



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)  
دانشکده مهندسی کامپیوتر

پایان نامه کارشناسی

## هدایت پهپاد با علائم دست مبتنی بر بینایی ماشین

نگارش  
سارا تاجرنیا

استاد راهنما

دکتر مهدی جوانمردی

اردیبهشت ۱۴۰۳

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ



دانشگاه صنعتی امیرکبیر  
(پلی‌تکنیک تهران)  
دانشکده مهندسی کامپیوتر

پایان نامه کارشناسی

هدایت پهپاد با علائم دست مبتنی بر بینایی  
ماشین

نگارش  
سارا تاجرنیا

استاد راهنما

دکتر مهدی جوانمردی

اردیبهشت 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

تاریخ: اردیبهشت ۱۴۰۳

## تعهدنامه اصالت اثر

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 مأخذ 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، نسخه‌برداری، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مأخذ بلامانع 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 آجرنیا  
اردیبهشت ۱۴۰۳

## چکیده

پهپادهای تجاری که به عنوان هواپیماهای بدون سرنشین<sup>۱</sup> نیز شناخته می‌شوند، به سرعت در حال رایج شدن هستند و در بسیاری از کاربردهای مختلف مانند نظارت برای رویدادهای ورزشی، حمل و نقل تجهیزات و کالاهای اضطراری، فیلمبرداری، عکاسی هوایی و بسیاری از فعالیت‌های دیگر مورد استفاده قرار می‌گیرند.

هدف این پژوهه توسعه سیستمی است که از حرکات دست به عنوان روشی برای کنترل پرواز پهپاد استفاده شود. بدین صورت که با استفاده از روش‌های بینایی ماشین<sup>۲</sup>، روشی بصری برای ارتباط بدون عامل بین پهپاد و اپراتور آن ایجاد می‌کنیم. روش‌های مبتنی بر بینایی ماشین بر توانایی دوربین هواپیماهای بدون سرنشین متکی هستند. بدین صورت که تصاویر اطراف را گرفته و با استفاده از ترجمه تصاویر و تشخیص الگوی دست، اطلاعات معناداری را استخراج می‌کنند. ساختار این پژوهه از دو مأذول اصلی تشکیل شده است: تشخیص حرکت دست<sup>۳</sup> و دستور به هواپیمای بدون سرنشین. برای مأذول اول از یک روش یادگیری عمیق<sup>۴</sup> استفاده شده است. الگوریتم‌ها و تکنیک‌های پردازش تصویر به عنوان روشی پویا برای شناسایی ژست‌ها و حرکات دست معرفی شده‌اند. مأذول دوم وظیفه ارتباط با پهپاد را بر عهده دارد. بدین صورت که پیام‌های بین سیستم پیشنهادی و پهپاد متصل به سیستم را ارسال و دریافت می‌کند و طبق آن پیام‌ها عملیات مورد نظر را اجرا می‌کند.

## واژه‌های کلیدی:

پهپاد، هواپیمای بدون سرنشین، ژست دست، بینایی ماشین، شبکه‌های عصبی پیچشی<sup>۵</sup>، حافظه طولانی

<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

<sup>6</sup>Long short-term memory

<sup>7</sup>Machine learning

<sup>8</sup>Human-drone interface

کوتاه مدت<sup>۶</sup>، یادگیری ماشین<sup>۷</sup>، رابط انسان و پهپاد<sup>۸</sup>

# فهرست مطالب

صفحه

عنوان

۱	.....	۱ مقدمه
۲	.....	۱-۱ مقدمه
۳	.....	۲-۱ چالش‌های استفاده از پهپاد
۳	.....	۳-۱ اهمیت استفاده از بینایی ماشین در پهپاد
۵	.....	۴-۱ تعریف مسئله
۵	.....	۱-۴-۱ چالش‌های اجرای پروژه
۵	.....	۱-۵ مراحل انجام پروژه
۶	.....	۶-۱ جمع‌بندی
۷	.....	۲ کارهای مشابه
۸	.....	۱-۲ مقدمه
۹	.....	۲-۲ مقالات مربوط به ویژگی‌های تصویر
۹	.. Hand Gesture Controlled Drones: An Open Source Library	۱-۲-۲ مقاله
۱۰	..... A real-time hand gesture recognition method	۲-۲-۲ مقاله
۱۲	..... مقالات مربوط به ورودی تصویر دست به مدل	۳-۲
۱۲	.... Hand Gestures For Drone Control Using Deep Learning	۱-۳-۲ مقاله
۱۳	UAV-GESTURE: A Dataset for UAV Control and Gesture Recognition	۲-۳-۲ مقاله
۱۴	..... مقالات مربوط به نقاط کلیدی دست	۴-۲
۱۵	. Hand Gesture Recognition system for Real-Time Application	۱-۴-۲ مقاله
	An improved hand gesture recognition system using keypoints	۲-۴-۲ مقاله
۱۶	..... and hand bounding boxes	
۱۷	..... Visual gesture recognition based on hand key points	۳-۴-۲ مقاله
۱۸	.... MediaPipe Hands: On-device Real-time Hand Tracking	۴-۴-۲ مقاله
۱۹	..... مقالات مربوط به اجرای مدل‌های بینایی کامپیوتر روی پهپاد	۵-۲
	Modeling relation among implementing AI-based drones and	۱-۵-۲ مقاله
۱۹	..... sustainable construction project success	

۲۰	Field of Control Engineering	۲-۵-۲
۲۱	جمع‌بندی	۶-۲
۲۲	روش انجام پروژه	۳
۲۳	مقدمه	۱-۳
۲۴	انتخاب ژست‌های دست متناسب با حرکت پهپاد	۲-۳
۲۵	دیتاست	۳-۳
۲۵	اهمیت ژست دست	۴-۳
۲۶	کنترل پهپاد	۵-۳
۲۶	ابزارها و نرم افزارهای مورد استفاده	۶-۳
۲۷	زبان برنامه‌نویسی پایتون	۱-۶-۳
۲۷	کتابخانه Scikit-learn	۲-۶-۳
۲۸	رابط برنامه‌نویسی Keras	۳-۶-۳
۲۸	کتابخانه‌های MediaPipe	۴-۶-۳
۲۹	کتابخانه NumPy	۵-۶-۳
۲۹	کتابخانه Matplotlib	۶-۶-۳
۳۰	کتابخانه OpenCV	۷-۶-۳
۳۰	DJI Tello پهپاد	۸-۶-۳
۳۱	مدیاپایپ	۷-۳
۳۲	محدود کننده جریان زمان واقعی <sup>۹</sup>	۱-۷-۳
۳۳	تشخیص دست <sup>۱۰</sup>	۲-۷-۳
۳۵	تشخیص به مستطیل <sup>۱۱</sup>	۳-۷-۳
۳۵	برش تصویر <sup>۱۲</sup>	۴-۷-۳

<sup>9</sup>Real Time Flow Limiter

<sup>10</sup>Hand Detection

<sup>11</sup>Detection To Rectangle

<sup>12</sup>Image Cropping

<sup>13</sup>Hand Landmark

<sup>14</sup>Landmarks To Rectangle

۳۶	۵-۷-۳ تشخیص نقاط عطف دست <sup>۱۳</sup>
۳۶	۶-۷-۳ نقاط عطف یه مستطیل <sup>۱۴</sup>
۳۷	۷-۷-۳ ارائه کننده حاشیه نویسی <sup>۱۵</sup>
۳۸	۸-۳ پیشپردازش
۳۹	۹-۳ پسپردازش
۳۹	۱۰-۳ جمعبندی
۴۰	<b>۴ نتایج و ارزیابی</b>
۴۱	۱-۴ مقدمه
۴۱	۲-۴ ارزیابی عملکرد مدلها
۴۱	۱-۲-۴ دقت
۴۲	۲-۲-۴ صحت
۴۳	۳-۲-۴ فراخوانی
۴۳	۴-۲-۴ امتیاز F1
۴۳	۵-۲-۴ گزارش معیارهای ارزیابی در مدلها
۴۳	۳-۴ نمودارهای دقت و خطای بر حسب دوره
۴۵	۴-۴ سرعت اجرای برنامه
۴۵	۵-۴ سختافزار مورد نیاز
۴۶	۶-۴ مقایسه دقت پروژه ما با کارهای مشابه
۴۶	۷-۴ جمعبندی
۴۸	<b>۵ نتیجه گیری و پیشنهادات</b>
۴۹	۱-۵ مقدمه
۴۹	۲-۵ محدودیت وجود پهپادهای منبع باز
۵۰	۳-۵ وجود چندین دست در تصویر
۵۰	۴-۵ شناسایی یک دست در دو مستطیل در نتیجه دو خروجی دست
۵۱	۵-۵ اجرا روی کارت گرافیکی <sup>۱۶</sup>

<sup>15</sup> AnnotationRenderer

<sup>16</sup> GPU

۵۲	۶-۵ اجرای پروژه با برنامه C++ . . . . .
۵۳	۷-۵ استفاده از مدل‌های موازی برای تشخیص ژست دست . . . . .
۵۳	۸-۵ توسعه برنامه با استفاده از سایر قسمت‌های بدن انسان . . . . .
۵۳	۱-۸-۵ کنترل پهپاد با حرکات چشم . . . . .
۵۴	۲-۸-۵ کنترل پهپاد با حرکات سر . . . . .
۵۴	۹-۵ جمع‌بندی . . . . .
۵۵	کتاب‌نامه . . . . .

# فهرست تصاویر

صفحه

شکل

۱۰	۱-۲ چارچوب کنترل پهپاد مبتنی بر ژست . . . . .
۱۰	۲-۲ ویژگی های Haar برای استفاده از آستانه رنگ پوست برای تشخیص دست . . . . .
۱۱	۳-۲ تشخیص کف دست و انگشتان . . . . .
۱۳	۴-۲ معماری VGG-16 . . . . .
۱۵	۵-۲ تشخیص نقطه کلید و تطبیق توسط SIFT . . . . .
۱۶	۶-۲ معماری ساختار شبکه های عصبی دو خط لوله . . . . .
۱۸	۷-۲ ورودی ها و خروجی های مدل تشخیص نقطه کلید دستی . . . . .
۲۱	۸-۲ ارتباط با پهپاد DJI Tello . . . . .
۲۳	۱-۳ تشخیص ژست دست با کمک نقاط کلیدی دست . . . . .
۲۴	۲-۳ نمونه ای از ژست های انتخاب شده در مجموعه داده ها . . . . .
۲۹	۳-۳ برخی کاربردهای کتابخانه MediaPipe . . . . .
۳۱	۴-۳ اطلاعات پهپاد DJI Tello . . . . .
۳۲	۵-۳ نمودار MediaPipe برای شناسایی نقاط کلیدی دست . . . . .
۳۴	۶-۳ خط لوله تشخیص دست . . . . .
۳۴	۷-۳ معماری مدل آشکارساز کف دست . . . . .
۳۵	۸-۳ پیدا کردن مستطیل حاوی دست . . . . .
۳۷	۹-۳ معماری مدل نقطه عطف دست. این مدل دارای سه خروجی است که یک استخراج کننده ویژگی را به اشتراک می گذاردند. هر سر توسط مجموعه داده های مربوطه که با همان رنگ مشخص شده اند آموزش داده می شود. . . . .
۳۷	۱۰-۳ موقعیت ۲۱ نقطه کلیدی در ناحیه دست . . . . .
۴۴	۱-۴ معیارهای ارزیابی برای تشخیص ژست دست در مدل MLP . . . . .
۴۴	۲-۴ منحنی های کم برازش، بهینه و بیش برازش . . . . .
۴۵	۳-۴ نمودار روند دقیق و خطاب بر حسب دوره در داده های آموزش و تست در مدل CNN . . . . .
۵۱	۱-۵ شناسایی یک دست دو مرتبه . . . . .

## فهرست جداول

صفحه

جدول

۴۲	۱-۴	جدول ارزیابی دقیق مدل‌ها
۴۵	۲-۴	جدول ارزیابی زمان پاسخگویی مدل‌ها
۴۶	۳-۴	جدول ارزیابی سخت‌افزار موردنیاز مدل‌ها
۴۷	۴-۴	جدول مقایسه پروژه با کارهای مشابه

# 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 پرندۀ‌ای دیده می‌شوند که عملکردهای متعددی مانند جمع آوری داده‌ها و سنجش از 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 پرندۀ در مشاغل مختلف مانند سیستم‌های تحویل بسته استفاده می‌شوند [۳]. به عنوان مثال، شرکت‌هایی مانند آمازون و UPS از پهپادهای چند روتور برای تحویل بسته‌های خود استفاده می‌کنند [۴]. در پی این موضوع، بسیاری از شرکت‌های تولید کننده پهپاد تشویق شدند 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. هزینه کم، قدرت کم، اندازه کوچک ضبط تصویر، و دستگاه‌های پخش جریان، آنها را به پهپادهای کاربردی و متعدد در بازار تبدیل می‌کند [۵]. در ادامه زمینه مطالعاتی جدیدی به نام رابط هواپیماهای بدون سرنشین و انسان<sup>۱</sup> گشوده شد تا تعامل بین پهپاد و انسان را پیشرفت دهد، که این تعامل مجموعه دستگاه‌های سنتی مانند کنترل رادیویی<sup>۲</sup> تا کنترل پهپادها با استفاده از وضعیت بدن و دست انسان را شامل می‌شود [۶].

یکی از رویکردهای مورد استفاده برای افزایش کاربرد و دسترسی به پهپادها، استفاده از بینایی ماشین است. این ویژگی معمولاً از طریق پردازش تصویر و با استفاده از شبکه‌های عصبی به کار می‌رود. 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، پست و این چنین موارد را انجام دهد [۷]. می‌توان گفت هدف اصلی استفاده از بینایی ماشین در پهپادها برای به حداقل رساندن دخالت انسان به صورت مستقیم است. این

<sup>1</sup>Human drone interface

<sup>2</sup>Radio Controller

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 دهند [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 مدتی نیاز به شارژ مجدد دارند. همچنین، محدودیت‌های محیطی نیز می‌تواند به چالش‌هایی بدل شوند؛ زیرا پهپادها به شرایط محیطی مانند آب و هوا، یا وزن و ارتفاع حساس هستند و این موارد می‌توانند در طراحی آنها تأثیر بهسزایی داشته باشد. در ادامه باید به میزان اهمیت امنیت اطلاعات هم اشاره کرد، زیرا پهپادها به دلیل استفاده از سیستم‌های موقعیت‌یاب و ارتباطات بی‌سیم ممکن است در برابر حملات سایبری آسیب‌پذیر باشند و اطلاعات مهمی که توسط آنها مخابره می‌شود، در معرض خطر قرار گیرد.

همچنین می‌توان به برخی چالش‌هایی که ما هم در این پروژه به صورتی با آنها سر و کار داریم و در تلاشیم آنها را از بین ببریم یا کمتر کنیم اشاره کرد. مانند انتقال اطلاعات، زیرا برای ارتباط با پهپادها از شبکه‌های بی‌سیم استفاده می‌شود و در شرایطی مانند اشباع شبکه یا فاصله بین پهپاد و کنترل‌کننده، ممکن است این ارتباط دچار مشکل 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 استفاده از بینایی ماشین در پهپاد

طبق اعلام پیش‌بینی اداره هوانوردی فدرال، بازار هواپیماهای بدون سرنشین تا سال ۲۰۲۵ به ۱۷ میلیارد خواهد رسید و ۷ میلیون هواپیمای بدون سرنشین به آسمان پرواز خواهند کرد. پهپادهای کنترل از راه دور به تدریج به دستگاه‌های نیمه خودکار یا کاملاً خودکار تبدیل می‌شوند که از پیاده سازی 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 سرنشین در حال تبدیل شدن به یک روش محبوب برای تعامل است. این پایان نامه یک سیستم کامل برای کنترل هواپیماهای بدون سرنشین با استفاده از حرکات دست پیشنهاد می‌کند. سیستم پیشنهادی باید در زمان واقعی<sup>۳</sup> کار کند و دقت<sup>۴</sup> خوبی 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. بدین صورت 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 [۱۰].

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 فضا را بهبود می‌بخشد.

استفاده از علائم دست سبب کاهش نیاز به دستگاه‌های کنترل 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 از قبل بیان شده است، این روش خطرات مرتبط با اتصالات بی‌سیم بین کنترلر و پهپاد را کاهش می‌دهد و دقت در کنترل پهپاد در محیط‌های پرتلاطم و مختلف را افزایش می‌دهد.

<sup>3</sup>Real-time

<sup>4</sup>Accuracy

<sup>5</sup>Image Processing

<sup>6</sup>Deep Neural Network

<sup>7</sup>Convolutional Neural Network(CNN)

## ۴-۱ تعریف مسئله

هدف این پروژه کنترل کردن پهپاد با استفاده از پردازش تصویر<sup>۵</sup> در زمان واقعی است. برای پیاده‌سازی آن می‌توان از یک شبکه عصبی عمیق<sup>۶</sup>، مانند یک شبکه عصبی کانولوشن<sup>۷</sup>، استفاده کرد. دلیل استفاده از این معماری قابلیت استخراج خودکار ویژگی‌ها با توجه به الگوریتم طبقه‌بندی تصاویر<sup>۸</sup> است. عملکرد شبکه عصبی کانولوشنال به این گونه است که ویژگی‌ها را با توجه به لایه‌های پنهان می‌آموزد، همچنین می‌تواند تعداد پارامترها را بدون به خطر انداختن دقت مدل تغییر دهد. با گذشت زمان محققان معماری‌های مختلفی از شبکه عصبی کانولوشن را برای دقت<sup>۹</sup> بهتر، زمان پردازش کمتر و پیچیدگی‌های ۱۰ گوناگون مطرح کردند.

## ۱-۴-۱ چالش‌های اجرای پروژه

وجود سخت‌افزاری مناسب برای اجرای این پروژه الزامی است. پهپاد انتخاب شده در ابتدا باید شامل یک دوربین با رزوولوشن نسبتاً بالا (حداقل \*\*\* پیکسل باشد) تا ژست دست تا فاصله سه متری از پهپاد به وضوح گرفته شود. در ادامه از آنجایی که زمان واقعی در این پروژه از اهمیت بالایی برخوردار است پهپاد باید پردازنده نسبتاً قوی داشته باشد تا بتواند به صورت مستقل و بدون نیاز به هیچ‌گونه سخت افزار خارجی مدل را اجرا کند، بدین صورت که در هر لحظه ورودی عکس گرفته‌شده از دوربین را به مدل بدهد و در کمترین زمان ممکن بتواند خروجی مدل را به دست آورده و دستور مورد نظر را روی پهپاد به اجرا درآورد. از دیدگاهی دیگر، از آنجایی که اجرای یک مدل بینایی ماشین یک برنامه سنگین است و اجرای آن برای عموم پهپادها انرژی زیادی می‌طلبد، لذا باید پهپادی را انتخاب کرد که از شامل باطربادی بادوام و باکیفیت باشد که هم در هنگام اجرای مدل بتواند انرژی موردنیاز پردازنده را فراهم کند و همچنین عمر کوتاه آن به مرور زمان برای استفاده کننده آزاردهنده نباشد.

## ۱-۵ مراحل انجام پروژه

۱. انتخاب ژست‌های مناسب و مفهومی برای کنترل پهپاد

۲. انتخاب چگونگی دخیره کردن دیتای عکس‌ها در دیتابست

<sup>8</sup>Image Classification

<sup>9</sup>Accuracy

<sup>10</sup>Complexity

۳. پیاده‌سازی کد برای جمع‌آوری دیتاست

۴. پیاده‌سازی مدل‌ها متناسب با دیتا

۵. آموزش به مدل‌ها

۶. بهینه‌سازی مدل‌ها

۷. تست مدل‌ها و انتخاب بهترین مدل

۸. پیاده‌سازی رأی‌گیری پنجره‌ای<sup>۱۱</sup>

۹. اجرای مدل روی پهپاد

## ۶-۱ جمع‌بندی

هدف این پروژه پیاده‌سازی برنامه‌ای کاربردی بر روی پهپاد است تا بتواند ۹ ژست دست از پیش تعیین شده را شناسایی و با توجه به آنها دستور داده شده از طرف کاربر را به پهپاد بدهد. در این پروژه موارد زیر از اهمیت بالایی برخوردار هستند:

- دقت بالای مدل زیرا در صورت انجام نادرست دستورات امکان برآورد هزینه مالی وجود دارد.
- زمان واقعی زیرا اجرای دستورات باید در کمترین زمان ممکن رخ دهد تا مورد پسند کاربر باشد.
- مدلی سبک تا توانایی اجرا روی پهپاد را داشته باشد.

---

<sup>11</sup>Window voting

# فصل دوم

## کارهای مشابه

## ۱-۲ مقدمه

در این فصل هدف ما بررسی پروژه‌های مشابه است تا بتوان از آنها در روند پروژه کمک گرفت. همچنین در این راه می‌توان با توجه به نتایج و ارزیابی پروژه‌های دیگر بسترهای را فراهم کرد تا نتیجه پروژه را با دیگر کارهای مشابه مقایسه کرد.

به صورت کلی پروژه‌هایی با هدف کنترل پهپاد با ژست دست در ۲ 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 IMU که نیازمند سخت‌افزار خاص برای پیدا کردن موقعیت نقاط دست است. مانند پروژه‌های Motion Estimation and Hand [۱۰] Gesture Recognition-Based Human–UAV Interaction Approach in Real Time و Hand gesture recognition with convolutional neural networks for the multimodal UAV [۱۱] control.
- وجود دستگاه کنترل کننده حرکت جهشی Leap Motion Controller که با توجه آن ویژگی‌های دست با دقت بالا اندازه گیری شده و با کمک شبکه‌های عصبی ژست دست تشخیص داده می‌شود. پروژه‌ی [۱۲] Deep Learning Based Hand Gesture Recognition and UAV Flight Controls نمونه‌ای از این جمله پروژه‌ها و [۱۳] Gesture control of drone using a motion controller هستند.

از بین این موارد پروژه ما مربوط به اولین گزینه است که تنها سخت‌افزار مورد نیاز به جز پهپاد دوربین نصب شده روی پهپاد است. که به بررسی نمونه‌ی این پروژه‌ها می‌پردازیم. پروژه‌های مشابه با کار ما که با کمک بینایی ماشین پهپاد را کنترل می‌کنند به ۴ دسته کلی تفکیک می‌شوند.

۱. استخراج ویژگی‌های تصویر در هر فریم که با توجه به نیازهای مسئله می‌تواند متفاوت باشد.
۲. تشخیص دست<sup>۱</sup> برای پیدا کردن موقعیت دست در هر فریم تصویر و ورودی پیکسل‌های RGB آن به مدل و در نهایت کلاس‌بندی ژست دست.
۳. استخراج نقاط کلیدی<sup>۲</sup> تصویر و ورودی آنها به مدل برای کلاس‌بندی.

<sup>1</sup>Hand detection

<sup>2</sup>Key point

## ۲-۲ مقالات مربوط به ویژگی‌های تصویر

مقالات به کار برده شده در این قسمت بر چگونگی تعیین ژست دست با توجه به تصویر داده شده تمرکز دارند. برخی از این مقالات چگونگی ارتباط با پهپاد را نیز پوشش می‌دهند، اما نکته مهم در این مقالات چگونگی استخراج ویژگی‌های تصویر و استفاده از آنها برای تعیین ژست دست است.

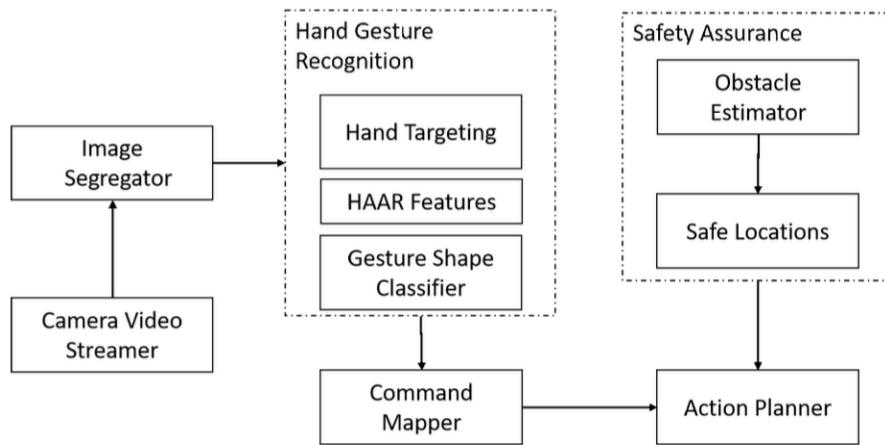
### ۱-۲-۲ مقاله Hand Gesture Controlled Drones: An Open Source Library

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 دست پویا برای کنترل پرواز پهپاد است. این تشخیص بر اساس ویژگی‌های Haar که با توجه به سایه‌ها و رنگ‌های درون تصویر تعیین می‌شوند، انجام می‌شود.

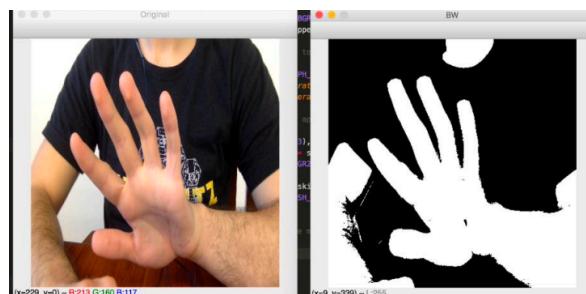
#### روش‌شناسی

در این پروژه، ابتدا از یک شبکه عصبی برای تشخیص موقعیت دست استفاده می‌شود که به عنوان یک ماژول پیش‌پردازش عمل می‌کند و با دقت بالایی موقعیت دست را تشخیص می‌دهد. پس از تشخیص موقعیت دست، ویژگی‌های Haar از تصویر استخراج می‌شوند که این ویژگی‌ها مجموعه‌ای از الگوریتم‌های تشخیص ویژگی هستند که از تصاویر استفاده می‌کنند تا ویژگی‌های خاصی از تصویر را شناسایی کنند. ویژگی‌های Haar 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 داده‌های ورودی، مثل ویژگی‌های Haar، می‌آموزند تا حرکات دست را تشخیص دهند و بر اساس آنها دستورات حرکت پهپاد را تعیین کنند. این سیستم شامل مراحل پیش‌پردازش داده‌ها، شناسایی شده به دستورات حرکت پهپاد می‌باشد. علاوه بر این، از مدل SVM (ماشین بردار پشتیبان) برای کلاس‌بندی و تشخیص حرکات دست استفاده شده است. SVM به عنوان یک ماشین یادگیری ماشینی است 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 ۲-۱ چارچوب کنترل پهپاد مبتنی بر ژست



شکل ۲-۲ ویژگی های Haar برای استفاده از آستانه رنگ پوست برای تشخیص دست

### نتیجه بدست آمده

این پروژه دقت بالایی در تشخیص ژست دست و کنترل پرواز پهپاد دارد. پنج حالت دست مدنظر در این پروژه قرار گرفته‌اند و دقت متوسط آن برابر با  $97.471$  درصد است که نشان‌دهنده عملکرد بسیار عالی است. اما لازم به ذکر است که این دقت در پس زمینه‌های بهم ریخته و همچنین در شرایط نوری مختلف بسیار متغیر است زیرا ویژگی Haar به سایه و رنگ‌های درون تصویر بسیار حساس است.

## ۲-۲-۲ مقاله A real-time hand gesture recognition method

در این مقاله در زمینه پردازش تصویر و تشخیص ژست‌های دست، از روش‌های مبتنی بر مدل ظاهری Appearance Model به عنوان یک رویکرد موثر استفاده شده است. این روش‌ها از ویژگی‌های تصویری

و حرکتی دست برای تشخیص و تعیین ژست‌های دست استفاده می‌کنند. در این مقاله، روش تشخیص ژست‌های دست به صورت زمان واقعی و قابل اعتماد است که از تشخیص دست، پیگیری دست<sup>۳</sup>، تقسیم‌بندی دست<sup>۴</sup> و تشخیص ویژگی‌های مقیاس-فضا<sup>۵</sup> برای تشخیص ژست‌های دست استفاده می‌کند.

### روش‌شناسی

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

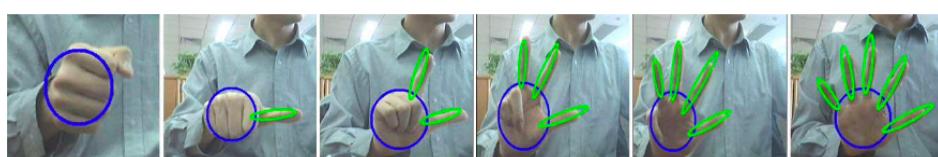
۱. استفاده از روش Adaboost برای تشخیص دست، که یک روش معتبر برای تشخیص اشیاء در تصاویر است.

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

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

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

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



شکل ۲-۳ تشخیص کف دست و انگشتان

<sup>3</sup>Hand tracking

<sup>4</sup>Hand segmentation

<sup>5</sup>Scale-space feature detection

### نتیجه

اعمال این روش‌ها نتایج قابل قبولی را به همراه داشته است. دقت مدل در تشخیص ژست‌های دست به صورت میانگین ۹۳.۸ درصد بوده و از جمله نتایج مهم آزمایشات می‌توان به تشخیص صحیح 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 ژست‌های دست دارد و می‌تواند به عنوان یک روش موثر برای تعاملات زمان واقعی استفاده شود. [۱۴]

## ۳-۲ مقالات مربوط به ورودی تصویر دست به مدل

مقالات این دسته از جمله پژوههایی هستند که بیشتر از آنکه بر روی پیش‌پردازش 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. تنها موردی که می‌توان در این نوع پروژه‌ها پیش رو گرفت 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. پس از آن باید توجه داشت معماری شبکه را به گونه‌ای برگزید تا مخصوص پردازش تصویر باشد و بتوان ویژگی‌های تصویر را خود استخراج کند.

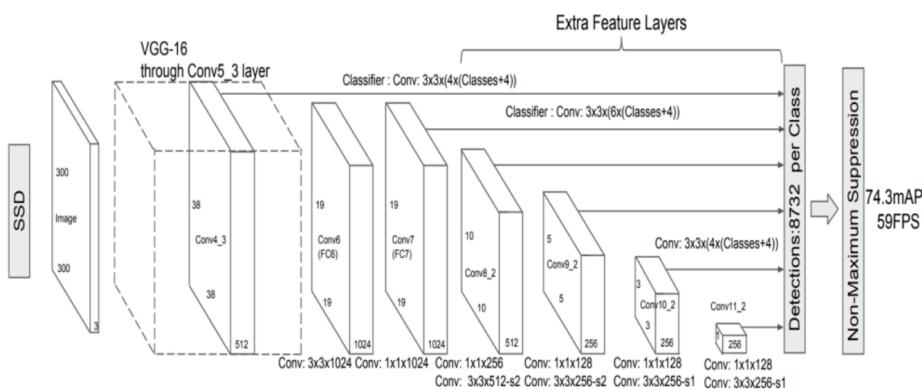
## ۱-۳-۲ مقاله Hand Gestures For Drone Control Using Deep Learning

این پروژه با هدف کنترل پهپادها با استفاده از حرکات دستی و با کمک شبکه‌های عمیق یادگیری انجام شده است تا بتوان ۹ حالت مختلف دست را شناسایی و به پهپاد، دستور موردنظر کاربر را داد.

### روش‌شناسی

در این تحقیق، از معماری شبکه عصبی عمیق VGG-16 برای تشخیص و تعیین حرکات دستی برای کنترل پهپادها استفاده شده است. شبکه VGG-16 یکی از معروف‌ترین و پرکاربردترین شبکه‌های عمیق در زمینه بینایی ماشین است که شامل ۱۶ لایه عصبی با لایه‌های کانولوشنال و پولینگ می‌باشد. این شبکه برای استخراج ویژگی‌های مهم از تصاویر استفاده می‌شود. ورودی این شبکه تصاویری است که از دوربین متصل به دستگاه اجرایی گرفته می‌شود و سپس این تصاویر به شبکه وارد می‌شوند. خروجی این شبکه شامل تشخیص حرکات دستی مانند حرکات مختلف انگشتان و دست‌ها می‌شود که سپس این

اطلاعات برای ارسال دستورات کنترلی به پهپاد استفاده می‌شود. این روش نه تنها امکان کنترل دقیق و موثر پهپادها را فراهم می‌کند بلکه ارتباط بین انسان و ماشین را نیز بهبود می‌بخشد.



شکل ۴-۲ معماری VGG-16

### نتیجه بدست آمده

در پروژه این پروژه 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 دقت بالایی برای کنترل پهپاد به حساب نمی‌آید. [۶]

## ۲-۳-۲ مقاله UAV-GESTURE: A Dataset for UAV Control and Gesture Recognition

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 تجربه کند.

### روش‌شناسی

در این مقاله، از شبکه P-CNN<sup>۶</sup> برای تشخیص حرکات دست استفاده شده است. این شبکه از اطلاعات حرکت و ظاهر را بر روی مسیرهای بخش‌های بدن انسان (مانند دست راست، دست چپ، بدن بالا و بدن کامل) جمع‌آوری می‌کند. این شبکه ابتدا موقعیت دست فرد را با استفاده از جعبه مرزی مشخص

<sup>6</sup>Pose-based Convolutional Neural Network

می‌کند و سپس تصویر دست با استفاده از فیلترهای مناسب وارد شبکه P-CNN می‌شود تا بتواند حرکت دست را پیش‌بینی کند.

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

اطلاعات ظاهر و حرکت را از بخش‌های مختلف بدن استخراج می‌کند و از دو شبکه پیش‌آموزش VGG-f استفاده برای محاسبه ویژگی‌های CNN استفاده می‌کند. برای بخش‌های ظاهر، از شبکه Action Tube استفاده می‌شود، در حالی که برای بخش‌های حرکتی از شبکه حرکتی از پیاده‌سازی استفاده می‌شود. ویژگی‌های استاتیک و پویا به طور جداگانه در طول زمان جمع‌آوری می‌شوند تا ویژگی‌های ویدیوی استاتیک و پویا به دست آید.

در نهایت، از روش‌های تجمعی مینیمم و ماکسیمم برای هر بعد از توصیف گر بر روی تمام ویدیوها استفاده می‌شود. این روش‌ها برای تشخیص حرکات با دقت بالا استفاده می‌شوند.

#### نتیجه

دارد، اما نیازمندی‌های پیچیده‌ای برای پیاده‌سازی واقعی دارد که از جمله موانعی است که باید در نظر گرفته شود. در نتیجه، این مقاله یک روش برای کنترل پهپاد با استفاده از حرکت دست پیاده‌سازی کرده است و از شبکه P-CNN برای تشخیص حرکات دست استفاده کرده است. نتایج نشان داده‌اند که این روش با دقت ۹۱.۹ درصد، قابلیت اجرا در پروژه‌های واقعی را دارد، اما نیازمندی‌های پیچیده‌ای برای پیاده‌سازی واقعی دارد که از جمله موانعی است که باید در نظر گرفته شود. [۱۵]

## ۴-۲ مقالات مربوط به نقاط کلیدی دست

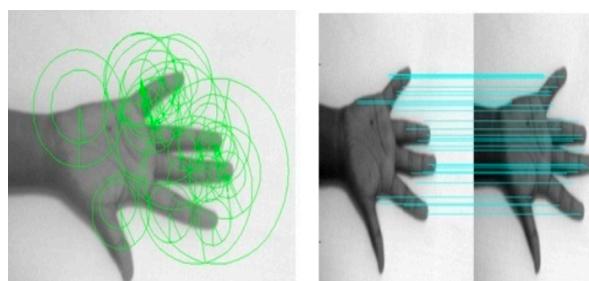
در این چنین مقالات ورودی شبکه بینایی ماشین برای تشخیص ژست، مختصات نقاط کلیدی دست هستند، که به نوعی یک ویژگی تصویر نیز تلقی می‌شوند. بدین صورت که در ابتدای کار دست کاربرد شناسایی شده و سپس نقاط کلیدی آن استخراج می‌شوند تا بتوان حجم داده ورودی به مدل را تا حد امکان ساده‌تر و در عین حال مفیدتر کرد.

## ۱-۴-۲ مقاله Hand Gesture Recognition system for Real-Time Application

این مقاله به بررسی سیستم تشخیص حرکات دست برای کاربردهای زمان واقعی می‌پردازد. در این سیستم، از الگوریتم SIFT برای استخراج ویژگی‌ها از تصاویر حرکتی استفاده شده و سپس از مدل Bag of Feature و ردهبند SVM برای تشخیص دقیق حرکات دست و دستیابی به عملکرد زمان واقعی استفاده شده است.

### روش‌شناسی

در مقاله اشاره شده، از الگوریتم SIFT برای استخراج نقاط کلیدی از تصاویر حرکتی دست استفاده شده است. الگوریتم SIFT یک الگوریتم معروف برای استخراج ویژگی‌های برجسته و تمایزدهنده از تصاویر است که از مقیاس، جهت و بخشی از تغییرات نوری مستقل برای استخراج این ویژگی‌ها استفاده می‌کند. این ویژگی‌ها به طور قابل توجهی مستقل از مقیاس و جهت تصویر هستند و می‌توانند برای تطبیق قابل اعتماد بین دیدگاه‌های مختلف یک شیء یا تصویر استفاده شوند.



شکل ۲-۵ تشخیص نقطه کلید و تطبیق توسط SIFT

### نتیجه

الگوریتم SIFT به عنوان یک ابزار قدرتمند برای استخراج ویژگی‌های برجسته و تمایزدهنده از تصاویر شناخته شده است و در این پژوهه با موفقیت برای تشخیص حرکات دست و دستیابی به دقت ۹۰.۸ درصد در تشخیص استفاده شده است.

[۱۶]

## ۲-۴-۲ مقاله An improved hand gesture recognition system using key-points and hand bounding boxes

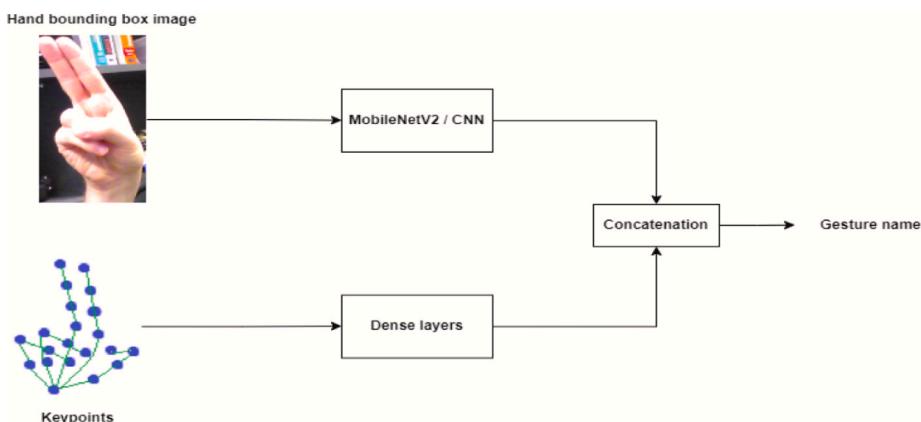
این مقاله یک سیستم بهبود یافته تشخیص حرکات دست با استفاده از نقاط کلیدی و جعبه‌های محدود کننده دست را معرفی می‌کند تا بتوان نقاط کلیدی دست را پیدا کرد.

### 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 ژست 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 متصل (FC) داده می‌شوند تا حرکات دست تشخیص داده شوند.

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



شکل ۲-۶ معماری ساختار شبکه‌های عصبی دو خط لوله

### نتیجه

در صورتی که دو مدل خروجی‌های متفاوتی را پیش‌بینی کنند، از روش‌های ترکیبی مانند ترکیب احتمالاتی یا استفاده از مدل‌های متفاوت برای شرایط ورودی مختلف استفاده می‌شود تا بهترین تصمیم برای تشخیص ژست دست گرفته شود.

دقت به دست آمده برای تشخیص ۶ ژست دست مختلف در این مقاله در سه دیتاست متفاوت به ترتیب برابر ۹۱، ۹۴ و ۹۶ درصد است.

[۱۷]

## ۳-۴-۲ مقاله Visual gesture recognition based on hand key points

این مقاله یک روش تشخیص حرکات ژستی بصری بر اساس نقاط کلید دست ارائه می‌دهد. این روش ابتدا نقاط کلید دست را در تصویر ورودی فعلی تشخیص می‌دهد و سپس حرکات تعریف شده را تشخیص می‌دهد.

### روش‌شناسی

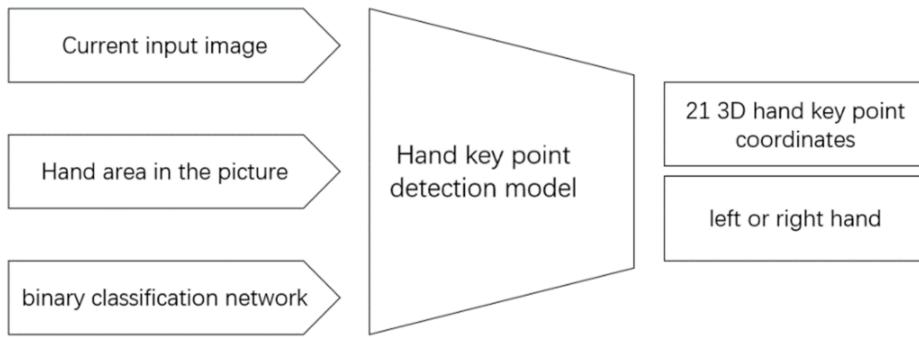
مدل ارائه شده در این مقاله شامل دو بخش اصلی است:

مدل تشخیص کف دست که مدل بر اساس ویژگی‌های سخت دست طراحی شده است. برای تشخیص حضور دست در تصویر، از یک مدل SSD استفاده شده است که به صورت زمان واقعی تشخیص انجام می‌دهد. در صورت وجود دست، نتایج توسط یک مستطیل تشخیص می‌شوند.

مدل تشخیص نقاط کلید دست که پس از تشخیص حضور دست در تصویر، از این مدل برای تشخیص ۲۱ مختصات نقطه کلید سه‌بعدی دست استفاده می‌شود. این مدل از الگوریتم نرم‌افزاری برای محاسبه مختصات افقی و عمودی هر نقطه دست استفاده می‌کند. 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 کلید تعیین می‌شود.

### نتیجه

این روش دارای دقیقیت بالا و عملکرد مناسبی است. دقیقیت متوسط مدل برابر ۸۵.۴ درصد تا ۹۸.۵ درصد است که با توجه به جزئیات مدل و پیاده‌سازی پیش‌پردازش می‌تواند انعطاف بالایی داشته باشد پس



شکل ۷-۲ ورودی ها و خروجی های مدل تشخیص نقطه کلید دستی

می تواند راهکار مناسبی تلقی شود.  
.[۱۸]

## ۴-۴-۲ مقاله MediaPipe Hands: On-device Real-time Hand Tracking

در این مقاله، از کتابخانه MediaPipe برای پیش‌بینی ۲۱ نقطه عطف دست و استفاده در پروژه‌های مختلف از جمله تشخیص ژست دست و افکتهای AR استفاده شده است. ما نیز در پروژه خود از این کتابخانه استفاده می‌کنیم تا یک مدل سبک و ساده پیاده‌سازی کنیم.

### روش‌شناسی

در این مقاله، از کتابخانه MediaPipe برای پیش‌بینی ژست دست استفاده شده است. برای پیاده‌سازی این پروژه، از دو شبکه کانولوشن 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 موقعیت‌یابی 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 ژست دست در پروژه‌های AR/VR و کنترل حرکات دست استفاده شود.

### نتیجه

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

[۱۹]

## ۵-۲ مقالات مربوط به اجرای مدل‌های بینایی کامپیوتر روی

### پهپاد

#### ۱-۵-۲ مقاله Modeling relation among implementing AI-based drones

#### and sustainable construction project success

این مقاله با به بررسی ارتباط بین استفاده از پهپادهای مبتنی بر هوش مصنوعی می‌پردازد. این تحقیق به بررسی تأثیرات مثبتی که ادغام این دو عنصر می‌تواند در بهبود کارایی و پایداری پروژه‌های ساختمانی داشته باشد، می‌پردازد. از آنجایی که استفاده از پهپادها در صنعت ساختمان به سرعت در حال افزایش است و شرکت‌های ساختمانی تحت فشار روزافزونی برای اتخاذ تکنیک‌های بیشتر پایدار و کارآمد قرار دارند، شناخت دقیق‌تری از ارتباط بین موانع و عوامل موفقیت لازم است تا بهترین روش‌ها برای از بین بردن موانع و تضمین موفقیت پهپادها در صنعت مشخص شود.

#### روش‌شناسی

در این تحقیق، از یک مدل معادلات ساختاری هوشمند برای بررسی رابطه بین موانع اجرای پهپادهای مبتنی بر هوش مصنوعی در صنعت ساختمان و موفقیت آنها 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.
- محدودیت‌های سخت‌افزاری: پهپادها ممکن است دارای محدودیت‌های سخت‌افزاری مانند ظرفیت پردازشی و حافظه باشند که ممکن است موانعی برای پیاده‌سازی مدل‌های پیچیده ایجاد کنند.
- 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 ساختمان است. با غلبه بر این چالش‌ها، می‌توان بهبود قابل توجهی در کارایی، ایمنی و پایداری پروژه‌های ساختمانی داشت و از پتانسیل بالقوه این تکنولوژی بهره‌مند شد.

### نتیجه

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.

[۲۰]

## ۲-۵-۲ مقاله Use of a DJI Tello Drone as an Educational Platform in the Field of Control Engineering

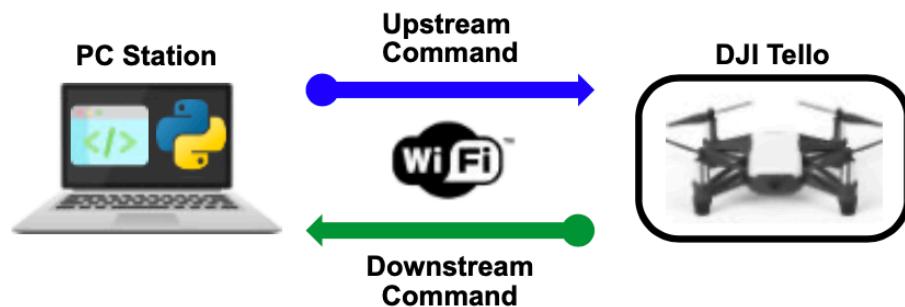
این مقاله یک رویکرد نوآورانه را برای استفاده از پهپاد DJI Tello به عنوان یک پلتفرم آموزشی در زمینه مهندسی کنترل ارائه می‌دهد. در این مقاله به بررسی نحوه استفاده از ویژگی‌های پهپاد برای آموزش مفاهیم کنترل به صورت عملی و جذاب می‌پردازند. همچنین به چگونگی استفاده از پهپاد DJI Tello به عنوان یک پلتفرم آموزشی برای آموزش مفاهیم کنترل می‌پردازد.

### روش‌شناسی

در این مقاله، از پهپاد DJI Tello به عنوان یک ابزار آموزشی و تحقیقاتی استفاده شده است. این پهپاد به دلیل داشتن حسگرهای متعدد و امکان برنامه‌نویسی با زبان پایتون، به عنوان یک ابزار ایده‌آل و انعطاف‌پذیر برای اهداف آموزشی و تحقیقاتی شناخته می‌شود. مقاله به بررسی ارتباط با پهپاد DJI Tello از طریق زبان برنامه‌نویسی پایتون، امکانات SDK رسمی ارائه شده توسط پهپاد، و نحوه ارتباط با پهپاد از طریق وای‌فای و پورت UDP می‌پردازد.

در این مقاله، انتخاب پهپاد DJI Tello برای نمایش مفاهیم ابتدایی مربوط به حوزه کنترل به دلایل

گوناگونی انجام شده است. از جمله دلایل انتخاب پهپاد می‌توان به محبوبیت رو به افزایش آن در بین عموم مردم، ویژگی‌های چندگانه آن شامل حسگرها، کنترل بازخورد و الگوریتم‌های تخمین برای انجام وظایف پیچیده، و همچنین قیمت مقرن به صرفه آن نسبت به سایر تجهیزات آموزشی اشاره کرد. برای پیاده‌سازی مدل‌های هوش مصنوعی بر روی پهپاد DJI Tello، از زبان برنامه‌نویسی پایتون و کتابخانه‌های مختلفی که برای ارتباط با حسگرها و سیستم‌های کنترل پروازی پهپاد ارائه شده استفاده می‌شود. این امکانات به دانشجویان این امکان را می‌دهد که مدل‌های هوش مصنوعی مختلفی را بر روی پهپاد پیاده‌سازی کرده و عملکرد آن‌ها را تحلیل کنند. به عنوان مثال، کاربران می‌توانند از شبکه‌های عصبی یا منطق فازی برای کنترل حرکت پهپاد استفاده کنند.



شکل ۸-۲ ارتباط با پهپاد DJI Tello

## نتیجه

نتیجه این مقاله نشان می‌دهد که استفاده از هوش مصنوعی بر روی پهپاد DJI Tello به خوبی عمل می‌کند 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. [۲۱]

## ۶-۲ جمع‌بندی

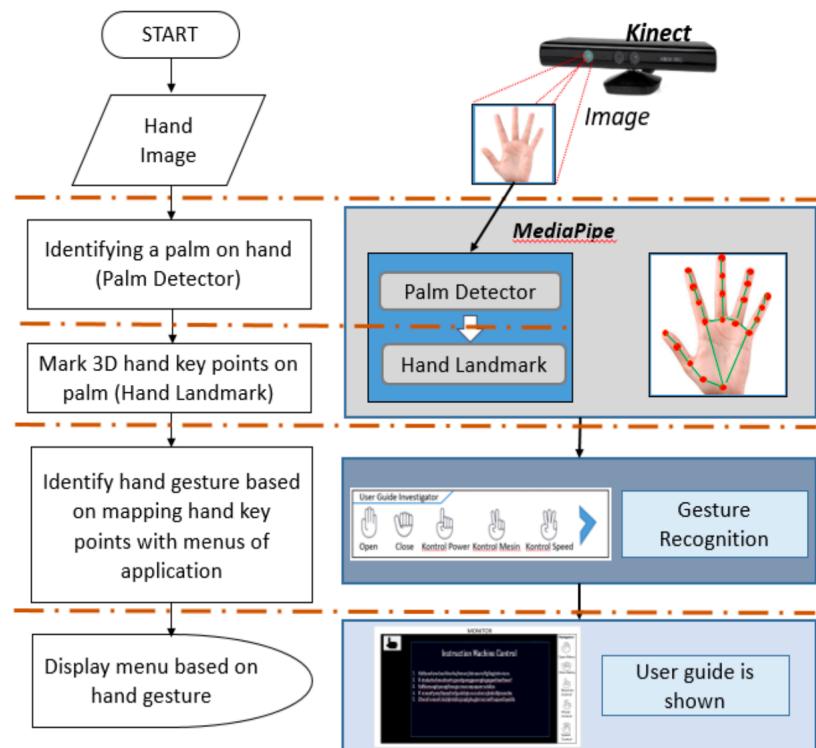
در تمام مقالات بررسی شده، پروژه‌ها به گونه‌ای پیاده‌سازی شده‌اند تا حرکات را به طوری کلاسیکی کنند که در خروجی حتماً یکی از ژست‌های درنظر گرفته شده انتخاب شود. لذا زمانی که دست در حالتی غیر از آنها قرار دارد، مدل طراحی شده حتماً یکی از ژست‌هایی را که به آن شبیه‌تر است را انتخاب می‌کند که این امر می‌تواند برای پیاده‌سازی روی پهپاد واقعی مشکل‌زا باشد و حتی هزینه مالی به ارمغان آورد. استخراج ویژگی‌های تصویر و یا ورودی خود تصویر به مدل می‌تواند بهینه عمل نکند.

# فصل سوم

## روش انجام پروژه

### ۱-۳ مقدمه

برای اجرای این پروژه راههای متفاوتی مورد بررسی قرار داده شد تا بتوان بهترین آنها را روی پهپاد پیاده‌سازی کرد. نتیجه نهایی پیاده سازی ۳ شبکه کانولوشن به صورت پی در پی است. شبکه اول برای آشکارسازی موقعیت کف دست است. بدین صورت که هر فریم گرفته شده از دوربین پهپاد پس از تغییر اندازه به یک ماتریس  $3 \times 256 \times 256$ \* $3 \times 3$ \* $3$  به ورودی مدل داده می‌شود و پس از پردازش آن خروجی یک ماتریس به عنوان ورودی داده می‌شود، در این مدل جعبه مرزی برش خورده دست به صورت یک ماتریس گرفته شده و خروجی آن برابر ۲۱ است که بعدی عطف دست و شاخص دست (راست یا چپ) است. در ادامه مدل سوم به عنوان ورودی، یک ماتریس  $21 \times 21$  می‌گیرد که مختصات نقاط طول و عرض هر نقطه عطف دست است س= چرا که عمق تصویر با توجه به ژستهای در نظر گرفته شده از اهمیت بالایی برخوردار نیست. و در خروجی پیش‌بینی می‌کند که کدام ژست دست مدنظر کاربر است. با توجه به این پروژه ۹ ژست گوناگون مدنظر قرار گرفته شده (کاربر می‌تواند ژستهای جدیدی اضافه کند)، خروجی شبکه کانولوشن شامل ۱۰ کلاس کلاس است که ۹ کلاس برای ژست‌ها و کلاسی برای زمانی که هیچ کدام از ژست‌ها انتخاب نشده در نظر گرفته شده.



شکل ۱-۳ تشخیص ژست دست با کمک نقاط کلیدی دست

## ۲-۳ انتخاب ژست‌های دست متناسب با حرکت پهپاد

انتخاب ژست‌های مناسب برای هر یک از حرکات پهپاد از اهمیت ویژه‌ای برخوردار است. چرا که ژست‌هایی که از لحاظ مفهومی به عملکرد پهپاد شبیه هستند راحت‌تر به خاطر سپرده شده و تجربه دلپذیرتری را در کاربر به وجود می‌آورند. ما برای این پروژه ۹ ژست دست را در نظر گرفته‌ایم تا بتوان حرکات پایه پهپاد را با آنها انجام داد. این حرکات شامل: حرکت رو به جلو، حرکت رو به عقب، حرکت به پایین، حرکت به بالا، حرکت به راست، حرکت به چپ، فرود آمدن، ایستادن در موقعیت کنونی و گرفتن عکس است که نمونه ژست دست آنها با توجه به حرکت پهپاد نشان داده شده است.



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

### ۳-۳ دیتاست

برای جمع آوری دیتاست مناسب پژوهه، از آنجایی که ژست‌های دست را بر اساس عملکرد پهپاد تعیین کردیم تا استفاده از آنها برای کاربر مورد پسند باشند، لذا پیدا کردن دیتاست آماده ناممکن است. جمع آوری دیتاست مناسب و جامع از اهمیت بالایی برخوردار است زیرا از آنجایی که مدل باید در زمان واقعی کار کند نمی‌توان از لایه‌های زیادی استفاده کرد. پس باید برای بالا بردن دقت مدل از موارد دیگری کمک گرفت، از جمله آنها می‌توان به پیش‌پردازش، پس‌پردازش و دیتاست مناسب اشاره کرد. از آنجایی که رویکردهای متفاوتی برای این پژوهه پیاده‌سازی شد تا بتوان به بهترین راهکار برای مدلی سبک و در عین حال جامع رسید، دیتاست‌های گوناگونی جمع آوری شد که هر یک از آنها ویژگی خاصی از تصویر را به عنوان دیتای مرود نیاز جمع آوری می‌کردند. از جمله ورودی کل عکس به صورت پیکسل‌های رنگی، پیدا کردن دست و ذخیره موقعیت آن به صورت پیکسل‌های  $256 \times 256$ ، پیدا کردن نقاط کلیدی دست و ذخیره موقعیت آنها در دیتاست و استفاده از آنها برای ورودی به مدل بهترین گزینه ممکن بود مه برای اجرای پژوهه از آن استفاده کردیم.

### ۴-۳ اهمیت ژست دست

وقتی مردم صحبت می‌کنند، ژست می‌گیرند. ژست جزء اساسی زبان است که اطلاعات معنادار و منحصر به فردی را انتقال می‌دهد. ژست‌ها به گوینده کمک می‌کنند تا اهداف خود را بهتر منعکس کند. آن‌ها نقش‌های بسیاری را در ارتباط، یادگیری و درک هم برای افرادی که آنها را مشاهده می‌کنند و هم برای کسانی که آنها را ایجاد می‌کنند، ایفا می‌کنند. وقتی مردم صحبت می‌کنند، دستان خود را حرکت می‌دهند. به حرکات خود به خودی دست که در ریتم گفتار ایجاد می‌شوند، حرکات هم گفتاری<sup>۱</sup> نامیده می‌شوند و مردم از همه فرهنگ‌ها و پیشینه‌های زبانی شناخته شده ژست می‌گیرند و برای ارتباط از حرکات هم گفتاری برای رساندن بهتر مفهوم خود کمک می‌گیرند. در واقع، نوزادان قبل از اینکه اولین کلمات خود را بیان کنند، از انواع ژست‌ها استفاده می‌کنند. دست‌های ما به ما کمک می‌کنند صحبت کنیم، فکر کنیم، و به خاطر بسپاریم، گاهی دانش منحصر به فردی را که هنوز نمی‌توان به زبان آورد، آشکار می‌کنند. به طوری که می‌توان گفت ژست‌ها اغلب به عنوان زبان گفتاری ثانویه در نظر گرفته می‌شود.<sup>[۲۲]</sup> ژست‌ها به ویژه زمانی مؤثر هستند که مزیتی نسبت به کلمات داشته باشند.<sup>[۲۳]</sup> توانایی

<sup>1</sup>co-speech gestures

<sup>2</sup>user experience

در ک شکل و حرکت دستها می تواند یک جزء حیاتی در بهبود تجربه کاربر<sup>۲</sup> در حوزه ها و پلتفرم های مختلف فناوری باشد. در ک مفهوم ژست دست در زمان واقعی برای افراد به طور طبیعی وجود دارد، یک کار بینایی کامپیوترا کاملاً چالش برانگیز است، زیرا دست ها اغلب خود یا یکدیگر را مسدود می کنند مانند انسداد انگشت، کف دست و لرزش دست و فاقد الگوهای کنتراست بالا هستند.<sup>[۱۹]</sup>

### ۵-۳ کنترل پهپاد

اکثر پهپادهای تجاری موجود در بازار یا دارای کنترلهای طراحی شده ویژه هستند، یا دارای فرستنده سیگنال اختصاصی و برنامه های نرم افزاری هستند که روی دستگاه های دستی کاربران مانند تلفن های همراه یا تبلت ها اجرا می شوند. در هر دو مورد، کنترل کننده فرمان هایی را با اطلاعات دقیق از طریق کانال های بی سیم مانند وای فای یا بلوتوث ارسال می کند. اخیراً محصولات تجاری وجود داشته است که حرکات دست را به عنوان یک مکانیسم کنترل قابل اجرا معرفی می کنند. برای گرفتن ژست ها، دو رویکرد وجود دارد.

- استفاده از دستکش های طراحی شده ویژه: کنترل کننده بر روی دستکشی که توسط کاربران استفاده می شود نصب می شود و در زمان واقعی انحراف، گام و چرخش دست را شناسایی می کند تا به حرکات مربوطه برای پهپاد را شناسایی و ارسال کند. محصولات عبارتند از Kd Interactive MenKind Motion Control Drone و Aura Drone

- استفاده از بینایی کامپیوترا از طریق دوربین: این دستگاهها از دوربین نصب شده روی پهپاد استفاده می کنند تا بتوانند در لحظه تشخیص دهنند که دست کاربر کجاست و در چه حالتی قرار دارد تا پهپاد را کنترل کند. محصولات عبارتند از DJI Spark Drone

### ۶-۳ ابزار ها و نرم افزار های مورد استفاده

برای پیاده سازی این پروژه از ابزارها، نرم افزارها و گتابخانه های گوناگونی استفاده شده است که در ادامه به توضیح دقیق آنها می پردازیم. قابل ذکر است که از گتابخانه هایی از جمله csv، copy، os و itertools نیز در قسمت هایی از پروژه به کار برده شده که به دلیل استفاده جزئی توضیح داده نشده.

### ۱-۶-۳ زبان برنامه‌نویسی پایتون

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.

انعطاف‌پذیری بالا دارد. می‌تواند برای انواع مختلف مدل‌های یادگیری ماشین از جمله شبکه‌های عصبی پیچشی<sup>۳</sup>، شبکه‌های عصبی بازگشتی<sup>۴</sup> و شبکه‌های مولد متخصص<sup>۵</sup> مورد استفاده قرار گیرد. TensorFlow قابلیت اجرای مدل‌ها بر روی پردازنده‌های چندگانه، پردازنده‌های گرافیکی<sup>۶</sup> و حتی TPU‌های خاص گوگل را دارد. همچنین به دلیل محبوبیت و پشتیبانی گسترده، منابع آموزشی و کتابخانه‌های جانبی فراوانی برای آن وجود دارد. به طور کلی، TensorFlow یکی از ابزارهای قدرتمند و پرکاربرد در حوزه یادگیری ماشین و هوش مصنوعی است.

### ۲-۶-۳ کتابخانه Scikit-learn

که با نام‌های Scikit-learn و scikits.learn 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، تقویت گرادیان، SciPy و NumPy و DBSCAN می‌باشد. Scikit-learn به طور ویژه برای تعامل با کتابخانه‌های k-means طراحی شده است و ابزارهای متنوعی برای پیش‌پردازش داده‌ها، انتخاب و ارزیابی مدل‌ها و کاهش ابعاد فراهم می‌کند. این کتابخانه به کاربران کمک می‌کند تا به راحتی از آن در پروژه‌های یادگیری ماشین خود 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.

<sup>3</sup>Convolutional neural network

<sup>4</sup>Recurrent neural network

<sup>5</sup>Generative adversarial network

<sup>6</sup>GPU

<sup>7</sup>API

### ۳-۶-۳ رابط برنامه‌نویسی Keras

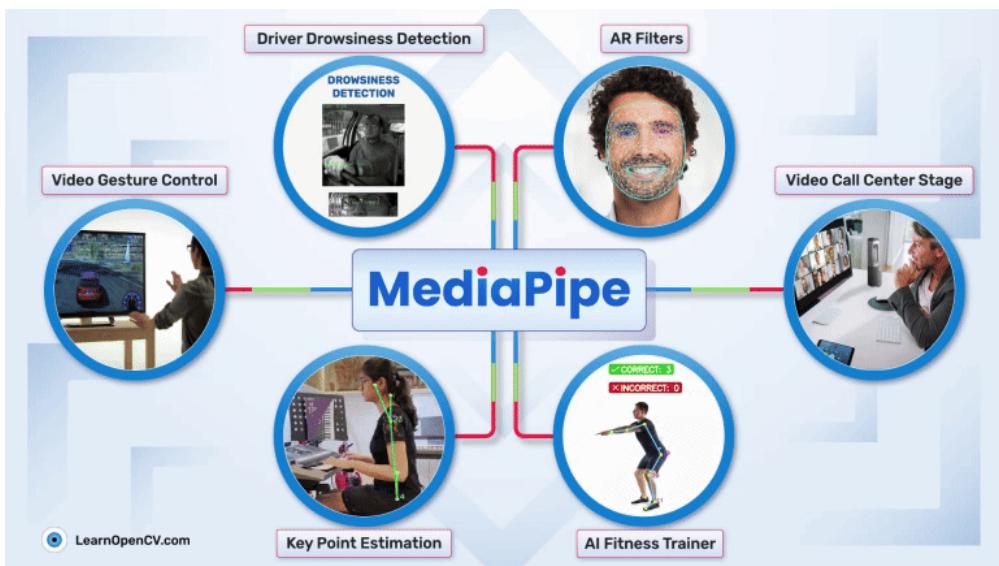
Keras یک رابط برنامه‌نویسی<sup>۷</sup> یادگیری عمیق است که به زبان پایتون نوشته شده و می‌تواند بر روی JAX و PyTorch و TensorFlow اجرا شود. هدف اصلی Keras کاهش پیچیدگی‌ها و بار شناختی توسعه دهنده‌گان است، به طوری که آنها بتوانند روی بخش‌های حیاتی و مهم پروژه‌های یادگیری ماشین تمرکز کنند. این رابط برنامه‌نویسی با رابط کاربری ساده و کاربرپسند، امکان توسعه سریع مدل‌های پیچیده را فراهم می‌کند. Keras همچنین 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. این کتابخانه با مستندات جامع و پشتیبانی از جامعه کاربری بزرگ، به یکی از ابزارهای محبوب در حوزه یادگیری عمیق تبدیل شده است.

### ۴-۶-۳ کتابخانه‌های MediaPipe

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 توسعه‌دهنده‌گان نرم‌افزار، که برنامه‌های کاربردی یادگیری ماشین را پیاده‌سازی می‌کنند، نمونه‌های اولیه فناوری را طراحی می‌کند تا بتوان پروژه‌ها را تا حد امکان ساده کرد. برنامه‌هایی که داده‌های حسی مثل ویدیو و صدا را با نرخ فریم بالا پردازش می‌کند تا تجربه کاربر را بهتر کند. مراحل پردازش یا مدل‌های استنتاجی ممکن است دشوار باشد، چون گاهی اتصال بین مراحل زیاد است. همچنین، توسعه برنامه برای پلتفرم زمانبر است. [۲۴]

Aین چالش‌ها را با انتزاع و اتصال مدل‌های مختلف به یک چارچوب مناسب حل می‌کند. با استفاده از MediaPipe، می‌توان یک لوله پردازش را به صورت گراف از اجزای مختلف، از جمله مدل‌های استنتاجی و عملکردهای پردازش رسانه‌ای، ساخت. همچنین این کتابخانه می‌تواند مطابق با نیازهای افراد خود سفارشی شود و در پلتفرم‌های مختلف توسعه پیدا کند [۲۵].

در مجموعه MediaPipe نیز از کتابخانه‌های مختلفی برای پیاده‌سازی برنامه‌ها استفاده می‌شود. از جمله آنها می‌توان به TensorFlow، PyTorch، CNTK، OpenCV و MXNet اشاره کرد. [۲۵]



شکل ۳-۳ برخی کاربردهای کتابخانه MediaPipe

### ۵-۶-۳ کتابخانه NumPy

کتابخانه‌ای برای محاسبات علمی در پایتون است که آرایه‌های چندبعدی و توابعی برای عملیات سریع روی آرایه‌ها ارائه می‌دهد. در هسته‌ی NumPy، شیء ndarray وجود دارد که آرایه‌های n-بعدی با نوع داده‌ی همگن را در بر می‌گیرد و بسیاری از عملیات‌های ریاضی در آن انجام می‌شوند. این کتابخانه امکان انجام عملیات ریاضی پیشرفته و سایر عملیات‌ها روی تعداد زیادی داده را با کارایی بالا فراهم می‌کند، که با استفاده از دنباله‌های پایتون معمولی کارآمدی و کد کمتری دارند.

### ۶-۶-۳ کتابخانه Matplotlib

Matplotlib یک کتابخانه متقابل پلتفرم<sup>۸</sup>، برای تجسم داده‌ها و نمودارهای گرافیکی (هیستوگرام، نمودارهای پراکنده، نمودار میله‌ای و غیره) برای پایتون است. توسعه دهندگان همچنین می‌توانند از رابطه‌های برنامه‌نویسی matplotlib برای جاسازی نمودارها در برنامه‌های رابط کاربری گرافیکی استفاده کنند.

یک اسکریپت matplotlib 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. لایه برنامه‌نویسی matplotlib دو رابط برنامه‌نویسی را پوشش می‌دهد:

<sup>8</sup>cross-platform

<sup>9</sup>Object Oriented

رابط برنامه‌نویسی `matplotlib.pyplot`, که سلسله مراتبی از اشیاء کد پایتون است که در بالای آن قرار دارد. و رابط برنامه‌نویسی اشیاء گرا<sup>۹</sup> که می‌تواند با انعطاف پذیری بیشتری نسبت به `pyplot` مونتاژ شوند. این رابط برنامه‌نویسی دسترسی مستقیم به لایه‌های Backend Matplotlib را فراهم می‌کند.

### ۷-۶-۳ کتابخانه OpenCV

یک کتابخانه متن باز برای بینایی کامپیوترا و یادگیری ماشین است که برای فراهم کردن زیرساخت مشترک برای برنامه‌های بینایی کامپیوترا و تسريع 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 کامپیوترا و یادگیری ماشین را فراهم می‌کند. OpenCV به طور گسترده‌ای در شرکت‌ها، گروه‌های تحقیقاتی و نهادهای دولتی برای انجام پروژه‌های بینایی کامپیوترا و یادگیری ماشین استفاده می‌شود. این کتابخانه رابطه‌ای برنامه‌نویسی متعددی از جمله MATLAB، C++، Java و Python را فراهم می‌کند.

### ۸-۶-۳ پهپاد DJI Tello

پهپاد DJI Tello یک پهپاد کوادکوپتر کوچک و قابل برنامه‌ریزی است که برای مصارف آموزشی و تست پروتوتایپ توسط DJI طراحی شده است. این پهپاد دارای ویژگی‌های ویژه‌ای مانند حرکات پایه‌ای کوادکوپترا و همچنین تکنولوژی کنترل پرواز DJI و یک پردازنده Intel بسیار قوی است. دوربین ۵ مگاپیکسلی این پهپاد امکان ضبط ویدیو با کیفیت خوب را فراهم می‌کند. همچنین، پهپاد دارای یک سیستم موقعیت‌یابی بصری<sup>۱۰</sup> است که شامل یک دوربین و یک ماژول مادون قرمز ۳ بعدی است و قادر است در فواصل ۳۰ متر تا ۳۰ متر ارتفاع کار کند.

هسته پهپاد به عنوان مرکز پردازشی و کنترلی آن عمل می‌کند و از یک پردازنده Intel قدرتمند پشتیبانی می‌کند. پهپاد DJI Tello برای اجرای پروژه‌های هوش مصنوعی مانند تشخیص اشیاء روی پهپاد، از زبان برنامه‌نویسی پایتون و SDK مربوطه پشتیبانی می‌کند. این SDK به کاربران این امکان را می‌دهد که پروتوتایپ‌های پایه‌ای پروژه‌های خود را توسعه دهند و آن‌ها را بر روی پهپاد اجرا کنند.

این پهپاد دارای باتری با جزئیات خاصیت مانند زمان پرواز و زمان شارژ است که می‌تواند از نظر عملکرد

<sup>10</sup>Vision Positioning System

و ماندگاری باتری نسبت به پهپادهای دیگر مزیت داشته باشد. همچنین، پروژه‌های هوش مصنوعی که روی پهپاد DJI Tello پیاده‌سازی می‌شوند، می‌توانند شامل تشخیص اشیاء، پیش‌بینی حرکت‌ها و یا حتی خودکارسازی فرآیندهای پروازی باشند. از جمله مدل‌های هوش مصنوعی که می‌تواند روی پهپاد DJI Tello پیاده‌سازی شود، می‌توان به YOLOv3 اشاره کرد که برای تشخیص اشیاء با دقت بالا استفاده می‌شود. [۲۶]

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)
Transmitter (EIRP)	19 dBm (CE)
	19 dBm (SRRC)

شکل ۴-۳ اطلاعات پهپاد DJI Tello

### ۷-۳ مدیاپایپ

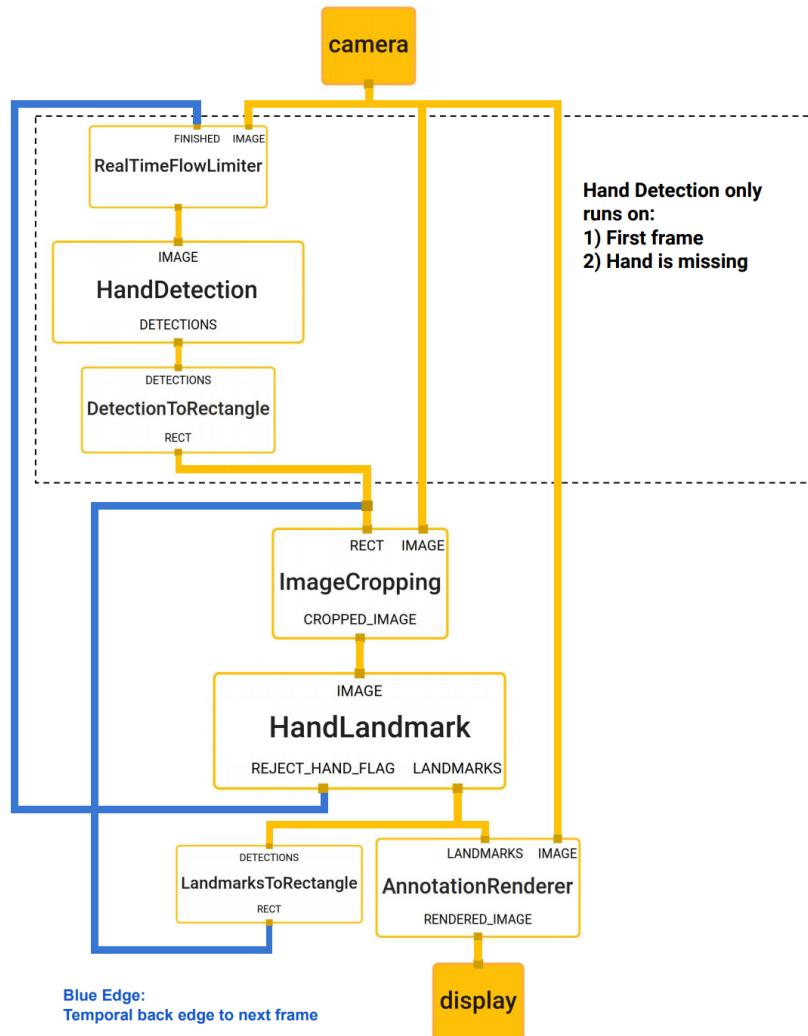
برای پیاده‌سازی شبکه‌های تشخیص کف دست و پیدا کردن نقاط عطف دست از مدل‌های از قبل آموزش دیده<sup>۱۱</sup> کتابخانه مدیاپایپ کمک گرفته شده است. مدیاپایپ از یک خط لوله یادگیری ماشین متشکل از چندین مدل که با هم کار می‌کنند استفاده می‌کند: یک مدل تشخیص کف دست<sup>۱۲</sup> که تصویر را از ورودی می‌گیرد و عکس محدوده دست را به عنوان خروجی دریافت می‌کند و یک مدل تشخیص نقاط عطف دست<sup>۱۳</sup> که عکس دست را به عنوان ورودی گرفته و مختصات ۲۱ نقطه کلیدی بندهای انگشتان دست را در ناحیه دست تشخیص می‌دهد. در عکس ۳-۵ مازول‌های مربوط به شناسایی نقاط کلیدی نشان داده شده که به تفکیک هر کدام را توضیح خواهیم داد.

<sup>11</sup>Pretrained

<sup>12</sup>Palm detection model

<sup>13</sup>Hand landmark model

<sup>14</sup>Real Time Flow Limiter



شکل ۳-۵ نمودار MediaPipe برای شناسایی نقاط کلیدی دست

### ۱-۷-۳ محدود کننده جریان زمان واقعی<sup>۱۴</sup>

ماژول محدود کننده جریان زمان واقعی وظیفه محدود کردن جریان پردازش به سرعت زمان واقعی را بر عهده دارد. این ماژول امکان کنترل پردازش داده‌ها را به صورت کارآمد و در زمان واقعی فراهم می‌کند، بدون ایجاد تأخیر یا بار اضافی بر سیستم. ورودی‌های این ماژول شامل جریان‌های بسته<sup>۱۵</sup> و مهر زمان‌ها<sup>۱۶</sup> می‌شود و خروجی‌های آن شامل جریان‌های بسته با نرخ محدود<sup>۱۷</sup> و اطلاعات مربوط به بسته‌های

<sup>15</sup>Packet Streams

<sup>16</sup>Time stamp

<sup>17</sup>Rate-Limited Packet Streams

<sup>18</sup>Input Buffering

<sup>19</sup>Timestamp Analysis

<sup>20</sup>Rate Limiting Algorithm

<sup>21</sup>Output Buffering

حذف شده ناشی از محدودیت نرخ پردازش است. معماری این ماژول شامل مراحل بافر ورودی<sup>۱۸</sup>، تجزیه و تحلیل مهر زمانی<sup>۱۹</sup>، الگوریتم محدود کردن نرخ<sup>۲۰</sup> و بافر خروجی<sup>۲۱</sup> است که به منظور محدود کردن نرخ جریان داده‌ها و انجام پردازش در زمان واقعی طراحی شده است. این ماژول کمک می‌کند تا بار پردازش کاهش یابد، تأخیر کاهش یابد، کیفیت خدمات حفظ شود و منابع محاسباتی بهینه‌سازی شوند.

[۱۹]

## ۲-۷-۳ تشخیص دست<sup>۲۲</sup>

ماژول تشخیص کف دست که یکی از دو ماژول اصلی است، در مدیاپایپ دارای دقت متوسط ۷.۹۵ درصد است که این دقت بالا با استفاده از استراتژی‌های مختلف به دست آمده است. این ماژول از یک مدل تشخیص کف دست استفاده می‌کند به جای تشخیص دست، زیرا تشخیص محدوده‌های اجسام سفت و سخت مانند کف دست و مشت بسیار ساده‌تر از تشخیص دست‌ها با انگشتان مفصلی است. از الگوریتم سرکوب غیر حداکثری برای حذف تشخیص‌های تکراری و انتخاب مرتبط‌ترین اشیاء شناسایی شده استفاده می‌شود که به کاهش مثبت کاذب و پیچیدگی محاسباتی کمک می‌کند. وظیفه اصلی این ماژول تشخیص دست در تصویر و محاسبه مکان دقیق دست است. این ماژول برای تشخیص دست در تصویر و محاسبه مکان دقیق دست استفاده می‌شود و پس از تشخیص دست، مکان دقیق دست را به ماژول‌های بعدی ارسال می‌کند.

ماژول تشخیص دست در مدیاپایپ یکی از ابزارهای کلیدی برای تشخیص دست‌ها در تصاویر و ویدیوها 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 و پارامترهای تنظیمات است، در حالی که خروجی آن شامل مستطیلهای محدوده دست‌ها، نمرات اطمینان و در صورت فعال بودن، دست غالب<sup>۲۳</sup> است. معماری این ماژول شامل مراحل پیش پردازش<sup>۲۴</sup>، مدل تشخیص دست و پس پردازش<sup>۲۵</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

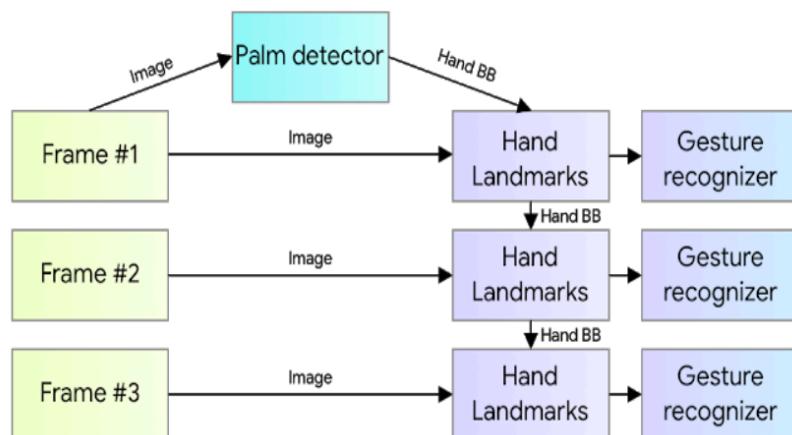
<sup>22</sup>Hand Detection

<sup>23</sup>handedness

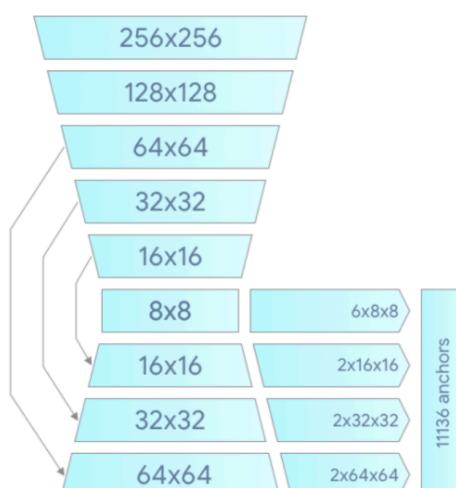
<sup>24</sup>Preprocessing

<sup>25</sup>Postprocessing

مختلفی از جمله تشخیص حرکات دست، رابطه‌های کاربری بدون لمس، تحلیل رفتار و ژست‌ها و کمک به افراد کم‌توان اهمیت دارد. این مازول به توسعه‌دهندگان این امکان را می‌دهد که به راحتی و با دقت بالا دست‌ها را در تصاویر و ویدیوها شناسایی کرده و از این اطلاعات برای پردازش‌های بعدی استفاده کنند.



شکل ۳-۶ خط لوله تشخیص دست



شکل ۳-۷ معماری مدل آشکارساز کف دست

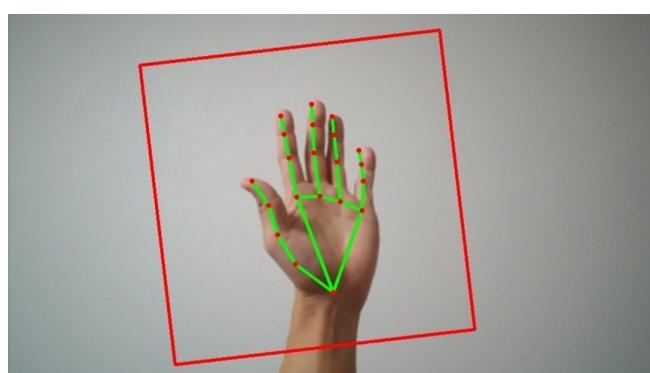
<sup>26</sup>Detection To Rectangle

<sup>27</sup>Detection Proto

<sup>28</sup>Image Frame

### ۳-۷-۳ تشخیص به مستطیل ۲۶

این مازول وظیفه تبدیل نتایج تشخیص اشیاء به مستطیل‌های محدودکننده دست دارد. این مازول ورودی‌هایی مانند تشخیص پروتوی<sup>۲۷</sup> و فریم تصویر<sup>۲۸</sup> را دریافت کرده و مستطیل‌های محدودکننده را به صورت نرمال‌سازی شده و یا مطلق برای نواحی تشخیص داده شده تولید می‌کند. معماری این مازول شامل مراحل پیش‌پردازش، تبدیل تشخیص<sup>۲۹</sup> و پس‌پردازش است. وظیفه اصلی این مازول تبدیل نتایج تشخیص به مستطیل‌های محدودکننده است که در کاربردهای مختلف مانند تشخیص و ردیابی چهره، تشخیص دست، امنیتی و واقعیت افزوده کاربرد دارد. [۱۹]



شکل ۳-۸ پیدا کردن مستطیل حاوی دست

### ۴-۷-۳ برش تصویر ۳۰

مازول برش تصویر برای برش و استخراج ناحیه‌های مورد نظر از تصاویر استفاده می‌شود. این مازول ورودی‌هایی مانند تصویر اصلی و جعبه مرزی<sup>۳۱</sup> یا مختصات برش را می‌پذیرد و تصویر برش خورده حاوی ناحیه مورد نظر را تولید می‌کند. معماری این مازول شامل مراحل پیش‌پردازش، عملیات برش<sup>۳۲</sup> و پس‌پردازش می‌باشد. وظیفه اصلی این مازول برش دقیق ناحیه‌های مورد نظر از تصاویر اصلی است که در کاربردهای مختلفی مانند پیش‌پردازش برای تشخیص چهره یا دست، تحلیل تصاویر پزشکی، واقعیت افزوده و پردازش تصویر در کاربردهای امنیتی مورد استفاده قرار می‌گیرد. این مازول به توسعه‌دهندگان این امکان را می‌دهد که ناحیه‌های خاصی از تصاویر را به صورت دقیق و مؤثر استخراج کرده و برای پردازش‌های بعدی آماده کنند. [۱۹]

<sup>29</sup>Detection Conversion

<sup>30</sup>Image Cropping

<sup>31</sup>bounding box

<sup>32</sup>Cropping Operation

### ۳-۷-۵ تشویص نقاط عطف دست<sup>۳۳</sup>

این مازول که دیگر اصلی است، یک ابزار قدرتمند برای تشخیص و ردیابی نقاط کلیدی دست است. این مازول وظیفه تشخیص و محاسبه نقاط عطف دست را بر عهده دارد. ورودی این مازول 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 ناحیه‌ای باشد که دست در آن قرار دارد (جعبه مرزی) که توسط مازول دیگری مانند تشخیص دست مشخص شده است.

خروجی این مازول شامل موقعیت سه‌بعدی ۲۱ نقطه کلیدی دست است که شامل مفاصل انگشتان و نوک انگشتان می‌باشد. این نقاط کلیدی به صورت مجموعه‌ای از مختصات (x, y, z) ارائه می‌شود که موقعیت هر نقطه را در فضای سه‌بعدی نشان می‌دهد و  $\square$  نشان‌دهنده عمق نقطه نسبت به تصویر ورودی است.

معماری این مازول شامل استفاده از شبکه‌های عصبی عمیق برای تشخیص و ردیابی نقاط کلیدی دست است. 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 پس پردازش برای تبدیل مختصات نقاط کلیدی به فرمت خروجی نهایی و اعمال اصلاحات لازم است.

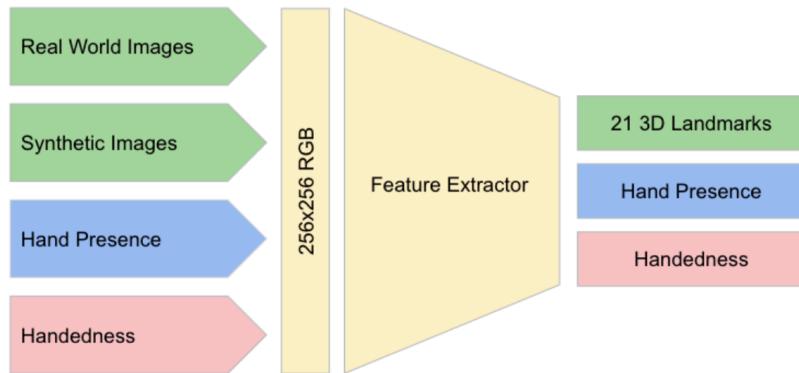
وظیفه اصلی این مازول تشخیص و ردیابی دقیق نقاط کلیدی دست‌ها در تصاویر و ویدیوها است. این قابلیت می‌تواند در برنامه‌های مختلفی مانند واقعیت افزوده (AR)، رابطه‌ای کاربری بدون لمس، تحلیل حرکات، تشخیص حرکات دست در زبان اشاره و ... استفاده شود و توسعه‌دهندگان را قادر می‌سازد از این قابلیت‌های پیشرفته در برنامه‌های خود بهره ببرند. [۱۹]

### ۳-۷-۶ نقاط عطف یه مستطیل<sup>۳۴</sup>

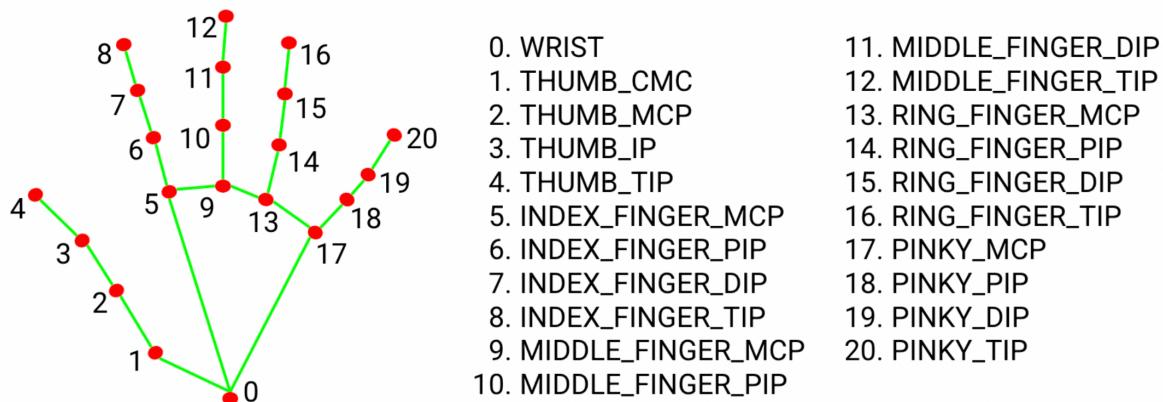
این مازول برای تبدیل نقاط کلیدی به مستطیل‌های محدود‌کننده استفاده می‌شود. این مازول وظیفه دقیق محصور کردن نقاط کلیدی یک شیء، مانند دست یا چهره، را بر عهده دارد. با استفاده از این مستطیل‌ها، می‌توان موقعیت دقیق‌تر اشیاء در تصویر را نمایش داد و از آنها به عنوان ورودی برای مازول‌های بعدی در گراف استفاده کرد. این مازول شامل مراحل پیش‌پردازش داده‌های ورودی، محاسبه مستطیل محدود‌کننده، نرمال‌سازی مختصات مستطیل‌ها و تولید خروجی نهایی می‌باشد. وظیفه اصلی

<sup>33</sup>Hand Landmark

<sup>34</sup>Landmarks To Rectangle



شکل ۳-۹ 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.



شکل ۳-۱۰-۲۱ نقطه کلیدی در ناحیه دست

این مازول تبدیل نقاط کلیدی به مستطیل‌های محدودکننده دقیق است که در بسیاری از کاربردها اهمیت دارد، از جمله ردیابی اشیاء، تحلیل حرکات، پیش‌پردازش برای مدل‌های دیگر و افزایش دقت در پردازش تصویر. [۱۹]

### ۷-۷-۳ ارائه کننده حاشیه نویسی<sup>۳۵</sup>

این مازول برای نمایش گرافیکی نتایج پردازش‌های مختلف بر روی تصاویر یا ویدیوها استفاده می‌شود. این مازول اطلاعات حاشیه‌نویسی شامل نقاط کلیدی، مستطیل‌های محدودکننده، خطوط، متن و سایر اشکال گرافیکی را به صورت بصری بر روی تصاویر نمایش می‌دهد. وظیفه اصلی این مازول شامل نمایش بصری نقاط کلیدی، مستطیل‌های محدودکننده، خطوط و اتصالات، متن و توضیحات بر روی تصاویر یا ویدیوها است. این مازول به توسعه دهنده‌گان امکان می‌دهد تا به راحتی و به صورت بصری نتایج

<sup>35</sup> AnnotationRenderer

پردازش‌های خود را مشاهده کنند و از این طریق به بهبود و ارزیابی عملکرد مدل‌ها و الگوریتم‌های خود بپردازنند. [۱۹]

### ۸-۳ پیش‌پردازش

برای ورود نقاط عطف دست به مدل تعیین ژست 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  $۲۰۴۸ \times ۲۰۴۸$  این اعداد از بین  $۰$  تا  $۲۰۴۸$  متغیر است. اگر این مختصات را به صورت مستقیم به مدل تعیین ژست دست بدھیم دقت مدل برابر  $۸۷$  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 فرمول زیر مختصات نقاط دیگر را به روز رسانی می‌کنیم.

$$X_{\text{rel}} = X_{\text{ref}} - X$$

پس از به نسبی کردن نقاط نسبت به مبدأ، آنها را با کمک فرمول زیر نرمال‌سازی می‌کنیم تا تمام طول و عرض نقاط به عددی میان صفر و یک به روز رسانی شوند.

$$X_{\text{new}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

در انتها این این مختصات را به عنوان ورودی به شبکه تعیین ژست دست می‌دهیم. با توجه به اینکه معماری هیچ یک از مدل‌ها تغییر نکرد و تنها داده‌های مختصات به روز رسانی شدند، دقت نهایی مدل به  $۹۷$  درصد افزایش پیدا کرد و پیش‌پردازش تاثیر بهسزایی در بهینه کردن پروژه داشت.

### ۹-۳ پسپردازش

با وجود اینکه دقت مدل پیاده‌سازی شده بالا 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 هزینه مالی. 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 FPS دوربین در نظر می‌گیریم بدین صورت که هر چه FPS دوربین پهپاد بیشتر باشد متغیر در نظر گرفته شده نیز بیشتر است، برای مثال در پروژه ما از آنجایی که دوربین پهپاد برابر ۳۰ FPS است ما این متغیر را ۱۰ قرار داده‌ایم. سپس حد مناسبی را نیز بین صفر تا یک قرار می‌دهیم که ما در پروژه آن را برابر ۷۰ قرار داده‌ایم. طبق این راه ما ۱۰ فریم متناوب گرفته شده از پهپاد را به مدل پیاده‌سازی شده می‌دهیم اما تنها در صورتی دستور پیش‌بینی شده را به پهپاد پهپاد می‌دهیم که حد ممکن را بدست آورند. برای مثال اگر ۷ یا بیشتر از ۱۰ 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 ۷ عدد از فریمهای یک ژست دست را پیش‌بینی نکنند، پهپاد در حالت قبلی خود باقی می‌ماند و دستوری به آن داده نمی‌شود.

### ۱۰-۳ جمع‌بندی

در این فصل قسمت‌های مختلف پیاده‌سازی شده پروژه با جزئیات کامل توضیح داده شد. از جمله آنها نرم‌افزارها و سخت‌افزارها، کتابخانه‌ها، پیش‌پردازش، پس‌پردازش و کد اصلی که شامل سه مدل به صورت پی در پی که دو تا از آنها با کمک کتابخانه MediaPipe و دیگری یک کلاس‌بندی با ۹ حالت دست است بود.

# فصل چهارم

## نتایج و ارزیابی

## ۱-۴ مقدمه

سیستم تشخیص حرکات دست به نقش مهمی در ایجاد تعامل کارآمد بین انسان و ماشین تبدیل شده است. پیاده‌سازی این سیستم با استفاده از تشخیص ژست دست، نوید گستره وسیعی از کاربردها را در صنعت فناوری می‌دهد. در این پروژه، معماری‌های گوناگونی مانند شبکه‌های عصبی پیچشی، شبکه‌های حافظه کوتاه‌مدت بلند، و شبکه‌های عصبی چندلایه مورد آزمایش قرار گرفتند تا بهترین پیاده‌سازی برای تشخیص حرکات دست انتخاب شود.

در این فصل، به بررسی نتایج به دست آمده از این پروژه پیاده‌سازی شده 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 خواهیم داد. 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 ارزیابی عملکرد مدل‌ها

این پروژه با مدل‌های گوناگونی پیاده‌سازی شد تا بتوان بهترین آنها را برای نتیجه نهایی بر روی پهپاد اجرا کرد. معیارهای ارزیابی شامل دقت، صحت<sup>۱</sup>، فراخوانی<sup>۲</sup>، امتیاز F1 و تعداد نمونه‌ها برای هر یک از ژست‌ها می‌باشد.

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

## ۱-۲-۴ دقت

دقت مدل معیاری است که نشان می‌دهد یک مدل یادگیری ماشینی چقدر قادر به پیش‌بینی یا تصمیم‌گیری بر اساس داده‌ها است. این معیار به صورت مجموع مثبت و منفی واقعی تقسیم بر تعداد کل نمونه‌ها محاسبه می‌شود.

<sup>1</sup>Precision

<sup>2</sup>Recall

دقت بصری ترین معیار عملکرد است و برابر نسبت مشاهدات پیش‌بینی شده درست به کل مشاهدات است. این معیار می‌تواند برای مقایسه عملکرد مدل‌های مختلف یا ارزیابی اثربخشی یک مدل خاص برای یک کار معین استفاده شود.

دقت به صورت مجموع مثبت و منفی واقعی تقسیم بر تعداد کل نمونه‌ها محاسبه می‌شود. همچنین این معیار زمانی مناسب است که کلاس‌ها به خوبی متعادل باشند و هزینه‌های مثبت کاذب و منفی کاذب مشابه باشد.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative}$$

معماری مدل دوره ۳ دقت داده آموزش دقت داده تست			
۱	۷	۶	MLP
۲	۸	۷	CNN
۳	۸	۵	LSTM

جدول ۱-۴ جدول ارزیابی دقت مدل‌ها

## ۲-۲-۴ صحت

صحت یک معیار آماری برای ارزیابی کیفیت یک مدل پیش‌بینی است. این یکی از معیارهای کلیدی است که برای تعیین عملکرد یک مدل، به ویژه در وظایف طبقه‌بندی استفاده می‌شود. صحت نسبت مثبت واقعی به مجموع مثبت‌های واقعی و مثبت کاذب (نمونه‌هایی که به اشتباہ به عنوان مثبت شناسایی شده‌اند) است.

صحت بالا نشان دهنده این است که یک مدل در جلوگیری از مثبت کاذب خوب عمل می‌کند، به عنوان مثال، نمونه‌های منفی را به عنوان مثبت طبقه‌بندی نمی‌کند. این امر به ویژه در برنامه‌هایی که هزینه مثبت کاذب بالا است، اهمیت دارد.

$$Precision = \frac{True\ Positive}{True\ Positive - False\ Positive}$$

### ۳-۲-۴ فراخوانی

معیار فراخوانی که به عنوان نرخ مثبت واقعی نیز شناخته می‌شود، این معیار برابر نمونه‌های داده‌ای است که یک مدل یادگیری ماشینی به درستی آن‌ها را تشخیص داده نسبت به کل نمونه‌های آن کلاس است. فراخوانی زمانی استفاده می‌شود که به حداقل رساندن منفی‌های کاذب از اهمیت ویژه‌ای برخوردار باشد. بدین صورت که هزینه بر روی مثبت کاذب کم باشد و یا هزینه از دست دادن مثبت واقعی زیاد باشد.

$$Recall = \frac{True\ Positive}{True\ Positive - False\ Negative}$$

### ۴-۲-۴ امتیاز F1

امتیاز F1 معیاری برای میانگین هارمونیک دقت و یادآوری است. امتیاز F1 که معمولاً به عنوان معیار ارزیابی در طبقه‌بندی باینری و چند کلاسه استفاده می‌شود، صحت و یادآوری را در یک متريک واحد ادغام می‌کند تا درک بهتری از عملکرد مدل به دست آورد.

امتیاز F1 یک معیار مفید برای اندازه‌گیری عملکرد مدل‌های طبقه‌بندی برای زمانی است که داده‌ها نامتعادل‌اند، زیرا نوع خطاهای - مثبت کاذب و منفی کاذب - و نه فقط تعداد پیش‌بینی‌های نادرست را در نظر می‌گیرد.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

### ۵-۲-۴ گزارش معیارهای ارزیابی در مدل‌ها

در این بخش، نتایج ارزیابی مدل‌ها برای تشخیص ۹ ژست مختلف دست ارائه شده است.

### ۴-۳ نمودارهای دقت و خطا بر حسب دوره

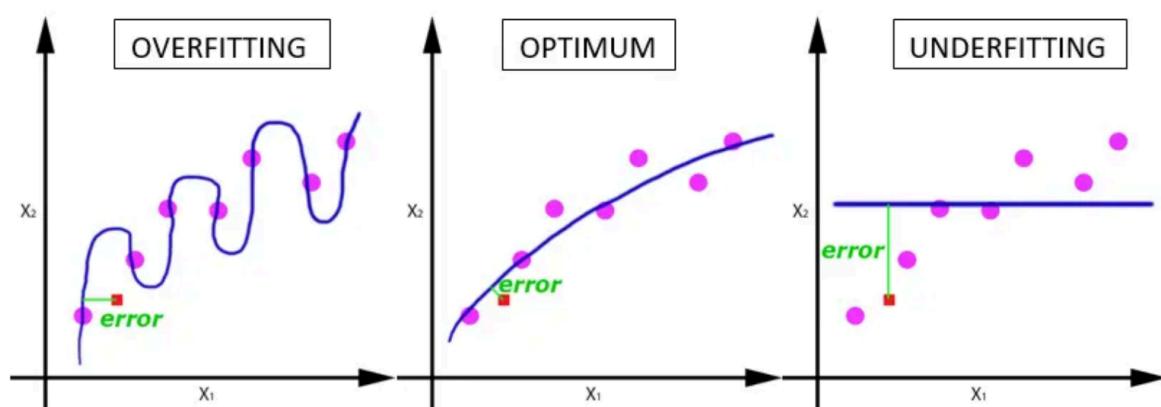
یک دوره زمانی است که کل مجموعه داده تنها یک بار از طریق شبکه عصبی به جلو و عقب منتقل می‌شود. از آنجایی که یک دوره آنقدر بزرگ است که نمی‌توان آن را به یکباره به سیستم وارد کرد، آن را به چند دسته کوچکتر تحت عنوان دوره تقسیم می‌کنیم. انتقال کل مجموعه داده از طریق یک شبکه عصبی کافی نبوده و باید مجموعه داده را چندین بار به یک شبکه عصبی ارسال کرد. ما برای این پروژه از

	precision	recall	f1-score	support
0	0.96	0.96	0.96	24
1	0.92	0.98	0.95	47
2	1.00	1.00	1.00	23
3	0.95	0.88	0.91	24
4	0.91	1.00	0.95	21
5	1.00	0.83	0.91	30
6	0.93	1.00	0.96	13
7	1.00	1.00	1.00	9
8	1.00	1.00	1.00	21
9	0.89	0.94	0.91	17
accuracy			0.95	229
macro avg	0.96	0.96	0.96	229
weighted avg	0.95	0.95	0.95	229

شکل ۱-۴ معیارهای ارزیابی برای تشخیص ژست دست در مدل MLP

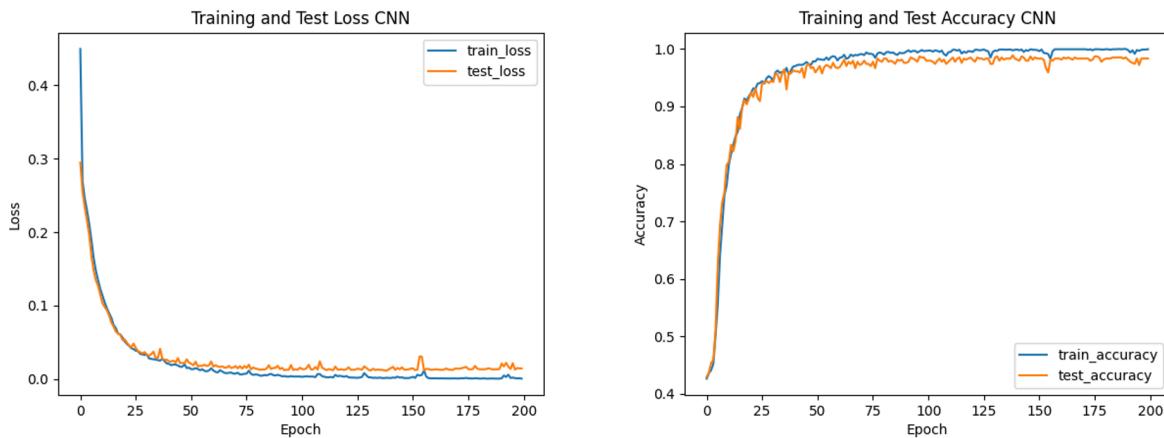
یک مجموعه داده محدود استفاده می کنیم و برای بهینه سازی یادگیری آن از گرادیان کاهشی استفاده می کنیم.

تعیین تعداد دوره از اهمیت بالایی برخوردار است. با افزایش تعداد دوره ها، تعداد دفعات تغییر وزن در شبکه عصبی بیشتر می شود و منحنی از کم برآش<sup>۴</sup> به منحنی بهینه<sup>۵</sup> و بعد به منحنی بیش برآش<sup>۶</sup> می رود. میزان دوره ها باید به گونه ای تعیین شوند که متعادل عمل کنند و بتوان منحنی را به بهینه ترین حالت ممکن رسانند.



شکل ۲-۴ منحنی های کم برآش، بهینه و بیش برآش

<sup>4</sup>Underfitting<sup>5</sup>Optimal<sup>6</sup>Overfitting



شکل ۴-۳ نمودار روند دقت و خطا بر حسب دوره در داده‌های آموزش و تست در مدل CNN

#### ۴-۴ سرعت اجرای برنامه

از آنجایی که یکی از بزرگترین اهداف این پروژه زمان واقعی بودن آن است باید میزان پاسخگویی مدل را نیز در نظر قرار داد. در جدول زیر زمان پاسخگویی مدل‌ها از زمانی که داده‌ها از طریق دوربین خوانده می‌شوند تا زمانی که دستور به پهپاد داده می‌شود نشان داده شده.

معماری مدل زمان پاسخگویی مدل	
۶	MLP
۷	CNN
۵	LSTM

جدول ۴-۴ جدول ارزیابی زمان پاسخگویی مدل‌ها

#### ۵-۴ سخت‌افزار مورد نیاز

این پروژه باید به گونه‌ای اجرا می‌شد که بر روی ساده‌ترین سیستم‌های کامپیوتری نیز قابل اجرا باشد، زیرا سخت‌افزار پهپادها به‌طور معمول دارای پردازنده‌های ضعیف‌تری هستند. همچنین، استفاده از کارت گرافیکی ممکن نبود، چرا که پهپادها قادر کارت گرافیکی می‌باشند. معما‌های پیاده‌سازی شده به نحوی طراحی شدند که تعادل میان دقت و بهره‌وری از سخت‌افزار حفظ شود، به طوری که هم قابلیت اجرای زمان واقعی داشته باشند و هم امکان پیاده‌سازی آن‌ها بر روی پهپاد فراهم باشد. از این‌رو،

معماری‌ها به گونه‌ای پیاده‌سازی شدند که بر روی پردازنده اجرا شوند و میزان استفاده از پردازنده برای آن‌ها به شرح زیر است:

معماری مدل پردازنده فضای ذخیره‌شده		
۷	۶	MLP
۸	۷	CNN
۸	۵	LSTM

جدول ۳-۴ جدول ارزیابی سخت‌افزار موردنیاز مدل‌ها

## ۶-۴ مقایسه دقت پروژه ما با کارهای مشابه

پروژه پیاده‌سازی از دقت بسیار بالایی برخوردار است که این دقت به دلیل ترکیب پیش‌پردازش، کتابخانه MediaPipe، مدل کلاس‌بندی و پس‌پردازش است. در این قسمت به مقایسه دقت پروژه پیاده‌سازی شده با پروژه‌های مشابه با هدف کنترل پهپاد با کمک تشخیص ژست دست پرداخته‌ایم.

## ۷-۴ جمع‌بندی

مدل‌هایی که در این پروژه ارائه شده‌اند، دارای دقت و کارایی مناسبی هستند که امکان استفاده آن‌ها در کاربردهای واقعی را فراهم می‌کند. به عبارت دیگر، این مدل‌ها می‌توانند به خوبی ژست‌های دست را تشخیص داده و از آن‌ها در عملیات مختلفی مانند کنترل دستگاه‌ها و رابطه‌های کاربری استفاده شود. علاوه بر این، با مقایسه عملکرد پروژه ما با کارهای مشابه در حوزه تشخیص ژست‌های دست، مشخص شده است که پروژه ما در مقایسه با آن‌ها دارای دقت مشابه یا حتی بالاتری است، به ویژه اگر تعداد کلاس‌های ژست دست در نظر گرفته شود. این نکته نشان می‌دهد که مدل‌هایی که در این پروژه ارائه شده‌اند، به خوبی تنوع و پیچیدگی ژست‌های دست 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 پروژه ما قادر است به عنوان یک راه حل کارآمد

نام مقاله	تعداد ژستهای دست	دقت تشخیص ژست دست
برنامه ما	۹	۶
Hand Gesture Controlled Drones: An Open Source Library[5]	۵	۹۷.۴۷۱
A real-time hand gesture recognition method[14]	۶	۹۳.۸
Hand Gestures For Drone Control Using Deep Learning[6]	۹	۸۳.۳
UAV-GESTURE: A Dataset for UAV Control and Gesture Recognition[15]	۱۳	۹۱.۹
Hand Gesture Recognition system for Real-Time Application[16]	۶	۹۰.۸
An improved hand gesture recognition system using keypoints and hand bounding boxes[17]	۶	۹۴
Hand-gesture recognition using computer-vision techniques[27]	۶	۹۳.۱
Hand Gesture Recognition using Image Processing and Feature Extraction Techniques[28]	۲۸	۸۹.۵۴
Applying Hand Gesture Recognition for User Guide Application Using MediaPipe[25]	۱۰	۹۵
MediaPipe Hands: On-device Real-time Hand Tracking[19]	۸	۹۴.۷
Hand Gesture Recognition System based in Computer Vision and Machine Learning[29]	۲۸	۹۳.۷۲
Hand Gesture Recognition Using Hidden Markov Models[30]	۵	۹۲.۱
Hand gesture recognition using Kinect[31]	۳۸	۸۴
Hand gesture recognition using neural networks[32]	۱۰	۸۹

جدول ۴-۴ جدول مقایسه پروژه با کارهای مشابه

برای تشخیص ژستهای دست در برنامه‌ها و سیستم‌های واقعی مورد استفاده قرار بگیرد.

# فصل پنجم

## نتیجه گیری و پیشنهادات

## ۱-۵ مقدمه

نتایج به دست آمده در این پژوهش نشان دهنده دقت بالای سیستم در تشخیص ژست‌ها و انتقال دستورات به پهپاد می‌باشد. با این حال، همچنان می‌توان بهبودهایی در عملکرد سیستم اعمال کرد تا دقت و کارایی آن افزایش یابد.

در این فصل، ما به بررسی و ارائه پیشنهاداتی می‌پردازیم که می‌توانند عملکرد سیستم کنترل پهپاد را از جنبه‌های مختلف بهبود بخشنند. این پیشنهادات شامل افزایش سرعت اجرای برنامه، بهبود دقت پیش‌بینی ژست‌ها 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 محدودیت وجود پهپادهای منبع باز

در مورد محدودیت‌های استفاده از پهپادهای منبع باز و شرکتی برای پژوهه کنترل پهپاد با استفاده از ژست دست بر مبنای بینایی ماشین، چندین موضوع موردنظر وجود دارد.

ابتدا، در مورد پهپادهای شرکتی، باید توجه داشت که این پهپادها عموماً 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 می‌تواند چالش‌هایی را ایجاد کند.

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 ژست دست خواهد شد.

## 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 مشکل ساز شود. زیرا الگوریتم‌ها باید بتوانند دست‌ها را تفکیک کرده و ژست‌های متفاوت آنها را تشخیص دهند. همچنین ممکن است دست‌های غیرمرتب با کاربر، مانند دستهای در پس‌زمینه یا دستهای دیگر افراد، نیز در تصویر حضور داشته باشند که این موضوع می‌تواند دقیقیت 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 دقیقیت 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 دقیقیت 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.

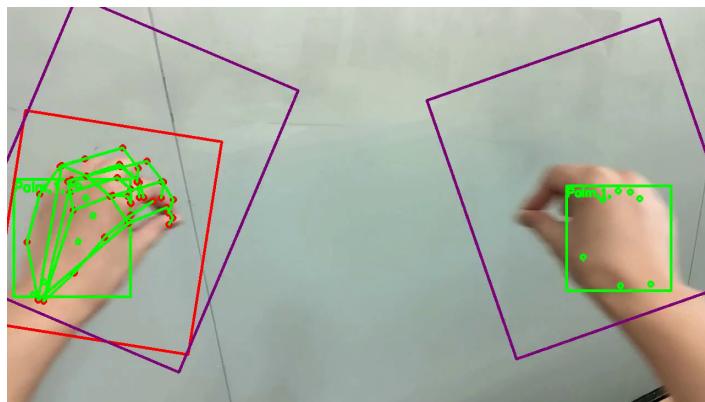
## ۴-۵ شناسایی یک دست در دو مستطیل در نتیجه دو خروجی

### دست

یکی از مشکلاتی که بسیاری از استفاده‌کنندگان کتابخانه MediaPipe با آن مواجه شده‌اند، مربوط به شناسایی یک دست با 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 پردازه خاصی که ما در دست داریم تفاوتی ایجاد نکرده است، زیرا ما یک گزینه را انتخاب کرده و ژست

دست را در آن پیش‌بینی می‌کنیم.

اما از بین بردن این مشکل می‌تواند بهبود و بهینه‌سازی کارایی کتابخانه برای سایر استفاده‌کنندگان مفید باشد. 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. این روش می‌تواند به استفاده‌کنندگان دیگر از کتابخانه MediaPipe کمک کند تا با مشکلاتی از این دست مواجه نشوند و از امکانات بهتری برای تشخیص دست بهره‌مند شوند.



شکل ۱-۵ شناسایی یک دست دو مرتبه

## ۵-۵ اجرا روی کارت گرافیکی<sup>۱</sup>

پروژه ما به صورتی پیاده‌سازی شده است که تنها بر روی پردازنده مرکزی قابل اجرا است، به علت عدم در دسترس بودن پهپادی که از کارت گرافیکی برای اجرای مدل‌ها استفاده کند. با این حال، اگر پهپادی با توانایی اجرای مدل‌ها بر روی کارت گرافیکی در دسترس باشد، ۳ مدل که به صورت پی‌درپی اجرا می‌شوند می‌توانند بر روی کارت گرافیکی اجرا شده تا عملکرد سیستم را بهبود بخشیده و تجربه کاربری را بهبود بخشند. استفاده از کارت 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 ژست دست شود و تجربه کاربری را بهبود بخشد. به علاوه، با استفاده از کارت گرافیکی، می‌توانیم بار محاسباتی را از پردازنده مرکزی کاهش داده و عملکرد کلی سیستم را بهبود بخشیم.

<sup>1</sup>GPU

## ۶-۵ اجرای پروژه با برنامه C++

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

- عملکرد و سرعت اجرای بالا: C++ 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 تفسری است. این ویژگی واقعی مانند کنترل پهپاد با استفاده از ژست دست بسیار حیاتی است، زیرا زمان پاسخ‌دهی سریع و دقیق در این سیستم‌ها اهمیت زیادی دارد.
- مدیریت دقیق حافظه: C++ 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: C++ دارای کتابخانه‌ها و چارچوب‌های قدرتمندی برای پردازش تصویر و بینایی ماشین مانند OpenCV است. OpenCV به زبان C++ 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: C++ 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 C++ می‌توان به طور مستقیم با اجزای سخت‌افزاری پهپاد ارتباط برقرار کرده و عملکرد سیستم را بهینه کرد.
- پشتیبانی قوی از چندریسمانی<sup>۲</sup>: C++ پشتیبانی قوی‌تری 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.

<sup>2</sup>Multithreading

## ۷-۵ استفاده از مدل‌های موازی برای تشخیص ژست دست

برای بهبود دقّت و کارایی پروژه کنترل پهپاد با استفاده از ژست دست، می‌توان از رویکردهای 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<sup>۳</sup> و مدل دیگر از معماری حافظه بلندمدت کوتاه‌مدت<sup>۴</sup> استفاده کند. تنها در صورتی که هر دو مدل تشخیص یکسانی از ژست دست داشته باشند، دستور نهایی به پهپاد ارسال می‌شود. این رویکرد می‌تواند به طور قابل توجهی دقّت تشخیص ژست دست را افزایش دهد و ریسک‌های مالی مرتبط با اشتباهات تشخیصی را کاهش دهد.

## ۸-۵ توسعه برنامه با استفاده از سایر قسمت‌های بدن انسان

یکی از مسیرهای جذاب برای توسعه آینده پروژه کنترل پهپاد با ژست 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.

## ۱-۸-۵ 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 دقّت 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 دقّت بیشتری 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

<sup>3</sup>Convolutional Neural Network

<sup>4</sup>Long Short Term Memory

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.

## ۲-۸-۵ 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 پژوهشی) بسیار مفید است. این روش می‌تواند به سادگی و طبیعی بودن تعامل با پهپاد کمک کند و تجربه کاربری را بهبود بخشد.

## ۹-۵ جمع‌بندی

در این فصل، ما به بررسی چندین چالش و محدودیت سیستم کنترل پهپاد پرداختیم و راهکارهای متعددی برای بهبود عملکرد آن ارائه دادیم.

اولین محدودیت، استفاده از پهپادهای منبع‌باز 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 دقیق سیستم را بهبود بخشد.

در ادامه، به اهمیت اجرای پروژه روی کارت گرافیکی و برنامه‌نویسی به زبان C++ پرداختیم که می‌تواند سرعت اجرای برنامه را افزایش دهد. استفاده از مدل‌های موازی برای تشخیص ژست دست نیز به عنوان یک راهکار مؤثر برای بهبود دقیق و سرعت پیشنهاد شد. 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.

# كتاب نامه

- [1] Walter, Ian and Khadr, Monette. Gesture controlled drone.
- [2] Puri, Vikram, Nayyar, Anand, and Raja, Linesh. 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] 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.
- [11] 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.
- [12] 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.
- [13] 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.
- [14] 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.
- [15] 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.
- [16] 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.
- [17] 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.
- [18] 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.
- [19] 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.
- [20] Waqar, Ahsan, Othman, Idris, Hamah Sor, Nadhim, Alshehri, Abdullah Mohammed, Almuhibah, 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.
- [21] Ghazi, Georges and Voyer, Julien. Use of a dji tello drone as an educational platform in the field of control engineering. *Proceedings of the Canadian Engineering Education Association (CEEA)*, 2023.
- [22] 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.
- [23] Kang, Seokmin and Tversky, Barbara. From hands to minds: Gestures promote understanding. *Cognitive Research: Principles and Implications*, 1:1–15, 2016.

- 
- [24] 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.
  - [25] 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.
  - [26] 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.
  - [27] Rios-Soria, David J, Schaeffer, Satu E, and Garza-Villarreal, Sara E. Hand-gesture recognition using computer-vision techniques. 2013.
  - [28] 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.
  - [29] 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.
  - [30] 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.
  - [31] Li, Yi. Hand gesture recognition using kinect. in *2012 IEEE International Conference*

*on Computer Science and Automation Engineering*, pp. 196–199. IEEE, 2012.

- [32] 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.