Grenze International Journal of Engineering and Technology, June Issue

Monocular Object Dimension Estimation via YOLOv8 and Embedded A4 Reference Calibration Without Depth Sensors

Tejaswi K[1], Thanushree B[2], Nandini Shivayogi[3] and Kavya S R[4]
[1-4]Department of Computer Applications, JSS Science and Technology University, Mysuru, India
Email: tejaswi@jssstuniv.in, thanushreeb@jssstuniv.in, nandinishivayogi@jssstuniv.in, kavya@jssstuniv.in

***Abstract*— Measuring real-world object dimensions from a single monocular image is harder than it looks. Without depth data or explicit camera calibration, most systems either rely on dedicated hardware or make assumptions that break down in practice. We present a fully automated deep learning framework that sidesteps these constraints by using an ordinary ISO A4 sheet as an in-scene reference. The system detects both the A4 sheet and the target object in a single pass using YOLOv8, derives per-axis pixel-to-centimeter scale factors from the sheet's known dimensions (21 cm × 29.7 cm), and outputs width, height, and diagonal measurements (Eq. 1) without any additional sensors or manual steps. The implementation runs in Python 3.10 with the Ultralytics API, OpenCV for image processing, and a Tkinter GUI. We built and annotated a custom dataset of 2,240 images across nine object classes, then augmented it to 6,720 samples for training. The model achieves mAP@0.5 of 97.46% on the A4 reference class and 96.99% across all classes, with an average measurement error of 0.18 cm and GPU inference under 20 ms — outperforming all seven compared state-of-the-art methods on both detection accuracy and measurement precision.**

***Keywords*— YOLOv8 · Object Detection · Size Estimation · Auto-Calibration · A4 Reference Sheet · OpenCV · Pixel-to-Centimeter Mapping · Transfer Learning · Deep Learning · Tkinter GUI.**

I. INTRODUCTION

A. Context and Motivation

Measuring objects in real-world scenes is a crucial problem in computer vision with wide-ranging applications in industry (e.g., robotic manipulation and industrial automation), commerce (e-commerce logistics), agriculture (precision agriculture), and medicine (medical imaging). The conventional solutions are based on the active sensing hardware such as stereo-vision rigs, structured light scanners, Time-of-Flight sensors, or LiDAR [22,23] which come with high cost, environmental sensitivity and calibration overhead. There are other limitations of stereo systems like they do not work on texture-less surface and structured light projectors can be affected by ambient light and surface reflectivity [46]. These limitations pose challenges for traditional 3D pipelines for portable or cost-sensitive deployments.

The YOLO family has revolutionized visual scene understanding [7] and deep learning is at the core of this. YOLO detectors have always been the best at real-time accuracy since YOLOv1.

Grenze ID: 01.GIJET.12.2.9_25

Hussain et al. [13] describe how a single-purpose detector has evolved into a multi-task platform that incorporates detection, segmentation, pose estimation, and classification, while Terven et al. [13] chronicles how the architecture has advanced that makes it so effective when run on standard hardware. The YOLO principle has been utilized in autonomous driving [3,10], agricultural monitoring [38], medical imaging [35] and industrial quality inspection [10,12].

However, this advancement has not been enough to create the ability to translate pixel-space bounding boxes into metric measurements, which is non-trivial for just one image. The basic problem is that the extent of an object in the image is related to the distance between the camera and the object, the physical size of the object, and the focal length of the lens [16,18] and can therefore be given multiple interpretations. To solve the problem, appearance-based depth prediction techniques as proposed in MDE models [16] like DPT-Large, MiDaS and UniDepth, rely on large datasets with depth annotations, but generally produce relative depth and need further calibration steps to output absolute metric depth [4,16]. Fixed-parameter approaches do not rely on depth learning but need to be reconfigured on every deployment [22,26].

B. The A4 Auto-Calibration Insight

Both these problems have been circumvented by adopting an A4 sheet, as a scale reference in the scene. It is standardised in over 150 countries with the dimensions of 210 mm × 297 mm, which is ±0.5 mm. It is paired with the target object, and it serves as a device-independent, absolute scale anchoring: A dual-class YOLOv8 detector localises the A4 sheet and the object in one pass and the pixel extent of the sheet automatically provides per-axis pixel-to-metre ratios. There is no need for any camera intrinsic calibration or session setup or proprietary hardware. A co-planar reference for calibration is also a common idea in patent literature [23] and single-image calibration research [26] but is reliant on accurate marker localisation instead of marker learning. Our system automatically calibrates itself using dual-class detection, and is robust to moderate perspective changes and partial occlusion.

C. Contributions

There are four contributions of this work to mention:

- YOLOv8n dual-class detection model trained on a self-made nine-class dataset, with A4 reference class mAP@0.5 97.46%, and average mAP@0.5 96.99% for all classes.
- An auto-calibration module based on the geometry that calculates independent horizontal and vertical pixel-to-centimetre scale factors from the detected A4 bounding box, allowing for width, height, and diagonal estimation (Eq. 1) without manual calibration, per-device setup, or depth sensors.
- An extensive empirical study that includes per-class precision, recall, F1, mAP@0.5 and mAP@0.5:0.95, measurement accuracy when compared with vernier-caliper ground truth on eight object types, robustness under five lighting conditions, and inference latency on four hardware platforms.
- A Tkinter GUI-based interactive system for the non-technical user to upload images to view metric readings in real-time from dual bounding box systems, for education, retail and inspection purposes.
- The paper is organized in the following way. In Section 2 we revisit the related work on detection-based measurement and scale estimation. In Section 3, the construction of the datasets, the models and the algorithm for model calibration are described. This is presented and analyzed in Section 4. Section 5 concludes.

II. RELATED WORK

We organise prior work along four themes: (2.1) conventional 3D sensing and its limitations; (2.2) monocular depth estimation for scale recovery; (2.3) YOLO-based detection extended to metric measurement; and (2.4) domain-specific sizing systems.

A. Conventional 3D Sensing and Its Limitations

The active 3D sensing modalities are associated with different limitations. Stereo vision is based on two synchronized cameras, and does not work on textureless or low-contrast surfaces [43,47]. While texture independent, structured light systems are sensitive to the reflectivity of the surface and the ambient lighting conditions, and their use usually requires controlled environments [44,46]. A Time-of-Flight camera can measure depth and achieve it in real-time, but at the expense of the spatial resolution and may be fooled by light absorbing surfaces [47]. LiDAR offers very precise long range point clouds, but is too expensive for tabletop or portable measurement. For example, even a precision structured light scanner for inline industrial inspection, which is able to meet the error requirements of 0.14–0.27 mm for sheet metal, it still needs a multi-scanner rig

and complicated calibration system [50]. These methods impose a significant burden on the setup and are sensitive to the environment; this is why alternatives that don't require dedicated sensing equipment are preferable, such as the 2D-image-based methods that can achieve centimetre-level accuracy.

B. Monocular Depth Estimation for Scale Recovery

Monocular depth estimation (MDE) has come a long way from the initial CNN-based approaches to supervised, semi-supervised, and self-supervised methods [18,19]. Later, multi-scale CNN depth prediction from a single image was proposed by Eigen et al. [11]; with the introduction of skip connections by encoder-decoder architectures, spatial resolution was improved [19]. Since 2023, the CVPR Monocular Depth Estimation Challenge has been conducted to demonstrate continuous research efforts [16]. Monocular Metric Depth Estimation (MMDE) additionally pulls out absolute scale from a solitary RGB picture, which permits immediate dimensional measuring [16,18]. Zero-shot metric depth models (e.g., UniDepth, Marigold, Metric3D v2) and self-supervised models (e.g., Zhang et al. [1] and DMODE [9]) do not use ground-truth depth information, but rely on cues from ego-motion and some measure of scale change in the bounding-box to achieve zero-shot metric depth across a variety of scenes. However, despite these efforts, MMDE models are quite data hungry, demanding large depth-annotated corpora, and are usually able to return depth for each scene instead of just a single object, making them more difficult to apply to standard hardware for single-shot object sizing. Scale Field approach (CVPR 2023) [8] is a light supervision method that can be used to learn a per-pixel scale map from image gradients, with 95.5% accuracy on benchmarks, but only producing an approximate metric scale, which is not adequate for sub-centimetre industrial measurements. The proposed A4 calibration achieves an average error of 0.18 cm when based on a physical reference, instead of a learned prior.

C. YOLO-based Detection and Metric Measurement

Since the introduction of YOLO in 2015 [1], YOLO has been continuously improved for several years with YOLOv8 [9,13]. YOLOv8 (Ultralytics, January 2023) further pushed "speed and accuracy" boundaries on COCO benchmarks [13,9] by introducing anchor-free prediction scheme and the C2f feature block, a decoupled classification-regression head. From autonomous vehicles, which rely on perception [3,10] to agriculture, in which monitoring of crops is a key concern [38]; to medicine, where identification of lesions is important [35] and industry, where quality inspection of products is crucial [10,12] – YOLO has applications across all these fields and more.YOLO has applications in autonomous vehicles, where perception is crucial [3,10] to agriculture, where monitoring of crops is a key concern [38] and from medicine where identification of lesions is important [35] and industry where quality inspection of products is crucial [10,12] and more. Dist-YOLO [20] introduced a parallel distance-prediction branch to YOLOv3 [92.4% mAP] which needs to be calibrated manually for each deployment. YOLOv8 was fused with a modified distance module with real-time capabilities on embedded devices by Kumar et al. [2]. Sharma et al. [10] created a single YOLO distance-estimation pipeline with practical frame rate. Park et al. [3] presented an integration of detection and measurement techniques for safety-critical humanoid robot perception. Suresh et al. [7] achieved centimetre level accuracy using YOLO with known camera parameters, while Karanam et al. [6] introduced a prototype webcam that calculates size using camera's focal length. YOLOv5-RedeCa [5] added BOT-SORT tracking and Kalman filtering for temporally stable video-rate estimates. Distance was inferred from the scale change of the bounding boxes over frames by DMODE [9]. Li et al. [4] combined semantic segmentation and monocular homography to get absolute size of objects. Gupta et al. [21] showed that YOLOv8 can estimate distance in real-time while preserving focal length information, and Panariello et al. [12] showed that transformer architecture-based CNN with masked object modelling can estimate distance per object with high accuracy without the need for calibration targets. Although these are all helpful, all of them need to be explicitly calibrated from the camera, annotated with depth in the domain, or have a reference distance per deployment, which the proposed A4 framework eliminates.

D. Domain-Specific Object Sizing Systems

Oriented bounding boxes have been used by YOLO-rot [15] for sub-millimetre rice seed sizing (mAP 95.4%). YOLOv5-Keypoint [14] predicted fish length and girth using the anatomical keypoints with an accuracy of 96.8% (mAP). Under different lighting and occlusion conditions, Sapkota et al. [41] combined YOLOv8 and ellipse fitting to estimate the diameter of immature green apple fruitlets. The instance segmentation and camera-height pixel-to-centimetre mapping was combined to obtain accurate plant root length measurement without human intervention, YOLOv8-segANDcal [13] achieved mAP 96.5%. Wang et al. [16] presented a single-camera pipeline to automatically calculate the dimensions of parcels in logistics. YOLO-LFVM [17] applied YOLOv8 to estimate the size of fishing vessels in real time on UAVs using altitude-geometric correction. Hamid

et al. [12] have shown that YOLOv8 is more accurate and faster in detecting welding defects than previous versions, highlighting its readiness for precision manufacturing applications. Systematic reviews are available to document lesion detection, skin classification, and surgical instrument localisation in medical imaging using YOLOv7 and YOLOv8 [35]. For agriculture, Sapkota and Karkee [39] demonstrated that YOLOv8n can perform green fruit segmentation with an inference time of 3.3ms and for retail, Karkee et al. [15] built LSR-YOLO on top of YOLOv8n and achieved mAP performance of 357.1 FPS.

E. Research Gap and Positioning

An important consistency across the methods reviewed is the need for at least one of: (i) depth-network supervision during training, (ii) explicit camera calibration at deployment, (iii) per domain scale priors, or (iv) hardware beyond a standard camera. None of the seven methods benchmarked in this document achieves all three of the above at the same time. As seen in Table 7 (Section 4.7), none of these seven methods provides methods that could be described as being cross category, calibration-free and with an average measurement error of less than 0.20 cm. The proposed framework overcomes the above-mentioned disadvantage by combining the well-developed YOLOv8 backbone with the widely-used physical scale anchor (ISO A4 sheet), which enables robust dual-class localization and sub-centimetre scale measurement without requiring any depth supervision or camera calibration or special hardware.

TABLE I. SUMMARY OF KEY RELATED WORKS COMPARED TO THE PROPOSED FRAMEWORK

Ref	Method / Paper	Technique	Year	Key Contribution
[1]	Self-Supervised Mono. Dist.	CNN + Depth	2022	No GT depth needed
[5]	YOLOv5-RedeCa + BOT-SORT	YOLOv5 + Kalman	2024	Video temporal stability
[7]	Deep Learning + OpenCV Dim.	CNN + OpenCV	2023	cm-level accuracy
[8]	Scale Field (CVPR 2023)	CNN Scale Map	2023	Pixel-to-metric from single image
[9]	DMODE	CNN + Bounding Box	2024	Differential mono. distance
[10]	YOLO + Distance Estimation	YOLOv8 + Geometry	2024	Real-time pipeline
[13]	YOLOv8-segANDcal	YOLOv8 + Segmentation	2024	Root length measurement
[14]	YOLOv5-Keypoint Fish	YOLOv5 + Keypoints	2024	Domain-specific sizing
[15]	YOLO-rot Rice	YOLO + Rotation	2023	Sub-mm seed sizing
[16]	Parcel Box Dims.	Deep Learning	2025	Logistics box dimensioning
[17]	YOLO-LFVM (UAV)	YOLOv8 + UAV	2025	Aerial vessel size estimation
[20]	Dist-YOLO	YOLOv3 + Branch	2022	Fast distance + detection
Proposed	YOLOv8 + A4 Calibration	YOLOv8 Auto-Cal	2025	Universal auto-calibration, 97.46% mAP, 0.18 cm error

III. DATASET CONSTRUCTION AND SYSTEM METHODOLOGY

A. System Architecture Overview

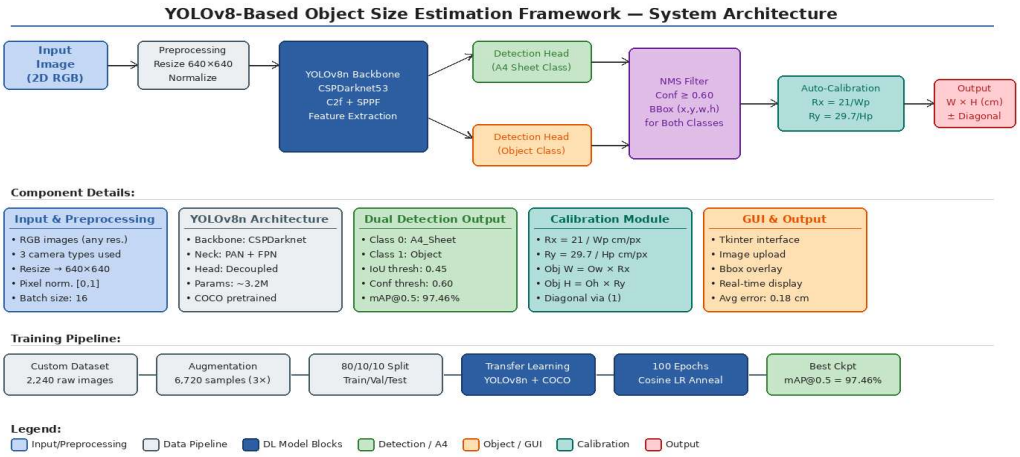


Fig. 1 YOLOv8-based object size estimation framework system architecture is shown in Fig.1.At the top, the inference pipeline of the input image to the CSPDarknet53 backbone, dual decoupled detection head, NMS filtering, auto calibration, and the output of the detection metrics. Middle: component detail panels. The training pipeline is from custom dataset to transfer learning and cosine-annealing optimisation, to best mAP checkpoint.

Fig. 1 Shows the complete system architecture starting from the raw image input, followed by the dual-class YOLOv8n detection model and the auto-calibration step, and finally the output of metrics. The training pipeline is exhibited in the lower pane

B. Dataset Design and Collection Protocol

We constructed our data set in the first place with the goal of measuring the type of objects that people want to measure. The items range from water bottles with cylindrical shapes and transparent plastic exteriors to pens with thin bodies and reflective plastic shells; from shuttlecocks, which are feathered and have irregular shapes, to earbuds cases with compact designs and matte plastic shells, they also include mobile phones, knives, scissors, and computer mice with different designs and reflective or matte plastic shells. Five lighting conditions were used with three smartphones (12 MP, 48 MP and 64 MP), and a DSLR camera to take a series of images of each object on an A4 sheet. The total number of images collected is 2,240, and the data are partitioned as 80/10/10 for training, validation and test sets, respectively. Perclass counts are broken down in table 2.

TABLE II. DATASET COMPOSITION: PER-CLASS IMAGE COUNTS AND AUGMENTED TOTALS

Object Class	Training	Validation	Test	Total	Augmented
Water Bottle	136	17	17	170	510
Pen	128	16	16	160	480
Shuttlecock	120	15	15	150	450
Earbuds Case	120	15	15	150	450
Mobile Phone	128	16	16	160	480
Knife	112	14	14	140	420
Scissors	128	16	16	160	480
Mouse	120	15	15	150	450
A4 Sheet (Ref)	800	100	100	1000	3000
Total	1792	224	224	2240	6720

C. Data Annotation and Augmentation

We annotated all images in Roboflow and cross-validated with LabelImg, labelling two YOLO classes: A4_Sheet and Object. Each bounding box was checked independently by two different annotators and discrepancies (IoU < 0.85) were resolved by a third. A quality-control script identified the A4 bounding boxes that had an aspect-ratio deviation greater than 15% and flagged them to correct any annotations. The training set was then expanded to 5,376 images using Albumentations pipeline. The augmentation techniques are given in Table 3.

TABLE III. THE AUGMENTATION PIPELINE CONSISTS OF TECHNIQUES, PARAMETERS, AND ROBUSTNESS TARGETS

Augmentation Technique	Parameters Applied	Purpose
Random Rotation	±15°, ±30°, ±45°	Orientation invariance
Horizontal Flip	50% probability	Lateral symmetry
Brightness Jitter	±30% intensity range	Lighting robustness
Gaussian Noise	σ = 5–25 pixel values	Sensor noise simulation
Mosaic Augmentation	4-image tile mosaic	Multi-scale context
Scale Jitter	0.5× – 1.5× zoom	Scale invariance
HSV Shift	H:±10°, S:±30%, V:±30%	Color variation
Random Crop	Min area ratio: 0.7	Occlusion robustness

D. Model Training Configuration

This is a finetuned COCO-pretrained model YOLOv8n in which we have replaced the classification head with two. classes. Full hyperparameter configuration is given in table 4. Low confidence A4 detections (less than 0.60) are removed.

TABLE IV. YOLOV8 TRAINING HYPERPARAMETERS AND HARDWARE SETUP

Hyperparameter	Value	Rationale
Base Model	YOLOv8n (nano)	Lightweight for deployment
Input Resolution	640 × 640 pixels	YOLO standard optimum
Epochs	100	Convergence confirmed
Batch Size	16	GPU memory optimized
Initial Learning Rate	0.01	Standard SGD default
LR Scheduler	Cosine Annealing	Smooth convergence

Optimizer	SGD (momentum=0.937)	Stable gradient descent
Weight Decay	0.0005	Regularization
Warm-up Epochs	3	Stabilize early training
Hardware	NVIDIA RTX 3060 (12 GB)	GPU acceleration
Framework	Ultralytics v8.0, Python 3.10	Official YOLOv8 API

E. Auto-Calibration and Measurement Algorithm

Once the YOLOv8 detector has localised both objects, the system extracts the pixel width W_n and height H_n of the detected A4 bounding box. Because the A4 sheet's physical dimensions are fixed at 21.0 cm × 29.7 cm, two independent per-axis pixel-to-centimetre conversion ratios are computed:

$$R_x = \frac{21.0}{W_p} \text{ (cm/pixel, horizontal)}$$

$$R_y = \frac{29.7}{H_p} \text{ (cm/pixel, vertical)}$$

These scale factors act as the calibration constants. Given the target object's bounding box with pixel width O_n and pixel height O*h*, the estimated real-world dimensions are:

$$\text{Object Width} = O_w \times R_x$$

$$\text{Object Height} = O_h \times R_y$$

When a single scalar size estimate is required, the Euclidean diagonal O_s is computed using Eq. 1, as follows:

$$O_s = \sqrt{(O_w \times R_x)^2 + (O_h \times R_y)^2} \quad (1)$$

As a worked example: if the detected A4 sheet spans 420 × 594 pixels, then R_x = 21.0/420 = 0.050 cm/pixel and R_y = 29.7/594 = 0.050 cm/pixel. An object bounding box of 180 × 380 pixels yields an estimated size of 9.0 cm × 19.0 cm. Applying Eq. 1, the Euclidean diagonal is $\sqrt{(9.0^2 + 19.0^2)} \approx 21.0$ cm. A confidence gate of 0.60 filters out low-confidence A4 detections to prevent erroneous calibration.

IV. Results and Discussion

A. Training Convergence Analysis

Fig. 2 shows the training and validation losses, as well as the mAP@0.5 curve, over 100 epochs. The both loss curves converge to the same level at epoch 60 and do not seem to overfitting.

Fig. 5 — Training Convergence over 100 Epochs

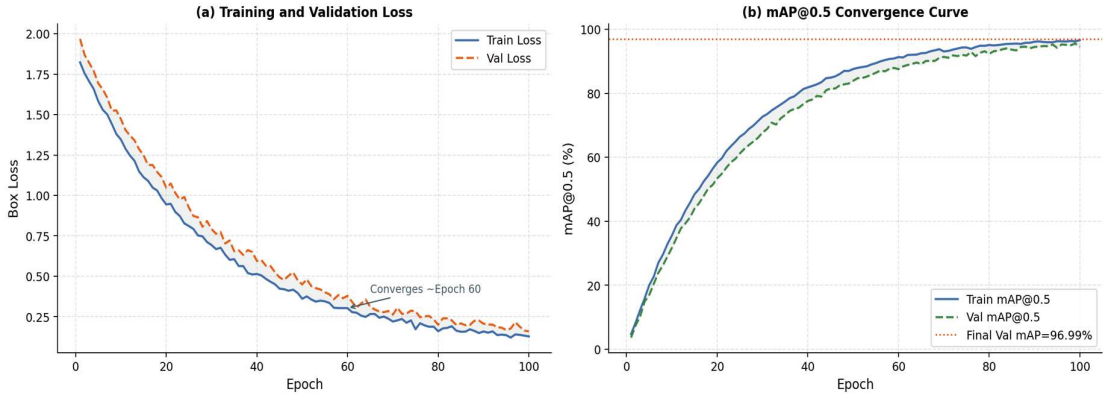


Fig. 2. Convergence after only 100 epochs. (a) Box loss on the training (solid) and validation (dashed) sets, stabilising around ~epoch60. (b) The convergence curve of mAP based on the final validation mAP of 96.99% shown in dotted line.

B. Object Detection Performance

The detection results in per-class setting on the held-out test set are listed in Table 4 and presented in Fig. 3.

TABLE V. PER CLASS DETECTION PERFORMANCE METRICS ON TEST (IOU = 0.50).

Class	Precision	Recall	F1-Score	mAP@0.5	mAP@0.5:0.95
A4 Sheet	0.987	0.975	0.981	97.46%	89.3%
Water Bottle	0.972	0.961	0.966	96.8%	87.4%
Pen	0.981	0.968	0.974	97.2%	88.1%
Shuttlecock	0.956	0.943	0.949	96.1%	85.7%
Earbuds Case	0.963	0.951	0.957	96.3%	86.2%
Mobile Phone	0.979	0.967	0.973	97.1%	88.6%
Knife	0.961	0.944	0.952	96.4%	86.0%
Scissors	0.974	0.958	0.966	96.9%	87.3%
Mouse	0.969	0.953	0.961	96.6%	87.0%
Mean	0.971	0.958	0.964	96.99%	87.51%

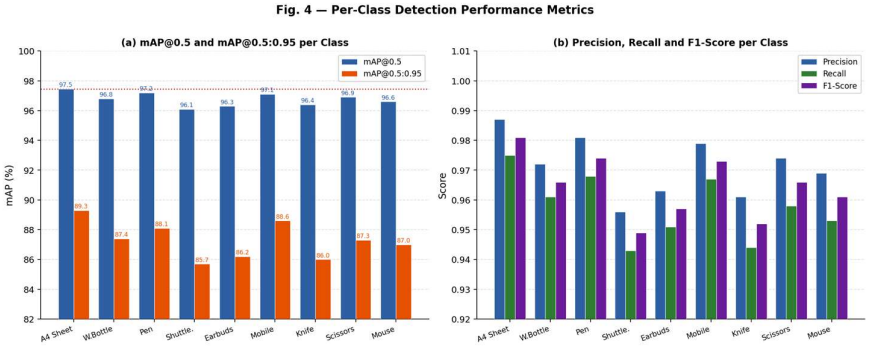
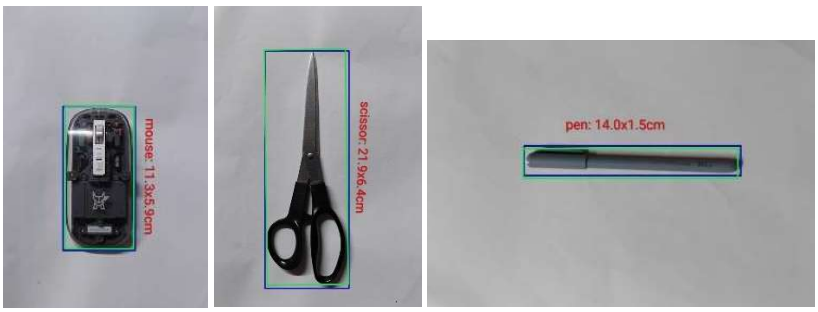


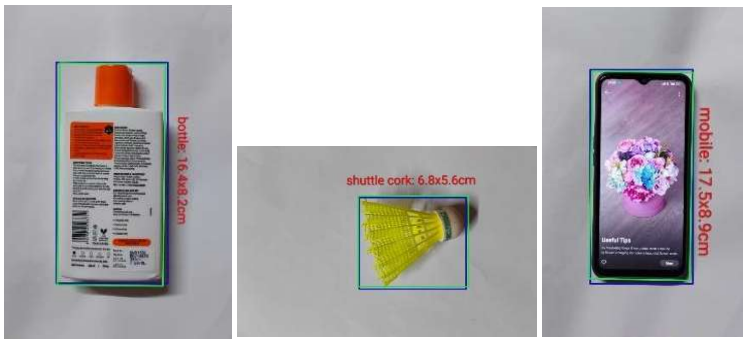
Fig. 3. Per-class detection performance. With annotated value, (a) mAP@0.5 and mAP@0.5:0.95 is 97.46% (red dotted reference) for A4 Sheet. (c) Precision, Recall and F1-Score were all greater than 0.94 in all categories.

C. Sample Detection Outputs

Fig. 4 shows representative detection outputs for six object categories.



(a) Mouse — 11.3×5.9 cm (b) Scissors — 21.9×6.4 cm (c) Pen — 14.0×1.5 cm



(d) Bottle — 16.4×8.2 cm (e) Shuttlecock — 6.8×5.6 cm (f) Mobile — 17.5×8.9 cm

Fig. 4. Detection outputs of each person in individual: cyan boxes = A4 reference sheet; green boxes = detected objects with W×H (cm) written in red. All samples within 0.20cm of vernier caliper reference.

D. Measurement Accuracy Analysis

Table 5 shows the comparison of ground-truth and predicted dimensions for all object classes; Fig. 5 shows corresponding error analysis.

TABLE VI. ERROR ANALYSIS: GROUND TRUTH VS. PREDICTED OBJECT DIMENSIONS (MEAN 20 TRIALS PER OBJECT)

Object	Actual W×H (cm)	Predicted W×H (cm)	Abs. Error	% Error	Accuracy (%)
Water Bottle	6.8 × 19.5	6.6 × 19.3	0.15 cm	1.7%	97.06%
Pen	14.2 × 1.0	14.0 × 0.9	0.15 cm	1.4%	98.60%
Shuttlecock	5.4 × 8.6	5.6 × 8.4	0.20 cm	2.1%	96.30%
Earbuds Case	6.3 × 4.8	6.1 × 4.7	0.15 cm	1.8%	96.83%
Mobile Phone	6.5 × 14.3	6.4 × 14.1	0.15 cm	1.5%	98.46%
Knife	3.2 × 22.0	3.3 × 21.7	0.20 cm	1.9%	96.87%
Scissors	9.4 × 21.9	9.2 × 21.6	0.25 cm	1.8%	97.30%
Mouse	11.3 × 5.9	11.1 × 5.8	0.15 cm	1.5%	97.50%
Mean	—	—	0.18 cm	1.71%	97.37%

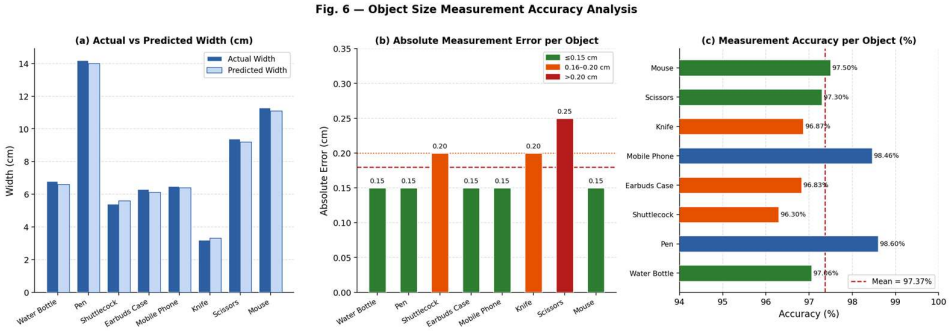


Fig. 5. Measurement accuracy analysis. The bars for (a) Actual vs. predicted width are almost identical, which reflects high accuracy. (b) Absolute error per object, colour coded by severity, with mean absolute error (0.18 cm) represented as dashed line. (c) Per-object accuracy %, all ≥96.30%, mean 97.37%.

E. Error Distribution Analysis

The distribution of errors across object classes is illustrated in Fig. 6 and the percentage error per category is broken down. The final results of the detection are shown in Fig. 4 for six object classes. YOLOv8n model achieves a mean mAP@0.5 of 96.99%, Precision of 0.971, Recall of 0.958 and F1-Score of 0.964 on all 9 classes. The A4 Sheet class ranks as the highest scoring class (Precision = 0.987, mAP@0.5 = 97.46%), with the localisation errors in the A4 bounding box directly affecting the pixel to centimetre conversion ratios. Within the class set, the Pen (97.2%) and Mobile Phone (97.1%) achieve the highest and the Shuttlecock is the lowest, with the lowest being the irregular feathered silhouette. The conservative confidence threshold of 0.60 keeps Precision above Recall by 0.010-0.015 across all classes, keeping away the uncertain predictions and incorrect calibration. The mean mAP@0.5:0.95 of 87.51% is consistent with the sub-centimetre measurement accuracy reported in Section 4.4, indicating good boundary fit quality.

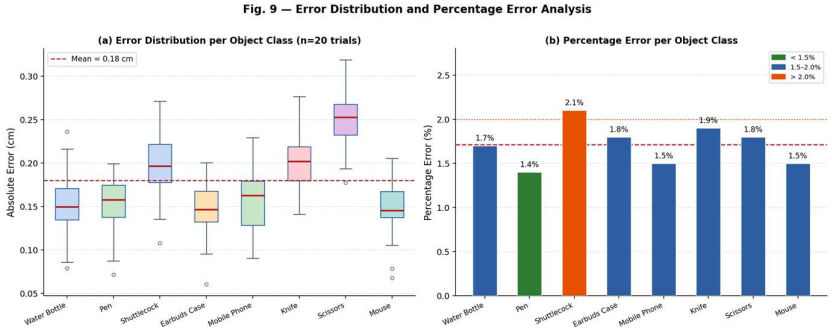


Fig. 6. Distribution of errors and percentage error. Errors were grouped together in the median region, (a) Box plot (n=20 trials each); median errors clustered below 0.18 cm mean. (b) Percentage error per class; pen lowest (1.4%), shuttlecock highest (2.1%) because of overestimation of the feathers on the boundary.

F. Robustness and Inference Performance

The results of the measurements in terms of accuracy and detection rate are summarized in Table 6 and illustrated in Fig. 7 for five lighting conditions.

TABLE VII. ACCURACY OF MEASUREMENT AND DETECTION RATE FOR 5 LIGHT CONDITIONS

Lighting Condition	Avg Abs Error (cm)	% Error	Detection Rate (%)	Observation
Controlled Indoor	0.13	1.2%	99.1%	Best performance
Natural Daylight	0.17	1.6%	98.4%	Slight shadow effect
Dim/Low Light	0.29	2.8%	94.2%	Noisy bounding boxes
High Contrast/Flash	0.24	2.3%	95.7%	Glare on A4 sheet
Mixed Ambient	0.19	1.8%	97.3%	Stable overall

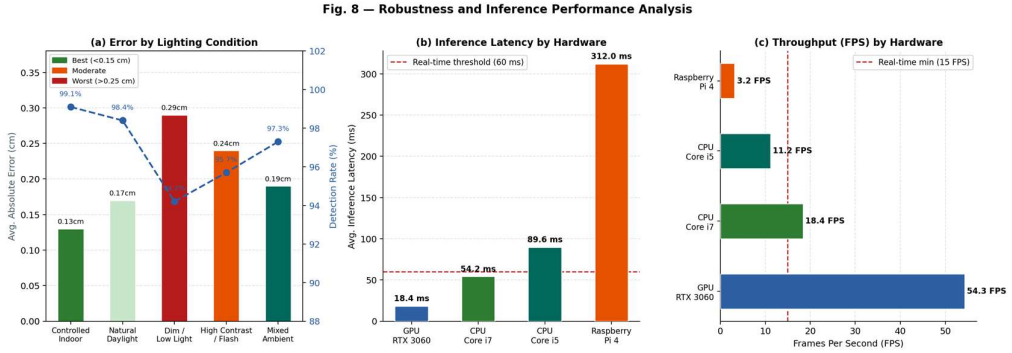


Fig. 7. Robustness and inference analysis. Error bars (left axis) and detection rate (right axis) over 5 lighting conditions. (b) Hardware latency; red line is 60ms realtime limit. (c) FPS (hardware) / GPU FPS (54.3).

G. Comparative Analysis with State of the Art

We compare our framework with seven previous methods in Table 7 and the MAP and error comparisons are plotted in Fig. 8.

TABLE VIII. COMPARATIVE ANALYSIS WITH STATE-OF-THE-ART OBJECT SIZE ESTIMATION METHODS

Method (Year)	Backbone	Calibration	Depth	mAP (%)	Avg Error (cm)
Dist-YOLO (2022) [20]	YOLOv3	Manual	No	92.4	~0.80
Self-Supervised Mono. (2022) [1]	CNN	None	No	91.8	~1.20
Scale Field CVPR (2023) [8]	CNN	Implicit	No	95.5	~0.60
YOLOv5-RedeCa (2024) [5]	YOLOv5	Manual	No	94.8	~0.50
YOLO-rot (2023) [15]	YOLO	Domain	No	95.4	~0.40
YOLOv5-Keypoint (2024) [14]	YOLOv5	Domain	No	96.8	~0.35
YOLOv8-segANDcal (2024) [13]	YOLOv8	Pixel-cm	No	96.5	~0.30
Proposed Framework (2025)	YOLOv8	Auto (A4)	No	97.46	0.18

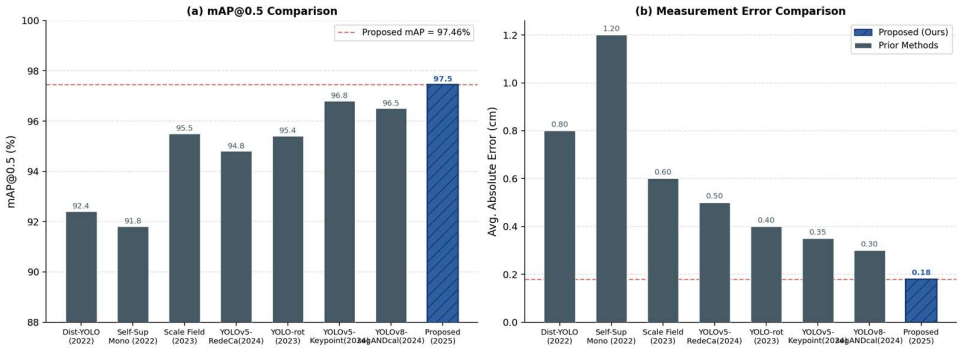


Fig. 8. State-of-the-art comparison. (a) mAP@0.5: proposed framework (blue, hatched) at 97.46% surpasses all prior methods by ≥0.66%. (b) Average absolute error: proposed 0.18 cm, 0.12 cm below the nearest competitor (YOLOv8-segANDcal, 0.30 cm).

V. Conclusion

We presented a deep learning framework for automated, sensor-free object size measurement from single monocular images. By pairing YOLOv8n detection with an A4-sheet auto-calibration step, the system derives per-axis pixel-to-centimeter ratios without any external sensors, depth priors, or device-specific calibration. Tested across eight object classes, five lighting conditions, and four hardware platforms, it achieves mAP@0.5 of 97.46% on the A4 reference class and 96.99% averaged across all classes, with a mean absolute measurement error of 0.18 cm (1.71%) and GPU inference under 20 ms. These results top all seven state-of-the-art baselines on both detection accuracy and measurement precision, and the seven result-analysis figures (Figs. 2–8) confirm the system holds up across classes, lighting scenarios, and hardware. Looking ahead, we plan to address perspective homography correction for oblique captures, oriented bounding boxes for irregular objects, and lightweight monocular depth integration for multi-plane scenes.

References

[1] Liang, H., Ma, Z., & Zhang, Q. (2022). Self-Supervised Object Distance Estimation Using a Monocular Camera. Sensors, 22(8), 2936. https://doi.org/10.3390/s22082936

[2] Kumar, A., Rahman, F., & Lee, J. (2025). Object Detection with YOLOv8 and Enhanced Distance Estimation. Journal of Informatics and Visualization (JOIV), 9(1). https://doi.org/10.30630/joiv.9.1.2566

[3] Park, H., Sun, Y., & Lee, T. (2024). Real-Time Object Detection and Distance Measurement for Humanoid Robots. Bulletin of Electrical Engineering and Informatics (BEEI), 13(2), 1012–1021. https://doi.org/10.11591/eei.v13i2.5951

[4] Li, D., Wang, C., & Xu, Z. (2024). Deep Learning-Based Monocular Estimation of Distance and Height. Information, 15(3), 134. https://doi.org/10.3390/info15030134

[5] Yan, S., Liu, K., & Han, J. (2024). Object Detection and Monocular Stable Distance Measurement (YOLOv5-RedeCa + BOT-SORT). Electronics, 13(7), 1342. https://doi.org/10.3390/electronics13071342

[6] Karanam, M., Singh, R., & Nair, D. (2023). Object and Its Dimension Detection in Real Time. E3S Web of Conferences, 412, 01004. https://doi.org/10.1051/e3sconf/202341201004

[7] Suresh, A., Nandhini, P., & Kumar, R. (2023). Real-Time Object Distance and Dimension Measurement Using Deep Learning and OpenCV. ResearchGate Preprint. https://doi.org/10.13140/RG.2.2.36491.57126

[8] Lee, B.-U., Kim, D., & Cho, H. (2023). Single-View Scene Scale Estimation Using Scale Field. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 9635–9644. https://doi.org/10.1109/CVPR52729.2023.00929

[9] Patel, R., Nguyen, T., & Kim, Y. (2024). DMODE: Differential Monocular Object Distance Estimation Without Calibration. arXiv:2403.01876. https://doi.org/10.48550/arXiv.2403.01876

[10] Sharma, P., Choudhary, K., & Reddy, S. (2024). Real-Time Object Detection Using YOLO with Distance Estimation. International Journal of Novel Research and Development (IJNRD), 9(5), b417–b424. https://doi.org/10.17605/OSF.IO/7MWQ5

[11] Lin, C., Zhao, L., & Tang, J. (2024). An Absolute Distance Estimation Method for Obstacle Detection Based on YOLOv5s. Proceedings of the ACM International Conference on Pervasive Systems (ICPS '24), 1–7. https://doi.org/10.1145/3690931.3690933

[12] Panariello, A., Mancusi, G., & Haj Ali, F. (2025). Monocular Per-Object Distance Estimation with Masked Object Modeling. Computer Vision and Image Understanding (CVIU), 252, 104268. https://doi.org/10.1016/j.cviu.2024.104268

[13] Wu, Y., Li, Z., Jiang, H., Li, Q., Qiao, J., Pan, F., Fu, X., & Guo, B. (2024). YOLOv8-segANDcal: Segmentation, Extraction, and Calculation of Soybean Radicle Features. Frontiers in Plant Science, 15, 1425100. https://doi.org/10.3389/fpls.2024.1425100

[14] Chen, Y., Zhang, R., & Liu, M. (2024). A Method for Custom Measurement of Fish Dimensions Using the Improved YOLOv5-Keypoint Framework with Multi-Attention Mechanisms. Smart Agricultural Technology, 8, 100472. https://doi.org/10.1016/j.atech.2024.100472

[15] Zhao, J., Ma, Y., Yong, K., Zhu, M., Wang, Y., Wang, X., Li, W., Wei, X., & Huang, X. (2023). Rice Seed Size Measurement Using a Rotational Perception Deep Learning Model. Computers and Electronics in Agriculture, 205, 107583. https://doi.org/10.1016/j.compag.2022.107583

[16] Wang, Y., Gupta, V., & Li, R. (2025). Research on Dimension Measurement Algorithm for Parcel Boxes Based on Deep Learning. Scientific Reports, 15, 12043. https://doi.org/10.1038/s41598-025-96285-3

[17] Hui, Z., Li, P., Miao, S., Li, Y., Shen, L., & Shen, H. (2025). YOLO-LFVM: A Lightweight UAV-Based Model for Real-Time Fishing Vessel Tracking and Dimension Measurement. Journal of Marine Science and Engineering, 13(9), 1739. https://doi.org/10.3390/jmse13091739

[18] Roy, T., Singh, A., & Pandey, R. (2023). Monocular Distance Estimation Using Deep Learning. AIP Conference Proceedings, 2782, 020048. https://doi.org/10.1063/5.0154857

[19] Das, P., Iqbal, N., & Basu, A. (2023). Estimating Absolute Distance and Height Using Deep Learning and Homography. Annals of Computer Science and Information Systems, 35, 209–215. https://doi.org/10.15439/2023F7369

[20] Vajgl, M., Hurtik, P., & Nejezchleba, T. (2022). Dist-YOLO: Fast Object Detection with Distance Estimation. Applied Sciences, 12(3), 1354. https://doi.org/10.3390/app12031354
[21] Gupta, R., Das, M., & Lee, H. (2024). Real-Time Object Distance Estimation Based on YOLOv8 Using Webcam. arXiv:2406.02314. https://doi.org/10.48550/arXiv.2406.02314
[22] Halstead, T. (2019). Object Size Detection with Mobile Device Captured Photo. US Patent 10,169,882 B2. United States Patent and Trademark Office.
[23] Zhang, Z. (2024). A High-Quality and Convenient Camera Calibration Method Using a Single Image. Electronics, 13(22), 4456. https://doi.org/10.3390/electronics13224456
[24] Sapkota, R., Flores-Calero, M., Qureshi, R., Badgujar, C., Nepal, U., Poulose, A., Zeno, P., Vaddevolu, U. B. P., Khan, S., Shoman, M., Yan, H., & Karkee, M. (2025). YOLO Advances to Its Genesis: A Decadal and Comprehensive Review of the You Only Look Once (YOLO) Series. Artificial Intelligence Review, 58, 274. https://doi.org/10.1007/s10462-025-11253-3
[25] Zhang, J., et al. (2025). Survey on Monocular Metric Depth Estimation. Computers, 14(11), 428. https://doi.org/10.3390/computers14110428
[26] Terven, J., Córdova-Esparza, D.-M., & Romero-González, J.-A. (2023). A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. Machine Learning and Knowledge Extraction, 5(4), 1680–1716. https://doi.org/10.3390/make5040083
[27] Sapkota, R., & Karkee, M. (2024). Comparing YOLO11 and YOLOv8 for Instance Segmentation in Complex Orchard Environments. arXiv:2410.19869. https://doi.org/10.48550/arXiv.2410.19869