



دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیووتر

گزارش پروژه کارشناسی

## هدایت پهپاد با علائم دست مبتنی بر بینایی ماشین

نگارش

سارا تاجرنیا

استاد راهنما

دکتر مهدی جوانمردی

خرداد ۱۴۰۳

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ



دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیووتر

گزارش پروژه کارشناسی

هدایت پهپاد با علائم دست مبتنی بر بینایی ماشین

نگارش

سارا تاجرنيا

استاد راهنما

دکتر مهدی جوانمردی

خرداد ۱۴۰۳

## صفحه فرم ارزیابی و تصویب پایان نامه- فرم تأیید اعضاء کمیته دفاع

در این صفحه فرم دفاع یا تایید و تصویب پایان نامه موسوم به فرم کمیته دفاع- موجود در پرونده آموزشی- را قرار دهید.

نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.



دانشگاه صنعتی امیرکبیر  
(پلی‌تکنیک تهران)

به نام خدا

## تعهدنامه اصالت اثر

تاریخ: خرداد ۱۴۰۳

اینجانب سارا تاجرنیا متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظرات و راهنمایی استادی دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است. در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، تسطیح، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مآخذ بلامنع است.

سارا تاجرنیا

امضا

این پیان نامه را تقدیم می کنم به محبا ترین همراهان زندگیم، پدر، مادر، برادران  
غزینزم که حضور شان همیشه گرما بخش روح من بوده است.

# سپاس‌گزاری

زندگی دفتری از خاطره هاست، یک نفر در دل شب، یک نفر در دل خاک، یک نفر همدم خوشبختی هاست، یک نفر همسفر سختی هاست، چشم تا باز کنیم، عمرمان می‌گذرد ما همه رهگذریم، آنچه باقیست فقط خوبی‌هاست.

تشکر می‌کنم از تمامی عزیزانی که در تمامی مراحل زندگی همراه من بوده‌اند.  
و همچنین از استاد گرامی جناب آقای دکتر مهدی جوانمردی که در انتخاب و پیشبرد این پروژه به عنوان استاد پروژه، کمک‌های فراوانی به این جانب داشتند، کمال تشکر را دارم.

سارا تاجرانی  
خرداد ۱۴۰۳

## چکیده

پهپادهای تجاری که به عنوان هواپیماهای بدون سرنشین<sup>۱</sup> نیز شناخته می‌شوند، به سرعت در حال رایج شدن هستند. این پهپادها در زمینه‌های مختلف مانند نظارت بر رویدادهای ورزشی، حمل و نقل تجهیزات و کالاهای اضطراری، فیلمبرداری، عکس‌برداری هواپی و ... مورد استفاده قرار می‌گیرند.

هدف این پروژه توسعه سیستمی است که بتوان با استفاده از آن، از حرکات دست به عنوان روشی برای کنترل پرواز پهپادها استفاده کرد. بدین صورت که با استفاده از روش‌های مبتنی بر بینایی ماشین<sup>۲</sup>، روشی بصری برای ارتباط بدون واسطه کنترلر، بین پهپاد و اپراتور آن ایجاد کرد. روش‌های مبتنی بر بینایی ماشین با استفاده از دوربین هواپیماهای بدون سرنشین که تصاویر اطراف را گرفته و پس از تحلیل تصاویر و تشخیص الگوی دست، اطلاعات معناداری از آن را استخراج می‌کنند. ساختار این پروژه از دو مأذول اصلی تشخیص حرکت دست<sup>۳</sup> و دستور به هواپیمای بدون سرنشین تشکیل شده است. برای مأذول اول از روش مبتنی بر یادگیری عمیق<sup>۴</sup> استفاده شده است تا بتوان با دقت بالا علامت دست را پیش‌بینی کرد. مأذول دوم نیز وظیفه ارتباط با پهپاد را بر عهده دارد که نتیجه تشخیص را برای پهپاد ارسال می‌کند تا از آن پیروی کند. در پروژه پیاده‌سازی شده نتیجه به دست آمده دارای دقت بالای ۹۸ درصد در تشخیص درست علائم دست است، همچنین پیاده‌سازی میدانی این پروژه بر روی پهپاد دی‌جی‌آی تلو عملکرد موفقیت‌آمیز این سیستم را نشان می‌دهد.

## واژه‌های کلیدی:

پهپاد، علامت دست، شبکه عصبی پیچشی<sup>۵</sup>، شبکه حافظه طولانی کوتاه مدت<sup>۶</sup>، شبکه عصبی بازگشتی<sup>۷</sup>،

<sup>1</sup>Unmanned Aerial Vehicles

<sup>2</sup>Computer Vision

<sup>3</sup>Hand Detection

<sup>4</sup>Deep Learning

<sup>5</sup>Convolutional Neural Network(CNN)

<sup>6</sup>Long Short-Term Memory(LSTM)

<sup>7</sup>Recurrent Neural Network(RNN)

<sup>8</sup>Human–Drone Interface

رابط انسان و پھپاد<sup>۸</sup>

# فهرست مطالب

صفحه

عنوان

۱	.....	۱	مقدمه
۲	.....	۱-۱	مقدمه
۳	.....	۲-۱	چالش‌های استفاده از پهپاد
۴	.....	۳-۱	اهمیت علائم دست
۵	.....	۴-۱	اهمیت استفاده از بینایی ماشین در پهپاد
۶	.....	۵-۱	تعريف مسئله
۷	.....	۱-۵-۱	چالش‌های اجرای پروژه
۸	.....	۶-۱	مراحل انجام پروژه
۹	.....	۷-۱	جمع‌بندی
۱۰	.....	۲	کارهای مشابه
۱۱	.....	۱-۲	مقدمه
۱۲	.....	۲-۲	کارهای مشابه مربوط به ویژگی‌های تصویر
۱۳	.....	۲-۲-۱	پهپادهای کنترل شده با علائم دست به صورت متن باز
۱۴	.....	۲-۲-۲	روش‌های تشخیص علائم دست به صورت بی‌درنگ
۱۵	.....	۳-۲	کارهای مشابه مربوط به ورودی تصویر دست به مدل
۱۶	.....	۱-۳-۲	کنترل پهپاد با تشخیص علائم دست با کمک یادگیری عمیق
۱۷	.....	۲-۳-۲	کنترل یوای‌وی با تشخیص علائم دست
۱۸	.....	۴-۲	کارهای مشابه مربوط به نقاط کلیدی دست
۱۹	.....	۱-۴-۲	تشخیص علائم دست برای استفاده‌های بی‌درنگ
۲۰	.....	۲-۴-۲	تشخیص علائم دست با استفاده از نقاط کلیدی و جعبه مرزی
۲۱	.....	۳-۴-۲	تشخیص علائم بصری مبتنی بر نقاط کلیدی
۲۲	.....	۴-۴-۲	تشخیص بی‌درنگ دست با کمک کتابخانه مدبایپایپ
۲۳	.....	۵-۲	کارهای مشابه پهپاد و ترکیب آن با مدل بینایی ماشین
۲۴	.....	۱-۵-۲	مدل‌های مبتنی بر هوش مصنوعی بر روی پهپادها
۲۵	.....	۲-۵-۲	پهپاد دی‌جی‌آی تلو در زمینه کنترل مهندسی

۲۳	.....	۶-۲ جمع‌بندی
۲۵	.....	۳ روش انجام پروژه
۲۶	.....	۱-۳ مقدمه
۲۷	.....	۲-۳ انتخاب علائم دست مناسب برای کنترل پهپاد
۲۸	.....	۳-۳ مجموع داده
۲۹	.....	۴-۳ ابزارها و نرم افزارهای مورد استفاده
۲۹	.....	۱-۴-۳ کتابخانه تنسورفلو
۳۰	.....	۲-۴-۳ کتابخانه سایکیت لرن
۳۰	.....	۳-۴-۳ رابط برنامه‌نویسی کراس
۳۱	.....	۴-۴-۳ کتابخانه‌های دی‌جی‌آی تلوپی <sup>۹</sup>
۳۱	.....	۵-۴-۳ کتابخانه‌های مدیاپایپ
۳۲	.....	۶-۴-۳ کتابخانه نامپای
۳۳	.....	۷-۴-۳ کتابخانه مت‌پلات لیب
۳۳	.....	۸-۴-۳ کتابخانه اپن سی‌وی
۳۴	.....	۹-۴-۳ پهپاد دی‌جی‌آی تلو
۳۵	.....	۵-۳ روند اجرای پروژه
۳۶	.....	۶-۳ تشخیص دست <sup>۱۰</sup>
۳۶	.....	۱-۶-۳ مدل تشخیص کف دست
۳۷	.....	۲-۶-۳ محدود کننده جریان بی‌درنگ <sup>۱۱</sup>
۳۸	.....	۳-۶-۳ تشخیص به مستطیل محدود کننده <sup>۱۲</sup>
۳۸	.....	۷-۳ تشخیص نقاط عطف دست <sup>۱۳</sup>
۳۸	.....	۱-۷-۳ مدل تشخیص نقاط عطف دست

<sup>9</sup>Djitelopy

<sup>10</sup>Hand Detection

<sup>11</sup>Real Time Flow Limiter

<sup>12</sup>Detection To Rectangle

<sup>13</sup>Hand Landmark

<sup>14</sup>Image Cropping

<sup>15</sup>Landmarks To Rectangle

۳۹	.....	۲-۷-۳ برش تصویر <sup>۱۴</sup>
۴۰	.....	۳-۷-۳ نقاط عطف یه مستطیل <sup>۱۵</sup>
۴۰	.....	۸-۳ پیش‌بینی علائم دست
۴۱	.....	۳-۱-۸ مدل‌های مازول کلاس‌بندی برای تعیین علائم دست
۴۵	.....	۳-۲-۸ پیش‌پردازش داده‌ها
۴۶	.....	۳-۳-۸ رأی‌گیری پنجره‌ای
۴۷	.....	۳-۹-۳ ارتباط با پهپاد دی‌جی‌آی تلو
۴۸	.....	۳-۱۰-۳ پیاده‌سازی میدانی پروژه
۴۸	.....	۳-۱۱ جمع‌بندی
۵۰	.....	۴ نتایج و ارزیابی
۵۱	.....	۴-۱ مقدمه
۵۱	.....	۴-۲ معیارهای ارزیابی مدل‌های بینایی ماشین
۵۱	.....	۴-۱-۲ دقت
۵۲	.....	۴-۲-۲ صحت
۵۲	.....	۴-۳-۲ فراخوانی
۵۲	.....	۴-۴-۲ امتیاز F1
۵۳	.....	۴-۳-۳ ارزیابی عملکرد مدل‌ها
۵۳	.....	۴-۱-۳-۳ گزارش معیارهای ارزیابی در مدل‌ها
۵۳	.....	۴-۲-۳-۳ ماتریس درهم‌ریختگی
۵۵	.....	۴-۴-۴ نمودارهای دقت و خطاب بر حسب دوره
۵۶	.....	۴-۵-۴ تاثیر پس‌پردازش رأی‌گیری پنجره‌ای بر دقت
۵۷	.....	۴-۶-۴ سرعت اجرای برنامه
۵۸	.....	۴-۷-۴ سخت‌افزار مورد نیاز
۵۸	.....	۴-۸-۴ مقایسه دقت پروژه با کارهای مشابه
۵۹	.....	۴-۹-۴ جمع‌بندی
۶۱	.....	۵ نتیجه گیری و پیشنهادات
۶۲	.....	۵-۱ مقدمه

۶۲	.....	۲-۵ نتیجه‌گیری
۶۲	.....	۳-۵ پیشنهادات
۶۲	.....	۱-۳-۵ محدودیت وجود پهپادهای منبع باز
۶۳	.....	۲-۳-۵ وجود چندین دست در تصویر
۶۴	.....	۳-۳-۵ شناسایی یک دست در دو مرتبه
۶۴	.....	۴-۳-۵ اجرا بر روی پردازنده‌های گرافیکی
۶۵	.....	۵-۳-۵ اجرای پروژه با برنامه سی پلاس پلاس
۶۶	.....	۶-۳-۵ استفاده از مدل‌های موازی برای تشخیص علائم دست
۶۶	.....	۷-۳-۵ توسعه برنامه با استفاده از سایر قسمت‌های بدن انسان
۶۷	.....	۴-۵ جمع‌بندی
۶۹	.....	منابع و مراجع

## فهرست تصاویر

شکل

صفحه

۱-۲	ویژگی‌های هار برای استفاده از آستانه رنگ پوست برای تشخیص دست . . . . .	۱۱
۲-۲	چارچوب کنترل پهپاد مبتنی بر علائم دست با کمک ویژگی‌های هار . . . . .	۱۱
۳-۲	تشخیص کف دست و انگشتان . . . . .	۱۳
۴-۲	معماری VGG-16 . . . . .	۱۴
۵-۲	تشخیص نقطه کلید و تطبیق توسط سیفت . . . . .	۱۷
۶-۲	معماری ساختار شبکه‌های عصبی دو خط لوله . . . . .	۱۸
۷-۲	ورودی‌ها و خروجی‌های مدل تشخیص نقاط کلیدی دست . . . . .	۱۹
۸-۲	ارتباط برنامه پایتون با پهپاد دی‌جی‌آی تلو . . . . .	۲۳
۹-۳	تشخیص علائم دست با کمک نقاط کلیدی دست . . . . .	۲۷
۱۰-۳	نمونه‌ای از علائم انتخاب شده در مجموعه داده‌ها . . . . .	۲۸
۱۱-۳	برخی کاربردهای کتابخانه مدیاپایپ . . . . .	۳۲
۱۲-۳	طلاعات پهپاد دی‌جی‌آی تلو . . . . .	۳۴
۱۳-۳	روندهای پروژه . . . . .	۳۵
۱۴-۳	معماری مدل آشکارساز کف دست . . . . .	۳۷
۱۵-۳	معماری مدل نقطه عطف دست . . . . .	۳۹
۱۶-۳	موقعیت ۲۱ نقطه کلیدی در ناحیه دست . . . . .	۳۹
۱۷-۳	نمونه پیش‌بینی علامت دست . . . . .	۴۱
۱۸-۳	نمونه‌ای از شبکه‌های عصبی چند لایه‌ای دارای دو لایه پنهان . . . . .	۴۲
۱۹-۳	نمونه معماری شبکه عصبی پیچشی . . . . .	۴۳
۲۰-۳	نمونه معماری شبکه عصبی بازگشتی . . . . .	۴۴
۲۱-۳	نمونه معماری شبکه حافظه طولانی کوتاه مدت . . . . .	۴۵
۲۲-۳	نمونه پیش‌وردازش داده‌ها . . . . .	۴۶
۲۳-۳	نمونه رأی‌گیری پنجره‌ای موفق و ناموفق . . . . .	۴۷
۲۴-۳	نمونه تصویر پیاده‌سازی مدل بر روی پهپاد دی‌جی‌آی تلو . . . . .	۴۹
۲۵-۴	معیارهای ارزیابی برای تشخیص علائم دست در مدل شبکه عصبی پیچشی . . . . .	۵۴

۵۵	۲-۴ ماتریس درهم ریختگی در مدل شبکه عصبی پیچشی . . . . .
۵۶	۳-۴ منحنی حالت های کم برازش، بهینه و بیش برازش . . . . .
۵۶	۴-۴ نمودار روند دقت و خطأ در دست های راست و چپ بر حسب دوره در داده های آموزش و تست در مدل شبکه عصبی پیچشی . . . . .
۶۴	۱-۵ شناسایی یک دست دو مرتبه . . . . .

## فهرست جداول

صفحه

جدول

۱-۳ لیست دستورالعمل‌های استفاده شده در کتابخانه دیجی‌آی تلوپی برای ارتباط با پهپاد ۴۸

۱-۴ جدول ارزیابی دقت مدل‌ها ۵۴ .....

۲-۴ جدول ارزیابی تاثیر رأی‌گیری پنجره‌ای در دقت با استفاده از یک ویدیو دوازده هزار فریمی ۵۷ .....

۳-۴ جدول ارزیابی زمان پاسخگویی مدل‌ها ۵۷ .....

۴-۴ جدول ارزیابی حافظه موقت موردنیاز مدل‌ها ۵۸ .....

۵-۴ جدول مقایسه پروژه با کارهای مشابه ۵۹ .....

# فصل اول

## مقدمه

## ۱-۱ مقدمه

پهپادها یا به عبارتی هواپیماهای بدون سرنشین امروزه در صنایع مختلف به عنوان یک فناوری بسیار گسترده و کارآمد مورد استفاده قرار می‌گیرند. هواپیماهای بدون سرنشین اساساً به عنوان رباتهای پرنده‌ای دیده می‌شوند که عملکردهای متعددی در صنایع مانند جمع‌آوری داده‌ها و سنجش محیط اطراف را بر عهده دارند [۱]. از جمله این صنایع می‌توان به کشاورزی، ساخت و ساز، خدمات حمل و نقل و نقشه‌برداری اشاره کرد. یکی از دلایل اصلی افزایش کاربرد این هواپیماهای بدون سرنشین، کارایی بالای آنها است. این فناوری نه تنها به دلیل سرعت بالا در پوشش دهی مساحت‌های گسترده، بلکه به دلیل قابلیت برنامه‌ریزی و استفاده در صنایع مختلف مورد توجه قرار می‌گیرد. همچنین، صرفه‌جویی در هزینه‌های مالی و جانی و افزایش امنیت نیز از جمله عوامل مهمی است که اهمیت پهپادها را بیشتر می‌کند [۲].

در حال حاضر، ربات‌های پرنده در مشاغلی همچون سیستم‌های تحویل بسته استفاده می‌شوند [۳]. به عنوان مثال، شرکت‌هایی مانند آمازون و UPS از پهپادهای باری تحویل بسته‌های خود استفاده می‌کنند [۴]. در پی این موضوع، بسیاری از شرکت‌های تولید کننده پهپاد تشویق شده‌اند تا انواع مختلفی از ویژگی‌های نرم‌افزاری و سخت‌افزاری مانند حسگرها را به پهپادها اضافه کنند، که ابتدایی ترین آنها دوربین است. دوربین بصری یک حسگر ضروری برای پهپادهای فعلی است که آنها را به پهپادهای کاربردی و متعدد در بازار تبدیل می‌کند [۵]. همراه با این تغییرات زمینه مطالعاتی جدیدی به نام رابط هواپیماهای بدون سرنشین و انسان گشوده شد تا تعامل بین پهپادها و انسان را پیشرفت دهد. این تعامل با استفاده از مجموعه دستگاه‌های سنتی مانند کنترلرهای رادیویی<sup>۱</sup> و یا کنترل پهپادها با استفاده از وضعیت بدن و دست انسان انجام می‌شود [۶].

یکی از رویکردهای مورد استفاده برای افزایش کاربرد و دسترسی به پهپادها، استفاده از روش‌های مبتنی بر بینایی ماشین است. این کار معمولاً از طریق پردازش تصویر همراه با استفاده از شبکه‌های عصبی<sup>۲</sup> انجام می‌شود. پهپادهایی که با مدل‌های بینایی ماشین آموزش می‌بینند، توانایی تحلیل تصاویر و ویدئوهایی که از محیط اطراف دریافت می‌کنند را دارا هستند. این قابلیت به پهپاد این امکان را می‌دهد که بدون نیاز به تداخل انسانی، وظایفی همچون امنیت، ارسال کالا، تصویربرداری و ... را انجام دهد [۷]. می‌توان گفت که هدف اصلی استفاده از بینایی ماشین در پهپادها به حداقل رساندن دخالت انسان به

<sup>1</sup> Radio Controller

<sup>2</sup> Neural Networks

صورت مستقیم است. این امر پهپاد را قادر می‌سازد تا تشخیص اشیاء، تشخیص چهره، تحلیل تصاویر، شناسایی الگوهای مختلف و مواردی از این دست را به صورت خودکار انجام دهد [۸].

## ۲-۱ چالش‌های استفاده از پهپاد

استفاده از پهپادها، با چالش‌های متعددی همراه است. یکی از این چالش‌ها، محدودیت زمان پرواز است که پس از مدت کوتاهی پهپادها نیاز به شارژ مجدد دارند. همچنین، محدودیت‌های محیطی نیز می‌توانند به چالش‌های سختی تبدیل شوند؛ زیرا پهپادها به شرایط محیطی مانند آب و هوا و ارتفاع حساس هستند و این موارد می‌توانند در عملکرد آنها تأثیر بهسزایی داشته باشد. در ادامه باید به میزان اهمیت امنیت اطلاعات به دست آمده از پهپادها نیز اشاره کرد، پهپادها به دلیل استفاده از سیستم‌های موقعیت‌یاب و ارتباطات بی‌سیم ممکن است در برابر حملات سایبری آسیب‌پذیر باشند و این آسیب‌پذیری اطلاعات مهمی را که توسط آنها مخابره می‌شود در معرض خطر قرار می‌دهد.

همچنین می‌توان به چالش‌هایی که ما در این پژوهه با آنها سروکار داریم و در تلاشیم که آنها را از بین ببریم و یا کمتر کنیم نیز اشاره کرد. یکی از این چالش‌ها انتقال اطلاعات است. برای ارتباط با پهپادها از شبکه‌های بی‌سیم استفاده می‌شود و در شرایطی مانند اشباع شبکه و یا افزایش فاصله بین پهپاد و کنترل‌کننده، ممکن است این ارتباط دچار اختلال شود. علاوه بر این، محدودیت محاسباتی پهپاد نیز با توجه به اهدافی که برای آن در نظر گرفته شده می‌تواند چالش برانگیز باشد؛ زیرا پهپادها به دلیل محدودیت‌های سخت‌افزاری و نرم‌افزاری، دارای پردازشگرها و حافظه‌های محدودی هستند [۹]. قابل ذکر است که با ادامه پیشرفت فناوری پهپاد، می‌توان انتظار داشت که ویژگی‌های جدید و نوآورانه‌ای برای از بین بردن این محدودیت‌ها و چالش‌ها به پهپادهای آینده اضافه شود.

## ۳-۱ اهمیت علائم دست

هنگام صحبت کردن، افراد از حرکات دست استفاده می‌کنند. حرکت‌های دست جزء اساسی زبان هستند که اطلاعات معنادار و منحصر به فردی را منتقل می‌کنند. این حرکات به گوینده کمک می‌کنند تا اهداف خود را بهتر منعکس کند و نقش‌های مهمی در ارتباط، یادگیری و درک، هم برای افرادی که آن‌ها را مشاهده می‌کنند و هم برای کسانی که آن‌ها را انجام می‌دهند ایفا می‌کنند.

<sup>3</sup>Co-Speech Gestures

به حرکات خود به خودی دست که در ریتم گفتار ایجاد می‌شوند، حرکات هم‌گفتاری<sup>۳</sup> گفته می‌شود. مردم از تمامی فرهنگ‌ها و پیشینه‌های زبانی شناخته شده برای ارتباط بهتر از حرکات هم‌گفتاری استفاده می‌کنند. حتی نوزادان قبل از بیان اولین کلمات خود، از انواع علائم استفاده می‌کنند. دست‌ها به ما کمک می‌کنند صحبت کنیم، فکر کنیم و به خاطر بسپاریم، گاهی اوقات دانشی را که هنوز نمی‌توان به زبان آورد، آشکار می‌کنند. به طوری که می‌توان گفت علائم دست اغلب به عنوان زبان گفتاری ثانویه در نظر گرفته می‌شوند [۱۰].

علائم دست به‌ویژه زمانی مؤثر هستند که مزیتی نسبت به کلمات داشته باشند [۱۱]. توانایی درک شکل و حرکت دست‌ها می‌تواند یک جزء حیاتی در بهبود تجربه کاربر<sup>۴</sup> در حوزه‌ها و پلتفرم‌های مختلف فناوری باشد. درک مفهوم علائم دست به صورت بی‌درنگ برای افراد به طور طبیعی وجود دارد، اما این کار زمانی که توسط هوش مصنوعی و بینایی ماشین رخ دهد می‌تواند چالش‌برانگیز باشد. زیرا دست‌ها اغلب خود و یا یکدیگر را مسدود می‌کنند مانند انسداد انگشتان یا کف دست، حتی لرزش دست نیز می‌تواند مشکلاتی به وجود آورد [۱۲].

## ۴-۱ اهمیت استفاده از بینایی ماشین در پهپاد

طبق اعلام پیش‌بینی اداره هوانوردی فدرال<sup>۵</sup>، بازار هواپیماهای بدون سرنشیین تا سال ۲۰۲۵ به ۱۷ میلیارد دلار خواهد رسید و ۷ میلیون هواپیمای بدون سرنشیین به آسمان پرواز خواهند کرد. پهپادهای کنترل از راه دور به تدریج به دستگاه‌های نیمه خودکار یا کاملاً خودکار تبدیل می‌شوند که از این دستگاه‌ها از پیاده‌سازی‌های مبتنی بر هوش مصنوعی بهره می‌برند. در این پروژه هدف ما هدایت پهپاد با استفاده از علائم دست مبتنی بر بینایی ماشین است که یک حوزه پژوهشی مهم در ترکیب هوش مصنوعی و رباتیک است. استفاده از حرکات دست در کنترل هواپیماهای بدون سرنشیین در حال تبدیل شدن به یک روش محبوب برای تعامل بین کاربر و پهپاد است.

<sup>4</sup>User Experience

<sup>5</sup>Federal Aviation Administration

<sup>6</sup>Real-Time

<sup>7</sup>Accuracy

## ۱-۵ تعریف مسئله

این پایان نامه یک سیستم کامل برای کنترل هواپیماهای بدون سرنشین با استفاده از حرکات دست پیشنهاد می‌کند. سیستم پیشنهادی باید به صورت بی‌درنگ<sup>۶</sup> کار کند و دقت<sup>۷</sup> بالایی را داشته باشد تا بتواند به بهترین نحو ممکن پهپاد را کنترل کند [۶].

در این روش، از سیستم بینایی ماشین به منظور تشخیص و تحلیل حرکات دست از روی تصاویر ویدئویی پهپاد استفاده می‌شود. با استفاده از الگوریتم‌های یادگیری عمیق و شبکه‌های عصبی، سیستم قادر است علائم و حرکات دست را تشخیص داده و به تفسیر آنها بپردازد. سپس، براساس تحلیل این حرکات، دستورات مربوطه برای حرکت و کنترل پهپاد را صادر کند. این روش نه تنها از دقت بالا برای تشخیص و تفسیر حرکات دست برخوردار است، بلکه قابلیت ارائه یک رابط کاربری بین انسان و پهپاد را نیز فراهم می‌کند. به طوری که با استفاده از حرکات دست کاربر قادر است به راحتی و بدون نیاز به دستگاه‌های کنترل خارجی، پهپاد را هدایت کند [۱۳].

استفاده از حرکات دست برای کنترل پهپاد مزایای زیادی دارد. ابتدا باید گفت که حرکات دست یک شکل طبیعی ارتباطی هستند و استفاده از آنها برای کنترل پهپاد یک روش شهودی و طبیعی برای تعامل با فناوری است. این امر باعث می‌شود که کاربران بتوانند به راحتی و با کمترین تلاش پهپاد را کنترل کنند. استفاده از حرکات دست به کاربر اجازه می‌دهد پهپاد را با سرعت و دقت بیشتری کنترل کند و محدودیت‌های مرتبط با دستگاه‌های کنترل سنتی را کاهش دهد. همچنین، این روش حرکت و دنبال کردن پهپاد را آسان‌تر می‌کند و امکان جابجایی پهپاد در فضاهای باز را بهبود می‌بخشد.

استفاده از علائم دست سبب کاهش نیاز به دستگاه‌های کنترلی پیچیده می‌شود و به این ترتیب، پهپاد را برای طیف وسیع‌تری از کاربران قابل دسترس می‌کند. این امر به کاربرانی که با دستگاه‌های کنترل سنتی آشنایی ندارند، امکان استفاده آسان از پهپاد را می‌دهد. همچنین، با توجه به چالش‌هایی که در بخش قبلی بیان شده است، این روش خطرات مرتبط با اتصالات بی‌سیم بین کنترلر و پهپاد را کاهش می‌دهد و دقت کنترل پهپاد در محیط‌های پرتلاظم و متفاوت از نظر آب و هوایی را افزایش می‌دهد.

برای پیاده‌سازی این پروژه از شبکه‌های عصبی عمیق<sup>۸</sup>، مانند شبکه‌های عصبی پیچشی، استفاده شده است. دلیل استفاده از این معما ری‌ها قابلیت استخراج خودکار ویژگی‌ها با توجه به الگوریتم دسته‌بندی تصاویر<sup>۹</sup> است. عملکرد شبکه‌های عصبی پیچشی به این گونه است که ویژگی‌ها را با استفاده از لایه‌های

<sup>8</sup>Deep Neural Network

<sup>9</sup>Image Classification

<sup>10</sup>Hidden Layers

پنهان<sup>۱۰</sup> می‌آموزد، همچنین می‌تواند تعداد پارامترها را بدون به خطر انداختن دقت مدل تغییر دهد.

### ۱-۵-۱ چالش‌های اجرای پروژه

وجود سخت‌افزاری مناسب برای اجرای این پروژه الزامی است. پهپاد انتخاب شده در ابتدا باید شامل یک دوربین با کیفیت تصویر<sup>۱۱</sup> نسبتاً بالا باشد که علائم دست تا فاصله سه متری از پهپاد به وضوح تشخیص داده شود. در ادامه از آنجایی که بی‌درنگ بودن در این پروژه از اهمیت بالایی برخوردار است پهپاد باید پردازنده نسبتاً قوی داشته باشد تا بتواند در کمترین زمان ممکن ویدیو را از دوربین دریافت کرده و انتقال دهد و پس از به دست آوردن خروجی سیستم دستور متناسب را اجرا کند. از دیدگاهی دیگر، از آنجایی که این ارتباطات در کنار حرکت پهپاد انرژی زیادی می‌طلبد، لذا باید پهپادی را انتخاب کرد که از نظر باطری بادوام و باکیفیت باشد تا به مرور زمان برای استفاده کننده آزاردهنده نباشد. همچنین پهپاد مد نظر ما برای این پروژه باید توانایی ارتباط با زبان برنامه‌نویسی پایتون را نیز داشته باشد تا بتوان دستورهای پیش‌بینی شده توسط شبکه‌های هوش مصنوعی پیاده‌سازی شده با این زبان را روی آن اجرا کرد.

### ۱-۶ مراحل انجام پروژه

۱. انتخاب علائم مناسب و مفهومی برای کنترل پهپاد

۲. پیاده‌سازی کد برای جمع‌آوری مجموعه‌داده

۳. پیاده‌سازی شبکه مربوط به پیدا کردن کف دست

۴. پیاده‌سازی شبکه مرتبط با پیدا کردن نقاط کلیدی دست

۵. پیاده‌سازی شبکه‌هایی برای تعیین علائم دست

۶. آموزش شبکه‌ها

۷. بهینه‌سازی شبکه‌ها

۸. تست مدل‌ها و انتخاب بهترین مدل

<sup>11</sup>Resolution

۹. پیاده سازی رأی‌گیری پنجره‌ای<sup>۱۲</sup>

۱۰. اجرای مدل روی پهپاد

## ۷-۱ جمع‌بندی

هدف این پروژه پیاده‌سازی برنامه‌ای کاربردی بر روی پهپاد است، تا به وسیله آن بتوان ۹ علائم دست که از پیش تعیین شده‌اند را شناسایی و با توجه به آنها دستور مرتبط با هر یک را به پهپاد ارسال کرد. در این پروژه موارد زیر از اهمیت بالایی برخوردار هستند:

- دقت بالای شبکه: در صورت انجام نادرست دستورات، امکان از دست رفتن مقدار هنگفتی هزینه مالی یا جانی شامل آسیب‌های وارد به پهپاد و نیروی انسانی وجود دارد.
- پیچیدگی کم و سرعت بالای تشخیص علائم دست: بی‌درنگ بودن اجرای دستورات مهم است و باید در کمترین زمان ممکن رخ دهد تا مورد پسند کاربر باشد.

در فصل بعدی به بررسی کارهای مشابهی که در این زمینه وجود دارند خواهیم پرداخت تا بتوانیم با استفاده از آن‌ها دقت سیستم خود را مقایسه کرده و از نظر عملکردی نتیجه قابل قبولی را به دست آوریم.

---

<sup>12</sup>Window Voting

# فصل دوم

## کارهای مشابه

## ۱-۲ مقدمه

در این فصل هدف ما بررسی کارهای مشابه است تا بتوانیم از آنها در روند بهبود پروژه کمک بگیریم. همچنین با توجه به نتایج و ارزیابی پروژه‌های دیگر می‌توان بستری فراهم کرد تا نتیاج به دست آمده از این پروژه را با دیگر کارهای مشابه مقایسه کرد.

به صورت کلی پروژه‌ای با هدف کنترل پهپاد با علائم دست در ۳ دسته قرار می‌گیرند.

- کنترل پهپاد با کمک بینایی ماشین که شامل شبکه‌هایی برای پردازش تصویر است.

• کنترل پهپاد با دستکش‌های سنسور دار از جمله سنسور تشخیص لختی<sup>۱</sup> که نیازمند سختافزار خاص برای پیدا کردن موقعیت نقاط کلیدی دست است. با استفاده از یک شبکه عصبی و موقعیت این نقاط می‌توان علائم دست را تشخیص داد. مانند پروژه‌های [۱۲] و [۱۴].

• کنترل پهپاد با کمک دستگاه کنترل کننده حرکت جهشی<sup>۲</sup> که با ردیابی دست، ویژگی‌های آن از جمله موقعیت کف دست و انگشتان، جهت دست و انگشتان، طول و عرض انگشتان و موقعیت مفصل و استخوان‌ها را با دقت بالا اندازه‌گیری کرده و با کمک شبکه‌های عصبی علائم دست را تشخیص می‌دهد. پروژه‌های [۱۵] و [۱۶] نمونه‌ای از این نوع پروژه‌ها هستند.

از بین این موارد این پروژه همانند اولین مورد است که تنها سختافزار مورد نیاز پهپاد، دوربین نصب شده بر روی آن است. پروژه‌های مشابه با این پروژه که با کمک پردازش تصویر، پهپاد را کنترل می‌کنند به سه دسته کلی تفکیک می‌شوند. این سه دسته عبارتند از:

۱. استخراج ویژگی‌های تصویر در هر فریم که با توجه به نیازهای مسئله می‌تواند متفاوت باشد، و پیش‌بینی علائم دست براساس ویژگی‌های استخراج شده در مرحله قبل.

۲. تشخیص دست<sup>۳</sup>، برای پیدا کردن موقعیت دست در هر فریم تصویر و سپس ورودی پیکسل‌های دست به صورت قرمز، سیز و آبی<sup>۴</sup> به مدل، و در نهایت کلاس‌بندی علائم دست.

۳. استخراج نقاط کلیدی<sup>۵</sup> دست در تصویر و سپس دادن آنها به عنوان ورودی به مدلی مشخص برای کلاس‌بندی و تشخیص علائم دست.

<sup>1</sup>IMU

<sup>2</sup>Leap Motion Controller

<sup>3</sup>Hand Detection

<sup>4</sup>RGB

<sup>5</sup>Key Point

در ادامه این فصل به بررسی مقالات در این سه زمینه و مقالاتی که بر روی پهپاد و پیاده‌سازی مدل‌های هوش مصنوعی تمرکز دارند می‌پردازیم.

## ۲-۲ کارهای مشابه مربوط به ویژگی‌های تصویر

مقالات به کار برده شده در این قسمت بر چگونگی تعیین علائم دست با توجه به تصاویر ورودی داده شده بر شبکه عصبی تمرکز دارند. برخی از این مقالات چکونگی ارتباط با پهپاد را نیز پوشش می‌دهند، اما نکته قابل بررسی برای ما چگونگی استخراج ویژگی‌های تصویر و استفاده از آنها برای تعیین علائم دست است.

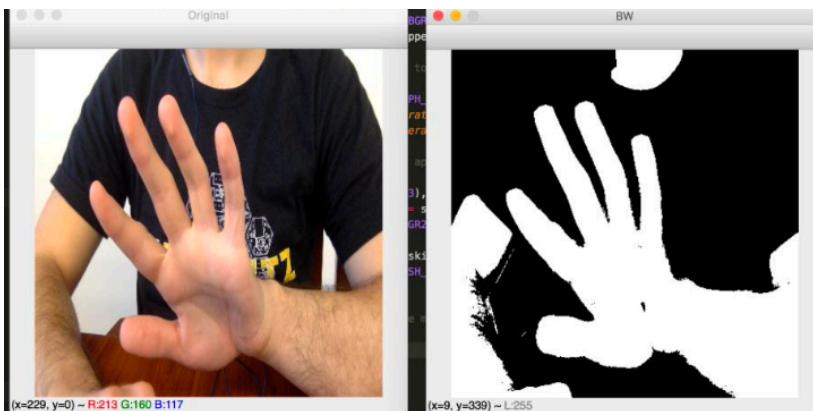
### ۱-۲-۲ پهپادهای کنترل شده با علائم دست به صورت متن باز

در این پروژه، تمرکز بر پیاده‌سازی یک سیستم کنترل برای هواپیماهای بدون سرنشین با استفاده از حرکات دست است که مشابه رویکرد مورد بحث در مقاله می‌باشد. هدف اصلی این پروژه استفاده از شبکه‌های عصبی یادگیری عمیق برای تشخیص لحظه‌های حرکات پویای دست برای کنترل پرواز پهپاد است. این تشخیص بر اساس ویژگی‌های هار<sup>۶</sup> که با توجه به سایه‌ها و رنگ‌های درون تصویر تعیین می‌شوند، انجام می‌شود [۵].

### روش‌شناسی

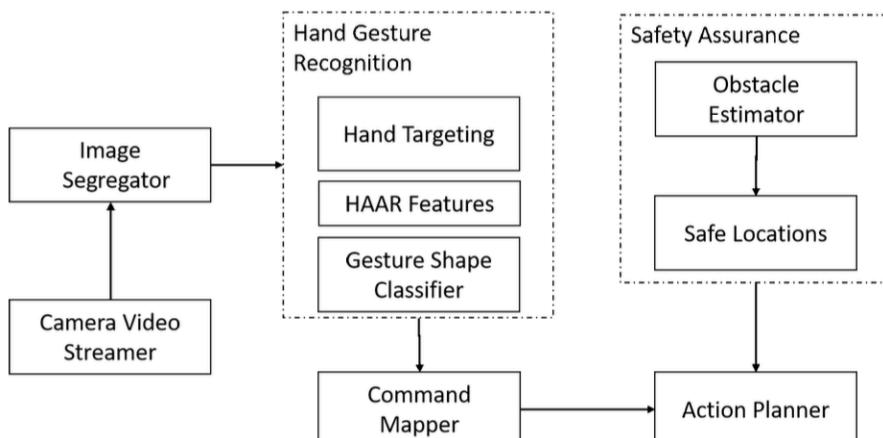
در این پروژه، ابتدا از یک شبکه عصبی برای تشخیص موقعیت دست استفاده می‌شود که به عنوان یک مازول پیش‌پردازش عمل می‌کند و با دقت بالا موقعیت دست را تشخیص می‌دهد. پس از تشخیص موقعیت دست، ویژگی‌های هار از تصویر استخراج می‌شوند. روش تشخیص آن‌ها شامل مجموعه‌ای از الگوریتم‌های تشخیص ویژگی هستند که از تصاویر استفاده می‌کنند تا ویژگی‌ها را شناسایی کنند. ویژگی‌های هار بر اساس تغییرات گرادیان در تصویر تعیین می‌شوند و به عنوان الگوهای محلی برای تشخیص حرکات دست استفاده می‌شوند. در ادامه، از شبکه‌های عصبی یادگیری عمیق برای تشخیص حرکات دست پویا برای کنترل پرواز پهپاد استفاده می‌شود. این شبکه‌ها عملکرد پیچیده‌ای دارند و با استفاده از ویژگی‌های هار به عنوان داده‌ی ورودی آموزش می‌بینند تا حرکات دست را تشخیص دهند و بر اساس آنها دستورات برای حرکت پهپاد را تعیین کنند.

<sup>6</sup>Haar



شکل ۲-۲ ویژگی‌های هار برای استفاده از آستانه رنگ پوست برای تشخیص دست [۵]

این سیستم شامل مراحل پیش‌پردازش داده‌ها، انتخاب ویژگی، مازول شبکه عصبی برای تشخیص علائم دست و مازول کنترل پهپاد برای ترجمه علائم شناسایی شده به دستورات حرکت پهپاد می‌باشد. علاوه بر این، از مدل ماشین بردار پشتیبان<sup>۷</sup> برای کلاس‌بندی و تشخیص حرکات دست استفاده شده است. ماشین بردار پشتیبان به عنوان یک ماشین یادگیری که برای مسائل دسته‌بندی و رگرسیون استفاده می‌شود و در این پروژه برای تشخیص حرکات دست و ترجمه آن‌ها به دستورات حرکت پهپاد مورد استفاده قرار گرفته است. این روش امکان کنترل دقیق و پویا برای پهپاد را فراهم می‌کند و از قابلیت‌های پیشرفته یادگیری عمیق برای تشخیص حرکات دست بهره می‌برد.



شکل ۲-۲ چارچوب کنترل پهپاد مبتنی بر علائم دست با کمک ویژگی‌های هار [۵]

<sup>7</sup>Support Vector Machine

## نتایج بدست آمده

این پروژه دقت بالایی در تشخیص علائم دست و کنترل پرواز پهپاد دارد. پنج حالت دست در این پروژه مدنظر قرار گرفته‌اند و دقت متوسط این مدل، برابر ۹۷.۴۷۱ درصد است که نشان‌دهنده عملکرد بسیار عالی آن است. اما لازم به ذکر است که این دقت در پس‌زمینه‌های بهم ریخته و شرایط نوری مختلف بسیار متغیر است زیرا ویژگی‌های هار به سایه و رنگ‌های درون تصویر بسیار حساس هستند.

### ۲-۲-۲ روش‌های تشخیص علائم دست به صورت بی‌درنگ

این مقاله در زمینه پردازش تصویر و تشخیص علائم دست، از روش‌های مبتنی بر مدل‌های ظاهری<sup>۸</sup> به عنوان یک رویکرد موثر استفاده کرده‌است. این روش از ویژگی‌های تصویری و حرکتی دست برای تشخیص و تعیین علائم دست استفاده می‌کند. در این مقاله، روش تشخیص علائم دست به صورت بی‌درنگ و قابل اعتماد است که از تشخیص دست، ردیابی دست<sup>۹</sup>، تقسیم‌بندی دست<sup>۱۰</sup> و تشخیص ویژگی‌های مقیاس-فضا<sup>۱۱</sup> برای تشخیص علائم استفاده می‌کند [۱۷].

#### روش‌شناسی

در این مقاله از ترکیب متنوعی از روش‌ها و ویژگی‌های تصویری استفاده می‌شود. این ویژگی‌ها به ترتیب به صورت زیر عمل می‌کنند.

۱. استفاده از روش آدابوست<sup>۱۲</sup> برای تشخیص دست، که یک روش معتبر برای تشخیص اشیاء در تصویر است.

۲. پیگیری دست با استفاده از تشخیص حرکت و رنگ، که از ترکیب تکنیک‌های جریان نوری و نشانه رنگ برای ردیابی دست در تصاویر استفاده می‌شود.

۳. تقسیم‌بندی دست با استفاده از اطلاعات حرکت و رنگ برای تمایز دست از پس‌زمینه و اشیاء دیگر.

<sup>8</sup>Appearance Model

<sup>9</sup>Hand Tracking

<sup>10</sup>Hand Segmentation

<sup>11</sup>Scale-Space Feature Detection

<sup>12</sup>AdaBoost

۴. تشخیص ویژگی‌های مقیاس-فضا برای تشخیص علائم دست، که برای شناسایی ساختارهای شبیه به کف دست و انگشتان استفاده می‌شود تا نوع علائم دست توسط ترکیب این ساختارها تعیین شود.

در نهایت این روش‌ها و ویژگی‌ها با هم ترکیب می‌شوند تا یک سیستم تشخیص دست پایدار و دقیق برای استفاده در رابط کاربری تعاملی و تشخیص علائم دست در زمینه‌های مختلف ارائه شود.



شکل ۲-۳ تشخیص کف دست و انگشتان [۱۷]

#### نتایج بدست آمده

اعمال این روش‌ها نتایج قابل قبولی را به همراه داشته است. دقت مدل در تشخیص علائم دست به صورت میانگین ۹۳.۸ درصد بوده و از جمله نتایج مهم آزمایشات می‌توان به تشخیص صحیح ۲۴۳۶ فریم از کل ۲۵۹۶ فریم ضبط شده اشاره کرد. این نتایج نشان می‌دهند که روش ارائه شده در این مقاله عملکرد قابل قبولی در تشخیص علائم دست دارد و می‌تواند به عنوان یک روش موثر برای تعاملات بی‌درنگ استفاده شود.

### ۳-۲ کارهای مشابه مربوط به ورودی تصویر دست به مدل

مقالات این دسته از جمله پژوههایی هستند که بیشتر از آنکه بر روی پیش پردازش کار کنند باید بر روی معماری خود شبکه تمرکز کنند. در این مقالات تمام یا بخشی از تصویر گرفته شده به صورت یک ماتریس از تصویر با پیکسل‌های متعدد به مدل داده می‌شود. رویکردی که در این نوع پژوهه‌ها در پیش گرفته می‌شود، تشخیص موقعیت دست است تا بتوان تنها قسمتی از تصویر را به ورودی شبکه داد که دست در آن وجود دارد تا در حد ممکن اندازه ورودی شبکه و دقت آن افزایش یابد. در اینگونه مقالات باید توجه داشت که معماری شبکه را به گونه‌ای برگزید تا در پردازش تصویر بتوان ویژگی‌های مورد نیاز

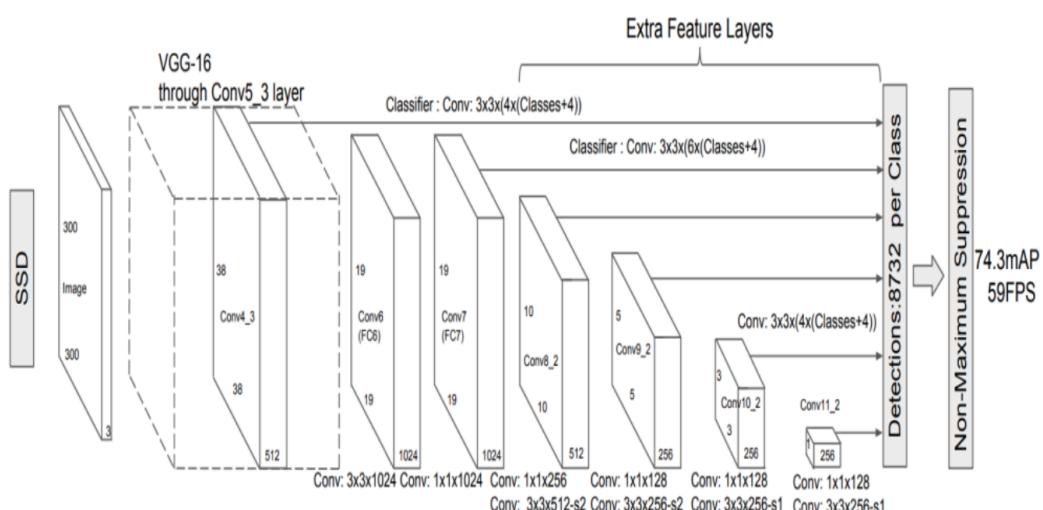
را استخراج کرد.

### ۱-۳-۲ کنترل پهپاد با تشخیص علائم دست با کمک یادگیری عمیق

این پروژه نیز با هدف کنترل پهپاد با استفاده از حرکات دست و شبکه‌های عمیق یادگیری انجام شده است تا بتواند ۹ حالت مختلف دست را شناسایی و به پهپاد، دستور موردنظر کاربر را ارسال کند [۶].

#### روش شناسی

در این تحقیق، از معماری شبکه عصبی عمیق VGG-16 برای تشخیص و تعیین حرکات دست و کنترل پهپاد استفاده شده است. شبکه ۱۶ VGG یکی از معروف‌ترین و پرکاربردترین شبکه‌های عمیق در زمینه بینایی ماشین است که شامل ۱۶ لایه عصبی با لایه‌های پیچشی و ادغام <sup>۱۳</sup> می‌باشد. این شبکه برای استخراج ویژگی‌های مهم از تصاویر، استفاده می‌شود. ورودی این شبکه تصاویری است که از دوربین متصل به پهپاد گرفته می‌شود و سپس این تصاویر به شبکه وارد می‌شوند. خروجی این شبکه شامل تشخیص حرکات دست مانند حرکات مختلف انگشتان و دست‌ها است. سپس این اطلاعات برای ارسال دستورات کنترلی به پهپاد استفاده می‌شوند.



شکل ۴-۲ معماری VGG-16 [۶]

<sup>13</sup>Pooling

### نتیجه بدست آمده

در این پژوهه ۹ حالت دست مدنظر قرار گرفته شده و دقت آن برابر ۸۳.۳ درصد است. که در بهترین حالت ممکن با پس زمینه‌ی مناسب بدست آمده و باید در نظر گرفت که این عدد دقت بالایی برای کنترل پهپاد به حساب نمی‌آید.

### ۲-۳-۲ کنترل یوای‌وی با تشخیص علائم دست

این مقاله به منظور کنترل پهپاد یا خلبان خودکار با استفاده از حرکت دست پیاده‌سازی شده است. به عنوان مثال، حرکت دست از چپ به راست نشان‌دهنده حرکت پهپاد به راست می‌باشد. برای اجرای این برنامه، شبکه P-CNN طراحی شده است تا بتواند مفهوم تصاویر را ترجمه کند [۱۸].

#### روش‌شناسی

در این مقاله، از شبکه P-CNN<sup>۱۴</sup> برای تشخیص حرکات دست استفاده شده است. این شبکه از اطلاعات علائم و حرکت دست و بخش‌هایی از بدن استفاده می‌کند. بدین صورت که ابتدا موقعیت دست را با استفاده از جعبه مرزی<sup>۱۵</sup> مشخص می‌کند و سپس تصویر دست را با استفاده از فیلترهای مناسب وارد شبکه P-CNN می‌کند تا بتواند هدف کاربر از حرکت دست را پیش‌بینی کند.

در خروجی این مدل، ۱۳ نوع حرکت مختلف وجود دارد که توسط مدل، مفهوم آنها پیش‌بینی می‌شود. این حرکات شامل کل دست از شانه تا انگشتان و حرکات آنها است. این مدل در پژوهه برای دستور دادن به هواپیماهای بزرگ بدون سرنشین در فرودگاه‌ها استفاده می‌شود.

P-CNN اطلاعات ظاهر و حرکت را از بخش‌های مختلف بدن استخراج و از دو شبکه پیش‌آموزش داده شده برای محاسبه ویژگی‌های شبکه عصبی پیچشی استفاده می‌کند. در این پژوهه برای بخش‌های علائم ظاهری، از شبکه VGG-f استفاده می‌شود، و برای بخش‌های حرکتی از شبکه Action Tube کمک گرفته می‌شود. در نهایت با ترکیب این دو شبکه با استفاده از روش‌های تجمعی مینیمم و ماکسیمم در هر بعد از تصویر و سپس کنار هم گذاشتن آنها می‌توان حرکت دست را با دقت بالا تشخیص داد.

<sup>14</sup>Pose-Based Convolutional Neural Network

<sup>15</sup>Bounding Box

### نتیجه بدست آمده

این مقاله با استفاده از شبکه P-CNN روشی برای کنترل پهپاد با استفاده از حرکت و علائم دست پیاده‌سازی کرده است. نتایج نشان داده‌اند که این روش با دقت ۹۱.۹ درصد، قابلیت اجرا در پروژه‌های واقعی را دارد. این پروژه نیازمندی‌های پیچیده‌ای برای پیاده‌سازی به همراه دارد که باید در نظر گرفته شود. این نیازها شامل سخت‌افزار و نرم‌افزارهای پیشرفته برای پردازش داده‌های حجمی و پیچیده، تنظیمات محیطی مانند نورپردازی و شرایط جوی، تنظیمات دقیق برای تشخیص حرکات و علائم، و مسائل امنیتی مرتبط با کنترل پهپاد است. این چالش‌ها نشان از پیچیدگی‌هایی است که برای اجرای موفق این روش در پروژه‌های واقعی باید در نظر گرفته شوند.

## ۴-۲ کارهای مشابه مربوط به نقاط کلیدی دست

در این چنین مقالات ورودی شبکه بینایی ماشین برای تشخیص علائم، مختصات نقاط کلیدی دست هستند، که به نوعی یکی از ویژگی‌های تصویر ورودی نیز تلقی می‌شوند. بدین صورت که در ابتدای کار دست کاربر شناسایی شده و سپس نقاط کلیدی آن استخراج می‌شوند تا بتوان حجم داده ورودی به مدل را تا حد امکان کاهش داده و در عین حال داده‌های ورودی را مفیدتر کرد.

### ۱-۴-۲ تشخیص علائم دست برای استفاده‌های بی‌درنگ

این مقاله به بررسی سیستم تشخیص حرکات دست به صورت بی‌درنگ می‌پردازد. در این سیستم، از الگوریتم سیفت<sup>۱۶</sup> برای استخراج ویژگی‌ها از تصاویر حرکتی استفاده شده است. سپس، از مدل دسته‌ای از ویژگی‌ها<sup>۱۷</sup> و ماشین بردار پشتیبانی برای تشخیص دقیق حرکات دست به صورت بی‌درنگ بهره گرفته شده است [۱۹].

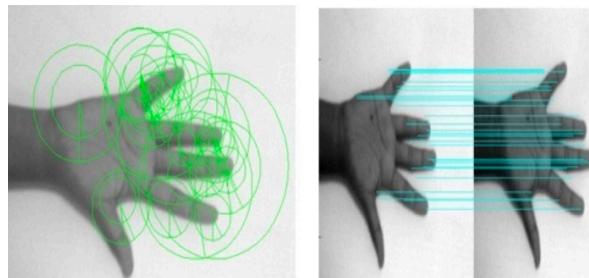
### روش‌شناسی

همان‌طور که گفته شد در این مقاله از الگوریتم سیفت برای استخراج نقاط کلیدی از تصاویر حرکتی دست استفاده شده است. الگوریتم سیفت یک الگوریتم معروف برای استخراج ویژگی‌های برجسته و تمایزدهنده از تصاویر است که از مقیاس، جهت و از تغییرات نوری برای استخراج این ویژگی‌ها استفاده

<sup>16</sup>SIFT

<sup>17</sup>Bag of Features

می‌کند. این ویژگی‌ها به طور قابل توجهی مستقل از مقیاس و جهت تصویر هستند و می‌توانند برای کاربردهای گوناگونی از جمله تطبیق قابل اعتماد بین تصاویر مختلف یک شیء مورد استفاده قرار گیرند.



شکل ۵-۲ تشخیص نقطه کلید و تطبیق توسط سیفت [۱۹]

#### نتیجه بدست آمده

الگوریتم سیفت به عنوان یک ابزار قدرتمند برای استخراج ویژگی‌های برجسته و تمایزدهنده از تصاویر، شناخته شده‌است. این پروژه توانایی تشخیص حرکات دست را با دقت ۹۰.۸ درصد دارد. در این مقاله از روشی نوین و متفاوت برای رسیدن به علائم دست استفاده شده‌است، اما قابل ذکر است که این دقت برای پروژه ما می‌تواند خطای زیادی داشته باشد و در روند پروژه استفاده از این الگوریتم کمک‌کننده نیست.

#### ۲-۴-۲ تشخیص علائم دست با استفاده از نقاط کلیدی و جعبه مرزی

این مقاله یک سیستم بهبود یافته تشخیص حرکات دست با استفاده از نقاط کلیدی و جعبه‌های محدود کننده دست را معرفی می‌کند [۲۰].

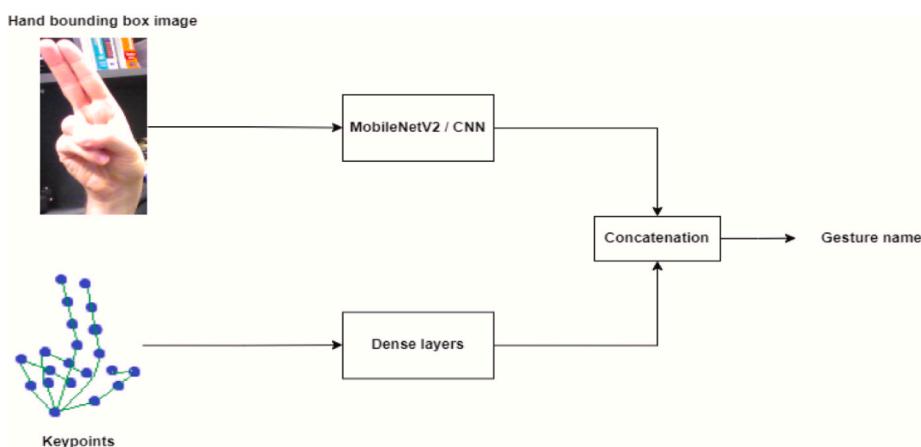
#### روش‌شناسی

این پروژه از دو لوله موازی به نام‌های "CNN + FC" و "MobileNetV2 + FC" تشکیل شده‌است که تصاویر جعبه محدود کننده دست و ویژگی‌های استخراج شده از نقاط کلیدی را ترکیب می‌کند و از این طریق علائم دست را پیش‌بینی می‌کند.

در مدل MobileNetV2 + FC، از یک معماری سبک به نام MobileNetV2 برای استخراج ویژگی‌ها از تصاویر جعبه محدود کننده دست استفاده می‌شود. سپس این ویژگی‌ها به یک شبکه عصبی کاملاً

<sup>18</sup>Fully Connected

متصل<sup>۱۸</sup> (FC) داده می‌شوند تا حرکات دست تشخیص داده شوند. در مدل CNN + FC، در ابتدا نقاط کلیدی دست را که اطلاعات مهمی درباره حرکات دست را شامل می‌شوند پیدا کرده و سپس نقاط به یک شبکه عصبی کاملاً متصل وارد می‌شوند تا حرکات دست تشخیص داده شوند. این مدل از لایه‌های پیچشی برای کاهش تعداد پارامترها استفاده می‌کند و سپس از لایه‌های کاملاً متصل برای تشخیص حرکات دست استفاده می‌کند.



[۲۰] شکل ۶-۲ معماری ساختار شبکه‌های عصبی دو خط لوله

#### نتیجه بدست آمده

در صورتی که دو مدل خروجی‌های متفاوتی را پیش‌بینی کنند، از روش‌های ترکیبی مانند ترکیب احتمالاتی استفاده می‌شود تا بهترین تصمیم برای تشخیص علائم دست گرفته شود. دقیق به دست آمده برای تشخیص ۶ علائم دست مختلف در این مقاله در سه مجموعه داده متفاوت به ترتیب برابر ۹۴، ۹۱ و ۹۶ درصد است.

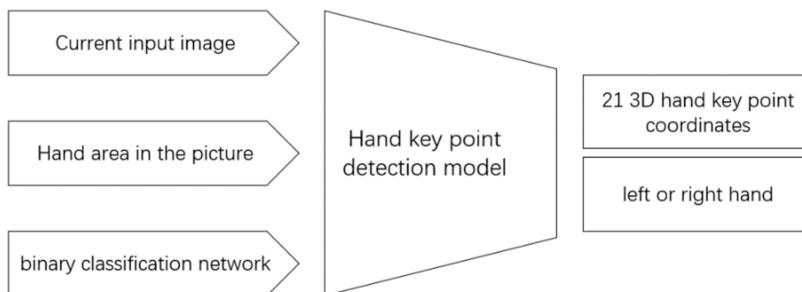
### ۳-۴-۲ تشخیص علائم بصری مبتنی بر نقاط کلیدی

این مقاله یک روش تشخیص حرکات علائم بصری بر اساس نقاط کلیدی دست ارائه می‌دهد. این روش ابتدا نقاط کلیدی دست را در تصویر ورودی فعلی تشخیص می‌دهد و سپس حرکات تعریف شده را پیش‌بینی می‌کند [۲۱].

#### روش‌شناسی

مدل ارائه شده در این مقاله شامل دو بخش اصلی است:

- مدل تشخیص کف دست که بر اساس ویژگی های سخت دست طراحی شده است. برای تشخیص حضور دست در تصویر، از یک مدل SSD استفاده شده است که به صورت بی درنگ تشخیص درست را انجام می دهد. و نتیجه آن به صورت یک مستطیل نشان داده می شود.
- مدل تشخیص نقاط کلیدی دست که پس از تشخیص حضور دست در تصویر، از این مدل برای تشخیص ۲۱ مختصات نقطه کلیدی دست استفاده می شود. این مدل از الگوریتم نرمال سازی برای محاسبه مختصات افقی و عمودی هر نقطه دست استفاده می کند. یک سیستم مختصات فضایی برای به دست آوردن مختصات عمق هر نقطه دست نسبت به مبدأ مختصات تعیین شده است. در نهایت، معنای حرکات در تصویر ورودی بر اساس رابطه مکانی بین مختصات نقاط کلیدی تعیین می شود.



شکل ۷-۲ ورودی ها و خروجی های مدل تشخیص نقاط کلیدی دست [۲۱]

#### نتیجه بدست آمده

این روش دارای دقت بالا و عملکرد مناسبی است. دقت متوسط مدل برابر  $85.4\%$  درصد تا  $98.5\%$  درصد است که با توجه به جزئیات مدل و پیاده سازی پیش پردازش می تواند انعطاف بالایی داشته باشد در نتیجه می تواند راهکار مناسبی تلقی شود.

#### ۴-۴-۲ تشخیص بی‌درنگ دست با کمک کتابخانه مدياپایپ

در این مقاله، از کتابخانه مدياپایپ<sup>۱۹</sup> برای پیش‌بینی ۲۱ نقطه کلیدی دست استفاده شده است. این پروژه در موارد گوناگونی از جمله تشخیص علائم دست و افکت‌های واقعیت افزوده<sup>۲۰</sup> کاربرد دارد [۱۲].

#### روش‌شناسی

برای پیش‌بینی علائم دست در این پروژه، از دو شبکه پیچشی استفاده شده است. شبکه اول برای پیدا کردن کف دست در تصویر استفاده می‌شود و شبکه دوم ورودی موقعیت عکس دست پیدا شده را دریافت و مختصات ۲۱ نقطه عطف را پیدا می‌کند. به کمک این دو شبکه، می‌توان به طور همزمان موقعیت دست‌ها را تشخیص داد و نقاط عطف آن‌ها را پیش‌بینی کرد تا برای تشخیص علائم دست در پروژه‌های واقعیت افزوده و واقعیت مجازی استفاده شود.

#### نتیجه بدست آمده

مدل‌های طراحی شده در این مقاله برای تشخیص نقاط عطف دست از دقت ۹۵.۷ درصد برخوردار هستند که دقت بسیار بالایی محسوب می‌شود. این مدل به نور و تصویر پس‌زمینه وابسته نیست و دقت متوسط آن در زمینه‌های مختلف اندازه‌گیری شده، لذا این موارد مدل را کاربردی‌تر می‌کند.

### ۵-۲ کارهای مشابه پهپاد و ترکیب آن با مدل بینایی ماشین

#### ۱-۵-۲ مدل‌های مبتنی بر هوش مصنوعی بر روی پهپادها

این مقاله به بررسی ارتباط بین استفاده از پهپادهای مبتنی بر هوش مصنوعی و موفقیت پروژه‌های ساخت‌وساز پایدار می‌پردازد. این تحقیقات ثابت ادغام این دو عنصر را در بهبود کارایی و پایداری پروژه‌های ساختمانی بررسی می‌کند. با توجه به افزایش سریع استفاده از پهپادها در صنعت، این مقاله به شناخت دقیقی از ارتباط بین این دو از جمله موانع و موفقیت‌ها می‌پردازد. این شناخت به شناسایی بهترین روش‌ها برای از بین بردن موانع و تضمین موفقیت پهپادها در ترکیب هوش مصنوعی کمک می‌کند [۲۲].

<sup>19</sup>MediaPipe

<sup>20</sup>Augmented Reality

### روش‌شناسی

در این تحقیق، از مدل معادلات ساختاری هوشمند برای بررسی موانع و موفقیت اجرای پهپادهای مبتنی بر هوش مصنوعی در صنعت استفاده شده است. این مدل شامل یک مجموعه جامع از عوامل موفقیت شامل کیفیت، ایمنی و عوامل محیطی است.

پیاده‌سازی مدل‌ها بر روی پهپادها چالش‌هایی ایجاد می‌کند که باید مورد توجه قرار گیرد. این چالش‌ها شامل موارد زیر می‌شود:

- **پیچیدگی فنی:** پیاده‌سازی مدل‌های هوش مصنوعی بر روی پهپادها نیازمند دانش و تخصص فنی بالا است. این امر نیازمند همکاری بین متخصصان مختلف از حوزه‌های مختلف می‌باشد.
- **حدودیت‌های سخت‌افزاری:** پهپادها ممکن است دارای محدودیت‌های سخت‌افزاری مانند ظرفیت پردازشی و حافظه باشند که این محدودیت‌ها موانعی را برای پیاده‌سازی مدل‌های پیچیده ایجاد می‌کنند.
- **امنیت و حریم خصوصی:** استفاده از هوش مصنوعی در پهپادها نیازمند رعایت استانداردهای امنیتی و حفظ حریم خصوصی است. این امر می‌تواند یک چالش مهم برای پیاده‌سازی مدل‌های هوش مصنوعی باشد.
- **آموزش و توسعه مدل‌ها:** پیاده‌سازی مدل‌های هوش مصنوعی بر روی پهپادها نیازمند آموزش و توسعه مدل‌های مناسب برای محیط و وظایف خاص پهپادها است.

این چالش‌ها نشان‌دهنده اهمیت اصلی پیاده‌سازی مدل‌های هوش مصنوعی بر روی پهپادها در صنعت است. با غلبه بر این چالش‌ها، می‌توان بهبود قابل توجهی در کارایی، ایمنی و پایداری این گونه پروژه‌ها داشت.

### نتیجه بدست آمده

این مقاله نه تنها به شناخت بهتر موانع استفاده از پهپادهای مبتنی بر هوش مصنوعی کمک می‌کند، بلکه راهکارهایی برای غلبه بر این موانع و افزایش موفقیت این تکنولوژی در صنعت ساختمان ارائه می‌دهد. از آنجا که پهپادها می‌توانند به صورت خودکار و هوشمند وظایف مختلفی را انجام دهند، این تکنولوژی می‌تواند به بهبود مدیریت پروژه، کاهش هزینه‌ها و زمان اجرا، افزایش کیفیت و ایمنی کارها کمک کند.

## ۲-۵-۲ پهپاد دیجی‌آی تلو در زمینه کنترل مهندسی

این مقاله یک رویکرد نوآورانه را برای استفاده از پهپاد دیجی‌آی تلو<sup>۲۱</sup> به عنوان یک پلتفرم آموزشی در زمینه مهندسی کنترل ارائه می‌دهد. این مقاله به بررسی نحوه استفاده از ویژگی‌های پهپاد برای آموزش مفاهیم کنترل به صورت عملی و جذاب و همچنین چگونگی استفاده از پهپاد دیجی‌آی تلو به عنوان یک پلتفرم آموزشی برای مفاهیم کنترل می‌پردازد [۲۳].

### روش‌شناسی

در این مقاله، از پهپاد دیجی‌آی تلو به عنوان یک ابزار آموزشی و تحقیقاتی استفاده شده است. این پهپاد به دلیل داشتن حسگرهای متنوع و امکان برنامه‌نویسی با زبان برنامه‌نویسی پایتون، به عنوان یک ابزار ایده‌آل و انعطاف‌پذیر برای اهداف آموزشی و تحقیقاتی شناخته می‌شود. مقاله به بررسی ارتباط با پهپاد دیجی‌آی تلو از طریق زبان برنامه‌نویسی پایتون، امکانات بسته توسعه نرم‌افزار<sup>۲۲</sup> رسمی ارائه شده توسط پهپاد، و نحوه ارتباط با پهپاد از طریق وای‌فای و پورت UDP می‌پردازد.

در این مقاله، انتخاب پهپاد دیجی‌آی تلو برای نمایش مفاهیم ابتدایی مربوط به حوزه کنترل به دلایل گوناگونی انجام شده است. از جمله دلایل انتخاب این پهپاد، محبوبیت رو به افزایش آن در بین عموم مردم، ویژگی‌های چندگانه آن شامل حسگرهای کنترل بازخورد و الگوریتم‌های تخمین برای انجام وظایف پیچیده، و همچنین قیمت مقرن به صرفه آن نسبت به سایر تجهیزات آموزشی است.

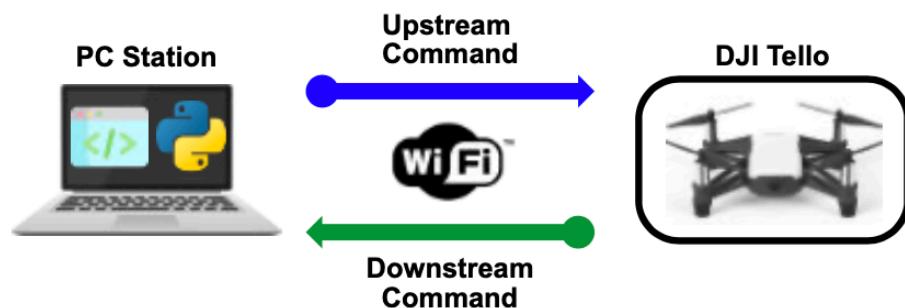
برای پیاده‌سازی مدل‌های هوش مصنوعی بر روی پهپاد دیجی‌آی تلو، از زبان برنامه‌نویسی پایتون و کتابخانه‌های مختلفی که برای ارتباط با حسگرهای کنترل پروازی پهپاد ارائه شده استفاده می‌شود. موارد گفته شده این امکان را می‌دهد که مدل‌های هوش مصنوعی توانایی پیاده‌سازی بر روی پهپاد را داشته باشند و بتوانند عملکرد آن را تحلیل کنند. به عنوان مثال، کاربران می‌توانند از شبکه‌های عصبی یا منطق فازی برای کنترل حرکت پهپاد استفاده کنند.

### نتیجه بدست آمده

نتیجه این مقاله نشان می‌دهد که استفاده از هوش مصنوعی بر روی پهپاد دیجی‌آی تلو عملکرد خوبی را از خود نشان می‌دهد و این ابزار آموزشی و تحقیقاتی می‌تواند به افراد کمک کند تا مفاهیم پیچیده کنترل و هوش مصنوعی را به صورت عملی و جذاب فرا بگیرند.

<sup>21</sup>DJI Tello

<sup>22</sup>Software Development Kit (SDK)



شکل ۲-۸ ارتباط برنامه پایتون با پهپاد DJI Tello [۲۳]

## ۶-۲ جمع‌بندی

در این فصل به بررسی چهار دسته متفاوت از مقالات پرداخته شد

- مقالاتی که با استفاده از استخراج ویژگی‌های تصویر از جمله نور، جهت، موقعیت و ... توانسته‌اند علائم دست را تشخیص دهند. این مقالات برای کلاس‌بندی تعداد علائم محدود و گوناگون از یکدیگر به خوبی عمل می‌کنند، اما در زمانی که تعداد علائم زیاد شده و حالات دست نزدیک به هم باشند، این دقیقتهای صورت چشم‌گیری کاهش پیدا می‌کند. از آنجایی که تمرکز پروژه ما تعیین علائم دست بدون محدودیت تعداد و حالت است این راهکارها عملکرد خوبی را از خود نشان نمی‌دهند.
- مقالاتی مربوط به ورودی تصویر دست، نه تنها دقیق‌تر از خود نشان نمی‌دهند، بلکه زمان اجرای آنها نسبت به دیگر راهکارها بسیار بیشتر است و می‌تواند پروژه را از بی‌درنگ بودن آن که یکی از بزرگ‌ترین اهداف است دور کند.
- مقالات مربوط به تعیین علائم دست با استفاده از نقاط کلیدی عملکرد بسیار خوب و دقیق‌تر از خود نشان داده‌است. این مقالات با توجه به اینکه از چندین مدل پی در پی برای تعیین علائم کمک می‌گیرند از دقیق‌تر از خود بخوبی برخوردارند، همچنین حجم پایین این مدل‌ها باعث می‌شود تا بتوان در زمان کوتاهی علائم دست را پیش‌بینی کرد. این مقالات راهکارهای مناسبی را برای کمک به پیاده‌سازی پروژه ما ارائه دادند.
- مقالات مربوط به آشنایی با پهپاد و اجرای مدل‌های بینایی ماشین روی آن دیدگاه جامعی در مورد پهپاد، مشخصات مورد نیاز آنها و چالش‌هایی برای اجرای این پروژه به ما نشان دادند.

در تمام مقالات بررسی شده، پروژه‌هایی با هدف تعیین علائم دست به گونه‌ای پیاده‌سازی شده‌اند تا حرکات را به گونه‌ای کلاس‌بندی کنند که در خروجی حتماً یکی از علائم درنظر گرفته‌شده انتخاب شود. لذا زمانی که دست در حالتی غیر از آنها قرار دارد، مدل طراحی شده حتماً یکی از علائمی را که به آن شبیه‌تر است انتخاب می‌کند که این امر میتواند برای پیاده‌سازی روی پهپاد واقعی مشکل‌زا باشد و حتی هزینه مالی را به ارمغان آورد. برای پیاده‌سازی پروژه ما باید مسیری را در پیش بگیریم تا بتوانیم این مشکل را برطرف کرده و در عین حال دقت بالایی داشته باشیم. در فصل بعدی به بررسی دقیق داده‌های استفاده شده در این پروژه، نحوه انجام آن و پیاده‌سازی‌های انجام شده می‌پردازیم.

## فصل سوم

### روش انجام پروژه

### ۱-۳ مقدمه

برای اجرای این پروژه، روش‌های مختلفی مورد بررسی قرار گرفت تا بتوان بهترین روش را به طور موثر بر روی پهپاد پیاده‌سازی کرد. در نهایت، استفاده از سه شبکه عصبی پیچشی به صورت متوالی به عنوان بهترین راه حل انتخاب شد. این سه شبکه به ترتیب وظایف زیر را بر عهده دارند:

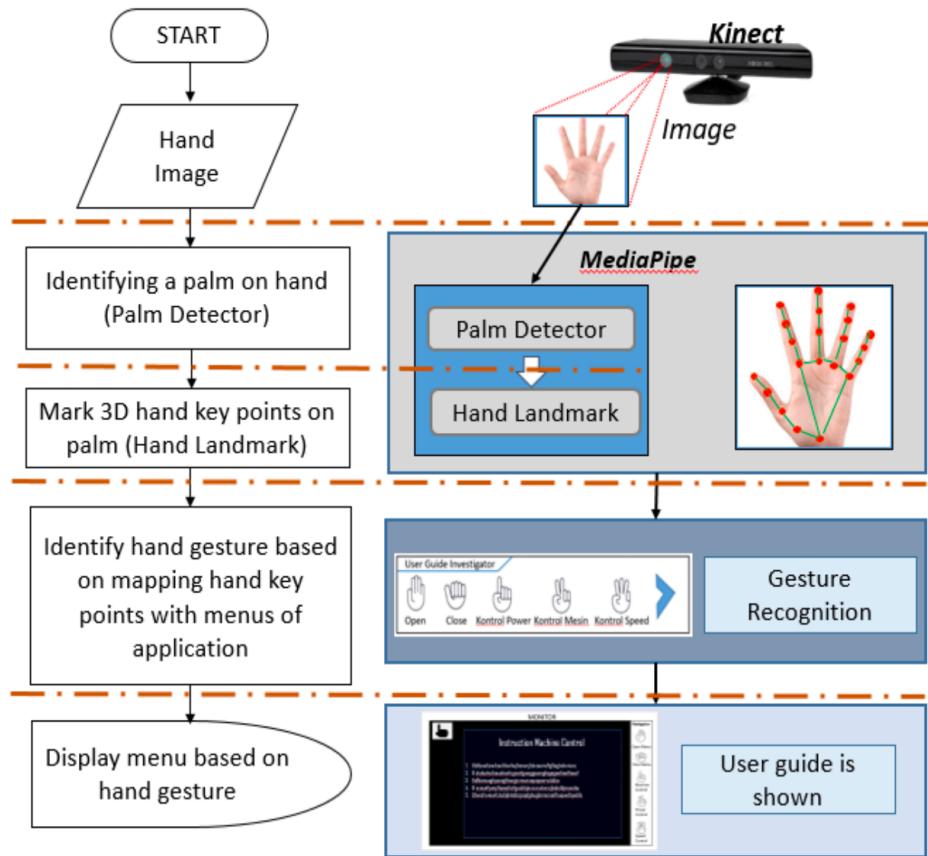
۱. آشکارسازی موقعیت کف دست: هر فریم گرفته شده از دوربین پهپاد به ورودی این مدل داده می‌شود. پس از پردازش، خروجی مدل یک جعبه محدود کننده<sup>۱</sup> دست را تولید می‌کند که موقعیت دقیق دست را در تصویر مشخص می‌سازد.

۲. تشخیص نقاط کلیدی دست: جعبه مرزی دست که از مدل اول به دست آمده است، به عنوان ورودی به مدل دوم می‌شود. این مدل جعبه مرزی را برش زده و ۲۱ نقطه کلیدی سه‌بعدی دست به همراه شاخص دست (راست یا چپ) را تولید می‌کند، که این ویژگی‌ها شامل مفاصل انگشتان و نقاط مهم دست می‌باشند.

۳. پیش‌بینی علائم دست: ورودی این مدل، یک ماتریس به ابعاد  $2 \times 21$  است که مختصات طول و عرض هر نقطه کلیدی دست را شامل می‌شود. به دلیل محدودیت پهپادها در اندازه‌گیری عمق تصویر و اهمیت کمتر عمق در تشخیص علائم مدنظر، فقط مختصات دو بعدی استفاده می‌شود. خروجی این مدل، پیش‌بینی علائم دست کاربر است. این پروژه ۹ علائم گوناگون را مد نظر قرار داده است (کاربر می‌تواند علائم جدیدی اضافه کند). بنابراین، خروجی شبکه پیچشی شامل ۱۰ کلاس است که ۹ کلاس برای علائم مختلف و یک کلاس برای زمانی که هیچ کدام از علائم موجود تشخیص داده نشود، در نظر گرفته شده است.

این سه مدل با همکاری یکدیگر، از تشخیص مختصات دست تا پیش‌بینی علائم دست را به صورت بهینه و کارآمد انجام می‌دهند، به طوری که می‌توانند روی پهپادهای موجود پیاده‌سازی شوند و در کاربردهای واقعی مورد استفاده قرار گیرند. بهینه‌سازی‌ها و پیش‌پردازش‌های انجام شده در این پروژه، این اطمینان را حاصل می‌کنند که سیستم با سرعت و دقت بالا عمل کرده و مناسب برای محیط‌های بی‌درنگ باشد. منبع کد پیاده‌سازی شده برای این پروژه نیز در لینک **گیت‌هاب** قابل مشاهده است.

<sup>1</sup>Bounding Box

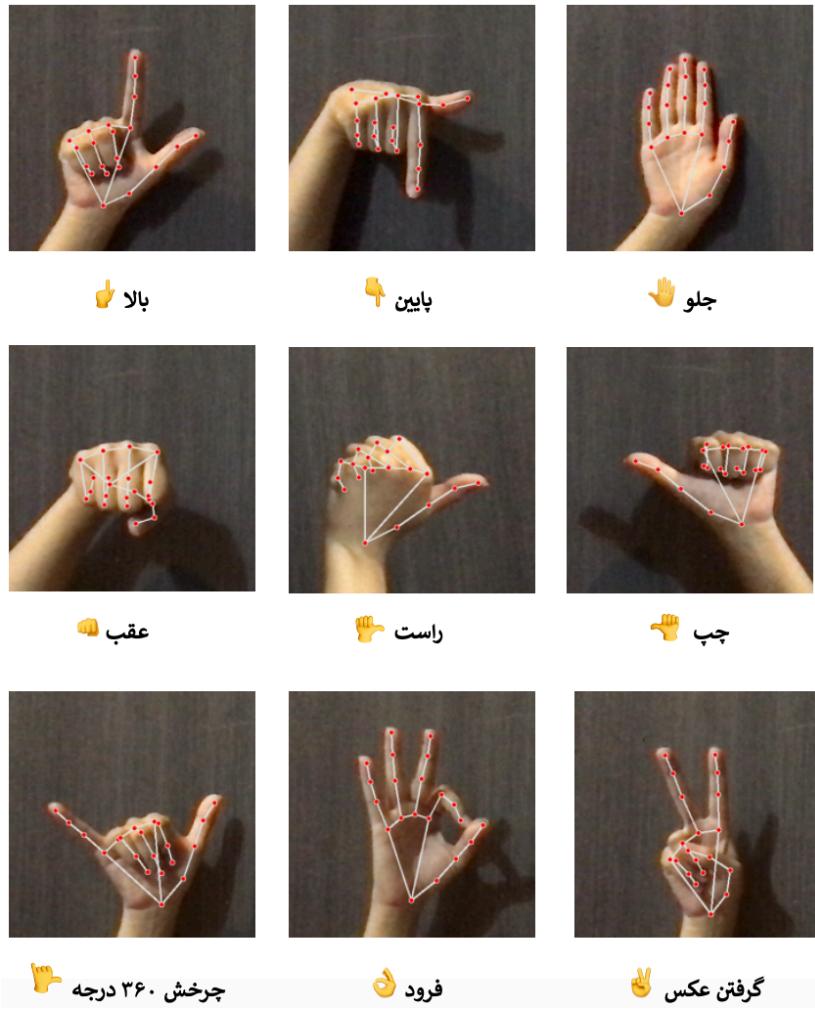


شکل ۳-۱ تشخیص علائم دست با کمک نقاط کلیدی دست [۲۴]

## ۲-۳ انتخاب علائم مناسب برای کنترل پهپاد

انتخاب علائم مناسب برای هر یک از حرکات پهپاد از اهمیت ویژه‌ای برخوردار است، زیرا علائم که از لحاظ مفهومی به عملکرد پهپاد شبیه هستند، راحت‌تر به خاطر سپرده می‌شوند و تجربه کاربری دلپذیرتری را ایجاد می‌کنند. برای این پروژه، نه علائم دست در نظر گرفته شده است تا حرکات پایه پهپاد با استفاده از آن‌ها انجام شود. این حرکات شامل: حرکت رو به جلو، حرکت رو به عقب، حرکت به سمت پایین، حرکت به سمت بالا، حرکت به راست، حرکت به چپ، فرود آمدن پهپاد، چرخش ۳۶۰ درجه به صورت ساعتگرد و گرفتن عکس است. این علائم به گونه‌ای انتخاب شده‌اند که با حرکات پهپاد همخوانی داشته باشند و کاربران بتوانند به راحتی آن‌ها را به خاطر بسپارند و از آن‌ها برای کنترل پهپاد استفاده کنند. نمونه‌های علائم دست متناسب با هر حرکت پهپاد، به طور دقیق در عکس ۲-۳ ارائه شده‌اند تا کاربران بتوانند به راحتی از آن‌ها استفاده کنند.

<sup>2</sup>Data Set



شکل ۳-۲ نمونه‌ای از علائم انتخاب شده در مجموعه داده‌ها

### ۳-۳ مجموع داده ۲

برای جمع‌آوری مجموع داده مناسب پروژه، از آنجایی که علائم دست بر اساس عملکرد پهپاد تعیین شده‌اند تا استفاده از آن‌ها برای کاربر مورد پسند باشد، پیدا کردن مجموع داده آماده غیرممکن است. جمع‌آوری مجموع داده جامع و مناسب از اهمیت بالایی برخوردار است زیرا مدل باید به صورت بی‌درنگ کار کند و نمی‌توان از مدل‌های سنگین یادگیری عمیق استفاده کرد. بنابراین، برای افزایش دقت مدل باید از روش‌های دیگری نظیر پیش‌پردازش، پس‌پردازش و استفاده از مجموع داده مناسب بهره گرفت. رویکردهای متفاوتی برای این پروژه پیاده‌سازی شدند تا بهترین راهکار با استفاده از مدلی با حجم کم و در عین حال جامع یافت شود. در نتیجه، مجموع داده‌های گوناگونی جمع‌آوری شدند که هر یک از آن‌ها ویژگی خاصی از تصویر را به عنوان داده‌ی مورد نیاز جمع‌آوری می‌کردند. این رویکردها شامل استفاده از ورودی کل عکس به صورت پیکسل‌های رنگی، پیدا کردن دست و ذخیره موقعیت آن به

صورت پیکسل‌های  $256 \times 256$  و پیدا کردن نقاط کلیدی دست و ذخیره موقعیت آن‌ها در مجموع داده بود. در نهایت، گزینه سوم به عنوان بهترین راه ممکن برای ذخیره داده‌ها و اجرای پروژه انتخاب شد. برای این پروژه، به منظور امکان اضافه کردن علائم دست جدید توسط کاربر، کدی پیاده‌سازی شد که با باز شدن دوربین و زدن یک حرف یکسان برای هر کلاس، اطلاعات آن تصویر استخراج شده و در مجموع داده ذخیره شود. این روش نه تنها جمع‌آوری داده را راحت‌تر می‌کند، بلکه به کاربر اجازه می‌دهد با زدن یک دکمه، کلاس جدیدی را ایجاد کند.

### ۴-۳ ابزارها و نرم افزارهای مورد استفاده

برای پیاده‌سازی این پروژه از ابزارها، نرم‌افزارها و کتابخانه‌های گوناگونی استفاده شده است که در ادامه به توضیح دقیق آن‌ها می‌پردازیم. قابل ذکر است که از کتابخانه‌هایی از جمله سی‌اس‌وی<sup>۳</sup>، کپی<sup>۴</sup>، ایترتولز<sup>۵</sup> و اواس<sup>۶</sup> نیز در قسمت‌هایی از پروژه به کار برده شده است که به دلیل استفاده جزئی توضیح داده نشده‌اند.

### ۴-۱ کتابخانه تنسورفلو<sup>۷</sup>

تنسورفلو یک کتابخانه نرم‌افزاری رایگان و منبع باز برای یادگیری ماشین و هوش مصنوعی است. این کتابخانه توسط گوگل برین توسعه داده شده و می‌تواند در طیف وسیعی از وظایف یادگیری ماشین مورد استفاده قرار گیرد. همچنین تمرکز ویژه‌ای بر آموزش و استنتاج شبکه‌های عصبی عمیق دارد.

تنسورفلو انعطاف‌پذیری بالایی دارد که می‌تواند برای انواع مختلف مدل‌های یادگیری ماشین از جمله شبکه‌های عصبی پیچشی، شبکه‌های عصبی بازگشتی و شبکه‌های مولد متخصص<sup>۸</sup> مورد استفاده قرار گیرد. تنسورفلو قابلیت اجرای مدل‌ها بر روی پردازنده‌های چندگانه، پردازنده‌های گرافیکی<sup>۹</sup> و واحد پردازشی تنسو<sup>۱۰</sup> را دارد. همچنین به دلیل محبوبیت و پشتیبانی گسترده، منابع آموزشی و کتابخانه‌های جانبی

<sup>3</sup>Csv

<sup>4</sup>Copy

<sup>5</sup>Iertools

<sup>6</sup>Os

<sup>7</sup>TensorFlow

<sup>8</sup>Generative Adversarial Network

<sup>9</sup>Graphics Processing Unit

<sup>10</sup>Tensor Processing Unit

فرانی برای آن وجود دارد. به طور کلی، تنسورفلو یکی از ابزارهای قدرتمند و پرکاربرد در حوزه یادگیری ماشین و هوش مصنوعی است [۲۵].

### ۲-۴-۳ کتابخانه سایکیت لرن<sup>۱۱</sup>

سایکیت لرن که با نام‌های Scikit-learn و Sklearn نیز شناخته می‌شود یک کتابخانه یادگیری ماشین رایگان و منبع باز برای زبان برنامه‌نویسی پایتون است. این کتابخانه شامل الگوریتم‌های مختلفی برای طبقه‌بندی، رگرسیون و خوشه‌بندی مانند ماشین‌های بردار پشتیبان، جنگل‌های تصادفی<sup>۱۲</sup>، تقویت گرادیان<sup>۱۳</sup>، میانگین-کی<sup>۱۴</sup> و دی‌بی اسکن<sup>۱۵</sup> می‌باشد. سایکیت لرن به طور ویژه برای تعامل با کتابخانه‌های نام‌پای<sup>۱۶</sup> و سای‌پای<sup>۱۷</sup> طراحی شده است و ابزارهای متعددی برای پیش‌پردازش داده‌ها، انتخاب و ارزیابی مدل‌ها و کاهش بعد فراهم می‌کند. این کتابخانه به کاربران کمک می‌کند تا به راحتی از آن در پروژه‌های یادگیری ماشین استفاده کنند. این کتابخانه به دلیل سادگی و کارایی خود در بین محققان و مهندسان داده بسیار محبوب است و امکانات وسیعی را برای توسعه و ارزیابی مدل‌های یادگیری ماشین فراهم می‌کند [۲۶].

### ۳-۴-۳ رابط برنامه‌نویسی کراس<sup>۱۸</sup>

کراس یک رابط برنامه‌نویسی<sup>۱۹</sup> یادگیری عمیق است که به زبان پایتون نوشته شده و می‌تواند بر روی تنسورفلو و پای‌تورچ<sup>۲۰</sup> اجرا شود. هدف اصلی کراس کاهش پیچیدگی‌ها و بار شناختی توسعه دهنده‌گان است، به طوری که آن‌ها بتوانند روی بخش‌های حیاتی و مهم پروژه‌های یادگیری ماشین تمرکز کنند. این رابط برنامه‌نویسی با رابط کاربری ساده و کاربرپسند، امکان توسعه سریع مدل‌های پیچیده را فراهم می‌کند. کراس عملکرد بالایی دارد و توسط سازمان‌های بزرگی نظیر ناسا، یوتیوب و Waymo برای تحلیل داده‌ها، بهبود الگوریتم‌های توصیه‌گر و توسعه سیستم‌های خودران مورد استفاده قرار می‌گیرد.

<sup>11</sup>Scikit-learn

<sup>12</sup>Random Forest

<sup>13</sup>Gradian activation

<sup>14</sup>K-means

<sup>15</sup>DBSCAN

<sup>16</sup>NumPy

<sup>17</sup>SciPy

<sup>18</sup>Keras

<sup>19</sup>Application Programming Interface

<sup>20</sup>PyTorch

این کتابخانه با مستندات جامع و پشتیبانی از جامعه کاربری بزرگ، به یکی از ابزارهای محبوب در حوزه یادگیری عمیق تبدیل شده است [۲۷].

#### ۴-۴-۳ کتابخانه‌های دیجی‌آی تلوپی<sup>۲۱</sup>

کتابخانه دیجی‌آی تلوپی یکی از کتابخانه‌های محبوب برای کنترل پهپاد دیجی‌آی تلو با استفاده از زبان برنامه‌نویسی پایتون است. این کتابخانه امکانات پیشرفته‌ای را به کاربران می‌دهد تا با استفاده از دستورات پیش‌فرض مختلف به راحتی حرکات پهپاد را کنترل کند. دیجی‌آی تلوپی شامل توابع بسیاری است که کاربران را قادر می‌سازد تا به سادگی حرکات پهپاد را کنترل کند.

علاوه بر این، این کتابخانه شامل برنامه‌هایی است که از کنترل حرکات پهپاد با استفاده از کلیدهای صفحه کلید و یا استفاده از دوربین پهپاد برای ایجاد پانوراما، پشتیبانی می‌کند. از آنجایی که کتابخانه دیجی‌آی تلوپی امکانات پیشرفته‌ای برای کنترل پهپاد دیجی‌آی تلو فراهم می‌کند، این کتابخانه می‌تواند یک ابزار قدرتمند برای توسعه و کنترل پروژه‌های مربوط به پهپاد باشد و به کاربران امکان کنترل دقیق و انعطاف‌پذیری را برای انجام وظایف مختلف فراهم کند.

همچنین این کتابخانه توانایی کنترل دسته ای از پهپادها به غیر از دیجی‌آی تلو را دارد و دارای تمام دستورالعمل‌هایی است که پهپاد دیجی‌آی تلو توانایی انجام آنها را دارد.

#### ۵-۴-۳ کتابخانه‌های مدیاپایپ<sup>۲۲</sup>

مدیاپایپ مجموعه‌ای از کتابخانه‌ها و ابزارهایی است که از تکنیک‌های هوش مصنوعی و یادگیری ماشین در برنامه‌های خود استفاده می‌کند. این کتابخانه برای برنامه‌نویسان یادگیری ماشین از جمله محققان، دانشجویان و توسعه‌دهندگان نرم‌افزار، که برنامه‌های کاربردی یادگیری ماشین را پیاده‌سازی می‌کنند، نمونه‌های اولیه و از پیش آموزش دیده‌ای را طراحی می‌کند تا پروژه‌ها را تا حد امکان ساده کند.

برنامه‌هایی که داده‌های حسی مانند ویدیو و صدا را با نرخ فریم بالا پردازش می‌کنند و به‌طور خاص برای بهبود تجربه کاربر طراحی شده‌اند چالش‌های مختلفی دارند. برای مثال مراحل پردازش یا مدل‌های استنتاجی ممکن است پیچیده باشند، زیرا گاهی اتصال بین مراحل بسیار زیاد است. همچنین، توسعه برنامه برای پلتفرم‌های مختلف نیز می‌تواند زمان بر باشد [۲۸].

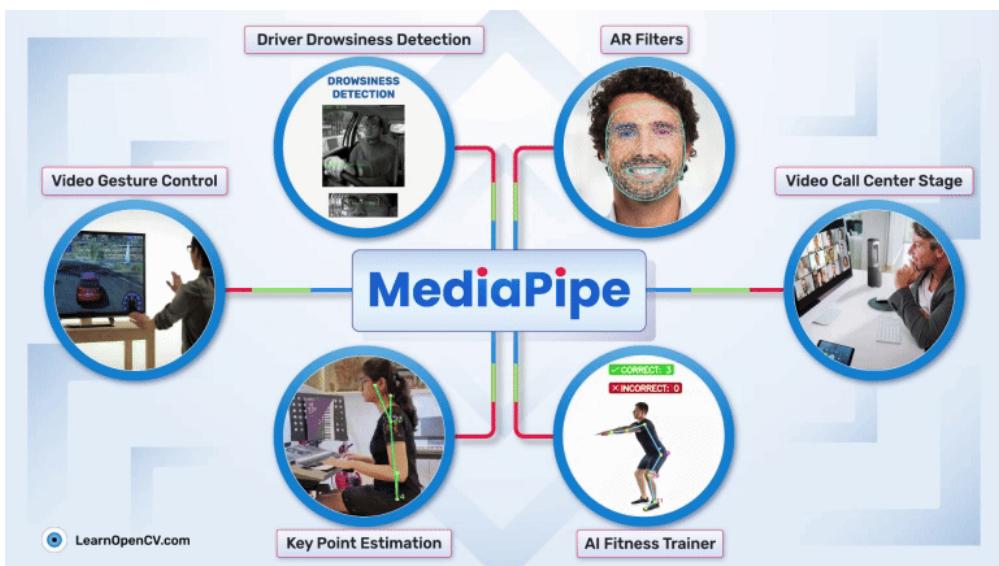
مدیاپایپ این چالش‌ها را با انتزاع و اتصال مدل‌های مختلف به یک چارچوب مناسب حل

<sup>21</sup>Djitelopy

<sup>22</sup>MediaPipe

می‌کند. با استفاده از مدیاپایپ، می‌توان یک لوله‌ی پردازش را به صورت گراف از اجزای مختلف، از جمله مدل‌های استنتاجی و عملکردهای پردازش رسانه‌ای، ساخت. این کتابخانه همچنین قابل سفارشی‌سازی است و می‌تواند بر روی پلتفرم‌های مختلف توسعه یابد.

در مجموعه مدیاپایپ نیز از کتابخانه‌های مختلفی برای پیاده‌سازی برنامه‌ها استفاده می‌شود. از جمله آن‌ها می‌توان به تنسورفلو، پای‌تورچ، اپن‌سی‌پی<sup>۲۳</sup>، CNTK و MXNet اشاره کرد [۲۹].



شکل ۳-۳ برخی کاربردهای کتابخانه مدیاپایپ [۲۹]

#### ۶-۴-۳ کتابخانه نامپای

نامپای کتابخانه‌ای برای محاسبات علمی در پایتون است که آرایه‌های چندبعدی و توابعی برای عملیات سریع بر روی این آرایه‌ها ارائه می‌دهد. در هسته‌ی نامپای، شیء Ndarray وجود دارد که آرایه‌های n-بعدی تشکیل شده از داده‌های همگن را در بر می‌گیرد و بسیاری از عملیات‌های ریاضی بر روی آن‌ها انجام می‌شود. این کتابخانه امکان انجام عملیات ریاضی پیشرفته و سایر عملیات‌ها روی تعداد زیادی داده را با کارایی بالا فراهم می‌کند، که نسبت به استفاده از توابع از پیش تعریف شده زبان برنامه‌نویسی پایتون کارآمدی و حجم کد کمتری دارد [۳۰].

<sup>23</sup>OpenCv

<sup>24</sup>Matplotlib

<sup>25</sup>Cross-Platform

### ۷-۴-۳ کتابخانه متپلات لیپ ۲۴

متپلات لیپ یک کتابخانه چند پلتفرمی<sup>۲۵</sup>، برای تجسم داده‌ها و نمودارهای گرافیکی از جمله هیستوگرام، نمودارهای پراکنده و نمودار میله‌ای برای زبان برنامه‌نویسی پایتون است. توسعه دهنده‌گان همچنین می‌توانند از رابطه‌های برنامه‌نویسی متپلات لیپ برای جاسازی نمودارها در برنامه‌های دارای رابط کاربری گرافیکی نیز استفاده کنند.

یک آغازگر<sup>۲۶</sup> متپلات لیپ درون زبان برنامه‌نویسی پایتون به گونه‌ای ساختار یافته است که چند خط کد تنها چیزی است که در بیشتر موارد برای تولید نمودار داده بصری مورد نیاز است. متپلات لیپ دو رابط برنامه‌نویسی را پوشش می‌دهد:

رابط برنامه‌نویسی پایپلات<sup>۲۷</sup>، که سلسله مراتبی از اشیاء درون کد پایتون است که در بالای آن قرار دارد. و رابط برنامه‌نویسی اشیاء گرا<sup>۲۸</sup> که می‌تواند با انعطاف پذیری بیشتری نسبت به Pyplot سرهم شوند. این رابط برنامه‌نویسی دسترسی مستقیم به لایه‌های پشتی متپلات لیپ را فراهم می‌کند [۳۱].<sup>۲۹</sup>

### ۸-۴-۳ کتابخانه اپن سی‌وی

اپن سی‌وی یک کتابخانه متن باز برای بینایی ماشین و یادگیری ماشین است که برای فراهم کردن زیرساخت مشترک برای برنامه‌های بینایی ماشین و تسریع استفاده از ادراک ماشین در محصولات تجاری طراحی شده است. این کتابخانه شامل بیش از ۲۵۰۰ الگوریتم بهینه‌سازی شده است که مجموعه جامعی از الگوریتم‌های کلاسیک و جدید بینایی ماشین و یادگیری ماشین را فراهم می‌کند. اپن سی‌وی به طور گسترده‌ای در شرکت‌ها، گروه‌های تحقیقاتی و نهادهای دولتی برای انجام پروژه‌های بینایی ماشین و یادگیری ماشین استفاده می‌شود. این کتابخانه رابطه‌ای برنامه‌نویسی متعددی از جمله سی پلاس پلاس، پایتون، جاوا و متلب را دارد که امکان انجام پروژه‌های بینایی ماشین با استفاده از زبان‌های برنامه‌نویسی مختلف را فراهم می‌کند [۳۲].

<sup>26</sup>Script

<sup>27</sup>Pyplot

<sup>28</sup>Object Oriented

<sup>29</sup>Matplotlib Backend

### ۹-۴-۳ پهپاد دی جی آی تلو

پهپاد دی جی آی تلو یک پهپاد کوادکوپتر کوچک و قابل برنامه ریزی است که برای مصارف آموزشی و تست پروتوتایپ توسط شرکت دی جی آی طراحی شده است. این پهپاد دارای ویژگی های ویژه ای مانند حرکات پایه ای کوادکوپتری و همچنین تکنولوژی کنترل پرواز دی جی آی و یک پردازنده بسیار قوی از شرکت اینتل است. دوربین ۵ مگاپیکسلی این پهپاد امکان ضبط ویدیو با کیفیت مناسب را فراهم می کند. همچنین، پهپاد دارای یک سیستم موقعیت یابی بصری<sup>۳۰</sup> است که شامل یک دوربین و یک مازول مادون قرمز ۳ بعدی است و قادر است در فواصل سه تا سی متر ارتفاع را کار کند.

هسته پهپاد به عنوان مرکز پردازشی و کنترلی آن عمل می کند و از یک پردازنده اینتل قدرتمند پشتیبانی می کند. پهپاد دی جی آی تلو برای اجرای پروژه های هوش مصنوعی مانند تشخیص اشیاء روی پهپاد، از زبان برنامه نویسی پایتون و بسته توسعه نرم افزار مربوطه پشتیبانی می کند. این بسته توسعه نرم افزار به کاربران این امکان را می دهد که نمونه های اولیه پروژه های خود را توسعه دهند و آن ها را بر روی پهپاد اجرا کنند.

این پهپاد دارای یک باتری با ویژگی های خاصی مانند زمان پرواز زیاد و زمان شارژ کم است که می تواند از نظر عملکرد و ماندگاری باتری نسبت به پهپادهای دیگر مزیت داشته باشد. همچنین، پروژه های هوش مصنوعی که روی پهپاد دی جی آی تلو پیاده سازی می شوند، می توانند شامل تشخیص اشیاء، پیش بینی حرکت ها و یا حتی خود کار سازی فرآیندهای پروازی باشند. از جمله مدل های هوش مصنوعی که می تواند روی پهپاد دی جی آی تلو پیاده سازی شود، می توان به یولو سه<sup>۳۱</sup> اشاره کرد که برای تشخیص اشیاء با دقیق بالا استفاده می شود [۳۳].

Weight	87 g
Dimensions	98×92.5×41 mm
Propeller	3 inches
	Telemetric sensor
	Barometer
Integrated Functions	LED
	Vision System
	Wi-Fi 2.4 GHz 802.11n
	Real-time streaming 720p
	USB battery charging port
Port	from 0° to 40°
Operating temperature range	from 2.4 to 2.4835 GHz
Operating frequency range	20 dBm (FCC)
	19 dBm (CE)
Transmitter (EIRP)	19 dBm (SRRC)

شکل ۴-۳ اطلاعات پهپاد دی جی آی تلو [۳۳]

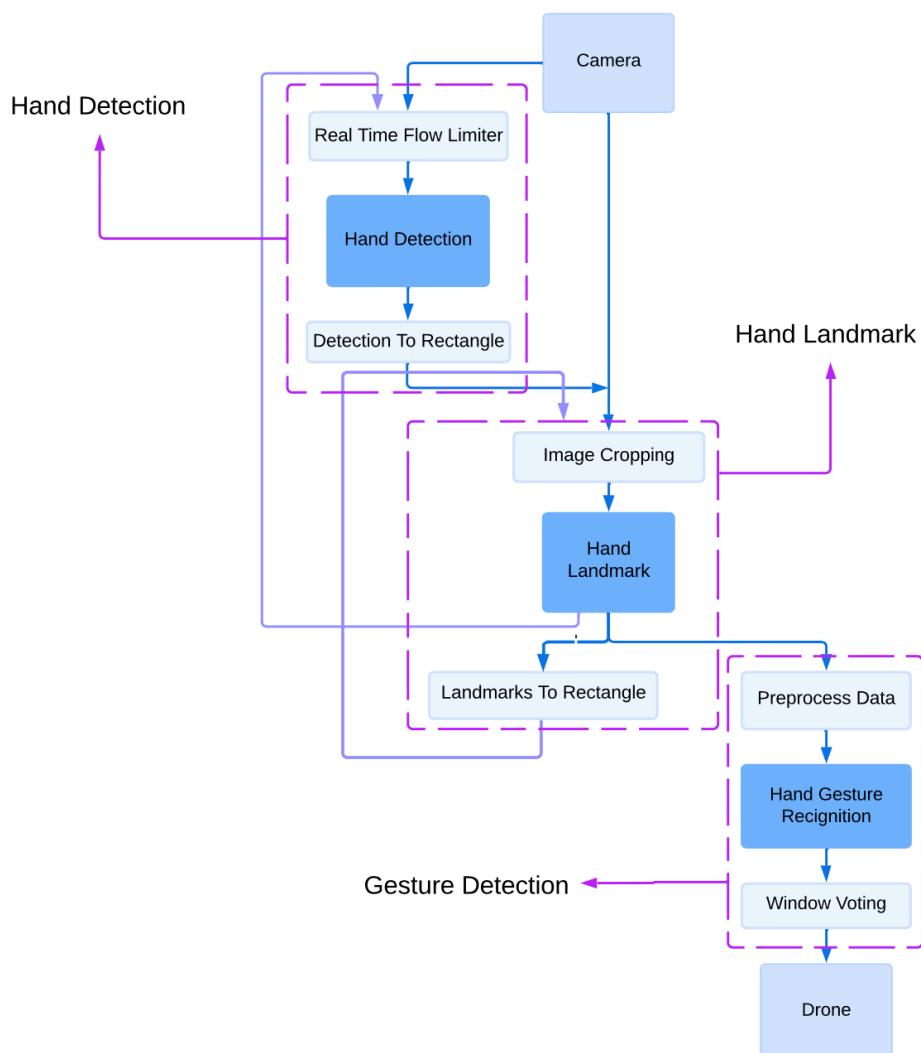
<sup>30</sup>Vision Positioning System

<sup>31</sup>YOLOv3

### ۵-۳ روند اجرای پروژه

در این پروژه از ۳ مدل اصلی به صورت متوالی برای پیش‌بینی تعیین علائم دست استفاده شده است. این سه مدل شامل تشخیص مرزهای دست، تشخیص نقاط کلیدی دست و در نهایت تعیین علائم دست است. پس از استفاده از این مدل‌های بینایی ماشین، دستور پیش‌بینی شده برای اجرا به پهپاد دی‌جی‌آی تلو فرستاده می‌شود.

برای بهینه کردن روند این پروژه مأذول‌های گوناگونی از جمله محدود کننده جریان بی‌درنگ، تشخیص با استفاده از مستطیل محدود کننده، برش تصویر، ورود نقاط عطف به مستطیل محدود کننده و ارائه کننده حاشیه نویسی نیز پیاده‌سازی شده‌اند تا روند پیش‌بینی علائم دست را با سرعت و دقیق‌تری اجرا کنند.



شکل ۳-۳ روند نمای پروژه

## ۳-۶-۳ تشخیص دست

### ۱-۶-۳ مدل تشخیص کف دست

ماژول تشخیص کف دست، یکی از سه ماژول اصلی است که با دقیق متوسط ۹۵.۷ درصد عمل می‌کند. این دقیق بالا با استفاده از استراتژی‌های مختلف به دست آمده است. این ماژول به جای تشخیص دست، از یک مدل تشخیص کف دست استفاده می‌کند زیرا تشخیص محدوده‌های اجسام سفت و سخت مانند کف دست و مشت بسیار ساده‌تر از تشخیص دست‌ها با انگشتان مفصلی است. در این ماژول از الگوریتم سرکوب غیر حداکثری برای حذف تشخیص‌های تکراری و انتخاب مرتبط‌ترین اشیاء شناسایی شده استفاده می‌شود که به کاهش نتیجه‌های مثبت کاذب و پیچیدگی محاسباتی کمک می‌کند و وظیفه اصلی آن تشخیص دست در تصویر و محاسبه مکان دقیق دست است.

این ماژول قادر است به صورت دقیق و کارآمد موقعیت دست‌ها را شناسایی کند و نواحی مربوطه را برای پردازش‌های بعدی فراهم کند. ورودی این ماژول شامل تصویر یا فریم ویدیو است، در حالی که خروجی آن شامل مستطیل‌های محدوده دست‌ها، نمرات اطمینان و در صورت فعال بودن، دست غالب<sup>۳۲</sup> است. معماری این ماژول شامل مراحل پیش پردازش<sup>۳۴</sup>، مدل تشخیص دست و پس پردازش<sup>۳۵</sup> است که به ترتیب شامل نرمال‌سازی و تغییر اندازه تصویر، شبکه عصبی تشخیص دست و فیلترینگ و محاسبه نواحی مستطیلی دست‌ها می‌باشد.

وظیفه اصلی ماژول شناسایی و محصور کردن دست‌ها در تصاویر و ویدیوها است که در کاربردهای مختلفی از جمله تشخیص حرکات دست، رابطه‌ای کاربری بدون لمس، تحلیل رفتار و علائم و کمک به افراد کم‌توان از اهمیت برخوردار است. این ماژول به توسعه‌دهندگان این امکان را می‌دهد که به راحتی و با دقیق بالا دست‌ها را در تصاویر و ویدیوها شناسایی کرده و از این اطلاعات برای پردازش‌های بعدی استفاده کنند.

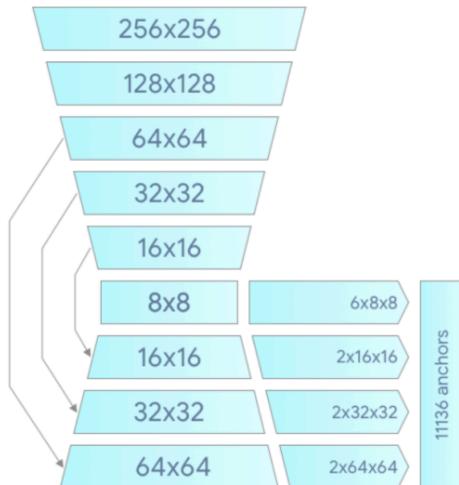
<sup>32</sup>Hand Detection

<sup>33</sup>Handedness

<sup>34</sup>Preprocessing

<sup>35</sup>Postprocessing

<sup>36</sup>Real Time Flow Limiter



شکل ۳-۶ معماری مدل آشکارساز کف دست [۱۲]

### ۲-۶-۳ محدود کننده جریان بی‌درنگ<sup>۳۶</sup>

ماژول محدود کننده جریان بی‌درنگ وظیفه‌ی محدود کردن پردازش برای رسیدن برنامه به سرعت بی‌درنگ را بر عهده دارد. این ماژول امکان کنترل پردازش داده‌ها را به صورت کارآمد و بی‌درنگ بدون ایجاد تأخیر یا باز اضافی بر سیستم فراهم می‌کند. ورودی‌های این ماژول شامل جریان‌های بسته<sup>۳۷</sup> و مهر زمان‌ها<sup>۳۸</sup> و خروجی‌های آن شامل جریان‌های بسته با نرخ محدود<sup>۳۹</sup> و اطلاعات مربوط به بسته‌های حذف شده‌ی ناشی از محدودیت نرخ پردازش است. معماری این ماژول شامل مراحل بافر ورودی<sup>۴۰</sup>، تجزیه و تحلیل مهر زمانی<sup>۴۱</sup>، الگوریتم محدود کردن نرخ<sup>۴۲</sup> و بافر خروجی<sup>۴۳</sup> است که به منظور محدود کردن نرخ جریان داده‌ها و انجام پردازش به صورت بی‌درنگ طراحی شده است. این ماژول کمک می‌کند تا باز پردازش و تأخیر کاهش یابد، کیفیت خدمات حفظ شود و منابع محاسباتی بهینه‌سازی شوند [۱۲].

<sup>۳۷</sup>Packet Streams

<sup>۳۸</sup>Time Stamp

<sup>۳۹</sup>Rate-Limited Packet Streams

<sup>۴۰</sup>Input Buffering

<sup>۴۱</sup>Timestamp Analysis

<sup>۴۲</sup>Rate Limiting Algorithm

<sup>۴۳</sup>Output Buffering

<sup>۴۴</sup>Detection To Rectangle

<sup>۴۵</sup>Detection Proto

<sup>۴۶</sup>Image Frame

### ۳-۶-۳ تشخیص به مستطیل محدودکننده<sup>۴۴</sup>

این مژول وظیفه تبدیل نتایج تشخیص اشیاء به مستطیل‌های محدودکننده دست را دارد. این مژول ورودی‌هایی مانند تشخیص پروتوی<sup>۴۵</sup> و فریم تصویر<sup>۴۶</sup> را دریافت کرده و مستطیل‌های محدودکننده را به صورت نرمال‌سازی شده و یا مطلق برای نواحی تشخیص داده شده تولید می‌کند. معماری این مژول شامل مراحل پیش‌پردازش، تبدیل تشخیص<sup>۴۷</sup> و پس‌پردازش است. وظیفه اصلی آن تبدیل نتایج تشخیص به مستطیل‌های محدودکننده است که در کاربردهای مختلف مانند تشخیص و ردیابی چهره، تشخیص دست، امنیتی و واقعیت افزوده کاربرد دارد [۱۲].

### ۷-۳ تشخیص نقاط عطف دست<sup>۴۸</sup>

#### ۱-۷-۳ مدل تشخیص نقاط عطف دست

این مژول یکی دیگر از مژول‌های مدل اصلی است که یک ابزار قدرتمند برای تشخیص و ردیابی نقاط کلیدی دست است. ورودی آن یک تصویر است که شامل دست یا دست‌هایی است که می‌خواهیم نقاط کلیدی آن‌ها را تشخیص دهیم. این تصویر باید از پیش‌پردازش شده و دارای جعبه‌ی مرزی‌ای باشد که دست در آن قرار دارد که توسط مژول تشخیص دست مشخص شده است.

خروجی این مژول شامل موقعیت سه‌بعدی بیست و یک نقطه کلیدی دست است که شامل مفاصل انگشتان و نوک انگشتان می‌باشد. این نقاط کلیدی به صورت مجموعه‌ای از مختصات طول، عرض و ارتفاع ارائه می‌شود که طول و عرض نشان دهنده موقعیت هر نقطه را در فضای دو بعدی و ارتفاع نشان دهنده عمق نقطه نسبت به تصویر ورودی است.

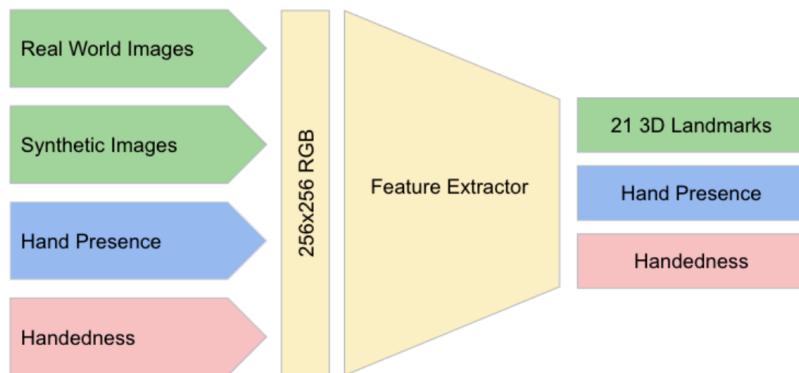
معماری این مژول شامل شبکه‌های عصبی عمیق است که از آن‌ها برای تشخیص و ردیابی نقاط کلیدی دست استفاده می‌کند. این معماری شامل مراحل پیش‌پردازش برای تغییر اندازه و نرمال‌سازی تصویر، تشخیص دست برای شناسایی ناحیه‌های دست، مدل نقاط عطف برای تشخیص دقیق نقاط کلیدی دست و پس‌پردازش برای تبدیل مختصات نقاط کلیدی به فرمت خروجی نهایی و اعمال اصلاحات لازم است. وظیفه اصلی این مژول تشخیص و ردیابی دقیق نقاط کلیدی دست‌ها در تصاویر و ویدیوها است. این قابلیت می‌تواند در برنامه‌های مختلفی مانند واقعیت افزوده<sup>۴۹</sup>، رابطه‌ای کاربری بدون لمس، تحلیل

<sup>47</sup>Detection Conversion

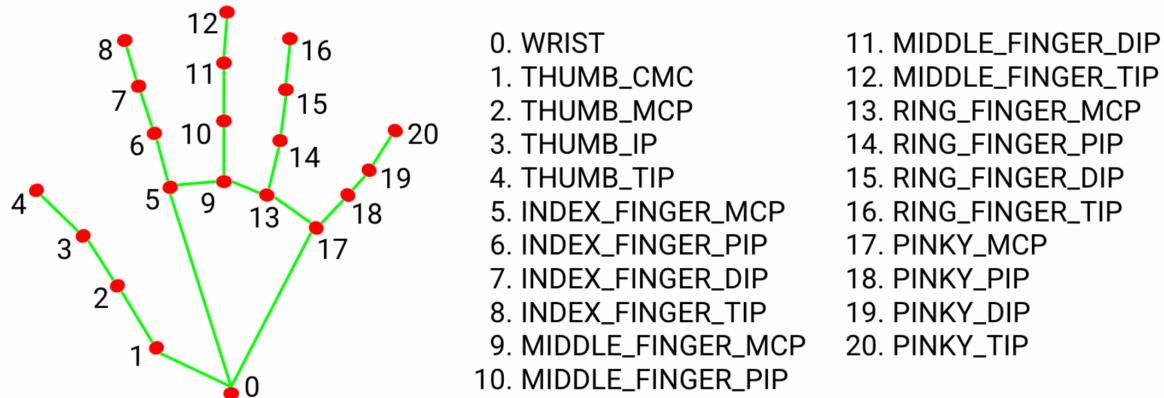
<sup>48</sup>Hand Landmark

<sup>49</sup>Augmented Reality

حرکات، تشخیص حرکات دست در زبان اشاره و ... استفاده شود و توسعه دهنده‌گان را قادر می‌سازد از این قابلیت‌های پیشرفت‌های خود بهره ببرند [۱۲].



شکل ۷-۳ معماری مدل نقطه عطف دست. این مدل دارای سه خروجی است که یک استخراج کننده ویژگی را به اشتراک می‌گذارند. هر سر توسط مجموعه داده‌های مربوطه که با همان رنگ مشخص شده آند آموزش داده می‌شود [۱۲].



شکل ۸-۳ موقعیت ۲۱ نقطه کلیدی در ناحیه دست [۱۲]

### ۲-۷-۳ برش تصویر<sup>۵۰</sup>

ماژول برش تصویر برای برش و استخراج ناحیه‌های مورد نظر از تصاویر استفاده می‌شود. این ماژول ورودی‌هایی مانند تصویر اصلی و جعبه مرزی<sup>۵۱</sup> یا مختصات برش را می‌پذیرد و تصویر برش‌خورده حاوی ناحیه مورد نظر را تولید می‌کند. معماری این ماژول شامل مراحل پیش‌پردازش، عملیات برش<sup>۵۲</sup> و پس پردازش می‌باشد. وظیفه اصلی این ماژول برش دقیق ناحیه‌های مورد نظر از تصویر اصلی است که در

<sup>50</sup>Image Cropping

<sup>51</sup>Bounding Box

<sup>52</sup>Cropping Operation

کاربردهای مختلفی مانند پیش‌پردازش برای تشخیص چهره یا دست، تحلیل تصاویر پزشکی، واقعیت افروده و پردازش تصویر در کاربردهای امنیتی مورد استفاده قرار می‌گیرد. این مازول به توسعه‌دهنگان این امکان را می‌دهد که ناحیه‌های خاصی از تصاویر را به صورت دقیق و مؤثر استخراج کرده و برای پردازش‌های بعدی آماده کنند [۱۲].

### ۳-۷-۳ نقاط عطف یه مستطیل<sup>۵۳</sup>

این مازول برای تبدیل نقاط کلیدی به مستطیل‌های محدود‌کننده استفاده می‌شود. هدف آن محصور دقیق نقاط کلیدی دست است. با استفاده از این مستطیل‌ها، می‌توان موقعیت دقیق‌تر اشیاء در تصویر را نمایش داد و از آن‌ها به عنوان ورودی برای مازول‌های بعدی در گراف استفاده کرد. این مازول شامل مراحل پیش‌پردازش داده‌های ورودی، محاسبه مستطیل محدود‌کننده، نرمال‌سازی مختصات مستطیل‌ها و تولید خروجی نهایی می‌باشد. وظیفه اصلی این مازول تبدیل نقاط کلیدی به مستطیل‌های محدود‌کننده دقیق است که در بسیاری از کاربردها اهمیت دارد، از جمله آن‌ها ردیابی اشیاء، تحلیل حرکات، پیش‌پردازش برای مدل‌های دیگر و افزایش دقت در پردازش تصویر است [۱۲]. تفاوت این مازول با مازول اول این است که در مازول اول ما ورودی خام از تصاویر و یا فریم‌های ویدیو را به عنوان ورودی داده و محدوده‌ی دست را تعیین می‌کنیم، اما در این مازول نقاط کلیدی استخراج شده به مازول داده می‌شوند تا مجدد یک مستطیل محدود‌کننده برای نقاط عطف استخراج شده تولید کند.

### ۳-۸ پیش‌بینی علائم دست

برای پیش‌بینی علائم دست از مدل‌های گوناگونی استفاده شده است که معماری و مشخصات هر یک در ادامه به تفکیک توضیح داده شده‌اند. این معماری‌ها شامل شبکه چندلایه‌ای پرسپترونی، شبکه عصبی پیچشی، شبکه عصبی بازگشتی و شبکه‌های حافظه کوتاه مدت بلند مدت است. دلیل استفاده از این مدل‌ها حجم کم و در عین حال دقت بالای آن‌ها است تا بتوان آن‌ها را با یکدیگر قیاس و در نهایت بهترین مدل را برگزید. ورودی این مدل، نقاط کلیدی دست و خروجی آن پس از پردازش، شامل ۱۰ کلاس است. شایان ذکر است که کاربر توانایی اضافه کردن کلاس و در نتیجه علائم جدید را نیز در صورتی که پهپاد قابلیت انجام آن را داشته باشد دارد.

<sup>۵۳</sup>Landmarks To Rectangle



شکل ۹-۳ نمونه پیش‌بینی علامت دست

### ۱-۸-۳ مدل‌های ماژول کلاس‌بندی برای تعیین علائم دست

#### شبکه عصبی چندلایه‌ای پرسپترونی

شبکه عصبی چندلایه‌ای پرسپترونی<sup>۵۴</sup> نوعی شبکه عصبی هوش مصنوعی است که از چندین لایه نورون تشکیل شده است. نورون‌های درون لایه‌ها معمولاً از توابع فعال‌سازی غیرخطی<sup>۵۵</sup> استفاده می‌کنند که به شبکه اجازه می‌دهد الگوهای پیچیده در داده‌ها را بیاموزد. شبکه‌های عصبی چندلایه‌ای پرسپترونی در یادگیری ماشین اهمیت بالایی دارند زیرا می‌توانند روابط غیرخطی در داده‌ها را یاد بگیرند و آن‌ها را به مدل‌های قدرتمندی برای استفاده در کارهایی مانند طبقه‌بندی، رگرسیون و تشخیص الگو تبدیل کنند.

شبکه عصبی چندلایه‌ای پرسپترونی نوعی شبکه عصبی پیش‌خور<sup>۵۶</sup> است که از نورون‌های کاملاً متصل با یک تابع فعال‌سازی از نوع غیر خطی تشکیل شده است و برای تشخیص داده‌هایی که به صورت خطی قابل تفکیک نیستند استفاده می‌شود.

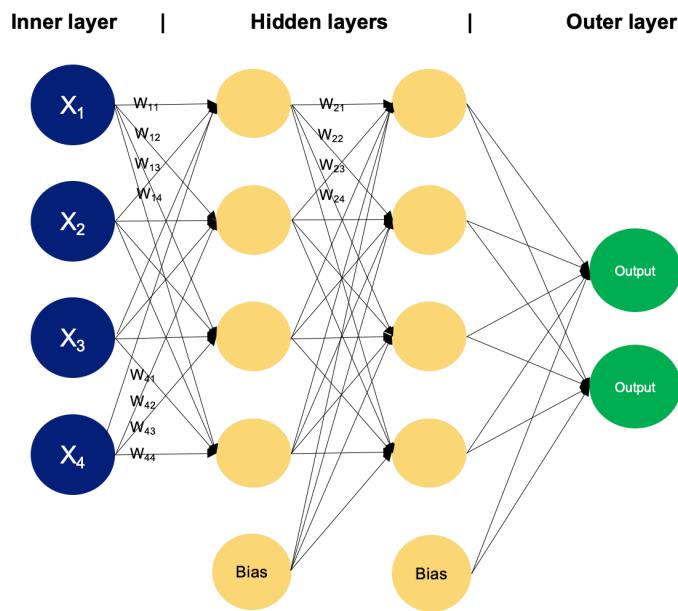
شبکه‌های عصبی چندلایه‌ای پرسپترونی به طور گسترده در زمینه‌های مختلف از جمله تشخیص تصویر، پردازش زبان طبیعی و تشخیص گفتار و ... استفاده می‌شوند. انعطاف پذیری آن‌ها در معماری و توانایی تقریب هر عملکرد تحت شرایط خاص آن‌ها را به یک بلوک اساسی در یادگیری عمیق و تحقیقات شبکه عصبی تبدیل می‌کند.

<sup>54</sup>Multilayer Perceptron Neural Network

<sup>55</sup>Nonlinear Activation Functions

<sup>56</sup>Feedforward

<sup>57</sup>Convolutional Neural Network



شکل ۳-۱۰ نمونه ای از شبکه‌های عصبی چند لایه‌ای دارای دو لایه پنهان [۳۴]

#### شبکه عصبی پیچشی

شبکه‌های عصبی پیچشی<sup>۵۷</sup> یک الگوریتم یادگیری عمیق است که از جمله کاربردهای آن در زمینه‌ی تشخیص اشیا می‌توان به مواردی مانند طبقه‌بندی، تشخیص و تقسیم بندی تصویر اشاره کرد. بسیاری از برنامه‌های کاربردی مانند اتومبیل‌های خودران، دوربین‌های نظارتی و موارد دیگر از شبکه‌های عصبی پیچشی استفاده می‌کنند.

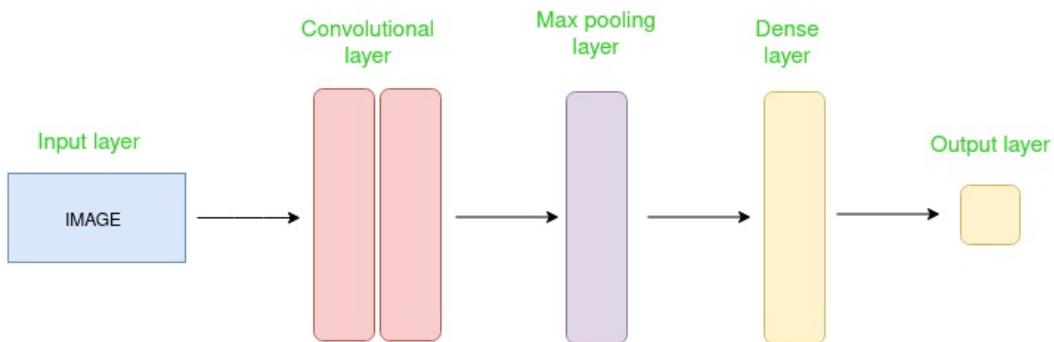
برخلاف مدل‌های سنتی یادگیری ماشین مانند پشتیبانی بردار و درخت تصمیم<sup>۵۸</sup> که نیاز به استخراج دستی ویژگی‌ها دارند، شبکه‌های عصبی پیچشی می‌توانند استخراج خودکار ویژگی‌ها را در مقیاس‌های بزرگ انجام دهند. این ویژگی یکی از دلایل اصلی کارآمد بودن این شبکه‌ها است. این شبکه‌ها می‌توانند الگوهای مختلف را از درون داده‌ها تشخیص دهند و ویژگی‌های داده را بدون توجه به موقعیت آن‌ها، اعم از چرخش، تبدیل‌های هندسی و یا جابجایی تصاویر استخراج کنند.

معماری شبکه‌های عصبی پیچشی سعی می‌کنند تا ساختار نورون‌ها را در سیستم بینایی انسان متشكل از چندین لایه تقلید کند، جایی که هر یک مسئول تشخیص یک ویژگی خاص در داده‌ها است. همانطور که در شکل ۳-۱۱ نیز نشان داده شده است، شبکه عصبی پیچشی از چندین لایه مانند لایه ورودی، لایه پیچشی، لایه ادغام<sup>۵۹</sup> و لایه‌های کاملاً متصل تشکیل شده است.

<sup>58</sup>Decision Tree

<sup>59</sup>Pooling

لایه پیچشی فیلترهایی را روی تصویر ورودی اعمال می‌کند تا ویژگی‌ها را استخراج کند، لایه ادغام ابعاد تصویر را کاهش می‌دهد تا محاسبات را سریع‌تر و کمتر کند و لایه کاملاً متصل پیش‌بینی نهایی را انجام می‌دهد. بدین صورت که شبکه فیلترهای بهینه را از طریق پس انتشار<sup>۶۰</sup> و نزول گرادیان<sup>۶۱</sup> می‌آموزد.



شکل ۳-۱۱ نمونه معماری شبکه عصبی پیچشی [۳۵]

### شبکه عصبی بازگشتی

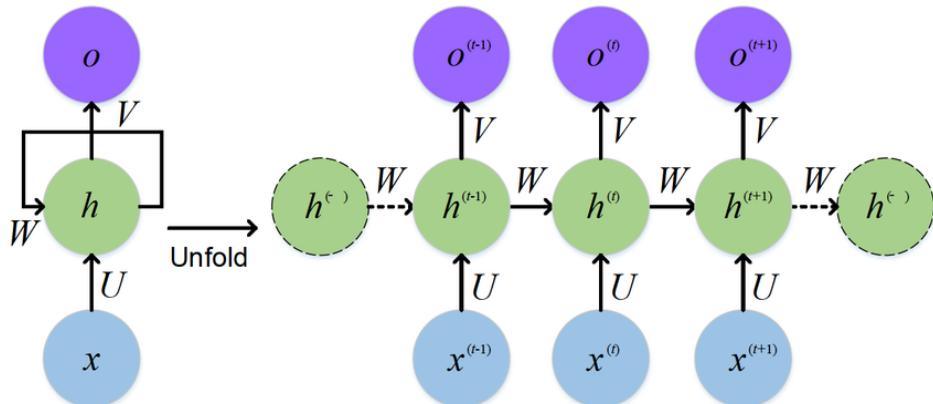
شبکه عصبی بازگشتی<sup>۶۲</sup> نوعی شبکه عصبی است که در آن خروجی مرحله قبل به عنوان ورودی به مرحله فعلی داده می‌شود. در شبکه‌های عصبی سنتی، تمامی ورودی‌ها و خروجی‌ها مستقل از یکدیگر هستند. برای مثال در مواردی که پیش‌بینی مدل متکی به موارد قبلی است و نیاز است آن‌ها را نیز مدنظر قرار دهد این شبکه بسیار کارآمد است. شبکه عصبی بازگشتی با کمک یک لایه پنهان این امکان را فراهم می‌کند. اصلی‌ترین و مهم‌ترین ویژگی شبکه عصبی بازگشتی حالت پنهان آن است که برخی از اطلاعات یک دنباله را به خاطر می‌سپارد. این حالت به عنوان حالت حافظه نیز شناخته می‌شود زیرا ورودی قبلی شبکه را به خاطر می‌آورد. و از پارامترهای یکسانی برای هر ورودی استفاده می‌کند چرا که وظیفه یکسانی را روی تمام ورودی‌ها یا لایه‌های پنهان برای تولید خروجی انجام می‌دهد. این برخلاف سایر شبکه‌های عصبی، پیچیدگی پارامترها را کاهش می‌دهد.

<sup>60</sup>Back Propagation

<sup>61</sup>Gradian Descent

<sup>62</sup>Recurrent Neural Network

<sup>63</sup>Long Short-Term Memory

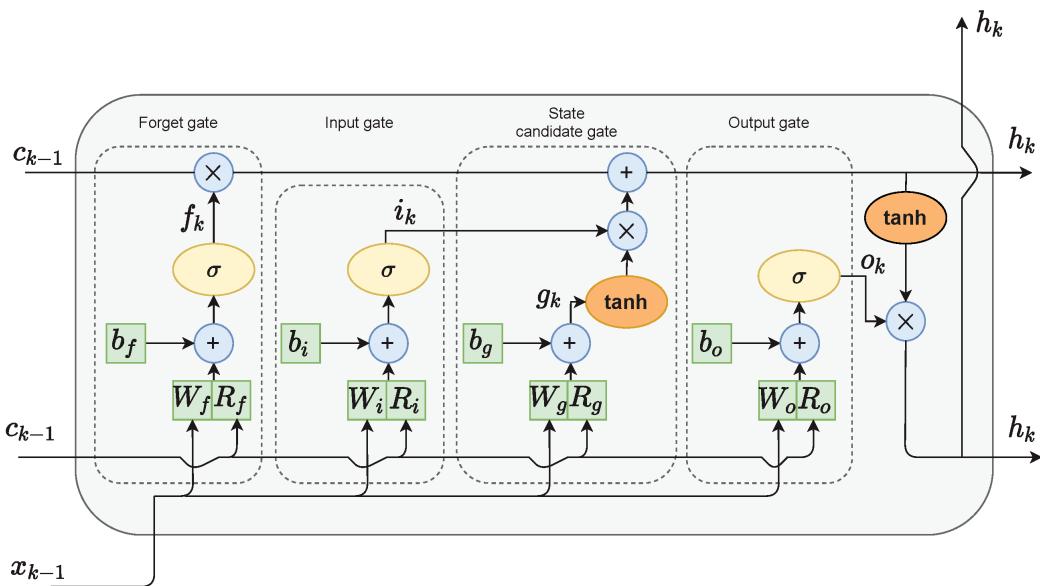


شکل ۳-۱۲ نمونه معماری شبکه عصبی بازگشتی [۳۶]

### شبکه حافظه طولانی کوتاه مدت

شبکه عصبی بازگشتی یک حالت پنهان دارد که در طول زمان منتقل می‌شود، این مساله می‌تواند یادگیری وابستگی‌های طولانی مدت را برای شبکه دشوار کند. شبکه‌های حافظه طولانی کوتاه مدت <sup>۶۳</sup> این مشکل را با معرفی یک سلول حافظه برطرف می‌کند، این سلول محفظه‌ای است که می‌تواند اطلاعات را برای مدت طولانی نگهداری کند. شبکه‌های حافظه طولانی کوتاه مدت قادر به یادگیری وابستگی‌های طولانی مدت در داده‌های متوالی هستند، که آن‌ها را برای کارهایی مانند ترجمه زبان، تشخیص گفتار و پیش‌بینی سری‌های زمانی مناسب می‌سازد. شبکه‌های حافظه طولانی کوتاه مدت همچنین می‌توانند در ترکیب با دیگر معماری‌های شبکه عصبی، مانند شبکه‌های عصبی پیچشی برای تجزیه و تحلیل تصویر و ویدئو استفاده شوند.

سلول حافظه توسط سه گیت کنترل می‌شود: گیت ورودی، دروازه فراموشی و گیت خروجی. این گیتها تصمیم می‌گیرند که چه اطلاعاتی را به سلول حافظه اضافه، حذف و از آن خروجی بگیرند. گیت ورودی کنترل می‌کند که چه اطلاعاتی به سلول حافظه اضافه می‌شود. دروازه فراموشی کنترل می‌کند که چه اطلاعاتی از سلول حافظه حذف می‌شود. و گیت خروجی کنترل می‌کند که چه اطلاعاتی از سلول حافظه خارج می‌شود. این گیت به شبکه‌های حافظه طولانی کوتاه مدت اجازه می‌دهد تا به‌طور انتخابی اطلاعات در جریان درون شبکه را حفظ و یا کنار بگذارند. این ویژگی امکان آموزش وابستگی‌های طولانی مدت را برای شبکه فراهم می‌کند.



شکل ۱۳-۳ نمونه معماری شبکه حافظه طولانی کوتاه مدت [۳۷]

### ۲-۸-۳ پیشپردازش داده‌ها

برای ورود نقاط عطف دست به مدل تعیین علائم، به ۲۱ مختصات طول و عرض نیاز داریم. خروجی مدل تعیین مختصات نقاط عطف دست برابر مختصات مطلق پیکسل‌ها نسبت به گوشه سمت چپ پایین تصویر است. این نقاط با توجه به اندازه عکس می‌توانند گسترده باشند برای مثال در یک عکس با اندازه  $2048 \times 2048$  این اعداد از بین ۰ تا  $2048$  متغیر اند. اگر این مختصات را به صورت مستقیم به مدل تعیین علائم دست بدھیم دقیق مدل برابر  $87$  درصد خواهد بود که به میزان کافی مورد قبول نیست. برای بهبود آن باید پیشپردازش‌هایی بر روی داده ورودی انجام شود.

از جمله این پیشپردازش‌ها می‌توان به نسبی کردن و نرمال‌سازی داده‌ها اشاره کرد. برای این کار ابتدا باید یک مرجع واحد در نظر گرفت تا نقاط، نسبت به آن مشخص شوند. در این پروژه ما مرجع را نقطه مشخص شده روی مج در نظر می‌گیریم. مختصات نقطه مرجع را برابر  $(0, 0)$  قرار می‌دهیم. سپس نسبت به آن و با توجه به رابطه (۱-۳) مختصات نقاط دیگر را به روز رسانی می‌کنیم.

$$X_{rel} = X_{ref} - X \quad (1-3)$$

پس از نسبی کردن نقاط نسبت به مبدأ، آن‌ها را کمک رابطه (۲-۳) نرمال سازی می‌کنیم تا تمام

طول و عرض نقاط به عددی میان صفر و یک به روز رسانی شوند.

$$X_{new} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2-3)$$

در انتهای این مختصات را به عنوان ورودی به شبکه تعیین علائم دست می‌دهیم. با توجه به اینکه معماری هیچ یک از مدل‌ها تغییر نکرده است و تنها داده‌های مختصات به روز رسانی شده‌اند، دقت نهایی مدل به ۹۷ درصد افزایش پیدا کرده است و پیش‌پردازش تاثیر بهسازی در افزایش دقت پروژه داشته است.

مختصات نقاط عطف دست به صورت مطلق

ID : 0	ID : 1	ID : 2	...	ID : 19	ID : 20
[ 307, 642]	[ 376, 595]	[ 421, 524]	...	[ 241, 339]	[252,309]

مختصات نقاط عطف دست نسبت به نقطه ۰ (نقطه مرجع دست)

ID : 0	ID : 1	ID : 2	...	ID : 19	ID : 20
[0,0]	[68,-47]	[113,-117]	...	[-65,-303]	[-55,-333]

مختصات نقاط عطف دست به صورت نرمال‌سازی شده

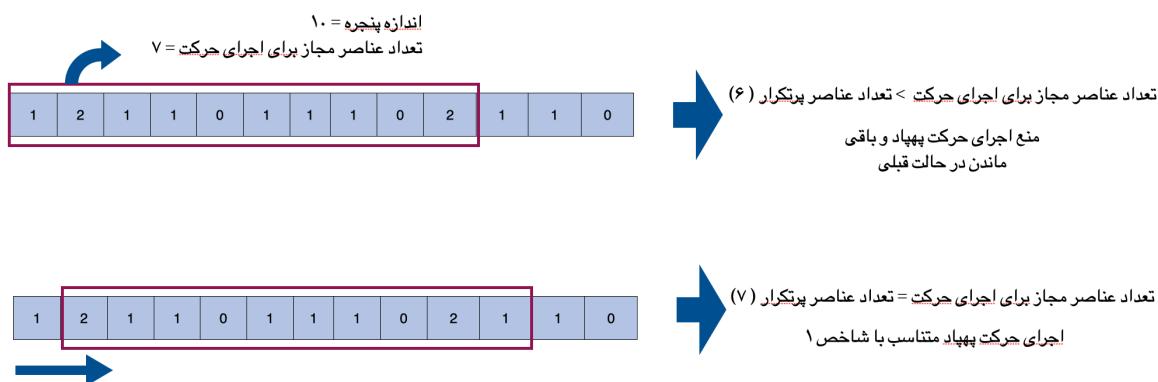
ID : 0	ID : 1	ID : 2	...	ID : 19	ID : 20
[0,0]	[0.20,-0.14]	[0.34,-0.35]	...	[-0.19,-0.90]	[-0.16, -0.99]

شکل ۱۴-۳ نمونه پیش‌پردازش داده‌ها

### ۳-۸-۳ رأی‌گیری پنجره‌ای

با وجود اینکه دقت مدل پیاده‌سازی شده بالا و مقدار قابل قبولی است و عملکرد بسیار چشم‌گیری از خود نشان می‌دهد، در عین حال پیش‌بینی اشتباه مدل می‌تواند عواقب زیان‌باری را به ارمغان آورد، از تجربه ناپسند برای کاربر گرفته تا برخورد پهپاد به اجسام و هزینه‌های مالی گزارف. لذا باید دقت انجام پروژه را از آنچه مدل پیش‌بینی می‌کند نیز بالاتر برد. برای این کار از رأی‌گیری پنجره‌ای استفاده کرده‌ایم. بدین صورت که متغیری را با توجه به تعداد فریم بر ثانیه دوربین در نظر می‌گیریم، هر چه تعداد فریم ضبط شده بر ثانیه در دوربین پهپاد بیشتر باشد متغیر در نظر گرفته شده نیز بیشتر است. در این پروژه از آنجایی که نرخ فریم بر ثانیه دوربین پهپاد برابر با سی است. مقدار این متغیر را برابر با ده در نظر می‌گیریم.

سپس یک حد بالا بین صفر تا یک برای تایید خروجی نهایی تعریف می‌کنیم. مقدار این متغیر نیز در این پروژه برابر با  $7.0$  در نظر گرفته شده است. طبق این راه ما ده فریم متناوب گرفته شده از پهپاد را به مدل پیاده‌سازی شده می‌دهیم اما تنها در صورتی دستور پیش‌بینی شده را به پهپاد می‌دهیم که حد ممکن را بدست آورد. برای مثال اگر هفت یا بیشتر از ده حرکت پیش‌بینی شده، دستور حرکت رو به جلو باشد آنگاه به پهپاد دستور داده می‌شود تا به جلو حرکت کند. در غیر این صورت اگر کمتر از هفت عدد از فریم‌ها یک حرکت یکسانی از دست را پیش‌بینی نکنند، پهپاد در حالت قبلی خود باقی می‌ماند و دستوری به آن داده نمی‌شود.



شکل ۳-۱۵ نمونه رأی‌گیری پنجره‌ای موفق و ناموفق

### ۹-۳ ارتباط با پهپاد دی‌جی‌آی تلو

پهپاد دی‌جی‌آی تلو یکی از پهپادهای کوچک و هوشمند است که به عنوان یک ابزار آموزشی و پژوهشی در زمینه مهندسی کنترل مورد استفاده قرار می‌گیرد. از آنجایی که دی‌جی‌آی تلو دارای سنسور دوربین، سنسور واحد اندازگیری اینرسی<sup>۶۴</sup> و سایر سنسورهای مفید برای جمع آوری داده‌ها برای تحلیل و کنترل در زمان واقعی است، این پهپاد به عنوان یک پلتفرم آموزشی بسیار مناسب برای پیاده‌سازی پروژه ما در نظر گرفته می‌شود.

ارتباط این پهپاد با مدل هوش مصنوعی که توسط برنامه پایتون نوشته شده است، با کمک کتابخانه دی‌جی‌آی تلوپی برقرار می‌شود. این کتابخانه دستورات ابتدایی، که پهپاد توانایی انجام آنها را دارد به صورت دستور العمل‌های بسیار ساده پیاده‌سازی کرده است. در ابتدای اجرای برنامه، با کمک دستورهای کتابخانه دوربین روشن شده و پهپاد بلند می‌شود. سپس در هر لحظه دوربین تصاویر گرفته شده را

<sup>64</sup>Inertial Measurement Unit

با کمک ارتباط بی‌سیم به سرور مربوطه می‌فرستد. در ادامه مدل پیاده‌شده با پردازش تصویر، علامت مربوط به دست را پیش‌بینی می‌کند و با ارتباط بی‌سیم و دستورالعمل مناسب فرمان را به پهپاد ارسال می‌کند.

از جمله دستورهایی از این کتابخانه که در پروژه استفاده شده‌اند در جدول ۳-۱ ذکر شده‌است.

دستورالعمل	حرکت پهپاد
connect()	وصل شدن به پهپاد
streamon()	روشن کردن دوربین
takeoff()	بلند شدن
get_frame_read()	گرفتن تصاویر دوربین
send_rc_control(left_right, forward_backward, up_down, yaw)	حرکت به ۶ جهت
rotate_counter_clockwise(degree)	چرخش ساعتگرد
land()	فروود آمدن
streamoff()	خاموش کردن دوربین
end()	قطع ارتباط با پهپاد

جدول ۳-۱ لیست دستورالعمل‌های استفاده شده در کتابخانه دی‌جی‌آی تلوپی برای ارتباط با پهپاد

### ۱۰-۳ پیاده‌سازی میدانی پروژه

پس از پیاده‌سازی تمام اجزای پروژه و رسیدن به دقت بالا، این پروژه بر روی پهپاد دی‌جی‌آی تلو به اجرا در آمده است. تصویر ۱۶-۳ قسمتی از اجرای میدانی پروژه است که در لینک **گیت‌هاب** و **آپارات** ویدیو کامل آن قابل مشاهده است.

### ۱۱-۳ جمع‌بندی

در این فصل قسمت‌های مختلف پیاده‌سازی شده پروژه با جزئیات کامل توضیح داده شد. که این توضیحات شامل نرم‌افزارها و سخت‌افزارها، کتابخانه‌ها، پیش‌پردازش، پس‌پردازش و کد اصلی که شامل



شکل ۳-۱۶ نمونه تصویر پیاده‌سازی مدل بر روی پهپاد دی‌جی‌آی تلو

سه مدل به صورت پی در پی است می‌شود. دو مدل از این سه مدل وظیفه استخراج ویژگی‌ها و نقاط کلیدی دست و مدل آخر نیز وظیفه کلاس‌بندی علائم ورودی داده شده را دارند. در فصل بعدی به بررسی نتایج به دست آمده از این معماری پیشنهاد شده و ارزیابی و مقایسه معماری خود با دیگر کارهای مشابه می‌پردازیم.

# فصل چهارم

## نتایج و ارزیابی

## ۱-۴ مقدمه

سیستم تشخیص حرکات دست نقش مهمی در زمینه ایجاد تعامل کارآمد بین انسان و ماشین ایفا می‌کند. پیاده‌سازی این سیستم با استفاده از تشخیص علائم دست، باعث افزایش وسیعی در استفاده کاربردی از هوش مصنوعی در صنعت فناوری می‌شود. در این پژوهه، معماری‌های گوناگونی مانند شبکه‌های عصبی پیچشی، شبکه‌های حافظه کوتاه‌مدت بلند، و شبکه‌های عصبی چندلایه‌ای پرسپترونی مورد آزمایش قرار گرفتند تا بهترین پیاده‌سازی برای تشخیص حرکات دست انتخاب شود.

در این فصل، به بررسی نتایج به دست آمده از این پیاده‌سازی‌های متفاوت می‌پردازیم و دقت و عملکرد سیستم در شرایط مختلف را ارزیابی می‌کنیم. همچنین، نقاط قوت و ضعف هر یک از معماری‌های مورد استفاده را تحلیل کرده و پیشنهاداتی برای بهبود سیستم ارائه می‌دهیم. هدف این فصل، ارائه یک تحلیل جامع از کارایی سیستم و شناخت دقیق‌تر از عواملی است که می‌توانند به ارتقاء عملکرد آن کمک کنند.

## ۲-۴ معیارهای ارزیابی مدل‌های بینایی ماشین

## ۱-۲-۴ دقت

دقت مدل معیاری است که نشان می‌دهد یک مدل یادگیری عمیق تا چه اندازه قادر به پیش‌بینی یا تصمیم‌گیری صحیح بر اساس داده‌ها است. این معیار به صورت نسبت مجموع مثبت و منفی واقعی به تعداد کل نمونه‌ها محاسبه می‌شود.

دقت، بصری‌ترین معیار عملکرد است و به نسبت مشاهدات پیش‌بینی شده صحیح به کل مشاهدات اشاره دارد. این معیار برای مقایسه عملکرد مدل‌های مختلف و ارزیابی اثربخشی یک مدل خاص برای یک وظیفه معین استفاده می‌شود. دقت زمانی مناسب است که توزیع کلاس‌ها متعادل باشد و هزینه نتیجه‌های مثبت کاذب و منفی کاذب مشابه باشند [۳۸].

(۱-۴)

$$Accuracy = \frac{True\ Positives + True\ Negatives}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives}$$

## ۴-۲-۴ صحت

صحت، یک معیار آماری برای ارزیابی کیفیت یک مدل پیش‌بینی است. این معیار یکی از کلیدی‌ترین معیارها برای تعیین عملکرد یک مدل، به ویژه در وظایف طبقه‌بندی، محسوب می‌شود. صحت نسبت مثبت واقعی به مجموع مثبت‌های واقعی و مثبت‌های کاذب (نمونه‌هایی که به اشتباه به عنوان مثبت شناسایی شده‌اند) را نشان می‌دهد.

صحت بالا نشان‌دهنده این است که یک مدل در جلوگیری از مثبت‌های کاذب عملکرد خوبی دارد، به این معنا که نمونه‌های منفی را به عنوان مثبت طبقه‌بندی نمی‌کند. این امر به ویژه در برنامه‌هایی که هزینه مثبت کاذب بالا است، اهمیت دارد [۳۹].

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (4-4)$$

## ۴-۲-۴ فراخوانی

فراخوانی، که به عنوان نرخ مثبت واقعی نیز شناخته می‌شود، معیاری است که نشان می‌دهد یک مدل یادگیری عمیق چقدر قادر به تشخیص درست نمونه‌های مثبت از بین کل نمونه‌های یک کلاس خاص است.

فراخوانی زمانی استفاده می‌شود که به حداقل رساندن منفی‌های کاذب از اهمیت ویژه‌ای برخوردار باشد. این بدان معناست که در کاربردهایی که هزینه منفی‌های کاذب بالا است یا از دست دادن نمونه‌های مثبت واقعی ضرر زیادی به همراه دارد، فراخوانی اهمیت بیشتری پیدا می‌کند [۴۰].

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (4-5)$$

## ۴-۲-۴ امتیاز F1

امتیاز F1 معیاری است که میانگین هارمونیک دقت و فراخوانی را محاسبه می‌کند. این امتیاز، که معمولاً به عنوان معیار ارزیابی در طبقه‌بندی باینری و چند کلاسه استفاده می‌شود، دقت و فراخوانی را در یک مشخصه<sup>۱</sup> واحد ادغام می‌کند تا ارزیابی جامعی از عملکرد مدل ارائه دهد.

امتیاز F1 به ویژه زمانی مفید است که داده‌ها نامتعادل باشند، زیرا نه تنها تعداد پیش‌بینی‌های نادرست

<sup>1</sup>Metric

را در نظر می‌گیرد، بلکه نوع خطاها - مثبت کاذب و منفی کاذب - را نیز مد نظر قرار می‌دهد. این امر باعث می‌شود که F1 معیار بهتری برای ارزیابی عملکرد مدل‌هایی باشد که با کلاس‌های نادر و یا داده‌های نامتعادل سروکار دارند [۴۱].

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4-4)$$

### ۳-۴ ارزیابی عملکرد مدل‌ها

این پروژه با مدل‌های گوناگونی پیاده‌سازی شده است تا بتوان بهترین آنها را برای نتیجه نهایی بر روی پهپاد اجرا کرد. معیارهای ارزیابی شامل دقت، صحت<sup>۲</sup>، فراخوانی<sup>۳</sup>، امتیاز F1 و تعداد نمونه‌ها برای هر یک از علائم می‌باشد.

معیارهای دقت، صحت، فراخوانی و امتیاز F1 معیارهای ضروری برای ارزیابی عملکرد مدل‌های بینایی ماشین و یادگیری عمیق هستند. این معیارها هر دو نتیجه مثبت کاذب و منفی کاذب را در نظر می‌گیرند و درک دقیقی از قابلیت‌های پیش‌بینی یک مدل را ارائه می‌دهند. این ارزیابی دقیق با بررسی کردن نقاط قوت و ضعف خاص هر مدل، به اصلاح مدل کمک می‌کند.

### ۱-۳-۴ گزارش معیارهای ارزیابی در مدل‌ها

در روند پروژه معیارهای ارزیابی برای هر <sup>۴</sup> مدل پیاده‌سازی شده به دست آمده است. مقایسه دقت‌ها در جدول ۱-۴ نشان داده شده‌اند و همچنین نتایج ارزیابی مدل شبکه عصبی پیچشی که بالاترین مقادیر ارزیابی را برای تشخیص نه علائم مختلف دست دارد در شکل ۱-۴ آمده است.

### ۲-۳-۴ ماتریس درهم‌ریختگی<sup>۴</sup>

ماتریس درهم‌ریختگی یک ابزار مهم در ارزیابی عملکرد مدل‌های طبقه‌بندی است. این ماتریس نشان‌دهنده تعداد نمونه‌هایی است که به درستی به هر کلاس تخصیص یافته‌اند و همچنین تعداد نمونه‌هایی که به اشتباه به کلاس‌های دیگر تخصیص داده شده‌اند. به عبارت دیگر، هر سطر این ماتریس نشان‌دهنده

<sup>2</sup>Precision

<sup>3</sup>Recall

<sup>4</sup>Confusion Matrix

معماری مدل	دقت داده تست	دقت داده آموزش	دوره	دقت داده
شبکه چند لایه‌ای پرسپترونی	۱۵۰	% ۹۸.۹۳	% ۹۷.۵۵	
شبکه عصبی پیچشی	۱۵۰	% ۹۹.۶۲	% ۹۸.۳۲	
شبکه حافظه طولانی کوتاه مدت	۱۵۰	% ۹۸.۳۶	% ۹۴.۹۰	
شبکه عصبی بازگشتی	۱۵۰	% ۹۷.۹۹	% ۹۵.۴۵	

جدول ۱-۴ جدول ارزیابی دقیقی مدل‌ها

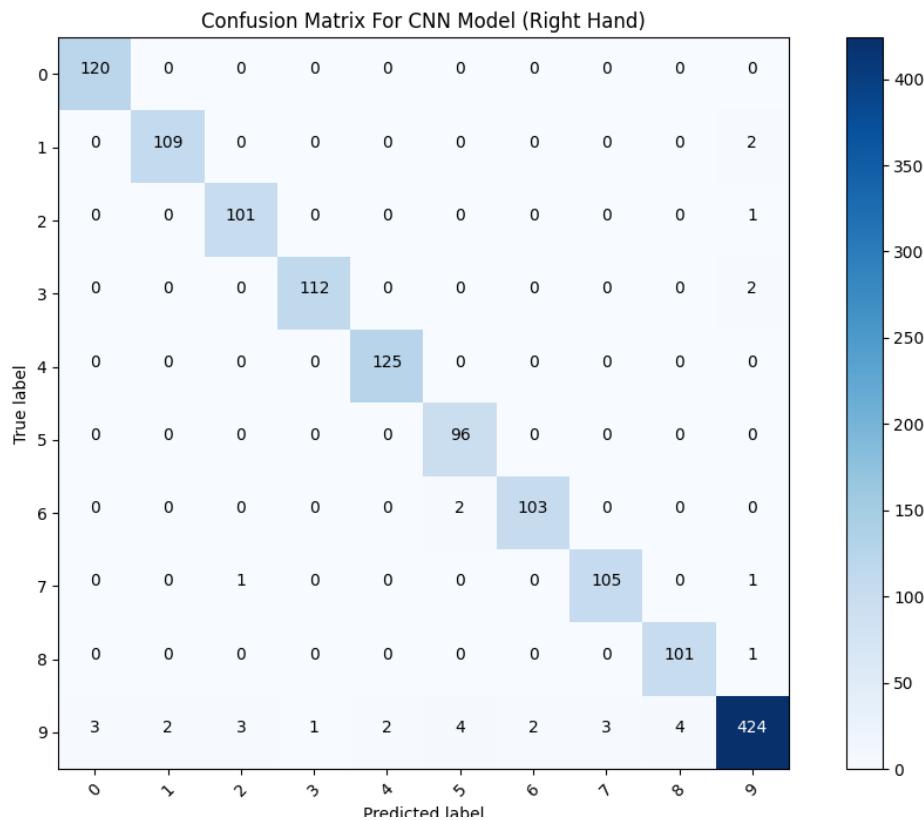
	precision	recall	f1-score	support
0	0.99	0.98	0.99	120
1	0.99	0.98	0.99	111
2	1.00	0.99	1.00	102
3	0.97	0.99	0.98	114
4	1.00	0.97	0.98	125
5	0.99	1.00	0.99	96
6	0.97	0.99	0.98	105
7	1.00	0.98	0.99	107
8	0.97	0.98	0.98	102
9	0.98	0.98	0.98	448
micro avg	0.98	0.98	0.98	1430
macro avg	0.99	0.98	0.99	1430
weighted avg	0.98	0.98	0.98	1430
samples avg	0.98	0.98	0.98	1430

شکل ۱-۴ معیارهای ارزیابی برای تشخیص علائم دست در مدل شبکه عصبی پیچشی

کلاس واقعی است و هر ستون نشان‌دهنده کلاسی است که مدل پیش‌بینی کرده است. اگر خانه  $z_i$  این ماتریس، عدد  $n$  را داشته باشد، این به این معنا است که مدل  $n$  بار نمونه‌های کلاس  $i$  را به درستی به کلاس  $i$  تخصیص داده است.

ماتریس درهم‌ریختگی حاوی اطلاعات مفیدی است زیرا به ما اطلاعات دقیقی از عملکرد مدل در هر کلاس را می‌دهد. از این اطلاعات می‌توان برای ارزیابی عملکرد کلی مدل، شناسایی نقاط ضعف و قوت مدل، و بهبود آن استفاده کرد. همچنان، این ماتریس به ما امکان می‌دهد بررسی کنیم که آیا مدل ما به نسبت هر کلاس اشتباه می‌کند یا اشتباهات آن به کلاس‌های خاصی متمرکز شده‌اند [۴۲]. ماتریس درهم ریختگی برای شبکه عصبی پیچشی که شبکه نهایی انتخاب شده است در شکل ۲-۴ آمده است.

<sup>5</sup>Epoch



شکل ۲-۴ ماتریس درهم ریختگی در مدل شبکه عصبی پیچشی

#### ۴-۴ نمودارهای دقت و خطأ بر حسب دوره<sup>۵</sup>

یک دوره زمانی است که کل مجموعه داده تنها یک بار از طریق شبکه عصبی انتشار پیدا کرده و پس از آن پس انتشار اتفاق می‌افتد. از آنجایی که یک دوره ممکن است بسیار بزرگ باشد و نتوان آن را به یکباره به سیستم وارد کرد، به چند دسته کوچکتر تحت عنوان "دسته"<sup>۶</sup> تقسیم می‌شود. انتقال کل مجموعه داده از طریق یک شبکه عصبی به تنها یکی کافی نیست و باید مجموعه داده را چندین بار به شبکه ارسال کرد. در این پروژه، از یک مجموعه داده محدود استفاده شده و برای بهینه‌سازی یادگیری آن از الگوریتم گرادیان نزولی استفاده می‌کنیم.

تعیین تعداد صحیح دوره‌ها از اهمیت بالایی برخوردار است. با افزایش تعداد دوره‌ها، تعداد دفعات تغییر وزن در شبکه عصبی بیشتر می‌شود و منحنی یادگیری از حالت کم‌برازش<sup>۷</sup> به حالت بهینه<sup>۸</sup> و در نهایت به حالت بیش‌برازش<sup>۹</sup> تغییر می‌یابد. تعداد دوره‌ها باید به گونه‌ای تعیین شود که تعادلی برقرار شود و

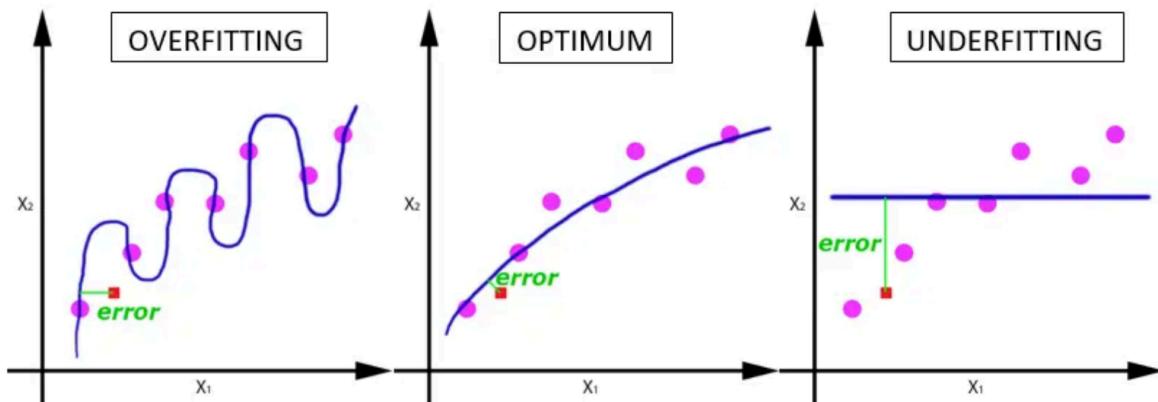
<sup>6</sup>Batch

<sup>7</sup>Underfitting

<sup>8</sup>Optimal

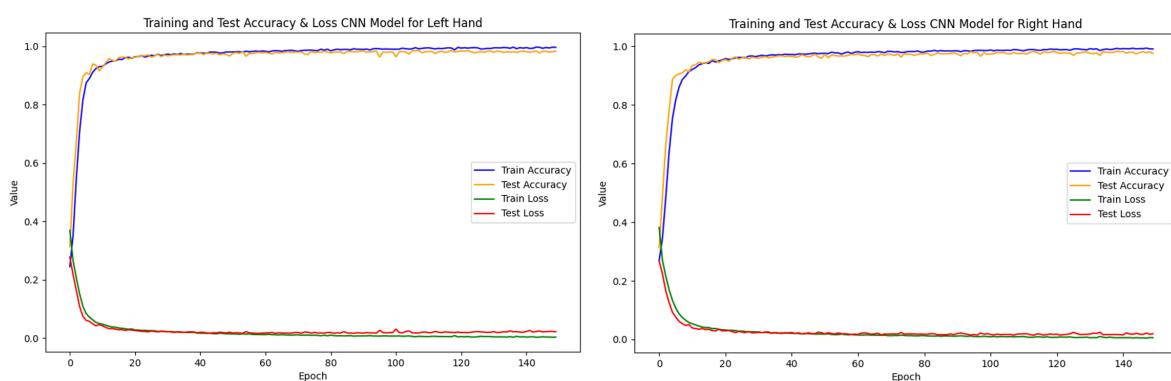
<sup>9</sup>Overfitting

بتوان منحنی یادگیری را به بهترین حالت ممکن رساند.



شکل ۳-۴ منحنی حالت‌های کمبازش، بهینه و بیشبرازش [۴۳]

در این پروژه، به منظور نظارت و ارزیابی عملکرد مدل، نمودارهای دقت و خطأ بر حسب دوره ترسیم شده‌اند، که در شکل ۴-۴ نمودار شبکه عصبی پیچشی آمده است. این نمودارها نشان می‌دهند که چگونه دقت مدل و میزان خطأ در طول زمان تغییر می‌کنند. بررسی این نمودارها می‌تواند به شناسایی نقاط بهینه و جلوگیری از بیشبرازش کمک کند.



شکل ۴-۴ نمودار روند دقت و خطأ در دستهای راست و چپ بر حسب دوره در داده‌های آموزش و تست در مدل شبکه عصبی پیچشی

## ۵-۴ تاثیر پسپردازش رأی‌گیری پنجره‌ای بر دقت

به طور کلی با توجه به جدول ۲-۴ دقت ما پس از پیاده‌سازی رأی‌گیری پنجره‌ای کاهش پیدا کرده است. اما لزوم این الگوریتم برای ما از اهمیت ویژه‌ای برخوردار است، چرا که در بیشتر مواردی که دقت به

درستی بیان نشده است در زمانی است که علائم دست کاربر نمادی را نشان داده اما پهپاد از آن پیروی نمی‌کند. این موضوع می‌تواند سبب ناخوشایندی کاربر شود، اما این الگوریتم سبب شده است تا بسیاری از مواردی که علائم دست در یک یا تعداد کمی از فریم‌ها توسط سیستم اشتباه برداشت شده است، یا حتی کاربر در مدت کوتاهی به اشتباه هدف خود را بیان کند، دستورنهایی صادر نشود تا پهپاد دچار مشکل نشود.

معماری مدل	دقت مدل بدون رأی‌گیری پنجره‌ای	دقت مدل با رأی‌گیری پنجره‌ای
شبکه عصبی چندلایه‌ای پرسپترونی	% ۹۷.۶۷	% ۹۸.۳۷
شبکه عصبی پیچشی	% ۹۸.۲۷	% ۹۸.۹۸
شبکه عصبی حافظه طولانی کوتاه مدت	% ۹۵.۲۴	% ۹۵.۹۲
شبکه عصبی بازگشتی	% ۹۵.۰۲	% ۹۵.۷۰

جدول ۲-۴ جدول ارزیابی تاثیر رأی‌گیری پنجره‌ای در دقت با استفاده از یک ویدیو دوازده هزار فریمی

## ۶-۴ سرعت اجرای برنامه

به منظور رسیدن به یکی از اهداف اصلی این پروژه، که بی‌درنگ بودن آن می‌باشد، لازم است که میزان پاسخ‌گویی مدل‌ها نیز مورد بررسی و ارزیابی قرار گیرد. در جدول ۳-۴ زمان پاسخ‌گویی مدل‌ها از زمانی که داده‌ها از طریق دوربین خوانده می‌شوند تا زمانی که دستور به پهپاد داده می‌شود، آورده شده است.

معماری مدل	زمان پاسخ‌گویی مدل
شبکه چندلایه‌ای پرسپترونی	$39 \pm 4$ میلی‌ثانیه
شبکه عصبی پیچشی	$41 \pm 7$ میلی‌ثانیه
شبکه عصبی حافظه طولانی کوتاه مدت	$44 \pm 1$ میلی‌ثانیه
شبکه عصبی بازگشتی	$41 \pm 2$ میلی‌ثانیه

جدول ۳-۴ جدول ارزیابی زمان پاسخ‌گویی مدل‌ها

## ۷-۴ سخت‌افزار مورد نیاز

این پروژه باید به گونه‌ای اجرا می‌شد که بر روی ساده‌ترین سیستم‌های کامپیوتری نیز قابل اجرا باشد، زیرا سخت‌افزار پهپادها به طور معمول دارای پردازنده و حافظه‌های موقت<sup>۱۰</sup> ضعیف‌تری هستند. همچنین، استفاده از پردازنده گرافیکی ممکن نبود، چرا که پهپادها فاقد پردازنده‌های گرافیکی می‌باشند. عماری‌های پیاده‌سازی شده به نحوی طراحی شدند که تعادل میان دقت و بهره‌وری از سخت‌افزار حفظ شود، به طوری که هم قابلیت اجرا به صورت بی‌درنگ را داشته باشند و هم امکان پیاده‌سازی آن‌ها بر روی پهپاد فراهم باشد. از این‌رو، عماری‌ها به گونه‌ای پیاده‌سازی شدند که بر روی پردازنده‌های سبک با میزان حافظه موقت کم اجرا شوند. میزان استفاده از حافظه موقت برای اجرای عماری‌های پیاده‌سازی شده در این پروژه به شرح زیر است:

معماری مدل	حجم مدل	حافظه موقت مورد نیاز برای اجرا
شبکه چندلایه‌ای پرسپترونی	۱۰۹ کیلوبايت	۹.۹۹۲ کیلوبايت
شبکه عصبی پیچشی	۲۷۰ کیلوبايت	۱۰.۹۵۳ کیلوبايت
شبکه عصبی حافظه طولانی کوتاه مدت	۶۲۷ کیلوبايت	۱۳.۹۷۴ کیلوبايت
شبکه عصبی بازگشتی	۴۰۱ کیلوبايت	۱۳.۱۵۸ کیلوبايت

جدول ۴-۴ جدول ارزیابی حافظه موقت موردنیاز مدل‌ها

## ۸-۴ مقایسه دقت پروژه با کارهای مشابه

پروژه پیاده‌سازی از دقت بسیار بالایی برخوردار است که این دقت به دلیل ترکیب پیش‌پردازش، کتابخانه مدیاپایپ، مدل کلاس‌بندی و پس‌پردازش است.

در این قسمت به مقایسه دقت پروژه پیاده‌سازی شده با پروژه‌های مشابه با هدف کنترل پهپاد با کمک تشخیص علائم دست پرداخته‌ایم.

<sup>10</sup>Ram

نام مقاله	تعداد علائم دست	دقت تشخیص علائم دست
برنامه ما	۹	% ۹۸.۳۲
پهپادهای کنترل شده با علائم دست به صورت متن باز [۵]	۵	% ۹۷.۴۷۱
روشهای تشخیص علائم دست به صورت بی‌درنگ [۱۷]	۶	% ۹۳.۸
تشخیص علائم دست برای کنترل پهپاد با استفاده از یادگیری عمیق [۶]	۹	% ۸۳.۳
علائم یوایوی: مجموعداده برای یوایوی کنترل و تشخیص علائم [۱۸]	۱۳	% ۹۱.۹
تشخیص علائم دست برای استفاده‌های بی‌درنگ [۱۹]	۶	% ۹۰.۸
روشی بهبود یافته برای تشخیص علائم دست با استفاده از نقاط کلیدی و جعبه مرزی [۲۰]	۶	% ۹۴
تشخیص حرکت دست با استفاده از تکنیک‌های بینایی ماشین [۴۴]	۶	% ۹۳.۱
تشخیص علائم دست با استفاده از تکنیک‌های پردازش تصویر و استخراج ویژگی [۴۵]	۲۸	% ۸۹.۵۴
استفاده از تشخیص حرکت دست برای برنامه راهنمای کاربر با استفاده از مدیاپایپ [۲۹]	۱۰	% ۹۵
دست‌های مدیاپایپ: تشخیص بی‌درنگ دست بر روی دستگاهها [۱۲]	۸	% ۹۴.۷
سیستم تشخیص حرکت دست مبتنی بر بینایی کامپیوتر و یادگیری ماشین [۴۶]	۲۸	% ۹۳.۷۲
تشخیص علائم دست با استفاده از مدل‌های پنهان مارکوف [۴۷]	۵	% ۹۲.۱
تشخیص علائم دست با استفاده از کینکت [۲۴]	۳۸	% ۸۴
تشخیص علائم دست با استفاده از شبکه‌های عصبی [۴۸]	۱۰	% ۸۹

جدول ۴-۵ جدول مقایسه پروژه با کارهای مشابه

#### ۹-۴ جمع‌بندی

مدل‌های ارائه شده در این پروژه دارای دقت و کارایی مناسبی هستند که امکان استفاده آن‌ها در کاربردهای واقعی را فراهم می‌کند. این مدل‌ها قادرند به خوبی علائم دست را تشخیص داده و از آن‌ها در عملیات مختلفی مانند کنترل دستگاهها و رابطه‌های کاربری استفاده کنند.

علاوه بر این، با مقایسه عملکرد این پروژه با کارهای مشابه در حوزه تشخیص علائم دست، مشخص شده است که پروژه‌ی ما دارای دقت مشابه یا حتی بالاتری است، به ویژه اگر تعداد کلاس‌های علائم دست در

نظر گرفته شود. این نکته نشان می‌دهد که مدل‌های ارائه شده به خوبی تنوع و پیچیدگی علائم دست را درک می‌کنند و قادر به تشخیص آن‌ها هستند، که این امر یکی از چالش‌های اصلی در این زمینه است. با توجه به این نتایج، می‌توانیم اطمینان داشته باشیم که پروژه‌ی ما قادر است به عنوان یک راه حل کارآمد برای تشخیص علائم دست در برنامه‌ها و سیستم‌های واقعی مورد استفاده قرار بگیرد. در فصل بعدی به جمع‌بندی کلی این پروژه و ارائه پیشنهاداتی برای بهبود عملکرد سیستم طراحی شده می‌پردازیم.

## فصل پنجم

### نتیجه گیری و پیشنهادات

## ۱-۵ مقدمه

در این فصل، ما به بررسی نتایج بدست آمده از این پروژه و ارائه پیشنهاداتی می‌پردازیم که می‌توانند عملکرد سیستم کنترل پهپاد را از جنبه‌های مختلف بهبود بخشنند. نتایج بدست آمده در این پروژه نشان‌دهنده دقت بالای سیستم در تشخیص علائم و انتقال دستورات به پهپاد می‌باشد. با این حال، همچنان می‌توان بهبودهایی در عملکرد سیستم اعمال کرد تا دقت و کارایی آن افزایش یابد. این پیشنهادات شامل افزایش سرعت اجرای برنامه، بهبود دقت پیش‌بینی علائم و کاهش محدودیت‌های منابع می‌باشند. هدف از این فصل، ارائه راهکارهای عملی و کاربردی است که بتوانند سیستم را به سطح بالاتری از کارایی و اطمینان برسانند.

## ۲-۵ نتیجه گیری

نتایج بدست آمده در این پروژه نشان‌دهنده عملکرد دقیق و موفقیت‌آمیز آن است. این پروژه به تمامی اهداف از پیش تعیین شده، از جمله دقت بالا، صحت قابل قبول، فراخوانی مناسب، امتیاز F1 بالا، اجرای بی‌درنگ، رضایت کاربر، خطای پایین و قابلیت پیاده‌سازی بر روی پهپاد دست یافته است. عملکرد دقیق سیستم در تشخیص و کنترل پهپاد با استفاده از علائم دست، نشان‌دهنده پتانسیل بالای این روش برای کاربردهای مختلف در صنعت و تکنولوژی است.

## ۳-۵ پیشنهادات

در این بخش پیشنهاداتی برای بهبود عملکرد پهپاد با کنترل دست بیان شده است.

### ۱-۳-۵ محدودیت وجود پهپادهای منبع باز

در مورد محدودیت‌های استفاده از پهپادهای منبع باز و شرکتی برای پروژه کنترل پهپاد با استفاده از علائم دست بر مبنای بینایی ماشین، چندین موضوع موردنظر وجود دارد. ابتدا، در مورد پهپادهای شرکتی، باید توجه داشت که این پهپادها معمولاً دارای سیستم‌های بسته و غیرقابل تغییر هستند که مانع از انجام تغییرات و سفارشی‌سازی‌های مورد نیاز برای اجرای الگوریتم‌های پیچیده بر روی آنها می‌شوند. همچنین، محدودیت‌های قانونی و مقرراتی مانند نیاز به مجوزهای خاص برای استفاده از پهپادهای تجاری در برخی کشورها می‌توانند چالش‌هایی را ایجاد کنند.

در مورد پهپادهای منبع باز، محدودیت‌های متفاوتی وجود دارد. این پهپادها ممکن است دارای سخت‌افزارهای قدرتمندی نباشند که بتوانند الگوریتم‌های پیچیده پردازش تصویر را اجرا کنند. همچنین، عدم پشتیبانی کامل از زبان‌های برنامه‌نویسی مدرن می‌تواند باعث پیچیدگی در توسعه و اجرای برنامه‌ها شود. با در نظر گرفتن این محدودیت‌ها، راهکار ساخت یک پهپاد سفارشی با سخت‌افزار مناسب و قابلیت برنامه‌نویسی مورد نیاز، می‌تواند راه حلی مناسب باشد. این رویکرد امکان اجرای الگوریتم‌های پیچیده و پیاده‌سازی برنامه‌های سفارشی را فراهم می‌کند، که باعث بهبود کارایی و انعطاف‌پذیری سیستم در ارتباط با کنترل پهپاد با علائم دست خواهد شد.

### ۲-۳-۵ وجود چندین دست در تصویر

وجود چندین دست در تصویر یک چالش بزرگ در برنامه است. در این مورد باید به این نکته توجه کنیم که در یک فریم تصویر، ممکن است چندین دست قابل مشاهده باشند که این می‌تواند برای الگوریتم‌های تشخیص دست مشکل‌ساز شود. زیرا الگوریتم‌ها باید بتوانند دست‌ها را تفکیک کرده و علائم متفاوت آنها را تشخیص دهند. همچنین ممکن است دست‌های غیرمرتبط با کاربر، مانند دست‌های در پس‌زمینه یا دست‌های دیگر افراد، نیز در تصویر حضور داشته باشند که این موضوع می‌تواند دقت الگوریتم را کاهش دهد.

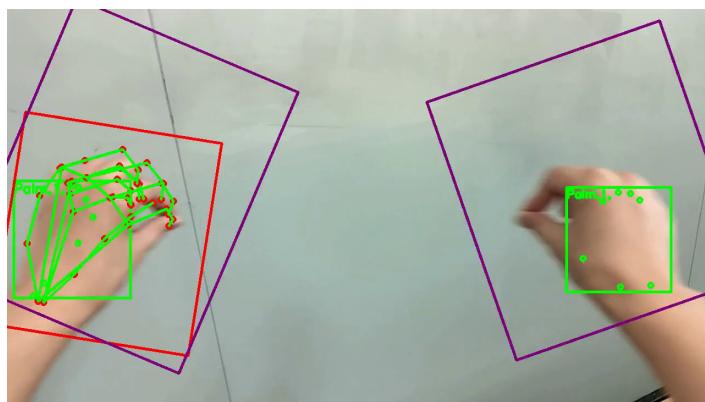
یکی از راهکارهای موردنظر برای حل این چالش، شخصی‌سازی مدل تشخیص دست با توجه به داده‌هایی که از کاربران جمع‌آوری می‌شود، است. این روش می‌تواند به کمک داده‌های جمع‌آوری شده از دست‌های واقعی کاربران، به مدل کمک کند تا الگوهای مختلف دست‌ها را بیشتر بشناسد و از این طریق دقت تشخیص را افزایش دهد. با این حال، این روش نیازمند زمان و تلاش برای جمع‌آوری و برچسب‌گذاری داده‌های موردنیاز است و ممکن است برای کاربران خوشایند نباشد.

راهکار دیگری که می‌توان برای حل این مسئله در نظر گرفت، استفاده از یک سیستم ارتباطی میان دست و پهپاد است. به این ترتیب، کاربر می‌تواند یک دستبند یا سیمی را در دست خود داشته باشد که این دستبند به عنوان نشانه تشخیص دست و ارتباط بین کاربر و پهپاد عمل کند. با این روش، پهپاد تنها دستی را که دارای دستبند است شناسایی می‌کند و به دستورات آن پاسخ می‌دهد. این راهکار می‌تواند از دقت و سرعت الگوریتم تشخیص بهره‌مند شود و به کاربر اطمینان بیشتری بدهد. با این حال، برای اجرای این روش نیازمند توسعه و اعمال یک مدل دیگر برای تشخیص دست با دستبند است که این می‌تواند زمان و هزینه بیشتری را به بار آورد.

### ۳-۳-۵ شناسایی یک دست در دو مرتبه

یکی از مشکلاتی که بسیاری از استفاده‌کنندگان کتابخانه مدیاپایپ با آن مواجه شده‌اند، مربوط به شناسایی یک دست در دو مستطیل متفاوت و در نتیجه دو خروجی است. این مشکل می‌تواند به علت همپوشانی بین دو مستطیلی که دست را شناسایی می‌کنند، ایجاد شود. با این حال، این مشکل برای پژوهه خاصی که ما در دست داریم تفاوتی ایجاد نکرده است، زیرا ما یک گزینه را انتخاب کرده و علائم دست را در آن پیش‌بینی می‌کنیم.

از بین بردن این مشکل می‌تواند بهبود و بهینه‌سازی کارایی این کتابخانه را برای سایر استفاده‌کنندگان افزایش دهد. برای حل این مشکل، می‌توان محدودیتی برای همپوشانی مستطیل‌های حاوی دست را در نظر گرفت. به این صورت که اگر دو مستطیل بیشتر از حد در نظر گرفته شده همپوشانی داشته باشند، یعنی یک دست در دو تصویر تشخیص داده شده باشد، تنها یکی از آنها به مدل بعدی ارسال شود.



شکل ۱-۵ شناسایی یک دست دو مرتبه

### ۴-۳-۵ اجرا بر روی پردازندۀ‌های گرافیکی

پژوهه ما به علت عدم دسترسی به پهپادی که از پردازندۀ‌های گرافیکی برای اجرای مدل‌ها استفاده می‌کند، به صورتی پیاده‌سازی شده است که تنها بر روی پردازندۀ مرکزی قابل اجرا است. با این حال، اگر پهپادی با توانایی اجرای مدل‌ها بر روی پردازندۀ‌های گرافیکی در دسترس باشد، سه مدل که به صورت پی‌درپی اجرا می‌شوند می‌توانند بر روی پردازندۀ‌های گرافیکی اجرا شده تا عملکرد سیستم را ارتقا داده و تجربه کاربری را بهبود بخشنند. استفاده از پردازندۀ‌های گرافیکی برای اجرای مدل‌ها می‌تواند سرعت پردازش را بسیار افزایش دهد، زیرا کارت‌های گرافیکی معمولاً دارای تعداد زیادی هسته محاسباتی هستند که برای پردازش موازی و سریع داده‌ها بسیار مناسب هستند. این امر می‌تواند منجر به بهبود قابل توجهی

در زمان پاسخ و دقت تشخیص علائم دست شود و تجربه کاربری را بهبود بخشد. به علاوه، با استفاده از پردازنده‌های گرافیکی، می‌توانیم بار محاسباتی را از پردازنده مرکزی کاهش داده و عملکرد کلی سیستم را بهتر کنیم.

### ۵-۳-۵ اجرای پروژه با برنامه سی پلاس پلاس

با توجه به مزایای زبان سی پلاس نسبت به پایتون، انتخاب این زبان برای انجام این پروژه می‌تواند به بهبود عملکرد و کارایی سیستم کمک شایانی کند. در ادامه به برخی از مزایای استفاده از زبان سی پلاس پلاس نسبت به پایتون در این پروژه پرداخته شده است:

- عملکرد و سرعت اجرای بالا: سی پلاس پلاس یک زبان برنامه‌نویسی کامپایل شده است که به کد ماشین تبدیل می‌شود، به همین دلیل عملکرد و سرعت اجرای بسیار بالاتری نسبت به پایتون دارد که یک زبان تفسیری است. این ویژگی برای پروژه‌های بی‌درنگ مانند کنترل پهپاد با استفاده از علائم دست بسیار حیاتی است، زیرا زمان پاسخ‌دهی سریع و دقیق در این سیستم‌ها اهمیت زیادی دارد.
- مدیریت دقیق حافظه: سی پلاس پلاس امکان مدیریت دقیق حافظه را فراهم می‌کند. این ویژگی می‌تواند بهینه‌سازی استفاده از منابع سیستم و کاهش مصرف حافظه را به دنبال داشته باشد، که برای سیستم‌هایی با منابع محدود مانند پهپادها بسیار مهم است. مدیریت صحیح حافظه می‌تواند به جلوگیری از نشت حافظه و افزایش پایداری سیستم کمک کند.
- کتابخانه‌ها و چارچوب‌های قدرتمند: سی پلاس پلاس دارای کتابخانه‌ها و چارچوب‌های قدرتمندی برای پردازش تصویر و بینایی ماشین مانند اپن سی‌وی است. اپن سی‌وی به زبان سی پلاس پلاس نوشته شده و عملکرد بهتری نسبت به معادله‌ای پایتونی خود دارد. این کتابخانه قابلیت‌های متعددی را برای پردازش تصاویر، شناسایی و تشخیص اشیا فراهم می‌کند که می‌تواند در پروژه‌های بینایی ماشین بسیار موثر باشد.
- دسترسی و کنترل سطح پایین سخت‌افزار: سی پلاس پلاس امکان دسترسی و کنترل سطح پایین سخت‌افزار را فراهم می‌کند. این ویژگی برای کنترل مستقیم پهپاد و بهینه‌سازی ارتباط با سنسورها و موتورها بسیار مفید است. با استفاده از سی پلاس پلاس می‌توان به طور مستقیم با

<sup>1</sup>Multithreading

اجزای سختافزاری پهپاد ارتباط برقرار کرده و عملکرد سیستم را بهینه کرد.

- پشتیبانی قوی از چندریسمانی<sup>۱</sup>: سی پلاس پلاس پشتیبانی قوی تری از چندریسمانی و پردازش موازی دارد. این قابلیت می‌تواند برای بهبود کارایی و پاسخ‌دهی سریع در پردازش‌های پیچیده بینایی ماشین و کنترل پهپاد بسیار مهم باشد. با استفاده از چندریسمانی، می‌توان عملیات مختلف را به صورت همزمان انجام داد و زمان اجرای کل سیستم را کاهش داد.

### ۶-۳-۵ استفاده از مدل‌های موازی برای تشخیص علائم دست

برای بهبود دقت و کارایی پروژه کنترل پهپاد با استفاده از علائم دست، می‌توان از رویکردی مبتنی بر اجرای موازی مدل‌ها بهره برد. در این روش، تصویر تشخیص داده شده از دست به طور همزمان به دو مدل با معماری‌های مختلف ارسال می‌شود. به عنوان مثال، یک مدل می‌تواند از معماری شبکه‌های عصبی پیچشی و مدل دیگر از معماری حافظه بلندمدت کوتاه‌مدت استفاده کند. تنها در صورتی که هر دو مدل تشخیص یکسانی از علائم دست داشته باشند، دستور نهایی به پهپاد ارسال می‌شود. این رویکرد می‌تواند به طور قابل توجهی دقت تشخیص علائم دست را افزایش دهد و ریسک‌های مالی مرتبط با اشتباهات تشخیصی را کاهش دهد.

### ۷-۳-۵ توسعه برنامه با استفاده از سایر قسمت‌های بدن انسان

یکی از مسیرهای جذاب برای توسعه آینده پروژه کنترل پهپاد با علائم دست، گسترش سیستم به گونه‌ای است که بتوان از قسمت‌های دیگر بدن انسان برای کنترل حرکت پهپاد استفاده کرد. این رویکرد می‌تواند به ایجاد تعاملات طبیعی‌تر و راحت‌تر بین کاربر و پهپاد منجر شود. در زیر به برخی از این ایده‌ها و مزایای آن‌ها پرداخته می‌شود.

#### کنترل پهپاد با حرکات چشم

یکی از نوآورانه‌ترین روش‌ها برای کنترل پهپاد، استفاده از حرکات چشم کاربر است. این سیستم می‌تواند به وسیله ردیابی حرکت چشم کاربر، مکانی نسبی را که کاربر به آن نگاه می‌کند، شناسایی کرده و پهپاد را به سمت آن نقطه هدایت کند. برای پیاده‌سازی این روش، از تکنیک‌های پیشرفته بینایی ماشین و ردیابی چشم استفاده می‌شود.

استفاده از حرکات چشم برای کنترل پهپاد مزایای متعددی دارد. این روش می‌تواند دقیق و کارایی سیستم را افزایش دهد، زیرا حرکات چشم نقاط دقیق‌تری را نسبت به حرکات دست مشخص می‌کنند. این امر به پهپاد امکان می‌دهد تا با دقیق‌تری به نقاط مورد نظر حرکت کند. علاوه بر این، استفاده از چشم برای کنترل پهپاد می‌تواند بسیار طبیعی‌تر و راحت‌تر از حرکات دست باشد، به ویژه در شرایطی که دست‌ها مشغول هستند یا امکان استفاده از آن‌ها وجود ندارد. این روش همچنین می‌تواند خستگی ناشی از استفاده مداوم از دست‌ها برای کنترل پهپاد را کاهش دهد، که در نتیجه تجربه کاربری بهتری را فراهم می‌کند.

### کنترل پهپاد با حرکات سر

روش دیگر برای کنترل پهپاد، استفاده از حرکات سر است. در این سیستم، کاربر می‌تواند با چرخش سر به جهات مختلف، مسیر حرکت پهپاد را تعیین کند. حرکات سر به طور طبیعی و بدون نیاز به تجهیزات اضافی قابل انجام هستند، که این امر تعامل با پهپاد را ساده‌تر می‌کند. استفاده از حرکات سر به ویژه در محیط‌هایی که استفاده از دست‌ها ممکن نیست (مثل محیط‌های کاری خاص یا شرایط پزشکی) بسیار مفید است. این روش می‌تواند به سادگی و طبیعی بودن تعامل با پهپاد کمک کند و تجربه کاربری را بهبود بخشد.

## ۴-۵ جمع‌بندی

در این فصل، به نتیجه بدست آمده و بررسی چندین چالش و محدودیت‌های سیستم کنترل پهپاد پرداخته‌شد و راهکارهای متعددی برای بهبود عملکرد آن ارائه شد. اولین محدودیت، استفاده از پهپادهای منبع باز است که با وجود مزایای فراوان، ممکن است دارای محدودیت‌های سخت‌افزاری و نرم‌افزاری باشند. برای غلبه بر این محدودیت، پیشنهاد شد که از پهپادهای با قابلیت‌های پیشرفته‌تر و بهینه‌سازی کدهای نرم‌افزاری استفاده شود. همچنین، راهکارهایی برای مدیریت چندین دست در تصویر و جلوگیری از شناسایی نادرست یک دست در دو مستطیل ارائه شد که می‌تواند دقیق‌تر سیستم را بهبود بخشد.

در ادامه، به اهمیت اجرای پروژه روی پردازنده‌های گرافیکی و برنامه‌نویسی به زبان سی‌پلاس پلاس پرداختیم که می‌تواند سرعت اجرای برنامه را افزایش دهد. استفاده از مدل‌های موازی برای تشخیص علائم دست نیز به عنوان یک راهکار مؤثر برای بهبود دقیق‌تر سیستم را بهبود بخشد. همچنین، روش‌های

جدیدی مانند کنترل پهپاد با حرکات چشم و سر معرفی شدند که امکانات بیشتری برای کاربران فراهم می‌کنند.

در نتیجه، با توجه به بررسی‌ها و پیشنهادات ارائه شده در این فصل، می‌توان انتظار داشت که با اعمال این بهبودها، سیستم کنترل پهپاد به سطح بالاتری از کارایی و اطمینان دست یابد. این راهکارها نه تنها عملکرد سیستم را بهبود می‌بخشند، بلکه تجربه کاربری بهتری را نیز فراهم می‌آورند.

## منابع و مراجع

- [1] Walter, Ian and Khadr, Monette. Gesture controlled drone.
- [2] Puri, Vikram, Nayyar, Anand, and Raja, Limesh. Agriculture drones: A modern breakthrough in precision agriculture. *Journal of Statistics and Management Systems*, 20(4):507–518, 2017.
- [3] Gatteschi, Valentina, Lamberti, Fabrizio, Paravati, Gianluca, Sanna, Andrea, Demartini, Claudio, Lisanti, Alberto, and Venezia, Giorgio. New frontiers of delivery services using drones: A prototype system exploiting a quadcopter for autonomous drug shipments. in 2015 IEEE 39th annual computer software and applications conference, vol. 2, pp. 920–927. IEEE, 2015.
- [4] Moore, T. Nypd considering using drones to fight crime. *New York Daily*, 2014.
- [5] Natarajan, Kathiravan, Nguyen, Truong-Huy D, and Mete, Mutlu. Hand gesture controlled drones: An open source library. in 2018 1st International Conference on Data Intelligence and Security (ICDIS), pp. 168–175. IEEE, 2018.
- [6] Hadri, Soubhi. Hand gestures for drone control using deep learning. 2018.
- [7] Zhu, Pengfei, Wen, Longyin, Bian, Xiao, Ling, Haibin, and Hu, Qinghua. Vision meets drones: A challenge. arXiv preprint arXiv:1804.07437, 2018.
- [8] Guvenc, Ismail, Koohifar, Farshad, Singh, Simran, Sichitiu, Mihail L, and Matolak, David. Detection, tracking, and interdiction for amateur drones. IEEE

- Communications Magazine, 56(4):75–81, 2018.
- [9] Hassanalian, Mostafa and Abdelkefi, Abdessattar. Classifications, applications, and design challenges of drones: A review. *Progress in Aerospace sciences*, 91:99–131, 2017.
- [10] Clough, Sharice and Duff, Melissa C. The role of gesture in communication and cognition: Implications for understanding and treating neurogenic communication disorders. *Frontiers in Human Neuroscience*, 14:323, 2020.
- [11] Kang, Seokmin and Tversky, Barbara. From hands to minds: Gestures promote understanding. *Cognitive Research: Principles and Implications*, 1:1–15, 2016.
- [12] Zhang, Fan, Bazarevsky, Valentin, Vakunov, Andrey, Tkachenka, Andrei, Sung, George, Chang, Chuo-Ling, and Grundmann, Matthias. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*, 2020.
- [13] Yoo, Minjeong, Na, Yuseung, Song, Hamin, Kim, Gamin, Yun, Junseong, Kim, Sangho, Moon, Changjoo, and Jo, Kichun. Motion estimation and hand gesture recognition-based human–uav interaction approach in real time. *Sensors*, 22(7):2513, 2022.
- [14] Ma, Yuntao, Liu, Yuxuan, Jin, Ruiyang, Yuan, Xingyang, Sekha, Raza, Wilson, Samuel, and Vaidyanathan, Ravi. Hand gesture recognition with convolutional neural networks for the multimodal uav control. in 2017 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS), pp. 198–203. IEEE, 2017.
- [15] Hu, Bin and Wang, Jiacun. Deep learning based hand gesture recognition and uav flight controls. *International Journal of Automation and Computing*, 17(1):17–29, 2020.

- [16] Sarkar, Ayanava, Patel, Ketul Arvindbhai, Ram, RK Ganesh, and Kapoor, Geet Krishna. Gesture control of drone using a motion controller. in 2016 international conference on industrial informatics and computer systems (ciics), pp. 1–5. IEEE, 2016.
- [17] Fang, Yikai, Wang, Kongqiao, Cheng, Jian, and Lu, Hanqing. A real-time hand gesture recognition method. in 2007 IEEE International Conference on Multimedia and Expo, pp. 995–998. IEEE, 2007.
- [18] Perera, Asanka G, Wei Law, Yee, and Chahl, Javaan. Uav-gesture: A dataset for uav control and gesture recognition. in Proceedings of the European Conference on Computer Vision (ECCV) Workshops, pp. 0–0, 2018.
- [19] Murugeswari, M and Veluchamy, S. Hand gesture recognition system for real-time application. in 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, pp. 1220–1225. IEEE, 2014.
- [20] Dang, Tuan Linh, Tran, Sy Dat, Nguyen, Thuy Hang, Kim, Suntae, and Monet, Nicolas. An improved hand gesture recognition system using keypoints and hand bounding boxes. Array, 16:100251, 2022.
- [21] Chen, Boxu, Yu, Lixin, Meng, Xiao, and Hua, Yang. Visual gesture recognition based on hand key points. in Journal of Physics: Conference Series, vol. 2024, p. 012037. IOP Publishing, 2021.
- [22] Waqar, Ahsan, Othman, Idris, Hamah Sor, Nadhim, Alshehri, Abdullah Mohammed, Almujibah, Hamad R, Alotaibi, Badr Saad, Abuhussain, Mohammed Awad, Bageis, Abdulrahman S, Althoey, Fadi, Hayat, Saleh, et al. Modeling relation among implementing ai-based drones and sustainable construction project success. Frontiers in Built Environment, 9:1208807, 2023.
- [23] Ghazi, Georges and Voyer, Julien. Use of a dji tello drone as an educational plat-

- form in the field of control engineering. Proceedings of the Canadian Engineering Education Association (CEEA), 2023.
- [24] Li, Yi. Hand gesture recognition using kinect. in 2012 IEEE International Conference on Computer Science and Automation Engineering, pp. 196–199. IEEE, 2012.
- [25] Introduction to tensorflow. <https://www.tensorflow.org/learn>. (Accessed on 06/02/2024).
- [26] scikit-learn - wikipedia. <https://en.wikipedia.org/wiki/Scikit-learn>. (Accessed on 06/02/2024).
- [27] About keras 3. <https://keras.io/about/>. (Accessed on 06/02/2024).
- [28] Lugaresi, Camillo, Tang, Jiuqiang, Nash, Hadon, McClanahan, Chris, Uboweja, Esha, Hays, Michael, Zhang, Fan, Chang, Chuo-Ling, Yong, Ming, Lee, Juhyun, et al. Mediapipe: A framework for perceiving and processing reality. in Third workshop on computer vision for AR/VR at IEEE computer vision and pattern recognition (CVPR), vol. 2019, 2019.
- [29] Harris, Moh, Agoes, Ali Suryaperdana, et al. Applying hand gesture recognition for user guide application using mediapipe. in 2nd International Seminar of Science and Applied Technology (ISSAT 2021), pp. 101–108. Atlantis Press, 2021.
- [30] What is numpy? — numpy v1.26 manual. <https://numpy.org/doc/stable/user/whatisnumpy.html>. (Accessed on 06/02/2024).
- [31] Introduction to matplotlib - geeksforgeeks. <https://www.geeksforgeeks.org/python-introduction-matplotlib/>. (Accessed on 06/02/2024).
- [32] About - opencv. <https://opencv.org/about/>. (Accessed on 06/02/2024).

- [33] Bhujbal, Kunal and Barahate, Sachin. Custom object detection based on regional convolutional neural network & yolov3 with dji tello programmable drone. in 7th International Conference on Innovation & Research in Technology & Engineering (ICIRTE), 2022.
- [34] Multilayer perceptrons in machine learning: A comprehensive guide | datacamp. <https://www.datacamp.com/tutorial/multilayer-perceptrons-in-machine-learning>. (Accessed on 06/07/2024).
- [35] Introduction to convolution neural network - geeksforgeeks. <https://www.geeksforgeeks.org/introduction-convolution-neural-network/>. (Accessed on 06/07/2024).
- [36] Feng, Weijiang, Guan, Naiyang, Li, Yuan, Zhang, Xiang, and Luo, Zhigang. Audio visual speech recognition with multimodal recurrent neural networks. pp. 681–688, 05 2017.
- [37] Zarzycki, Krzysztof and Dławiński, Maciej. Lstm and gru neural networks as models of dynamical processes used in predictive control: A comparison of models developed for two chemical reactors. Sensors, 21(16), 2021.
- [38] Accuracy vs. precision vs. recall in machine learning: what's the difference? <https://www.evidentlyai.com/classification-metrics/accuracy-precision-recall#:~:text=Accuracy%20is%20a%20metric%20that,often%20the%20model%20is%20right%3F>. (Accessed on 06/03/2024).
- [39] Precision. <https://c3.ai/glossary/machine-learning/precision/#:~:text=Precision%20is%20one%20indicator%20of,the%20number%20of%20false%20positives>. (Accessed on 06/03/2024).
- [40] Understanding precision, recall, f1-score, and support in machine learning evaluation | by nirajan acharya |

medium.

<https://medium.com/@nirajan.acharya777/>

[understanding-precision-recall-f1-score-and-support-in-machine-learning-evaluation-10f33a2a2a](https://medium.com/@nirajan.acharya777/understanding-precision-recall-f1-score-and-support-in-machine-learning-evaluation-10f33a2a2a)  
(Accessed on 06/03/2024).

[41] F1-score in machinaal leren - geeksforgeeks. <https://www.geeksforgeeks.org/f1-score-in-machine-learning/>. (Accessed on 06/03/2024).

[42] Confusion matrix - an overview | sciencedirect topics. <https://www.sciencedirect.com/topics/engineering/confusion-matrix>. (Accessed on 06/03/2024).

[43] www.reddit.com. [https://www.reddit.com/r/aiwars/comments/183u611/neural\\_networks\\_generalization\\_center\\_vs/](https://www.reddit.com/r/aiwars/comments/183u611/neural_networks_generalization_center_vs/). (Accessed on 06/07/2024).

[44] Rios-Soria, David J, Schaeffer, Satu E, and Garza-Villarreal, Sara E. Hand-gesture recognition using computer-vision techniques. 2013.

[45] Sharma, Ashish, Mittal, Anmol, Singh, Savitoj, and Awatramani, Vasudev. Hand gesture recognition using image processing and feature extraction techniques. Procedia Computer Science, 173:181–190, 2020.

[46] Trigueiros, Paulo, Ribeiro, Fernando, and Reis, Luís Paulo. Hand gesture recognition system based in computer vision and machine learning. Developments in medical image processing and computational vision, pp. 355–377, 2015.

[47] Min, Byung-Woo, Yoon, Ho-Sub, Soh, Jung, Yang, Yun-Mo, and Ejima, Toshiaki. Hand gesture recognition using hidden markov models. in 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation, vol. 5, pp. 4232–4235. IEEE, 1997.

[48] Murthy, GRS and Jadon, RS. Hand gesture recognition using neural networks. in 2010 IEEE 2nd international advance computing conference (IACC), pp. 134–138. IEEE, 2010.