Improving Detection, Recognition and Identification of Humans and Objects by UAVs using AI & ML (YOLO, OpenCV)

Internship Report, August 2025

Shubhan Anchan

Grade XII Student, Bombay International School

Contents

Acknowledgements

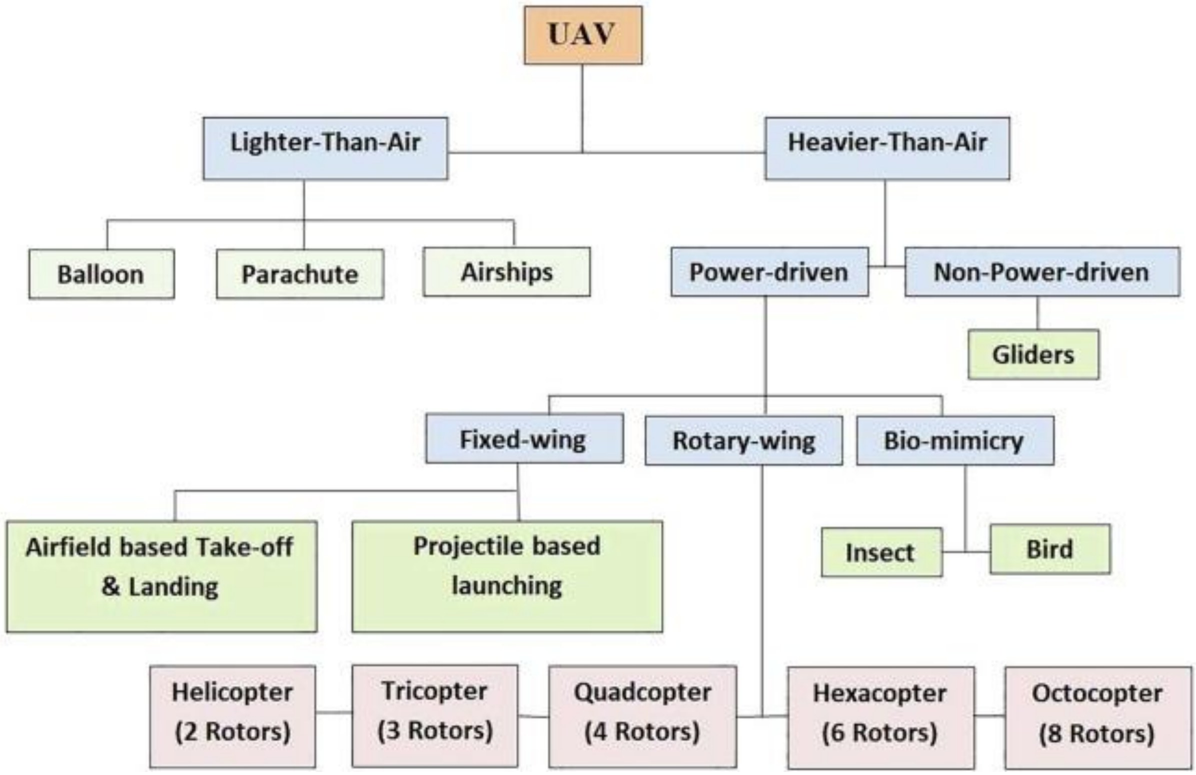
I would like to acknowledge the guidance provided by the following leaders at Asteria Aerospace:

- Mr Neel Mehta, Co-Founder, Asteria Aerospace
- Sasitharan A, Head of Validation & Verification
- NS SriPrasad, Validation & Verification Engineer
- Naveen N, Product Manager
- Mohammed Sabir, UAV Test Engineer

Introduction to Drones/ UAVs & UAS

Drones or Unmanned Aerial Vehicles

Drones, also known as unmanned aerial vehicles (UAVs), are aircraft systems that operate without an onboard human pilot and are instead controlled remotely or programmed for autonomous flight. A drone is essentially an aircraft equipped with sensors, cameras, navigation systems, and sometimes cargo space, designed to fly without direct human presence onboard.



Classification based on weight:

- Nano: <250 gm
- Micro: 250 gm to 2kg
- Small: 2 to 25 kg
- Medium: 25 to 150 kg
- Large: >150 kg

Classification based on Design:

- Fixed Wing
- Multi-Rotor
- Hybrid Vertical Take-off and Landing (VTOL)

Unmanned Aircraft Systems

They are part of a larger system known as an unmanned aircraft system (UAS), which includes the drone itself, its control station, and the communication link between them. UAVs can be piloted remotely by operators or programmed with artificial intelligence to perform autonomous missions.

Where do Drones/ UAVs find application?

Industrial and Infrastructure Applications

UAVs are widely utilized for inspecting infrastructure such as power lines, wind turbines, solar farms, bridges, and dams, providing aerial intelligence that enhances safety and reduces inspection time. In construction, drones support project monitoring, mapping, and surveying, helping ensure quality control and progressive reporting for stakeholders.

Remote Asset Maintenance e.g. cross country oil and gas pipelines, solar farms in deserts, hydroelectric dams etc,

Agriculture and Environmental Monitoring

UAVs enable precision agriculture by assisting with crop health assessment, irrigation management, spraying fertilizers and pesticides, and yield estimation. They play crucial roles in environmental monitoring, including forest health analysis, wildfire detection, pollution monitoring, and mapping deforestation.

Agriculture

- Seed dispersal over large acreage
- Measured watering, seed development and growth
- Pesticide spraying over large acreage

Disaster Management and Search & Rescue

Drones are indispensable for disaster management, where they can access dangerous areas to aid in rapid damage assessment, victim location, and delivery of essential supplies and medical kits during natural or man-made catastrophes. UAVs greatly improve the speed and effectiveness of search and rescue missions by providing real-time aerial imagery and supporting teams working in difficult or hazardous terrain.

Fire-fighting

- Discharge of water over hard-to-access fires (e.g. skyscrapers, wildfires)

Security, Surveillance, and Law Enforcement

The use of drones for surveillance includes monitoring large areas for security purposes, patrolling roads, antipiracy efforts, livestock monitoring, and crowd control. Law enforcement agencies employ drones for crime detection, evidence gathering, and traffic management, reducing risks for personnel while improving oversight capabilities.

Military

- Surveillance
- Reconnaissance
- Logistics – transport of goods to remote areas
- Loiter Munitions (LM)
- Direction of Own Artillery Fire (DOOAF)

Logistics, Delivery, and Healthcare

UAVs revolutionize logistics by transporting supplies, packages, medical kits, and even emergency vaccines to remote or inaccessible locations. Companies like Amazon have piloted drone-based delivery systems, while healthcare providers use drones for delivering medicines or reaching remote patients rapidly in crisis situations.

Surveying, Mapping, and Research

In scientific research and geospatial surveying, drones provide efficient aerial photography, create high-resolution maps and models, and support LiDAR or photogrammetry for topographical analysis. This expedites area coverage for urban planning, mining exploration, and conservation activities.

Geo-location

- Analysis and classification of traffic, based on which decisions (e.g.) on where to establish a petrol pump can be taken

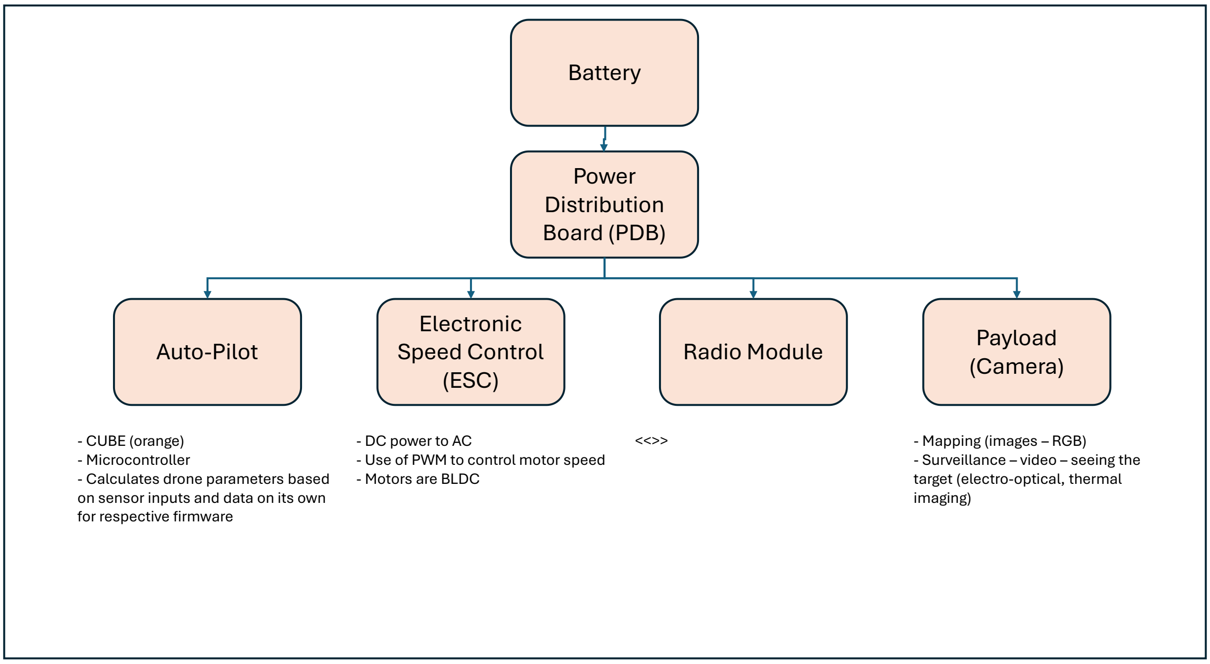
Insurance and Waste Management

Insurance companies deploy drones for damage assessment and claims verification, making post-disaster inspections faster and more reliable. In waste management, drones help identify illegal dumping sites and monitor landfill operations.

Introduction to Workings of UAVs and UAS

What are the key components of Unmanned Aerial Vehicles (UAVs)?

Unmanned Aerial Vehicles (UAVs) operate through a blend of mechanical, electronic, and software systems that enable autonomous or remote flight, navigation, and task execution.



Core Components

- Frame (Chassis): Serves as the backbone, holding all elements together. Usually made from lightweight materials like carbon fiber or aluminum, the frame's design directly affects aerodynamic efficiency and durability.
- Motors & Propellers: Motors (often brushless for efficiency) spin propellers, generating lift and thrust. The number, arrangement, and size of these affect stability, speed, and maneuverability.
- Electronic Speed Controllers (ESCs): ESCs regulate the power supplied to the motors based on signals from the flight controller, enabling controlled acceleration, deceleration, and precise maneuvers.
- Power System (Battery): UAVs primarily use rechargeable lithium-polymer (LiPo) batteries to supply power to all electronic and propulsion components.

Flight Control and Stability

- Flight Controller (FC): The “brain” of the UAV, the flight controller processes data from sensors and the pilot's inputs. It adjusts motor power and direction, maintaining balance and executing flight commands.
- Sensors:
- Gyroscopes and Accelerometers: Detect orientation and motion.
- Magnetometers and GPS: For directional alignment and positioning.
- Barometers: Measure altitude.
- Other sensors: May include cameras, LiDAR, ultrasonic sensors, depending on the use-case.

Remote Control or Autopilot

- UAVs are either manually controlled through a ground station or autonomously run pre-programmed flight plans using GPS and onboard computing.

Communication and Navigation

- Radio Transmitter & Receiver: UAVs communicate with ground stations or remote controls using radio waves, typically in 2.4 GHz or 5.8 GHz frequency ranges, allowing for real-time command transmission and telemetry.
- Telemetry Systems: Provide live feedback, such as battery status, flight location, camera feed, and sensor data to the operator.

Additional Systems and Payloads

- Camera Systems/Gimbals: For aerial imaging, surveillance, or videography, stabilized with gimbals for smooth movement.
- Landing Gear: Protects the UAV and payload during takeoff and landing.
- Payload: Depending on application, may include delivery packages, agricultural systems, sensors, or scientific equipment.

How do UAVs Fly?

Bernoulli’s Principle

Kinetic Energy, Potential Energy and Pressure Energy is constant, for a moving fluid

Sum of All Energies for a given volume of fluid is constant.

(Kinetic + Potential + Pressure Energy) per unit volume = Contant

(½ Mv^2 + Mgh + PV)/V = Constant

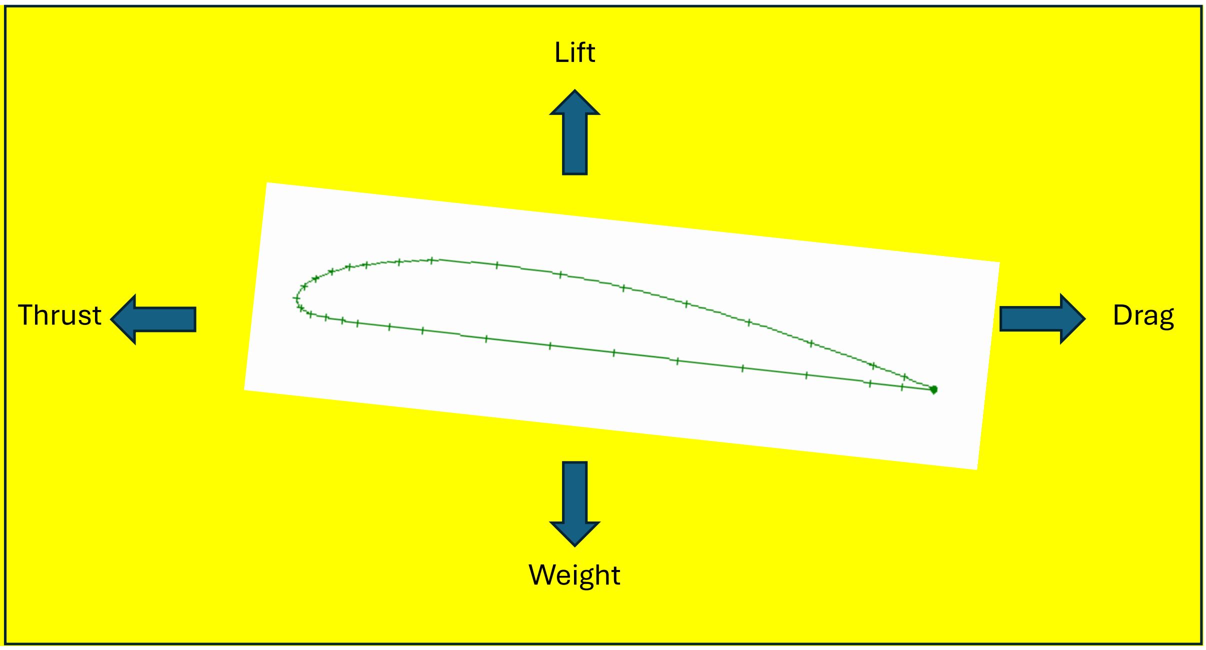
Mv^2/ 2V + Mgh/ V + P = Contant

RhoXv^2/2 + RhoXgXh + P = Constant

By implication, for a fluid at constant height, Pressure is inversely proportional to velocity.

Let us consider an Aerofoil to illustrate this concept:

There are four forces working on an Aerofoil.



Considering Bernoulli's equation:

1/2 X Rho X v1^2 + Rho X g X h1 + P1 = 1/2 X Rho X 21^2 + Rho X g X h2 + P2

(At constant height h, as speed v increases, Pressure P decreases and vice versa)

Bernoulli's principle is fundamental in managing the four main forces—lift, thrust, drag, and weight—in Unmanned Aerial Vehicles (UAVs). By understanding how air pressure varies with velocity around airfoils and moving UAV components, engineers can optimize flight performance, stability, and efficiency.

Bernoulli's Principle Explained

Bernoulli's principle states that an increase in the speed of a fluid (such as air) results in a simultaneous decrease in pressure. In UAVs, this is most commonly applied to the airflow over wings or propeller blades. A wing's upper surface is usually curved (cambered) while the lower surface is flatter. As the air travels faster over the curved upper surface than the straight lower surface, the pressure above the wing drops below that underneath, causing an upward lift force.

Lift

Lift is directly generated by the pressure difference caused by Bernoulli's principle. For UAVs:

The shape and orientation (angle of attack) of UAV wings or rotors cause air above to move faster than below, creating lower pressure on top and higher pressure beneath.

As a result, the net upward force (lift) opposes the UAV's weight (gravity), enabling flight and altitude control.

Engineers manipulate wing curvature, angle, and speed to fine-tune lift for specific UAV applications, such as surveillance or cargo delivery.

Illustration:

```text
           /\
          /  \   <-- Airfoil (wing cross-section)
      ____/    \____
    /              \
Fast airflow,       Slow airflow,
low pressure        high pressure

    ↑ Lift force generated (difference in pressure)
```

Thrust

Thrust in UAVs is managed using Bernoulli's principle through propeller and rotor blade design:

Propellers are shaped as rotating airfoils; as the blade moves, the airspeed difference above and below creates a pressure differential.

Forward thrust is produced by pushing air backwards, with the shape and angle of the propeller harnessing Bernoulli's effect to efficiently generate thrust and minimize energy loss.

Engineers optimize rotational speed and blade geometry to maximize thrust and minimize drag.

Drag

Drag is a resistive force that opposes thrust. Bernoulli's equation helps engineers manage drag by shaping UAV components:

Streamlined airfoils reduce turbulent airflow, ensuring smooth pressure distribution and less resistance.

Features like winglets or fairings guide airflow, lowering the pressure drag that slows UAVs down.

Balancing Bernoulli's effect and minimizing sudden changes in shape help UAVs maintain efficient flight profiles.

Weight

Weight is the force of gravity acting downward. Bernoulli's principle doesn't alter weight directly, but controlling lift through wing design and angle of attack lets UAVs overcome weight during various flight maneuvers.

Integrated Control in UAVs

In UAV flight, all four forces interact:

Lift (via Bernoulli's principle) must counteract weight for sustained flight.

Thrust (also using Bernoulli's effect in rotors/propellers) must overcome drag for forward motion.

Engineers use predictive modeling, wind tunnel testing, and real-time sensors (pitot tubes, static ports) calibrated based on Bernoulli's principle to actively manage all forces and maintain stable, efficient UAV operation.

Schematic Illustration

```text
         ↑ Lift           ← Drag
        /        \       /
      /            \   /
U A V ——→ Thrust ——→
        ↓ Weight
```

Conclusion

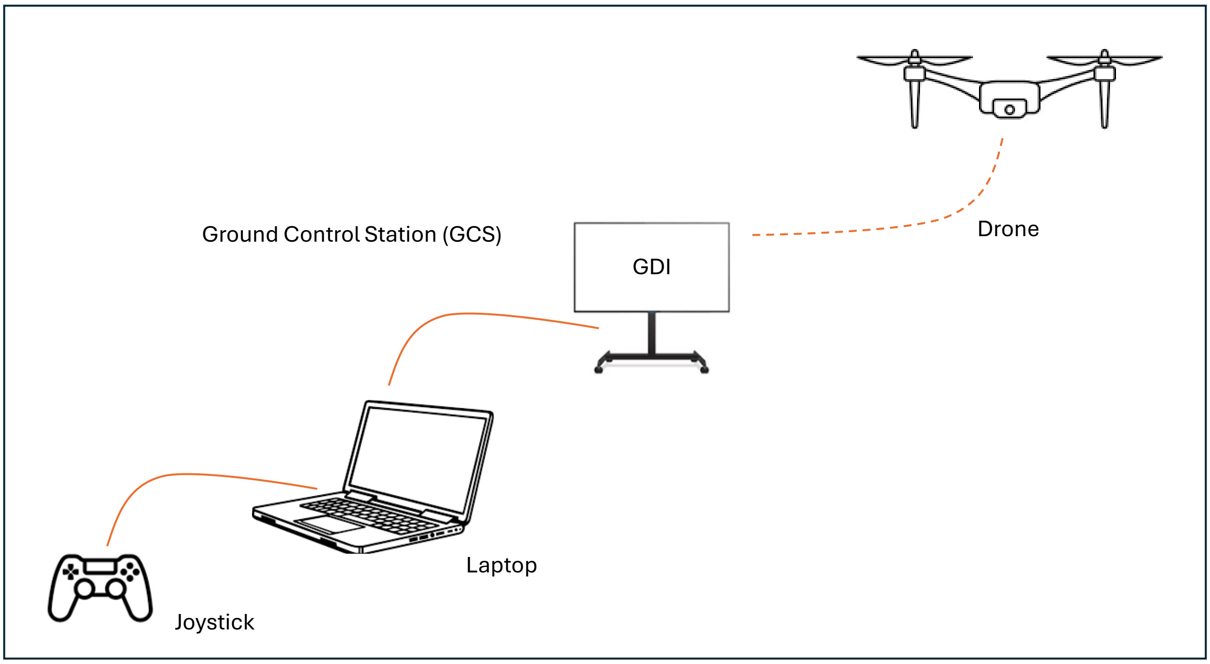
Bernoulli's principle is central to UAV aerodynamics, shaping how engineers design wings, rotors, and propellers to balance lift, thrust, drag, and weight. By manipulating airspeed and pressure, UAVs achieve efficient, controlled flight through real-time adjustment and advanced design.

UAVs generate lift using propellers powered by motors. The flight controller receives input from a remote operator or follows autonomous software routines. Data from sensors are constantly processed to adjust motor speeds via ESCs, keeping the UAV stable and oriented correctly. Navigation can be manual or automatic, guided by GPS and inertial sensors, enabling the UAV to execute complex missions such as surveying, mapping, delivery, or inspection.

In essence, UAVs integrate advanced hardware and software that allow for resilient flight, balanced stability, and autonomous or remote operation, enabling their effective deployment across diverse applications

What are the key components of Unmanned Aircraft Systems (UAS)?

An Unmanned Aircraft System (UAS) is a comprehensive setup consisting of several interconnected components that together enable the remote or autonomous operation of an Unmanned Aerial Vehicle (UAV), supporting its flight, control, data capture, and mission execution. This integration is vital for safety, reliability, and effective use in various civilian, commercial, and defense sectors.



Unmanned Aerial Vehicle (UAV):

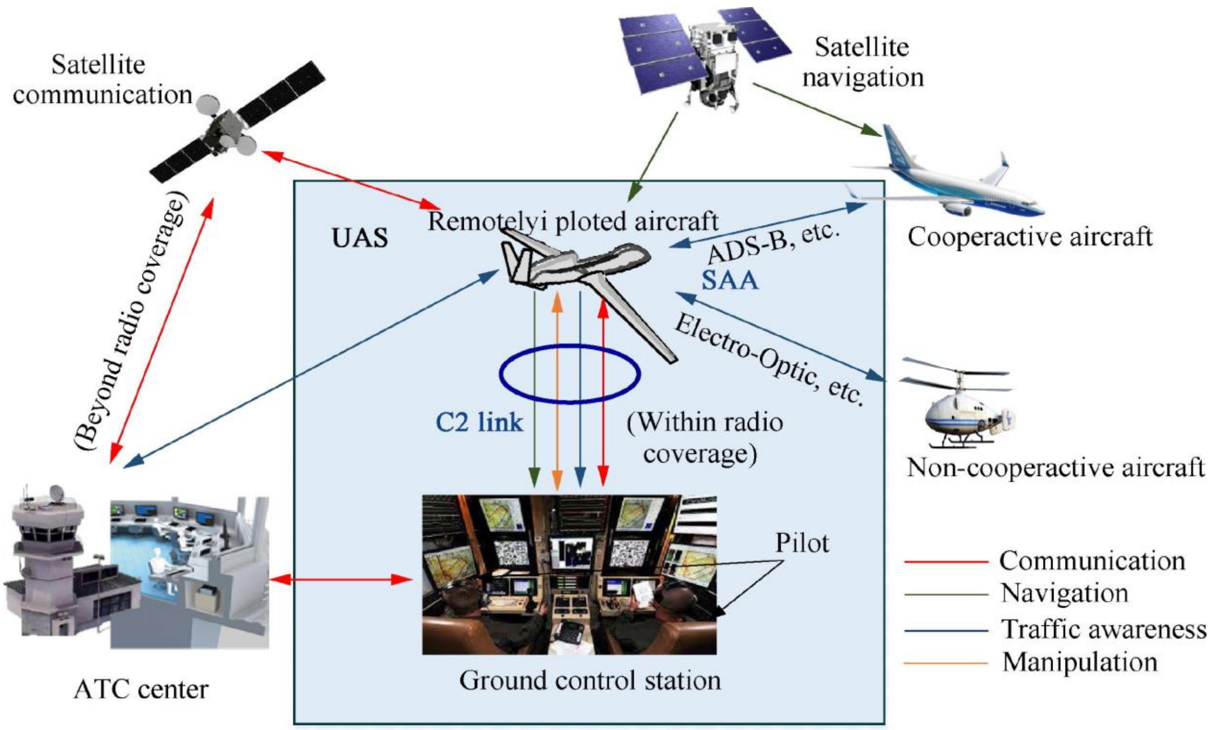
The flying platform itself, which may be remotely piloted or fly autonomously. UAVs can be fixed-wing, rotary-wing, or multi-rotor and typically consist of a frame, motors and propellers (for lift and maneuvering), flight control unit, sensors, payload(s), battery, and communication modules.

Ground Control Station (GCS):

The human or autonomous command/control center, usually stationed on the ground, from where operators send flight commands, receive telemetry, and monitor mission status. It features command interfaces, video screens, communication hardware, and software for planning and executing flights.

Communication Links (C2/C3 System):

The command-and-control system, often a radio-based data link, maintains real-time two-way connectivity between the UAV and GCS for flight instructions, telemetry feedback, video streams, and emergency overrides. These links are essential for safety, control, and timely data exchange.



How does an Unmanned Aircraft System (UAS) Work?

System Initialization and Pre-Flight Setup

The UAV is prepared by checking hardware integrity, charging batteries, configuring payloads (like cameras or sensors), and calibrating sensors and navigation aids. The Ground Control Station operator programs autonomous flight plans or manual flight paths into mission software, specifying waypoints, altitudes, speeds, and payload instructions.

Launch and Control

UAV takeoff is initiated either manually (with a joystick/controller), via computer interface, or autonomously using preprogrammed protocols.

During flight, commands are relayed from GCS to the UAV over secure communication links. The flight controller on the UAV processes these commands and manages onboard sensors to maintain stability, altitude, position, and orientation.

Navigation and Mission Execution

Onboard sensors (GPS, gyroscope, accelerometer, magnetometer, barometer, cameras) continuously collect data used for navigation, obstacle avoidance, and mission-specific tasks like surveying or imaging. The UAV adjusts its flight path in real time, autonomously or through manual commands. Some systems feature high degrees of autonomy for collision avoidance, return-to-home, or dynamic re-tasking in response to sensor inputs or operator decisions.

Data Capture, Processing, and Transfer

The UAV's payload (such as cameras or sensors) collects mission data (images, videos, measurements), often processed onboard or streamed directly to the Ground Control Station. Telemetry including position, altitude, battery status, and payload data is continuously sent back to the GCS for monitoring and decision-making.

Landing and Post-Flight Operations

Autonomous or remote landing protocols are executed, often with additional safeguards (like real-time monitoring, obstacle sensors, precision landing systems). Post-flight, the captured data is reviewed, analyzed, and stored, while UAV hardware is inspected for maintenance and readiness for future missions.

A UAS is distinguished by its seamless integration of airborne, ground-based, and communication elements—enabling real-time control, precise navigation, robust data acquisition, and adaptability for various applications ranging from surveillance to mapping and delivery.

Introduction to Detection, Recognition and Identification

Unmanned Aerial Vehicles (UAVs) or drones have revolutionized surveillance, search and rescue, environmental monitoring, agriculture, and industrial inspection by providing rapid data acquisition and scalable coverage of large areas. Equipped with high-resolution cameras and advanced sensors, UAVs can capture imagery in real time, enabling automated object and human detection systems for diverse applications such as security, traffic analysis, disaster response, and precision farming.

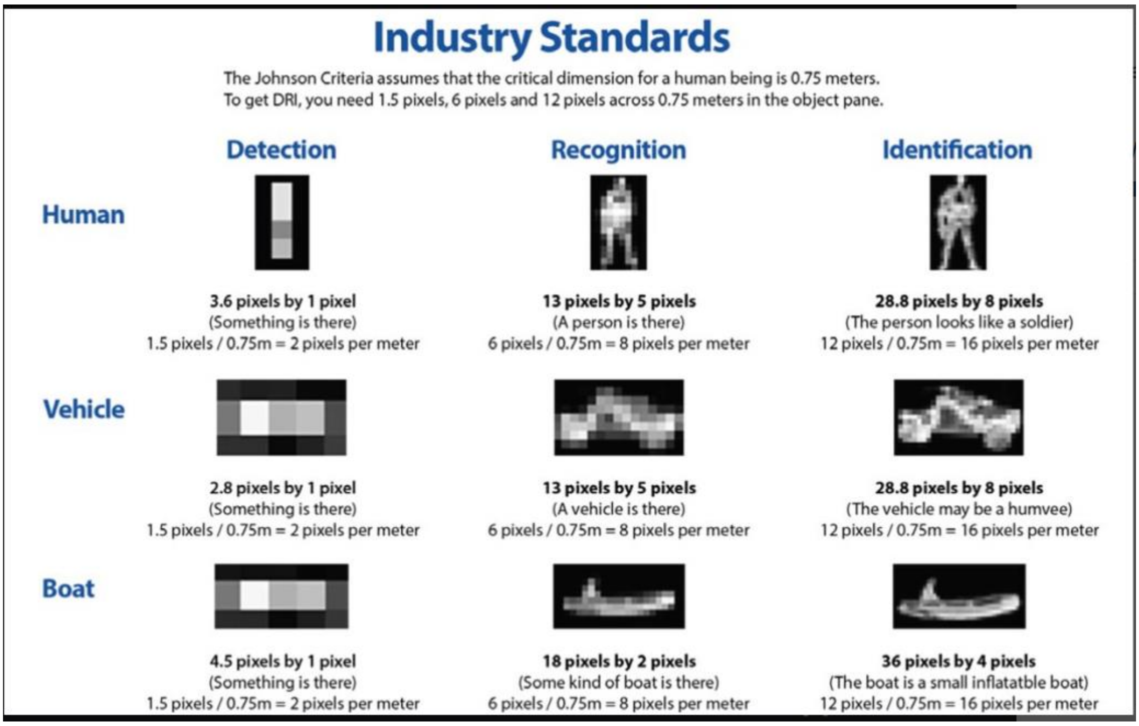
DRI: Concepts and Standards

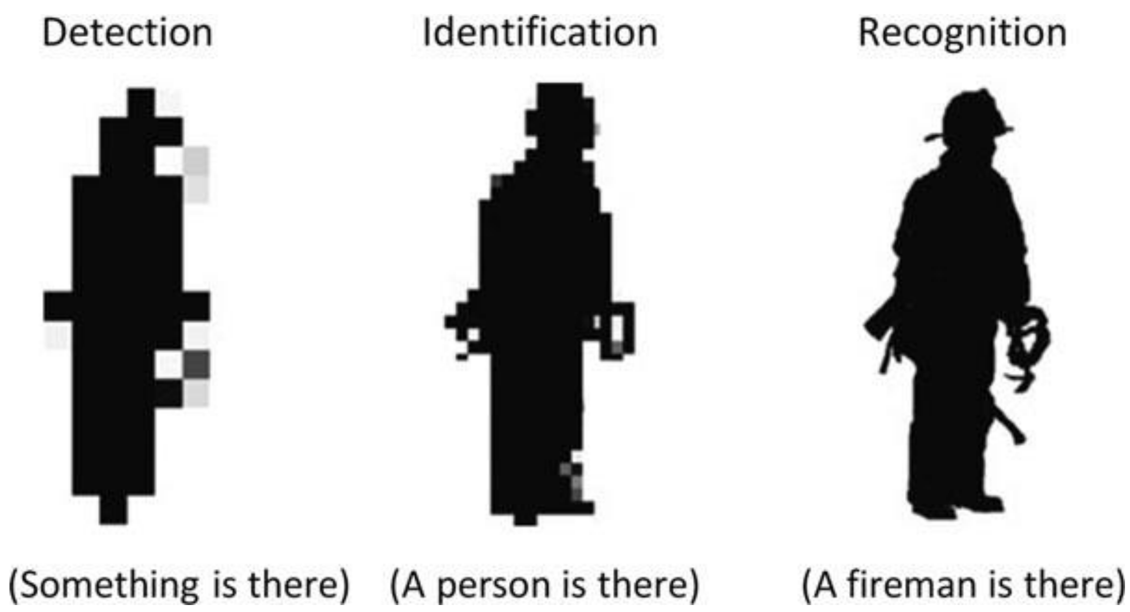
DRI represents three hierarchical levels of capability:

- Detection: The ability of a drone's sensor to discern that something is present at a given location—confirming an object exists, but not distinguishing its type.
- Recognition: The capacity to determine the broader class of the object (for example, whether it is a human, vehicle, or animal), though not its precise identity.
- Identification: The ability to distinguish specific details about the object, such as recognizing an individual person, vehicle make and model, or discerning a soldier from a civilian.

These standards are typically quantified using the Johnson Criteria. This model estimates the minimum number of pixels covering a target at a given distance required for each DRI level:

- Detection: 2 pixels across the target
- Recognition: 8 pixels across the target
- Identification: 12.8 pixels across the target





UAV Technologies for DRI

Imaging Sensors: Drones use electro-optical (daylight) and thermal cameras for DRI tasks. Thermal sensors, in particular, allow for detection at night or through smoke, but often at shorter identification ranges compared to high-resolution daylight cameras.

Artificial Intelligence and Deep Learning: Advanced computer vision and deep learning algorithms (such as YOLO, SSD, or CNNs) process live video streams and interpret images for object/person detection and tracking at various distances. AI-powered systems filter background noise, optimize real-time identification, and enable rapid decision-making during missions.

Sensor Fusion: Some UAVs combine data from multiple sensors (visual, infrared, acoustic) to improve detection accuracy, range, and operational performance—especially in challenging conditions like poor visibility or cluttered environments.

Detection, recognition, and identification ranges vary widely depending on the UAV platform, sensor type, and environment. For high-performance electro-optical sensor payloads:

- Detection of a person may be possible at distances up to 1 km to 2 km.
- Recognition typically occurs at roughly half the detection distance.
- Identification requires even closer proximity, often just a few hundred meters for small human targets.

Applications of DRI

Surveillance and Security: Drones continuously scan large areas for intruders, suspicious objects, or vehicles, providing 24/7 monitoring for critical infrastructure and public events. AI video analytics enable instant alerts and smart incident response.

Search and Rescue: UAVs equipped with thermal cameras and advanced sound detection algorithms can localize missing persons by their heat signature or auditory distress signals, even in difficult terrain or after disasters.

Crowd and Perimeter Monitoring: Real-time object/person tracking enables rapid identification of threats, crowd density, or unauthorized access in public and private spaces.

Limitations and Challenges of DRI

Environmental Conditions: Weather, lighting, and atmospheric effects can degrade sensor performance and reduce effective DRI ranges.

Sensor Calibration and Resolution: Higher pixel density and frequent calibration are critical for optimal identification accuracy.

Computational Resources: Real-time video processing and deep learning require significant on-board or cloud computing capabilities, presenting challenges for smaller UAV platforms.

Detection, Recognition, and Identification (DRI) form the backbone of intelligent drone surveillance and monitoring systems, combining powerful imaging sensors and cutting-edge AI for precise, actionable data about humans and objects in the environment.

Introduction to Validation and Verification

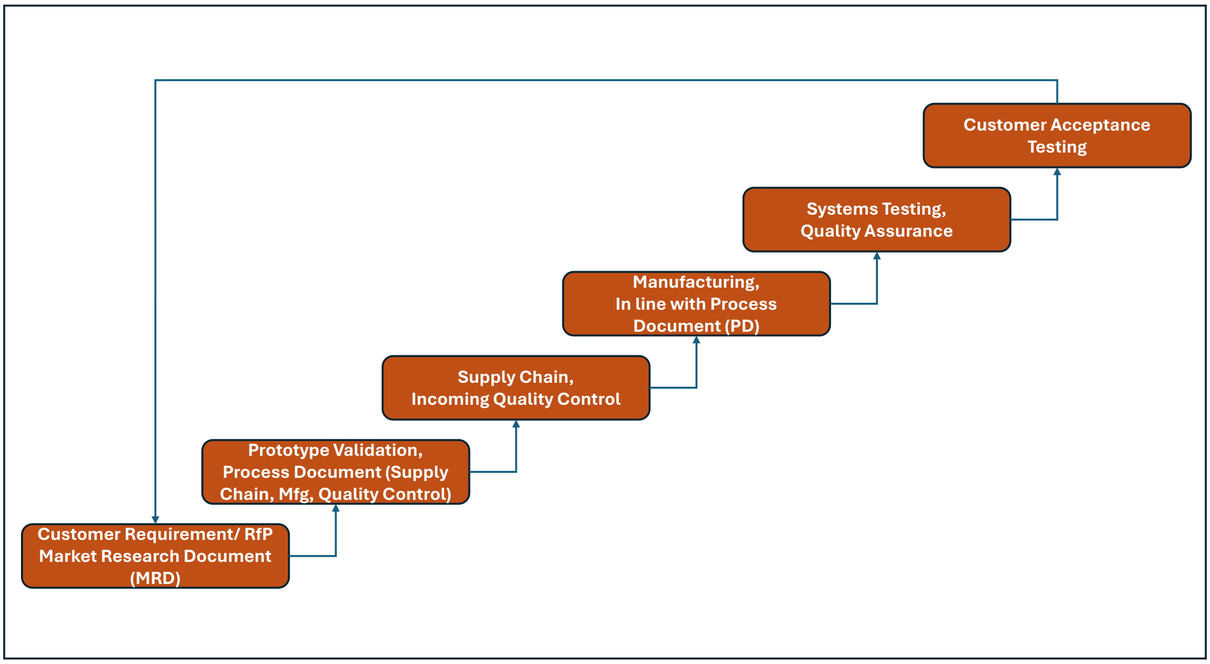
Verification: Drone Software, Systems and Components meets standards

Validation: The Drone is fulfilling all its tasks and meeting the requirements

The exhibit below summarizes the key aspects of Verification and Validation, for Drones and UAS.

	VERIFICATION	VALIDATION
What is …	• *Verification* confirms the drone adheres to its design specifications and standards. • Essentially, verification answers "Are we building the drone right?". • Verification is often performed during development and manufacturing	• *Validation* ensures the drone meets user needs and requirements • Validation answers "Are we building the right drone?" • Validation is often performed before and after deployment.
Areas of Focus	• **Conformance to design**: Ensuring the drone's components, software, and overall system meet the predefined design specifications. • **Compliance with standards**: Confirming the drone adheres to relevant industry and regulatory standards. • **Internal consistency**: Verifying that different parts of the drone work together as intended and that there are no inconsistencies in the design.	• **Meeting user needs**: Ensuring the drone effectively fulfills the tasks and requirements of its intended users. • **Real-world applicability**: Assessing if the drone performs reliably in the operational environment it is designed for. • **User satisfaction**: Gathering feedback from users to understand their experience with the drone and identify areas for improvement.
Tests Under-taken	• **Static tests**: Checking the functionality of individual components like motors, sensors, and communication systems. • **Software verification**: Testing the drone's software for bugs, performance issues, and adherence to coding standards. • **Structural integrity tests**: Ensuring the drone's frame can withstand expected stresses and strains.	• **Flight tests**: Evaluating the drone's stability, maneuverability, and performance in different conditions. • **Payload validation**: Testing the drone's ability to carry and deploy payloads for its intended purpose. • **Navigation and control validation**: Assessing the drone's ability to follow pre-programmed routes, avoid obstacles, and respond to user inputs. • **Environmental testing**: Validating the drone's performance in various weather conditions, temperature ranges, and altitudes.

Please find below a workflow summarizing activities from Customer Research to Customer Acceptance.



Market Development and Customer Requirement

Business Development articulates to a problem statement (in discussion with Customer) or responds to a Request for Proposal (RFP) from Customer, capture List of parameters

- Endurance (flight time)
- Environmental conditions for operation
- Technical Requirements
- GPS/ Camera Need

This requirement/ RfP goes from the Business Development team to the Product Manager. The Product Manager talks to the Systems Engineer, who identifies the Components suited to the Requirement/ RfP.

Market Research Document (Gathering what market needs)

- *Establishes what is customer need*
- *Captures what is available in the market document*
- *Compares what Asteria offers and what competition provides*
- *Identifies something that is innovative for Asteria*
- *Helps the customer realize the innovative need met*

The team then holds a DEMO and showcases the capabilities of the Drone solution. The bid is submitted.

Prototype Development and Process Definition

Once bid is awarded, Systems Engineer and R&D will build and test prototype, until Requirements are met. Product Manager and Systems Engineer will record Prototype parameters, that has passed testing criteria. This will be taken forward into Manufacturing stage.

Product Requirement Document (PRD)

- *Identifies how earlier customer needs/ opportunities were missed (if any)*
- *Establishes how the customer need is met (with current MRD as input)*

A Process Document (PD, step-by-step instruction guide for manufacturing) is prepared. During the preparation of the PD, Product Planning and Control (PPC) delegates manhours and fits process to the timeline.

Technical Requirement Document (TRD)

- *Establishes what technical specifications are required to achieve what is asked for in the PRD*

Supply Chain and Incoming Quality Control

PPC works with Supply Chain Management (SCM) to identify components & vendor sources and obtain order commitments for deliveries within PPC deadlines. Once components ordered are delivered in line with purchase orders place, Incoming Quality Control (IQC) tests every incoming component in line with standards and validation criteria.

Manufacturing

After clearance by IQC, components are taken into Stores for manufacturing. Before start of Manufacturing, the Process Document (PD) and Components/ Material all allocation to the Product Manufacturing Team (PMT). The PMT assembles the product stage-by-stage in line with the PD. Stage-wise inspections are conducted by Quality Control (QC).

Systems Testing and Quality Assurance

There are two inspections conducted for each system – Static (at desk) and Dynamic (at field). The product – after quality testing – goes to Quality Assurance (QA) for Verification. After Verification (ie Systems Testing) – the product goes to the Finished

Product Warehouse. Here, Pre-Delivery Inspection (PDI) and Joint Receipt Inspection (JRI) is undertaken by Customer Support and Quality teams. The Drone is delivered to the Customer after all inspection is completed and cleared.

Customer Acceptance

The Acceptance Test Procedure (ATP) team from Asteria trains the Customer for 2-3 days. The Customer undertakes two-step Acceptance Test Procedure (ATP) – one at premises (in-house) and second at site (field). After Customer completes ATP, Payment to Asteria is released. Asteria Customer Support then takes over to address any queries, provide engineering support and coordinate to undertake service maintenance and repairs. In doing so, the Asteria Engineering team analyses telemetry data and flight parameters. The Asteria Service team fixes mechanical problems (if any).

All points of validation match the TRD with the PRD.

Real-time computer vision on UAV platforms is essential for quick responses in dynamic environments, enabling applications such as live threat detection, immediate search and rescue, automated traffic management, and environmental monitoring. Validation and verification metrics are vital to ensure accuracy, reliability, and robustness of these automated systems under challenging conditions like motion blur, variable lighting, and small object sizes.

Using Artificial Intelligence and Machine Learning to improve the Accuracy of DRI

Measuring the Effectiveness of Object DRI by Drones

Key metrics for evaluating object detection models are Intersection over Union (IoU), Precision, Recall, and Mean Average Precision (mAP).

Intersection over Union (IoU) is A measure of overlap between the predicted bounding box and the actual (ground truth) bounding box. It's the ratio of the area of intersection between the two boxes to the area of their union. A perfect detection has an IoU of 1, while no overlap yields an IoU of 0. A threshold (e.g., 0.5) is set to determine if a prediction is a true positive.

Precision Measures the accuracy of the detector's positive predictions. It's calculated as True Positives / (True Positives + False Positives).

Recall Measures how effectively the detector finds all the actual positive objects. It's calculated as True Positives / (True Positives + False Negatives).

Average Precision (AP), For each class, is the area under the precision-recall curve, which plots precision against recall at various confidence thresholds.

Mean Average Precision (mAP) is A single metric that summarizes the performance of an object detection model across all object classes.

A higher mAP indicates better overall performance, meaning the model is both accurate in its detections (high precision) and has successfully found most of the objects (high recall).

F1 Score provides balanced assessment considering both precision and recall: $F1=2\times(Precision\times Recall)/(Precision+Recall)$. This metric proves particularly valuable for drone applications where both missed detections and false alarms carry significant consequences.

These metrics hold significance, owing to three considerations:

Localization: IoU assesses how well the model localizes objects.

Classification: Precision, recall, and the precision-recall curve evaluate the model's ability to correctly classify objects and distinguish them from backgrounds or other classes.

Comprehensive Evaluation: mAP provides a holistic view of the model's performance by averaging the results across different object classes and considering the trade-off between precision and recall

State-of-the-art computer vision frameworks—YOLO (You Only Look Once) and OpenCV—enable accurate, robust, and real-time analysis of aerial imagery.

Introduction to YOLO

YOLO (You Only Look Once) is a highly popular real-time object detection algorithm that analyzes an entire image in a single pass by dividing it into a grid and predicting bounding boxes and class probabilities for each grid cell. This grid-based approach with anchor boxes allows YOLO to efficiently detect multiple objects of various sizes and classes within one frame, making it well-suited for fast applications like UAVs (drones). YOLO trades some localization accuracy for significantly faster inference speeds compared to traditional region proposal methods, enabling real-time detection.

In UAV applications, YOLO is extensively used for object detection, recognition, and identification due to its speed and generalization. However, detecting small objects from UAV imagery remains challenging because of small object sizes and complex backgrounds typical in aerial views. To address this, many enhanced versions of YOLO have been developed for UAV-specific tasks, such as SOD-YOLO and GL-YOMO, which improve small object detection by adding attentional mechanisms, multi-scale feature fusion, and motion analysis. For example, GL-YOMO combines YOLO detection with multi-frame motion detection for higher accuracy in long-distance UAV detection scenarios.

Recent UAV-specific YOLO models also incorporate improvements like small object detection layers, Soft-NMS to refine confidence, lightweight modules for faster edge computing, and GhostHead networks to boost both accuracy and inference speed. These adaptations help YOLO overcome challenges unique to UAV imagery such as small target size, occlusions, and varying lighting, leading to better precision, recall, and mean Average Precision (mAP) scores compared to baseline YOLO models.

Introduction to OpenCV

OpenCV is a widely used open-source library for computer vision and image processing, and it plays a crucial role in enabling object detection, recognition, and identification for UAV (Unmanned Aerial Vehicle) applications. It offers both traditional image processing techniques as well as interfaces for integrating modern deep learning object detectors—making it highly suitable for real-time UAV deployments.

OpenCV (Open Source Computer Vision Library) provides a comprehensive toolkit for image and video manipulation, including algorithms for feature extraction, object detection, image recognition, and image segmentation. It supports multiple languages

(C++, Python, etc.) and is optimized for efficiency, which is critical for deployment on resource-constrained UAV platforms.

Object Detection with OpenCV

OpenCV enables object detection through both classic methods (such as Haar Cascades, HOG with SVM) and deep learning approaches (such as YOLO, SSD, and others).

Haar Cascades work by classifying regions in grayscale images using pre-trained classifiers—best for faces and simpler tasks.

HOG (Histogram of Oriented Gradients) combined with linear SVM is well-suited for pedestrian detection and uses edge-based features.

For more robust and modern requirements, OpenCV interfaces seamlessly with deep learning models, e.g., YOLO, which detect and classify multiple objects in real time—making it ideal for UAV use where speed and accuracy are needed.

UAV Use Cases: Detection, Recognition, and Identification

On UAVs, OpenCV is used in several core functions:

Object Detection: Identifies and localizes objects in aerial frames or video streams—these might be drones, vehicles, people, or wildlife. Techniques include contour detection, background subtraction for moving objects, and bounding-box generation.

Recognition: Once potential objects are detected, recognition leverages either classic image descriptors or deep neural networks to classify and label those objects. Models can be pre-trained on public datasets or custom-trained for UAV-specific tasks (search and rescue, surveillance, monitoring).

Identification: This involves verifying, tracking, or more narrowly identifying objects (such as distinguishing different vehicles or species). OpenCV aids in generating unique signatures or connecting with more advanced algorithms for target identification

Application of YOLO and OpenCV in Object DRI

System Architecture

The typical pipeline for drone-based detection systems consists of:

- Data Acquisition (UAV captures high-resolution aerial images/video)
- Preprocessing (noise reduction, frame extraction, enhancement)
- Detection (YOLO model for bounding box prediction)
- Recognition & Identification (class probability, object labeling)
- Validation & Post-processing (OpenCV cropping, confidence filtering, tracking)

Each stage plays a vital role in the UAV's autonomous perception and decision-making.

Data Acquisition and Preprocessing

- Data Acquisition: UAVs are equipped with RGB or multispectral cameras delivering image streams at high altitudes or wide angles.
- Preprocessing: OpenCV routines such as Gaussian blur (noise reduction), frame extraction from videos, resizing, and normalization are applied to enhance data quality and reduce computational overhead.
- Background Removal: Advanced image segmentation (quick-shift or thresholding) may be used to isolate foreground objects/humans prior to detection.

Detection using YOLO

- YOLO Algorithm:
    - YOLOv8/v9 models process images in a single computational pass, outputting bounding boxes, confidence scores, and class probabilities for objects (e.g., human, vehicle).
    - The architecture predicts, for each grid cell, the bounding box coordinates (x,y,w,hx,y,w,h), objectness score P(object)$P(object)$, and class probabilities ([P(class1),P(class2),…]$[P(class1),P(class2),\ldots]$).
    - Non-Max Suppression (NMS): Filters overlapping boxes to ensure detection uniqueness.
- YOLO in Aerial Settings:
    - Models are fine-tuned for aerial image datasets to address altitude, angle, and scale variance.

    - Specific class weights are adjusted to prioritize 'human' or 'vehicle' detection, improving precision for target objects.

Recognition and Identification

- Recognition: Highest class probability from YOLO’s output is assigned for each valid bounding box (e.g., person, vehicle).
- Identification:
    - Extracted detection coordinates enable region cropping via OpenCV.
    - Additional feature extraction—such as angle computation between detected joints, distance between landmarks, or 3D shape estimation—is implemented for human action classification or object type confirmation.
    - Deep learning classifiers (CNN, Kernel Discriminant Analysis) can be stacked for refined recognition and individual identification.

Validation and Post-Processing

- OpenCV-Based Validation:
    - Cropping: Detected bounding boxes are indexed and cropped using OpenCV for focused analysis or validation.
    - Confidence Filtering: Only detections above a set confidence threshold are retained for validation.
    - Rule-Based Tracking: Multi-frame tracking with cross-correlation improves detection persistence and reduces drops due to occlusion or confidence fluctuation.
        - Bounding boxes are preserved from prior frames and re-evaluated against new detections using intersection-over-union (IOU).
        - Confidence scores are dynamically updated for persistent tracking, enabling prediction during missed detections.
- Performance Metrics:
    - Validation workflows calculate precision, recall, F1-score by matching detected bounding boxes with ground truth annotations.
    - Latency, frame rate (e.g., 15–30 fps), and computational throughput metrics are reported, ensuring UAV deployments meet real-time operational requirements.

YOLO Integration with OpenCV

Modern drone-based detection systems leverage the synergy between YOLO's real-time detection capabilities and OpenCV's computer vision processing power. The YOLO-8 model achieves remarkable performance with 91% accuracy in person detection from drone imagery, processing 900×900 pixel resolution images with runtime speeds of 238.41 milliseconds. This integration utilizes OpenCV's image slicing functionality to extract bounding box coordinates and crop specific regions of drone images, enabling precise isolation of detected persons or vehicles.

The validation pipeline incorporates YOLOv8's single-stage architecture with Cross-Stage Partial Networks (CSP) and Path Aggregation Network (PANet) components, optimized for UAV systems operating in dynamic environments. OpenCV handles the preprocessing and post-processing stages, including confidence value filtering and Non-Maximum Suppression (NMS) to eliminate duplicate detections.

Enhanced Models for Indoor Detection

Specialized models like YOLO-IHD demonstrate superior performance in complex indoor environments, achieving 77.71% mAP@0.5 with 78.83% precision and 71.60% recall for human detection. The model incorporates optimized convolutional layers and attention mechanisms specifically designed to process complex visual data from indoor UAV operations, making it highly reliable for disaster response and indoor rescue missions.

Real-Time Processing Hardware Platforms

Nvidia Jetson (Nano, Xavier NX): Highly popular for drone applications due to onboard GPU acceleration, these devices process YOLOv5, YOLOv8 Nano, and custom lightweight models at frame rates (FPS) suitable for real-time tasks (often 15–30+ FPS with Nano and 35+ FPS with Xavier NX).

Raspberry Pi 4: Capable of running optimized YOLO models with reasonable real-time performance by using reduced input image size and lightweight YOLO variants for live drone video streams.

Intel Movidius/FPGA/VPU: For ultra-lightweight inference, OpenVINO allows model acceleration on Intel platforms, which is useful for specialized security or autonomous navigation drones.

Workflow for YOLO and OpenCV in UAV-based Object DRI

Please find below an illustrative workflow for integrating YOLO and OpenCV toolsets towards UAV-based validation of objects

1. **Data Collection and Preparation**
   - Collect diverse aerial imagery and video data from drones/UAVs under varying conditions.
   - Annotate objects and humans in the frames with bounding boxes and class labels.
   - Split the data into training, validation, and test sets (e.g., 70%/20%/10%).
2. **Model Training with YOLO**
   - Train a YOLO model (e.g., YOLOv8) on the training dataset.
   - Use data augmentation to enhance model robustness (e.g., scaling, rotation, illumination changes).
   - Save model checkpoints for validation testing.
3. **Inference and Prediction Extraction**
   - Use YOLO for real-time object/human detection on the validation dataset.
   - Extract bounding boxes, confidence scores, and class predictions.
   - Apply Non-Maximum Suppression (NMS) to remove duplicate detections.
4. **Preprocessing with OpenCV**
   - Use OpenCV to preprocess images as needed (resize, color conversion).
   - Crop detected bounding box regions for detailed analysis or recognition tasks.
   - Generate image slices for multi-scale evaluation if necessary.
5. **Validation Metric Calculation**
   - Calculate Intersection over Union (IoU) between predicted and ground-truth bounding boxes using OpenCV functions.
   - Compute validation metrics:
     - Precision, Recall
     - Mean Average Precision (mAP)
     - F1 Score

        - Confusion Matrix per class
    - Visualize Precision-Recall curves and F1 score trends.

6. **Threshold Optimization**
    - Evaluate model performance at multiple confidence and IoU thresholds.
    - Select optimal thresholds balancing false positives and false negatives.

7. **Performance Analysis**
    - Analyze class-wise metric distributions to identify strengths and weaknesses.
    - Perform error analysis: False positives, false negatives, and localization errors.
    - Use OpenCV to visualize detections on images with bounding boxes, labels, and confidence scores.

8. **Batch Validation and Reporting**
    - Automate batch processing of the entire validation dataset using OpenCV and YOLO.
    - Generate comprehensive validation reports with quantitative metrics and qualitative visual results.
    - Save intermediate results for iterative model tuning.

9. **Real-Time Validation (Optional)**
    - Deploy the YOLO model on an embedded system.
    - Use OpenCV video capture to validate real-time inference performance (FPS, latency).
    - Monitor detection accuracy live on drone video feeds.

10. **Continuous Validation**
    - Integrate a pipeline to periodically validate new data.
    - Update model and thresholds if performance degrades.

This workflow ensures a systematic, rigorous, and scalable validation process for drone vision systems using YOLO and OpenCV integration.

Assessment and Conclusion

YOLO and OpenCV together improve the effectiveness of detecting, recognizing, and imaging humans versus objects by drones and UAVs in several key ways:

1. Real-Time High-Accuracy Detection by YOLO

YOLO models, especially versions like YOLOv8, are designed for fast, real-time detection with high accuracy. YOLO's single-pass detection architecture enables precise localization and classification of objects and humans simultaneously in drone images or video frames. For human detection, YOLOv8 achieves up to 91% accuracy in drone imagery, outperforming other models like Faster R-CNN or YOLOv5 in both precision and speed. This is crucial for drones operating at varying altitudes and speeds where quick and robust human detection is needed.

2. Effective Differentiation Between Humans and Objects

YOLO models provide strong discrimination capabilities between humans and other objects due to their deep convolutional layers trained on large and diverse aerial datasets. For drones, which capture images from high altitudes and varying angles, YOLO adapts well to detecting scaled and partially occluded humans, maintaining higher mean average precision (mAP) for people compared to many objects. OpenCV assists by preprocessing images (e.g., resizing, noise reduction) and postprocessing YOLO outputs (like applying Non-Maximum Suppression) to refine detections.

3. Enhanced Image Processing and Feature Extraction by OpenCV

OpenCV complements YOLO by handling image enhancement, noise filtering, and region cropping, crucial for improving recognition accuracy post-detection. It allows drones to isolate detected humans and objects for further recognition or tracking, applying computer vision algorithms like background subtraction, segmentation, and feature extraction, which are vital in complex aerial scenes.

4. Robustness Across Environmental Conditions

Together, YOLO and OpenCV help drones maintain effective detection under challenging conditions such as varying altitudes, speeds, lighting, and backgrounds. Studies show that drone altitude and object size affect detection performance; YOLO's architecture combined with OpenCV's image processing improves robustness to these variables, making human detection consistently reliable compared to other object classes which might have more variability in shape or texture.

5. Superior Speed-Accuracy Trade-Off

YOLO models excel in runtime efficiency allowing drones to perform human and object detection with low latency, critical for real-time applications like surveillance, search and rescue, and monitoring. OpenCV's optimized functions support this by efficiently managing image input/output and data manipulation. Compared to more complex, slower models like Faster R-CNN, YOLO integrated with OpenCV provides a practical balance suitable for embedded drone systems.

6. Specialized Deep Learning and Feature Extraction for Human Recognition

In addition to detection, advances integrate YOLO with deep-learning-based human feature extraction pipelines using OpenCV, such as extracting pose landmarks or 3D point clouds for detailed human recognition and action classification in aerial videos. This is less common for general object detection, marking a critical enhancement for human-focused UAV applications.

Summary Table

Model/Platform	FPS (typical)	Edge Suitability	Optimization Methods	Application
YOLOv8 Nano	35+ (Xavier), 15+ (Nano)	Excellent	Pruning, Quantization	Real-time visual navigation, inspection, surveillance
LEAF-YOLO	30+	Excellent	Lightweight modules	Small object UAV search and rescue
YOLO11-S Optimized	30+	Very Good	Reduced width/depth	Maritime rescue, fast UAV detection in dense scenes
Raspberry Pi 4	8–20	Good	Input size reduction	Budget surveillance, basic tracking
Jetson Nano/Xavier	15–35+	Excellent	Hardware acceleration	Mapping, SAR missions, traffic monitoring
OpenVINO (Intel)	10–30	Good	INT8 quantization	Security, autonomous navigation in restricted environments

Together, YOLO and OpenCV significantly enhance drones' effectiveness in detecting, recognizing, and imaging humans as compared to other objects, enabling high precision, real-time functionality crucial for many aerial applications.

References

1. https://www.irjms.com/wp-content/uploads/2025/07/Manuscript_IRJMS_04370_WS.pdf
2. https://jowua.com/wp-content/uploads/2025/04/2025.I1.011.pdf
3. https://www.nature.com/articles/s41598-024-75807-1
4. https://www.sciencedirect.com/science/article/abs/pii/S1051200424005888
5. https://www.ewadirect.com/proceedings/ace/article/view/27858
6. https://par.nsf.gov/servlets/purl/10394513
7. https://keylabs.ai/blog/best-practices-for-image-preprocessing-in-image-classification/
8. https://www.frontiersin.org/journals/physics/articles/10.3389/fphy.2025.1553224/full
9. https://pmc.ncbi.nlm.nih.gov/articles/PMC12526591/
10. https://labelyourdata.com/articles/object-detection-metrics
11. https://www.mecs-press.org/ijigsp/ijigsp-v16-n2/IJIGSP-V16-N2-7.pdf
12. https://pmc.ncbi.nlm.nih.gov/articles/PMC10857234/
13. https://opencv.org/blog/opencv-applications-in-2023/
14. https://arxiv.org/html/2502.05292v1
15. https://github.com/Ayushkumawat/Advanced-Aerial-Drone-Detection-System
16. https://www.sciencedirect.com/science/article/pii/S030147972202521X
17. https://ieeexplore.ieee.org/iel8/6287639/10820123/10755076.pdf
18. https://ieeexplore.ieee.org/document/6215657/
19. https://ieeexplore.ieee.org/document/10840022/
20. https://www.tandfonline.com/doi/full/10.1080/22797254.2025.2507744
21. https://www.tandfonline.com/doi/full/10.1080/10106049.2024.2339294
22. https://learning-gate.com/index.php/2576-8484/article/view/1737
23. https://www.sciencedirect.com/science/article/pii/S2542660525002215

24. https://ieeexplore.ieee.org/document/10227141

25. https://www.nature.com/articles/s41598-024-84747-9

26. https://blog.roboflow.com/image-processing-with-opencv/

27. https://www.nature.com/articles/s41598-025-19145-w

28. https://opencv.org/blog/image-classification/

29. https://www.sciencedirect.com/science/article/abs/pii/S0262885624003676

30. https://www.geeksforgeeks.org/computer-vision/opencv-projects-ideas-for-beginners/

31. https://dl.acm.org/doi/10.1145/3757110.3757179

32. https://www.sciencedirect.com/science/article/pii/S1574954125003152

33. https://www.nature.com/articles/s41598-025-16878-6