

Monocular Inter-UAV Distance Estimation Using Machine Learning and Geometric Scaling

Robert Pommeranz¹, Kevin Tebbe¹, Jakob Fischer¹, Ralf Heynicke¹, Gerd Scholl¹

*¹Electrical Measurement Engineering
Helmut-Schmidt-University, Hamburg, Germany
robert.pommeranz@hsu-hh.de*

Summary

Reliable Inter-UAV distance measurement is a prerequisite for formation control and collision avoidance in cooperative Unmanned Aerial Vehicle (UAV) swarms. This paper presents the system integration and experimental characterization of a monocular distance estimation pipeline that combines a fine-tuned YOLOv8s object detector with pinhole-model geometric scaling to convert bounding-box pixel widths into real-time depth estimates. The pipeline runs on an NVIDIA Jetson Orin NX companion computer via DeepStream SDK. The camera is intrinsically calibrated, and the effective projected width of each target UAV is determined from short-range hover reference measurements rather than nominal dimensions. An uncertainty propagation model is derived that combines camera calibration uncertainty, target width variability, and detector bounding-box jitter into a depth confidence interval. Two independent ground truth sources are used for validation: Laser Range Finder (LRF) measurements at thirteen waypoints (5 m – 50 m) with a DJI Matrice 4T target, and dual-rover RTK GNSS during a dynamic reference flight with a Holybro X650 target. A mean absolute error (MAE) below 2 m is achieved for ranges up to 35 m, with the DJI Matrice 4T. The pipeline requires only a low-cost camera and runs on embedded hardware, making it directly applicable to vision-based swarm coordination on weight- and power-constrained platforms.

Keywords: UAV, Object Detection, Distance Estimation, Machine Learning, Monocular Vision

Introduction

The rapid expansion of commercial and custom-built Unmanned Aerial Vehicles (UAVs) has fundamentally altered both civil airspace and the modern battlefield [1]. Swarm-based Counter Unmanned Aerial System (C-UAS) architectures have enabled multiple cooperative interceptors to engage simultaneous threats through coordinated behavior. A prerequisite for such swarms is continuous, online knowledge of the range to each neighboring UAV to fly in formation, avoid collisions, or coordinate based on mission requirements [2], [3]. Passive monocular ranging is equally relevant when an unknown UAV is detected by an active sensor such as radar, as the camera has a higher update rate and can provide continuous range tracking without generating further detectable emissions.

High accuracy ranging sensors like Laser Range Finder (LRF), stereo cameras or Ultra-Wideband (UWB) radio, achieve centimeter-level precision, but their cost, weight, and power budget can be prohibitive on smaller platforms [3]. A monocular camera, by contrast, is already present on most aerial vehicles. When the physical dimensions of the tracked target are known, the pinhole projection model yields a

depth estimate from a single frame without any cooperative signal [4].

Recent advances in real-time object detection [5], [6] and hardware-accelerated inference on companion computers like Nvidia Jetson now enable detection, tracking and distance estimation online and with low latency. Nevertheless, the accuracy of the geometric approach degrades quadratically with range, and its practical limits have not been systematically characterized at inter-UAV operational ranges. This paper addresses the gap by presenting the system integration and experimental characterization of a monocular inter-UAV distance estimation pipeline deployed on an embedded Jetson Orin NX mounted on a Holybro X650 quadcopter [7]. The distance estimation itself follows the pinhole projection model, and the contribution lies in the complete system realization and field validation. This work integrates a Camera Serial Interface (CSI) camera, YOLOv8s inference, object tracking, and geometric distance estimation into a single hardware-accelerated pipeline. Furthermore, the uncertainties of the model are quantified following the Guide to the Expression of Uncertainty in Measurement (GUM), combining camera calibration

uncertainty, target-width variability, and empirical detector pixel jitter [8]. Experimental validation characterizes the practical limits of this approach.

Related Work

Detecting smaller UAVs in real time on embedded hardware is a challenging task because computational resources are limited. One-Stage detectors are the state of the art for real-time applications due to their efficiency. This approach is established for example with YOLO [5]. Open-source datasets are available for training and benchmarking on UAV detection. A popular dataset is the Anti-UAV, which focuses on single-object tracking and comprises 160 video sequences of infrared (IR, 640 px × 512 px) and electro-optical (EO, 1920 px × 1080 px) imagery, of which approximately 100 annotated sequences are provided as training data [9]. The Drone-vs-Bird Challenge provides another dataset which consists of 77 annotated video sequences or 106.466 single images in different environments and small UAVs [10]. The focus of this dataset is that flying birds are as background clutter at larger distances.

Estimation of depth from a single image is a vision problem which has been solved by two main approaches: geometric and learning-based. The geometric approach relies on the pinhole-model scaling, which provides a solution for depth estimation when the target dimensions are known, or a true depth value is available at least once [4], [11]. For instance, Pang et al. [12] utilized a known-size marker to measure distance, while Bui et al. [13] extended this method to simultaneously measure distance and angle. Idohen et al. [14] achieved sub-millimeter errors at short range through curve fitting. However, these studies primarily evaluate their methods at short, controlled ranges, and the accuracy of geometric approaches at operational UAV engagement distances, where the quadratic depth-error growth dominates, has received less attention. Learning-based approaches, on the other hand, rely on large, annotated datasets with depth labels for training. Vajgl et al. [15] integrated a distance regression head into YOLO (Dist-YOLO), achieving 11 % mean relative error up to 150 m on the KITTI dataset [16]. However, this dataset is not representative of aerial UAV

scenarios, and the features in the images are different from those in the KITTI dataset.

Identifying the same object across frames even when the detection is partially lost is crucial for maintaining a constant bounding box around the target. Methods based on correlation filters, such as Discriminative Correlation Filter (DCF), are computationally efficient and can run in real time on embedded hardware [17]. Deep association trackers like DeepSORT leverage appearance features for improved tracking performance, but at a higher computational cost [18]. NVIDIA's DeepStream SDK provides GPU-accelerated implementations of such algorithms (NvDCF, NvSORT, NvDeepSORT) [19].

Methodology

This section describes the algorithmic pipeline for monocular UAV distance estimation, deployed on a Holybro X650 observer platform [7]. The pipeline proceeds in three stages: image capture, hardware-accelerated detection and tracking on the companion computer, and geometric distance estimation with uncertainty propagation.

The camera is calibrated using Zhang's method, which is implemented in OpenCV, recovering the intrinsic matrix K (focal length f_x , f_y and principal point c_x , c_y) and distortion coefficients [20]. Prior to distance estimation, each detected bounding box is undistorted using these coefficients so that the pixel width w_{px} is consistent with ideal pinhole geometry.

The detection and tracking pipeline runs on the Nvidia Jetson Orin NX using DeepStream SDK 7.1 for hardware-accelerated video processing. A YOLOv8s model [6] is fine-tuned on the Drone-vs-Bird Detection Challenge dataset [10]. The dataset is partitioned into training (80 %), validation (10 %), and test (10 %) subsets. Training uses 1280 px × 1280 px input resolution, batch size 32, and early stopping after 50 epochs without validation improvement (max 100 epochs). The trained weights are exported to ONNX and compiled into a TensorRT engine (FP16) using DeepStream-YOLO¹. For tracking, the NvDCF algorithm was selected for its high accuracy and efficiency [19].

The YOLO detection configuration is set so that the detection happens on every second frame to reduce the computational load, while the

¹ <https://github.com/marcoslucianops/DeepStream-Yolo> (accessed 23.03.2026)

tracker maintains the bounding box on the skipped frames and holds the object for up to ten frames without a new detection. This configuration was chosen to reduce the GPU load while maintaining a high bounding-box update rate.

With the calibrated focal length f_x and the known physical width W of the target UAV, the distance along the optical axis is estimated from the pixel width w_{px} using the pinhole camera model:

$$Z = \frac{f_x \cdot W}{w_{px}} \quad (1)$$

Substituting $w_{px} = f_x W / Z$ into the derivative $\frac{\partial Z}{\partial w_{px}} = -f_x W / w_{px}^2$, the following error approximation is obtained:

$$\Delta Z \approx -\frac{Z^2}{f_x W} \Delta w_{px} \quad (2)$$

showing that the depth error grows quadratically with distance.

The combined relative depth uncertainty is:

$$\frac{u_c(Z)}{Z} = \sqrt{\left(\frac{u(f_x)}{f_x}\right)^2 + \left(\frac{u(W)}{W}\right)^2 + \left(\frac{u(w_{px})}{w_{px}}\right)^2} \quad (3)$$

In Eq. (3) the only variable term is $u(w_{px})/w_{px}$, because bounding box decreases with distance and the object width and focal length stay constant in this setup. The baseline calibration records the pixel-width empirical standard deviation $s(w_{px})_{5m}$ over 1000 frames at five meters as a Type A evaluation; the standard uncertainty is set equal to this value $s(w_{px})_{5m} = u(w_{px})_{5m}$. As the target moves further away and its projected pixel width decreases, the absolute pixel-width jitter will also decrease. Therefore, the uncertainty $u(w_{px})$ is modeled as a proportional function (Eq. (4)) of the detected width, where $\rho = u(w_{px})_{5m} / w_{px,5m}$ is the relative jitter ratio, and a minimum jitter of 1.5 px is imposed.

$$u(w_{px}) = \max(1.5, \rho \cdot w_{px}) \quad (4)$$

The uncertainty of the target width is modeled as a Type B evaluation using a uniform distribution between the motor-to-motor and tip-to-tip propeller spans.

$$u(W) = \frac{W_{max} - W_{min}}{\sqrt{12}} \quad (5)$$

Two RTK GNSS rovers provide the 3D baseline between the observer and target platforms. Applying the Law of Propagation of Uncertainty $D = \sqrt{\Delta E^2 + \Delta N^2 + \Delta U^2}$ yields the UAV distance uncertainty:

$$u(D_{RTK}) = \frac{\sqrt{2}}{D} \sqrt{u_h^2(\Delta E^2 + \Delta N^2) + u_v^2 \Delta U^2} \quad (6)$$

where u_h and u_v are the per-rover horizontal and vertical position uncertainties. The factor $\sqrt{2}$ arises because the baseline is the difference of two independent rover positions, doubling the variance. Horizontal and vertical components are propagated separately since RTK receivers specify distinct accuracies for each axis.

The DJI Matrice 4T LRF specifies a range-dependent accuracy of $\pm(a + bD)[m]$ for a distance $D > 3$ m. Modelling this as a uniform distribution over the stated error interval and converting it to standard deviation gives Eq. (7), where a is a constant offset and b (dimensionless) is the range-dependent coefficient:

$$u(D_{LRF}) = \frac{a + bD}{\sqrt{3}} \quad (7)$$

To validate the system, the residual $\Delta = Z - D_{GT}$ between the camera estimate and the ground truth (reference) is evaluated using $u_c(\Delta) = \sqrt{u^2(Z) + u^2(D_{GT})}$. The measurement is considered acceptable if the absolute residual falls within the expanded uncertainty bounds ($|\Delta| \leq U(\Delta)$), where $U(\Delta) = k \cdot u_c(\Delta)$ using a coverage factor of $k = 2$.

Experimental Setup

The observer, shown in Fig. 1, is a Holybro X650 quadcopter equipped with a Jetson Orin NX companion computer. A Raspberry Pi Camera Module V2 (Sony IMX219, 8 MP, rolling shutter, F/2.0, 3.04 mm focal length, 62.2° HFOV) is used to capture the images. It operates at 1280 px × 720 px @60fps in a forward-facing mount, pitched upward from horizontal to keep the target in the field of view during level flight [21]. The intrinsic camera calibration yields $f_x = 1291.9$ px with a Type A standard uncertainty $u(f_x) = 0.47$ px. The RTK GNSS antenna is mounted on top of the frame; the camera optical center is offset from the

antenna by approximately $d_{cam} \approx (0.30, 0, -0.15)$ in the body frame (forward, right, down) in meter. Because the target adjusts its altitude to remain near the image center during the trials, the optical-axis depth closely approximates the Euclidean range, and no additional pitch correction is applied.



Fig. 1: UAV equipped with camera, RTK GNSS antenna, and Nvidia Jetson, to estimate the distance.

Two UAVs are used as targets: a Holybro X650 (motor-to-motor span: 0.65 m; tip-to-tip including propellers: 1.03 m) and a DJI Matrice 4T (motor-to-motor span: 0.39 m; tip-to-tip including propellers: 0.66 m). Based on Eq. (5) the mean and uncertainty values for both UAV dimensions (width) are shown in Tab. 1.

Tab. 1: Target width and uncertainty values

Target	$\bar{W}$ [m]	$u(W)$ [mm]
DJI M4T	0.53	78.0
X650	0.84	110.0

The uncertainty of the reference systems is derived from manufacturer specifications and error propagation. Both X650 UAVs carry a u-blox ZED-F9P dual-frequency GNSS receiver; a third unit serves as RTK base station, streaming RTCM3 corrections [22]. In RTK Fixed mode the manufacturer specifies the position accuracy as: $\leq 0.01 \text{ m} + 1 \text{ ppm}$ (CEP horizontal; 50th-percentile vertical). Converting to standard uncertainty gives $u_h = 8.5 \text{ mm}$ and $u_v = 15 \text{ mm}$ per rover. Since the baseline is the difference between two independent rover positions, the combined Euclidean distance uncertainty ranges from $\sqrt{2}u_h \approx 12 \text{ mm}$ for a purely horizontal baseline to $\sqrt{2}u_v \approx 21 \text{ mm}$ when the two platforms differ in altitude.

According to the manufacturer datasheet, the DJI Matrice 4T integrated LRF achieves a range-dependent accuracy of $0.2 + 0.0015D$ for more than three meters. Using the equation (7), the corresponding uncertainty ranges from 0.12 m at 5 m to 0.14 m at 50 m.

Results and Discussion

A precision of 0.984, recall of 0.970, mAP@0.5 of 98.8 %, and mAP@0.5:0.95 of 78.4 % are achieved by the fine-tuned YOLOv8s model on the test set. The mAP@0.5 is sufficient to provide reliable bounding-box information for the distance estimation in Eq. (1). However, the mAP@0.5:0.95 value indicates imprecise localization at stricter Intersection over Union (IoU) thresholds, which translates directly into bounding-box width jitter. Fig. 2 shows both UAVs at 5 m distance. These images are chosen to visualize the fact that the bounding box does not always cover the UAV correctly.

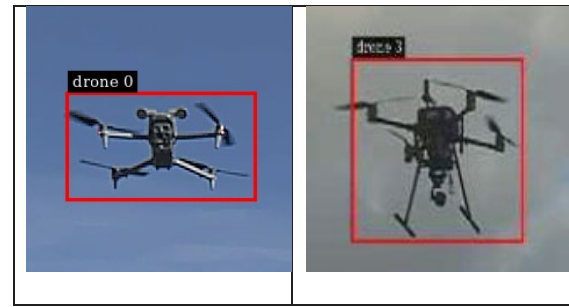


Fig. 2: Bounding-Box at 5m of the DJI M4T (left) and the Holybro X650 (right)

The DJI Matrice 4T was tracked at thirteen hovering waypoints in the range of 5 m – 50 m using the target width $W = 0.53 \text{ m}$, $u(W) = 78.0 \text{ mm}$ from Tab. 1. Beyond 50 m, detection became unreliable and the target was lost. The per-waypoint measurement and uncertainty values are summarized in Tab. 2. Across the complete static-hover measurement, the estimator achieves a Mean Absolute Error (MAE) between 0.24 m and 7.86 m, depending on the distance, and a mean signed error (bias) of -1.3 m. This systematic bias, which grows with range according to Eq. (2), is fully covered within the uncertainty bound at every distance. Fig. 3 shows the boxplot of every waypoint and the absolute error is below 2 m up to 35 m.

Tab. 2: *Measurement and uncertainty values with DJI M4T and LRF*

Dist. [m]	$\bar{w}_{bbox}$ [px]	$U(k=2)$ [m]	MAE [m]
5	164.0	1.58	0.24
7	120.3	2.17	0.15
9	92.3	2.77	0.31
11	76.1	3.38	0.33
13	63.7	3.98	0.50
15	55.0	4.59	0.59
20	42.5	6.22	0.34
25	34.7	7.94	0.64
30	28.8	9.78	0.90
35	25.0	11.74	1.10
40	23.0	13.84	2.85
45	21.2	16.10	4.56
50	20.3	18.53	7.86

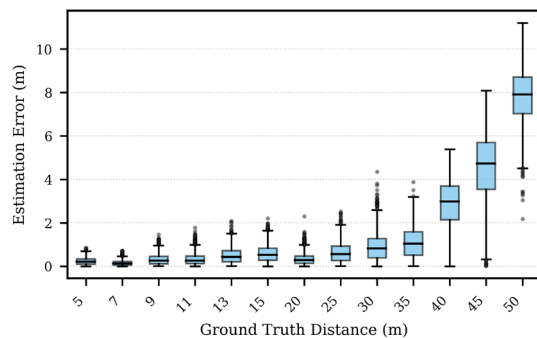


Fig. 3: *Absolute error of the distance estimation across all thirteen waypoints*

The second measurement involves two Holybro X650 UAVs, each equipped with a u-blox ZED-F9P RTK receiver. One UAV executed a predefined reference flight, departing from approximately 5 m at a constant velocity of 2 m/s to a maximum distance of 95 m and returning to the starting point. The target width $W = 0.84$ m, $u(W) = 110.0$ mm from Tab. 1 was used directly. The residual plot in Fig. 4 reveals a systematic negative bias which grows over distance. The MAE during the flight is 3.12 m, and the bias is -3.01 m.

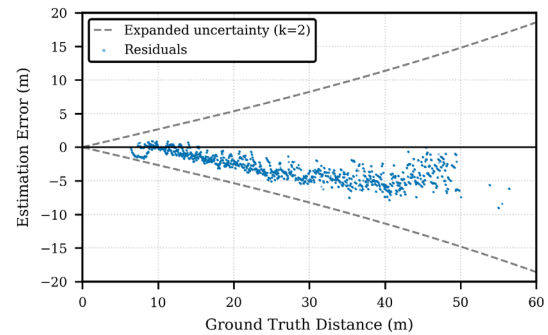


Fig. 4: *Residual of the raw camera estimates across the reference flight (RTK GNSS)*

Conclusion

This paper presents the system integration and experimental characterization of a monocular inter-UAV distance estimation pipeline combining a fine-tuned YOLOv8s object detector with pinhole-model geometric scaling, deployed entirely on an embedded NVIDIA Jetson Orin NX companion computer. An uncertainty propagation model was derived to predict the per-frame depth confidence from camera calibration uncertainty, target UAV width variability, and detector bounding-box jitter. The pipeline was validated against two independent ground truth sources.

In the LRF reference measurement trails the per-waypoint MAE remained below 1.1 m out to 35 m, rising to 4.56 m at 45 m and 7.86 m at 50 m. Therefore, all waypoints satisfy the predicted uncertainty.

With the RTK GNSS reference flight, the estimator achieved a MAE of 3.12 m, and detected the UAV up to 50 m. As with the previous measurement, the measurement values lie in the predicted uncertainty range.

This experiment has shown that a target UAV with a width of at least 0.6 m can be ranged out to approximately 50 m, where the bounding-box size decreases to less than 20 px, which is the limit of the detector. The pipeline constitutes a lightweight, low-power ranging solution that requires only a single calibrated camera and runs fully onboard.

Acknowledgement

This research is funded by dttec.bw – Digitalization and Technology Research Center of the Bundeswehr. dttec.bw is funded by the European Union – NextGenerationEU (project “Digital Sensor-2-Cloud Campus Platform” (DS2CCP), <https://dttecbw.de/home/forschung/hsu/projektds2ccp>).

References

- [1] K. Tebbe, R. Pommeranz, R. Heynicke, and G. Scholl, “From An Engineering Perspective: Small Drones Affecting The Course Of Warfare,” Jul. 2025, doi: 10.5281/ZENODO.15794252.
- [2] J. N. Yasin, S. A. S. Mohamed, M.-H. Hagbayan, J. Heikkinen, H. Tenhunen, and J. Plosila, “Unmanned Aerial Vehicles (UAVs): Collision Avoidance Systems and Approaches,” *IEEE Access*, vol. 8, pp. 105139–105155, 2020, doi: 10.1109/ACCESS.2020.3000064.
- [3] N. Gageik, P. Benz, and S. Montenegro, “Obstacle Detection and Collision Avoidance for a UAV With Complementary Low-Cost Sensors,” *IEEE Access*, vol. 3, pp. 599–609, 2015, doi: 10.1109/ACCESS.2015.2432455.
- [4] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, 2. edition, 17. printing. Cambridge: Cambridge Univ. Press, 2018.
- [5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.
- [6] Jocher, Glenn and Chaurasia, Ayush and Qiu, Jing et al., *Ultralytics YOLOv8*. (2023). Ultralytics. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [7] R. Pommeranz, K. Tebbe, R. Heynicke, and G. Scholl, “A Modular and Scalable System Architecture for Heterogeneous UAV Swarms Using ROS 2 and PX4-Autopilot.” doi: 10.48550/arXiv.2510.27327.
- [8] “Evaluation of measurement data — Guide to the expression of uncertainty in measurement.” Accessed: Apr. 15, 2026. [Online]. Available: <https://www.bipm.org/doi/10.59161/JCGM100-2008E>
- [9] N. Jiang et al., “Anti-UAV: A Large-Scale Benchmark for Vision-Based UAV Tracking,” *IEEE Trans. Multimed.*, vol. 25, pp. 486–500, 2023, doi: 10.1109/TMM.2021.3128047.
- [10] A. Coluccia, A. Fascista, L. Sommer, A. Schumann, A. Dimou, and D. Zarpalas, “The Drone-vs-Bird Detection Grand Challenge at ICASSP 2023: A Review of Methods and Results,” *IEEE Open J. Signal Process.*, vol. 5, pp. 766–779, 2024, doi: 10.1109/OJSP.2024.3379073.
- [11] A. Torralba, P. Isola, and W. T. Freeman, *Foundations of computer vision*. in Adaptive computation and machine learning series. Cambridge, Massachusetts London, England: The MIT Press, 2024.
- [12] Y. Pang, Y. Zhao, J. Chen, S. Wang, and H. Chen, “Viewing distance measurement using a single camera,” in *2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference*, Chongqing, China: IEEE, Dec. 2014, pp. 512–515. doi: 10.1109/ITAIC.2014.7065103.
- [13] M. T. Bui, R. Daskocil, and V. Krivanek, “Distance and angle measurement using monocular vision,” in *2018 18th International Conference on Mechatronics - Mechatronika (ME)*, IEEE, Dec. 2018, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/8624694>
- [14] E. E. Idohien, O. M. Afolayan, and U. A. Umar, “Object distance estimation using a monovision camera,” *IAES Int. J. Robot. Autom. IJRA*, vol. 12, no. 4, p. 325, Dec. 2023, doi: 10.11591/ijra.v12i4.pp325-331.
- [15] M. Vajgl, P. Hurtik, and T. Nejezchleba, “Dist-YOLO: Fast Object Detection with Distance Estimation,” *Appl. Sci.*, vol. 12, no. 3, p. 1354, Jan. 2022, doi: 10.3390/app12031354.
- [16] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, “Vision meets robotics: The KITTI dataset,” *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1231–1237, Sep. 2013, doi: 10.1177/0278364913491297.
- [17] A. Lukežič, T. Vojříř, L. Čehovin Zajc, J. Matas, and M. Kristan, “Discriminative Correlation Filter Tracker with Channel and Spatial Reliability,” *Int. J. Comput. Vis.*, vol. 126, no. 7, pp. 671–688, Jul. 2018, doi: 10.1007/s11263-017-1061-3.
- [18] N. Wojke, A. Bewley, and D. Paulus, “Simple online and realtime tracking with a deep association metric,” in *2017 IEEE International Conference on Image Processing (ICIP)*, Beijing: IEEE, Sep. 2017, pp. 3645–3649. doi: 10.1109/ICIP.2017.8296962.
- [19] “NVIDIA DeepStream 7.1 SDK Documentation: Gst-nvtracker.” Accessed: Mar. 25, 2026. [Online]. Available: https://docs.nvidia.com/metropolis/deepstream/7.1/text/DS_plugin_gst-nvtracker.html
- [20] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000, doi: 10.1109/34.888718.
- [21] “Camera - Raspberry Pi Documentation.” Accessed: Apr. 02, 2026. [Online]. Available: <https://www.raspberrypi.com/documentation/accessories/camera.html>
- [22] u-blox, “ZED-F9P-04B Data sheet.” [Online]. Available: https://content.u-blox.com/sites/default/files/ZED-F9P-04B_DataSheet_UBX-21044850.pdf