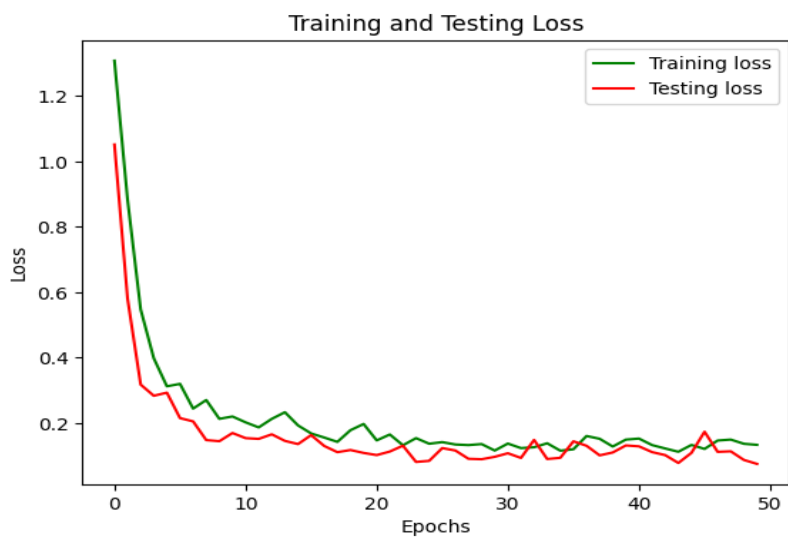
در نتیجه، سیستم پیشنهادی تشخیص خواب‌آلودگی راننده می‌تواند یک ابزار موثر برای نظارت لحظه‌ای بر راننده و تشخیص خواب‌آلودگی باشد. نتایج ارزیابی سیستم بر روی دو مجموعه داده، دقت و اثربخشی بالای آن را نشان می‌دهد و ادغام آن در یک سیستم کامل تشخیص خواب‌آلودگی راننده، پتانسیل آن را برای بهبود ایمنی راننده افزایش می‌دهد. این سیستم می‌تواند یک افزودنی ارزشمند به سیستم‌های موجود نظارت بر راننده باشد و بالقوه می‌تواند جان‌ها را با جلوگیری از تصادفات ناشی از خواب‌آلودگی راننده نجات دهد.

جدول 2-2. نتایج و ارزیابی مدل مقاله

| Metric | Value |
|---------------------------|--------|
| Maximum Testing Accuracy | 97.06 |
| Maximum Training Accuracy | 97.62 |
| Minimum Training Loss | 0.0628 |
| Minimum Testing Loss | 0.0755 |



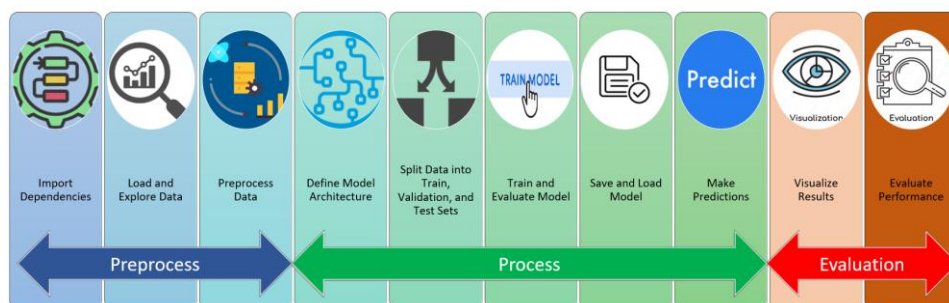
نمودار 2-1. Training and Testing Accuracy



نمودار 2-2. Training and Testing Loss

۳. راه حل پیشنهادی

۳-۱. دیاگرام



تصویر 3-1. دیاگرام راه حل پیشنهادی

۳-۲. توضیح مراحل دیاگرام

۳-۲-۱. پیش پردازش داده‌ها

۳-۲-۱-۱. وارد کردن کتابخانه‌های مورد نیاز

در این قسمت، کتابخانه‌های مورد نیاز برای پردازش تصویر، پردازش داده، ریاضیات، پلات کردن و آموزش مدل شبکه عصبی وارد می‌شوند. کتابخانه‌هایی مانند cv2، pandas، numpy، matplotlib، sklearn، tensorflow و keras.

۳-۲-۱-۲. بارگذاری و بررسی داده‌ها

در این بخش داده‌ها از پوشه‌های train بارگذاری می‌شوند. یک تصویر نمونه از پوشه Open_Eyes بارگذاری و نمایش داده می‌شود. سپس ابعاد این تصویر چک می‌شود. بعد از آن با یک حلقه تمام تصاویر در پوشه‌های Closed_Eyes و Open_Eyes را یکی یکی بارگذاری و نمایش می‌دهد.

۳-۲-۱-۳. پیش پردازش داده‌ها

این مرحله شامل پیش پردازش داده‌ها برای آماده سازی آن‌ها برای مدل است. تمام تصاویر به اندازه 224×224 پیکسل تغییر اندازه داده می‌شوند و به فرمت RGB تبدیل می‌شوند. سپس این تصاویر پیش پردازش شده و برچسب‌های آن‌ها در لیستی به نام training_data ذخیره می‌شوند.

۳-۲-۲. پردازش داده‌ها

۳-۲-۲-۱. تعریف معماری مدل

در این قسمت، معماری مدل شبکه عصبی تعریف می‌شود. از تکنیک Transfer Learning با استفاده از مدل پیش‌آموزش شده MobileNet استفاده می‌شود که در ادامه به معرفی آن خواهیم پرداخت. چند لایه جدید شامل Flatten و Dense به آخر مدل اضافه می‌شود تا مدل را برای مسئله دو کلاسی بسته یا باز بودن چشم‌ها سفارشی کند.

۳-۲-۲-۲. تقسیم داده‌ها به مجموعه‌های آموزش، اعتبارسنجی و آزمایش

داده‌های آماده شده به سه دسته آموزش، اعتبارسنجی و آزمایش با نسبت‌های 70٪، 15٪ و 15٪ تقسیم می‌شوند.

۳-۲-۲-۳. آموزش و ارزیابی مدل

در این مرحله، مدل روی داده‌های آموزشی با استفاده از داده‌های اعتبارسنجی آموزش داده می‌شود. تابع fit از keras برای این کار استفاده می‌شود. تعداد epoch ۴۰ تعریف شده است.

۳-۲-۲-۴. ذخیره و بارگذاری مدل

پس از آموزش، مدل آموزش دیده در یک فایل ذخیره می‌شود. سپس همین مدل ذخیره شده برای استفاده در مراحل بعدی بارگذاری می‌شود.

۳-۲-۲-۵. پیش‌بینی

در این قسمت، دو روش مختلف برای پیش‌بینی روی داده‌های آزمایشی نشان داده شده است:

الف) در نسخه اول، یک شاخص تصادفی از مجموعه داده‌های آزمایشی انتخاب می‌شود. تصویر مربوطه بارگذاری و نمایش داده می‌شود. سپس مدل، برچسب پیش‌بینی شده را با یک آستانه 0.9 محاسبه می‌کند. برچسب واقعی و پیش‌بینی شده چاپ می‌شوند.

ب) در نسخه دوم، یک شاخص خاص از مجموعه داده‌های آزمایشی انتخاب می‌شود. تصویر مربوطه بارگذاری می‌شود. مدل روی آن پیش‌بینی انجام می‌دهد و برچسب پیش‌بینی شده را با آستانه 0.9 محاسبه می‌کند. برچسب واقعی و پیش‌بینی شده چاپ می‌شوند و تصویر همراه با برچسب‌ها نمایش داده می‌شود.

۳-۲-۳. ارزیابی

۳-۲-۳-۱. مصورسازی نتایج

در این بخش، روندهای مختلف مربوط به آموزش مدل رسم می‌شوند. سه نمودار جداگانه

برای loss تنها در حالت آموزش، loss در حالت آموزش و اعتبارسنجی، و دقت در حالت آموزش و اعتبارسنجی رسم می‌شوند.

۱-۳-۲. ارزیابی عملکرد

در نهایت، در این مرحله Accuracy، Precision، Recall و F1-Score مدل روی داده‌های آزمایش محاسبه و چاپ می‌شود.

۳-۳. موبایل نت

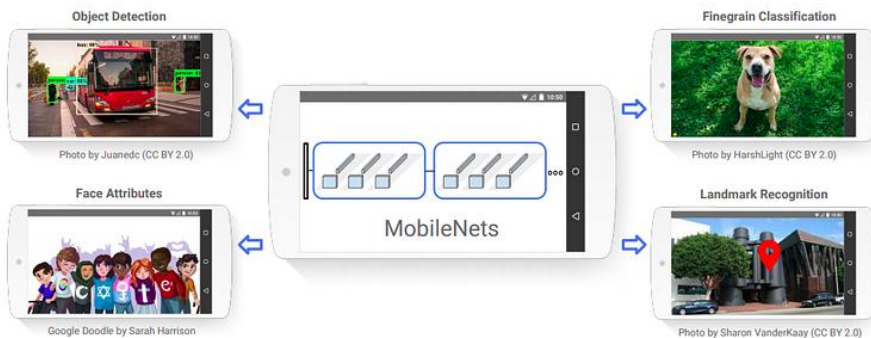
۱-۳-۳. مقدمه

شبکه‌های عصبی موبایل (MobileNets) نوعی مدل هوش مصنوعی هستند که برای امکان طبقه‌بندی تصاویر کارآمد و دقیق و تشخیص اشیا بر روی دستگاه‌های موبایل و تعبیه‌شده طراحی شده‌اند. این مدل‌ها به طور خاص برای اجرا بر روی دستگاه‌های کم مصرف با منابع محاسباتی محدود، بهینه‌سازی شده‌اند، که آن‌ها را برای موبایل و اپلیکیشن‌های لبه‌ای ایده‌آل می‌کند. موبایل نت در سال‌های اخیر محبوبیت فزاینده‌ای پیدا کرده‌است، به‌ویژه در توسعه برای موبایل و اپلیکیشن‌های لبه‌ای که نیازمند پردازش تصویر و قابلیت‌های بینایی ماشین هستند. آن‌ها در طیف گسترده‌ای از صنایع از جمله بهداشت و درمان، خدمات مالی، خرده‌فروشی و سرگرمی مورد استفاده قرار گرفته‌اند و حتی در دستگاه‌ها و اپلیکیشن‌های محبوب موبایل نیز گنجانده شده‌اند. [10]

۲-۳-۳. معرفی

شبکه‌های عصبی موبایل (MobileNets) یک خانواده از معماری‌های کارآمد شبکه‌های عصبی کانولوشنال هستند که توسط محققان گوگل در سال 2017 ایجاد شدند. هدف از MobileNets طراحی شبکه‌های عصبی بود که برای انتشار بر روی دستگاه‌های موبایل و تعبیه‌شده با منابع محاسباتی محدود، بهینه‌سازی شده باشند. محققان با استفاده از دو تکنیک کلیدی، کانولوشن‌های جداگانه عمقی و گلوگاه‌های خطی، به این هدف دست یافتند. معماری اصلی MobileNet دارای 28 لایه بود و روی مجموعه داده ImageNet برای

طبقه‌بندی تصاویر آموزش دیده شد. این شبکه دارای اندازه ورودی 224×224 پیکسل بود و برای دستیابی به دقت بالا در حالی که از پارامترهای کمتر و محاسبات کمتری نسبت به شبکه‌های عصبی کانولوشنال سنتی استفاده می‌کرد، طراحی شده بود. از زمان معرفی معماری اصلی MobileNet، نسخه‌های متعددی از جمله MobileNetV2، MobileNetV3-Small و MobileNetV3-Large پیشنهاد شده‌اند که بر پایه طراحی اولیه بنا شده و کارایی و دقت شبکه را بیشتر بهبود می‌بخشند.. [9]



تصویر 3-2. نمایی از Mobilenet

۳-۳-۳. معماری

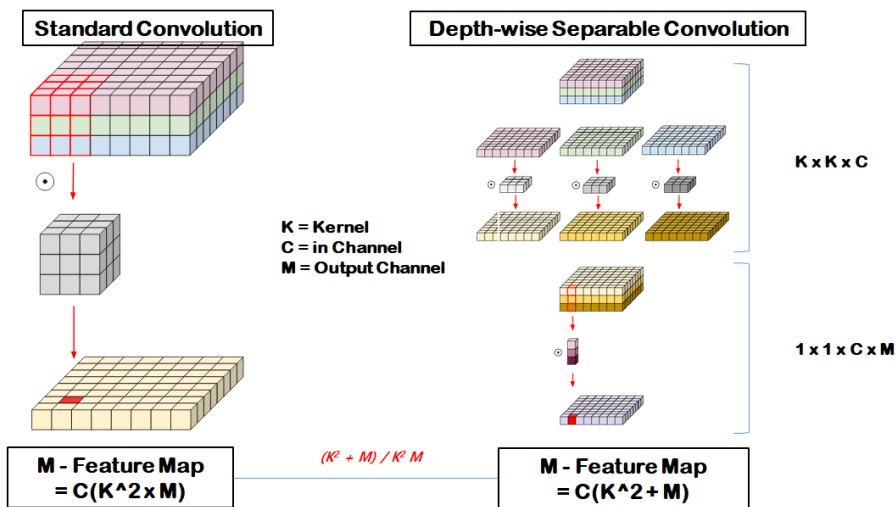
معماری MobileNet از دو نوع لایه اصلی تشکیل شده است: لایه‌های کانولوشن عمقی و لایه‌های کانولوشن نقطه‌ای.

کانولوشن عمقی یک لایه است که در آن نقشه‌های ویژگی ورودی با یک مجموعه از فیلترها تراوش می‌شوند، که هر فیلتر تنها یک کانال ورودی را پوشش می‌دهد. این بدان معنی است که هر کانال ورودی به طور جداگانه تراوش می‌شود و نقشه‌های ویژگی خروجی دارای تعداد کانال‌های یکسان با نقشه‌های ویژگی ورودی هستند.

کانولوشن نقطه‌ای یک مجموعه از کانولوشن‌های 1×1 است که به خروجی کانولوشن عمقی اعمال می‌شود. این شامل تراوش نقشه‌های ویژگی خروجی با یک مجموعه از فیلترهای 1×1 است، که هر فیلتر تمام کانال‌های ورودی را پوشش می‌دهد. این عملیات

تعداد کانال‌ها را در نقشه‌های ویژگی خروجی افزایش می‌دهد، در حالی که ابعاد مکانی آن‌ها را کاهش می‌دهد. [9]

با تقسیم کانولوشن استاندارد به این دو عملیات جداگانه، کانولوشن جداگانه عمقی تعداد پارامترها و محاسبات مورد نیاز را کاهش می‌دهد، در حالی که دقت را حفظ یا حتی بهبود می‌بخشد. این آن را به ویژه برای دستگاه‌های موبایل و تعبیه‌شده با منابع محاسباتی محدود مفید می‌کند.



تصویر 3-3. کانولوشن قابل تفکیک عمقی vs استاندارد

۴-۳-۳. دلایل انتخاب مدل

موارد استفاده موبایل‌نت به شرح زیر است:

تشخیص اشیا: MobileNet ها به طور معمول برای تشخیص اشیا در کاربردهای زمان واقعی مانند خودروهای خودران، سیستم های نظارتی و رباتیک مورد استفاده قرار می گیرند. MobileNet ها می توانند با دقت بالا اشیا را در تصاویر و ویدیوها تشخیص دهند و کارایی آنها آنها را برای پردازش زمان واقعی مناسب می کند.

طبقه بندی تصاویر: MobileNet ها به طور گسترده برای کاربردهای طبقه بندی تصاویر

مانند تشخیص و دسته بندی تصاویر مورد استفاده قرار می گیرند. آنها می توانند تصاویر را با دقت بالا طبقه بندی کنند و کارایی آنها را برای دستگاه های موبایل و تعبیه شده مناسب می سازد.

تشخیص چهره: MobileNet ها می توانند برای تشخیص چهره در کاربردهای موبایل و تعبیه شده مانند سیستم های امنیتی و اپلیکیشن های موبایل استفاده شوند.

انتقال سبک: MobileNet ها می توانند برای انتقال سبک، که شامل اعمال سبک یک تصویر به تصویر دیگر است، مورد استفاده قرار گیرند. این برای کاربردهایی مانند ایجاد تصاویر و ویدیوهای هنری مفید است.

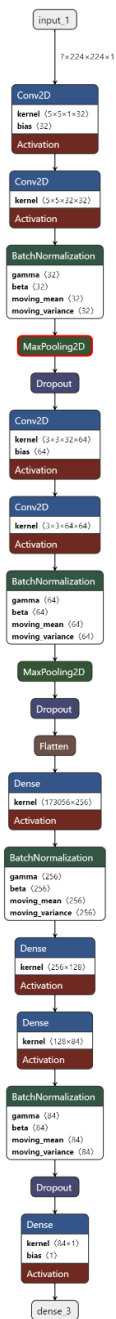
تحلیل ویدیو: MobileNet ها می توانند برای تحلیل ویدیو، که شامل تجزیه و تحلیل داده های ویدیویی در زمان واقعی است، استفاده شوند. این برای کاربردهایی مانند سیستم های نظارتی، پایش ترافیک و تشخیص فعالیت مفید است.

با توجه به کاربردهای موبایل نت و کاربرد مناسب آن برای پروژه ما از MobileNet در این پروژه تشخیص خواب آلودگی استفاده کردیم چون نیاز داشتیم یه مدل کارآمد و دقیق برای طبقه بندی تصاویر چشم باز و بسته با منابع محاسباتی محدود داشته باشیم در نتیجه تصمیم گرفتیم از این مدل استفاده کنیم و همانطور که در بخش پیاده سازی و نتایج مشاهده خواهید کرد نتیجه مطلوب نیز حاصل می شود.

۴-۳. مدل طراحی شده

در ابتدا مدلی طراحی کرده بودم که در ادامه معماری آن را مشاهده می کنید دقت حاصل از این مدل تقریباً بین ۸۷-۹۱ درصد شده بود و با توجه به این که نسب به مدل مقاله [] دقت کمتری داشت در اینجا به قرار دادن نتایج بسنده می کنم و در فصل بعد درباره پیاده سازی آن توضیحی داده نمی شود.

برای ویژال کردن مدل از netron.app استفاده شد و نتیجه به صورت زیر است:



تصویر 3-4. نمایی از مدل طراحی شده به صورت ویزال

```

model.evaluate(x_test, y_test)
✓ 3.8s

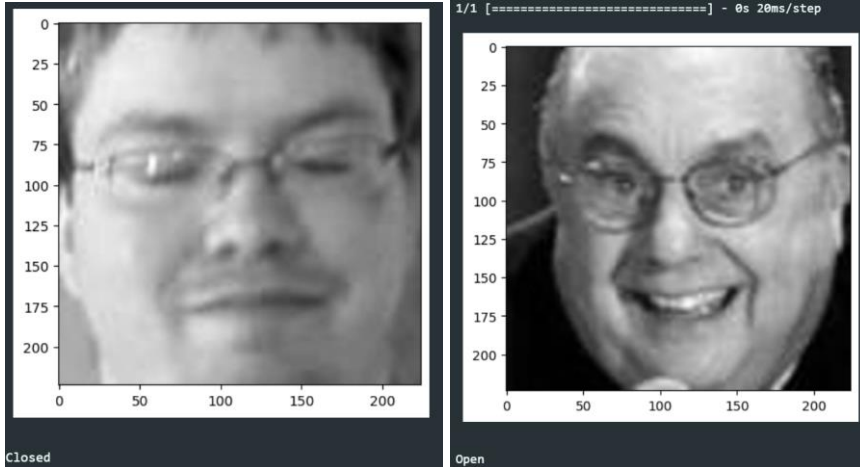
25/25 [=====] - 4s 149ms/step - loss: 0.6581 - accuracy: 0.8225
[0.6581031084060669, 0.8224999904632568]

from keras.models import load_model
best_model = load_model('bestModel.h5')
best_model.evaluate(x_test, y_test)
✓ 4.5s

25/25 [=====] - 4s 149ms/step - loss: 0.4162 - accuracy: 0.8700
[0.41618263721466064, 0.8700000047683716]
    
```

تصویر 3-5. ارزیابی مدل طراحی شده

حال چندین مورد از عکس‌ها را به صورت رندم انتخاب کرده و نتایج با این که دقت تقریباً ۹۰ درصد است به صورت اتفاقی تمام تصاویر را درست پیش‌بینی شد:



تصویر 3-6. پیش‌بینی با استفاده از مدل طراحی شده

به دلیل حجم زیاد فقط دو نمونه قرار داده شد جهت مشاهده سایر موارد می‌توانید به فایل مربوط به کد مراجعه کنید.

۴. پیاده‌سازی و تحلیل نتایج

۴-۱. مقدمه

در این پروژه، با استفاده از زبان برنامه‌نویسی پایتون و از کتابخانه‌هایی نظیر OpenCV، numpy، pandas، matplotlib برای پیش‌پردازش تصاویر و scikit-learn برای ارزیابی مدل، سعی داریم تا یک سیستم تشخیص خواب‌آلودگی را پیاده‌سازی کنیم. این سیستم قادر است تصاویر افراد را از دو دسته‌ی چشم‌های باز و چشم‌های بسته تشخیص دهد.

۴-۲. پیش‌پردازش

مرحله‌ی پیش‌پردازش در این پروژه با خواندن تصاویر و تبدیل آنها به آرایه‌های سیاه و سفید آغاز می‌شود. سپس تصاویر به ابعاد مناسب برای ورود به مدل (۲۲۴ در ۲۲۴ پیکسل) تغییر اندازه می‌یابند. در این مرحله، تمام تصاویر در دو دسته‌ی چشم‌های باز و چشم‌های بسته دسته‌بندی شده و به همراه برچسب متناظرشان در آرایه‌های ورودی و خروجی ذخیره

می‌شوند. سپس مقادیر پیکسل‌ها نرمال‌سازی می‌شوند تا مقادیر آنها در بازه‌ی صفر تا یک قرار گیرد.

۳-۴. پردازش

در مرحله‌ی پردازش، از یک مدل پیش‌آموز به نام MobileNet برای ایجاد مدل پیش‌بینی استفاده می‌شود. لایه‌های جدیدی به مدل اضافه می‌شود تا با وظیفه‌ی تشخیص خواب‌آلودگی سازگار شود که به شرح زیر است:

لایه: Flatten

این لایه پس از لایه‌های پیچشی قرار می‌گیرد و وظیفه‌ی آن تبدیل خروجی لایه‌های پیچشی به یک بردار یک‌بعدی است. این عمل باعث می‌شود که اطلاعات استخراج شده از تصویر به شکل یک بردار برای لایه‌های بعدی ارسال شود.

لایه: Dense

در این لایه، اطلاعاتی که توسط لایه Flatten به دست آمده‌اند، به صورت پرکردنی به این لایه وارد می‌شود و به عنوان ورودی برای این لایه استفاده می‌شود. این لایه شامل یک تابع فعال‌سازی sigmoid است که از آن برای تبدیل خروجی به فرم احتمال استفاده می‌شود تا بتوانیم احتمال تشخیص خواب‌آلودگی را بین ۰ و ۱ بدست آوریم.

سپس مدل با استفاده از تابع هزینه binary_crossentropy و بهینه‌ساز adam کامپایل می‌شود. در آخر، مدل با داده‌های آموزشی آموزش داده می‌شود و از داده‌های اعتبارسنجی برای ارزیابی استفاده می‌شود.

۴-۴. تحلیل نتایج

در انتها به بررسی نتایج به دست آمده می‌پردازیم که به شرح زیر است: عملکرد مدل بر روی داده‌های آموزش و اعتبارسنجی با استفاده از معیارهای دقت، بازیابی و اسکور F1

گزارش می شود. همچنین دقت مدل بر روی داده های آزمون محاسبه شده و گزارش می شود.

```
train report
53/53 [=====] - 5s 85ms/step
      precision    recall  f1-score   support

     0       1.00      1.00      1.00       819
     1       1.00      1.00      1.00       877

 accuracy          1.00       1696
 macro avg          1.00       1696
weighted avg          1.00       1696
```

تصویر 4-1. گزارش از عملکرد مجموعه آموزش مدل نهایی

```
validation report
12/12 [=====] - 1s 80ms/step
      precision    recall  f1-score   support

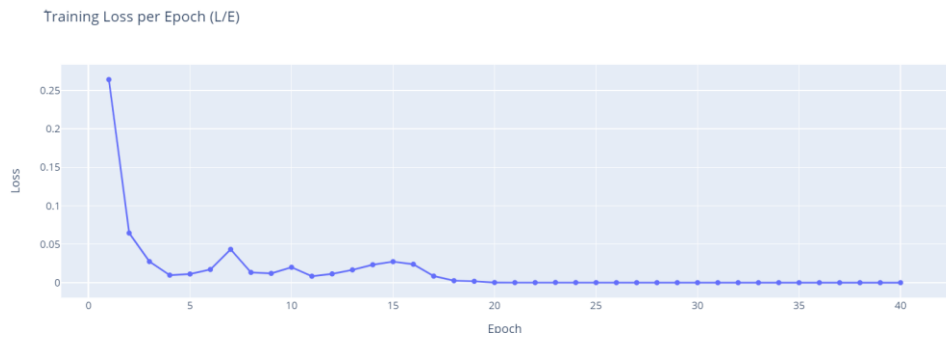
     0       0.97      0.99      0.98       190
     1       0.99      0.97      0.98       173

 accuracy          0.98       363
 macro avg          0.98       363
weighted avg          0.98       363
```

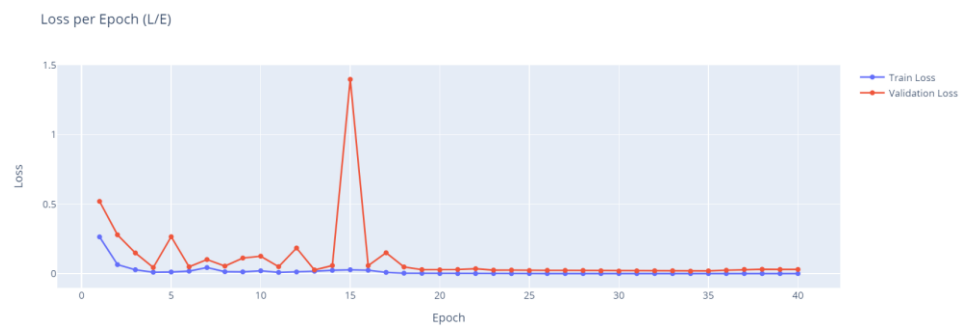
تصویر 4-2. گزارش از عملکرد مجموعه Validation مدل نهایی

```
12/12 [=====] - 1s 80ms/step - loss: 0.0388 - accuracy: 0.9918
Test Accuracy: 0.9917582273483276
```

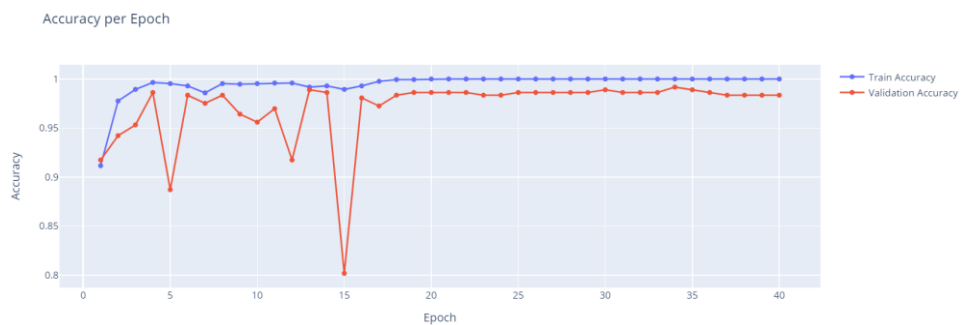
تصویر 4-3. دقت مدل نهایی برای مجموعه آزمایش



نمودار 1-4 Training Loss



نمودار 2-4 Training and Validation Loss



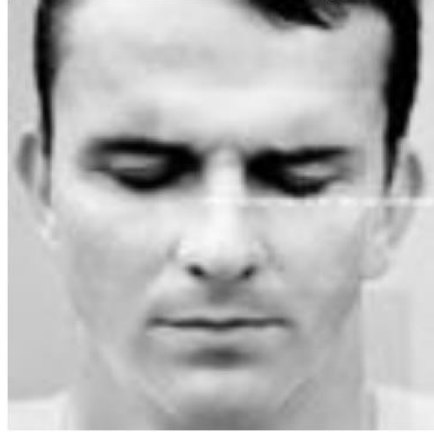
نمودار 3-4 Training and Validation Accuracy

در نهایت چندین مورد از پیش‌بینی‌های انجام شده که تصاویر را به صورت رندم و یا با ایندکس مشخص انتخاب کرده‌اند مشاهده می‌کنید و پیش‌بینی‌ها همانطور که انتظار می‌رفت درست هستند.

True Label: Open Eyes
Predicted Class: Open Eyes



True Label: Closed Eyes
Predicted Class: Closed Eyes



True Label: Closed Eyes
Predicted Class: Closed Eyes



True Label: Open Eyes
Predicted Class: Open Eyes



تصویر 4-4. پیش‌بینی تصاویر از مجموعه تست

به دلیل کنجکاوی سعی کردم چند تصویر که تا حدی نزدیک به دیتاست ما هستند را انتخاب کنم و عملکرد مدل را بار دیگر تست کنم که نتیجه نسبتاً قابل قبول بود و از ۵ تصویر ۴ مورد درست تشخیص داده شدند که مشاهده می‌کنید:

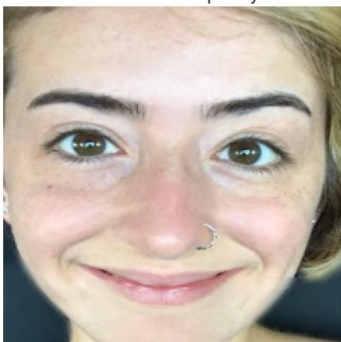
Predicted Class: Closed Eyes



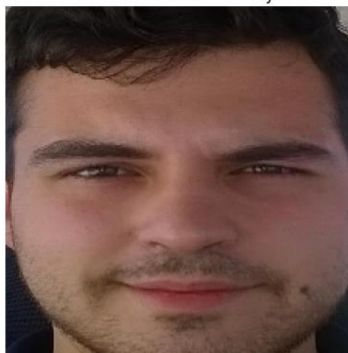
Predicted Class: Open Eyes



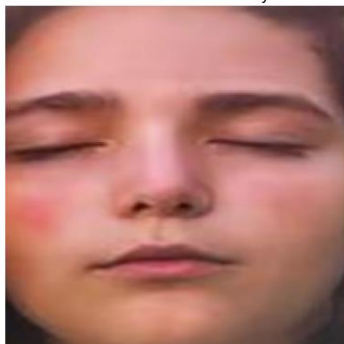
Predicted Class: Open Eyes



Predicted Class: Closed Eyes



Predicted Class: Closed Eyes



تصویر 4-5. پیش‌بینی تصاویر خارج از دیتاست

خلاصه و نتیجه گیری

۵-۱. خلاصه و نتیجه گیری

در این پروژه، یک مدل یادگیری عمیق برای تشخیص باز یا بسته بودن چشم با استفاده از شبکه عصبی مبتنی بر معماری MobileNet پیاده سازی شد. مدل با استفاده از یک مجموعه داده آموزشی حاوی تصاویر چشم باز و بسته آموزش داده شد. پیش پردازش داده ها شامل بارگیری تصاویر، تغییر اندازه آنها به ابعاد 224×224 پیکسل و نرمالیزه کردن آنها بود.

پس از آموزش مدل با داده های آموزشی و اعتبارسنجی، نتایج بسیار رضایت بخشی به دست آمد. گزارش طبقه بندی برای داده های آموزشی و اعتبارسنجی، دقت بالایی را نشان داد که نشان دهنده عملکرد خوب مدل در تشخیص چشم باز و بسته است.

برای ارزیابی نهایی مدل، دو روش مختلف بر روی داده های آزمون اعمال شد. در روش اول، یک نمونه تصادفی از داده های آزمون انتخاب شد و پیش بینی مدل برای آن نمایش داده شد. در روش دوم، یک شاخص خاص از داده های آزمون انتخاب شد، پیش بینی مدل برای آن محاسبه شد و نتیجه به همراه برجسب واقعی نمایش داده شد. هر دو روش، نتایج قابل قبولی را نشان داد که تأییدی بر عملکرد خوب مدل است.

علاوه بر این، دقت مدل بر روی داده های آزمون محاسبه شد و مقدار بالایی را نشان داد که بیانگر توانایی مدل در تعمیم پذیری و عملکرد خوب بر روی داده های جدید است.

علاوه بر دقت بالا بر روی داده های آزمون، نتایج نشان می دهد که مدل دچار پدیده اور فیتینگ (برازش بیش از حد) نیز نشده است. با مقایسه نمودارهای از دست رفتن و دقت در

مراحل آموزش و اعتبارسنجی، مشاهده می‌شود که اختلاف بین این نمودارها در دوره‌های آخر آموزش زیاد نیست. این نشان می‌دهد که مدل توانسته الگوهای کلی داده‌ها را به خوبی یاد بگیرد و روی داده‌های جدید نیز عملکرد خوبی دارد.

عدم اوریفیتینگ یکی از نکات مهم در یادگیری عمیق است، زیرا مدل‌های اورفیت شده معمولاً روی داده‌های آزمون عملکرد ضعیفی دارند و قابلیت تعمیم‌پذیری کمی خواهند داشت. بنابراین، می‌توان گفت که مدل پیاده‌سازی شده، علاوه بر دقت بالا، از قابلیت تعمیم‌پذیری خوبی نیز برخوردار است.

در نهایت، نمودارهایی برای نمایش روند از دست رفتن در طول دوره‌های آموزش و دقت آموزش و اعتبارسنجی ارائه شد که به درک بهتر روند یادگیری مدل کمک می‌کند.

با توجه به نتایج به دست آمده، می‌توان نتیجه گرفت که مدل پیاده‌سازی شده، عملکرد رضایت بخشی در تشخیص خواب آلودگی داشته است و می‌تواند در کاربردهای مختلف از جمله سیستم‌های هشدار دهنده خواب آلودگی رانندگان، مورد استفاده قرار گیرد.

به عنوان فعالیتی در آینده می‌توان در نظر داشت که سیستمی طراحی شود که تصویر چشم استخراج شده توسط سیستم با مجموعه داده تطبیق داده می‌شود. با کمک مجموعه داده، سیستم تشخیص می‌دهد که اگر چشم‌ها برای یک محدوده خاص بسته بودند، می‌تواند یک هشدار برای هشدار به راننده صدا کند و اگر چشم‌ها بعد از هشدار باز بودند، می‌تواند به ردیابی ادامه دهد. اگر چشم‌ها باز بودند، امتیازی که تنظیم شده بود کاهش یابد و اگر چشم‌ها بسته بودند، امتیاز افزایش یابد.

6. فهرست منابع

- [1] Mehta, P., Sinha, A., & Singh, P. (Year of publication). Driver Drowsiness Detection System. Department of Electrical and Computer Engineering, New York University.
- [2] Gwak, J., Hirao, A., Shino, M. (2020). An investigation of early detection of driver drowsiness using ensemble machine learning based on hybrid sensing.
- [3] Kepesiova, Z., Ciganek, J., & Kozak, S. (2020). Driver drowsiness detection using convolutional neural networks. In 2020 Cybernetics & Informatics (K&I).
- [4] You, F., Li, X., Gong, Y., Wang, H., & Li, H. (2019). A real-time driving drowsiness detection algorithm with individual differences consideration. IEEE Access, 7, 179396-179408.
- [5] Mehta, S., Dadhich, S., Gumber, S., & Bhatt, A. J. (2019). Real-time driver drowsiness detection system using eye aspect ratio and eye closure ratio. SSRN Electronic Journal
- [6] Sathasivam, S., Mahamad, A. K., Saon, S., Sidek, A., Som, M. M., & Ameen, H. A. (2020). Drowsiness detection system using eye aspect ratio technique. In 2020 IEEE Student Conference on Research and Development (SCORED).
- [7] Singh, J., Kanojia, R., Singh, R., Bansal, R., Bansal, S. (Year of publication). Driver Drowsiness Detection System – An Approach By Machine Learning Application. Computer Science and Engineering Department, Meerut Institute of Engineering and Technology, Meerut, India.
- [8] <https://ieee-dataport.org/open-access/yawdd-yawning-detection-dataset>

[9] <https://medium.com/@godeep48/an-overview-on-mobilenet-an-efficient-mobile-vision-cnn-f301141db94d>

[10] <https://medium.com/@nocodingai/mobilenet-fc34af9f58a5>

[11] <https://towardsdatascience.com/the-mostly-complete-chart-of-neural-networks-explained-3fb6f2367464>

پیوست الف: نمونه یک پیوست

این‌جانب رضا تهرانی دانش‌آموخته کارشناسی‌ارشد رشته زبان و ادبیات فارسی دانشکده ادبیات دانشگاه ایران پدیدآور پایان‌نامه با عنوان راهنمای نگارش پایان‌نامه و رساله با راهنمایی دکتر امیر شریف گواهی و تعهد می‌کنم که بر پایه قوانین و مقررات، از جمله «دستورالعمل نحوه بررسی تخرلفات پژوهشی» و همچنین «مصادیق تخرلفات پژوهشی» مصوب وزارت علوم، تحقیقات و فناوری (۲۵ اسفند ۱۳۹۳):

- این پایان‌نامه دستاورد پژوهش این‌جانب و محتوای آن از درستی و اصالت برخوردار است؛
- حقوق معنوی همه کسانی را که در به‌دست‌آمدن نتایج اصلی پایان‌نامه تأثیرگذار بوده‌اند، رعایت کرده‌ام و هنگام کاربرد دستاورد پژوهش‌های دیگران در آن، با دقت و به‌درستی به آن‌ها استناد کرده‌ام؛
- این پایان‌نامه و محتوای آن را تاکنون این‌جانب یا کس دیگری برای دریافت هیچ‌گونه مدرک یا امتیازی در هیچ‌جا ارائه نکرده‌ام؛
- همه حقوق مادی این پایان‌نامه از آن دانشگاه ایران است و آثار برگرفته از آن با وابستگی سازمانی دانشگاه ایران منتشر خواهد شد؛
- در همه آثار برگرفته از این پایان‌نامه، نام استاد(ان) راهنما و اگر استاد راهنمای نخست تشخیص دهد، نام استاد(ان) مشاور و نشانی رایانامه سازمانی آنان را می‌آورم؛
- در همه گام‌های انجام این پایان‌نامه، هرگاه به اطلاعات شخصی افراد یا اطلاعات سازمان‌ها دسترسی داشته یا آن‌ها را به‌کار برده‌ام، رازداری و اخلاق پژوهش را رعایت کرده‌ام.

واژه‌نامه توصیفی

پایان‌نامه: واژه پایان‌نامه ریشه‌ای لاتین دارد که به معنای «مسیر» است.

واژه‌نامه فارسی به انگلیسی

پایان‌نامه: Dissertation

رساله: Thesis

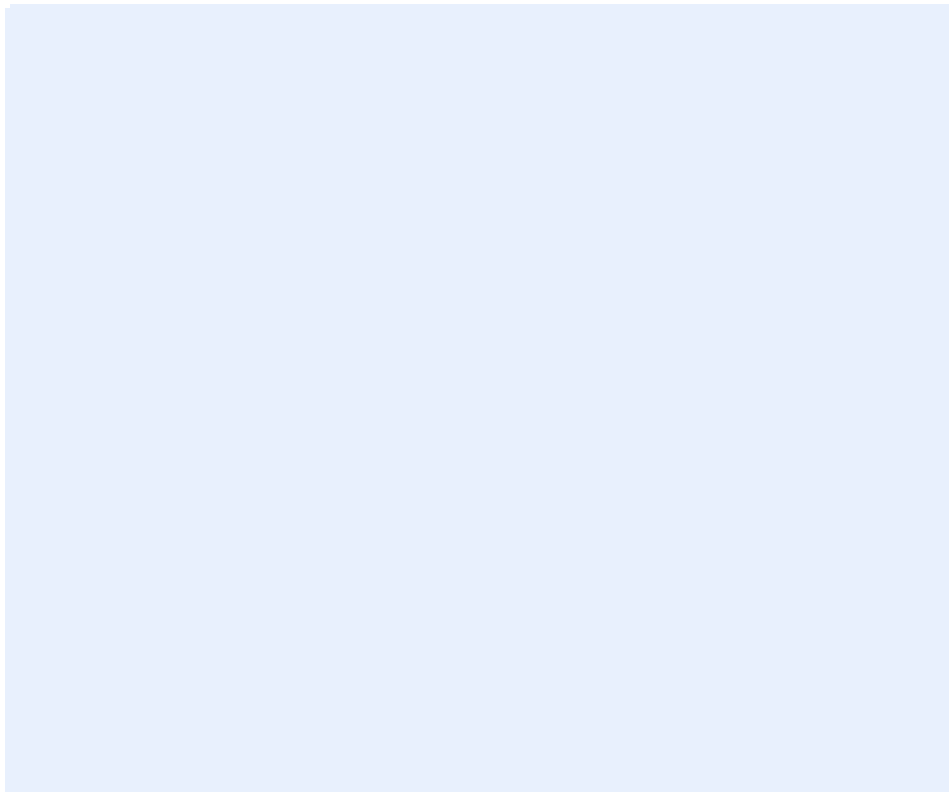
واژه‌نامه انگلیسی به فارسی

Dissertation: پایان‌نامه

Thesis: رساله

فهرست مقاله‌های برگرفته از پایان‌نامه

برگ تأیید هیئت داوران / صورت جلسه دفاع (به زبان انگلیسی)



Abstract: Insert abstract here.

Research Aim: Early Detection of Driver Sleepiness Using Convolutional Neural Networks (CNNs) for Road Safety Enhancement.

Research method: This study utilizes the Closed Eye in the Wild (CEW) dataset to train a convolutional neural network (CNN) model for analyzing the facial and eye regions of drivers.

Findings: The results demonstrate that the proposed model utilizing CNN can accurately detect driver sleepiness with a precision of 99.15%.

Conclusion: Driver sleepiness is a significant factor in road accidents worldwide, often resulting from fatigue and lack of sleep.

Based on the obtained results, the use of the proposed method can serve as an early warning system to prevent road accidents caused by driver sleepiness. This approach could be implemented in vehicles and transportation fleets to prevent road accidents resulting from driver drowsiness.

Keywords: Driver drowsiness, Early detection, Convolutional neural networks, Image processing, Road safety



دانشگاه صنعتی نوشیروانی بابل

**Babol Noshirvani
University of Technology**

Institution/University of...

Faculty of...

A Dissertation Submitted in Partial Fulfillment for the Bachelor of
Science Degree

Detecting drowsiness with neural networks

By

Damirchi

Supervisor

Dr.Farideh Ebrahimi

Advisor
Insert the name(s) of advisor(s) here.

(May 2024)