Drone Software: A Comprehensive Survey

Soumik Das

Abstract

Drones or Unmanned Aerial Vehicles (UAVs), have quickly become essential tools across industries from farming and construction to logistics, research, and public safety. The real power of these systems comes from their software, which controls everything from basic flight stability to advanced autonomous missions. This report explores some of the most widely used UAV software platforms, including PX4, ArduPilot, DroneKit, DroneUp, Skydio Autonomy, DJI Terra, and PIX4D. For each solution, we outline its main features, supported parameters, and the types of applications it enables. This survey is designed to give developers, researchers, and enthusiasts a clear, comparative view of today's UAV software landscape, serving as a guide for future innovation in intelligent aerial systems.

Keywords: UAV, Drone software, ArduPilot, PX4, DroneKit

1 Introduction

Unmanned Aerial Vehicles (UAVs), or drones, have rapidly emerged as a key enabler in diverse sectors such as agriculture, emergency response, logistics, infrastructure monitoring, and urban air mobility [1]. Compared to traditional ground or aerial vehicles, UAVs offer flexibility, cost-effectiveness, and reduced environmental impact, making them a viable solution for a wide variety of applications[2]. Based on the operator's ability to maintain visual contact, drone operations are divided into three categories, namely Visual Line of Sight (VLoS), Extended Visual Line of Sight (EVLoS), and Beyond Visual Line of Sight (BVLoS) [3]. For controlling any drone, software is required that is responsible for its basic functionalities like navigation, mission planning, vehicle state position, etc. In this report, I take a closer look at some of the most popular UAV software platforms PX4[1], ArduPilot[2], DroneKit[3], DroneUp[4], Skydio Autonomy[5], DJI Terra[6] and PIX4D[7].

[1] PX4- `https://docs.px4.io/main/en/`
[2] ArduPilot-`https://ardupilot.org/`
[3] DroneKit-`https://dronekit.io/`
[4] DroneUp-`https://www.droneup.com/`
[5] Skydio Autonomy-`https://www.skydio.com/skydio-autonomy`
[6] DJI Terra-`https://enterprise.dji.com/dji-terra`
[7] PIX4D-`https://www.pix4d.com/`

2 Literature Survey

This section mainly deals with the key features and parameters used for each of the drone's software, which are mentioned in detail below:

2.1 PX4 Autopilot:

PX4 Autopilot is an open-source flight control software for drones, VTOL (Vertical Takeoff and Landing) aircraft, and other unmanned vehicles. It provides advanced features for autonomous flight, including waypoint navigation, mission planning, and various flight modes. PX4 is designed to run on a variety of hardware platforms and is widely used in both academic and commercial applications.

2.1.1 Features:

Modular architecture: PX4 is built with a modular architecture, allowing developers to easily extend and customize its functionality. This makes it suitable for a wide range of applications, from small hobby drones to large industrial UAVs.

Supported hardware: PX4 supports a wide range of hardware platforms, including popular flight controllers like Pixhawk and various other custom hardware designs. It also supports different sensors such as GPS, accelerometers, gyroscopes, and magnetometers.

Flight modes: PX4 supports various flight modes, including manual, stabilized, altitude hold, position hold, mission, and return to launch (RTL). These modes allow users to perform different types of flight operations, from manual piloting to fully autonomous missions.

Mission planning: PX4 includes tools for mission planning and autonomous navigation. Users can define waypoints, set flight paths, and specify actions to be performed at each waypoint, such as taking pictures or triggering other payloads.

Safety: PX4 incorporates several safety features to ensure reliable and safe operation of UAVs. This includes fail-safe mechanisms such as return-to-home in case of signal loss, geofencing to prevent drones from flying into restricted areas, and battery monitoring to manage power consumption.

Developer-friendly: PX4 is open-source and has an active community of developers contributing to its development. It provides extensive documentation, tutorials, and APIs for developers to build custom applications and integrate additional features.

2.1.2 Parameters:

Altitude Control: MC_ROLL_P, MC_PITCH_P, MC_YAW_P: Proportional gains for roll, pitch, and yaw control.
MC_ROLLRATE_P, MC_PITCHRATE_P, MC_YAWRATE_P: Proportional gains for roll, pitch, and yaw rate control.
MC_ROLLRATE_D, MC_PITCHRATE_D, MC_YAWRATE_D: Derivative gains for roll, pitch, and yaw rate control.

Position Control: MPC_XY_P, MPC_Z_P: Proportional gains for position control in the horizontal and vertical axes.
MPC_XY_VEL_P, MPC_Z_VEL_P: Proportional gains for velocity control in the horizontal and vertical axes.
MPC_XY_CRUISE, MPC_Z_VEL_MAX: Maximum velocity limits in the horizontal and vertical axes.
MPC_XY_JERK_MAX, MPC_Z_JERK_MAX: Maximum jerk limits in the horizontal and vertical axes.

Velocity Control: MC_ROLLRATE_P, MC_ROLLRATE_I, MC_ROLLRATE_D: Roll rate PID gains for velocity control.
MC_PITCHRATE_P, MC_PITCHRATE_I, MC_PITCHRATE_D: Pitch rate PID gains for velocity control.
MC_YAWRATE_P, MC_YAWRATE_I, MC_YAWRATE_D: Yaw rate PID gains for velocity control.

Flight Modes: NAV_FW_ALT_RAD: Altitude in fixed-wing mode.
MC_POS_MODE: Position control mode.

Autonomy: COM_OBS_AVOID: Obstacle avoidance mode.
NAV_DLL_ACT: Action for Downlink Loss (DLL) failsafe.

PID Controller: MC_PITCHRATE_P, MC_PITCHRATE_I, MC_PITCHRATE_D: Pitch rate PID gains.
MC_ROLLRATE_P, MC_ROLLRATE_I, MC_ROLLRATE_D: Roll rate PID gains.
MC_YAWRATE_P, MC_YAWRATE_I, MC_YAWRATE_D: Yaw rate PID gains.

Rate Controller: MC_RAT_RLL_FILT, MC_RAT_YAW_FILT: Cutoff frequency of the rate controller filters for roll, pitch, and yaw axes.

Battery: BAT_N_CELLS: Number of cells in the battery.
BAT_V_EMPTY, BAT_V_CHARGED: Voltage thresholds for empty and fully charged battery.

BAT_CAPACITY: Battery capacity in ampere-hours (Ah).
BAT_LOW_THR, BAT_CRIT_THR: Voltage thresholds for low battery and critical battery levels.

Safety: RTL_RETURN_ALT: Altitude at which Return-to-Launch (RTL) mode begins.
NAV_RCL_ACT: Action taken if radio control is lost (e.g., RTL, land).
COM_LOW_BAT_ACT: Action taken if battery is critically low (e.g., RTL, land).
GF_ACTION: Action taken if geofence is breached (e.g., RTL, land).

Navigation: NAV_DLL_ACT: Determines the action to take when a waypoint is reached during a mission.
NAV_DLL_ACT_DELAY: Delay before executing the action specified by NAV_DLL_ACT.
NAV_DLL_ALT: Altitude used for waypoint reach detection.
NAV_DLL_ALT_MODE: Altitude mode used for waypoint reach detection.
NAV_DLL_DIS: Distance from waypoint considered as reached.

GPS and Sensor: GPS_*: Parameters related to GPS configuration and accuracy.
SENS_*: Parameters related to sensor calibration and fusion (e.g., accelerometers, gyroscopes, magnetometers).

Geospatial Data: Geospatial data in PX4 serves as the backbone for autonomous navigation, mission planning, and operational safety. Through GPS positioning, PX4 accurately determines the vehicle's location, velocity, and altitude, facilitating precise navigation along predefined flight paths or waypoints. Geospatial information is instrumental in mission planning, where users define waypoints and geofences to ensure compliance with regulatory requirements and operational constraints.

Mission planning: Mission planning in PX4 involves using ground control station (GCS) software like QGroundControl to define autonomous flight missions. Users specify waypoints on a map interface, setting parameters such as altitude, speed, and actions to be performed at each waypoint.

Environment Perception: Environment perception in PX4 enables unmanned vehicles to understand and interact with their surroundings autonomously. Utilizing a suite of sensors including cameras, LiDAR, ultrasonic sensors, and range finders, PX4 processes data to create a 3D map of the environment, facilitating obstacle avoidance and collision prevention.

Security and Privacy: Security and privacy are paramount in PX4 Autopilot, ensuring the integrity and safety of unmanned vehicle operations. Through rigorous code review, authentication mechanisms, and encrypted communication, PX4 safeguards against unauthorized access and manipulation of critical systems and telemetry data. Firmware signing and verification prevent the installation of tampered firmware, while data minimization and anonymization protect user privacy.

Miscellaneous: SYS_AUTOSTART: Determines whether the system automatically starts the main application on boot.
SYS_MC_EST_GROUP: Selects the attitude estimation algorithm (e.g., EKF2, LPE).

2.2 Ardu Pilot:

ArduPilot is an open-source autopilot software suite designed for unmanned aerial vehicles (UAVs). Developed by a global community of volunteers, ArduPilot offers a comprehensive set of features for controlling drones, enabling autonomous flight, mission planning, and customization. It is widely used by hobbyists, researchers, and professionals for a variety of applications, including aerial photography, surveying, mapping, and research.

2.2.1 Features:

Autonomous Flight: ArduPilot enables drones to fly autonomously, following predefined missions or waypoints. Users can plan complex missions with waypoints, altitude changes, and specific actions.

Flight Modes: It supports a variety of flight modes catering to different needs, such as stabilize mode for manual control, altitude hold, loiter, guided mode for GPS-based navigation, auto mode for fully autonomous missions, RTL (Return to Launch), and more.

Mission Planning: Users can plan missions using ground control station (GCS) software like Mission Planner or QGroundControl. Mission planning includes defining waypoints, actions, and parameters for the drone to execute during flight.

Telemetry and Ground Control: ArduPilot communicates with the ground control station over telemetry links, providing real-time flight data and enabling remote control and monitoring of the drone. This allows operators to monitor flight status, receive telemetry data, and adjust parameters during flight.

Hardware Compatibility: ArduPilot is compatible with a wide range of flight controller hardware, including Pixhawk, Cube, and others. This flexibility allows users to choose hardware that fits their specific needs while still leveraging the capabilities of ArduPilot.

Sensor Integration: It supports integration with various sensors and peripherals, including GPS, magnetometers, accelerometers, gyroscopes, rangefinders, and more. This enables precise navigation, altitude control, and stability.

Customization and Extensibility: Being open-source, ArduPilot allows users to modify and customize the codebase to suit their specific requirements. Users can add new features, integrate additional sensors or peripherals, or tailor the software to specific use cases.

Community Support: ArduPilot has a large and active community of developers and users who provide support, documentation, and contribute to the ongoing development of the software. The community forums and wikis offer resources for troubleshooting, sharing knowledge, and collaborating on projects.

2.2.2 Parameters:

Basic Configuration: SYSID_THISMAV: System ID for the vehicle.
SERIALx_PROTOCOL: Communication protocol for telemetry.
BRD_PWM_COUNT: Number of PWM outputs.
FRAME_CLASS: Type of vehicle frame (e.g., quadcopter, hexacopter, plane, rover).

Flight Modes: FLTMODE1-6: Defines the flight modes available to the user.
MODE_CHANNEL: RC channel used for switching flight modes.

Vehicle State: The vehicle state typically includes data such as GPS position, altitude (roll, pitch, yaw), velocity, battery voltage, current battery capacity, and remaining battery capacity, as well as sensor readings such as IMU (Inertial Measurement Unit), barometer, and GPS health. ArduPilot also supports additional sensor data, such as rangefinder readings for terrain following or altitude control, and optical flow sensor data for position estimation.

PID Tuning: PID parameters for various control loops such as roll, pitch, yaw, altitude, and position control.

Telemetry and Communication: SERIALx_BAUD: Baud rate for telemetry communication.
SRx_ parameters: Defines the rate at which telemetry data is sent.

Mission Planning: In ArduPilot, mission planning empowers users to create autonomous flight missions through ground control software like Mission Planner or QGroundControl. Utilizing intuitive interfaces, users define waypoints on a map, specifying parameters such as altitude, speed, and actions to be executed at each waypoint. Once the mission is configured, it's uploaded to the vehicle's autopilot system.

GPS and Navigation: GPS_TYPE, GPS_AUTO_CONFIG: GPS parameters.
WPNAV: waypoint navigation

Battery and Power Management: BATT_CAPACITY: Battery capacity.
BATT_VOLT_MULT: Voltage multiplier for battery voltage sensing.

Safety and Fail-Safes: FS parameters for configuring various fail-safe behaviors.
ARMING_CHECK: Specifies conditions that must be met before arming the vehicle.

Sensor Calibration: Compass calibration parameters (COMPASS_).
Accelerometer and gyroscope calibration parameters.

Logging and Data Storage: LOG_BITMASK: Selects which data to log.
LOG_DISARMED: Controls whether logging occurs when the vehicle is disarmed.

Advanced Settings: AUTOTUNE_AXES: Specifies which axes to tune automatically.
AHRS: altitude estimation.

2.3 DroneKit:

DroneKit is an open-source software development kit (SDK) designed for building applications that interact with drones. Developed by 3DR (3D Robotics), DroneKit provides a set of APIs (Application Programming Interfaces) and tools for communicating with drones and controlling them autonomously or semi-autonomously.

2.3.1 Features:

Cross-Platform Compatibility: DroneKit supports multiple platforms, including Python for desktop applications and Android for mobile applications. This allows developers to create drone applications for various devices and environments.

Telemetry Access: DroneKit provides APIs for accessing real-time telemetry data from the drone, including GPS position, altitude, velocity, battery level, sensor readings, and more. Developers can use this data for monitoring and decision-making in their applications.

Flight Control Commands: Developers can send control commands to the drone using DroneKit APIs. These commands include controlling flight modes, setting waypoints, adjusting parameters, and executing specific maneuvers or actions. This enables developers to create applications that autonomously control the drone's behavior.

Mission Planning: DroneKit supports mission planning capabilities, allowing developers to create autonomous flight missions with predefined waypoints, actions, and parameters. Mission plans can be uploaded to the drone and executed autonomously, enabling complex flight operations without manual intervention. DroneKit provides a simulation environment for testing and debugging drone applications without the need for physical hardware. Developers can simulate drone behavior, test mission plans, and validate application logic in a virtual environment before deploying to real drones.

Integration with Ground Control Stations (GCS): DroneKit can be integrated with existing ground control station software, such as Mission Planner or QGroundControl. This allows developers to leverage the features of these GCS applications while adding custom functionality through DroneKit.

Community and Documentation: DroneKit has an active community of developers and enthusiasts who contribute to its development and provide support. The project offers comprehensive documentation, tutorials, and example code to help developers get started and build applications using DroneKit.

Open-Source: DroneKit is open-source, allowing developers to view, modify, and contribute to the codebase. This fosters innovation and collaboration within the drone development community, driving the evolution of the SDK and enabling new possibilities for drone applications.

2.3.2 Parameters:

Flight Mode: Developers may need to specify or retrieve information about flight modes supported by the drone, such as stabilization mode, altitude hold, GPS-guided navigation, or autonomous mission mode.

Vehicle State: Information about the drone's state, including its current position (latitude, longitude, altitude), velocity, orientation (roll, pitch, yaw), battery level, and sensor readings, can be accessed through DroneKit APIs.

Mission Planning: When creating autonomous missions or flight plans, developers may define parameters such as waypoints (latitude, longitude, altitude), actions to perform at each waypoint, loiter times, and other mission-specific settings.

Communication: Parameters related to communication protocols used for interfacing with the drone, such as baud rate, port settings, or network configurations, may need to be configured depending on the communication interface being used (e.g., serial, UDP, MAVLink).

Control: While DroneKit primarily focuses on providing high-level control interfaces, developers may still interact with lower-level control parameters indirectly through APIs for setting control commands, adjusting flight parameters, or configuring behavior settings.

Simulation: When using DroneKit's simulation environment for testing and debugging, developers may have access to simulation-specific parameters such as environmental conditions, simulated sensor data, or simulation time settings.

2.4 DroneUp:

DroneUp is a U.S.-based drone services provider offering complete UAV solutions ranging from mission planning to data delivery. It integrates both software and operational aspects for end-to-end drone management and is often used in logistics, surveying, public safety, and infrastructure inspection.

2.4.1 Features:

Comprehensive Mission Management: Offers full-stack drone services including flight planning, compliance, live tracking, and data analysis.

Real-time Tracking and Telemetry: Provides real-time visibility into UAV positions and system health.

Automated Compliance Tools: Built-in features for managing FAA compliance, airspace restrictions, and safety protocols.

Fleet and Operator Management: Central dashboard for managing drone fleets, pilots, certifications, and maintenance logs.

Enterprise Integration: DroneUp can be integrated with enterprise systems for logistics, delivery, and inspection data pipelines.

2.4.2 Parameters:

Navigation : Altitude, Velocity, Flight Mode, Obstacle Avoidance Enabled, GPS Waypoints.

Battery Management: Battery voltage, current draw, remaining flight time, and power efficiency.

Mission Planning: Route constraints, flight path optimization, payload delivery tasks.

Safety Controls: Geofencing limits, RTL (Return-to-Launch) altitude, loss of control actions.

Telemetry: Real-time telemetry feed for altitude, GPS coordinates, battery status, and wind conditions.

Environment Data: Wind speed/direction, visibility, weather impact.

Communication Status: Uplink/downlink signal strength, latency monitoring.

2.5 Skydio Autonomy:

Skydio Autonomy refers to the advanced artificial intelligence and computer vision technology developed by Skydio, a company specializing in autonomous drones. Skydio's autonomy system enables its drones to navigate complex environments with a high degree of precision and obstacle avoidance capabilities, allowing them to fly safely and effectively without constant human intervention.

2.5.1 Features:

Autonomous Flight: Skydio drones are equipped with advanced autonomy systems that allow them to navigate complex environments, avoid obstacles, and follow subjects autonomously. This capability is particularly useful for tasks such as filming action shots or conducting inspections where precise and obstacle-free flight is essential.

Obstacle Avoidance: Skydio drones utilize multiple sensors, including cameras and depth sensors, to detect and avoid obstacles in real-time. This allows them to navigate through densely wooded areas, urban environments, and other challenging landscapes while avoiding collisions.

Subject Tracking: Skydio drones excel at tracking and following subjects, whether they're moving on foot, on a bike, or in a vehicle. The drones use computer vision algorithms to recognize and track subjects, keeping them in the frame as they move.

High-Quality Cameras: Skydio drones are equipped with high-quality cameras capable of capturing stunning 4K video and high-resolution still images. This makes them suitable for professional photography and videography applications.

Compact and Portable Design: Many Skydio drones feature a foldable design, making them compact and easy to transport. This portability makes them ideal for traveling or hiking to remote locations.

Intuitive Controls: Skydio drones can be controlled via a smartphone app or a dedicated controller. The user interface is designed to be intuitive, allowing both beginners and experienced pilots to operate the drone effectively.

Smart Flight Modes: Skydio drones offer various intelligent flight modes, such as Orbit, Cable Cam, and Waypoints, which allow users to capture cinematic shots with ease.

Extended Flight Time: Skydio drones feature efficient battery systems that provide extended flight times compared to many other consumer drones. This allows users to capture more footage or complete longer missions without needing to land and recharge frequently.

Enterprise Solutions: In addition to consumer-focused drones, Skydio offers enterprise solutions tailored to specific industries such as public safety, infrastructure inspection, and construction. These solutions may include specialized payloads, software integrations, and support services.

Continuous Innovation: Skydio is known for its commitment to innovation, regularly releasing software updates and introducing new features to improve the capabilities and performance of its drones.

2.5.2 Parameters:

Position and Orientation: Parameters related to the drone's position (latitude, longitude, altitude) and orientation (roll, pitch, yaw) are fundamental for flight control and navigation.

Velocity: Parameters indicating the drone's speed and direction of movement are crucial for navigation and obstacle avoidance algorithms.

Obstacle Detection: Parameters related to obstacle detection, such as obstacle distance, size, and relative velocity, are essential for determining potential collision risks and planning avoidance maneuvers.

Tracking: Skydio Autonomy refers to the advanced autonomous flight capabilities offered by Skydio drones. While Skydio drones are equipped with advanced obstacle avoidance and tracking features, these capabilities are primarily focused on following and avoiding obstacles rather than external tracking of objects or subjects.

Real-time Mapping: Real-time mapping offers capabilities through its advanced autonomous flight features. Leveraging onboard cameras and computer vision algorithms, Skydio drones continuously capture high-resolution imagery of their surroundings during flight. This imagery is processed in real-time to generate maps and 3D reconstructions of the environment, providing users with up-to-date spatial information as the drone navigates through its surroundings.

Environment Perception: Parameters defining the selected flight mode or behavior, such as manual control, autonomous follow, waypoint navigation, or orbit mode.

Flight Mode: Parameters related to environmental factors such as lighting conditions, weather conditions, and terrain characteristics may be used to adapt drone behavior and optimize performance.

Camera Settings: Parameters controlling camera settings such as exposure, focus, white balance, and resolution may be adjusted to optimize image quality for different lighting conditions and applications.

Battery Status: Parameters indicating battery level, voltage, and remaining flight time are crucial for monitoring the drone's power supply and ensuring safe operation.

Communication Status: Parameters related to communication links between the drone, remote controller, and any connected devices (such as smartphones or tablets) are monitored to ensure reliable data transmission and control.

Flight Plan : Parameters defining way points, route constraints, and mission objectives in autonomous flight planning scenarios.

2.6 DJI Terra:

DJI Terra is a 3D mapping and modeling software by DJI, designed primarily for aerial photogrammetry, terrain reconstruction, and asset inspection using DJI drones.

2.6.1 Features:

Photogrammetry and 3D Modeling: Processes imagery to produce accurate 2D orthomosaics, 3D reconstructions, and point clouds.

Real-Time Mapping: Enables rapid mission feedback and situational awareness by mapping terrain during flight.

Obstacle-Free Mapping Tools: Adaptive mission planning to avoid obstructions in real time.

Advanced Processing Engine: High-speed, high-precision photogrammetry processing for construction and surveying use-cases.

Waypoints and Route Planning: Supports automated waypoint generation based on mapping areas and terrain height data.

Ground Control Points (GCPs) Support: Allows high-accuracy output through GCP integration.

User-Friendly Interface: Intuitive UI for mission design and post-processing analysis.

2.6.2 Parameters:

Mapping Settings:

- Flight altitude, overlap percentage (front/side), camera angle.
- Speed limit, image capture rate.

Processing :

- Reconstruction detail (low, medium, high).
- Output formats (3D model, orthomosaic, DSM).

Geospatial Settings:

- Coordinate system, ground sampling distance (GSD).
- GCP input and error thresholds.

Battery and Safety Settings:

- Return-to-home on low battery, fail-safe height.

Sensor Input:

- Camera focal length, resolution, and frame rate.

Flight Mode:

- Grid-based mapping, linear corridor mapping, oblique mode.

Real-Time Mapping:

- Terrain following toggle, adaptive exposure control.

2.7 PIX4D:

PIX4D is a popular software suite used for photogrammetry and drone mapping. It allows users to process aerial imagery captured by drones or other unmanned aerial vehicles (UAVs) to create 2D maps, 3D models, and point clouds. The software is widely used in various industries such as agriculture, construction, mining, surveying, and environmental monitoring.

2.7.1 Features:

Photogrammetry Processing: PIX4D uses photogrammetric algorithms to process overlapping images captured by drones. It reconstructs 3D models, point clouds, orthomosaics, and digital surface models (DSMs) from these images.

Accurate Georeferencing: The software provides tools for accurate georeferencing of the generated outputs, allowing users to integrate their data with geographic information systems (GIS) and other mapping software.

Terrain and Volume Calculation: PIX4D can calculate terrain elevations and volumes accurately, making it useful for applications such as earthwork monitoring, stockpile measurement, and terrain analysis.

Multi spectral and Thermal Processing: It supports processing of multispectral and thermal images captured by specialized sensors, enabling applications like precision agriculture, environmental monitoring, and thermal inspections.

Advanced Editing and Analysis Tools: Users can edit and analyze their 2D maps and 3D models within the software, including adding annotations, measurements, and customizing visualizations.

Integration with Drone Platforms: PIX4D integrates with various drone platforms, allowing users to plan flights, capture images, and process data seamlessly within the software ecosystem.

Cloud Processing and Collaboration: PIX4D offers cloud-based processing options, enabling users to upload their data for processing on remote servers. It also facilitates collaboration by allowing users to share and access projects online.

Training and Support: The company provides training resources, webinars, and technical support to help users get started with the software and make the most of its features.

2.7.2 Parameters:

Image Quality Settings: These settings control the quality and resolution of the images used in the processing pipeline. Parameters may include image scale, image overlap, and image format.

Calibration: PIX4D typically requires camera calibration parameters to accurately reconstruct 3D models and perform photogrammetric processing. These parameters include intrinsic camera parameters such as focal length, principal point, and lens distortion coefficients.

Processing Options: PIX4D offers various processing options tailored to different types of projects and desired outputs. Users can specify parameters such as processing template (e.g., aerial mapping, corridor mapping, agriculture), point cloud density, and output formats (e.g., orthomosaic, point cloud, DSM, 3D model).

Ground Control Points (GCPs): GCPs are reference points with known coordinates used to georeference and improve the accuracy of the outputs. Parameters related to GCPs include the number of GCPs, their coordinates, and their distribution across the project area.

Geospatial Regulation: Pix4D offers robust geospatial and regulatory features within its suite of software solutions, catering to the needs of professionals in fields such as surveying, agriculture, construction, and public safety. Geospatial capabilities enable users to process aerial imagery captured by drones into highly accurate maps, 3D models, and point clouds. These outputs are georeferenced, allowing for precise measurements, analysis, and visualization of spatial data.

Real-time Mapping: PIX4D software facilitates real-time mapping capabilities by leveraging drones equipped with cameras to capture imagery of the surrounding environment during flight. Through live video feeds and onboard processing, PIX4D enables users to generate maps and 3D models in real-time, providing instant spatial information for various applications.

Image Matching: PIX4D employs image matching algorithms to identify corresponding features in overlapping images. Parameters related to image matching include keypoint detection settings, feature matching algorithms, and matching sensitivity.

Terrain and Surface: For projects involving terrain modeling and elevation mapping, users can specify parameters related to terrain interpolation, surface smoothing, and contour generation.

Quality Assessment: PIX4D provides tools for assessing the quality and accuracy of the generated outputs. Parameters related to quality assessment include error thresholds, quality reports, and visual inspection tools.

Multispectral and Thermal: For projects involving multispectral or thermal imagery, users can specify parameters related to sensor calibration, band selection, and radiometric corrections.

Security and Privacy: Security and privacy are paramount in PIX4D's software solutions, with robust encryption techniques employed to protect data during transmission and storage. Access controls and user authentication mechanisms restrict access to authorized personnel only, ensuring confidentiality.

3 Conclusion:

After the survey, we can conclude that PX4 and ArduPilot stand out as the most comprehensive solutions, supporting advanced navigation, mission planning, and safety features. In contrast, DroneKit, DroneUp, and Skydio Autonomy offer more limited capabilities, while DJI Terra and PIX4D remain specialized for mapping and data processing rather than full flight control. Ultimately, the choice of software should be guided by the specific needs of the application, with considerations of safety, autonomy, and operational flexibility playing a critical role.

References

[1] F. B. Sorbelli, P. Chatterjee, F. Corò, S. Ghobadi, L. Palazzetti, and C. M. Pinotti, "A novel graph-based multi-layer framework for managing drone bvlos operations," *IEEE Transactions on Network and Service Management*, pp. 1–1, 2024.

[2] F. B. Sorbelli, P. Chatterjee, F. Coro, L. Palazzetti, and C. M. Pinotti, "A novel multi-layer framework for bvlos drone operation: A preliminary study," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 1–6, 2023.

[3] F. B. Sorbelli, P. Chatterjee, P. Das, and C. M. Pinotti, "Risk assessment in bvlos operations for uavs: Challenges and solutions," in *2024 20th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, pp. 300–307, 2024.