

دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده مهندسی برق و کامپیوتر
گروه الکترونیک



کنترل ماوس کامپیوتر با تشخیص حرکت دست در تصویر

پایان نامه برای دریافت درجه کارشناسی ارشد در رشته مهندسی برق
گرایش مدارات مجتمع

یلدا فروتن

اساتید راهنما

دکتر احمد کلهر و دکتر صمد شیخائی

شهریور ۱۳۹۹

تعهدنامه اصالت اثر

باسم‌هی تعالیٰ

اینجانب یلدا فروتن تائید می‌کنم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب است و به دستاوردهای پژوهشی دیگران که در این نوشته از آنها استفاده شده است مطابق مقررات ارجاع گردیده است. این پایان‌نامه قبل‌اً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نشده است.

نام و نام خانوادگی دانشجو: یلدا فروتن
تاریخ و امضای دانشجو:

کلیه حقوق مادی و معنوی این اثر
متعلق به دانشگاه تهران می باشد.

تقدیم به:

پدر و مادر مهربانم

و

برادر دلسوژم

قدردانی

منت خدای را عز و جل که طاعتش موجب قریبی است و به شکر اندرش مزید نعمت.
در آغاز از دو استاد گرانقدر و مهربان خود، جناب آقای دکتر احمد کلهر و جناب آقای دکتر صمد شیخائی،
صمیمانه قدردانی می‌کنم که بدون وجود حمایت‌ها و راهنمایی‌های ایشان این پایان‌نامه به انجام نمی‌رسید.
همچنین از همکری‌های تمامی دوستان ارجمند در آزمایشگاه مدارات و سیستم‌های پیشرفته مخابرات داده
که مرا یاری دادند کمال امتنان را دارم.

در پایان، بوسه می‌زنم بر دستان خداوندگاران مهر و مهربانی، پدر و مادر عزیز و برادر مهربانم که در چندین
ماه گذشته و با وجود شرایط سخت زندگی همواره پشتیبان من بوده‌اند.

يلدا فروتن

شهریور ۱۳۹۹

چکیده

از دیرباز بشر آرزو بلند کردن اجسام را تنها با استفاده از اشارات دست خود داشته است. اگرچه تا به امروز نتوانسته اشیا را بدون تماس مستقیم جابه‌جا کند؛ اما هوش مصنوعی این امکان را به او داده که با استفاده از اشارات دست خود جلوی یک دوربین و بدون تماس با شی خارجی، سیستم‌های هوشمند را کنترل کند.

در این پژوهش یک واسط کاربری در راستای برقراری ارتباط انسان و رایانه طراحی می‌شود تا نشان‌گر رایانه با استفاده از اشارات دست کاربر و بدون تماس کنترل گردد. تصاویر مربوط به اشارات دست انسان از طریق دوربین رایانه و با استفاده از ابزارهای پردازش تصویر، کتابخانه‌ها و توابع موجود آن به صورت فریم‌های متوالی دریافت می‌شوند و مورد پردازش قرار می‌گیرند. سپس با استفاده از الگوریتم‌های مبتنی بر شبکه‌های عصبی ناحیه دست تشخیص داده می‌شود.

از طرفی در راستای کنترل نشان‌گر رایانه توسط دست کاربر، یک مجموعه دادگان متشکل شده از ۶۷۲۰ نمونه تصویری برای ۴ کلاس مشت، کف دست، اشاره به چپ و اشاره به راست جمع‌آوری گردید. این تصاویر توسط ۱۵ کاربر و با پس‌زمینه‌های ساده و شرایط نوری مختلف تهیه شده است. با استفاده از مجموعه دادگان جمع‌آوری شده، یک شبکه عصبی کانولوشنی بر پایه شبکه EfficientNet-B0 و لایه‌های تماماً متصل آموزش می‌بیند. شبکه آموزش‌دیده به دو صورت ذخیره می‌شود تا اولاً تصاویر خروجی از تشخیص دهنده دست را طبقه‌بندی کند و دوماً در صورت مشابهت با تصاویر موجود در مجموعه دادگان به فرمان مدنظر در جهت کنترل ماوس تبدیل شود.

در نهایت بر اساس برچسب نگاشته شده توسط شبکه طبقه‌بند، یکی از فرامین روشن یا خاموش شدن ماوس، جابه‌جایی نشان‌گر، کلیک کردن و راست‌کلیک کردن اجرا می‌شود و الگوریتم فریم بعدی را در راستای تکرار فرآیند، دریافت می‌کند. کنترل کننده ماوس طراحی شده در این پژوهش به دقت ۹۲.۶ درصد رسیده و در پس‌زمینه‌های مختلف قابل استفاده است. همچنین در راستای طراحی ماوس ارائه شده از زبان برنامه‌نویسی پایتون بهره برده شده است.

واژگان کلیدی: تشخیص اشارات دست، مجموعه دادگان، شبکه‌های عصبی کانولوشن، طبقه‌بندی، ماوس رایانه و تشخیص اشیا

فهرست مطالب

ت	فهرست تصاویر
ج	فهرست جداول
ج	فهرست الگوریتم‌ها
۱	فصل ۱: مقدمه
۱	۱-۱ بیان مسئله
۳	۲-۱ اهداف پژوهش
۳	۳-۱ محدودیت‌های پژوهش
۴	۴-۱ روش انجام پژوهش
۵	۵-۱ خلاصه فصل‌ها
۶	فصل ۲: مروری بر مطالعات انجام شده
۶	۱-۲ مقدمه
۶	۲-۲ مروری بر سیر تکامل ماوس
۸	۳-۲ تشخیص حرکت دست
۱۵	۴-۲ نتیجه‌گیری
۱۶	فصل ۳: مفاهیم و مقدمات پژوهش
۱۶	۱-۳ مقدمه
۱۶	۲-۳ یادگیری عمیق

۱۷	شبکه عصبی مصنوعی	۳-۳
۱۹	معماری لایه‌ها	۱-۳-۳
۲۱	آموزش شبکه عصبی	۲-۳-۳
۲۲	مجموعه دادگان	۴-۳
۲۳	یادگیری انتقالی	۵-۳
۲۴	تشخیص اشیا	۶-۳
۲۵	۱-۶-۳ تشخیص اشیا با استفاده از شبکه‌های عصبی کانولوشن	
۲۶	نتیجه‌گیری	۷-۳
۲۷	فصل ۴: الگوریتم پیاده‌سازی شده	
۲۷	۱-۴ مقدمه	
۲۷	۲-۴ مجموعه دادگان	
۲۹	۳-۴ تشخیص دست	
۳۱	۱-۳-۴ عملکرد الگوریتم تشخیص دهنده دست	
۳۴	۴-۴ طبقه‌بندی	
۳۵	۱-۴-۴ آماده‌سازی مجموعه دادگان	
۳۵	۲-۴-۴ آموزش شبکه عصبی کانولوشنی به منظور طبقه‌بندی	
۴۰	۳-۴-۴ طراحی شبکه کانولوشن به منظور شباهت‌سنگی	
۴۶	۵-۴ کنترل نشان‌گر با اشارات پیش‌بینی شده	
۴۶	۱-۵-۴ حالت کف دست: شروع و جابه‌جایی نشان‌گر	
۴۸	۲-۵-۴ حالت اشاره به چپ: کلیک کردن	
۴۹	۳-۵-۴ حالت اشاره به راست: راست‌کلیک کردن	
۵۰	۴-۵-۴ حالت مشت: خاموش کردن	
۵۴	۶-۴ نتیجه‌گیری	
۵۵	فصل ۵: ارزیابی الگوریتم ارائه شده و نتایج	
۵۵	۱-۵ مقدمه	

۵۵	۲-۵ تاخیر ماووس طراحی شده
۵۷	۳-۵ ارزیابی طبقه‌بند طراحی شده
۵۹	۴-۵ ارزیابی ماووس طراحی شده
۶۴	۵-۵ نتیجه‌گیری
۶۵	فصل ۶: بحث و نتیجه‌گیری
۶۵	۱-۶ مقدمه
۶۵	۲-۶ جمع‌بندی
۶۷	۳-۶ نوآوری
۶۷	۴-۶ محدودیت‌ها
۶۸	۵-۶ کارهای آینده
۷۰	مراجع

فهرست تصاویر

۱۸	نمونه‌ای از یک نورون در شبکه‌های عصبی	۱-۳
۱۹	نمونه‌ای از ترکیب لایه‌ها در شبکه‌های عصبی	۲-۳
۲۰	یک شبکه عصبی کانولوشنی و نحوه اتصالات آن	۳-۳
۲۱	نمونه‌ای از یک شبکه عصبی کانولوشنی با آرایش سه‌بعدی	۴-۲
۲۴	تفاوت تشخیص اشیا و طبقه‌بندی تصاویر	۵-۳
۲۹	نمونه‌ای از تصاویر اشارات دست تعریف شده در پژوهش برای آموزش شبکه	۱-۴
۳۰	نمونه‌ای از تصاویر اشارات دست تعریف شده در پژوهش برای اعتبارسنجی شبکه	۲-۴
۳۲	تشخیص دست و قرار دادن قادر در اطراف آن توسط تشخیص‌دهنده SSD	۳-۴
۳۳	فریم بریده شده از ناحیه کادر محدود توسط الگوریتم تشخیص‌دهنده دست	۴-۴
۳۷	نقشه حرارتی نمایان‌گر تأثیر ترکیب ابعاد مختلف بر برجسته‌سازی ویژگی‌های یک تصویر	۵-۴
۳۸	لایه‌های اضافه شده به شبکه EfficientNet	۶-۴
۴۱	خلاصه نحوه عملکرد شبکه عصبی Siamese	۷-۴
۴۳	انواع تصاویر ممکن جهت اعمال به شبکه و ابزار تفکیک حالت‌های ناخواسته از حالت‌های مدنظر	۸-۴
۴۷	روشن کردن ماوس طراحی شده و جایه‌جایی نشان‌گر با استفاده از کف دست	۹-۴
۴۹	نحوه کلیک کردن توسط ماوس طراحی شده	۱۰-۴
۵۰	نحوه راست‌کلیک کردن توسط ماوس طراحی شده	۱۱-۴
۵۱	خاموش کردن ماوس طراحی شده با استفاده از دست کاربر در حالت مشت	۱۲-۴
۵۶	نمودار دقیق برای دادگان آموزش و اعتبارسنجی در ۲۰ اپیاک	۱-۵

۲-۵ نمودار خطاب‌ای دادگان آموزش و اعتبارسنجی در ۲۰ ایپاک	۵۷
۳-۵ ماتریس درهم‌ریختگی مربوط به بخش طبقه‌بندی با استفاده از دادگان آزمایش	۵۸
۴-۵ پس‌زمینه‌های منتخب جهت ارزیابی کنترل‌کننده ماوس طراحی شده	۶۰
۵-۵ ماتریس درهم‌ریختگی برای کنترل‌کننده ماوس طراحی شده	۶۳

فهرست جداول

۱-۴	فاصله بین تصاویر دادگان اعتبارسنجی و بردار مرجع معرف هر دسته	۴۴
۱-۵	مقایسه دو شبکه طبقه‌بند بر پایه EffiecientNet-B0 و VGG16	۵۹
۲-۵	نتایج ارزیابی کنترل‌کننده ماوس طراحی شده در پس زمینه سفید	۶۱
۳-۵	نتایج ارزیابی کنترل‌کننده ماوس طراحی شده در پس زمینه ساده	۶۱
۴-۵	نتایج ارزیابی کنترل‌کننده ماوس طراحی شده در پس زمینه شلوغ	۶۲
۵-۵	مقادیر دقت برای سه پس زمینه مختلف به تفکیک هر حالت از کنترل ماوس طراحی شده	۶۳

فهرست الگوریتم‌ها

۱-۴	تشخیص دهنده دست استفاده شده به همراه ورودی و خروجی‌های آن	۳۳
۲-۴	طبقه‌بند طراحی شده با استفاده از شبکه‌های عصبی کانولوشنی	۴۰
۳-۴	تفکیک حالت‌های ناخواسته با استفاده از یک شبکه عصبی مبتنی بر شباهت‌سنجدی	۴۵
۴-۴	بخش کنترل‌کننده و تشخیص دهنده فرامین به ماوس طراحی شده	۵۲
۵-۴	نمای کلی ماوس طراحی شده به همراه ورودی و خروجی بخش‌های مختلف آن	۵۲
۶-۴	نحوه عملکرد ماوس طراحی شده به همراه ورودی و خروجی آن	۵۳

فصل ۱

مقدمه

در این فصل تمرکز بر بیان موضوع تحقیق و ارائه آن به عنوان یک واسط در راستای تعامل مابین انسان و رایانه است. پیاده‌سازی یک روش جایگزین برای ماوس و کنترل نشان‌گر رایانه با استفاده از حوزه یادگیری ماشین خصوصاً شبکه‌های عصبی موضوع مورد بررسی در این فصل و فصل‌های آتی است که در ادامه به آن پرداخته می‌شود.

۱-۱ بیان مسئله

امروزه رایانه‌ها نقش اساسی در زندگی افراد داشته و به عموم زوایا زندگی شخصی و اجتماعی آن‌ها نفوذ کرده‌اند. از طرفی به دلیل تولید انبوه و افزایش دسترسی به رایانه‌های شخصی، همواره تأثیر آن‌ها در زندگی روزمره افراد بیش‌تر می‌شود. هدف از علم تعامل انسان با رایانه^۱ نیز کنترل، بررسی و بهبود ارتباط بین بشر و ماشین بوده که کاربردهای آن را می‌توان به سه دسته تقسیم‌بندی کرد:

- کنترل ماشین
- تشخیص زبان اشاره
- سیستم‌های بازی مبتنی بر پردازش تصویر

¹Human-computer Interaction (HCI)

دسته کنترل ماشین با کمک بینایی ماشین^۲ دارای کاربردهای همچون کنترل ربات‌ها، دستگاه‌های صوتی و تصویری و ... است. دسته دوم با تشخیص زبان اشاره، سعی بر بهبود ارتباطات افراد ناتوان در صحبت کردن و شنیدن دارد. هدف از سیستم‌های بازی مبتنی بر پردازش تصویر نیز حذف کنسول‌های متداول که به صورت سخت‌افزاری هستند، است [۱] و [۲].

در اکثر کاربردهای گفته شده، دست انسان نقش اساسی در برقراری ارتباط بین انسان و رایانه را داشته و عمدۀ راه ارتباطی از طریق همین دست و انگشتان دست او است؛ بنابراین در راستای برقراری یک تعامل طبیعی استفاده از دست و تبدیل اشارات آن به فرامین معنی‌دار برای رایانه یا ماشین، مناسب است. اگرچه تعامل انسان و رایانه در برخی کاربردها از ارتباط متنی به گرافیکی تکمیل یافته است، اما واسطه‌هایی همچون صفحه کلید و ماوس^۳ همچنان متداول هستند. به همین علت در سال‌های اخیر ارائه روش‌های جایگزین در راستای کنترل رایانه توسط انسان، از موضوعات مورد توجه علاقمندان به حوزه یادگیری ماشین^۴ بوده است [۳] و [۲].

تشخیص حرکت دست^۵ با استفاده از کف دست، مکان انگشتان یا شکل دست این امکان را به کاربر می‌دهد که بدون استفاده از سخت‌افزارهایی چون ماوس، رایانه شخصی خود را بدون تماس مستقیم با یک سخت‌افزار واسطه، کنترل کنند. در این پژوهش نیز سعی شده تا با استفاده از تشخیص حرکت دست کاربر و تبدیل آن به فرمان مدنظر، نشان‌گر ماوس بدون سخت‌افزار واسطه و تنها یک دوربین کنترل گردد [۴].

اگرچه انسان‌ها در درک حالت‌های مختلف و حتی پیچیده دست خوب عمل می‌کنند، اما این کار برای ماشین سخت است؛ زیرا با توجه به ساختار دست انسان و مفصل‌های آن، تعداد حالت‌هایی که دست می‌تواند داشته باشد زیاد است و ساخت یک نگاشت که این حالت را به فرامین مدنظر تبدیل کند، به شدت غیرخطی است. علاوه بر درجه آزادی دست، پس‌زمینه‌های مختلف و تغییرات در نور ساخت این نگاشت را سخت‌تر هم می‌کند. در نتیجه این عمل برای ابزارهای سنتی تر بینایی ماشین که نیاز به استخراج ویژگی دارند، دشوار است [۵].

با معرفی شبکه‌های عصبی^۶ و نحوه یادگیری عمیق^۷ آن‌ها، پیاده‌سازی مسائل با درجه غیرخطینگی بالا که با استفاده از روش‌های سنتی یادگیری ماشین دشوار بود، امکان‌پذیر شده است. اگرچه شبکه‌های عصبی در حوزه‌های مختلف کاربرد عمده دارند اما می‌توان آن‌ها را یک طبقه‌بند^۸ دانست که کل تصویر را به یک برچسب

²Machine Vision

³Mouse

⁴Machine Learning

⁵Hand Gesture Recognition

⁶Neural Network (NN)

⁷Deep Learning

⁸Classifier

نگاشته می‌کنند. بخش عمده این پژوهش، حل یک مسئله طبقه‌بندی است تا بتوان حالت‌های مختلف دست را طبقه‌بندی نمود و با استفاده از آن‌ها، نشان‌گر رایانه را به فرمان گرفت تا بتوان عملیاتی همچون روشن یا خاموش کردن ماوس، کلیک کردن^۹، راست‌کلیک کردن^{۱۰} و جابه‌جایی نشان‌گر ماوس را بدون هیچ سخت‌افزار خاص، تنها با یک دوربین و در چارچوب یادگیری عمیق پیاده‌سازی کرد [۶].

۲-۱ اهداف پژوهش

در این پژوهش یک مجموعه دادگان^{۱۱} متناسب با حالت‌های مختلف دست توسط نویسنده این پایان‌نامه، جمع‌آوری می‌شود تا بتوان با استفاده از آن‌ها یک شبکه عصبی کانولوشنی^{۱۲} را آموزش داد. پس از پیاده‌سازی یک شبکه برای طبقه‌بندی تصاویر موجود در مجموعه دادگان، شبکه آموزش دیده ذخیره می‌شود تا هنگام اجرای برنامه به تصاویر ورودی، یک برچسب اختصاص داده شود. البته ممکن است تصاویر ورودی به شبکه، مربوط به دسته‌های تعریف شده نباشند؛ بنابراین لازم است تصاویر مربوطه به حالت‌های ناخواسته نیز از حالت‌های مدنظر و تعریف شده در مجموعه دادگان جدا شوند.

پس از جداسازی تصاویر حالت‌های ناخواسته و تعیین برچسب تصاویر مربوط به حالت‌های مدنظر، در جهت کنترل نشان‌گر رایانه، یک سری فرامین تعریف می‌شوند تا عمل روشن یا خاموش کردن کنترل کننده ماوس طراحی شده، کلیک کردن، راست‌کلیک کردن یا جابه‌جایی نشان‌گر انجام شود. لازم به ذکر است هنگام اجرای برنامه، تصاویر مورد نیاز شبکه عصبی کانولوشن با استفاده از الگوریتم تشخیص دست استخراج می‌شوند و به عنوان ورودی به شبکه عصبی از پیش آموزش دیده، اعمال می‌گردند.

۳-۱ محدودیت‌های پژوهش

از چالش‌های پیش‌رو در این پژوهش می‌توان به جمع‌آوری مجموعه دادگان متناسب با کاربرد و ایجاد قدرت تعمیم به دادگان جدید، اشاره کرد. پردازش سنگین محاسبات در راستای آموزش شبکه‌های عصبی به دلیل بالا

⁹Clicking

¹⁰Right-click

¹¹Dataset

¹²Convolutional Neural Network (CNN)

بودن تعداد پارامترها و لایه‌ها که منجر به ضرب ماتریسی شده، از چالش‌های دیگر این پژوهش است. همچنین در هنگام اجرای برنامه، در راستای تعامل بهتر کاربر و رایانه، به پردازنده گرافیکی^{۱۳} نیاز است که البته بدون آن نیز، امکان کنترل ماوس طراحی شده وجود دارد.

۴-۱ روشنگاری پژوهش

نحوه عملکرد کلی کنترل کننده ماوس طراحی شده بدین صورت است که فریم‌های متوالی توسط دوربین دریافت شده و به یک الگوریتم تشخیص دهنده دست وارد می‌شوند. در صورت وجود دست در فریم دریافت شده، ناحیه دست از تصویر اصلی جدا می‌شود و فریم بریده شده به شبکه طبقه‌بند و شبکه شباهت‌سنج اعمال می‌گردد.

از طرفی لازم است حالت‌هایی برای دست انسان تعریف گردد تا کاربر با نشان دادن این حالات در جلو دوربین نشان‌گر رایانه را کنترل کند؛ بنابراین مجموعه دادگانی مشکل از این حالت‌ها تهیه می‌شود تا یک شبکه عصبی با استفاده از این دادگان آموزش بینند. شبکه عصبی استفاده شده در این پژوهش بر پایه شبکه EfficientNet-B0 و چندین لایه تماماً متصل است. سپس شبکه آموزش دیده شده ذخیره می‌گردد تا در جهت طبقه‌بندی تصاویر خروجی مربوط به مرحله تشخیص دهنده دست، مورد استفاده قرار بگیرد.

همچنین از شبکه عصبی ذخیره شده در جهت مقایسه شباهت بین تصاویر موجود در مجموعه دادگان جمع آوری شده و فریم‌های بریده شده استفاده می‌شود؛ به گونه‌ای که با استفاده از نمونه‌های موجود در هر دسته از مجموعه دادگان، یک بردار مرجع آن دسته، ساخته و با تصاویر خروجی تشخیص دهنده مقایسه می‌شوند. در واقع هدف از طراحی شبکه شباهت‌سنج حذف آن دسته از تصاویر مورد تأیید تشخیص دهنده دست است که شامل حالت‌های ناخواسته هستند.

در صورت وجود میزان شباهت لازم مابین خروجی تشخیص دهنده و نمونه‌های موجود در مجموعه دادگان، برچسب پیش‌بینی شده توسط طبقه‌بند، دارای اعتبار است و به بخش کنترل کننده و تخصیص دهنده وظایف اعمال می‌شود. در نهایت بسته به برچسب خروجی و مختصات نقطه مرکزی فریم بریده شده، اعمالی همچون روشن کردن ماوس، کلیک یا راست‌کلیک کردن، جابه‌جایی نشان‌گر و خاموش کردن ماوس انجام خواهد شد.

¹³Graphics Processing Unit (GPU)

۱-۵ خلاصه فصل‌ها

در فصل اول که مقدمه‌ای بر پایان‌نامه است، مسئله مورد پژوهش، اهداف و محدودیت‌های آن و درنهایت خلاصه‌ای از روش ارائه‌شده در این پژوهش عنوان گردید. در فصل دوم تحت عنوان مروری بر مطالعات انجام‌شده، پیشینه استفاده از دست در راستای کنترل نشان‌گر رایانه مرور می‌شود؛ به گونه‌ای که سیر تغییرات ماوس و کارهایی که در راستای بهبود و حتی حذف آن صورت گرفته، بیان می‌شود. همچنین در این فصل کارهای انجام‌شده در راستای تشخیص حرکت دست و کنترل نشان‌گر رایانه یا سایر ماشین‌ها از طریق آن بررسی می‌شود.

در فصل سوم مفاهیم اساسی در حوزه یادگیری عمیق بررسی و روش‌های مورد استفاده در راستای توضیح عملکرد شبکه‌های عصبی خصوصاً شبکه‌های عصبی کانولوشن عنوان می‌گردد. همچنین چگونگی تغییر شبکه‌های کانولوشن از یک طبقه‌بند به تشخیص دهنده اشیا نیز بررسی می‌شود.

در فصل چهارم روش مدنظر در این پژوهش پیاده‌سازی می‌شود تا با حرکت دست در جلو یک دوربین، نشان‌گر رایانه به فرمان درآید. سپس در فصل پنجم نتایج به دست آمده در جهت ارزیابی کنترل کننده نشان‌گر رایانه بررسی می‌شوند. در فصل ششم نیز به نتیجه‌گیری و پیشنهادهای آتی در راستای بهبود کنترل کننده ارائه شده، پرداخته می‌شود.

فصل ۲

مروری بر مطالعات انجام شده

۱-۲ مقدمه

در این پژوهش، تمرکز بر حل مسئله تشخیص حرکت دست و طبقه‌بندی حالت‌های مختلف آن در راستای کنترل نشان‌گر ماوس با استفاده از الگوریتم‌های یادگیری عمیق است. در ابتدای فصل پیش‌رو سیر تکامل ماوس و اقدامات انجام شده در راستای بهبود آن بررسی می‌شوند. سپس روش‌های تشخیص‌دهنده دست در راستای برقراری تعامل کاربر با سیستم‌های هوشمند و نحوه پیاده‌سازی این تعامل ارائه می‌گردد. در نهایت نیز محدودیت‌ها و نقاط قوت روش‌های مختلف که مبتنی بر روش‌های کلاسیک بینایی ماشین یا یادگیری عمیق هستند، عنوان می‌شود تا در پیشبرد این پژوهش مورد استفاده قرار گیرند.

۲-۲ مروری بر سیر تکامل ماوس

بیش از نیم قرن از ساخت اولین ماوس توسط داگлас اینگلبارت^۱ گذشته و تا امروز پیشرفت‌های زیادی در راستای بهبود عملکرد این سخت افزار شده است. از پیشرفت‌هایی که در طراحی ماوس انجام شده است، می‌توان به نحوه ثبت حرکات ماوس (mekanikی، نوری یا لیزری)، نحوه اتصالات و ابعاد حرکات ماوس (دوبعدی یا سه‌بعدی) اشاره کرد [۴].

¹Douglas Engelbart

فصل ۲. مروری بر مطالعات انجام شده

علاوه‌گم پیشرفت چشمگیر در راستای بهبود ماوس، این سخت افزار همچنان مبتنی بر تماس بوده و لازم است کاربر با آن تماس مستقیم داشته باشد که ماوس را محدود به کنترل از فاصله نزدیک می‌کند. همچنین کنترل نشانگر رایانه نیاز به یک سخت افزار واسط همچون ماوس دارد. حال که با شیوع ویروسی همچون کوید ۱۹ در پایان سال ۲۰۱۹ میلادی، می‌توان پایان عمر ابزارهای کنترل ماشین، مبتنی بر تماس انسان را پیش‌بینی کرد.

علاوه بر تلاش‌های گفته شده در راستای بهبود ماوس، تلاش‌هایی برای حذف آن نیز شده است که اکثراً مبتنی بر روش‌های کلاسیک یادگیری ماشین بوده‌اند. استفاده از حسگر کینکت^۲ و یا ماوس ذهنی که براساس سیگنال‌های امواج مغزی^۳ کار می‌کند، از جمله تلاش‌هایی برای کنترل نشانگر بدون استفاده از ماوس بوده است [۸] و [۹].

در مقاله [۸] حسگر کینکت به گونه‌ای تعییه شده تا با استفاده از یک دوربین مادون قرمز^۴ فاصله اشیا و حسگر مربوطه محاسبه گردد. لازم به ذکر است در این مقاله تنها از داده‌های مربوط به فاصله استفاده شده است، که دستگاه را در مقابل نور مقاوم کرده و حتی در شرایط نوری کم و حتی تاریکی قابل استفاده خواهد بود. حسگر کینکت را در زیر مانتیور قرار می‌دهند و فرد در محدوده دید آن، علامت از پیش تعریف شده همانند شروع و پایان، کلیک کردن، راست کلیک و حرکت نشانگر را با دست خود نشان می‌دهد [۸].

پیاده‌سازی یک کنترل کننده نشانگر رایانه که با امواج مغزی کار می‌کند نیز بدین صورت است که با استفاده از یک هدست^۵ سیگنال‌های الکتروانسفالوگرام^۶ دریافت شده و سپس با حذف سیگنال‌های ناخواسته، پیش‌پردازش می‌شوند. سیگنال‌های باقی مانده شامل اطلاعاتی مفیدی هستند که می‌توان با استخراج ویژگی‌های منحصر به فرد آن‌ها، الگوریتم‌های یادگیری ماشین را در راستای طبقه‌بندی این امواج، آموزش داد [۹].

به عنوان مثال هنگامی که فرد نشانگر ماوس را در جهت حرکت به سمت بالا تصور می‌کند، سیگنال‌های مغزی او شامل ویژگی‌هایی بوده که نسبت به حرکت در جهت پایین، متمایز هستند. با استفاده از همین ویژگی‌ها، دسته‌های مدنظر تعریف شده و به دو طبقه‌بند ماشین بردار پشتیبان^۷ و شبکه عصبی اعمال می‌شوند. در نهایت نیز خروجی طبقه‌بندها به دستوری در راستای کنترل ماوس تبدیل خواهد شد [۹].

علاوه بر استفاده از سیگنال‌های EEG از سیگنال‌های الکترومايوگرافی^۸ که سیگنال‌های مربوط به عصب و عضله هستند استفاده می‌شود؛ به گونه‌ای که با استفاده از یک دست‌بند این سیگنال‌ها دریافت می‌شوند و از

²Kinect Sensor

³EEG Signals

⁴Infrared

⁵Headset

⁶Electroencephalogram (EEG)

⁷Support Vector Machine (SVM)

⁸Electromyography (EMG)

آنچایی که برای حالت‌های مختلف دست متفاوت هستند، می‌توان با یکی از ابزارهای طبقه‌بندی یادگیری ماشین همچون k-NN آن‌ها را طبقه‌بندی کرد [۱۰].

اگرچه استفاده از سنسور کینکت، یک هدست یا یک دستبند که به ترتیب حرکت دست، سیگنال‌های مغزی و سیگنال‌های عضله را به فرمان ماوس تبدیل می‌کنند، سخت‌افزار ماوس را به کلی حذف کرده‌اند اما این روش‌ها نیاز به سخت‌افزارهای جایگزین داشته که قیمت آن‌ها بیشتر از قیمت ماوس رایانه است. همچنین این ابزار سنگین‌تر و حجمی‌تر از ماوس بوده و همراه داشته آن‌ها در هر شرایطی سخت‌تر از به همراه داشتن ماوس است. در واقع در صورت حذف سخت‌افزار ماوس، بهتر است از سخت‌افزار جایگزینی استفاده نشود و یک نرم‌افزار در راستای کنترل نشان‌گر رایانه ارائه شود.

۳-۲ تشخیص حرکت دست

علم و فناوری تعامل انسان و رایانه به مطالعه، طراحی، پیاده‌سازی و ارزیابی تعاملات بین کاربران انسانی و رایانه‌ها می‌پردازد. در واقع HCI نقطه تقاطع چندین علم از جمله علوم رایانه، روانشناسی، جامعه‌شناسی، انسان‌شناسی و طراحی صنعتی است [۱۱] و [۱۲].

یکی از روش‌های طبیعی تعامل انسان و رایانه تشخیص حرکت دست بوده که پتانسیل حذف ابزاری همچون ماوس و صفحه کلید را دارد. در واقع تشخیص حرکت دست یک پیاده‌سازی محاسباتی از حرکات دست انسان در هنگام استفاده از ماشین‌ها است که این حرکات را به فرامین معنی‌دار تبدیل می‌کند [۱۲].

کارهای انجام‌شده در زمینه تشخیص حرکات دست مبتنی بر تماس یا مبتنی بر بینایی ماشین و بدون تماس هستند. معمولاً کارهای قدیمی‌تر مبتنی بر تماس بوده‌اند. به عنوان مثال دستکش داده در راستای تشخیص حرکت دست و دنبال کردن حرکات آن طراحی شده است. درون دستکش‌هایی از این قبیل حسگرهای دما، سرعت و... وجود دارد. این قطعات و اتصالات مربوط به آن‌ها موجب محدود شدن حرکات دست می‌شوند که برقراری ارتباط طبیعی با ماشین را مشکل می‌سازند. علاوه بر محدودسازی حرکات دست، سنگینی، قیمت بالا و لزوم همراه فرد بودن از مشکلات دیگر این دستکش‌ها است [۱۳].

با پیشرفت بینایی ماشین و اکتساب اطلاعات از تصاویر، به تدریج این دستکش‌ها نیز ساده‌تر شدند. به عنوان مثال اتصالات درون آن حذف گردید و در راستای دنبال کردن حرکات دست، از یک دوربین استفاده شد. در واقع

فصل ۲. مروری بر مطالعات انجام شده

با استفاده از یک دوربین رنگ دستکش یا شکل کف دست و محل انگشتان، تشخیص داده می‌شود و حرکات دست نیز دنبال می‌گردند [۱۴].

به تدریج دستکش‌ها نیز حذف گردید و جای خود را به نوارهایی متصل بر نوک انگشتان دادند؛ به گونه‌ای که با دنبال کردن رنگ آن‌ها می‌توان نشان‌گر ماوس را به حرکت درآورد. به عنوان مثال در تکنولوژی حس ششم، از چهار نوار رنگی استفاده شده است تا به دوربین در راستای تشخیص انگشتان و در نهایت حرکت آن‌ها کمک کند [۱۵]. در نهایت این نوارها نیز حذف گردید و الگوریتم‌های تشخیص دست، کاملاً بدون تماس و مبتنی بر تشخیص شی در فریم‌های دریافتی توسط دوربین شدند.

فناوری‌های مبتنی بر هوش مصنوعی^۹ را می‌توان جایگزین روش‌های مبتنی بر تماس دانست که از سیستم‌های پردازش تصویر^{۱۰} همچون دوربین استفاده می‌کنند. شاید برترین ویژگی این نوع روش‌ها درجه آزادی برای حرکت دست انسان باشد که در تعامل انسان با ماشین و خصوصاً رایانه اساسی است. در واقع این الگوریتم‌ها، ترکیبی از پردازش تصویر و بینایی ماشین هستند. پردازش تصویر، با استفاده از یک دوربین تصاویر را به عنوان ورودی می‌گیرد، آن‌ها را به فرم دیجیتال تبدیل می‌کند و عملیاتی همچون تغییر اندازه، فیلتر کردن و حذف نویز بر آن‌ها اعمال می‌گردد. از طرفی بینایی ماشین با ایجاد قوه بینایی در ماشین‌ها، منجر به درک آن‌ها شده و به آن‌ها قدرت انتخاب می‌دهد؛ به گونه‌ای که با آموزش یک شبکه بر مجموعه دادگان مناسب، ماشین توانایی تفکیک حالت‌های مختلف آن‌ها را پیدا می‌کند.

در روش‌های مبتنی بر ترکیب پردازش تصویر و بینایی ماشین، تصویر شامل حالت دست به عنوان ورودی از طریق دوربین به سیستم اعمال، ناحیه دست تعیین و به ناحیه تعیین شده، یک برچسب اختصاص داده می‌شود. در گذشته بینایی ماشین تنها اطلاعاتی را که توسط انسان پردازش شده بودند، درک می‌کرد که در ادامه نمونه‌هایی از این قبیل پژوهش‌ها آمده است.

در مقاله [۱۶] با استفاده از دوربین تصاویر گرفته شده و عملیات پیش‌پردازش بر آنها اعمال می‌گردد. به عنوان مثال تصاویر از فضای رنگی RGB به خاکستری برد و با اعمال فیلتر، نویز آن‌ها حذف می‌گردد. سپس با استفاده از تعیین یک آستانه برای رنگ پوست، پس‌زمینه تصاویر حذف می‌گردد؛ بنابراین این مدل تا حدود اندکی در برابر تغییرات شرایط محیطی و یا پیچیده بودن پس‌زمینه مقاوم است. مرحله بعدی استخراج ویژگی

⁹Artificial Intelligence

¹⁰Image Processing

است که با استفاده از روش کدینگ باینری تک‌زنی^{۱۱} و الگوریتم تبدیل ویژگی مستقل از مقیاس^{۱۲} صورت گرفته است. سپس ویژگی‌های استخراج شده به منظور آموزش طبقه‌بند مورد استفاده قرار می‌گیرند تا تصاویر ورودی به الگوریتم را در دسته مناسب قرار دهند. در نهایت تصاویر مربوط به هر دسته به فرمان مدنظر در جهت تعامل با رایانه تبدیل می‌شوند [۱۶].

در مقاله [۱۷] به جای فریم، ویدئو وارد بخش پردازش می‌شود تا قطعه‌بندی تصویر^{۱۳} در دو مرحله تشخیص پوست و تخمین مدل میانه^{۱۴}، به ترتیب در راستای تشخیص دست و حذف پس‌زمینه صورت گیرد. در این مقاله از هیچ روش دارای یادگیری استفاده نشده و تنها دنباله حرکت مربوط به دست و انگشتان آن استخراج شده است. در نهایت مختصات دست محاسبه و در راستای کنترل ماوس از آن استفاده می‌شود [۱۷].

در برخی مقالات همچون مقاله [۱۸] از ترکیب دست و سر برای کنترل نشان‌گر ماوس استفاده شده است. بدین صورت که هر فریم از پس‌زمینه تفرق شده و با استفاده از تعریف یک آستانه برای رنگ پوست، بخش مربوط به دست یا سر جدا می‌گردد. بعد از جداسازی دست، در راستای استخراج ویژگی بزرگترین کانتور انتخاب می‌شود و با استفاده از الگوریتم کم‌شیفت که مبتنی بر رنگ است، حرکات دست دنبال می‌گردد [۱۸].

در مقاله [۱۹] نیز از قطعه‌بندی تصویر استفاده شده است. به این صورت که ابتدا پیکسل‌های نماینده پوست در ویدئو تشخیص داده می‌شوند. پیکسل‌های پوست با تعیین آستانه بر روی سه کanal RGB مشخص می‌گردند. در واقع با این کار، قسمت پوست از پس‌زمینه جدا می‌گردد و فریم‌ها به صورت تصویر باینری با پس‌زمینه مشکی و پوست سفید تبدیل می‌شوند [۱۹].

در این مرحله تصویر خروجی مربوط به پوست کاربر نویزی بوده و دارای نقاط توخالی است که باید پر شوند. یک روش ساده برای پر کردن نقاط خالی بین پیکسل‌های پوست، یا حتی تهی کردن پیکسل‌های مربوط به پس‌زمینه، استفاده از پیکسل‌های همسایه است. پس از حذف نویز، یک تصویر سیاه و سفید از پوست انسان (سر و دست) و پس‌زمینه باقی می‌ماند. بدینهی است در این مرحله استفاده از روش بزرگ‌ترین کانتور^{۱۵} مناسب نیست؛ زیرا دست و سر هر کدام می‌توانند کانتور بزرگ‌تر را تشکیل بدهند. در این مقاله از یک طبقه‌بند مبتنی بر شبکه‌های عصبی و مدل پیش‌پردازش VGGNet استفاده شده است تا تصاویر سر و دست تفکیک گرددند [۱۹]. همان طور که مشاهده شد، در الگوریتم‌های مختلف فریم‌های شامل دست به عنوان ورودی وارد سیستم شده

¹¹Monogenic Binary Coding (MBC)

¹²Scale Invariant Feature Transform (SIFT)

¹³Image Segmentation

¹⁴Approximate Median Model

¹⁵Contour

فصل ۲. مروری بر مطالعات انجام شده

و با استفاده از تعیین یک محدود در فضای رنگی، بخش مربوط به پوست یا دست جدا می‌گردد. سپس ویژگی‌های منحصر به فرد هر حالت استخراج می‌شود و با استفاده از این ویژگی‌های متمایزکننده، حالت‌های مختلف دست طبقه‌بندی می‌شود. در نهایت نیز برچسب متناظر با دسته‌ها به فرمان تعریف شده برای کنترل نشان‌گر رایانه تبدیل می‌شوند. البته در برخی موارد برای تفکیک حالت‌های مختلف دست از زاویه بین انگشت شست و اشاره نیز استفاده می‌شود [۲۰]. همچنین می‌توان از فضاهای رنگی دیگر برای تعیین آستانه رنگ پوست کاربر استفاده کرد؛ به گونه‌ای که با استفاده از رنگ پوست و مقادیر آن در فضای رنگی HSV بین پوست و سایر اجزا تصویر تمیز قائل شد [۲۱].

به عنوان مثال، در مقاله [۲۲] یک دوربین با استفاده از یک پایه نگهدارنده در بالای یک صفحه تماماً آبی قرار داده شده است. فریم‌های دریافت شده توسط دوربین، به فضای رنگی HSV تبدیل می‌شوند. برای مقدار رنگ ^{۱۶} یک آستانه تعریف شده است تا بخش مربوط به دست و پس‌زمینه تفکیک شوند. از آن جایی که رنگ آبی تفاوت زیادی با رنگ پوست انسان دارد، صفحه پس‌زمینه به رنگ آبی انتخاب شده است [۲۲].

در نهایت یک تصویر سفید و سیاه برای دست و پس‌زمینه آبی به دست می‌آید. در راستای طبقه‌بندی حرکات دست، یک زاویه برای دست تعریف شده تا با استفاده از آن، حالت‌های تعریف شده برای کلیک یا راست‌کلیک کردن و حرکت نشان‌گر از یکدیگر متمایز گردند. بالاترین پیکسل مربوط به دست در تصویر، اولین پیکسل دست از سمت راست و اولین پیکسل از سمت چپ نقاط تشکیل دهنده این زاویه هستند به گونه‌ای که بالاترین پیکسل از دست، رأس زاویه را تشکیل می‌دهد [۲۲].

به طور کلی الگوریتم‌های تشخیص حرکت دست شامل دو مرحله اصلی هستند که در ادامه قابل مشاهده هستند:

۱. تشخیص دست در تصویر

۲. طبقه‌بندی حالت دست و تخصیص فرمان به آن

در راستای تشخیص رنگ پوست انسان می‌توان فضای رنگی RGB را به یک فضای رنگی مناسب همچون HSV یا YCrCb تبدیل کرد و محدوده رنگ پوست را مشخص نمود. در واقع مزیت این فضاهای رنگی آن است که تنها یک مؤلفه رنگ دارند که می‌توان با انتخاب یک بازه مناسب از آن، محدوده رنگی پوست انسان را جدا

^{۱۶}Hue

نمود. حال لازم است هر پیکسل از تصویر با یک آستانه تعریف شده برای رنگ پوست مقایسه گردد تا پیکسل های مربوط به پوست از پیکسل های پس زمینه تفکیک شوند.

در نتیجه مرحله قبل، یک ماسک تهیه می شود که پیکسل هایی که با رنگ پوست انسان همخوانی دارند، سفید و سایر پیکسل های تصویر سیاه هستند. پس از تهیه ماسک، کانتورهای مربوط به دست به دست آمده و مناسب ترین آنها انتخاب می گردد. در نهایت نیز با روش های مبتنی بر ظاهر دست، همچون اندازه یک زاویه خاص یا روش های مبتنی بر بینایی ماشین همچون روش های کلاسیک یا شبکه های عصبی، حالت های مختلف دست طبقه بندی می شوند.

تا به اینجا روش های تشخیص حرکت دست با رویکرد مبتنی بر کانتور^{۱۷} بررسی شدند. همان طور که پیشتر گفته شد، این رویکردها تلاش بر حذف پس زمینه و استخراج دست و تشخیص پیکسل به آن را با استفاده از بافت یا رنگ پوست دارند. روش های مبتنی بر کانتور سراسرت و به راحتی قابل پیاده سازی هستند اما قدرت تعمیم این روش ها کم است. همچنین از آنجایی که این روش ها پیکسل های مربوط به پوست را از سایر پیکسل ها جدا نمی کنند، وابسته به رنگ پوست افراد مختلف بوده و بعضی نیاز است تا کاربر، به صورت دستی رنج پوستی خود را وارد کند [۲۳].

به عنوان مثال این روش ها در شرایط نوری متفاوت که منجر به تغییرات رنگ پوست می شود، خوب عمل نمی کنند. علاوه بر آن، این روش ها در شرایطی که پس زمینه تصاویر شلوغ یا متفاوت از پس زمینه هایی است که الگوریتم با آن آموخته دیده، از پویایی^{۱۸} پایینی برخوردار هستند. در ادامه نمونه های متتنوع تری بررسی می گردد. نمونه هایی که تا به اینجا معرفی گردید، فریم ها را به صورت ایستا بررسی می کند؛ به گونه ای که هر فریم مستقل از فریم های قبلی درنظر گرفته می شود. در برخی موارد تشخیص حرکت دست به صورت پویا است و فریم ها به صورت ترکیبی از بعد فضای زمان بررسی می گردند. به عنوان مثال، پس از تشخیص دست توسط یکی از حالت های گفته شده، اطلاعات به یک دنبال کننده وارد و بر اساس فریم های قبلی، دست دنبال می شود. در واقع تشخیص دهنده دست اطلاعات کامل در ابعاد فضای داشته و بر کلیت هر فریم واقف است. در حالی که دنبال کننده از فریم های قبلی نیز اطلاعات دارد؛ اما تنها بر حالتی از دست که دنبال می کند، مشرف است و در صورت تغییر حالت دست، آن را گم می کند [۲۴].

البته از آن جایی که ماهیت دست متغیر است و ممکن است حالت های مختلفی از دست بر یک فرمان

¹⁷Contour-based Approaches

¹⁸Reliability

فصل ۲. مروری بر مطالعات انجام شده

دلالت داشته باشد، دنبال کردن دست کار سختی خواهد بود. برای رفع این مشکل از یه دوربین دیگر در راستای تشخیص عمق تصاویر نیز استفاده می شود که دیگر صرفه اقتصادی ندارد [۲۵].

الگوریتم تبدیل ویژگی مستقل از مقیاس (SIFT) یکی از سریع ترین ابزارهای استخراج ویژگی است که تا قبل از شبکه های عصبی مورد توجه محققین در این حوزه بوده است. در واقع SIFT با استفاده از مجموعه تصاویر مرجع، تعدادی نقاط کلیدی ^{۱۹} به عنوان ویژگی های منحصر به فرد دست انتخاب می کند. سپس در تصاویر جدید به دنبال نقاط کلیدی ذخیره شده می گردد و شی مدنظر را پیدا می کند.

به عنوان مثال، در مقاله [۲۶] با استفاده از الگوریتم SIFT، دست در فریم های متوالی دنبال می شود. در این مقاله دیگر از روش های سنتی همچون حذف پس زمینه استفاده نشده است. در نهایت نیز هشت حالت شامل شروع و پایان برای دست تعریف شده است تا بتوان با استفاده از آن ها، ماشین های مختلف همچون پنکه، ماشین لباس شویی و ... را کنترل کرد [۲۶].

با پیشرفت یادگیری عمیق و امکان دسترسی به مجموعه دادگان بزرگ، شبکه های عصبی را می توان جایگزین مناسبی برای الگوریتم های کلاسیک بینایی ماشین دانست؛ چرا که با استفاده از این شبکه ها مشکلات گفته شده همچون تغییرات نوری و پس زمینه های متفاوت، بهبود یافته است.

از طرفی استفاده از شبکه های عصبی در بیشتر کاربردها خصوصاً زمینه های بینایی ماشین ^{۲۰} به نتایج بهتری رسیده، زیرا مرحله استخراج ویژگی را حذف کرده و دخالت انسان در طراحی الگوریتم کمتر شده است. البته روش های مبتنی بر یادگیری عمیق به دلیل پیچیدگی بیشتر نسبت به روش های مبتنی بر کانتور دارای سرعت پردازش پایین هستند. همچنین این الگوریتم ها نیاز به مجموعه دادگان بزرگ داشته که در برخی موارد جمع آوری دادگان متناسب با کاربرد کاری دشوار خواهد بود.

در واقع می توان از تشخیص دهنده های اشیا که بر پایه شبکه های عصبی ساخته شده اند، در جهت تشخیص دست استفاده کرد. در فصل آتی، الگوریتم های تشخیص اشیا مبتنی بر یادگیری عمیق توضیح داده می شود. همچنین گفته می شود که برخی از این روش ها مناسب برای کاربردهای بلاذرنگ با هزینه کاهش دقت هستند. الگوریتم های معروف تشخیص اشیا مبتنی بر یادگیری عمیق، الگوریتم های SSD ^{۲۱} و YOLO ^{۲۲} هستند.

در مقاله [۲۷] در راستای ایجاد ارتباط بین افراد ناشنوا و رایانه، ده حالت برای دست تعریف شده است. در

¹⁹Keypoints

²⁰Computer Machine

²¹Single-stage Detector (SSD)

²²You Only Look Once (SSD)

ابتدا با استفاده از تشخیص دهنده SSD ناحیه دارای سر و شانه تعیین می‌گردد. در واقع تشخیص سر و شانه به دلیل ماهیت ثابت آن‌ها ساده‌تر از تشخیص دست است. حال در صورت تشخیص سر و شانه کاربر، ناحیه دارای شانه و سر به یک تشخیص دهنده SSD دیگر وارد می‌شود تا دست تشخیص داده شود و حالت آن مشخص گردد [۲۷].

همچنین در مقاله [۲۸] چهار حالت از اشارات دست درنظر گرفته شده است و با استفاده از معماری VGG16، الگوریتم تشخیص دهنده دست SSD آموزش داده می‌شود. مجموعه دادگان مورد استفاده در این مقاله ۴ حالت دست بوده که به آن‌ها چهار فرمان اختصاص داده می‌شود. تصاویر مورد استفاده در این مقاله دارای سه پس زمینه شلوغ هستند [۲۸].

در مقاله [۲۹] از الگوریتم تشخیص دهنده YOLOv2 با اندکی تغییرات در جهت شناسایی ده حالت دست، استفاده شده است. تغییرات ایجاد شده در جهت کاهش حجم محاسباتی تشخیص دهنده بوده و برخی لایه‌های الگوریتم اصلی حذف شده‌اند. همچنین با استفاده از بسط تیلور، نگاشتهای مهم ویژگی ارزیابی می‌شوند. سپس از بیرون اندازی انتخابی^{۲۳} استفاده شده تا وابستگی شبکه تشخیص دهنده نسبت به ویژگی‌های کم‌اهمیت کاهش یابد؛ به گونه‌ای که پس از هر لایه کانولوشنی یک مازول بیرون‌اندازی انتخابی قرار داده شده است تا ویژگی‌هایی که مقدار بسط تیلور آن‌ها کمترین است، حذف گردند [۲۹].

در مقاله [۳۰] با استفاده از مجموعه دادگان دست الگوریتم YOLOv2 به گونه‌ای آموزش داده شده تا در هنگام تشخیص دست توسط این الگوریتم، برچسب متاظر با حالت آن نیز مشخص گردد. همچنین پیش از ورود نمونه‌های مجموعه دادگان به تشخیص دهنده دست، ویژگی‌های آن‌ها توسط ResNet-50 استخراج می‌گردد تا در نهایت سیستم پخش چندرسانه‌ای کنترل گردد [۳۰].

اگرچه مقالات اخیر در جهت کنترل نشان‌گر رایانه نبوده‌اند، اما به دلیل وجود تشخیص دست و طبقه‌بندی حالت‌های آن به صورت در این بخش ارائه شدند. به طور کلی استفاده از الگوریتم‌های تشخیص اشیا که بر پایه شبکه‌های عصبی هستند دقت بالاتری نسبت به الگوریتم‌های مبتنی بر روش‌های کلاسیک بینایی ماشین دارند؛ اما به دلیل پردازش سنگین آن‌ها از سرعت پایین‌تری برخوردار هستند. در فصل چهارم یکی از روش‌های تشخیص دست با استفاده از الگوریتم SSD معرفی می‌گردد و به تصاویر خروجی آن یک برچسب متناسب در جهت کنترل نشان‌گر رایانه اختصاص داده می‌شود.

²³Selective-dropout

۴-۲ نتیجه‌گیری

در این فصل علاوه بر بررسی سیر تکامل ماوس، نمونه‌ای از ابزارهای تعامل انسان با رایانه بر مبنای تشخیص دست بررسی گردید. گفته شد روش‌های سخت افزاری که مبتنی بر تماس بوده تعامل کاربر با رایانه را دچار محدودیت می‌کند. همچنین روش‌های مبتنی بر بینایی ماشین که از روش‌های کلاسیک یادگیری ماشین استفاده می‌کنند، غیرقابل اعتماد و وابسته به شرایط محیط هستند. در فصل آتی، مفاهیم و پیش‌نیازهای پژوهش پیش‌رو تعریف می‌گردند تا در فصل چهارم از آن‌ها بهره برده شود.

فصل ۳

مفاهیم و مقدمات پژوهش

۱-۳ مقدمه

در فصل پیش رو به بررسی اجمالی مفاهیم اساسی یادگیری عمیق خصوصاً شبکه‌های عصبی کانولوشن و نحوه آموزش آن‌ها پرداخته می‌شود. همچنین عملکرد برخی ابزارهای کمکی که در راستای بهبود این شبکه‌ها وجود دارد، بررسی می‌شود. در انتهای فصل نیز روش‌های تشخیص اشیا مبتنی بر شبکه‌های کانولوشن به همراه معایب و مزایای آن‌ها عنوان می‌گردد.

۲-۳ یادگیری عمیق

یادگیری عمیق یکی از روش‌های یادگیری ماشین در راستای پردازش تصاویر دیجیتالی بوده که توانایی حل مسائلی همچون طبقه‌بندی^۱، تقسیم‌بندی و تشخیص اشیا^۲ در داده‌های تصویری را دارد. امروزه با کمک یادگیری عمیق مسائلی که تا چند سال پیش حل نشده بودند، با دقیقی نزدیک به انسان حل می‌شوند. یکی از مهم‌ترین مشکلات الگوریتم‌های سنتی یادگیری ماشین، مرحله استخراج ویژگی است؛ چرا که مرحله یافتن ویژگی برای تصاویر خصوصاً هنگامی که تعداد دسته‌ها افزایش می‌یابد، کاری طاقت‌فرسا است.

¹Classification

²Object Detection

یادگیری عمیق با معرفی مفهوم یادگیری پایان به پایان^۳، مجموعه دادگانی که شامل تصاویر و برچسب متناظر با آنها است را به شبکه اعمال می‌کند و بر اساس روابط بین نمونه‌های موجود در مجموعه دادگان، شبکه را آموزش می‌دهد. می‌توان پیاده‌سازی یادگیری عمیق را به صورت یک بلوک در نظر گرفت که داده‌ها به آن وارد و خروجی‌های مطلوب نیز برای شبکه مشخص می‌گردد. حال این بلوک است که باید پارامترهای خود را در راستای رسیدن هر ورودی به خروجی مطلوب متناظر خود، تنظیم کند. اگرچه یادگیری عمیق به استخراج ویژگی احتیاج نداشته اما عدم شفافیت این بلوک پردازشی از معایب یادگیری عمیق است [۳۱]

در واقع روش‌های سنتی بینایی ماشین قابل اثبات بوده و از منظر توان بهینه هستند. در حالی که الگوریتم‌های یادگیری عمیق دقت بالاتر را در ازای افزایش توان محاسباتی ارائه می‌دهند؛ بنابراین استفاده از روش‌های ترکیبی^۴ که روش‌های سنتی بینایی ماشین و یادگیری عمیق را ادغام می‌کند، مزایای هر دو روش را دارند. همچنین یادگیری عمیق که از روش‌هایی همچون شبکه‌های عصبی کانولوشن استفاده می‌کند، هزینه‌های دیگری همچون نیاز به مجموعه دادگان بزرگ، پردازش محاسباتی قوی و زمان یادگیری طولانی‌تر دارد [۹].

۳-۳ شبکه عصبی مصنوعی

بشر همواره در حال جمع‌آوری اطلاعات توسط حسگرهای خود یا همان حواس پنجگانه است. یکی از حواس پنجگانه انسان سیستم بینایی او بوده که فقط تغییرات روشنایی را درک می‌کند و این مغز است که توانایی تشخیص دادن، دنبال کردن و برچسب زدن اشیا را دارد. در واقع انسان می‌تواند با دقت بالا محل اشیا مختلف را بیابد و آن‌ها را طبقه‌بندی کند [۳۲].

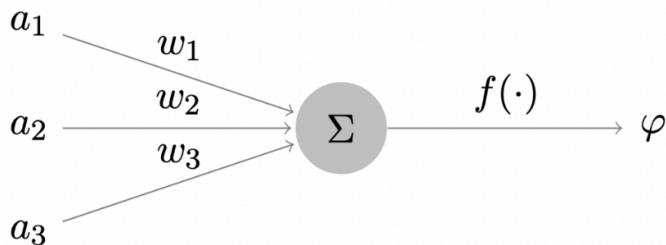
در یادگیری ماشین، شبکه عصبی مصنوعی^۵ را می‌توان یک تقلید از درک بصری دانست؛ چرا که سیستم بینایی مدلی از نور را به عنوان ورودی می‌گیرد، آن را پردازش می‌کند و خروجی قابل تفسیر را ارائه می‌دهد. در نتیجه سیستم بینایی را می‌توان به صورت یکتابع ریاضی پیچیده درنظر گرفت. اگرچه اتصالات نورون‌های درون مغز انسان پیچیده‌تر از نورون‌های شبیه‌سازی شده است اما ایده شبکه‌های عصبی و پس از آن شبکه‌های عصبی کانولوشن الهام گرفته از روابط نورون‌های درون مغز انسان است [۳۲].

³End-to-end Learning

⁴Hybrid Approaches

⁵Artificial Neural Network (ANN)

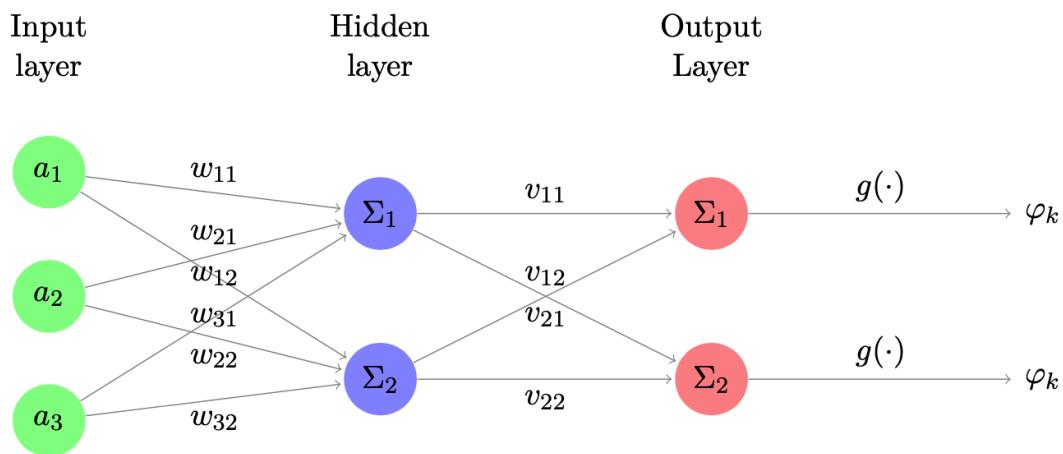
در شبکه‌های عصبی، حاصل ضرب ورودی یا یک ویژگی خاص در وزن متضاد به خود محاسبه می‌شود. برای تسهیل فرآیند یادگیری وزن‌های شبکه، حاصل ضرب های محاسبه شده در هر لایه با یکدیگر و همچنین یا یک بایاس جمع می‌شوند و به تابع فعال ساز^۶ غیرخطی اعمال می‌شوند. از آنجایی که تابع فعال ساز ReLU نماینده بیولوژیکی بهتری برای عملکرد مشابه درون مغز بوده، استفاده از آن خصوصاً در شبکه‌های عصبی کانولوشن، مرسوم است. در شکل ۱-۳ نمونه‌ای از یک نورون شبیه‌سازی شده که در شبکه‌های عصبی مورد استفاده قرار می‌گیرد، آمده است [۳۲].



شکل ۱-۳: نمونه‌ای از یک نورون در شبکه‌های عصبی [۳۳]

نورون‌های مختلف با یکدیگر ترکیب می‌شوند و لایه‌ها را می‌سازند. ورودی همه نورون‌های ترکیب شده در یک لایه، یکسان بوده اما وزن‌های هر نورون و ورودی، منحصر به فرد خواهد بود. از طرفی لایه‌های مختلف نیز ترکیب می‌شوند و شبکه عصبی را می‌سازند. از پارامترهای مهم در شبکه‌های عصبی می‌توان به تعداد نورون‌ها، تعداد لایه‌ها، توابع فعال ساز و بعضی اندازه ورودی اشاره کرد که یک روش کلی برای انتخاب آن‌ها وجود نداشته و با سعی و خطا در راستای بهبود دقت شبکه باید انتخاب می‌شوند. در شکل ۲-۳ نمونه‌ای از یک شبکه عصبی ارائه شده است. همان طور که مشاهده می‌شود ورودی یا ویژگی‌های هر لایه به همه نورون‌های لایه بعد خود متصل هستند [۳۴].

^۶Activation Function



شکل ۳-۲: نمونه‌ای از ترکیب لایه‌ها در شبکه‌های عصبی [۳۳]

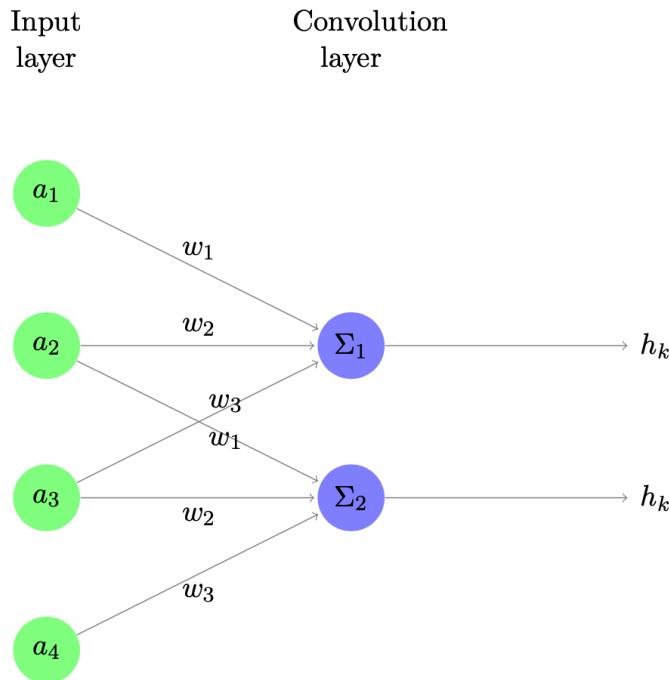
یکی از مشکلات شبکه‌های عصبی، استفاده از لایه‌های تماماً متصل^۷ بوده به طوری که همه ورودی‌ها به همه نورون‌ها متصل است. این موضوع در شکل ۲-۳ نیز قابل مشاهده است. اگرچه لایه‌های تماماً متصل برای شبکه‌های کوچک مشکل‌ساز نمی‌شوند؛ اما در شبکه‌های بزرگ که داده‌های با ابعاد بالا دارند، باعث افزایش سیم‌کشی‌های درون شبکه شده که منجر به سخت شدن محاسبه پارامترهای شبکه و در نهایت یادگیری آن می‌شود. در واقع شبکه‌های کانولوشنی برای مقابله با بزرگ شدن ماتریس وزن و حجم محاسباتی معرفی شده‌اند. چرا که در این نوع از شبکه‌های عصبی، اتصالات دیگر به صورت تماماً متصل نبوده و دارای اتصالات محلی هستند؛ به طوری که ورودی‌ها تنها به نورون‌های نزدیک به خود متصل می‌گردند. در شکل ۳-۳ نمونه‌ای از یک شبکه کانولوشنی و نحوه اتصالات آن آمده است [۳۴].

۱-۳-۳ معماری لایه‌ها

همواره یکی از دغدغه‌های اصلی در علوم داده، تنظیم فرآپارامترها^۸ بوده است؛ زیرا انتخاب این فرآپارامترها از یک قاعده کلی پیروی نکرده و باید در راستای بهبود عملکرد شبکه با سعی و خطا تعیین گردند. در شبکه‌های عصبی نیز انتخاب تعداد لایه‌ها و به طور کلی یافتن معماری مناسب در راستای تعیین این فرآپارامترها حائز اهمیت است.

⁷Fully Connected (FC)

⁸Hyper-parameters



شکل ۳-۳: یک شبکه عصبی کانولوشنی و نحوه اتصالات آن [۳۳]

پیش‌تر اشاره شد از ترکیب نورون‌ها لایه‌ها تشکیل می‌شوند. حال مجموعه‌ای از این لایه‌ها که شامل لایه ورودی، لایه‌های پنهان و لایه خروجی هستند، شبکه عصبی را می‌سازند.

از آنجایی که شبکه‌های کانولوشن زیرمجموعه شبکه‌های عصبی هستند، ساختاری لایه‌ای دارند. مهم‌ترین بلوک در شبکه‌های کانولوشن، لایه کانولوشن^۹ بوده که عمل کانولوشن را بر روی ورودی اعمال می‌کند. بر اساس مشخصات لایه کانولوشن یا همان فرآپارامترها اندازه خروجی این لایه مشخص می‌شود. این فرآپارامترها می‌توانند اندازه یا تعداد فیلتر، گام^{۱۰} و مقدار لایه‌گذاری^{۱۱} باشند. اندازه فیلتر، گام و میزان لایه‌گذاری بر عرض و ارتفاع خروجی اثر می‌گذارند. در حالی که تعداد فیلتر عمق ورودی را مشخص می‌سازد. از مزایای اصلی لایه کانولوشن، استفاده از وزن‌های مشترک بوده که منجر به کاهش تعداد پارامترهای شبکه و در نهایت حجم محاسباتی می‌شود. تا به اینجا لایه‌های کانولوشن به عنوان استخراج‌کننده ویژگی‌های تصاویر معرفی گردید. حال برای بررسی کردن ویژگی‌های استخراج‌شده از لایه‌های تلفیق^{۱۲} به ویژه لایه تلفیق بیشینه^{۱۳} استفاده می‌شود. در واقع از این لایه

⁹Convolution Layer

¹⁰Stride

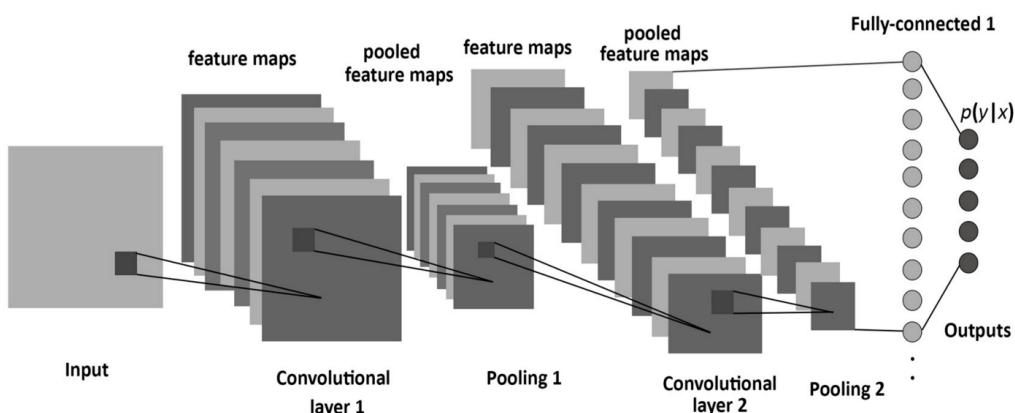
¹¹Padding

¹²Pooling Layer

¹³Max-pooling Layer

به عنوان کاهش دهنده ابعاد ویژگی های استخراج شده استفاده می شود و در این لایه پارامتر یادگیری وجود ندارد. معمولاً در شبکه های کانولوشن پس از استخراج ویژگی و کاهش ابعاد، از لایه های تماماً متصل استفاده می شود. مشاهده گردید که در لایه های تماماً متصل نورون ها به همه نورون های لایه قبل متصل هستند. در واقع تفاوت لایه تماماً متصل با لایه کانولوشن آنجا است که در لایه های کانولوشن هر نورون فقط به مجموعه محدودی از نورون ها در لایه قبل متصل است [۳۵].

در شکل ۴-۲ نمونه ای از یک شبکه کانولوشن به همراه لایه های مختلف آمده است. در این شبکه از لایه های کانولوشن، تلفیق و تماماً متصل استفاده شده است.



شکل ۴-۳: نمونه ای از یک شبکه عصبی کانولوشنی با آرایش سه بعدی [۳۵]

۲-۳-۳ آموزش شبکه عصبی

هدف اصلی شبکه های عصبی، یافتن مقادیر بهینه پارامترهای شبکه (وزن ها و بایاس) در راستای کمینه کردن یک تابع خطا^{۱۴} تعریف شده است. به بیان دیگر نورون ها به منظور رسیدن به یک خروجی آموزش داده می شوند تا فاصله از خروجی مطلوب کاهش یابد. در رابطه ۱-۳ یک نمونه از تابع خطا مورد استفاده در شبکه های عصبی تحت عنوان Cross Entropy Loss ارائه شده است که معمولاً برای کاربردهای طبقه بندی استفاده می شود. در این رابطه M نمایان گر تعداد دسته ها، y_c نمایان گر خروجی یا برچسب واقعی شبکه برای دسته c و p_c نمایان گر خروجی یا همان برچسب پیش بینی شده توسط شبکه برای دسته c است.

¹⁴Loss Function

$$CE = - \sum_{c=1}^M y_c \log(p_c) \quad (1-3)$$

الگوریتم پس انتشار^{۱۵} روشی برای یافتن مشتق تابع خطا بر حسب پارامترهای شبکه است. در واقع روش پس انتشار با محاسبه گرادیان گفته شده، مشخص می‌سازد که تغییر در پارامترها چه تغییری در رفتار کلی شبکه ایجاد می‌کند. در راستای بهینه‌سازی شبکه لازم است ترکیبی مناسب از پارامترها به منظور کمینه کردن تابع خطا پیدا شود. به بیان دیگر هدف از آموزش شبکه آن است که نقطه مینیمم تابع خطا در فضایی با ابعاد تعداد پارامترهای شبکه یافته شود [۳۶].

در راستای افزایش قدرت تعمیم شبکه‌های عصبی کانولوشن، هنگام آموزش شبکه از دو روش نرمال‌سازی دسته‌ای^{۱۶} و بیرون‌اندازی^{۱۷} نورون‌ها استفاده می‌شود. در واقع بیرون‌اندازی با حذف برخی نورون‌ها به صورت تصادفی به شبکه نویز وارد می‌کند تا شبکه برای شرایط واقعی نیز آماده گردد. نرمال‌سازی دسته‌ای نیز تأثیری مشابه بیرون‌اندازی ایجاد می‌کند و منجر می‌شود تا نورون‌های هر لایه تا حدودی مستقل از لایه‌های دیگر عمل کنند [۳۷].

۴-۳ مجموعه دادگان

همان طور که پیش‌تر اشاره شد، یکی دیگر از تفاوت‌های یادگیری عمیق و سایر الگوریتم‌های بینایی ماشین، حجم مجموعه دادگانی است که شبکه نیاز دارد با استفاده از آن‌ها آموزش بینند تا به قدرت تعمیم مناسب برسد. در الگوریتم‌های سنتی تر علاوه بر دادگان، یک سری ویژگی به صورت دستی نیز استخراج و به شبکه داده می‌شود؛ اما در یادگیری عمیق با تعداد دادگان بیش‌تر، شبکه به تنهایی این ویژگی‌های منحصر به فرد را یاد می‌گیرد. اگرچه حذف مرحله استخراج ویژگی به صورت دستی، تغییری مثبت در بینایی ماشین به حساب می‌آید؛ اما افزایش دادگان کار آسانی نبوده و پردازش مجموعه دادگان بزرگ‌تر نیز به قدرت محاسباتی بیش‌تری نیاز دارد. به عنوان مثال برای مسائلی همچون تقسیم‌بندی که مهم است هر پیکسل در دسته درست قرار بگیرد، افزودن داده

¹⁵Back-propagation

¹⁶Batch Normalization (BN)

¹⁷Drop-out

کاری طاقت‌فرسا خواهد بود [۳۸].

علاوه بر اهمیت تعداد دادگان هر دسته در حوزه یادگیری عمیق، نمونه‌های درون هر دسته باید بیشترین تنوع ممکن را داشته باشند تا بتوان شبکه آموزش دیده را به داده‌های دنیا واقعی تعمیم داد. درنظر گرفتن شرایط نوری و پس زمینه‌های مختلف همچنین استفاده از انواع جسم مورد نظر و درنظر داشتن ویژگی‌های گوناگون آن، تنوع دادگان هر دسته را افزایش می‌دهد و امکان تعمیم شبکه آموزش دیده با این دادگان، به دادگان جدید نیز فراهم می‌گردد. در این پژوهش یک مجموعه دادگان متناسب با هدف کنترل نشان گر رایانه با استفاده از اشارات دست کاربر، جمع‌آوری شده است که در فصل چهارم معرفی می‌گردد.

۵-۳ یادگیری انتقالی

از آنجایی که مجموعه دادگان نقش به سزاپی در شبکه‌های عصبی دارد، هنگامی که دادگان کم است، امکان نگاشت یک تابع بر آنها نیز از بین می‌رود. از طرفی ممکن است توان پردازش بر روی همین داده‌های کم نیز مسئله‌ساز باشد. یکی از روش‌های مفید برای آموزش شبکه عصبی با مجموعه دادگان کوچک یا توان محاسباتی کم، استفاده از یادگیری انتقالی^{۱۸} است؛ به گونه‌ای که از معماری یک شبکه یا وزن‌های آن که بر روی مجموعه دادگان بزرگ و متفاوت، با توان پردازشی بالا آموزش دیده شده، استفاده کرد.

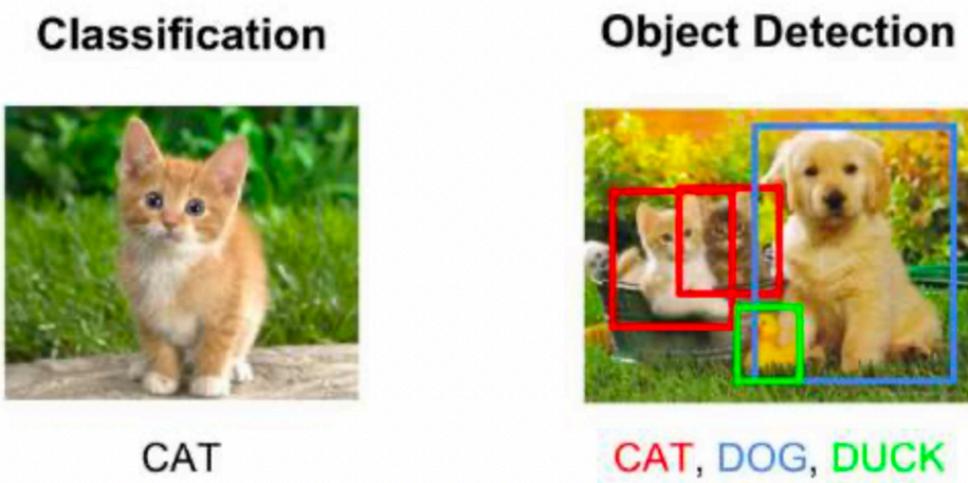
در واقع کارآمدی یادگیری انتقالی به این دلیل است که بیشتر ویژگی‌های اولیه همانند تشخیص لبه در تصاویر مختلف مشترک است. در این پژوهش، در راستای طبقه‌بندی مجموعه دادگان جمع‌آوری شده از معماری شبکه EfficentNet که بر روی مجموعه دادگان ImageNet آموزش دیده است، استفاده شده که در فصل چهارم به آن پرداخته می‌شود. این شبکه نسبت به سایر شبکه‌های کانولوشنی که با دادگان ImageNet آموزش دیده‌اند، از دقت قابل قبول در ازای تعداد پارامتر کمتر برخوردار است. لازم به ذکر است متناسب با شبکه مدنظر جهت طراحی، معمولاً در انتهای شبکه پیش‌آموزش از لایه‌های تماماً متصل استفاده می‌شود [۳۹].

¹⁸Transfer Learning

۶-۳ تشخیص اشیا

قدرت درک انسان نسبت به محیط پیرامون خود از پیچیده‌ترین توانایی‌های بشر است که در مغز او شکل می‌گیرد. انسان با مشاهده پیرامون خود، اطلاعاتی از قبیل اندازه، شکل، بافت، محل و یا فاصله مابین اشیا را درک می‌کند. در واقع بشر در راستای شناسایی محیط اطراف خود، ابتدا از طریق قوه بینایی، محل اشیا را تخمین زده و آن‌ها را طبقه‌بندی می‌کند.

بینایی ماشین نیز به گونه‌ای طراحی شده تا همانند ادراک بصری انسان، با تصاویری که بیش از یک شی در آن‌ها وجود دارد مقابله کند و سعی در تخمین ناحیه و سپس طبقه‌بندی آن‌ها دارد. در شکل ۵-۳ تفاوت تشخیص اشیا^{۱۹} و طبقه‌بندی تصاویر ارائه شده است. در واقع طبقه‌بندی تصاویر، سعی بر آن دارد که کل تصویر را به یک برچسب نگاشت دهد؛ اما در تشخیص اشیا، پس از تعیین محل اشیا درون یک تصویر، به آن‌ها برچسب داده می‌شود؛ به گونه‌ای که یک تصویر ممکن است چندین برچسب داشته باشد [۴۰].



شکل ۵-۳: تفاوت تشخیص اشیا و طبقه‌بندی تصاویر [۴۰]

^{۱۹}Object Detection

۱-۶-۳ تشخیص اشیا با استفاده از شبکه‌های عصبی کانولوشن

اگرچه پیدایش شبکه‌های کانولوشن قدرت بینایی ماشین‌ها را افزایش داد، اما این شبکه‌ها همچنان گنجایش محدودی داشته و توانایی طبقه‌بندی یک تصویر که شامل چندین اشیا است، را ندارند. همان‌طور که پیش‌تر اشاره گردید، تشخیص اشیا شامل دو مرحله تعیین محل اجسام و طبقه‌بندی آن‌ها است؛ این در حالی است که با اعمال شبکه کانولوشن به کل تصویر، مرحله تعیین ناحیه اجسام وجود ندارد. علاوه بر آن، امکان تشخیص چندین برچسب به یک تصویر در شبکه‌های کانولوشن وجود ندارد [۴۰].

در راستا مجهز کردن شبکه‌های کانولوشنی برای حل مسائل تشخیص اشیا، ابتدا تعدادی منطقه موردنظر^{۲۰} در تصویر تعیین می‌شود و سپس شبکه کانولوشن در جهت طبقه‌بندی این مناطق مورد استفاده قرار می‌گیرد. در واقع با تعیین منطقه موردنظر و مشخص نمودن یک کادر محدود^{۲۱} در اطراف شی، تصاویر اولیه به تصاویر قابل اعمال به شبکه کانولوشن، تبدیل می‌گردند؛ چرا که این تصاویر شامل تنها یک جسم هستند و شبکه عصبی کانولوشن به آن‌ها تنها یک برچسب اختصاص می‌دهد [۴۰].

از آنجایی که اشیا ممکن است میزان مساحت متفاوتی اشغال کنند، اندازه کادر محدود متفاوت خواهد بود و در نتیجه تعداد مناطق موردنظر افزایش می‌یابند. این موضوع باعث افزایش حجم محاسباتی می‌شود و محدودیت‌هایی برای کاربردهای بلاذرنگ^{۲۲} ایجاد می‌کند. در فصل دوم برخی الگوریتم‌های تشخیص اشیا برای تشخیص دادن دست، با استفاده از روش‌های سنتی یادگیری ماشین، بررسی گردید. در ادامه به الگوریتم‌های تشخیص اشیا، پس از ظهور شبکه‌های عصبی کانولوشن اشاره می‌شود.

الگوریتم‌های تشخیص اشیا مبتنی بر شبکه کانولوشن متعدد بوده و به یکی از دو دسته تک مرحله‌ای یا دو مرحله‌ای تعلق دارند. تشخیص دهنده‌های تک مرحله‌ای همانند YOLO یا SSD دارای سرعت بالا و دقت قابل قبول هستند که امکان استفاده در کاربردهای بلاذرنگ را ایجاد می‌کنند. تشخیص دهنده‌های دو مرحله‌ای در ابتدا مناطق متعددی را انتخاب کرده و سپس در این مناطق عمل طبقه‌بندی را انجام می‌دهند؛ در صورتی که در تشخیص دهنده‌های تک مرحله‌ای، تصویر به خانه‌های کوچک تقسیم شده و هر خانه به یک طبقه‌بند اعمال می‌گردد. تشخیص دهنده‌های دو مرحله‌ای همانند R-CNN دارای دقت بالا اما سرعت پایین هستند و معمولاً از این روش‌ها برای کاربردهای بلاذرنگ استفاده نمی‌شود [۴۰].

²⁰Region of Interest (RoI)

²¹Bounding Box

²²Real-time

۷-۳ نتیجه‌گیری

در این فصل، ابتدا مباحث اولیه در یادگیری عمیق و شبکه‌های عصبی بررسی گردید. سپس شبکه‌های کانولوشن به عنوان ابزاری مناسب برای طبقه‌بندی تصاویر معرفی شدند. همچنین محدودیت شبکه‌های کانولوشنی در جهت تشخیص اشیا بیان گردید و چگونگی بهبود این شبکه‌ها برای تعیین محل اشیا در تصویر عنوان شد. در نهایت الگوریتم‌های متناسب در جهت تشخیص اشیا و معایب و مزایای دور رویکرد مختلف آن‌ها معرفی شد. در فصل‌های آتی از مفاهیم عنوان شده در این فصل و فصل گذشته در جهت پیشبرد پژوهش استفاده می‌گردد.

فصل ۴

الگوریتم پیاده‌سازی شده

۱-۴ مقدمه

در ابتدای این فصل یک مجموعه دادگان متناسب با حالت‌های مدنظر از اشارات دست، در راستای کنترل نشان‌گر رایانه، جمع‌آوری می‌گردد و در جهت آموزش یک شبکه عصبی کانولوشنی استفاده می‌شوند. سپس شبکه عصبی پیاده‌سازی شده ذخیره می‌شود تا دو شبکه طبقه‌بند و شباهت‌سنج طراحی گردند. در نهایت شبکه‌های طراحی شده با کمک الگوریتم تشخیص‌دهنده دست، تصاویر ورودی به الگوریتم ماوس طراحی شده را طبقه‌بندی می‌کند و به فرامین مدنظر در راستای کنترل نشان‌گر تبدیل می‌شوند.

۲-۴ مجموعه دادگان

همان طور که در فصل‌های پیشین گفته شد، هدف از این پژوهش کنترل نشان‌گر ماوس با استفاده از حرکات دست انسان است. در واقع تصویر مربوط به دست از طریق دوربین به شبکه طراحی شده اعمال می‌شود تا برچسب متناظر آن پیش‌بینی شود و عمل متناظر با برچسب پیش‌بینی شده در راستای کنترل نشان‌گر رایانه انجام گردد. در راستای طراحی یک شبکه عصبی، ابتدا لازم است مجموعه‌ای از تصاویر به منظور آموزش به شبکه اعمال گردد و ارتباط بین نمونه‌های موجود در مجموعه دادگان شبیه‌سازی شود تا شبکه طراحی شده در مرحله آزمایش

بتواند تصاویر جدید را در دسته مربوطه، قرار دهد. بدین منظور مجموعه دادگان متناسب با پژوهش توسط نویسنده این پایان‌نامه جمع‌آوری گردید و حالت‌های متناسب در راستای کنترل نشان‌گر رایانه تعریف شد که برای کاربردهای دیگر قابل استفاده هستند.

مجموعه دادگان تهیه شده شامل ۶۷۲۰ تصویر رنگی مربوط به ۶ زن و ۹ مرد، در ۱۸ مکان مختلف با شرایط نوری متفاوت و پس‌زمینه ساده هستند که به ۴ دسته تقسیم می‌شوند. حالت‌های تعریف شده در مجموعه دادگان به صورت زیر هستند:

- حالت اشاره به راست
- حالت اشاره به چپ
- حالت کف/پشت دست
- حالت مشت

به منظور جلوگیری از انحراف دادگان به سمت دست راست یا دست چپ، از افراد خواسته شده تا تصاویر مربوط به هر دو دست خود را ارسال نمایند؛ در نتیجه شبکه‌ای که با این مجموعه دادگان آموزش می‌بیند برای هر دو گروه راست‌دستان و چپ‌دستان مناسب باشد.

از مجموع ۶۷۲۰ تصویر، ۵۱۲۰ تصویر برای دادگان آموزش^۱ و ۱۶۰۰ تصویر برای دادگان اعتبارسنجی^۲ درنظر گرفته شده است. لازم به ذکر است مجموعه دادگان آموزش دارای توزیع آماری متفاوت نسبت به دادگان اعتبارسنجی و آزمایش است؛ تصاویر مربوط به آموزش شبکه توسط دوربین رایانه و بدون واسطه تهیه گردیده‌اند. این در حالی است که تصاویر مربوط به دادگان اعتبارسنجی با دوربین و به واسطه تشخیص دهنده SSD جمع‌آوری شده‌اند. در بخش آموزش شبکه عصبی این موضوع توضیح داده می‌شود.

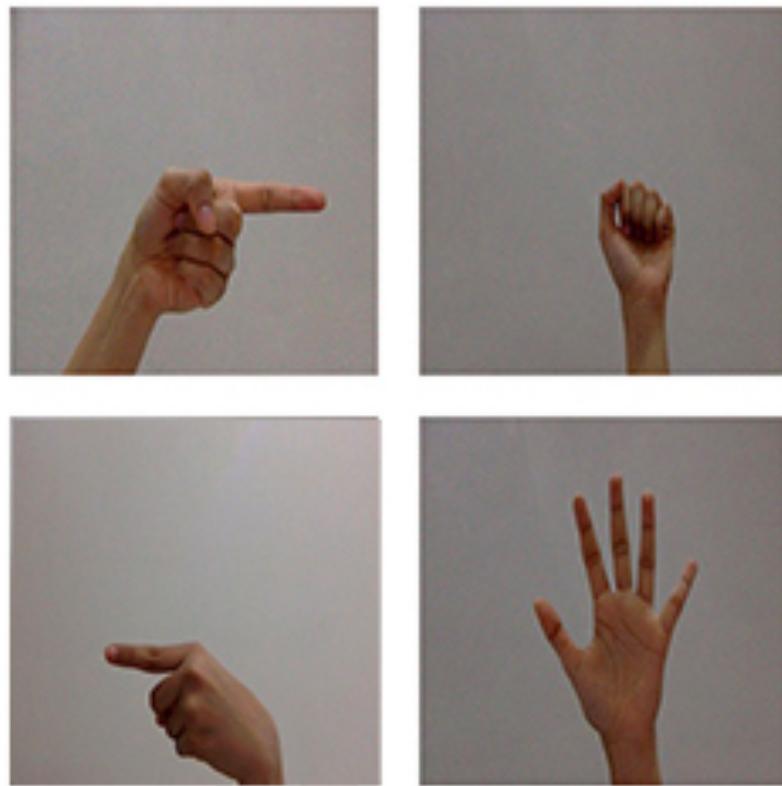
هنگام جمع‌آوری مجموعه دادگان، به دلیل رزولوشن متفاوت در رایانه‌های مختلف، ابعاد تصاویر مربوط به دادگان آموزش یکسان نبوده و بسته به رزولوشن دوربین رایانه افرادی که در تهیه این مجموعه کمک کرده‌اند متفاوت است که برای کاهش حجم تصاویر، به ابعاد ۳۰۰ در ۳۰۰ تبدیل شده‌اند. در حالی که تصاویر مربوط به دادگان اعتبارسنجی با ابعاد ۱۰۰ در ۱۰۰ هستند. البته در زمان آموزش شبکه، همه تصاویر اعم از آموزش و اعتبارسنجی به ابعاد ۷۰ در ۷۰ تبدیل و از سه کanal رنگی استفاده شده است.

¹Training Set

²Validation Set

فصل ۴. الگوریتم پیاده‌سازی شده

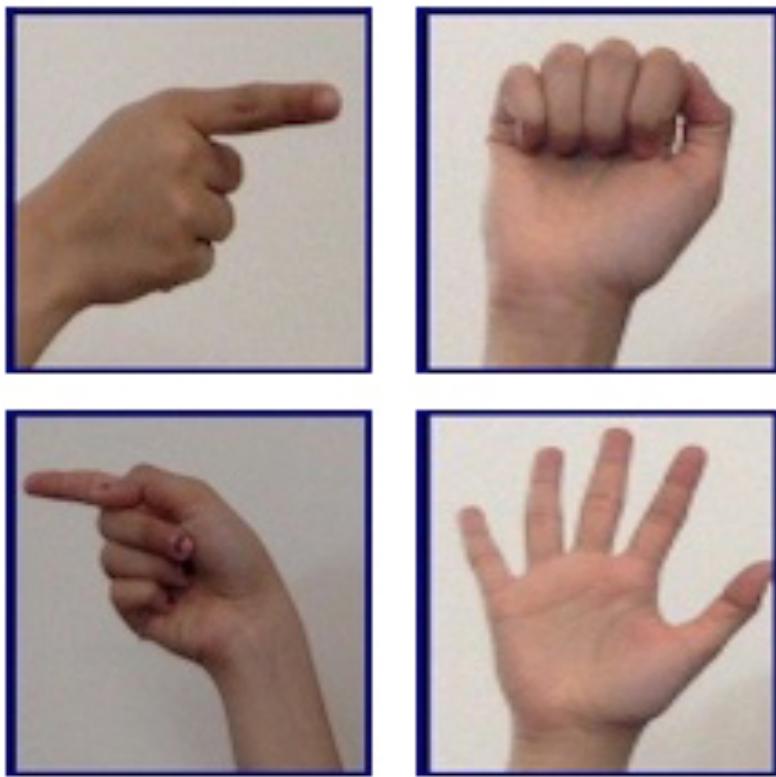
در ادامه نمونه‌ای از تصاویر مربوط به آموزش و اعتبارسنجی شبکه آمده است. همان طور که در تصاویر قابل مشاهده است، رزولوشن تصاویر اعتبارسنجی پایین‌تر از تصاویر آموزش است و ابعاد به صورت دستی یکسان شده است. همچنین لازم به ذکر است نمونه تصاویر موجود برای دادگان آموزش مربوط به دست راست و نمونه تصاویر برای دادگان اعتبارسنجی مربوط به دست چپ کاربر هستند.



شکل ۱-۴: نمونه‌ای از تصاویر اشارات دست تعریف شده در پژوهش برای آموزش شبکه

۳-۴ تشخیص دست

در راستای کنترل نشان‌گر با استفاده از اشارات درنظر گرفته شده، لازم است وجود یا عدم وجود دست در تصاویر ورودی به الگوریتم ماوس طراحی شده مشخص گردد. در این پژوهش به منظور تشخیص دست از تشخیص دهنده SSD که با مجموعه دادگان دست آموزش دیده، استفاده شده است. در واقع پس از تشخیص وجود دست و ناحیه آن، برشی از تصویر شامل ناحیه دست استخراج می‌شود. سپس لازم است برچسب مربوط به اشاره دست



شکل ۴-۲: نمونه‌ای از تصاویر اشارات دست تعریف شده در پژوهش برای اعتبارسنجی شبکه

بریده شده، مشخص گردد که در بخش طبقه‌بندی توضیح داده می‌شود. همچنین در ادامه بخش مربوط به تشخیص دست توضیح داده می‌شود [۴۱].

پیش‌تر توضیحاتی درباره روش‌های تشخیص اشیا و دو رویکرد تک مرحله‌ای و دو مرحله‌ای داده شد. در این پژوهش از رویکرد تک مرحله‌ای و الگوریتم SSD استفاده می‌شود که با مجموعه دادگان EgoHands آموزش دیده است. این دادگان حدود ۱۵۰۰۰ تصویر از دست داشته که در سطح پیکسل قطعه‌بندی شده‌اند. از طرفی برای آموزش SSD از روش یادگیری انتقالی استفاده شده است. روش یادگیری انتقالی که پیش‌تر نیز توضیح داده شد، منجر به کوتاه شدن زمان آموزش شبکه و بهبود عملکرد آن می‌شود. چرا که ویژگی‌های اولیه تصاویر همچون لبه در دادگان مختلف یکسان هستند [۴۱] و [۴۲].

تشخیص دهنده دست استفاده شده در این پژوهش یکی از سریع‌ترین الگوریتم‌های SSD است. پیش‌تر نیز گفته شد در راستای کنترل نشان‌گر ماوس، سرعت ضروری است؛ بنابراین لازم است محاسبات در زمان کم انجام شوند و الگوریتم نهایی توانایی اجرا به صورت بلاذرنگ را داشته باشد.

۱-۳-۴ عملکرد الگوریتم تشخیص دهنده دست

نحوه عملکرد بخش تشخیص دهنده دست بدین صورت است که با استفاده از دوربین مربوط به رایانه، یک فریم گرفته می‌شود. پیش از اعمال فریم به الگوریتم تشخیص دهنده دست، لازم است تا فریم پیش‌پردازش شود. در واقع در این بخش دو پیش‌پردازش بر روی فریم انجام می‌شود. ابتدا لازم است فریم به فضای رنگی RGB تبدیل شود؛ چرا که برای تهیه فریم از کتابخانه OpenCV استفاده شده است و این کتابخانه تصاویر را در فضای رنگی BGR ذخیره می‌کند. سپس ابعاد فریم در فضای رنگی جدید به 300×300 تغییر داده می‌شود. حال می‌توان فریم پردازش شده را به الگوریتم SSD اعمال نمود.

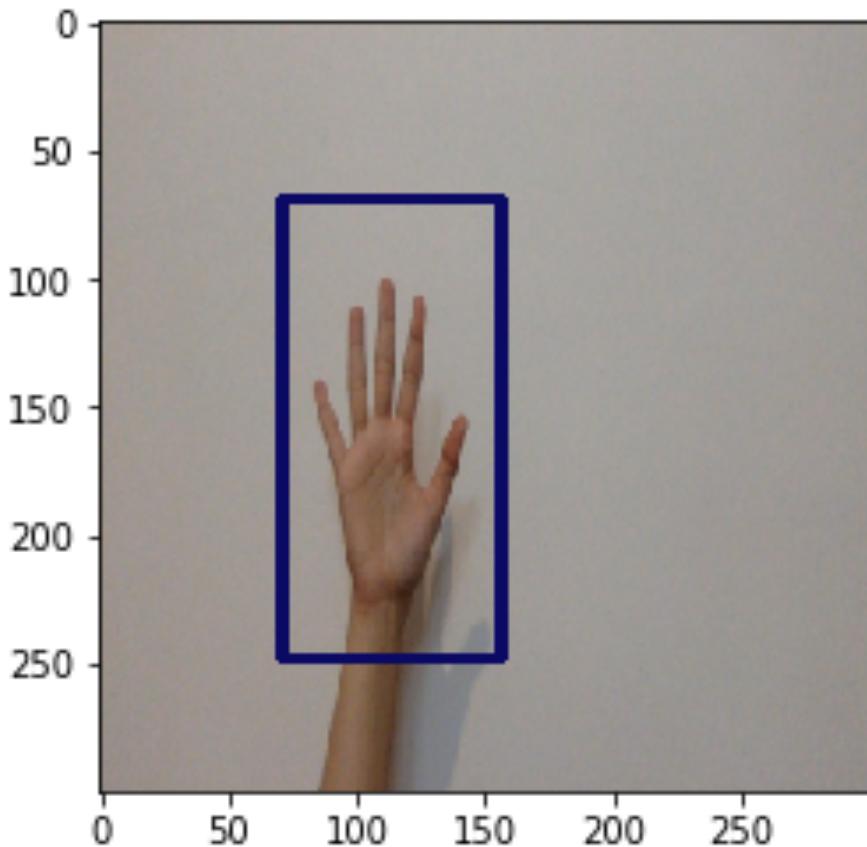
در این مرحله الگوریتم SSD با استفاده از تصویری که به آن اعمال شده، دو پارامتر تولید می‌کند. یکی از پارامترها امتیازی^۳ است که شبکه SSD در صورت وجود دست در کادرهای مدنظر تولید می‌کند. در هر تصویر کادری انتخاب می‌گردد که دارای بیشترین امتیاز است. حال اگر بزرگترین امتیاز مربوط به تصویر بیش از یک مقدار آستانه باشد، فریم شامل تصویر دست بوده و از آن در بخش‌های بعدی استفاده می‌شود. در غیر این صورت، فریم قادر تصویر دست بوده و شبکه SSD فریم جدید را می‌گیرد. لازم به ذکر است آستانه مدنظر در این بخش با سعی و خطأ به دست آمده که 0.25 است [۴۳].

اگر امتیاز فریم بیش از مقدار آستانه بود، الگوریتم SSD یک کادر به دور دست قرار می‌دهد تا ناحیه شامل دست مشخص گردد. شکل ۳-۴ از نمونه فریم‌هایی است که در آن ناحیه دست توسط الگوریتم SSD تشخیص داده شده و به دور آن یک کادر قرار داده است.

در نهایت با استفاده از مختصات کادر محدود برای فریم شامل دست، دو خروجی استخراج می‌شود و به دو قسمت طبقه‌بندی و تخصیص فرمان در جهت کنترل نشان‌گر اعمال می‌گردد. خروجی اول فریمی است که بخش دست آن بریده شده است و پس از تغییر اندازه آن به یک تصویر رنگی با ابعاد 70×70 وارد بخش طبقه‌بندی می‌شود. در شکل ۴-۴ نمونه‌ای از یک فریم شامل دست که الگوریتم SSD تشخیص داده و ناحیه شامل دست آن جدا شده، آمده است. لازم به ذکر است توزیع این دست همانند تصاویر اعتبارسنجی است که پیش‌تر معرفی گردید.

خروجی دوم که در این پژوهش از الگوریتم SSD برداشته شده، مختصات مرکز کادر محدودی است که اطراف دست قرار داده می‌شود. برای یافتن مختصات مرکز کادر، از مختصات بیشینه و کمینه مربوط به طول و

³Score



شکل ۳-۴: تشخیص دست و قرار دادن کادر در اطراف آن توسط تشخیص دهنده SSD

عرض کادر محدود استفاده می‌شود. در روابط ۱-۴ نحوه محاسبه مختصات مرکز کادر محدود برای فریم‌های بریده شده توسط الگوریتم SSD آمده است. در بخش کنترل نشان‌گر نحوه استفاده از این مختصات توضیح داده می‌شود.

لازم به ذکر است منظور از x_{center} و y_{center} مختصات نقطه مرکزی فریم بریده شده است. علاوه بر آن x_{min} و x_{max} به ترتیب عرض از مبدا گوشہ سمت چپ و عرض از مبدا گوشہ سمت راست و y_{min} و y_{max} به ترتیب طول از مبدا گوشه پایین و طول از مبدا گوشه بالا هستند که در واقع مختصات کادر محدود الگوریتم SSD است.

$$x_{center} = \frac{x_{min} + x_{max}}{2}, y_{center} = \frac{y_{min} + y_{max}}{2} \quad (1-4)$$

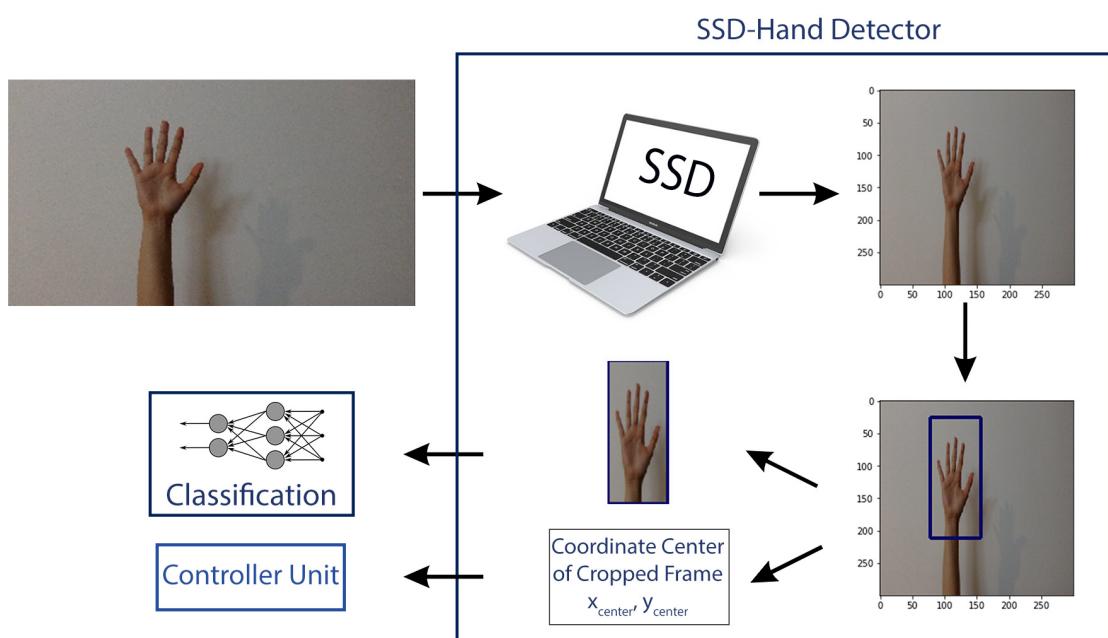
فصل ۴. الگوریتم پیاده‌سازی شده



شکل ۴-۴: فریم بریده شده از ناحیه کادر محدود توسط الگوریتم تشخیص دهنده دست

بنابراین در صورت وجود دست، خروجی‌های تشخیص دهنده مورد استفاده در این پژوهش، مختصات مرکز کادر محدود و فریم بریده شده هستند. در الگوریتم ۱-۴ نحوه عملکرد کلی بخش تشخیص دهنده دست SSD در راستای یافتن دست، ورودی و خروجی‌هایی این بخش مشخص شده است.

الگوریتم ۱-۴ تشخیص دهنده دست استفاده شده به همراه ورودی و خروجی‌های آن



۴-۴ طبقه‌بندی

در بخش قبل مشاهده گردید که با استفاده از دوربین رایانه، یک فریم به الگوریتم تشخیص دهنده اعمال می‌شود و در صورت وجود دست در تصویر، الگوریتم دو خروجی به بخش‌های طبقه‌بندی و تشخیص دهنده فرامین می‌دهد. یکی از این خروجی‌ها فریم بریده شده شامل دست است. خروجی دوم نیز مختصات مرکز فریم شامل ناحیه دست است. هدف از این قسمت، استفاده از مجموعه دادگان جمع‌آوری شده در راستای طراحی یک شبکه عصبی کانولوشنی است. سپس با استفاده از شبکه آموزش دیده، دو مدل برای طبقه‌بندی و شباهت‌سنجی دادگان جدید در جهت تشخیص برچسب و بررسی مفید بودن آن‌ها طراحی می‌گردد.

در واقع برای تصاویر ورودی به شبکه طبقه‌بند می‌توان دو حالت درنظر گرفت:

- فریم بریده شده یکی از چهار حالت تعریف شده در مجموعه دادگان باشد.

- فریم بریده شده متفاوت از حالت‌های تعریف شده در مجموعه دادگان است.

اگر فریمی که به طبقه‌بند وارد می‌شود، جز حالت‌های تعریف شده در مجموعه دادگان باشد، با استفاده از طبقه‌بندی که در این بخش طراحی خواهد شد، برچسب آن مشخص می‌گردد. حال اگر فریم حالتی خارج از حالت‌های مجموعه دادگان باشد، امکان تشخیص یک برچسب جدید به واسطه طبقه‌بند طراحی شده با شبکه عصبی کانولوشنی، وجود ندارد. یکی دیگر از مشکلات شبکه‌های کانولوشنی و حتی الگوریتم‌های تشخیص دهنده، آن است که برای مجموعه دادگان بسته طراحی شده‌اند؛ به گونه‌ای که در این شبکه‌ها، آخرین لایه تماماً متصل به تابع Softmax داده می‌شود تا یک توزیع احتمالاتی بر روی تعداد دسته تعریف شده بسازد [۴۴].

بنابراین شبکه‌های عصبی و الگوریتم‌های مبتنی بر آن به دلیل طبیعت بسته تابع Softmax، برای تعداد دسته‌های محدود مناسب هستند و همه تصاویر حتی حالت‌های خارج از مجموعه دادگان را به یکی از حالت‌های موجود در مجموعه دادگان نگاشت می‌دهند. در این پژوهش برای بررسی دو حالت گفته شده، یک شبکه عصبی شباهت‌سنج طراحی می‌گردد تا فاصله فریم‌های بریده شده را با حالت‌های موجود بسنجد.

در بخش طبقه‌بندی دو شبکه طراحی می‌گردد؛ شبکه اول با استفاده از مجموعه دادگان جمع‌آوری شده، به فریم‌هایی که جز حالت‌های مدنظر هستند، برچسب مناسب اختصاص می‌دهد. شبکه دوم فاصله بین فریم‌ها و تصاویر موجود در مجموعه دادگان را می‌سنجد. اگر فاصله گفته شده کمتر از آستانه تعریف شده برای چهار دسته موجود باشد، فریم جز یکی از دسته‌های مدنظر است، در غیر این صورت فریم ورودی به طبقه‌بند خارج از

فصل ۴. الگوریتم پیاده‌سازی شده

حالات‌های تعریف شده در مجموعه دادگان جمع‌آوری شده برای این پژوهش است. در ادامه این دو شبکه به همراه نحوه طراحی و عملکرد آن‌ها توضیح داده می‌شود.

۱-۴-۴ آماده‌سازی مجموعه دادگان

در این قسمت یک شبکه عصبی کانولوشنی طراحی می‌گردد که توسط مجموعه دادگان جمع‌آوری شده آموزش می‌بیند و اعتبارسنجی می‌شود. پیش‌تر راجع به مجموعه دادگان توضیح داده شد و گفته شد که داده‌های آموزش و اعتبارسنجی به ترتیب ۵۱۲۰ و ۱۶۰۰ تصویر رنگی هستند. پیش از اعمال این تصاویر به شبکه، ابعاد همه تصاویر، اعم از دادگان آموزش و اعتبارسنجی، به ابعاد 70^*70^*3 تغییر داده شده است.

همچنین گفته شد مجموعه دادگان آموزش و اعتبارسنجی هم‌جنس نیستند. دادگان آموزش توسط دوربین رایانه و بدون هیچ نرم‌افزار واسطی جمع‌آوری شده‌اند. در حالی که تصاویر اعتبارسنجی از جنس تصاویری است که شبکه در مرحله اجرا با آن برخورد خواهد داشت. در واقع این تصاویر خروجی الگوریتم SSD است که در مرحله آزمایش به بخش طبقه‌بندی اعمال می‌گردد. به همین علت در مرحله اعتبارسنجی نیز از تصاویری که الگوریتم SSD تولید می‌کند، استفاده می‌شود. در واقع استفاده از توزیع متفاوت برای دادگان آموزش و اعتبارسنجی منجر به بهبود عملکرد شبکه و افزایش قدرت تعمیم آن می‌شود [۴۵].

لازم به ذکر است در راستای آموزش شبکه عصبی به منظور طبقه‌بندی، نمونه‌های موجود در مجموعه دادگان باید شامل برچسب باشند تا شبکه ارتباط بین نمونه‌ها را بر اساس خروجی مطلوب یاد بگیرد و به کمک همین روابط به دادگان جدید برچسب متناسب اختصاص دهد.

۲-۴-۴ آموزش شبکه عصبی کانولوشنی به منظور طبقه‌بندی

در این بخش با استفاده از مجموعه دادگانی که در قسمت قبل پیش‌پردازش^۴ شده‌اند، شبکه عصبی کانولوشن در جهت طبقه‌بندی آموزش می‌بیند. برای آموزش شبکه از ابزار یادگیری انتقالی استفاده شده است که در فصل مفاهیم و مقدمات پژوهش نیز توضیح داده شد. در این پژوهش از شبکه EfficientNet که در اواخر سال ۲۰۱۹ معرفی شد، استفاده شده است. شبکه EfficientNet نسبت به شبکه‌های کانولوشنی دیگر در حدود ۶ درصد

⁴Preprocessing

افزایش دقت داشته است. همچنین در حدود ۵ الی ۱۰ مرتبه نسبت به سایر شبکه‌ها کارآمدتر^۵ است [۳۹].

در واقع شبکه EfficientNet از سه بعد برای تغییر مقیاس شبکه استفاده می‌کند: عمق^۶ شبکه، پهنا^۷ شبکه و رزولوشن^۸ تصاویر ورودی. افزایش عمق شبکه یکی از رایج‌ترین روش‌های تغییر مقیاس شبکه است. شبکه عمیق ویژگی‌های پیچیده و گران‌بها تصاویر را بهتر استخراج می‌کند و قدرت تعمیم بیشتری نیز دارد [۳۹].

بیش‌تر شبکه‌های عصبی همانند ResNet برای بهبود شبکه از روش افزایش لایه استفاده می‌کنند که منجر به بهبود دقت می‌شود. البته افزودن لایه مشکلاتی همچون ناپدید شدن گرادیان^۹ را داشته و از یک جایی به بعد شبکه اشباع می‌گردد و امکان افزایش دقت بسیار کاهش می‌یابد؛ بنابراین افزودن لایه تغییر چندانی در عملکرد شبکه ایجاد نخواهد کرد و تنها با رفع محاسباتی است که افزایش می‌یابد. به عنوان مثال دقت دو شبکه ResNet152 و ResNet1000 نزدیک به هم است [۴۶].

افزایش پهنا نیز هنگامی استفاده می‌شود که اندازه شبکه مهم بوده و هدف کوچک ماندن آن است. از آنجایی که شبکه‌های کوچک راحت‌تر آموزش می‌بینند، بدیهی است که یک شبکه ایده‌آل دارای دقت بالا و ابعاد کوچک است تا علاوه بر داشتن عملکرد قابل قبول، به سرعت نیز آموزش بینند. اما استفاده از شبکه با پهنا بیش‌تر و عمق کم‌تر افزایش دقت را اشباع می‌کند و بهبود عملکرد آن تقریباً متوقف می‌گردد [۳۹].

در تصاویر با رزولوشن بالا ویژگی‌های جزئی^{۱۰} بهتر استخراج می‌گردند. به همین علت است که در تشخیص اشیا و شناسایی^{۱۱} آن‌ها از تصاویر با رزولوشن بالا استفاده می‌شود تا امکان ایجاد تمایز بین گروه‌های مختلف مربوط به یک مجموعه همچون انواع اتومبیل، گیاهان و ... فراهم شود. البته افزایش رزولوشن و تأثیر آن بر افزایش دقت شبکه نیز به صورت خطی نیست [۳۹].

ایده اصلی استفاده شده در شبکه EfficientNet ترکیب سه بعد گفته شده در راستای تغییر اندازه شبکه برای هم‌افزایی^{۱۲} دقت آن است؛ به گونه‌ای که برای افزایش دقت، ترکیبی از عمق، پهنا و رزولوشن افزایش می‌یابد. همچنین گفته شده است هر یک از ابعاد تغییر مقیاس، دارای یک مقدار بهینه در مقابل سایر ابعاد هستند [۳۹]. طبق گفته مقاله EfficientNet برای افزایش مقیاس شبکه باید ۲۰ درصد به عمق آن، ۱۰ درصد به پهنا آن

⁵More Efficient

⁶Depth

⁷Width

⁸Resolution

⁹Vanishing Gradient

¹⁰Fine-grained Features

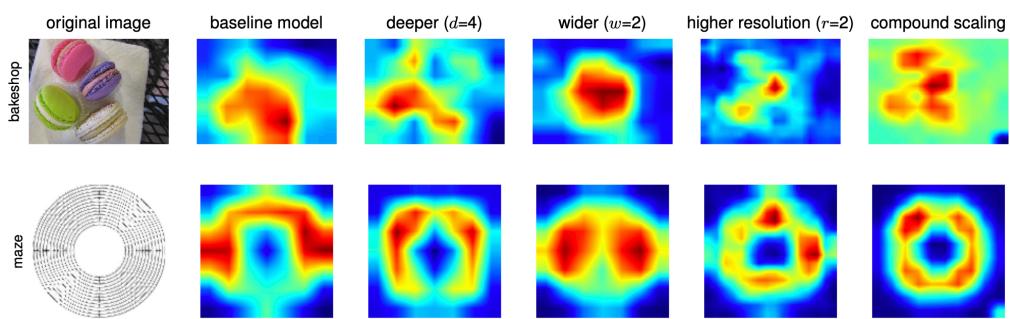
¹¹Recognition

¹²Synergy

فصل ۴. الگوریتم پیاده‌سازی شده

اضافه شود و ۱۵ درصد رزولوشن تصاویر بالا رود تا شبکه طراحی شده کارآمد باقی بماند. در واقع نسبت گفته شده بین ابعاد مختلف برای شبکه EfficientNet از B7 تا B0 که در آنها تعداد لایه‌ها زیاد شده است، یکسان است [۳۹].

در شکل ۵-۴ یک نقشه حرارتی^{۱۳} از مقاله EfficientNet ارائه شده است که نشان می‌دهد استفاده از مقیاس‌بندی در ابعاد گفته شده و ترکیب آن‌ها چه تأثیری در نحوه برجسته کردن اشیا موجود در یک تصویر دارد [۳۹].



شکل ۵-۴: نقشه حرارتی نمایان‌گر تأثیر ترکیب ابعاد مختلف بر برجسته‌سازی ویژگی‌های یک تصویر [۳۹]

در این پژوهش به منظور طراحی یک شبکه عصبی کانولوشنی از مدل EfficientNet-B0 استفاده شده است؛ زیرا طبق گفته مقاله برای مجموعه دادگان کوچک، استفاده از شبکه با عمق کم کافی است و مزیت سرعت بالا و بار محاسباتی کم را نیز خواهد داشت [۳۹].

تا اینجا خلاصه‌ای از عملکرد شبکه EfficientNet ارائه گردید. حال از این شبکه در راستای طراحی یک طبقه‌بند استفاده می‌شود. در ابتدا مجموعه دادگان پیش‌پردازش شده به عنوان ورودی به شبکه EfficientNet-B0 اعمال می‌گرددند. همان طور که پیش‌تر گفته شد، تصاویر ورودی به شبکه عصبی دارای ابعاد $70*70*3$ هستند. همچنین وزن‌های شبکه EfficientNet-B0 با استفاده از وزن‌های ImageNet مقداردهی اولیه^{۱۴} شده‌اند. در راستای تشکیل دادگان آزمایش در جهت بررسی دقت شبکه، ۵۰ درصد از نمونه‌های دادگان اعتبارسنجی به صورت تصادفی به عنوان دادگان آزمایش درنظر گرفته شده‌اند؛ بنابراین از ۱۶۰۰ نمونه موجود در مجموعه دادگان اعتبارسنجی، ۸۰۰ نمونه به دادگان آزمایش اختصاص داده شده و ۸۰۰ نمونه باقیمانده به منظور اعتبارسنجی شبکه استفاده شده است.

¹³Heatmap

¹⁴Initialization

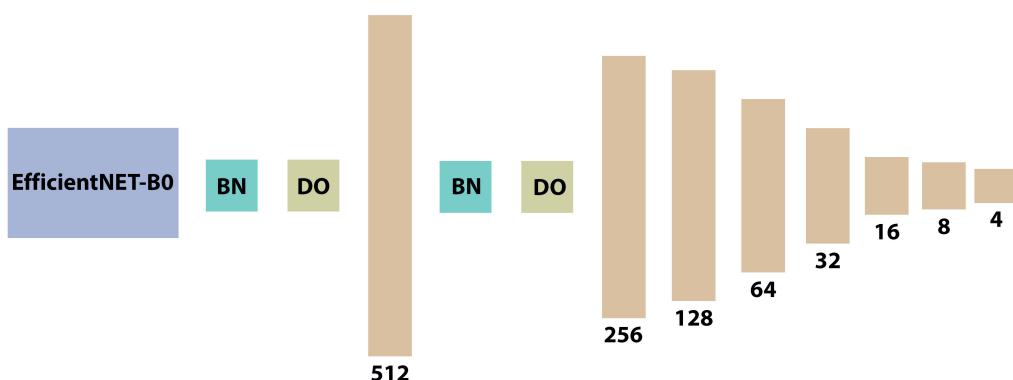
فصل ۴. الگوریتم پیاده‌سازی شده

آخرین لایه EfficientNet-B0 یک لایه با ابعاد $1*1*1280$ است. که پس از آن دو لایه و ۵۰ درصد Dropout برای کاهش پیچیدگی شبکه و افزایش قدرت تعمیم آن استفاده می‌شود. سپس یک لایه تماماً متصل با ۵۱۲ نورون نیز اضافه می‌گردد.

تابع فعال‌ساز استفاده شده برای این لایه یک تابع جدید به نام Swish است که در مقاله EfficientNet به آن معرفی شده است. اگرچه معمولاً در شبکه‌های کانولوشنی از تابع فعال‌ساز ReLU استفاده می‌شود، اما این تابع در مقادیر نامثبت، صفر است و گرادیان آن نیز برای این مقادیر صفر خواهد بود. حال که استفاده از تابع فعال‌ساز Swish در معماری EfficientNet و بر روی مجموعه دادگان ImageNet در حدود ۰.۶ درصد دقت را بهبود می‌بخشد. در رابطه ۴-۲ تابع فعال‌ساز Swish معرفی شده است. در واقع منظور از x تابع همانی است. همچنین سیگموئید یا sigmoid یک تابع ریاضی است که مقادیر ورودی خود را به اعدادی بین صفر تا یک نگاشته می‌کند.

$$swish(x) = x * \text{sigmoid}(x) \quad (4-2)$$

پس از لایه تماماً متصل با ۵۱۲ نورون، مجدداً از Batch Normalization و ۵۰ درصد Dropout استفاده شده است. سپس یک لایه با ۲۵۶ نورون، تابع فعال‌ساز Swish و Batch Normalization قرار گرفته است. سپس ۵ لایه تماماً متصل دیگر به همراه تابع فعال‌ساز Swish قرار داده شد. در آخر نیز از یک لایه تماماً متصل با ۴ نورون و تابع Softmax استفاده گردید. در واقع تابع Softmax یک توزیع احتمالاتی به دسته‌های مختلف تخصیص می‌دهد که تصویر ورودی چند درصد متعلق به هر دسته است و بیشترین احتمال، تصویر را از آن خود می‌کند. در شکل ۴-۶ لایه‌های اضافه شده به شبکه EfficientNet-B0 قابل مشاهده است.



شکل ۴-۶: لایه‌های اضافه شده به شبکه EfficientNet

فصل ۴. الگوریتم پیاده‌سازی شده

لازم به ذکر است شبکه در حدود ۴.۸۸۷ میلیون پارامتر داشته که ۴.۸۴۲ میلیون از آن‌ها پارامترهای قابل آموزش شبکه هستند. در نهایت شبکه با استفاده از مجموعه دادگان آموزش و اعتبارسنجی، آموزش دید و وزن‌های آن ذخیره گردید. برای آموزش شبکه از بهینه‌ساز^{۱۵} Adam با نرخ یادگیری^{۱۶} ۰.۰۰۱ استفاده شده است. شبکه در ۲۰ اپیک^{۱۷} آموزش دیده است و در صورتی که دقت دادگان اعتبارسنجی در دو اپیک متواالی بهبود نیابد، با نرخ ۵۰ درصد کاهش می‌یابد. در نهایت شبکه طبقه‌بند برای ۴ دسته مدنظر بر روی دادگان آزمایش، که شبکه آن‌ها را مشاهده نکرده است، به دقت ۹۹ درصد رسید.

در الگوریتم ۲-۴ نحوه آموزش طبقه‌بند طراحی شده در این پژوهش، آمده است. شبکه عصبی کانولوشنی با استفاده از روش یادگیری انقالی، شبکه EfficientNet-B0 و لایه تماماً متصل طراحی گردید. این شبکه با مجموعه دادگان پیش‌پردازش شده نیز آموزش داده شد و وزن‌های آن ذخیره گردید.

در نهایت با اعمال تصویر جدید به این شبکه و وزن‌های ذخیره شده آن، می‌توان برچسب تصویر ورودی جدید را پیش‌بینی نمود. لازم به ذکر است یکی از خروجی‌های الگوریتم SDD فریم بریده شده و دارای دست بود که به بخش طبقه‌بندی برای تعیین برچسب آن اعمال می‌گردد. البته پیش از اعمال فریم بریده شده به طبقه‌بند، لازم است تعلق تصویر به مجموعه دادگان مدنظر بررسی شود که در مرحله بعد به آن پرداخته می‌شود.

همان طور که پیش‌تر گفته شد، دادگان آموزش توسط افراد مختلف جمع‌آوری شده است ولی دادگان اعتبارسنجی با استفاده از الگوریتم SDD تهیه شده‌اند. از میان دادگان اعتبارسنجی نیز تنها ۵۰ درصد از آن‌ها در آموزش شبکه طراحی شده، دخالت داشته‌اند و ۵۰ درصد باقی مانده به عنوان دادگان آزمایش استفاده شده‌اند.

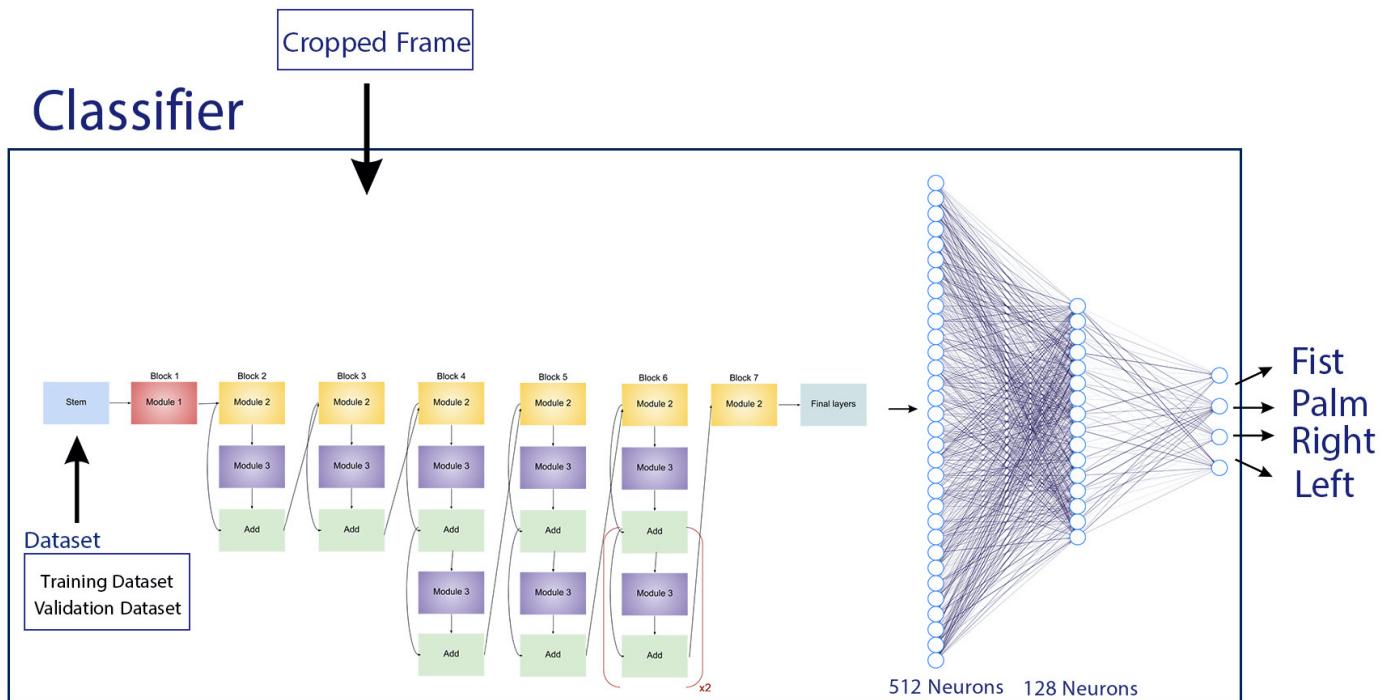
تا به اینجا با استفاده از مجموعه دادگان جمع‌آوری شده یک شبکه عصبی کانولوشن آموزش داده شد؛ بنابراین در صورتی که تصویر خروجی الگوریتم SDD عضو دسته‌های تعریف شده در مجموعه دادگان باشد، برچسب متناسب با آن پیش‌بینی می‌شود. پیش‌تر نیز اشاره شد که طبقه‌بند مبتنی بر شبکه‌های کانولوشن تنها بر مجموعه دادگان بسته به خوبی عمل می‌کند؛ بنابراین شبکه طراحی شده تنها در صورتی که فریم‌های وارد به آن جز یکی از دسته‌های تعریف شده در مجموعه دادگان باشد به درستی عمل می‌کند. در بخش بعدی راه حلی برای این موضوع مطرح خواهد شد تا پیش از اعمال تصویر به طبقه‌بند، عضویت حالت متناظر با آن در مجموعه دادگان مدنظر بررسی گردد.

¹⁵Optimizer

¹⁶Learning Rate

¹⁷Epoch

الگوریتم ۲-۴ طبقه‌بند طراحی شده با استفاده از شبکه‌های عصبی کانولوشنی



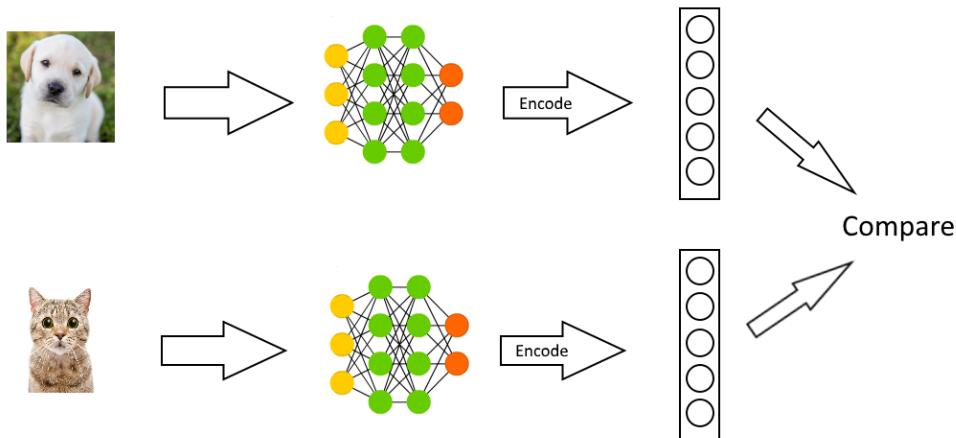
۳-۴-۴ طراحی شبکه کانولوشن به منظور شباهت سنجی

تا به اینجا مسئله طبقه‌بندی مجموعه دادگان جمع‌آوری شده و طراحی یک شبکه در راستای تخصیص برچسب به دادگان جدید حل گردید. همان طور که پیش‌تر عنوان شد، شبکه طراحی شده در زمان آزمایش ممکن است با تصویری مواجهه شود که به هیچ‌کدام از دسته‌های موجود در مجموعه دادگان، متعلق نیست. در آن صورت تابع فعال‌ساز Softmax همچنان تصویر را به یکی از ۴ دسته اختصاص می‌دهد؛ زیرا این تابع مناسب توزیع‌های احتمالاتی بسته است..

تشخیص دسته‌های ناخواسته که در ازای آن‌ها نباید عملی صورت بگیرید، ضروری است. در واقع یکی از مشکلات یادگیری عمیق حذف حالات‌های ناخواسته در مجموعه دادگان است؛ زیرا برای حل این مشکل لازم است تعداد زیادی تصویر از حالات‌های ناخواسته نیز در نظر گرفته شود، شبکه با مجموعه دادگانی که حالات‌های ناخواسته و ناخواسته دارد، آموزش بیند و در هنگام تخصیص عمل به حالات‌های مدنظر، به حالات‌های ناخواسته عملی منصوب نشود.

فصل ۴. الگوریتم پیاده‌سازی شده

جمع‌آوری داده برای حالت‌های مدنظر کاری طاقت‌فرسا است و جمع‌آوری داده برای حالت‌های ناخواسته سختی کار را چندین برابر می‌کند؛ زیرا معمولاً تعداد حالت‌های ناخواسته چندین برابر حالت‌های مدنظر و جمع‌آوری آن‌ها سخت‌تر است. به عنوان مثال در این پژوهش که حالت‌های دست در راستای کنترل ماوس درنظر گرفته شده است، جمع‌آوری دادگان برای همه حالت‌های دست به دلیل درجه آزادی بالای حرکات آن دشوار است. استفاده از شبکه‌های One-shot Learning برای حل مسئله حالت‌های ناخواسته مرسوم است. در واقع این شبکه‌ها همچون شبکه عصبی Siamese به گونه‌ای آموزش نمی‌بینند که تصاویر را طبقه‌بندی کنند بلکه یک تابع شباهت^{۱۸} را یاد می‌گیرند تا بین دو تصویر میزان شباهت را محاسبه کنند. در شکل ۷-۴ نحوه عملکرد شبکه Siamese ارائه شده است [۴۷].



شکل ۷-۴: خلاصه نحوه عملکرد شبکه عصبی Siamese [۴۷]

در شبکه‌های عصبی مبتنی بر شباهت حجم مجموعه دادگان مسئله نبوده و برای هر دو تصویری می‌توان از آن‌ها استفاده کرد. برای استفاده از این شبکه‌ها لازم است یک شبکه عصبی کانولوشن توسط مجموعه دادگان مشابه آموزش بینید و وزن‌های آن ذخیره گردد. سپس دو تصویر مدنظر جداگانه به شبکه کانولوشن اعمال گرددند تا تصاویر به صورت بردار، البته با ابعاد کمتر، رمزنگاری^{۱۹} شوند. حال دو بردار به دست آمده مقایسه می‌شوند و با استفاده از یک آستانه تعیین شده، همدسته بودن یا نبودن دو تصویر رمزنگاری شده، مشخص می‌گردد.

در این پژوهش نیز از شبکه عصبی مبتنی بر شباهت استفاده شده است تا راه حلی برای حذف حالت‌های

¹⁸Similarity

¹⁹Encode

ناخواسته باشد. ابتدا لازم است تعریفی برای حالت‌های ناخواسته که در راستای کنترل نشان‌گر رایانه ارائه گردد. در مجموعه دادگان مورد استفاده در این پژوهش از ۴ حالت دست تعریف شده است که پیش‌تر نیز توضیح داده شد. هر حالت دیگر خارج از این ۴ دسته، ناخواسته بوده و باید در دسته حالت‌های ناخواسته قرار گیرد. لازم به ذکر است حالت‌های ناخواسته می‌توانند شامل دست یا حتی بدون دست باشد. به عنوان مثال اگر تصویر شامل دستی باشند که اشاره به بالا را نشان می‌دهد یا حتی یک پس‌زمینه خالی باید در دسته حالت‌های ناخواسته قرار گیرد. در واقع می‌توان برای حالت‌های ناخواسته نیز دو حالت درنظر گرفت:

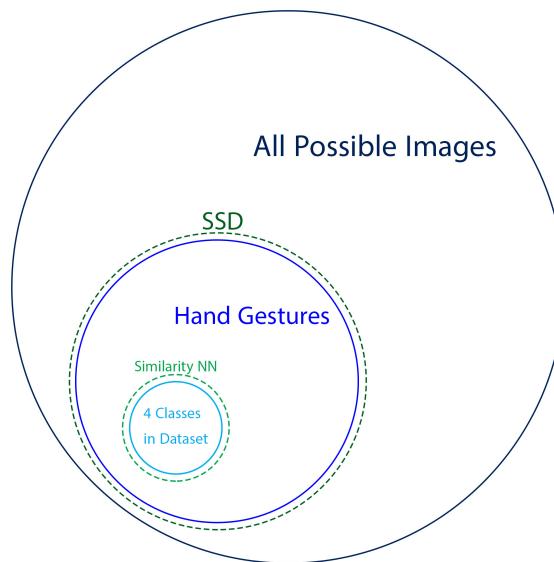
- حالت اول: تصویر بدون دست باشد.

- حالت دوم: دست در تصویر باشد اما جز ۴ دسته تعریف شده نباشد.

حالت اول که می‌توانند شامل تصاویر خود فرد، پس‌زمینه و ... در حالی که دستی قابل مشاهده نیست، باشد با استفاده از الگوریتم SSD جدا شده‌اند؛ زیرا پیش‌تر نیز گفته شد که فریم‌های خروجی از این الگوریتم لزوماً شامل تصویر دست هستند. در بخش تشخیص دهنده به این موضوع اشاره گردید که در صورت وجود دست، SSD اطراف آن یک کادر قوار می‌دهد تا ناحیه دست را جدا کند. حال اگر فریم فاقد دست باشد، الگوریتم فریم جدید را گرفته و پیش‌پردازش می‌کند.

بنابراین تنها در صورتی که تصویر خروجی الگوریتم تشخیص دهنده شامل دست باشد، به عنوان ورودی به بخش طبقه‌بندی وارد می‌شود. حال لازم است بین حالت‌های تعریف شده برای دست و سایر حالت‌های آن تمایز قائل شد که حالت دوم تعریف شده در بالا است. در شکل ۸-۴ حالت‌های ممکن برای فریم ورودی به الگوریتم کنترل حرکت ماوس و نحوه مقابله با حالت‌های ناخواسته ارائه شده است. برای حل مسئله تفکیک حالت‌های تعریف شده در مجموعه دادگان و سایر حالت‌های دست از یک شبکه عصبی مبتنی بر محاسبه شباهت استفاده شده است که در ادامه معرفی می‌گردد.

در این پژوهش برای حذف حالت‌هایی از دست که ناخواسته هستند، یک شبکه عصبی در راستای محاسبه شباهت بین تصویر در مرحله اجرای برنامه و تصاویر مجموعه دادگان جمع‌آوری شده، طراحی گردید. برای طراحی این شبکه از همان شبکه عصبی کانولوشن که در بخش طبقه‌بند طراحی شد، استفاده شده است؛ بدین صورت که پس از ذخیره وزن‌های شبکه کانولوشن، لایه‌های آخر تماماً متصل حذف شدن و خروجی قسمت EfficientNet به عنوان رمزنگار استفاده شده است. در واقع با استفاده از این شبکه رمزنگار، تصاویر ورودی به بردار ۱۲۸۰ درایه‌ای تبدیل می‌شوند. پیش‌تر نیز اشاره گردید که ابعاد تصاویر ورودی به صورت $70*70*3$ است.



شکل ۴-۸: انواع تصاویر ممکن جهت اعمال به شبکه و ابزار تفکیک حالت‌های ناخواسته از حالت‌های مدنظر

پس از تشکیل یک شبکه رمزنگار که در لایه خروجی 1280×1280 نورون دارد، تصاویر هر دسته از مجموعه دادگان، هر دو بخش آموزش و اعتبارسنجی، وارد این شبکه شده و بردار خروجی که دارای 1280×1 درایه است، ذخیره گردید. بدیهی است به تعداد مجموعه دادگان (6720)، بردار با 1280×1 درایه تشکیل می‌شود.

پس از تشکیل بردارهای رمزنگار برای همه تصاویر مجموعه دادگان، میانگین بردارهای تشکیل شده برای هر دسته که در حدود 1670×1 تصویر هستند، محاسبه شد. سپس این بردارها به عنوان بردار مرجع ذخیره گردید. بدیهی است برای هر دسته یک بردار مرجع وجود داشته و چهار بردار مرجع تشکیل خواهد شد.

برای محاسبه فاصله بین تصاویر و در نهایت تعیین یک آستانه برای هر دسته، از فاصله اقلیدسی^{۲۰} استفاده شده که در رابطه $3-4$ معرفی شده است. منظور از x_1 تا x_{1280} درایه‌های بردار x تصویر رمزنگاری شده است. از طرفی بردار m همان بردارهای مرجع بوده که توسط دادگان اعتبارسنجی ساخته شده‌اند و دارای درایه‌های m_1 تا m_{1280} هستند.

$$d(x, m) = \sqrt{(x_1 - m_1)^2 + (x_2 - m_2)^2 + \dots + (x_{1280} - m_{1280})^2} \quad (3-4)$$

²⁰Euclidean Distance

فصل ۴. الگوریتم پیاده‌سازی شده

در نتیجه با استفاده از فاصله اقلیدسی و بردارهای مرجع برای هر دسته، فاصله تصاویر مربوط به دادگان اعتبارسنجی نسبت به بردار مرجع مربوط به دسته هر تصویر، محاسبه شد که در جدول ۱-۴ ارائه شده است.

جدول ۱-۴: فاصله بین تصاویر دادگان اعتبارسنجی و بردار مرجع معرف هر دسته

دسته	کمترین فاصله اقلیدسی	بیشترین فاصله اقلیدسی
مشت	۲۲	۳۴.۹
کف دست	۱۸.۹۵	۳۴.۲۹
اشاره به راست	۲۸.۲۵	۴۵.۵
اشاره به چپ	۲۱	۴۸

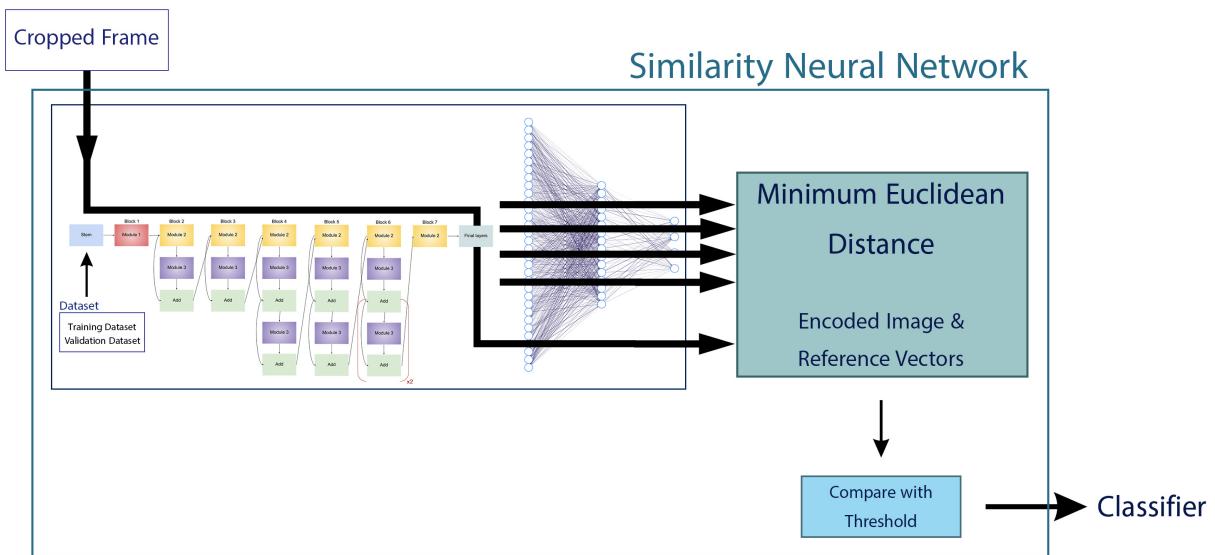
بیشترین فاصله بین تصاویر هر دسته و بردار مرجع مربوطه، به عنوان آستانه درنظر گرفته شده است. لازم به ذکر است در فضای ویژگی داده‌های مربوط به هر دسته در نزدیکی هم و با فاصله از دسته‌های دیگر قرار می‌گیرند؛ بنابراین با تعریف یک آستانه برای هر دسته می‌توان دادگان مدنظر و ناخواسته را تفکیک کرد. بدین صورت که هنگام اجرای الگوریتم ارائه شده در این پژوهش، وقتی فریم بریده شده از الگوریتم SSD وارد بخش طبقه‌بندی می‌شود، ابتدا نسبت به بردارهای مرجع هر دسته سنجیده می‌شود و کمترین فاصله از بردار مرجع انتخاب می‌گردد.

پس از انتخاب کمترین فاصله از بردار مرجع این احتمال وجود دارد که فریم اعمال شده به شبکه مربوط به دسته‌ای باشد که فاصله از بردار مرجع آن کمترین شده است. در این حالت فاصله با آستانه درنظر گرفته شده برای آن دسته سنجیده می‌شود.

اگر فاصله تصویر رمزنگاری شده از بردار مرجع، کمتر از آستانه باشد، تصویر مربوط به حالت‌های مدنظر است و برچسب پیش‌بینی شده توسط طبقه‌بند دارای اعتبار است. در صورتی که کمترین فاصله تصویر از بردارهای مرجع بیشتر از آستانه باشد، تصویر جز حالت‌های ناخواسته است و الگوریتم SSD فریم جدید را دریافت می‌کند.

فصل ۴. الگوریتم پیاده‌سازی شده

الگوریتم ۳-۴ تفکیک حالت‌های ناخواسته با استفاده از یک شبکه عصبی مبتنی بر شباهت‌سنجد



الگوریتم ۳-۴ چگونگی عملکرد شبکه شباهت‌سنجد و نحوه تفکیک دادگان ناخواسته را نشان می‌دهد. از آنجایی که تشخیص دهنده دست SSD تنها فریم‌های شامل دست را به شبکه شباهت اعمال می‌کند، تصاویر ناخواسته که فاقد دست هستند، در این مرحله حذف می‌شوند. حال فریم‌های شامل دست که از ناحیه دست بریده شده‌اند، علاوه بر طبقه‌بند به شبکه عصبی شباهت نیز وارد می‌شوند. شبکه عصبی شباهت‌سنجد تصویر را رمزگذاری کرده و از آن یک بردار با 128^0 درایه می‌سازد.

سپس فریم رمزگذاری شده با 4^0 بردار مرجع که معرف ۴ دسته تعریف شده در مجموعه دادگان هستند، مقایسه می‌گردد و بردار مرجعی که کمترین فاصله را از فریم رمزگذاری شده دارد، انتخاب می‌گردد. حال فاصله بردار مرجع انتخاب شده و فریم رمزگاری شده، با آستانه تعیین شده برای دسته متناظر با بردار مرجع انتخاب شده، مقایسه می‌گردد. در صورتی که فاصله فریم رمزگاری شده و بردار مرجع، از آستانه کمتر باشد، فریم بریده شده وارد طبقه‌بند مبتنی بر شبکه‌ها عصبی کانولوشن می‌شود تا برچسب آن پیش‌بینی گردد. حال اگر فاصله مذکور بیشتر از آستانه دسته بردار مرجع انتخابی باشد، به الگوریتم SSD گفته می‌شود یک فریم جدید دریافت کند.

پیش‌تر نیز عملکرد شبکه عصبی کانولوشن که برچسب تصاویر ورودی را تعیین می‌کند، بررسی شد. در واقع فریم بریده شده که خروجی الگوریتم SSD است، ابتدا چک می‌شود که مربوط به دادگان مدنظر برای کنترل

نshan گر ماوس هست یا نه و در صورت مربوط بودن، برچسب خروجی شبکه طبقه‌بند معتبر است. لازم به ذکر است استفاده از شبکه شباهت برای تعیین برچسب نیز ممکن بوده و هنگام انتخاب مرجعی که کمترین فاصله را از تصویر رمزگاری شده دارد و بررسی آستانه، دسته آن مرجع به عنوان برچسب انتخاب شود؛ اما احتمال خطأ در این روش به دلیل همپوشانی دسته‌ها وجود دارد؛ به همین علت، همزمان با شبکه شباهت‌سنج، فریم خروجی از الگوریتم SSD وارد طبقه‌بند نیز شده و پس از عبور از شبکه شامل EfficientNet-B0 و لایه‌های تماماً متصل به یکی از چهار دسته مدنظر با پیش‌ترين احتمال نگاشته می‌شود.

۵-۴ کنترل نshan گر با اشارات پیش‌بینی شده

تا به اینجا گفته شد که با استفاده از دوربین لپ‌تاپ، یک فریم وارد الگوریتم SSD وجود یا عدم وجود دست در این فریم بررسی می‌شود. همچنین گفته شد در صورت وجود دست در فریم دریافتی، الگوریتم SSD برای بخش‌های بعد دو خروجی آماده می‌کند. خروجی اول فریم شامل دست است که از قسمت کادر محدود (ناحیه دست) بریده شده است و به بخش طبقه‌بندی وارد می‌شود. خروجی دوم که الگوریتم SSD می‌سازد، در واقع مختصات نقطه مرکزی فریم خروجی تشخیص دهنده است که وارد بخش کنترل‌کننده نshan گر می‌شود.

همچنین مشاهده گردید بخش طبقه‌بندی پس از دریافت فریم بریده شده، تنها در صورتی که آن فریم مربوط به دسته‌های تعریف شده در مجموعه دادگان باشد، یک برچسب دارای اعتبار به آن اختصاص می‌دهد. حال در این قسمت از پژوهش، با استفاده از بخش‌های قبلی و خروجی‌های آن‌ها، در راستای کنترل نshan گر ماوس، به برچسب‌های پیش‌بینی شده توسط طبقه‌بند وظایفی اختصاص داده می‌شود. در ادامه هر یک از برچسب‌ها و وظایفی که به آن‌ها تخصیص داده شده است، توضیح داده می‌شود.

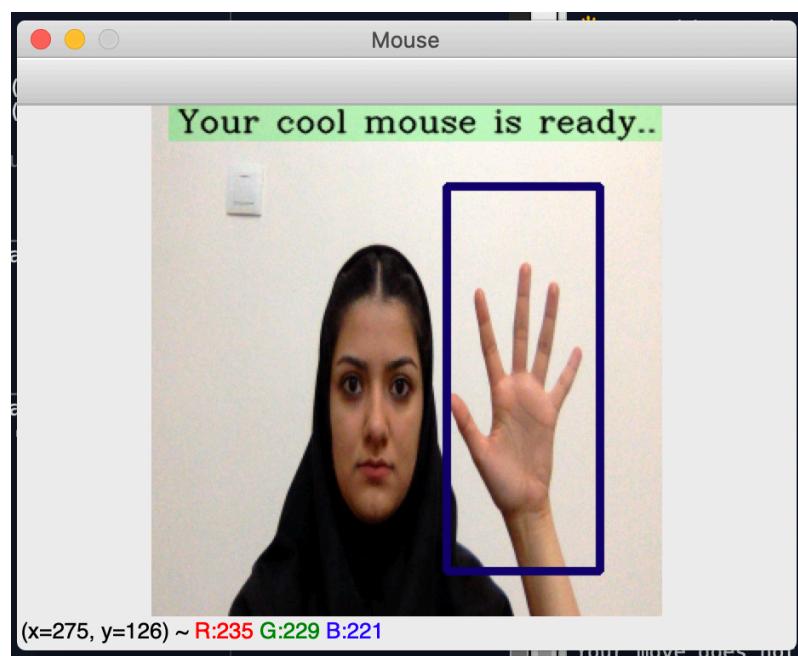
۱-۵-۴ حالت کف دست: شروع و جابه‌جایی نshan گر

وظیفه روشن کردن ماوس طراحی شده و همچنین جابه‌جایی نshan گر، به حالت کف دست اختصاص داده شده است؛ به گونه‌ای که پس از اجرای الگوریتم و نshan دادن دست توسط کاربر یک صفحه باز می‌شود که تصویر محدوده جلو دوربین را به صورت بلاذرنگ نمایش می‌دهد. همچنین به دور دست کاربر، توسط الگوریتم SSD یک کادر آبی گذاشته می‌شود. البته در این مرحله فقط دست فرد شناسایی شده و ماوس همچنان خاموش است.

فصل ۴. الگوریتم پیاده‌سازی شده

در راستای روشن کردن ماوس طراحی شده لازم است تا کاربر کف دست خود را در جلو دوربین قرار دهد تا ماوس شروع به کار کند و تا قبل از آن که کاربر کف دست خود را به دوربین نشان دهد، هر حرکت دیگری تشخیص داده نمی‌شود. در بالای این صفحه جدید نیز یک قادر سبز نشان‌دهنده آمادگی ماوس برای کنترل توسط دست قرار داده شده است.

در واقع وقتی کاربر کف دست خود را به دوربین نشان می‌دهد، الگوریتم SSD دست او را تشخیص داده، به دور آن قادر می‌گذارد و فریم بریده شده را به طبقه‌بند می‌دهد. حال طبقه‌بند به فریم بریده شده برچسب کف دست را اختصاص می‌دهد. از اینجا به بعد ماوس طراحی شده روشن بوده و فرد می‌تواند توسط آن نشان‌گر را کنترل کند که در شکل ۹-۴ نیز قابل مشاهده است. اعداد نمایش داده شده در گوش سمت چپ شکل ۹-۴، مختصات نشان‌گر بر روی صفحه باز شده و پارامترهای رنگی آن مختصات را نمایش می‌دهد. لازم به ذکر است اندازه تصویر نمایش کف دست توسط کاربر در شکل ۹-۴ تغییر یافته است و نسبت این تغییر اندازه برای محورهای x و y یکسان نبوده که منجر به کشیده شده تصویر از بالا و پایین شده است.



شکل ۹-۴: روشن کردن ماوس طراحی شده و جابه‌جایی نشان‌گر با استفاده از کف دست

علاوه بر منصب کردن حالت کف دست به روشن کردن ماوس طراحی شده، وظیفه جابه‌جایی نشان‌گر واگذار شده است و در بالای صفحه یک نوار بنفس نشان‌دهنده حالت ردیابی^{۲۱} یا همان جابه‌جایی نشان‌گر قرار داده

²¹Tracking Mode

می‌شود. نحوه عملکرد این بخش بدین گونه است که پس از تشخیص حالت کف دست توسط طبقه‌بند، به بخش کنترل کننده گفته می‌شود که مختصات مرکز فریم بریده شده را از SSD دریافت کند. حال لازم است مختصات مرکز فریم بریده شده به مختصات نشان‌گر تبدیل شود؛ بنابراین با استفاده از جایه جایی دست، این مختصات تغییر می‌کند که منجر به تغییر مکان نشان‌گر نیز می‌شود.

از آنجایی که مختصات فریم بریده شده توسط الگوریتم SSD با رزولوشن ۳۰۰ در ۳۰۰ است، لازم است این مختصات به رزولوشن سیستمی که کنترل کننده نشان‌گر بر آن اجرا می‌شود، تبدیل گردد. به عنوان مثال برای سیستمی با رزولوشن تصویر ۱۴۴۰ در ۹۰۰ لازم است تا مختصات فریم بریده شده به صورت زیر تغییر یابد.

$$x_{sys} = \frac{1440}{300}x_{center}, y_{sys} = \frac{900}{300}y_{center} \quad (4-4)$$

بدیهی است در راستای پوشش همه مختصات‌های صفحه نمایش توسط نشان‌گر، لازم است مختصات مربوط به الگوریتم SSD به مختصات صفحه نمایش تبدیل گردد که در رابطه ۴-۴ ارائه گردید. منظور از x_{center} و y_{center} مختصات نقطه مرکزی فریم بریده شده است که به مختصات متناظر خود در صفحه نمایش یعنی x_{sys} و y_{sys} تبدیل شده‌اند.

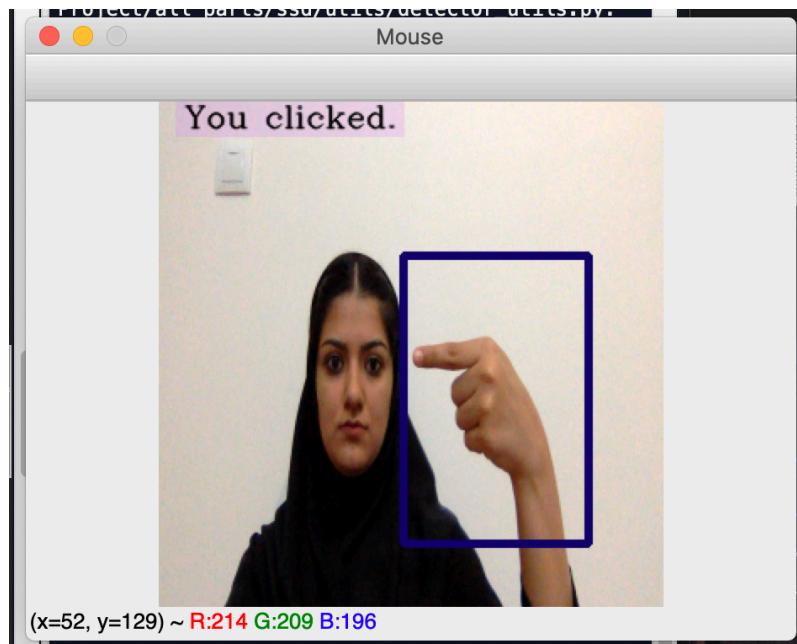
در مرحله بعد، مختصات فعلی نشان‌گر با مختصات تبدیل شده، کالیبره می‌شود تا نشان‌گر مختصات فعلی خود را حفظ کند و در صورت جایه‌جا شدن کف دست کاربر، مختصات نشان‌گر نسبت به حالت قبلی خود تغییر کند. در نهایت مختصات تبدیل یافته به نشان‌گر اعمال می‌گردد تا نشان‌گر در راستای تغییر مکان کف دست، جایه‌جا شود.

۲-۵-۴ حالت اشاره به چپ: کلیک کردن

یکی از حالت‌های تعریف شده در مجموعه دادگان اشاره به چپ بوده که وظیفه کلیک کردن به آن داده شده است؛ به گونه‌ای که پس از روشن شدن ماوس طراحی شده، اگر کاربر حالت اشاره به چپ را جلو دوربین اجرا کند، در محل فعلی نشان‌گر یک کلیک زده می‌شود. از آنجایی که ماوس طراحی شده، ۱۵ فریم بر ثانیه کار می‌کند، تا زمانی که کاربر حالت دست خود را تغییر دهد، چندین بار به طور متوالی کلیک زده می‌شود؛ بنابراین برای

فصل ۴. الگوریتم پیاده‌سازی شده

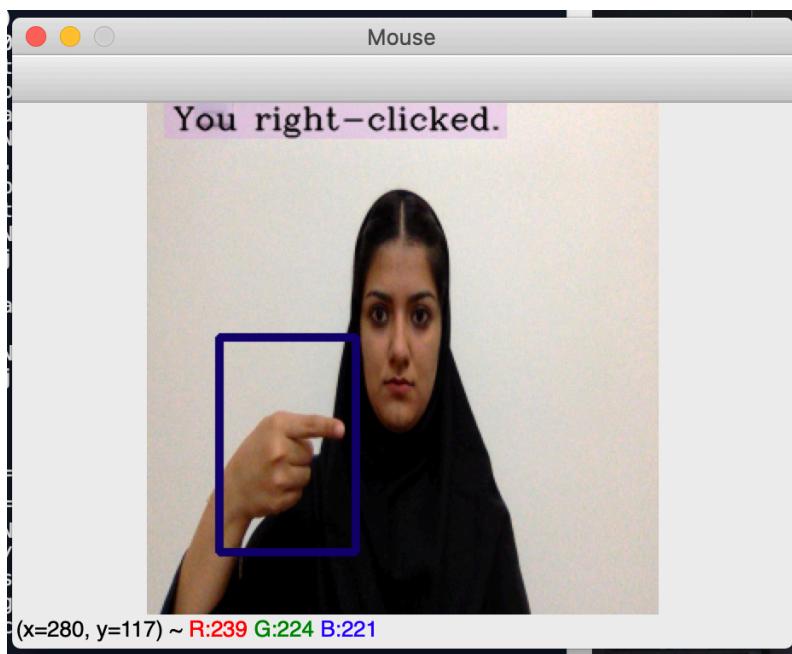
جلوگیری از این مسئله، ۵.۰ ثانیه به کاربر فرصت داده می‌شود تا حالت دست خود را تغییر دهد. در شکل ۱۰-۴ نحوه کلیک کردن در ماوس طراحی شده آمده است. همچنین هنگامی که کاربر به کلیک کردن اشاره می‌کند، در بالای صفحه یک نوار بنفس رنگ به همراه متن شما کلیک کردید، نشان داده می‌شود که در شکل ۱۰-۴ نیز قابل مشاهده است.



شکل ۱۰-۴: نحوه کلیک کردن توسط ماوس طراحی شده

۳-۵-۴ حالت اشاره به راست: راست‌کلیک کردن

یکی دیگر از حالتهای تعریف شده در مجموعه دادگان جمع‌آوری شده، حالت اشاره به راست است که از آن برای راست‌کلیک کردن استفاده می‌شود؛ به گونه‌ای که اگر کاربر با دست خود حالت اشاره به راست را نشان دهد، در محل فعلی نشان‌گر راست‌کلیک زده می‌شود. در این قسمت نیز مشکل، راست‌کلیک‌های متوالی تازمان تغییر حالت دست وجود دارد که به فرد فرصت ۵.۰ ثانیه داده می‌شود. همچنین هنگامی که کاربر، راست‌کلیک می‌کند، در بالای صفحه یک نوار بنفس نمایان‌گر شما راست‌کلیک کردید، قرار داده شده است. در شکل ۱۱-۴ نحوه راست‌کلیک کردن توسط ماوس طراحی شده، ارائه گردیده است.



شکل ۴-۱۱: نحوه راست‌کلیک کردن توسط ماوس طراحی شده

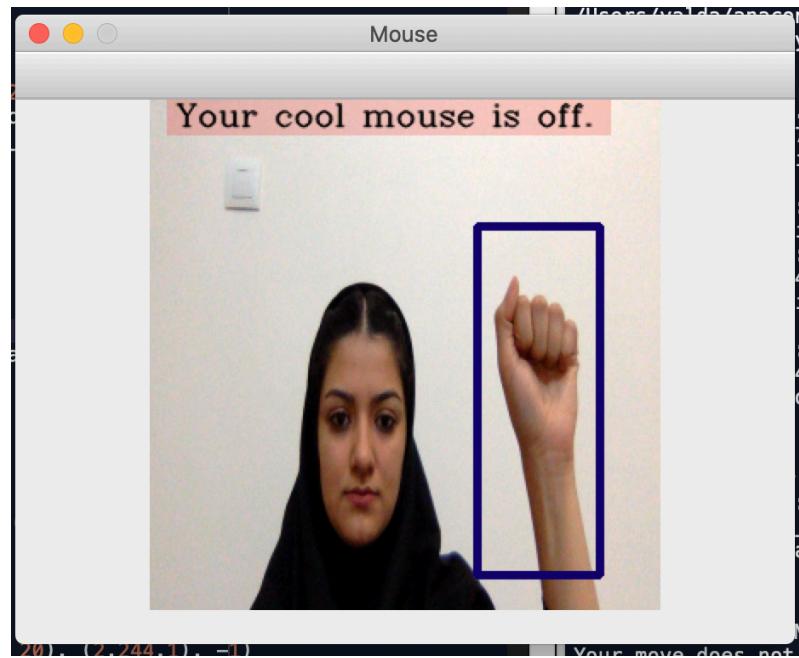
۴-۵-۴ حالت مشت: خاموش کردن

حالت چهارم تعریف شده در مجموعه دادگان، دست در حالت مشت بود که در بخش کنترل‌کننده عمل خاموش کردن ماوس به آن اختصاص داده شده است؛ بنابراین اگر بخش طبقه‌بندی برچسب مشت را برای فریم بریده شده پیش‌بینی کند، در بخش کنترل‌کننده نشان‌گر عمل خاموش شدن ماوس انجام می‌شود و بعد از ۳ ثانیه ماوس خاموش می‌گردد. همچنین در بالای صفحه نمایش‌گر فرد و دست او، یک نوار قرمز نمایان‌گر خاموش شدن ماوس نمایش داده می‌شود. در شکل ۴-۱۲ فرمان لازم برای خاموش شدن و نحوه عملکرد ماوس طراحی شده آمده است.

در الگوریتم ۴-۴ نیز نحوه عملکرد بخش کنترل‌کننده ارائه شده است. همان‌طور که پیش‌تر نیز اشاره شد، برچسب پیش‌بینی شده توسط طبقه‌بند، در صورتی که فریم شباهت قابل قبولی با دادگان اعتبارسنجی داشته باشد، به بخش کنترل‌کننده اعمال می‌شود. همچنین گفته شد در صورتی که شباهت فریم بریده شده با بردارهای مرجع ساخته شده توسط تصاویر مجموعه دادگان اعتبارسنجی کافی نباشد، به الگوریتم SSD گفته می‌شود که یک فریم جدید دریافت کند.

پس از وارد شدن برچسب پیش‌بینی شده به بخش کنترل‌کننده، دو حالت ممکن است اتفاق بیفتد:

فصل ۴. الگوریتم پیاده‌سازی شده



شکل ۱۲-۴: خاموش کردن ماوس طراحی شده با استفاده از دست کاربر در حالت مشت

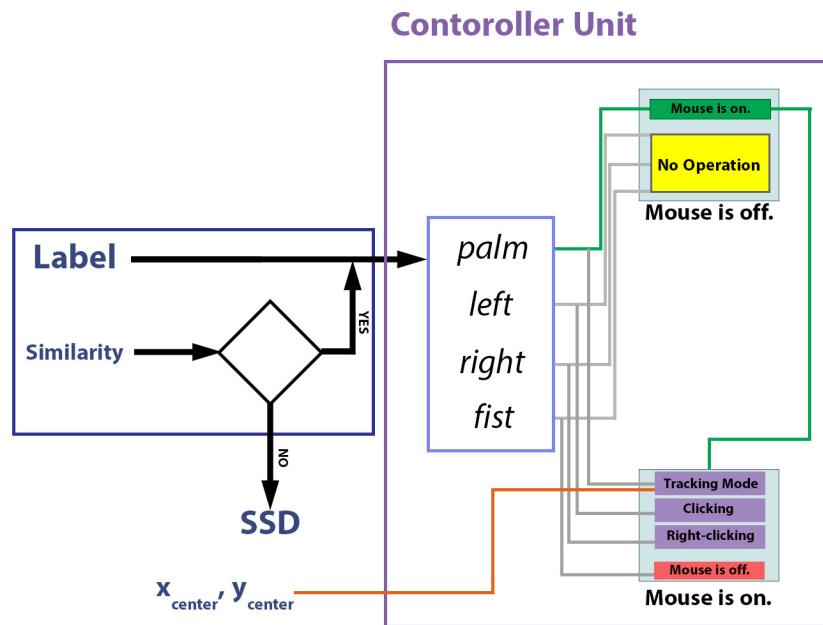
۱. ماوس طراحی شده خاموش باشد.

۲. ماوس طراحی شده روشن باشد.

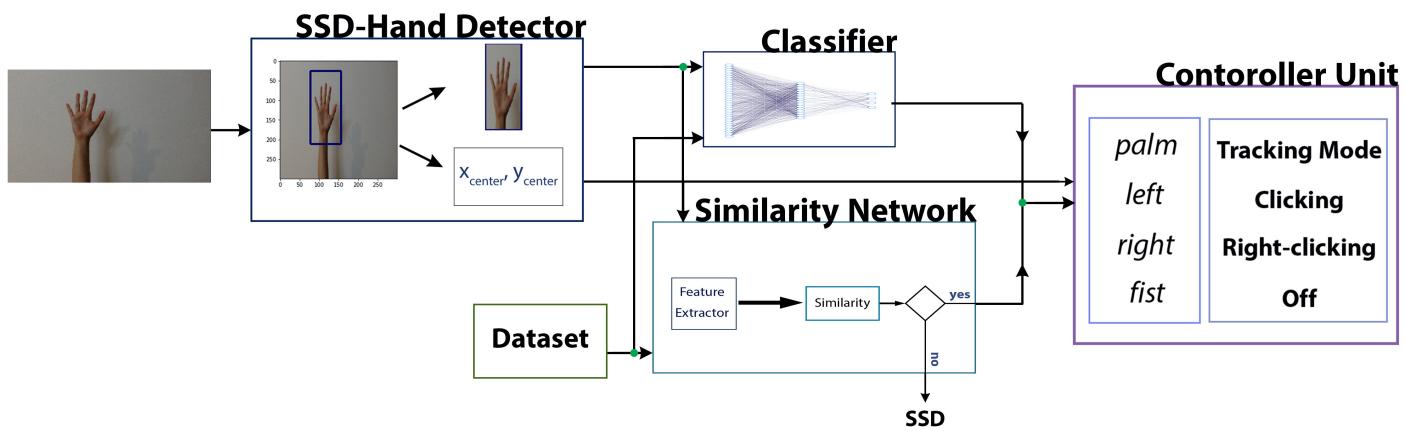
بدیهی است ابتدا ماوس خاموش است؛ بنابراین اولین برچسب وارد شده به بخش کنترل، وارد حالت اول می‌شود. پیش‌تر توضیح داده شد که تنها در صورتی که کاربر کف دست خود را جلو دوربین قرار دهد، ماوس روشن می‌شود و تا قبل از نشان دادن کف دست، ماوس خاموش بوده و هیچ‌یک از اشارات دست کاربر به فرمان در جهت کنترل نشان‌گر تبدیل نخواهد شد. پس از روشن شدن ماوس، کنترل ماوس توسط چهار حالت تعریف شده در مجموعه دادگان ممکن می‌گردد و کاربر می‌تواند نشان‌گر ماوس را جایه‌جا کند، در محل فعلی کلیک یا راست‌کلیک زده و ماوس را خاموش کند.

پیش‌تر نیز اشاره شد که جایه‌جایی نشان‌گر ماوس علاوه بر برچسب کف دست، نیاز به مختصات مرکز فریم داشته که از طریق بخش SSD، به بخش کنترل کننده اعمال و به مختصات مورد استفاده در نشان‌گر تبدیل می‌شود. لازم به ذکر است دو فرمان روشن و خاموش کردن ماوس به منظور کاهش محاسبات و پردازش ماوس انجام شده است تا بازه‌های زمانی‌ای که کاربر قصد استفاده از ماوس طراحی شده را ندارد، پردازشی صورت نگیرد.

الگوریتم ۴-۴ بخش کنترل کننده و تشخیص دهنده فرامین به ماوس طراحی شده



الگوریتم ۵-۴ نمای کلی ماوس طراحی شده به همراه ورودی و خروجی بخش‌های مختلف آن



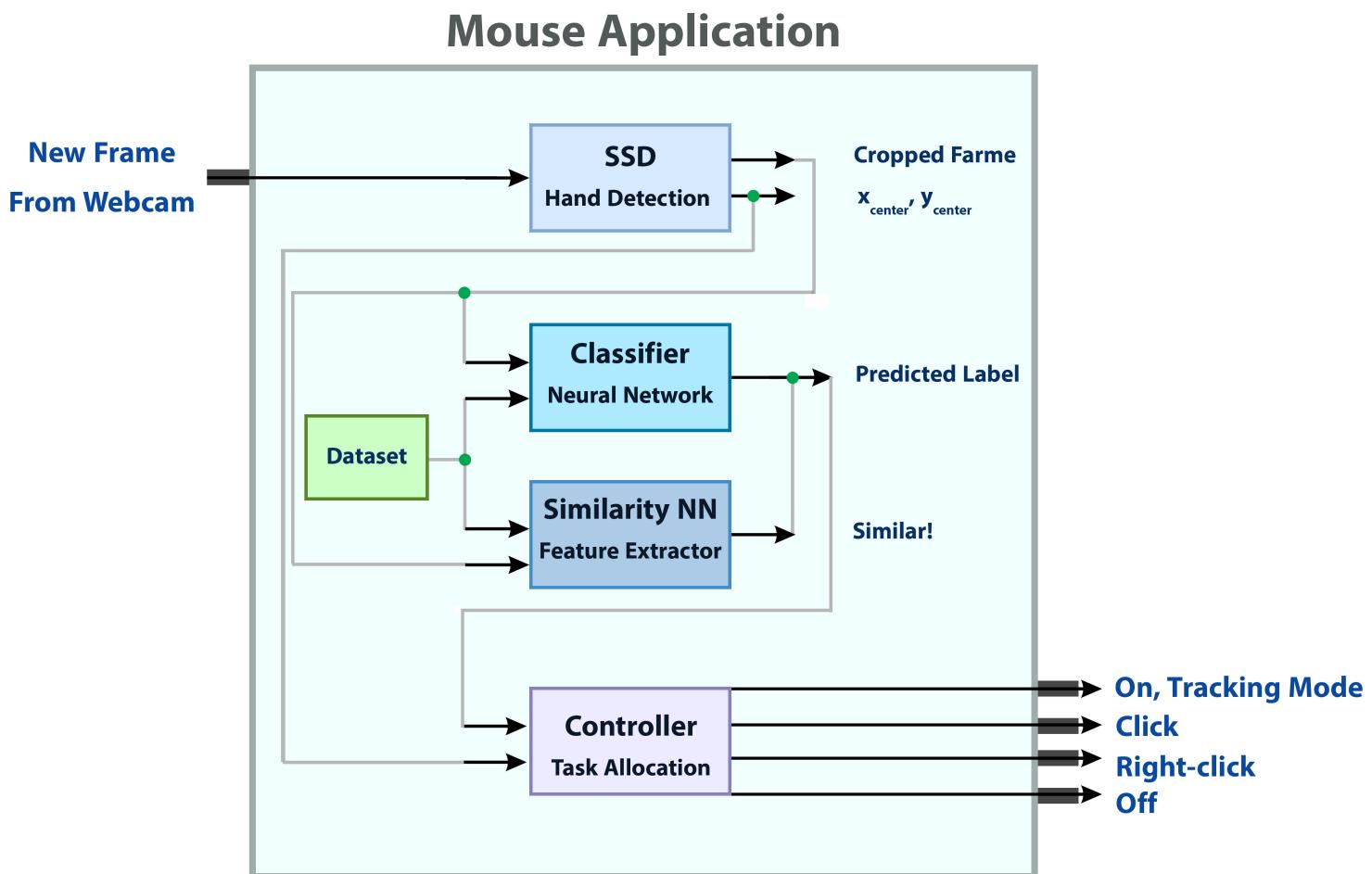
در الگوریتم ۵-۴ نیز نمای کلی ماوس طراحی شده به همراه بخش‌های مختلف آن و ورودی و خروجی‌های هر بخش ارائه شده است. لازم به ذکر است در این شکل برخی جزئیات از بخش‌های تشخیص دهنده دست، طبقه‌بندی و واحد کنترل کننده حذف گردیده است؛ الگوریتم بدین صورت عمل می‌کند که دوربین رایانه با استفاده از الگوریتم‌های پردازش تصویر و کتابخانه‌های آن یک فریم دریافت کرده و به بخش تشخیص دهنده دست

فصل ۴. الگوریتم پیاده‌سازی شده

می‌فرستد. بخش تشخیص دهنده دست دو خروجی ناحیه دست و مرکز آن ناحیه را به دست آورده و به دو بخش طبقه‌بندی و تخصیص وظایف می‌دهد.

در بخش طبقه‌بندی توسط طبقه‌بند یک برچسب به فریم شامل دست نگاشته می‌شود. همچنین میزان شباهت فریم شامل دست با بردارهای مرجع سنجیده می‌شود تا در صورت وجود شباهت کافی، برچسب طبقه‌بند معتبر تلقی شود. در نهایت برچسب به بخش تخصیص دهنده وظایف رفته و به یکی از فرآمین لازم جهت کنترل ماوس تبدیل می‌گردد. در الگوریتم ۶-۴ نیز فلوچارت کنترل کننده ماوس طراحی شده به همراه ورودی و خروجی آن، ارائه گردیده است.

الگوریتم ۶-۴ نحوه عملکرد ماوس طراحی شده به همراه ورودی و خروجی آن



۶-۴ نتیجه‌گیری

در این فصل یک مجموعه دادگان برای حالت‌های مختلف دست در جهت کنترل نشان‌گر رایانه ارائه گردید. همچنین با استفاده از معماهای آماده شبکه عصبی کانولوشنی و لایه‌های مختلف، یک شبکه عصبی در جهت طبقه‌بندی نمونه‌های موجود در مجموعه دادگان، آموزش داده شد. سپس با استفاده از برچسب‌های خروجی طبقه‌بند، یک واحد در جهت کنترل ماوس طراحی شده معرفی گردید. در فصل بعد ماوس طراحی شده ارزیابی می‌گردد.

فصل ۵

ارزیابی الگوریتم ارائه شده و نتایج

۱-۵ مقدمه

تا به اینجا نحوه پیاده‌سازی مدل طراحی شده در راستای کنترل نشان‌گر رایانه ارائه گردید؛ به گونه‌ای که توسط ۴ حالت تعریف شده برای دست، می‌توان ماوس طراحی شده را روشی یا خاموش کرد و یکی از حالت‌های کلیک، راست‌کلیک و جابه‌جایی نشان‌گر را انجام داد. در فصل پیش‌رو به ارزیابی الگوریتم ارائه شده پرداخته می‌شود؛ به گونه‌ای که دقت شبکه عصبی آموزش دیده بر اساس دو معماری مختلف بررسی می‌گردد. همچنین ماوس طراحی شده در شرایط محیطی مختلف ارزیابی می‌گردد. بخش ارزیابی سیستم توسط کارت پردازنده گرافیکی GTX 1080 انجام شده است.

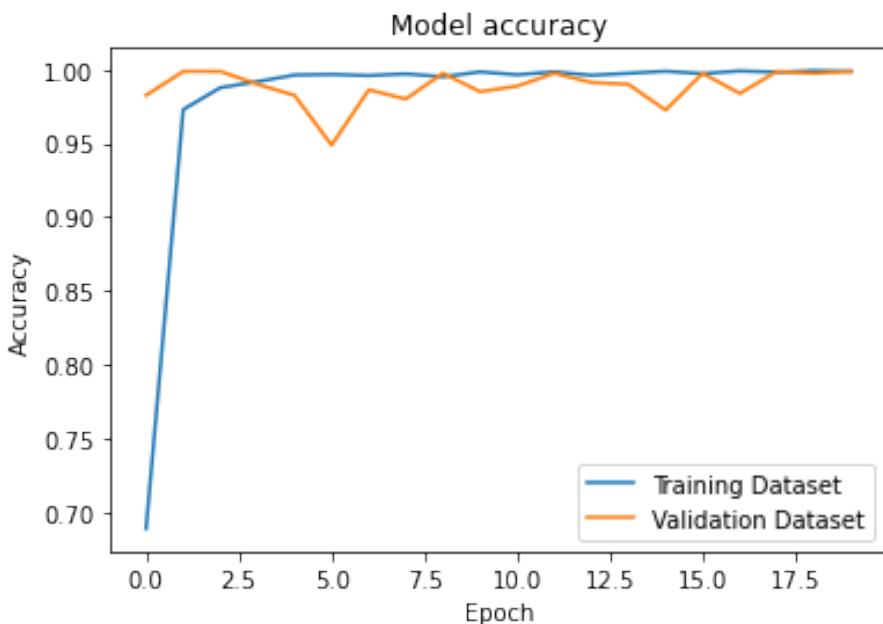
۲-۵ تاخیر ماوس طراحی شده

یکی از چالش‌های مهم در این پژوهش پردازش سنگین بلوک‌های استفاده شده است که در نهایت منجر به تأخیر در عملکرد ماوس می‌شود. در واقع تاخیر کننده ماوس طراحی شده در زمان اجرا، تابعی از تأخیر دو بلوک تشخیص دهنده دست و طبقه‌بندی است که هر دو تابعی از ساخت افزار مورد استفاده نیز هستند. بلوک طبقه‌بندی نیز شامل دو بخش نگاشت تصویر به یکی از حالات موجود در مجموعه دادگان و محاسبه شباهت و مقایسه با

یک آستانه است.

در این پژوهش در راستای تشخیص دست از الگوریتم SSD استفاده شده است. حال که الگوریتم تشخیص دهنده SSD با استفاده از کارت گرافیک Nvidia Titan X توانایی پردازش ۵۸ فریم بر ثانیه^۱ را دارد [۴۳]. الگوریتم تشخیص دهنده دست استفاده شده در این پژوهش با استفاده از کارت گرافیک GTX 1080 توانایی پردازش ۲۰ فریم بر ثانیه را دارد. در نهایت قدرت پردازش کننده ماوس طراحی شده ۱۵ فریم بر ثانیه است.

لازم به ذکر است در راستای نمایش تصویر کاربر، از توابع موجود در کتابخانه OpenCV استفاده شده است. فراخوانی این تابع در هنگام اجرای برنامه، موجب ایجاد تأخیر شده و تعداد فریم بر ثانیه ماوس طراحی شده برای کارت گرافیکی همچون Intel Iris Plus Graphics 640، دو فریم بر ثانیه خواهد بود. همچنین اگر تصویر فرد نمایش داده نشود، الگوریتم توانایی پردازش ۵ فریم بر ثانیه را دارد.

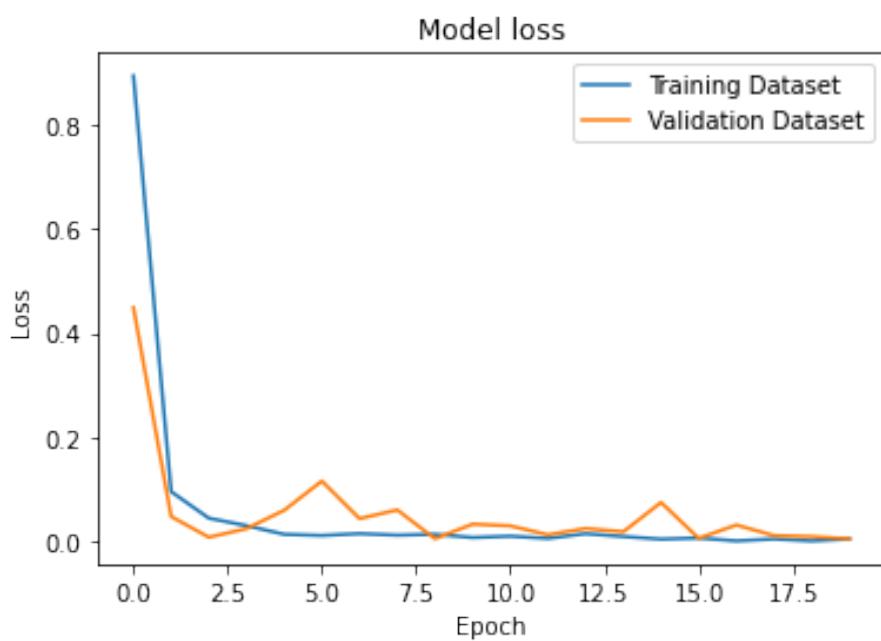


شکل ۱-۵: نمودار دقت برای دادگان آموزش و اعتبارسنجی در ۲۰ ایپاک

^۱Frames per Second

۳-۵ ارزیابی طبقه‌بند طراحی شده

در بخش طبقه‌بندی توضیح داده شد که شبکه طبقه‌بند بر روی مجموعه دادگان آموزش و ۵۰ درصد از دادگان اعتبارسنجی آموزش دیده است. در نهایت شبکه بر روی دادگان آزمایش که ۵۰ درصد باقیمانده از تصاویر اعتبارسنجی که هرگز ندیده است، به دقت ۹۹ درصد رسید. البته از آنجایی که دادگان آزمایش هم‌جنس دادگان اعتبارسنجی بوده، نسبت به دادگان آموزش ساده‌تر هستند. به دلیل آن که مجموعه دادگان اعتبارسنجی توسط الگوریتم تشخیص‌دهنده دست SSD تهیه شده‌اند، پس زمانیه محدودتری دارند. از طرفی این تصاویر لزوماً مورد تأیید الگوریتم تشخیص‌دهنده هستند. در ادامه نیز نمودار دقت و خطای Cross Entropy برای دادگان جمع‌آوری شده در ۲۰ ایپاک ارائه شده است.



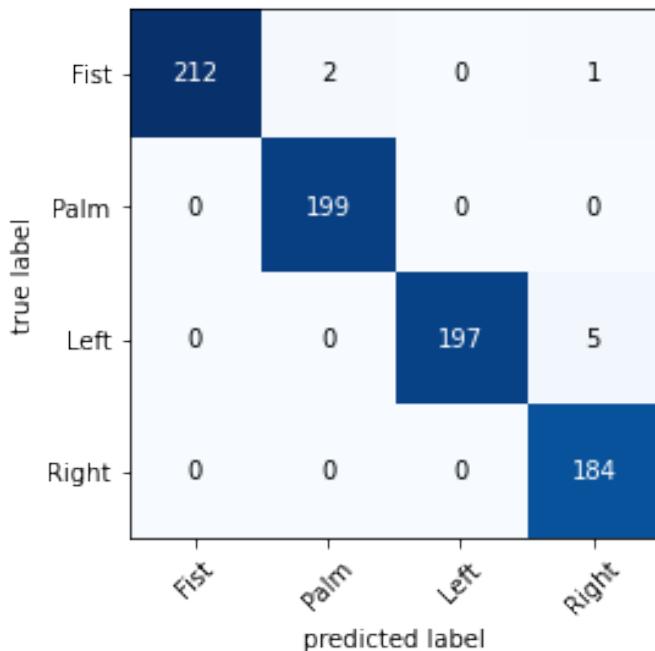
شکل ۲-۵: نمودار خطای Cross Entropy برای دادگان آموزش و اعتبارسنجی در ۲۰ ایپاک

در شکل ۳-۵ ماتریس درهم‌ریختگی^۲ برای دادگان آزمایش ارائه شده است. پیش‌تر گفته شد که تعداد دادگان آزمایش ۸۰۰ بوده که هر دسته تقریباً شامل ۲۰۰ تصویر رنگی است. از آنجایی که نمونه‌ها به صورت تصادفی انتخاب شده‌اند، تعداد نمونه‌های درون هر دسته یکسان نیست. دسته اول تا چهارم به ترتیب مربوط به مشت، کف دست، اشاره به چپ و اشاره به راست هستند.

²Confusion Matrix

فصل ۵. ارزیابی الگوریتم ارائه شده و نتایج

لازم به ذکر است شبکه طراحی شده ۸ تصویر را با برچسب اشتباه پیش‌بینی کرده است. به عنوان مثال، شبکه طبقه‌بند برای ۵ نمونه از تصاویر اشاره به راست، برچسب اشاره به چپ را پیش‌بینی کرده است. همچنین مشاهده می‌شود که شبکه در تشخیص حالت اشاره به راست، ضعیف تر از سایر حالت‌ها عمل کرده و حالت اشاره به سمت چپ را با دقت ۱۰۰ درصد تشخیص داده است.



شکل ۳-۵: ماتریس درهم‌ریختگی مربوط به بخش طبقه‌بندی با استفاده از دادگان آزمایش

پیش‌تر اشاره شد که برای آموزش شبکه عصبی کانولوشنی از مدل EfficientNet-B0 استفاده شده است و در انتهای این مدل چندین لایه تماماً متصل قرار داده شده است. همچنین در راستای بررسی عملکرد این مدل از یک شبکه VGG16 نیز بهره برده شده تا عملکرد شبکه آموزش دیده مقایسه گردد. جهت مقایسه دو شبکه بر پایه VGG16 و EfficientNet-B0 از سه معیار دقت مجموعه دادگان آزمایش، تعداد پارامترهای شبکه و زمان مورد نیاز استفاده شده است که در جدول ۱-۵ قابل مشاهده هستند.

فصل ۵. ارزیابی الگوریتم ارائه شده و نتایج

جدول ۱-۵: مقایسه دو شبکه طبقه‌بند بر پایه EfficientNet-B0 و VGG16

شبکه	دقت دادگان آزمایش	تعداد پارامترها	زمان مورد نیاز
EfficientNet-B0	۹۹٪	۴.۹۸ میلیون	۶۷۱ us/step
VGG16	۱۰۰٪	۱۵.۹۵ میلیون	۷۷۶ us/step

همان طور که در جدول ۱-۵ قابل مشاهده است دقت شبکه طراحی شده بر پایه VGG16 برای مجموعه دادگان آزمایش بیشتر از دقت شبکه بر پایه EfficientNet-B0 برای همین دادگان است؛ اما افزایش دقت در شبکه VGG16 هزینه افزایش تعداد پارامترها را دارد که در شبکه VGG16 در حدود ۳۰۲ برابر تعداد پارامترهای شبکه بر پایه EfficientNet-B0 است.

زیاد بودن تعداد پارامترهای یک شبکه ممکن است منجر به کاهش قدرت تعییم آن به دادگان جدید شود. همچنین شبکه با تعداد پارامترهای بیشتر به زمان یادگیری طولانی تر و بعضاً سخت افزار قوی تر نیاز دارد. از طرفی شبکه طبقه‌بند مبتنی بر VGG16 نیازمند زمان بیشتری در هر قدم بوده که معرف سرعت شبکه در هنگام اجرا است. در نهایت به منظور پیش‌بینی فریم‌های خروجی از مرحله تشخیص دست و بررسی شباهت این فریم‌ها با تصاویر موجود در دادگان اعتبارسنجی، از شبکه عصبی کانولوشنی بر پایه EfficientNet-B0 استفاده شده است.

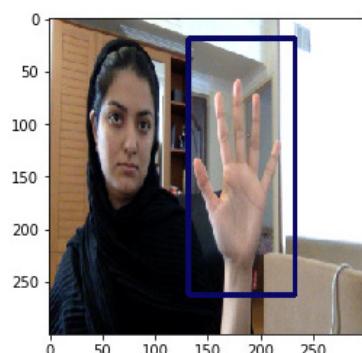
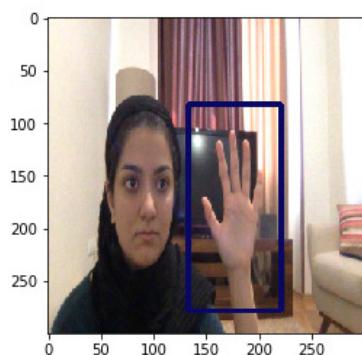
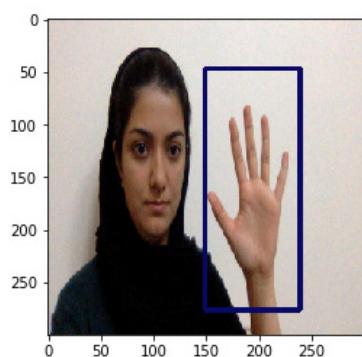
۴-۵ ارزیابی ماوس طراحی شده

جهت ارزیابی مدل طراحی شده در راستای کنترل ماوس، سه پس‌زمینه مختلف برای اجرای الگوریتم انتخاب گردید که در شکل ۴-۵ ارائه شده است. همان طور که در این تصاویر قابل مشاهده است، سه پس‌زمینه سفید، ساده و شلوغ در جهت ارزیابی ماوس طراحی شده انتخاب شده است. لازم به ذکر است تصاویر نشان داده شده، خروجی الگوریتم تشخیص دهنده دست بوده که ابعاد ۳۰۰ در ۳۰۰ دارند.

حال لازم است کنترل کننده ماوس طراحی شده در سه پس‌زمینه انتخاب شده ارزیابی گردد. همچنین جهت افزایش تنوع از دو شرایط نوری متفاوت برای هر محیط استفاده شده است. علاوه بر آن، دو فاصله متفاوت مابین دست و دوربین درنظر گرفته شد و کاربر از هر دو دست خود برای کنترل نشان‌گر استفاده کرده است.

فصل ۵. ارزیابی الگوریتم ارائه شده و نتایج

در جهت ارزیابی الگوریتم لازم است حالت های روشن کردن، کلیک کردن، راست کلیک کردن و خاموش کردن ماوس بررسی شود. لازم به ذکر است حالت جایه جایی نشان گر که مشابه حالت روشن کردن ماوس است، بررسی نشده است. برای هر یک از شرایط گفته شده ۱۰ فریم درنظر گرفته شد. از آنجایی که دو دست، دو شرایط نوری و دو فاصله انتخاب شده است، برای هر یک از حالات کنترل ماوس، ۸۰ فریم بررسی می شود. نتایج مربوط به ارزیابی الگوریتم با پس زمینه سفید در جدول ۲-۵ ارائه شده است.



شکل ۵-۴: پس زمینه های منتخب جهت ارزیابی کنترل کننده ماوس طراحی شده

فصل ۵. ارزیابی الگوریتم ارائه شده و نتایج

حالات هایی که در جدول ۲-۵ دارای عدد ۱۰ هستند، فرمان های ماوس را به درستی تشخیص داده اند. شرایط نوری اول تاریکتر بوده و در برخی موارد ضعیفتر از شرایط نوری دوم عمل کرده است. همچنین در بین حالت های کنترل ماوس، حالت راست کلیک با دقت کمتر به نسبت سایر حالت ها عمل می کند. ممکن است شباهت این حالت با دو حالت خاموش کردن و کلیک کردن منجر به خطأ شده باشد. البته در بخش طبقه بند نیز دقت حالت اشاره به راست عنوان گردید و گفته شد با دقت کمتری نسبت به سایر حالت ها تشخیص داده شده است.

جدول ۲-۵: نتایج ارزیابی کنترل کننده ماوس طراحی شده در پس زمینه سفید

شرایط نوری دوم				شرایط نوری اول				پس زمینه: سفید	فاصله
دست چپ	دست راست	دست چپ	دست راست	دست چپ	دست راست	دست چپ	دست راست		
دور	نزدیک	دور	نزدیک	دور	نزدیک	دور	نزدیک		
۱۰	۹	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	خاموش
۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	روشن، جابه جایی
۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	کلیک
۸	۱۰	۱۰	۹	۱۰	۸	۹	۷		راست کلیک

در جدول ۳-۵ شرایط گفته شده، برای پس زمینه ساده نیز بررسی شده است. در مورد حالت کلیک کردن تغییری ایجاد نشده است؛ اما برای خاموش کردن و روشن کردن یا جابه جایی ماوس، دقت عملکرد کاهش یافته است که با وجود پس زمینه ای که دیگر سفید یکنواخت نیست، قابل توجیه است. البته عملکرد راست کلیک نسبت به حالت پس زمینه اول بهتر بوده است.

جدول ۳-۵: نتایج ارزیابی کنترل کننده ماوس طراحی شده در پس زمینه ساده

شرایط نوری دوم				شرایط نوری اول				پس زمینه: ساده	فاصله
دست چپ	دست راست	دست چپ	دست راست	دست چپ	دست راست	دست چپ	دست راست		
دور	نزدیک	دور	نزدیک	دور	نزدیک	دور	نزدیک		
۱۰	۱۰	۸	۱۰	۱۰	۱۰	۱۰	۱۰	۹	خاموش
۱۰	۱۰	۱۰	۱۰	۱۰	۹	۱۰	۱۰	۱۰	روشن، جابه جایی
۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	۱۰	کلیک
۱۰	۹	۱۰	۹	۱۰	۱۰	۱۰	۱۰	۱۰	راست کلیک

لازم به ذکر است از آنجایی که در پس زمینه دوم تعداد فریم های بیشتری به دسته حالت ها ناخواسته نگاشت

فصل ۵. ارزیابی الگوریتم ارائه شده و نتایج

می‌شوند، الگوریتم تنها حالت‌هایی را که کاملاً واضح هستند، در نظر می‌گیرد. در واقع هزینه بهبود عملکرد برای حالت راست‌کلیک، تعداد فریم بیشتری است که توسط الگوریتم پردازش شده است تا عمل راست‌کلیک اجرا گردد.

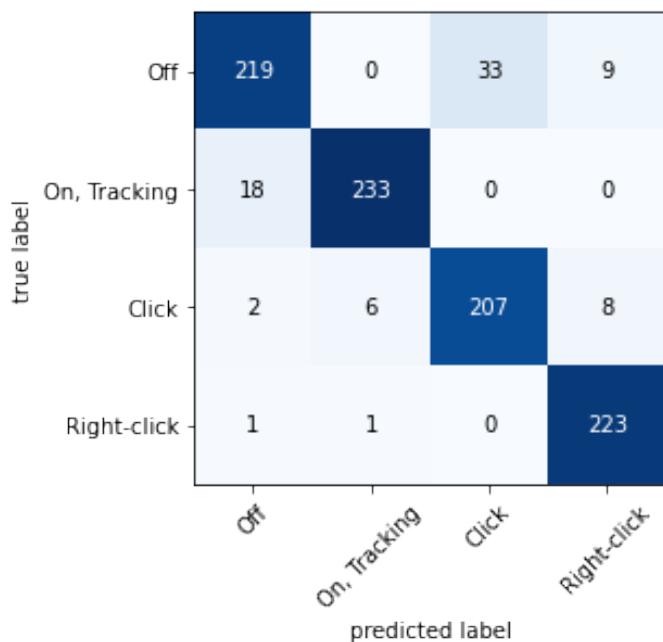
ارزیابی کنترل‌کننده ماوس طراحی شده برای پس‌زمینه شلوغ در جدول ۴-۵ ارائه شده است. لازم به ذکر است هنگام کنترل ماوس توسط دست چپ، پس‌زمینه و رنگ پوست شباهت زیادی داشته و تشخیص دهنده دست توانایی تفکیک دست از فضای پشت آن را ندارد؛ بنابراین دقت الگوریتم تشخیص دهنده دست و در نتیجه آن دقت طبقه‌بند برای این پس‌زمینه کاهش یافته است. به عنوان مثال در حالت جایه‌جایی نشان‌گر با دست چپ، به دلیل شباهت رنگ پس‌زمینه با پوست، فقط انگشت شست به عنوان دست تشخیص داده شده و به آن فرمان راست‌کلیک نگاشت داده شده است.

جدول ۴-۵: نتیاج ارزیابی کنترل‌کننده ماوس طراحی شده در پس‌زمینه شلوغ

شرایط نوری دوم				شرایط نوری اول				پس‌زمینه: شلوغ
دست چپ		دست راست		دست چپ		دست راست		
دور	نزدیک	دور	نزدیک	دور	نزدیک	دور	نزدیک	فاصله خاموش
۳	۵	۹	۱۰	۷	۹	۱۰	۱۰	روشن، جایه‌جایی
۱۰	۱۰	۱۰	۱۰	۶	۸	۱۰	۱۰	کلیک
۹	۹	۱	۱۰	۲	۳	۱۰	۱۰	راست‌کلیک
۱۰	۹	۱۰	۱۰	۸	۷	۱۰	۱۰	

در نهایت الگوریتم کنترل‌کننده ماوس برای فریم‌های بررسی شده با سه پس‌زمینه گفته شده، دارای ماتریس درهم‌ریختگی به صورت شکل ۵-۵ است. همان‌طور که مشاهده می‌شود، عملکرد کلیک کردن نسبت به سایر حالت‌ها ضعیف بوده که به دلیل هم‌رنگی پوست دست با پس‌زمینه شلوغ است.

فصل ۵. ارزیابی الگوریتم ارائه شده و نتایج



شکل ۵-۵: ماتریس درهم ریختگی برای کنترل کننده ماوس طراحی شده

با استفاده از مقادیر به دست آمده در راستای ارزیابی کنترل کننده ماوس طراحی شده، با وجود پس زمینه های سفید، ساده و شلوغ، نتایج جدول ۵-۵ برای دقت الگوریتم به دست آمد. پیش تر نیز گفته شد، ماوس طراحی شده در پس زمینه ساده نسبت به پس زمینه کاملاً سفید دارای دقت بالاتر است؛ اما هزینه آن تعداد فریم های بیشتری است که کاربر برای نشان دادن حالت مدنظر خود به الگوریتم تشخیص دهنده نیاز دارد. در نهایت دقت ماوس طراحی شده با پس زمینه های ارزیابی شده ۹۲.۶ درصد است.

جدول ۵-۵: مقادیر دقت برای سه پس زمینه مختلف به تفکیک هر حالت از کنترل ماوس طراحی شده

مجموع	پس زمینه شلوغ	پس زمینه ساده	پس زمینه سفید	
۹۱.۲۵%	۷۸.۷۵%	۹۶.۲۵%	۹۸.۷۵%	خاموش
۹۷.۰۸%	۹۲.۵%	۹۸.۷۵%	۱۰۰%	روشن، جابه جایی
۸۹.۱۷%	۶۷.۵%	۱۰۰%	۱۰۰%	کلیک
۹۲.۹۲%	۹۲.۵%	۹۷.۵%	۸۸.۷۵%	راست کلیک
۹۲.۶%	۸۲.۸۱%	۹۸.۱۳%	۹۶.۸۸%	مجموع

یکی از مزایای کنترل کننده ماوس طراحی شده نسبت به سایر الگوریتم های دست که در جهت کنترل نشان گر

رايانه يا ماشین‌ها مورد استفاده قرار می‌گيرند، عدم وابستگی به رنگ پوست کاربر است. در فصل دوم دیده شد که در جهت کنترل ماوس، معمولاً از یک محدوده برای رنگ پوست انسان استفاده می‌شود و پس زمینه بر اساس آن حذف می‌گردد. همچنین می‌توان به حذف حالت‌های ناخواسته که برای دست با توجه به شکل و ماهیت آن ضروری است، اشاره کرد.

۵-۵ نتیجه‌گیری

در این فصل به ارزیابی کنترل‌کننده نشان‌گر رایانه طراحی شده پرداخته شد و عملکرد آن از منظر دقیق مژول طبقه‌بندی، تأخیر و دقیق الگوریتم کلی ماوس بررسی گردید. در فصل آینده یک جمع‌بندی از آنچه تا به اینجا گفته شده است، ارائه می‌گردد. همچنین پیشنهادهایی در باب بهبود الگوریتم کنترل‌کننده نشان‌گر عنوان می‌گردد.

فصل ۶

بحث و نتیجه‌گیری

۱-۶ مقدمه

در پژوهش پیش‌رو الگوریتمی در راستای کنترل نشان‌گر رایانه با استفاده از یک دوربین و حرکات دست کاربر ارائه گردید. در این پژوهش از شبکه‌های عصبی در جهت پیش‌برد پژوهش چه در بخش تشخیص دهنده دست و چه در بخش طبقه‌بندی و بررسی شباهت استفاده شده است. در این فصل به جمع‌بندی آنچه تاکنون ارائه گردید، پرداخته می‌شود. علاوه بر آن در راستای بهبود و تکمیل پژوهش، پیشنهاداتی مبنی بر کارهای آتی عنوان خواهد شد.

۲-۶ جمع‌بندی

همان طور که مشاهده گردید، هدف از این پژوهش طراحی و پیاده‌سازی یک الگوریتم در جهت برقراری ارتباط انسان و رایانه توسط دست او و بدون تماس است؛ بنابراین با توجه به شیوع بیماری کویید ۱۹ استفاده از کنترل کننده طراحی شده می‌تواند در مکان‌های عمومی در راستای کنترل رایانه یا سایر سیستم‌های هوشمند مناسب باشد. در این پژوهش، اشارات دست کاربر به منظور راه ارتباطی کاربر با رایانه درنظر گرفته شده است تا با استفاده از آن نشان‌گر ماوس رایانه کنترل گردد. همچنین مشاهده گردید که این پژوهش علاوه بر روش‌های یادگیری ماشین

همچون شبکه‌های کانولوشنی و یادگیری عمیق از ابزارهای پردازش تصویر، شامل کتابخانه‌ها و توابع موجود آن نیز بهره برده است؛ به گونه‌ای که فریم‌های تهیه شده توسط دوربین، وارد بخش تشخیص دهنده دست می‌شوند و ناحیه دست جدا می‌گردد. از طرفی با استفاده از مجموعه دادگان جمع‌آوری شده یک شبکه عصبی کانولوشنی آموزش داده شد تا به نواحی جداسده شامل دست، برچسب متناسب تشخیص داده شود. ساختار کلی و مراحل پیاده‌سازی ماوس طراحی شده را می‌توان به صورت زیر درنظر گرفت:

۱. دریافت فریم توسط دوربین و اعمال آن به ماوس طراحی شده

۲. استفاده از الگوریتم تشخیص دهنده دست

• جدا کردن دست از فریم دریافت شده

• محاسبه مختصات نقطه مرکزی فریم بریده شده

۳. جمع‌آوری مجموعه دادگان متناسب به تعداد ۶۷۲۰ تصویر رنگی

۴. استفاده از شبکه عصبی کانولوشنی

• آموزش یک شبکه عصبی برپایه EfficientNET-B0 و لایه‌های تماماً متصل با استفاده از مجموعه دادگان جمع‌آوری شده

• ذخیره کردن بخش EfficientNET-B0 و وزن‌های آن به عنوان رمزنگار و طراحی شبکه شباهت

- تشکیل ۴ بردار مرجع با استفاده از تصاویر اعتبارسنجی

- مقایسه فریم بریده شده و بردارهای مرجع با استفاده از شبکه شباهت‌سنج

• پیش‌بینی برچسب متناظر با فریم بریده شده

• معتبر بودن برچسب پیش‌بینی شده در صورت تأیید شبکه شباهت‌سنج

۵. تشخیص وظیفه به برچسب پیش‌بینی شده در راستای کنترل نشانگر

• روشن کردن نشانگر با استفاده از کف دست

• کلیک کردن در محل فعلی نشانگر با اشاره به چپ

- راست‌کلیک کردن در محل فعلی نشان‌گر با اشاره به راست
- جابه‌جایی نشان‌گر بر اساس مختصات نقطه مرکزی فریم بریده‌شده و جابه‌جایی کف دست
- خاموش کردن ماوس با مشت کردن دست

درنهایت کنترل کننده ماوس طراحی شده به دقت ۹۲.۶ درصد برای پس‌زمینه‌های مختلف می‌توان از آن استفاده کرد. البته ماوس برای محیط‌های ساده به دقت ۹۷ درصد رسیده است.

۳-۶ نوآوری

در این پژوهش یک مجموعه دادگان از چهار حالت دست با تعداد ۶۷۲۰ تصویر رنگی با ابعاد ۱۰۰ در ۱۰۰ توسط نویسنده این پایان‌نامه جمع‌آوری گردید. این مجموعه شامل تصاویر هر دو دست افراد مختلف و در مکان‌های متفاوت با پس‌زمینه ساده است که در شرایط نوری، زوایا و فواصل مختلف تهیه شده‌اند. ایجاد تغییرات در نمونه‌ها موجب تعمیم مجموعه دادگان به نمونه‌های جدید می‌شود که این مجموعه دادگان را برای کاربردهای دیگر نیز قابل استفاده کرده است.

همچنین در راستای کنترل ماوس با استفاده از اشارات دست، از تشخیص دهنده‌های اشیا مبتنی بر شبکه‌های عصبی استفاده شده است. در این پژوهش با بهره بردن از الگوریتم SSD که یک تشخیص دهنده اشیا تک مرحله‌ای مبتنی بر شبکه‌های عصبی است، محل دست کاربر تعیین می‌گردد و به شبکه‌های طبقه‌بند در جهت طبقه‌بندی و حذف حالت‌های ناخواسته داده می‌شود تا در نهایت به برچسب پیش‌بینی شده یک وظیفه مناسب اعمال گردد.

۴-۶ محدودیت‌ها

محدودیت اصلی در این پژوهش پردازش سنگین و در نتیجه آن تأخیر ناشی از بخش‌های تشخیص دهنده دست و طبقه‌بندی است که نیاز به کارت گرافیک دارد. البته در صورتی که محدوده دوربین از پیش به کارت آموزش داده شده باشد، لزوم آنکه کارت در هنگام کار با ماوس خود را ببیند کاهش یافته و نیازی به کارت گرافیک وجود ندارد. یکی دیگر از محدودیت‌های موجود در کنترل کننده ماوس طراحی شده، تأثیر تغییرات نوری شدید بر عملکرد

ماوس است. اگرچه کنترل‌کننده تا حدود قابل قبولی نسبت به تغییرات نوری متفاوت است، اما در شرایط نوری کم، به دلیل عدم عملکرد مناسب تشخیص‌دهنده دست، امکان کنترل ماوس وجود نخواهد داشت.

۵-۶ کارهای آینده

از آنجایی که دست انسان حالت‌های مختلفی را می‌تواند پوشش دهد، تعریف حالت‌های بیشتر در مجموعه دادگان و تشخیص وظایفی همچون اسکرول کردن^۱ به کنترل بهتر ماوس کمک خواهد کرد. همچنین جمع‌آوری تعداد دادگان بیشتر در محیط‌هایی با پس‌زمنیه‌های شلوغ به بهبود عملکرد شبکه عصبی و افزایش قدرت تعییم آن کمک خواهد کرد.

در راستای افزایش سرعت ماوس طراحی شده و کاهش پردازش، می‌توان پس از تعیین محل دست توسط الگوریتم تشخیص‌دهنده، برای جابه‌جا کردن نشان‌گر تا زمان تغییر حالت یا خارج شدن دست از صفحه، از یک دنبال‌کننده دست استفاده کرد. از آنجایی که دنبال‌کننده‌ها از فریم‌های قبلی نیز اطلاعات دارند، بر اساس ویژگی‌های ظاهری دنبال شی می‌گردند. در واقع دنبال‌کننده‌ها باز محاسباتی کمتری نسبت به تشخیص‌دهنده اشیا داشته و در یک بازه زمانی مشخص، تعداد فریم‌های بیشتری را پردازش می‌کنند.

یکی دیگر از کارهایی که برای بهبود سرعت و کاهش باز محاسباتی ماوس طراحی شده می‌توان انجام داد، استفاده از مدل‌های تشخیص‌دهنده حرکت^۲ است؛ به گونه‌ای که با استفاده از الگوریتم‌های پردازش تصویر و تشخیص لبه^۳، ویژگی‌های مربوط به دسته‌های موجود استخراج شوند و مدل نسبت به این ویژگی‌های تعریف شده، حساس گردد تا تنها در صورتی که تغییری در این ویژگی‌ها صورت گرفت از طبقه‌بند استفاده شود. بدین صورت لازم نیست همه فریم‌ها به طبقه‌بند اعمال گردد که منجر به افزایش سرعت پردازش برنامه نیز می‌شود.

در راستای کنترل نشان‌گر رایانه با کنترل‌کننده طراحی شده در شرایط نوری کم می‌توان از یک دوربین مادون قرمز در جهت دریافت فریم‌ها استفاده کرد. در این صورت لازم است تشخیص‌دهنده دست و شبکه عصبی با نمونه تصاویر مربوط به دوربین مادون قرمز آموزش بیینند. البته استفاده از دوربین مادون قرمز دیگر صرفه اقتصادی نخواهد داشت.

¹ Scroll

² Motion Detection

³ Edge Detection

فصل ۶. بحث و نتیجه‌گیری

در نهایت بنا به کاربرد ماوس، می‌توان با استفاده از دادگانی که در محیط مدنظر قرار دارند، شبکه طبقه‌بندی را آموزش داد تا دقیق‌تر کننده بسته به آن پس‌زمینه افزایش یابد.

مراجع

- [1] H. S. Hasan and S. A. Kareem, “Human computer interaction for vision based hand gesture recognition: a survey,” in *2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, pp. 55–60, IEEE, 2012.
- [2] M. G. Atigh, “Design and implementation of computer interfaces for mobile control,” November 2016.
- [3] A. Królak, “Use of haar-like features in vision-based human-computer interaction systems,” in *2012 Joint Conference New Trends In Audio & Video And Signal Processing: Algorithms, Architectures, Arrangements And Applications (NTAV/SPA)*, pp. 139–142, IEEE, 2012.
- [4] V. Bhame, R. Sreemathy, and H. Dhumal, “Vision based hand gesture recognition using eccentric approach for human computer interaction,” in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 949–953, IEEE, 2014.
- [5] G. Fanelli, J. Gall, and L. Van Gool, “Real time head pose estimation with random regression forests,” in *CVPR 2011*, pp. 617–624, IEEE, 2011.
- [6] J. S. Sonkusare, N. B. Chopade, R. Sor, and S. L. Tade, “A review on hand gesture recognition system,” in *2015 International Conference on Computing Communication Control and Automation*, pp. 790–794, IEEE, 2015.
- [7] D. C. Engelbart and W. K. English, “A research center for augmenting human intellect,” in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pp. 395–410, 1968.
- [8] Z. Ren, J. Yuan, J. Meng, and Z. Zhang, “Robust part-based hand gesture recognition using kinect sensor,” *IEEE transactions on multimedia*, vol. 15, no. 5, pp. 1110–1120, 2013.

- [9] M. H. Alomari, A. AbuBaker, A. Turani, A. M. Baniyounes, and A. Manasreh, “Eeg mouse: A machine learning-based brain computer interface,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 5, no. 4, pp. 193–198, 2014.
- [10] M. E. Benalcázar, C. Motoche, J. A. Zea, A. G. Jaramillo, C. E. Anchundia, P. Zambrano, M. Segura, F. B. Palacios, and M. Pérez, “Real-time hand gesture recognition using the myo armband and muscle activity detection,” in *2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM)*, pp. 1–6, IEEE, 2017.
- [11] J. M. Carroll, “Human-computer interaction: psychology as a science of design,” *Annual review of psychology*, vol. 48, no. 1, pp. 61–83, 1997.
- [12] S. S. Rautaray and A. Agrawal, “Vision based hand gesture recognition for human computer interaction: a survey,” *Artificial intelligence review*, vol. 43, no. 1, pp. 1–54, 2015.
- [13] J.-H. Kim, N. D. Thang, and T.-S. Kim, “3-d hand motion tracking and gesture recognition using a data glove,” in *2009 IEEE International Symposium on Industrial Electronics*, pp. 1013–1018, IEEE, 2009.
- [14] R. Y. Wang and J. Popović, “Real-time hand-tracking with a color glove,” *ACM transactions on graphics (TOG)*, vol. 28, no. 3, pp. 1–8, 2009.
- [15] P. Mistry and P. Maes, “Sixthsense: a wearable gestural interface,” in *ACM SIGGRAPH ASIA 2009 Art Gallery & Emerging Technologies: Adaptation*, pp. 85–85, 2009.
- [16] S. Veluchamy, L. Karlmarx, and J. J. Sudha, “Vision based gesturally controllable human computer interaction system,” in *2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, pp. 8–15, IEEE, 2015.
- [17] C. Dhule and T. Nagrare, “Computer vision based human-computer interaction using color detection techniques,” in *2014 Fourth International Conference on Communication Systems and Network Technologies*, pp. 934–938, IEEE, 2014.
- [18] A. Agrawal, R. Raj, and S. Porwal, “Vision-based multimodal human-computer interaction using hand and head gestures,” in *2013 IEEE Conference on Information & Communication Technologies*, pp. 1288–1292, IEEE, 2013.
- [19] H. Huang, Y. Chong, C. Nie, and S. Pan, “Hand gesture recognition with skin detection and deep learning method,” in *Journal of Physics: Conference Series*, vol. 1213, p. 022001, IOP Publishing, 2019.

- [20] H. Park, “A method for controlling the mouse movement using a real time camera,” *Brown University, Providence, RI, USA, Department of computer science*, 2008.
- [21] U. Noreen, M. Jamil, and N. Ahmad, “Hand detection using hsv model,” *Hand*, vol. 6, no. 12, 2015.
- [22] H.-S. Grif and T. Turc, “Human hand gesture based system for mouse cursor control,” *Procedia Manufacturing*, vol. 22, pp. 1038–1042, 2018.
- [23] Globefire, “globefire/hand detection tracking opencv-”
- [24] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, “Visual object tracking using adaptive correlation filters,” in *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 2544–2550, IEEE, 2010.
- [25] J. Suarez and R. R. Murphy, “Hand gesture recognition with depth images: A review,” in *2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication*, pp. 411–417, IEEE, 2012.
- [26] R. Golash and Y. K. Jain, “Economical and user-friendly design of vision-based natural-user interface via dynamic hand gestures,” *International Journal of Advanced Research in Engineering and Technology*, vol. 11, no. 6, 2020.
- [27] C. Yi, L. Zhou, Z. Wang, Z. Sun, and C. Tan, “Long-range hand gesture recognition with joint ssd network,” in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1959–1963, IEEE, 2018.
- [28] P. Liu, X. Li, H. Cui, S. Li, and Y. Yuan, “Hand gesture recognition based on single-shot multibox detector deep learning,” *Mobile Information Systems*, vol. 2019, 2019.
- [29] Z. Ni, J. Chen, N. Sang, C. Gao, and L. Liu, “Light yolo for high-speed gesture recognition,” in *2018 25th IEEE International Conference on Image Processing (ICIP)*, pp. 3099–3103, IEEE, 2018.
- [30] U. Tanmaie and C. S. Rao, “Hand posture detection and classification using you only look once (yolo v2) object detector,”
- [31] N. O’Mahony, T. Murphy, K. Panduru, D. Riordan, and J. Walsh, “Real-time monitoring of powder blend composition using near infrared spectroscopy,” in *2017 Eleventh International Conference on Sensing Technology (ICST)*, pp. 1–6, IEEE, 2017.
- [32] S. Hayou, A. Doucet, and J. Rousseau, “On the selection of initialization and activation function for deep neural networks,” *arXiv preprint arXiv:1805.08266*, 2018.

فصل ٦. بحث و نتیجه‌گیری

- [33] C. B. G.-L. GHENEA and A.-A. NEACŞU, “Proiect de diplomă,”
- [34] D. Cornelisse, “An intuitive guide to convolutional neural networks,” *Free Code Camp*, 2018.
- [35] P. Mukherjee, “Convolution neural networks vs fully connected neural networks,” Jan 2019.
- [36] R. Rojas, *Neural networks: a systematic introduction*. Springer Science & Business Media, 2013.
- [37] F. D, “Batch normalization in neural networks,” Oct 2017.
- [38] N. O’Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, G. V. Hernandez, L. Krpalkova, D. Riordan, and J. Walsh, “Deep learning vs. traditional computer vision,” in *Science and Information Conference*, pp. 128–144, Springer, 2019.
- [39] M. Tan and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” *arXiv preprint arXiv:1905.11946*, 2019.
- [40] L. Hulstaert, “Going deep into object detection,” Apr 2018.
- [41] V. Dibia, “victordibia/handtracking,” Aug 2020.
- [42] S. Bambach, S. Lee, D. J. Crandall, and C. Yu, “Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions,” in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.
- [43] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*, pp. 21–37, Springer, 2016.
- [44] A. Bendale and T. E. Boult, “Towards open set deep networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1563–1572, 2016.
- [45] “Improving deep neural networks: Hyperparameter tuning, regularization and optimization.”
- [46] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.
- [47] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in *ICML deep learning workshop*, vol. 2, Lille, 2015.

Abstract:

A long time ago, the human being was always dreaming about picking things up, taking a simple hand gesture. Although up to now, no one could've moved things without having direct touch, Artificial Intelligence allows individuals to control the intelligent system by using hand gestures in front of a camera and without having a single contact. In the proposed thesis, a user interface is designed to establish an interaction between humans and computers, so the pointer can be controlled using users' hand gestures. The hand gestures' frames are captured and processed through a webcam and using techniques, libraries, and functions of image processing. Afterward, the hand location is detected by neural network algorithms. In order to control the computer's cursor using hand gestures, a hand dataset is collected, which has 6720 image samples, including 4 classes which are fists, palms, pointing to the left, and pointing to the right. The images of the dataset are captured by 15 persons in simple backgrounds and different light conditions. The collected dataset trains a convolutional neural network based on EfficientNet-B0 and fully-connected layers. The trained network is saved for two proposes: first, to classify the output frames of the hand detector and predict a label for each frame; Second, to compare the output frames with the images of the dataset. Eventually, the predicted label converts to command for controlling the cursor. The defined commands are turning the mouse on or off, moving the cursor, left and right-clicking. Accuracy of the presented mouse reaches to 92.6 percent and is appropriate to use in different backgrounds. Also, Python programming language is used in order to design the presented mouse.

Keywords: Hand Gesture Recognition, Dataset, Convolutional Neural Network, Classification, Computer Mouse, and Object Detection



**University of Tehran
College of Engineering
Faculty of Electrical and
Computer Engineering**



Control of Computer Mouse Using Hand Movement Detection in Motion Pictures

A Thesis submitted to the Graduate Studies Office
In partial fulfillment of the requirements for
The degree of Master of Science
in Electrical Engineering - Integrated Circuits

By:
Yalda Foroutan

Supervisors:
Dr. Ahmad Kalhor and Dr. Samad Sheikhaei

September 2020