Periodica Polytechnica Mechanical Engineering, 69(4), pp. 347–368, 2025

Real-time Object Positioning in Vibrating Environments Using DeepLabV3+ and ResNet50-based Semantic Segmentation

Aline de Faria Lemos[1]*, Balázs Vince Nagy[1]

[1] Department of Mechatronics, Optics and Mechanical Engineering Informatics, Faculty of Mechanical Engineering, Budapest University of Technology and Economics, Műegyetem rkp. 3., H-1111 Budapest, Hungary
* Corresponding author, e-mail: alinefaria@mogi.bme.hu

Received: 06 August 2025, Accepted: 02 September 2025, Published online: 21 October 2025

Abstract

In environments where both objects and imaging systems experience mechanical vibrations, accurate position measurement poses a significant challenge. Conventional techniques, such as laser-based, contact-based, or image-based methods, often fail under such conditions, particularly when motion artifacts eliminate stable reference points within the image. This work presents a generalized and robust method for object localization under controlled vibration, improving previous approaches by using a single semantic segmentation network (DeepLabV3+ with a ResNet50 backbone) to simultaneously segment both the object and a static reference. This unified architecture eliminates the need for separate models or manual handling of regions of interest. The method retains the use of a local coordinate system anchored at the reference centroid for vibration-resilient position estimation, but extends it to a wider variety of object shapes and configurations. Validation with ten distinct objects under induced vibrations (5–10 Hz) showed reliable performance, with submillimeter localization accuracy (MAE < 0.23 mm, RMSE < 0.29 mm) and strong correlation with ground truth (PCC > 0.99). The system also maintained real-time operation at 94 fps, supporting scalability to dynamic applications. These findings demonstrate that the proposed framework enables fast, precise, and vibration-robust object tracking, supporting applications in automated manufacturing, robotic systems, and industrial quality assurance where vibration has traditionally limited the effectiveness of image-based techniques.

Keywords

semantic segmentation, non-contact measurement, real-time industrial vision, image processing, DeepLabV3+ based semantic segmentation

1 Introduction

Accurate object localization is critical in domains such as manufacturing, robotics, and motion analysis, where precision and reliability directly impact system performance. However, traditional measurement systems, including laser-based, contact-based, and image-based methods, often fail under adverse conditions such as mechanical vibrations, high temperatures, or visual obstructions such as steam or poor illumination [1–5]. In particular, image-based techniques that rely on fixed image borders or stable lighting are highly susceptible to errors when camera or object vibrations distort the scene [6, 7].

Machine learning approaches, particularly convolutional neural network (CNN), have emerged as promising alternatives, demonstrating resilience to varying environmental conditions and successful applications in diverse domains, including manufacturing [8–11], medical imaging [12, 13], and autonomous systems [14–16].

Previous work by [7] introduced a method to mitigate vibration effects on position measurement by incorporating a static reference within the image and using two separate segmentation architectures to segment the object and the reference independently. That approach was specifically applied to steel strips in an industrial rolling mill environment, where vibration parameters could not be controlled or measured. Although effective for its application, this method was limited in scope and generalizability.

This manuscript presents a significant advancement by proposing a generalized and unified semantic segmentation framework that processes the entire image, including both the object and static reference, within a single

Cite this article as: de Faria Lemos, A., Nagy, B. V. "Real-time Object Positioning in Vibrating Environments Using DeepLabV3+ and ResNet50-based Semantic Segmentation", Periodica Polytechnica Mechanical Engineering, 69(4), pp. 347–368, 2025. https://doi.org/10.3311/PPme.41854

network architecture, eliminating the need for separate regions of interest. The approach applies a local coordinate system anchored at the reference centroid, consistent with previous work, but extends it to multiple object types and shapes, demonstrating broader applicability.

Moreover, unlike previous studies conducted in uncontrolled industrial settings, our method is validated in a controlled experimental environment where vibration parameters are independently induced and measured. This controlled setup enables a rigorous assessment and generalization of the proposed positioning method to various scenarios beyond the original application.

Although unified semantic segmentation networks have been employed in other contexts, their integration with a local coordinate system for vibration-robust object positioning in a controlled experimental environment, supporting multiple object types, represents a novel and practical advancement over previous methods. By streamlining the segmentation process through a unified network and validating the approach under controlled vibration conditions, this work advances the robustness and generalizability of image-based object localization. These improvements enable more reliable, real-time measurement and tracking in challenging dynamic environments such as automated manufacturing, robotics, and quality control systems, where vibration-induced errors have traditionally limited performance.

The system utilizes DeepLabV3+ with a ResNet50 backbone as the segmentation network. Controlled experiments involving ten different objects subjected to independent vibrations confirm the robustness of the approach, achieving submillimeter accuracy with a mean absolute error (MAE) of 0.2226 mm, a root mean square error (RMSE) of 0.2860 mm, and real-time processing at 94 fps. These results demonstrate the suitability of the method for accurate and reliable object localization in dynamic and vibration-prone environments.

2 Theoretical basis

2.1 Semantic segmentation for object localization

Semantic segmentation is a core computer vision technique that assigns a class label to every pixel in an image [17], offering fine-grained object localization capabilities in complex scenes [7]. Unlike traditional object detection methods based on bounding boxes, segmentation offers precise boundary delineation [18], which is critical in industrial applications that require submillimeter accuracy under conditions such as vibration, occlusion, or variable lighting [7].

The rise of CNNs has markedly advanced segmentation performance, surpassing conventional edge detection or template matching techniques [19]. CNN-based models are widely adopted across multiple domains, including industrial inspection [20–23], medical imaging [12, 13, 24, 25], autonomous driving [14, 26], aerial imagery analysis [15, 27], and pose estimation [16, 28, 29], due to their robustness against noise, spatial distortions, and domain variability.

Most semantic segmentation models follow an encoder-decoder structure. The encoder extracts hierarchical features by progressively reducing spatial resolution using convolution and pooling layers [30, 31]. However, pooling can eliminate fine details, particularly affecting the segmentation of small or thin structures. To address this, dilated (atrous) convolutions can be employed to increase the receptive field without losing resolution-critical information.

The decoder stage restores spatial resolution through upsampling or transposed convolutions and often incorporates skip connections to retain localization information. U-Net [32, 33], originally proposed for biomedical imaging, exemplifies this design. It combines encoder-decoder architecture with symmetric skip connections, improving segmentation accuracy for small or intricate shapes by reintroducing fine-grained spatial features lost during downsampling.

2.2 Overview of DeepLabV3+ and ResNet50

Pre-trained segmentation architectures leverage large-scale datasets to extract transferable feature representations, reducing the need for extensive task-specific training while maintaining high accuracy. Among these, DeepLabV3+ paired with a ResNet50 backbone offers a compelling balance between performance and computational efficiency [34, 35].

DeepLabV3+ extends earlier DeepLab variants by incorporating atrous spatial pyramid pooling (ASPP), which captures multi-scale contextual information *via* parallel dilated convolutions with different rates [36, 37]. This allows the model to learn from both local details and global structure while preserving spatial resolution. The refined decoder further improves boundary delineation, addressing a common limitation in segmentation tasks involving closely spaced or irregularly shaped objects.

The ResNet50 backbone provides deep feature extraction through residual connections, which mitigates vanishing gradient issues and enhances training convergence in deep networks [36–38]. This backbone supports robust representation learning under varied imaging conditions and has been widely adopted in real-time industrial vision systems due to its accuracy, generalization, and inference speed.

Together, DeepLabV3+ and ResNet50 provide an architecture well suited to high-precision applications such as vibration-robust object localization, where segmentation fidelity and temporal performance are equally critical.

3 Methods

Section 3 presents the proposed methodology for estimating object positions in images affected by camera and object vibrations. Building on previous work, the approach introduces modifications to improve generalization in varying conditions. The methodology includes the experimental setup (Section 3.1), the acquisition and labeling of the datasets (Sections 3.2 and 3.3), and the CNN segmentation model, detailing both the network architecture and the training strategy (Sections 3.4.1 and 3.4.2). Object position estimation (Section 3.5) and performance evaluation (Section 3.6) are also described.

3.1 Experimental setup (image acquisition)

Image acquisition was performed in a controlled laboratory environment to ensure consistency and minimize external interference. The setup included a black fabric background, a fixed white square reference marker, and a set of white geometric objects: two spheres, three cubes, one pyramid, two cones, and two yellow cylinders of different sizes (see Table 1 for dimensions).

To simulate vibration conditions, the camera and object were mounted on independent shaker tables operating at distinct frequencies. The camera was positioned at a fixed distance of 0.8 m from the object using rigid dark gray supports. The camera table oscillated at 5 Hz with a peak amplitude of 0.85 mm, while the object table vibrated at 10 Hz with amplitudes of 1.0 mm and 1.7 mm.

The images were captured using a Basler acA1600-20uc color camera equipped with a Sony ICX274 CCD sensor (1624 × 1234 px, 4.4 × 4.4 µm pixel size, global shutter) at 40 fps *via* a USB 3.0 interface. A Basler C125-1620-5M C-mount lens (16 mm, f/2.0) was used.

The camera was mounted on a Brüel and Kjær (LDS) V455 M5 shaker powered by an LDS PA1000L amplifier, and the object on a V406 M4 shaker driven by an LDS PA100E amplifier. The vibration parameters were validated using a PCB-352C23 accelerometer connected to an NI cDAQ-9174 system with an NI 9234 data acquisition card (sampling at 51.2 kHz). The complete experimental setup is illustrated in Fig. 1.

Table 1 Description of the objects used in the experiment

Object code	Object description	Height (mm)	Width (mm)	Diameter (mm)
1	Larger white sphere	–	–	150
2	Larger white cube	97	97	–
3	Medium white cube	80	80	–
4	White pyramid	200	120	–
5	Larger white cone	250	–	90
6	Small white sphere	–	–	100
7	Small white cone	120	–	67
8	Smaller white cube	60	60	–
9	Larger yellow cylinder	160	60	–
10	Smaller yellow cylinder	110	60	–

3.2 Dataset

The dataset consists of images of ten distinct objects with varying geometries and sizes, as detailed in Table 1. Representative examples are shown in Fig. 2.

A total of 682 images were acquired under varying vibration conditions. To increase variability and improve model generalization, data augmentation was applied, which included random rotations and scaling, resulting in an expanded dataset of 4,092 images. Examples of augmented images are shown in Fig. 3.

3.3 Labeling

Pixel-wise annotations were created using Labelme [39], which generates labeled regions in JavaScript Object Notation (JSON) format. These annotations were converted into PNG mask images, where each pixel was assigned one of three classes: background (0), object (1) or reference (2). The background includes all non-target regions; the object denotes the element of interest for tracking, and the reference is a static feature segmented to enable relative position analysis.

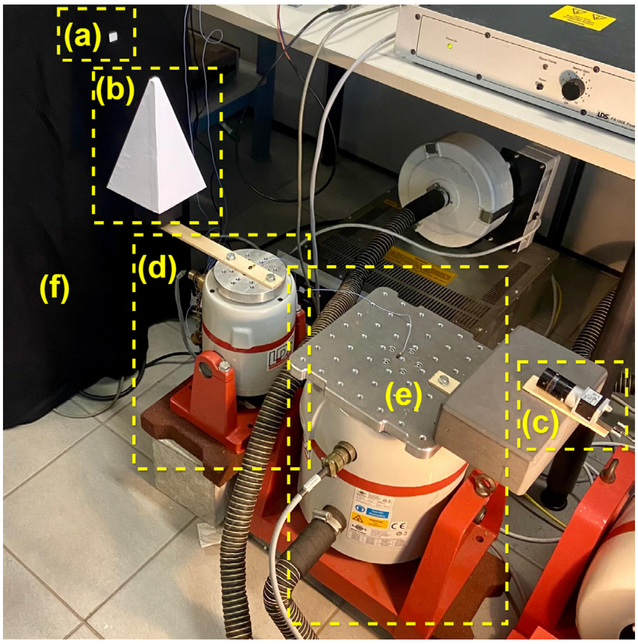


Fig. 1 Experimental setup for image acquisition, including: (a) the reference object; (b) the object; (c) the camera; (d) shaker table of the object; (e) shaker table of the camera; and (f) the black background

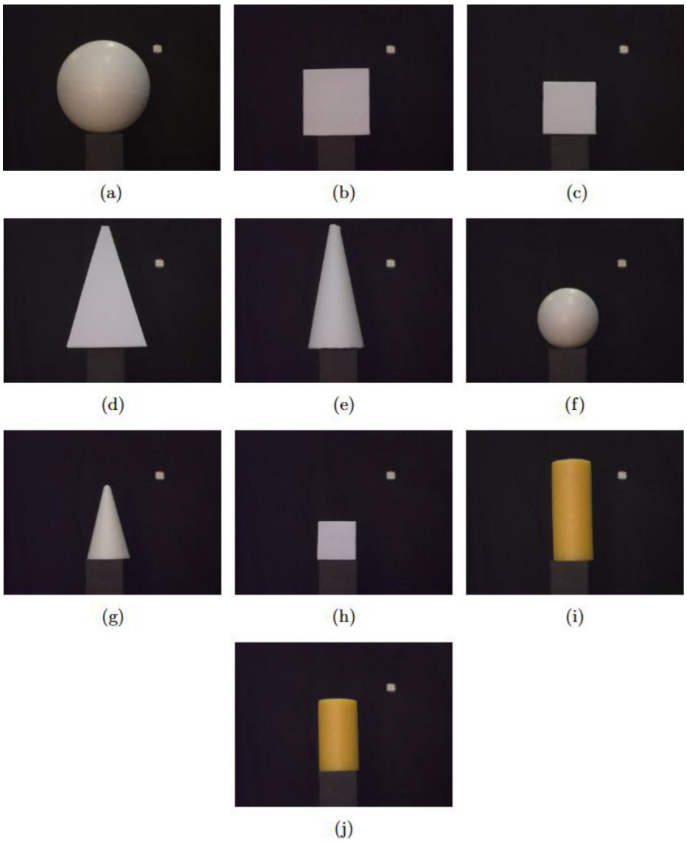


Fig. 2 Samples of acquired images, one per object used in the dataset: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2

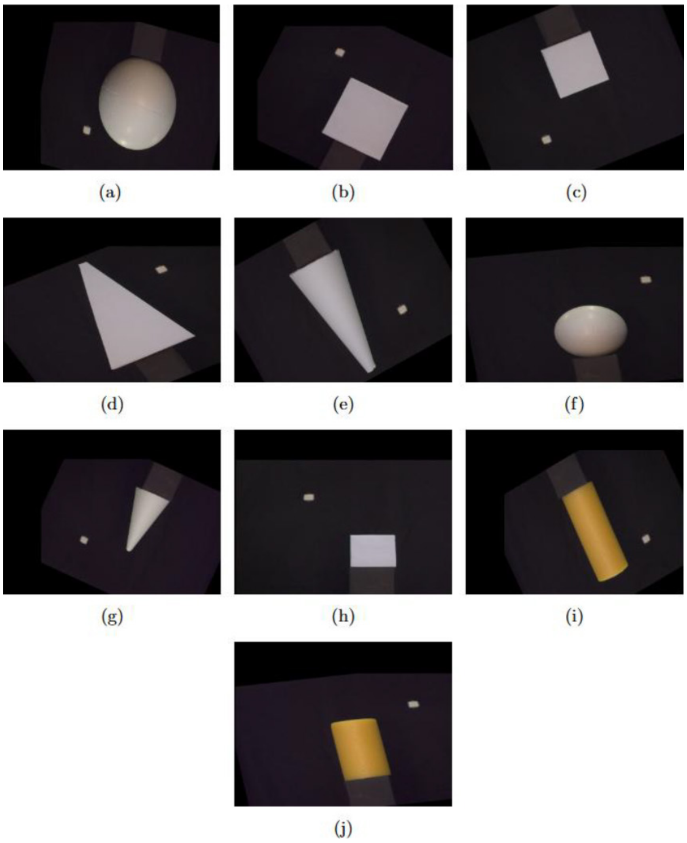


Fig. 3 Samples of augmented images, one per object used in the dataset: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2

3.4 CNN model

3.4.1 Network architecture

This work adopts DeepLabV3+ with a ResNet50 encoder pre-trained on ImageNet as the primary architecture for semantic segmentation, selected for its strong performance and ability to capture multiscale contextual information through ASPP and encoder-decoder refinement. The model is used as implemented in the segmentation models pytorch library, which provides a high-level interface based on the original design by Chen et al. [36].

To evaluate performance relative to alternative designs, ten additional models were considered: eight custom lightweight CNN networks and two pretrained benchmarks. The custom models follow a simplified encoder-decoder structure, varying in the number of convolutional and transposed convolutional layers (2 or 3), and in the number of filters per layer (8, 16, 32, or 64). The encoder uses 3 × 3 convolutions, with dilated convolutions (dilation = 2) in deeper layers to expand the receptive field while preserving spatial detail. The decoder restores spatial resolution using transposed convolutions, followed by a 1 × 1 convolution for pixel-wise class prediction. Rectified Linear Unit (ReLU) activations are applied after all convolutional and transposed convolutional layers, except the final output layer. An example custom architecture is shown in Fig. 4, and the complete set of configurations is detailed in Table 2. For the downsampling path, the first convolutional layer uses a 3 × 3 kernel with padding 1, and the deeper layers use 3 × 3 kernels with padding 1 and dilation 2; for the upsampling path, the transposed convolution layers use 2 × 2 kernels with stride 2, except the final layer, which uses a 1 × 1 kernel for class prediction.

The two additional benchmark models include U-Net with a ResNet34 encoder [40] and FPN with a ResNet50 encoder [41], both initialized with ImageNet weights.

3.4.2 Training strategy

Training was conducted using PyTorch on a NVIDIA GeForce GTX 1080 GPU (8GB VRAM) with CUDA acceleration. Input images were downsampled by 50% to reduce computational cost. The dataset was split into training (60%), validation (20%), and test (20%) sets.

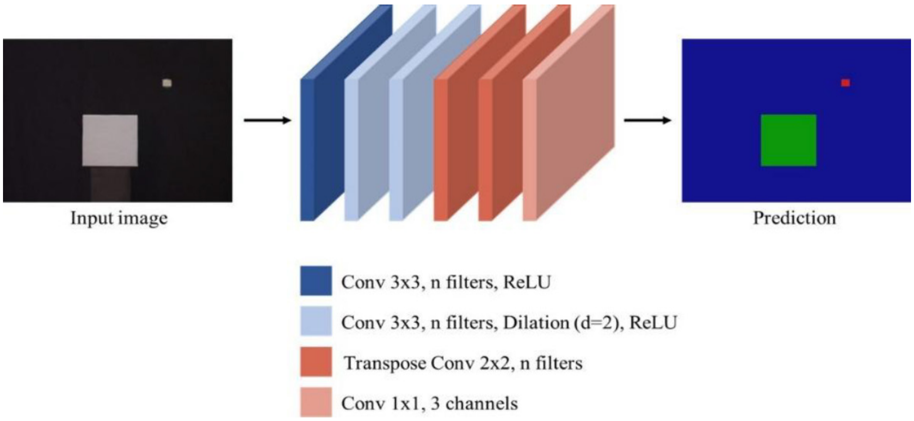

Input image
Prediction
Conv 3x3, n filters, ReLU
Conv 3x3, n filters, Dilation (d=2), ReLU
Transpose Conv 2x2, n filters
Conv 1x1, 3 channels

Fig. 4 Schematic simplified illustration of one of the custom lightweight CNNs architecture with 3 convolution layers and *n* filters

Table 2 Summary of architectural configurations for the 8 CNNs evaluated, including convolution and transposed convolution layers*

Model	Architecture	No. of filters	Downsampling			Upsampling		
			Layer 1 Conv2D	Layer 2 Conv2D	Layer 3 Conv2D	Layer 1 Conv2D Transp	Layer 2 Conv2D Transp	Layer 3 Conv2D
1	2	8	✓	–	✓	✓	–	✓
2	2	16	✓	–	✓	✓	–	✓
3	2	32	✓	–	✓	✓	–	✓
4	2	64	✓	–	✓	✓	–	✓
5	3	8	✓	✓	✓	✓	✓	✓
6	3	16	✓	✓	✓	✓	✓	✓
7	3	32	✓	✓	✓	✓	✓	✓
8	3	64	✓	✓	✓	✓	✓	✓

* A check mark (✓) indicates that the corresponding layer is included in the model, while a dash (–) indicates that it is not

To address class imbalance, weighting factors of 0.1 (background), 1.5 (object), and 5.0 (reference) were applied. Focal loss was employed to emphasize hard-to-classify samples. The models were optimized using the Adam optimizer with a learning rate and a weight decay of 1 × 10^{-4}. Mixed precision training and gradient scaling were used to improve efficiency and numerical stability. Validation loss was monitored after each epoch to evaluate generalization.

The pre-trained models followed the same training pipeline. The input images were padded to ensure that the dimensions are divisible by 32, according to architectural constraints. All training parameters and class weights were kept constant in all models to isolate the impact of architectural differences.

3.5 Estimation of object position

The location of the object was defined in two coordinate systems: a global system X0Y, originating in the bottom left corner of the image, and a local system *xcyc*, centered at the reference centroid *c* (see Fig. 5). In Fig. 5, red dots mark the centroids of the object and the static reference, which are used to determine the object's position relative to the reference. The position of the object relative to the reference was expressed as coordinates (x, y), as defined in Eqs. (1) and (2):

$$x_{obj} = X_{obj} - X_{ref}, \tag{1}$$

$$y_{obj} = Y_{obj} - Y_{ref}. \tag{2}$$

By anchoring the local coordinate system to a static scene element, this approach enables a consistent mapping between the image and real-world coordinates, supporting precise localization relative to a fixed point.

3.6 Evaluation metrics

The proposed method was evaluated through both quantitative and qualitative analyses, comprising four components: semantic segmentation assessment (Section 3.6.1), visual inspection of predictions (Section 3.6.2), measurement system evaluation (Section 3.6.3), and computational efficiency.

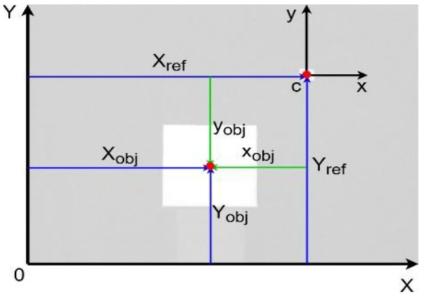


Fig. 5 Illustration of an acquired image with the global coordinate system X0Y (origin at the bottom-left corner) and the local coordinate system *xcyc* (centered at the static reference)

3.6.1 Semantic segmentation evaluation

Segmentation performance was quantified using four standard pixel-level metrics: Intersection over Union (IoU), Recall, F1 score, and Specificity [7], computed independently for the background, object, and reference classes.

1. IoU: measures the overlap between predicted and ground truth masks, defined as:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{3}$$

2. Recall: indicates the proportion of true positive pixels correctly identified:

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{4}$$

3. F1 score: harmonic mean of precision and recall:

$$\mathrm{F1} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{5}$$

4. Specificity: reflects the proportion of true negative pixels correctly identified:

$$\mathrm{Specificity} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}. \tag{6}$$

Here, TP, FP, FN, and TN denote true positives, false positives, false negatives, and true negatives, respectively.

3.6.2 Visual inspection of predictions

The qualitative evaluation was performed by overlaying the predicted masks on the original input images. This visual assessment complements quantitative metrics by revealing systematic misclassifications and evaluating model robustness under varying conditions.

3.6.3 Measurement system evaluation

The accuracy of the measurement system was evaluated using the Pearson correlation coefficient (PCC), the MAE, and the RMSE.

The PCC [42], defined in Eq. (7), quantifies the linear correlation between the predicted and ground truth coordinates:

$$r = \frac{\sum\left(z_{pred,i} - \bar{z}_{pred}\right)\left(z_{true,i} - \bar{z}_{true}\right)}{\sqrt{\sum\left(z_{pred,i} - \bar{z}_{pred}\right)^2}\sqrt{\sum\left(z_{true,i} - \bar{z}_{true}\right)^2}}. \tag{7}$$

In Eq. (7) $z_{pred,i}$ and $z_{true,i}$ denote the predicted and ground truth coordinates, respectively, and the overbars represent mean values. A PCC close to 1 indicates a strong positive correlation.

The MAE [43], shown in Eq. (8), measures the average absolute difference between predictions and ground truth:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|z_{pred,i} - z_{true,i}\right|. \tag{8}$$

Lower MAE values indicate higher measurement accuracy.

The RMSE [43], defined in Eq. (9), emphasizes larger errors by squaring the deviations:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(z_{pred,i} - z_{true,i}\right)^2}. \tag{9}$$

Compared to MAE, RMSE is more sensitive to large deviations, making it especially useful for identifying substantial prediction errors.

4 Results and discussions

Section 4 presents the evaluation of the proposed system through three key analyses: semantic segmentation evaluation (Section 4.1), which assesses the segmentation performance using quantitative metrics; visual inspection of predictions (Section 4.2), which provides a qualitative assessment of the predicted segmentation masks; and measurement system evaluation (Section 4.3), which examines the accuracy of object localization within the defined coordinate system.

4.1 Semantic segmentation evaluation

Segmentation performance was assessed using IoU, Recall, F1 score, and Specificity, which respectively evaluate region overlap, sensitivity to true positives, overall classification performance, and background rejection. Results for custom and pre-trained models are summarized in Tables 3 and 4.

Table 3 presents results for eight custom lightweight CNN. Background (Class 0) and object (Class 1) were segmented with high accuracy (IoU > 0.90, F1 > 0.94). Reference segmentation (Class 2) showed significantly lower performance (IoU: 0.0000–0.4980), likely due to its small size and slight blur from camera focus. The best custom model (architecture 8: three convolutional layers, 64 filters) achieved an IoU of 0.4980 and an F1 score of 0.6452 for Class 2.

To address the poor performance in Class 2, pre-trained models were evaluated (Table 4). DeepLabV3+ with ResNet50 achieved the highest accuracy (IoU: 0.7219, F1: 0.8353), followed by U-Net with ResNet34 (IoU: 0.7012). In contrast, FPN with ResNet50 performed poorly (IoU: 0.1426).

DeepLabV3+ with ResNet50 was selected for final segmentation due to its superior performance across all classes. Its multiscale feature extraction enabled improved accuracy in small and poorly defined regions, supporting robust object and reference localization.

4.2 Visual inspection of predictions

To complement the quantitative metrics (Section 4.1), a qualitative evaluation was conducted through visual inspection of the predicted segmentation masks. This

Table 3 Semantic segmentation metrics for the eight custom lightweight CNNs architectures

	Model	1	2	3	4	5	6	7	8
	Architecture	2	2	2	2	3	3	3	3
	Filters	8	16	32	64	8	16	32	64
Class 0	IoU	0.9929	0.9903	0.9658	0.9905	0.9893	0.9904	0.9904	0.9898
	Recall	0.9931	0.9905	0.9658	0.9907	0.9895	0.9905	0.9905	0.9899
	F1	0.9964	0.9951	0.9825	0.9952	0.9947	0.9952	0.9952	0.9949
	Spec.	0.9980	0.9981	0.9995	0.9985	0.9988	0.9986	0.9989	0.9989
Class 1	IoU	0.9231	0.9017	0.7456	0.9039	0.8955	0.9034	0.9031	0.9015
	Recall	0.9980	0.9978	0.9990	0.9952	0.9988	0.9986	0.9972	0.9932
	F1	0.9599	0.9480	0.8523	0.9490	0.9446	0.9490	0.9487	0.9479
	Spec.	0.9916	0.9890	0.9645	0.9898	0.9880	0.9890	0.9893	0.9896
Class 2	IoU	0.0014	0.0117	0.0806	0.3227	0.0000	0.0000	0.1479	0.4980
	Recall	0.0014	0.0119	0.0846	0.3774	0.0000	0.0000	0.1799	0.6706
	F1	0.0027	0.0222	0.1345	0.4462	0.0000	0.0000	0.2233	0.6452
	Spec.	0.9999	0.9999	0.9999	0.9995	1.0000	1.0000	0.9998	0.9991

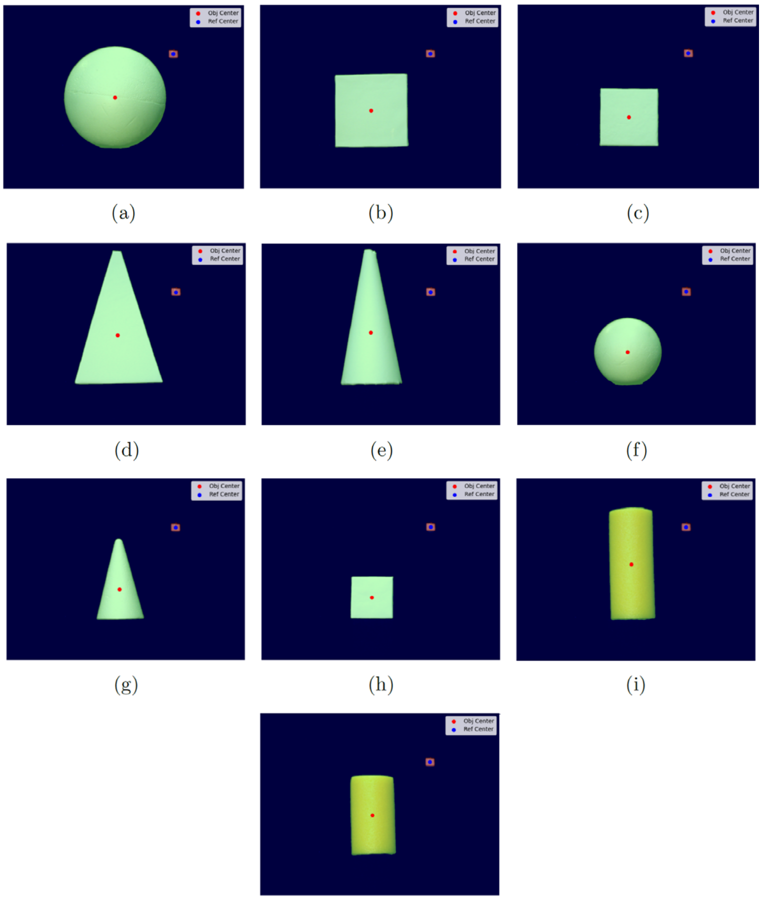
Table 4 Semantic segmentation metrics for the pre-trained models (U-Net with ResNet34, DeepLabV3+ with ResNet50, and FPN with ResNet50)

		U-Net with ResNet34	DeepLabV3+ with ResNet50	FPN with ResNet50
Class 0	IoU	0.9963	0.9685	0.9905
	Recall	0.9963	0.9963	0.9905
	F1	0.9981	0.9837	0.9952
	Spec.	0.9990	0.9966	0.9997
Class 1	IoU	0.9716	0.9684	0.8909
	Recall	0.9998	0.9996	0.9997
	F1	0.9856	0.9837	0.9411
	Spec.	0.9969	0.9966	0.9890
Class 2	IoU	0.7012	0.7219	0.1426
	Recall	0.9987	0.9971	0.1480
	F1	0.8227	0.8353	0.2197
	Spec.	0.9994	0.9994	0.9999

assessment offers direct insight into the model's ability to delineate object and reference regions.

Fig. 6 presents representative results for each of the 10 objects, using the DeepLabV3+ model with ResNet50, which demonstrated superior performance, especially in Class 2. Each image in Fig. 6 shows overlays of the predicted masks for background, object, and static reference, along with markers (red and blue dots) indicating the predicted centers of the object and reference.

The visual results show strong agreement between predicted and actual regions, with high spatial alignment of the predicted centers and their ground truth locations. These



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. 6 Representative segmentation results for each of the 10 objects in the experiment: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2

findings support the quantitative results in Table 4 and confirm the model's robustness across diverse object instances.

Additional visualizations for all tested architectures are provided in Appendix A.

4.3 Measurement system evaluation

Measurement accuracy was assessed by comparing predicted and ground truth object centers using PCC, MAE, and RMSE (Table 5).

For the static reference (local coordinate origin), high agreement was observed: PCC reached 0.9636 (*x*) and 0.9573 (*y*), with MAE below 0.09 mm and RMSE near 0.11 mm for both axes, confirming the submillimeter accuracy.

In the global coordinate system, the predictions of the objects showed even stronger correlation, PCCs of 0.9979 (*x*) and 0.9999 (*y*), with MAE values of 0.1235 mm (*x*) and 0.1940 mm (*y*). The corresponding RMSE values were 0.1544 mm and 0.2522 mm.

In the local frame, accuracy remained high (PCC: 0.9971–0.9998), with MAE below 0.23 mm and RMSE below 0.29 mm for both axes. These results confirm consistent localization performance across coordinate systems.

Relative to object size, these errors are negligible. For instance, the largest object (200 mm × 120 mm pyramid) and the smallest (60 mm cube) both exhibited errors several orders of magnitude smaller than their dimensions, demonstrating the scalability of the system.

Figs. 7 and 8 show a close alignment between predicted and ground truth coordinates in the local frame. Figs. 9 and 10 illustrate absolute errors, with horizontal errors below 0.7 mm and vertical errors peaks near 1.0 mm, consistent with slightly higher MAE and RMSE (y-axis).

Additional error analyzes in Appendix B support these findings for both the reference and object in global coordinates.

Overall, the system achieves submillimeter localization accuracy across both coordinate systems and operates in real-time at 94 FPS, making it suitable for high-speed, precision-demanding applications.

4.4 Limitations and future work

While the proposed method demonstrates submillimeter accuracy and real-time performance, several limitations must be considered. The experiments were conducted under controlled laboratory conditions, with vibration frequencies

Table 5 Evaluation metrics for the measurement system, including PCC, MAE, and RMSE

	Object		Static ref.		Object	
	Global coordinate		Global coordinate		Local coordinate	
Metric	*x*	*y*	*x*	*y*	*x*	*y*
PCC	0.9979	0.9999	0.9636	0.9573	0.9971	0.9998
MAE (mm)	0.1235	0.1940	0.0882	0.0883	0.1370	0.2226
RMSE (mm)	0.1544	0.2522	0.1118	0.1126	0.1726	0.2860

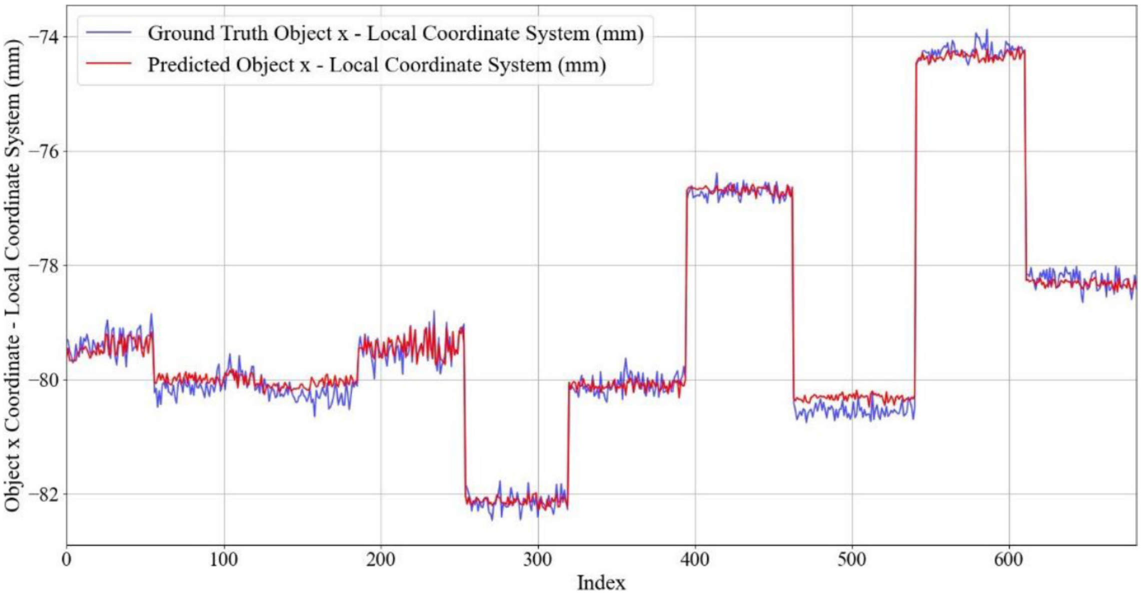


Fig. 7 Comparison of the x-coordinate of the center of the object in the local coordinate system of the original images

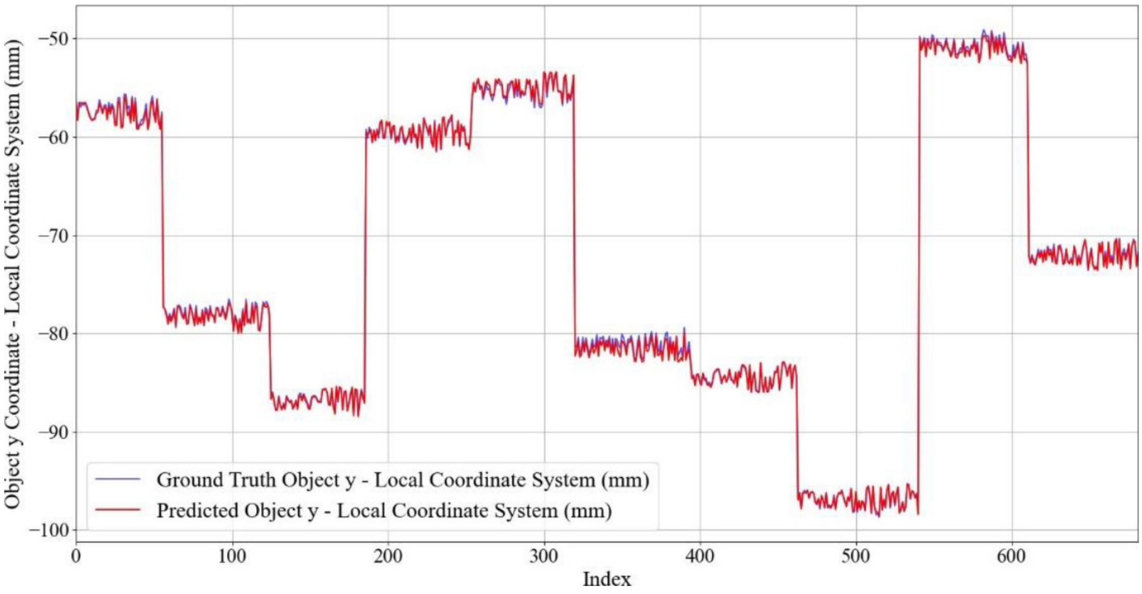


Fig. 8 Comparison of the y-coordinate of the center of the object in the local coordinate system of the original images

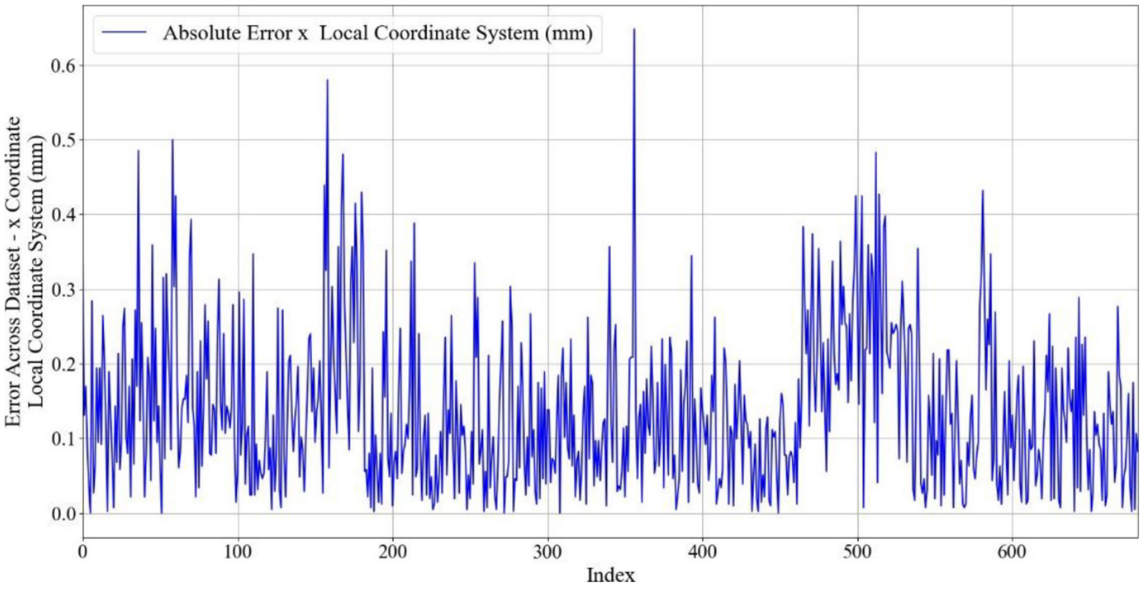


Fig. 9 Absolute error of the x-coordinate of the center of the object in the local coordinate system

restricted to 10 Hz to comply with the Nyquist criterion and avoid aliasing at the chosen camera frame rate. Although this ensured reliable validation, it does not fully reflect the complexity of real-world industrial environments, where higher-frequency vibrations, variable illumination, occlusions, and cluttered backgrounds are common. In addition, the evaluation was limited to a single-camera setup with a fixed reference marker, which does not capture scenarios involving partial reference occlusion or multi-camera integration. Finally, computational constraints restricted the study to lightweight custom networks and selected pretrained models; while adequate for real-time performance, deeper or more advanced architectures may further enhance accuracy at the cost of higher computational demand.

Future research should therefore explore the applicability of the method under more challenging conditions, including higher vibration frequencies, uncontrolled lighting, and occluded or cluttered scenes. Extensions to multi-camera systems and adaptive reference strategies could broaden the robustness of the approach, while investigations into more sophisticated segmentation architectures may improve accuracy provided real-time feasibility is maintained. Together, these directions would extend the framework toward deployment in realistic industrial environments.

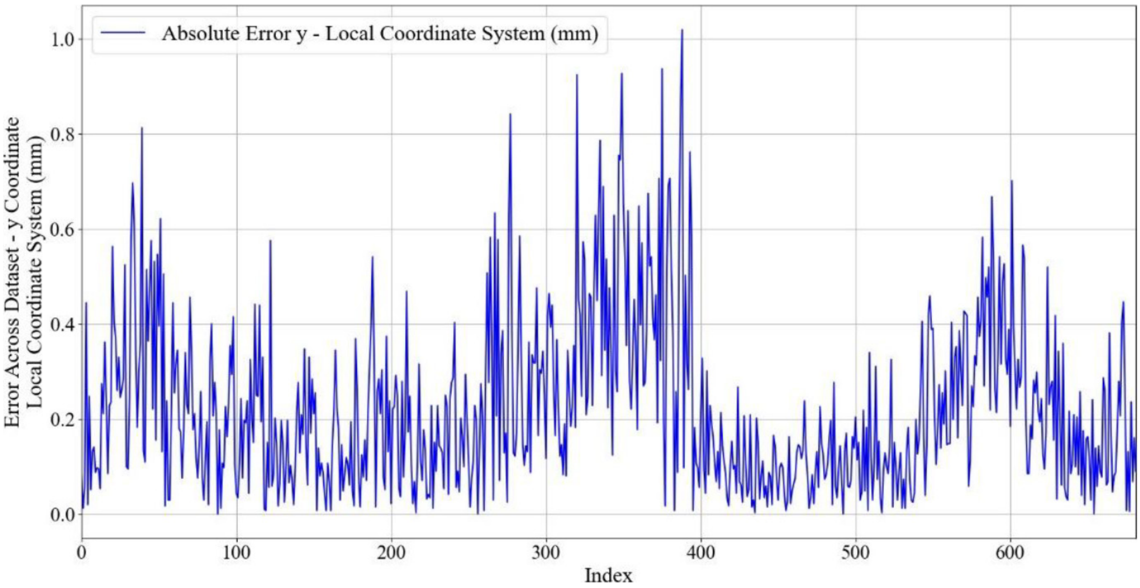


Fig. 10 Absolute error of the y-coordinate of the center of the object in the local coordinate system

5 Conclusions

This study proposed a generalized, real-time method for vibration-robust object localization using DeepLabV3+ with a ResNet50-based semantic segmentation network. By jointly segmenting both the object and a static reference in a unified architecture, the method eliminates the need for region-of-interest preprocessing and enables the use of a local coordinate system anchored at the reference point for robust positional measurement.

The proposed generalization was achieved by evaluating the method on ten distinct objects, under independently induced object and camera vibrations in a controlled laboratory environment. This contrasts with previous studies that addressed vibration related measurement issues for specific industrial use cases and single object types. The controlled setup allowed a systematic assessment of the robustness, segmentation precision, and localization precision of the method.

Validated through quantitative and visual evaluation, the system achieved submillimeter localization accuracy and real-time performance at 94 fps. Importantly, the camera frame rate was selected to exceed twice the highest vibration frequency in the system, ensuring compliance with the Nyquist criterion and preventing aliasing.

Due to computational performance constraints, the study focused on lightweight custom architectures and pretrained networks; more complex or deeper custom models were not explored but may offer further gains if real-time requirements can be met.

The findings support the scalability and applicability of the method in vibrating industrial settings where traditional measurement approaches can fail. Future work may extend this approach to real-world production lines with increased complexity, including occlusions, variable lighting, and higher-frequency vibrations, as well as explore multi-camera setups and advanced post-processing for further performance gains.

References

[1] Demilia, G., Gaspari, A., Natale, E. "Measurements for Smart Manufacturing in an Industry 4.0 Scenario A Case-Study on A Mechatronic System", In: 2018 Workshop on Metrology for Industry 4.0 and IoT, Brescia, Italy, 2018, pp. 1–5. ISBN 978-1-5386-2498-2 https://doi.org/10.1109/METROI4.2018.8428341

[2] Andreev, D. V. "The role of measurement and the significance of metrology in industrial production", Journal of Physics: Conference Series, 2373(5), 052031, 2022. https://doi.org/10.1088/1742-6596/2373/5/052031

[3] Zha, Y., Luo, Y., Ding, Y., Yu, T. "Research on non-contact measurement based on machine vision", In: 2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2022, pp. 1956–1960. ISBN 978-1-6654-3186-6 https://doi.org/10.1109/ITOEC53115.2022.9734442

[4] Saariluoma, H., Piiroinen, A., Unt, A., Hakanen, J., Rautava, T., Salminen, A. "Overview of Optical Digital Measuring Challenges and Technologies in Laser Welded Components in EV Battery Module Design and Manufacturing", Batteries, 6(3), 47, 2020. https://doi.org/10.3390/batteries6030047

[5] Chalupka, U., Rothe, H. "Challenges in system design of an optronic, laser-based measurement system for projectile trajectories", In: Novel Optical Systems Design and Optimization XIX, San Diego, CA, USA, 2016, 99480F. ISBN 9781510602885
https://doi.org/10.1117/12.2236830

[6] Terzić, S., Lazarević, D., Nedić, B., Šarkoćević, Ž., Dedić, J. "Machining contact and non-contact inspection technologies in industrial application", Journal of Production Engineering, 21(1), pp. 55–60, 2018.
https://doi.org/10.24867/JPE-2018-01-055

[7] Lemos, A. d. F., da Silva, L. A. R., Nagy, B. V. "Automatic monitoring of steel strip positioning error based on semantic segmentation", The International Journal of Advanced Manufacturing Technology, 110(11), pp. 2847–2860, 2020.
https://doi.org/10.1007/s00170-020-05859-w

[8] Lee, S. J., Yun, J. P., Koo, G., Kim, S. W. "End-to-end recognition of slab identification numbers using a deep convolutional neural network", Knowledge-Based Systems, 132, pp. 1–10, 2017.
https://doi.org/10.1016/j.knosys.2017.06.017

[9] Ferguson, M., Ak, R., Lee, Y.-T. T., Law, K. H. "Automatic localization of casting defects with convolutional neural networks", In: 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 2017, pp. 1726–1735. ISBN 978-1-5386-2716-7
https://doi.org/10.1109/BigData.2017.8258115

[10] Xu, Z.-W., Liu, X.-M., Zhang, K. "Mechanical Properties Prediction for Hot Rolled Alloy Steel Using Convolutional Neural Network", IEEE Access, 7, pp. 47068–47078, 2019.
https://doi.org/10.1109/ACCESS.2019.2909586

[11] Masci, J., Meier, U., Ciresan, D., Schmidhuber, J., Fricout, G. "Steel defect classification with Max-Pooling Convolutional Neural Networks", In: The 2012 International Joint Conference on Neural Networks (IJCNN), Brisbane, Australia, 2012, pp. 1–6. ISBN 978-1-4673-1488-6
https://doi.org/10.1109/IJCNN.2012.6252468

[12] Rashed, E. A., Gomez-Tames, J., Hirata, A. "End-to-end semantic segmentation of personalized deep brain structures for non-invasive brain stimulation", Neural Networks, 125, pp. 233–244, 2020.
https://doi.org/10.1016/j.neunet.2020.02.006

[13] Pham, D. L., Xu, C., Prince, J. L. "Current Methods in Medical Image Segmentation", Annual Review of Biomedical Engineering, 2(1), pp. 315–337, 2000.
https://doi.org/10.1146/annurev.bioeng.2.1.315

[14] Ess, A., Mueller, T., Grabner, H., van Gool, L. "Segmentation-Based Urban Traffic Scene Understanding", In: Proceedings of the British Machine Vision Conference, London, UK, 2009, pp. 84.1–84.11. ISBN 1-901725-39-1
https://doi.org/10.5244/C.23.84

[15] Munawar, H. S., Ullah, F., Qayyum, S., Khan, S. I., Mojtahedi, M. "UAVs in Disaster Management: Application of Integrated Aerial Imagery and Convolutional Neural Network for Flood Detection", Sustainability, 13(14), 7547, 2021.
https://doi.org/10.3390/su13147547

[16] Zheng, C., Wu, W., Chen, C., Yang, T., Zhu, S., Shen, J., Kehtarnavaz, N., Shah, M. "Deep Learning-based Human Pose Estimation: A Survey", ACM Computing Surveys, 56(1), 11, 2023.
https://doi.org/10.1145/3603618

[17] Roberts, G., Haile, S. Y., Sainju, R., Edwards, D. J., Hutchinson, B., Zhu, Y. "Deep Learning for Semantic Segmentation of Defects in Advanced STEM Images of Steels", Scientific Reports, 9(1), 12744, 2019.
https://doi.org/10.1038/s41598-019-49105-0

[18] Huang, S.-Y., Hsu, W.-L., Hsu, R.-J., Liu, D.-W. "Fully Convolutional Network for the Semantic Segmentation of Medical Images: A Survey", Diagnostics, 12(11), 2765, 2022.
https://doi.org/10.3390/diagnostics12112765

[19] Chike, O. G., Ahmad, N., Faiz Wan Ali, W. F. "Neural network prediction of thermal field spatiotemporal evolution during additive manufacturing: an overview", The International Journal of Advanced Manufacturing Technology, 134(5), pp. 2107–2128, 2024.
https://doi.org/10.1007/s00170-024-14256-6

[20] Xiao, L., Lu, M., Huang, H. "Detection of powder bed defects in selective laser sintering using convolutional neural network", The International Journal of Advanced Manufacturing Technology, 107(5), pp. 2485–2496, 2020.
https://doi.org/10.1007/s00170-020-05205-0

[21] Liu, L.-J., Zhang, Y., Karimi, H. R. "Resilient machine learning for steel surface defect detection based on lightweight convolution", The International Journal of Advanced Manufacturing Technology, 134(9), pp. 4639–4650, 2024.
https://doi.org/10.1007/s00170-024-14403-z

[22] Huang, W.-T., Yang, S.-C., Chou, F.-I., Chou, J.-H. "Automatic recognition of grinding quality of titanium alloy based on the convolutional neural network", The International Journal of Advanced Manufacturing Technology, 135(7), pp. 3941–3959, 2024.
https://doi.org/10.1007/s00170-024-14692-4

[23] Wang, Z., Yin, Y., Yin, R. "Multi-tasking atrous convolutional neural network for machinery fault identification", The International Journal of Advanced Manufacturing Technology, 124(11), pp. 4183–4191, 2023.
https://doi.org/10.1007/s00170-022-09367-x

[24] Zhang, Z., Wu, C., Coleman, S., Kerr, D. "DENSE-INception U-net for medical image segmentation", Computer Methods and Programs in Biomedicine, 192, 105395, 2020.
https://doi.org/10.1016/j.cmpb.2020.105395

[25] Almotairi, S., Kareem, G., Aouf, M., Almutairi, B., Salem, M. A.-M. "Liver Tumor Segmentation in CT Scans Using Modified SegNet", Sensors, 20(5), 1516, 2020.
https://doi.org/10.3390/s20051516

[26] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B. "The Cityscapes Dataset for Semantic Urban Scene Understanding", In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 3213–3223. ISBN 978-1-4673-8852-8
https://doi.org/10.1109/CVPR.2016.350

[27] Ševo, I., Avramović, A. "Convolutional Neural Network Based Automatic Object Detection on Aerial Images", IEEE Geoscience and Remote Sensing Letters, 13(5), pp. 740–744, 2016.
https://doi.org/10.1109/LGRS.2016.2542358

[28] Wang, Z., Fan, J., Jing, F., Liu, Z., Tan, M. "A pose estimation system based on deep neural network and ICP registration for robotic spray painting application", The International Journal of Advanced Manufacturing Technology, 104(1), pp. 285–299, 2019.
https://doi.org/10.1007/s00170-019-03901-0

[29] Hasan, M. K., Calvet, L., Rabbani, N., Bartoli, A. "Detection, segmentation, and 3D pose estimation of surgical tools using convolutional neural networks and algebraic geometry", Medical Image Analysis, 70, 101994, 2021.
https://doi.org/10.1016/j.media.2021.101994

[30] Badrinarayanan, V., Kendall, A., Cipolla, R. "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12), pp. 2481–2495, 2017.
https://doi.org/10.1109/TPAMI.2016.2644615

[31] Lin, G., Milan, A., Shen, C., Reid, I. "RefineNet: Multi-path Refinement Networks for High-Resolution Semantic Segmentation", In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 5168–5177. ISBN 978-1-5386-0458-8
https://doi.org/10.1109/CVPR.2017.549

[32] Huang, L., Miron, A., Hone, K., Li, Y. "Segmenting Medical Images: From UNet to Res-UNet and nnUNet", In: 2024 IEEE 37th International Symposium on Computer-Based Medical Systems (CBMS), Guadalajara, Mexico, 2024, pp. 483–489. ISBN 979-8-3503-8473-4
https://doi.org/10.1109/CBMS61543.2024.00086

[33] Chen, J., Wang, B., Lin, Y., Chen, X., Jiang, Q., Cui, C. "Efficient-Unet: Intelligent identification of abrasive grain on the entire surface of monolayer brazing wheel based on encoder–decoder network", The International Journal of Advanced Manufacturing Technology, 131(12), pp. 6027–6037, 2024.
https://doi.org/10.1007/s00170-024-13305-4

[34] Melinda, M., Aqif, H., Junidar, J., Oktiana, M., Basir, N. B., Afdhal, A., Zainal, Z. "Image Segmentation Performance using Deeplabv3+ with Resnet-50 on Autism Facial Classification", INFOTEL, 16(2), pp. 441–456, 2024.
https://doi.org/10.20895/infotel.v16i2.1144

[35] Sahaya Pushpa Sarmila Star, C., Inbamalar, T. M., Milton, A. "Automatic semantic segmentation of breast cancer in DCE-MRI using DeepLabV3+ with modified ResNet50", Biomedical Signal Processing and Control, 99, 106691, 2025.
https://doi.org/10.1016/j.bspc.2024.106691

[36] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H. "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation", In: 15th European Conference Computer Vision (ECCV 2018), Munich, Germany, 2018, pp. 833–851. ISBN 978-3-030-01233-5
https://doi.org/10.1007/978-3-030-01234-2_49

[37] Roy Choudhury, A., Vanguri, R., Jambawalikar, S. R., Kumar, P. "Segmentation of Brain Tumors Using DeepLabv3+", In: 4th International Workshop, BrainLes 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, 2018, pp. 154–167. ISBN 978-3-030-11726-9
https://doi.org/10.1007/978-3-030-11726-9_14

[38] Eissa, M. M., Napoleon, S. A., Ashour, A. S. "DeepLab V3+ Based Semantic Segmentation of COVID-19 Lesions in Computed Tomography Images", Journal of Engineering Research (ERJ), 6(5), pp. 184–192, 2022.
https://doi.org/10.21608/erjeng.2022.171310.1116

[39] Wada, K. "LabelMe: Build datasets for AI with Private and Flexible annotation", [online] Available at: https://labelme.io [Accessed: 01 June 2024]

[40] Ronneberger, O., Fischer, P., Brox, T. "U-Net: Convolutional Networks for Biomedical Image Segmentation", In: 18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015), Munich, Germany, 2015, pp. 234–241. ISBN 978-3-319-24574-4
https://doi.org/10.1007/978-3-319-24574-4_28

[41] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S. "Feature Pyramid Networks for Object Detection", In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 936–944.
https://doi.org/10.1109/CVPR.2017.106

[42] Berman, J. J. "Chapter 4 - Understanding Your Data", In: Data Simplification: Taming Information with Open Source Tools, Morgan Kaufmann, 2016, pp. 135–187. ISBN 978-0-12-803781-2
https://doi.org/10.1016/B978-0-12-803781-2.00004-7

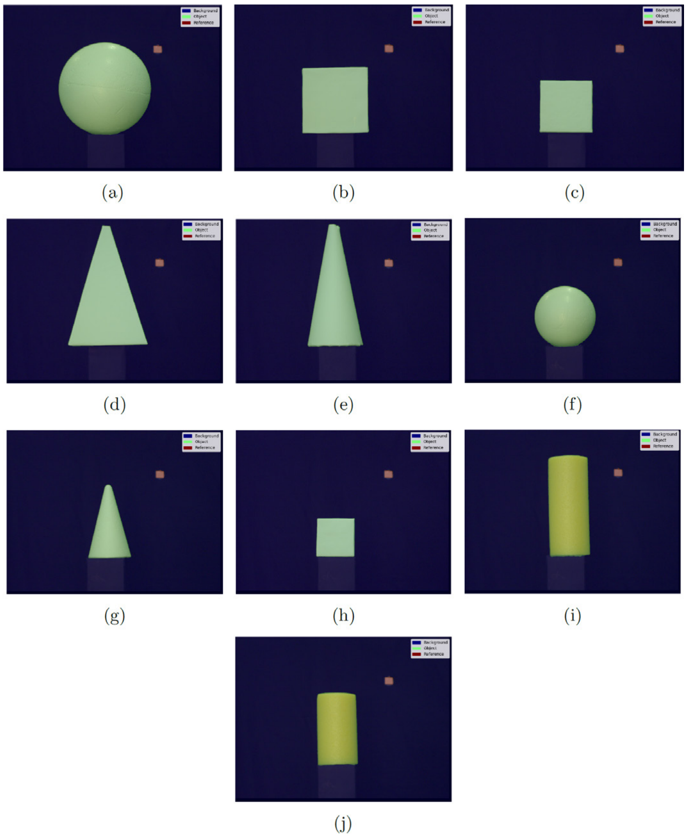
[43] Karunasingha, D. S. K. "Root mean square error or mean absolute error? Use their ratio as well", Information Sciences, 585, pp. 609–629, 2022.
https://doi.org/10.1016/j.ins.2021.11.036

Appendix A

To complement the quantitative results in Section 4.1, Appendix A presents samples of qualitative segmentation results for all tested models. Figs. A1 to A11 display predicted masks overlaid on the original images, covering all 10 test objects.

The models include DeepLabV3+ with ResNet50, U-Net with ResNet34, FPN with ResNet50, and the eight custom lightweight segmentation CNNs (with 2 or 3 convolutional blocks and 8 to 64 filters). Each figure contains one example per object, using a consistent legend for background (Class 0), object (Class 1), and static reference (Class 2).

These visual comparisons reveal that models with pre-trained encoders, especially DeepLabV3+, exhibit more precise boundary delineation and robust identification of static references. In contrast, custom lightweight models tend to underperform, particularly in segmenting Class 2.



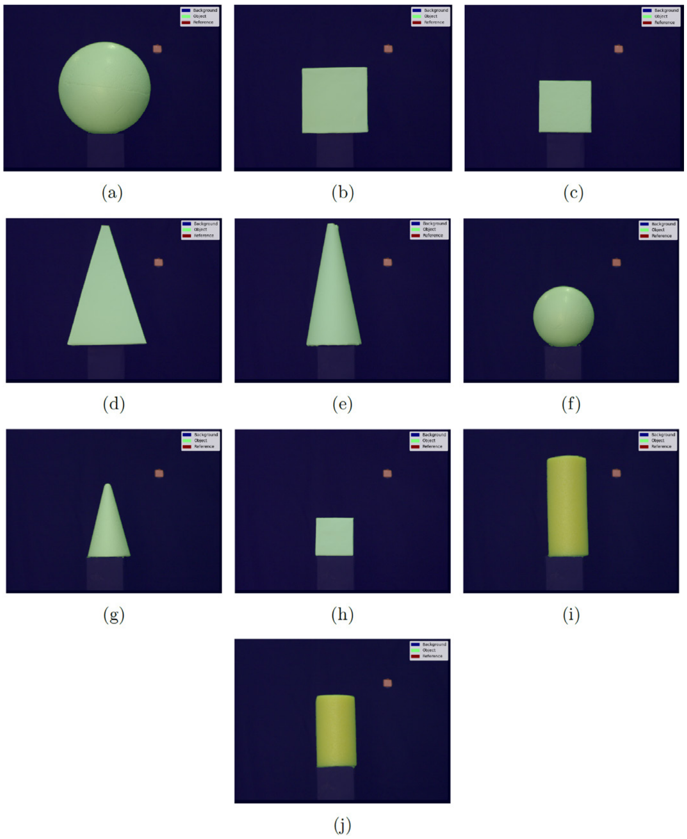
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. A1 DeepLabV3+ with ResNet50 encoder demonstrates highest segmentation quality among all models, with clear separation of classes and minimal false predictions: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



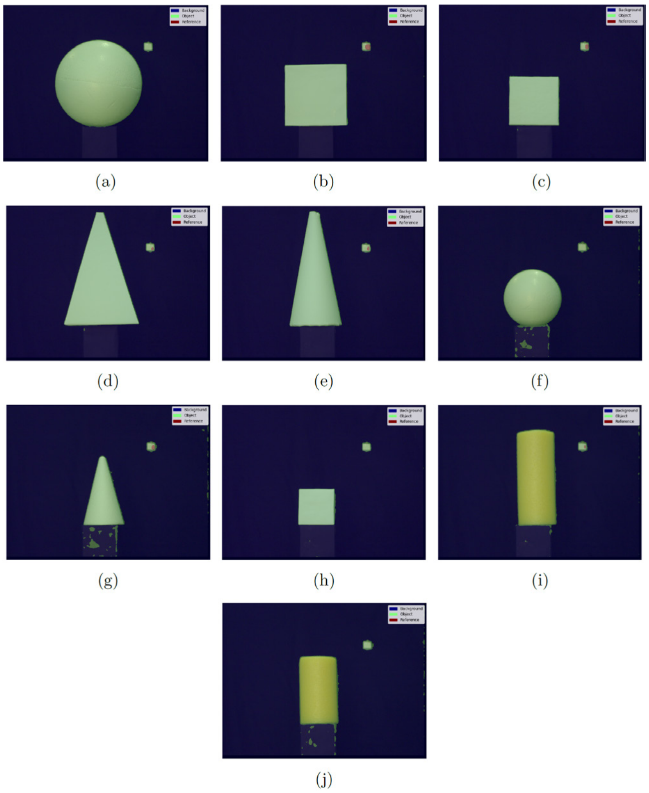
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

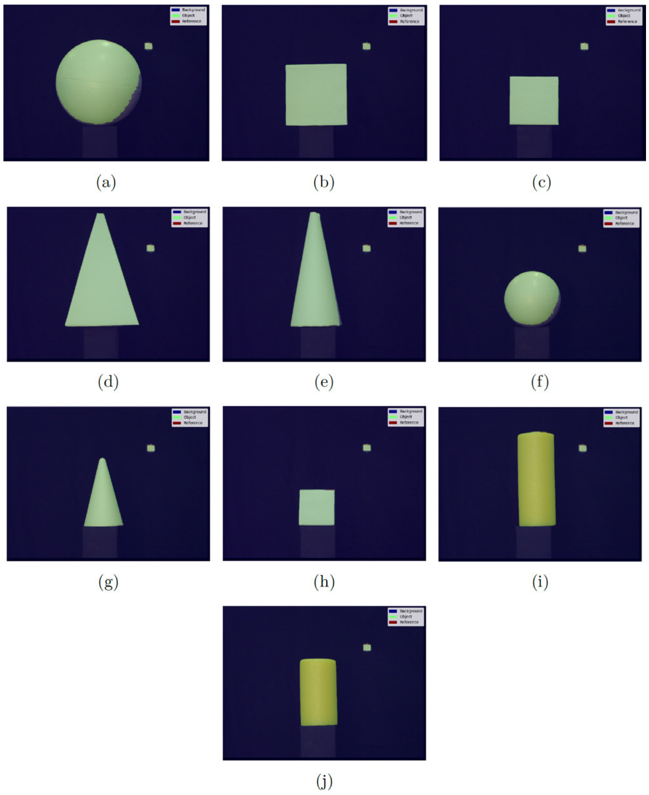
(j)

Fig. A2 Results produced by the pre-trained U-Net model with a ResNet34 encoder: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. A3 While the performance of pre-trained FPN with ResNet50 is better than the Custom CNNs, the model struggles with classification of Class 2 and mislabels background (Class 0) as object (Class 1): (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. A4 Segmentation results for 10 test objects using the Custom CNN with 2 convolutional blocks and 8 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2

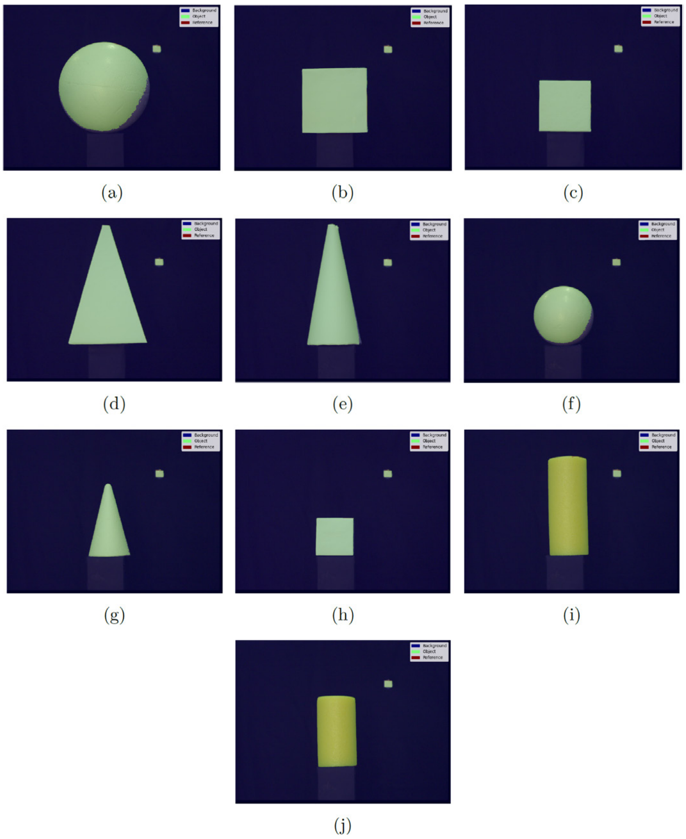


Fig. A5 Results for 10 test objects using the Custom CNN with 2 convolutional blocks and 16 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2

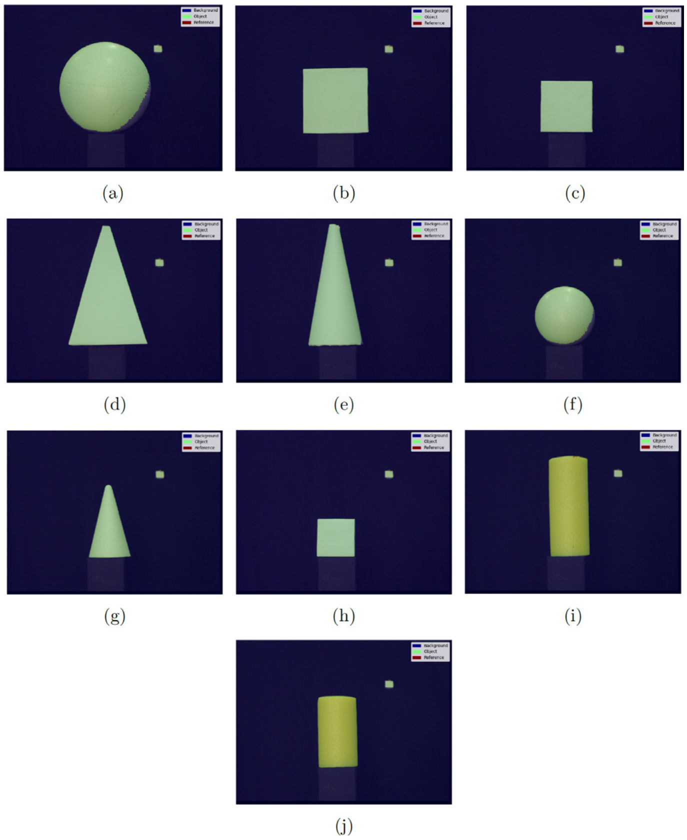
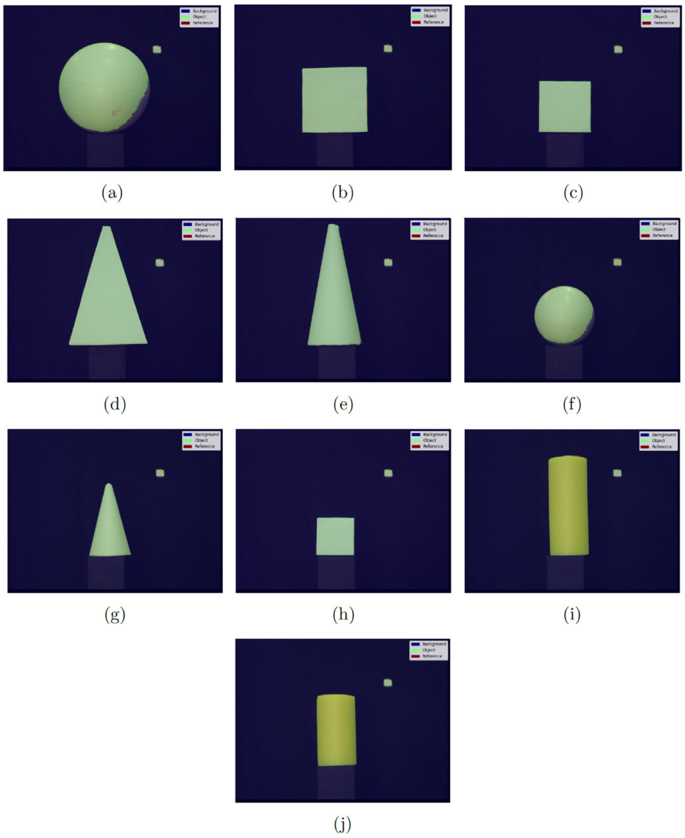


Fig. A6 Results for 10 test objects using the Custom CNN with 2 convolutional blocks and 32 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



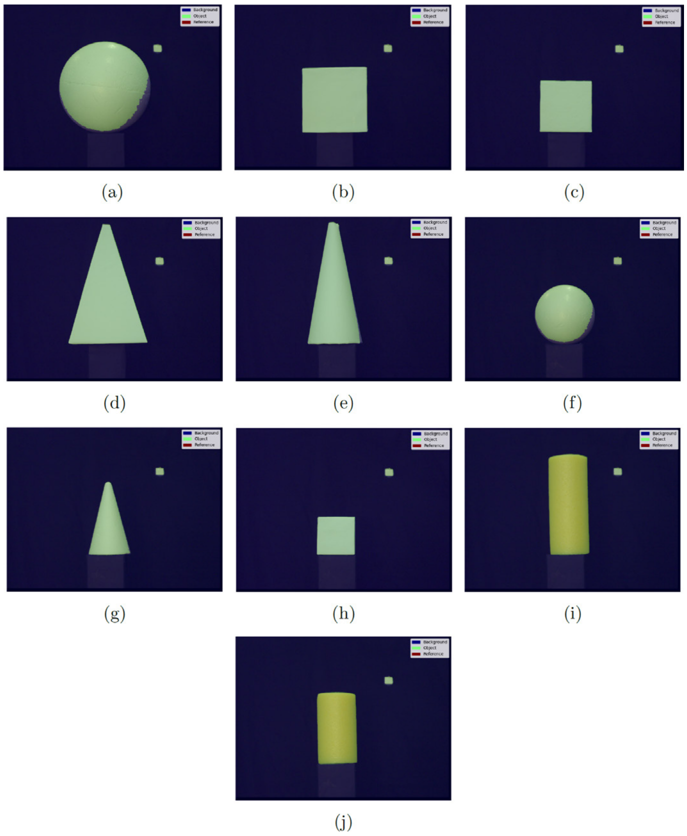
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. A7 Results for 10 test objects using the Custom CNN with 2 convolutional blocks and 64 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



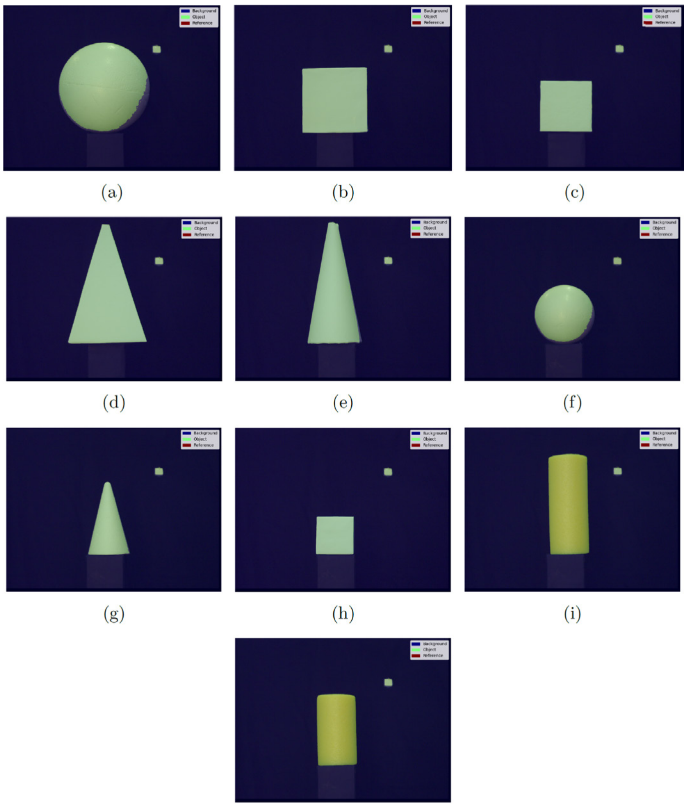
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. A8 Results for 10 test objects using the Custom CNN with 3 convolutional blocks and 8 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



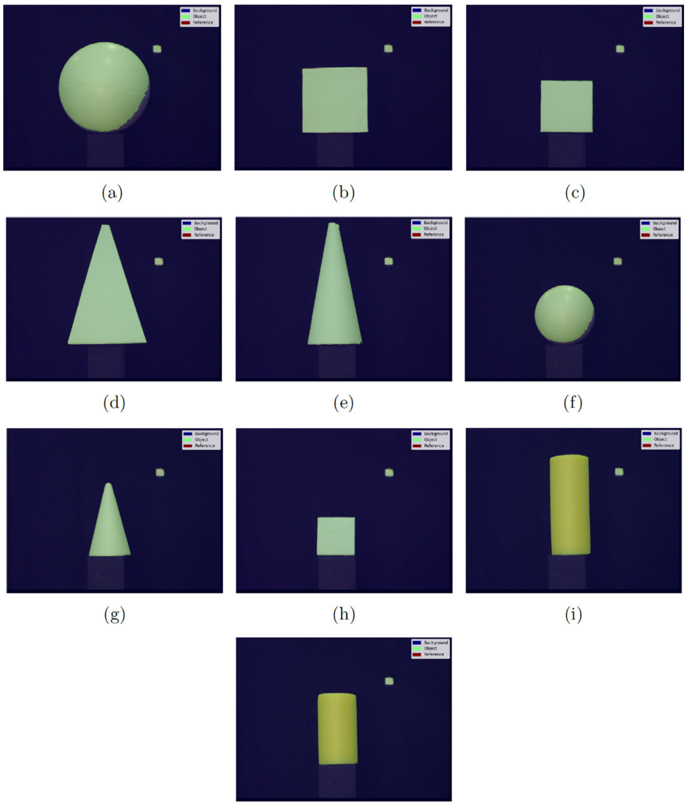
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. A9 Results for 10 test objects using the Custom CNN with 3 convolutional blocks and 16 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



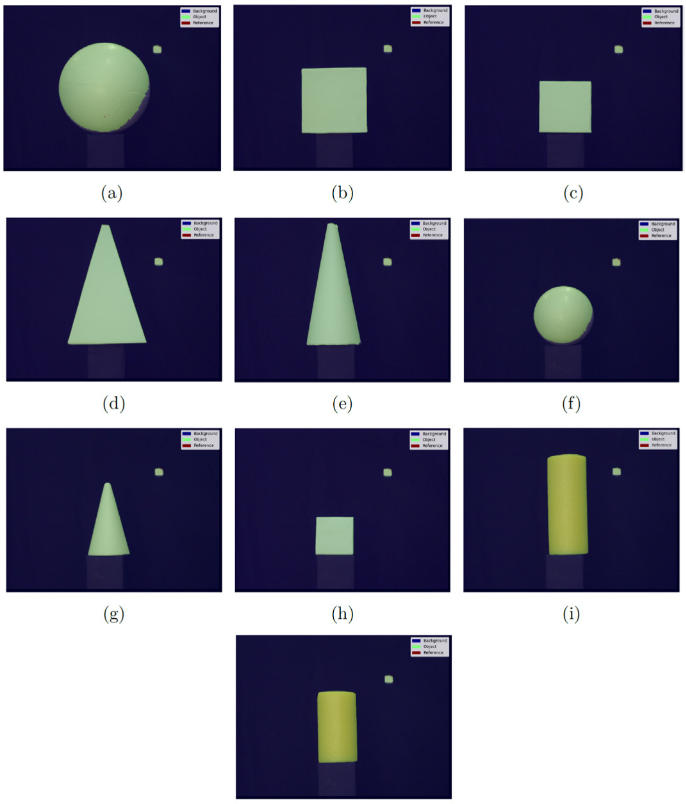
(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. A10 Results for 10 test objects using the Custom CNN with 3 convolutional blocks and 32 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j)

Fig. A11 Results for 10 test objects using the Custom CNN with 3 convolutional blocks and 64 filters: (a) sphere 1; (b) cube 1; (c) cube 2; (d) pyramid; (e) cone 1; (f) sphere 2; (g) cone 2; (h) cube 3; (i) cylinder 1; (j) cylinder 2

Appendix B

Appendix B presents additional visual analyses to complement the results in Section 4.3, evaluating the system's measurement accuracy in the global coordinate frame. The plots compare predicted coordinates of the object and static reference against ground-truth values, along with their corresponding absolute errors.

Figs. B1 and B2 show the predicted and ground-truth x and y-coordinates of the static reference center. Although slight deviations are observed in the x-coordinate predictions, attributable to the reference's relatively small segmented area and slight blurring due to camera focus on the object, the predictions closely track the ground truth. This is corroborated by the absolute error plots (Figs. B3 and B4), which show errors consistently below 0.6 mm.

Figs. B5 and B6 illustrate the corresponding comparisons for the object center, with predicted coordinates closely matching ground-truth values across all frames. Absolute errors (Figs. B7 and B8) remain under 1.0 mm throughout.

These visual results reinforce the system's precision and robustness in both local and global frames, supporting the reliability indicated by the quantitative evaluation metrics.

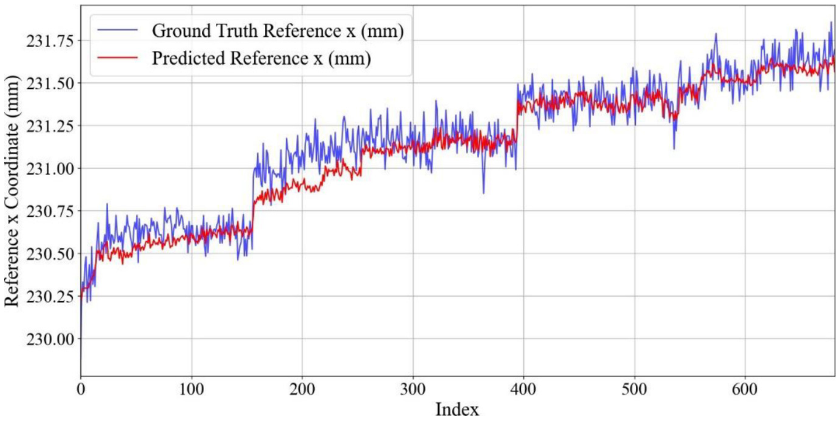


Fig. B1 Comparison of the x-coordinate of the center of the static reference in the global coordinate system of the original images

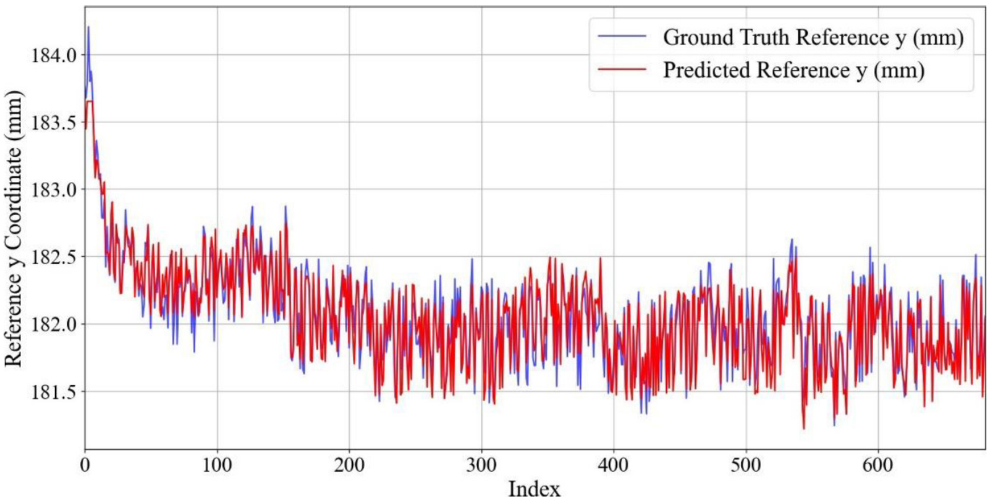


Fig. B2 Comparison of the y-coordinate of the center of the static reference in the global coordinate system of the original images

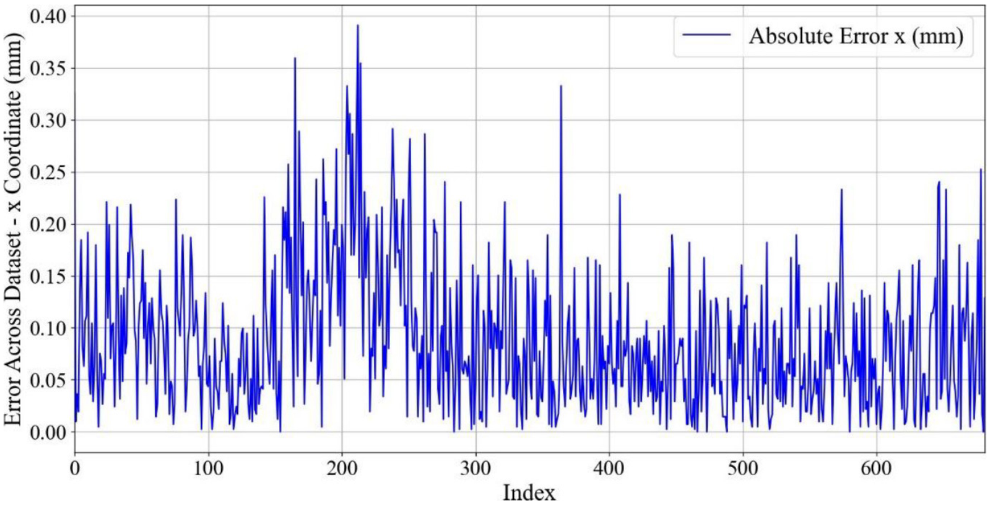


Fig. B3 Absolute error of the x-coordinate of the static reference in the global coordinate system

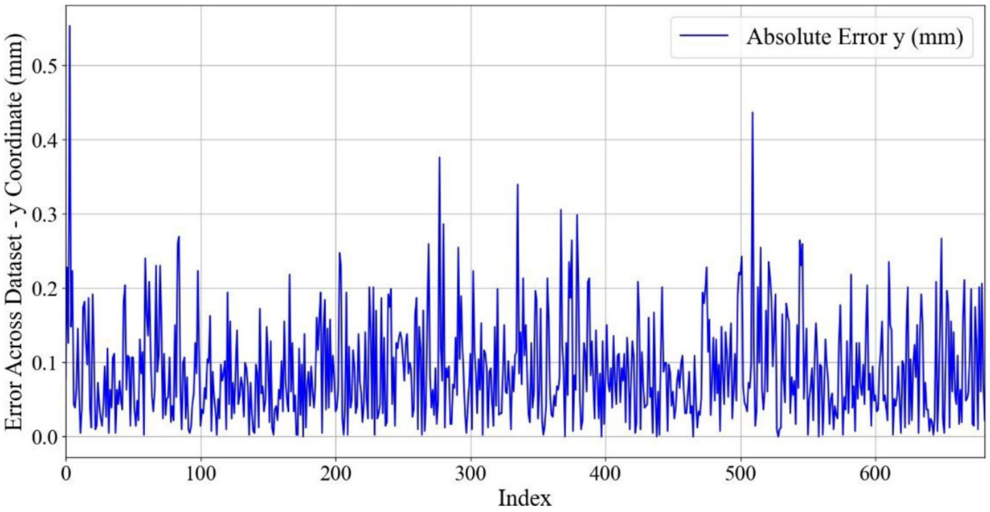


Fig. B4 Absolute error of the y-coordinate of the static reference in the global coordinate system

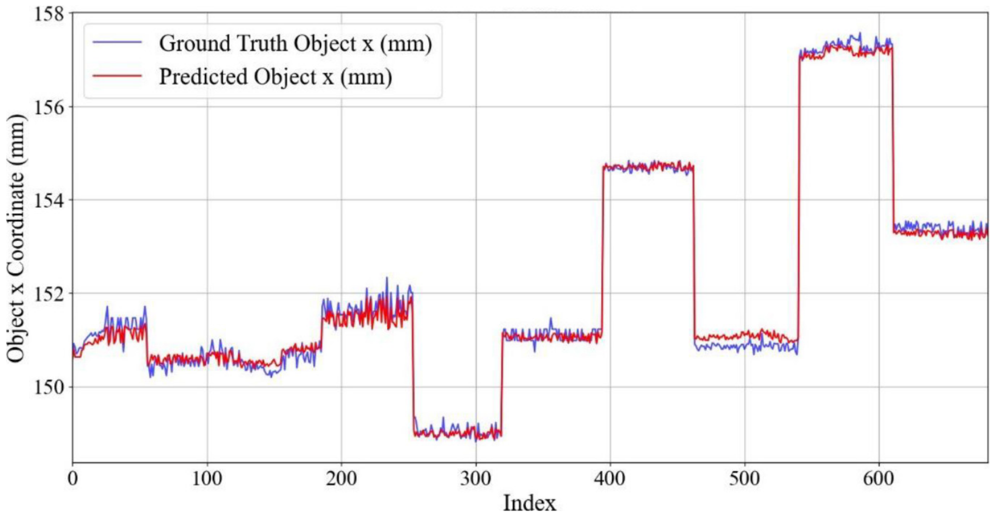


Fig. B5 Comparison of the x-coordinate of the center of the object in the global coordinate system of the original image

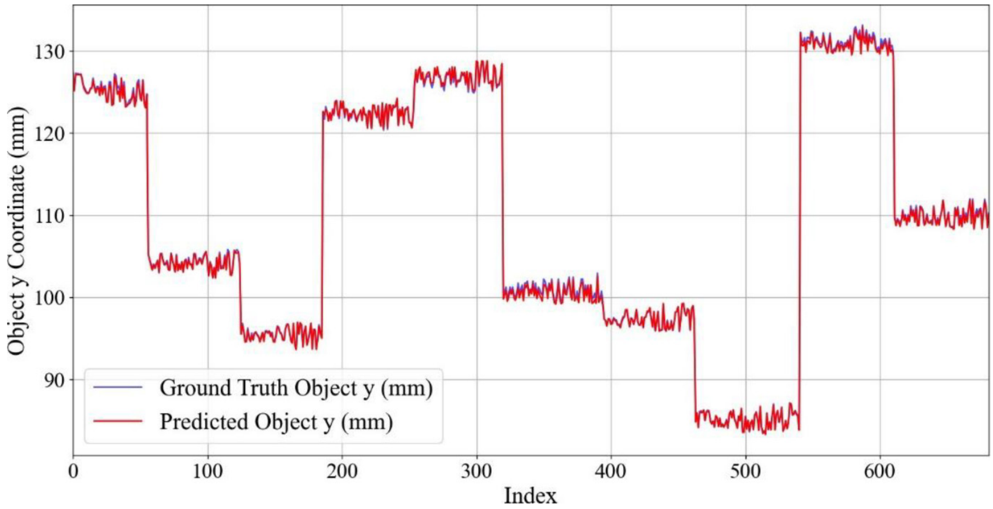


Fig. B6 Comparison of the y-coordinate of the center of the object in the global coordinate system of the original image

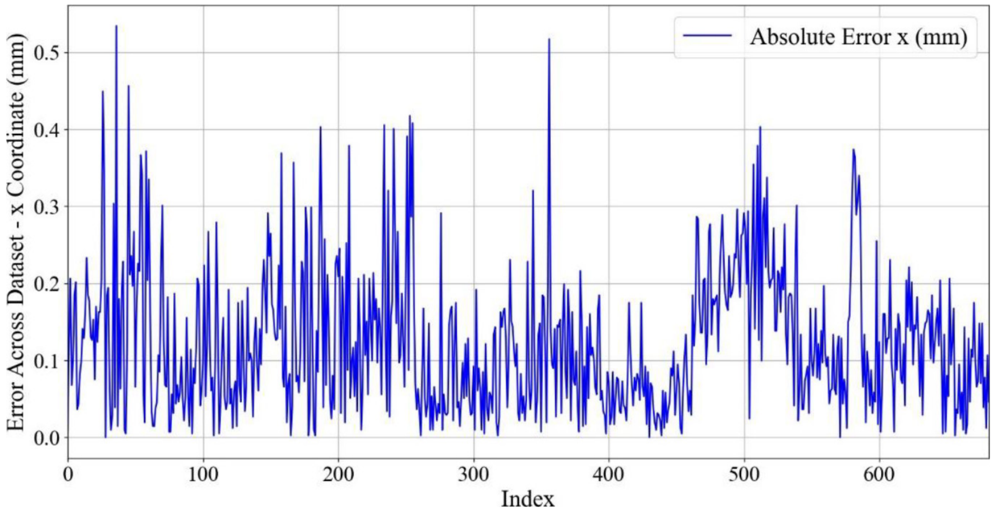


Fig. B7 Absolute error of the x-coordinate of the object in the global coordinate system

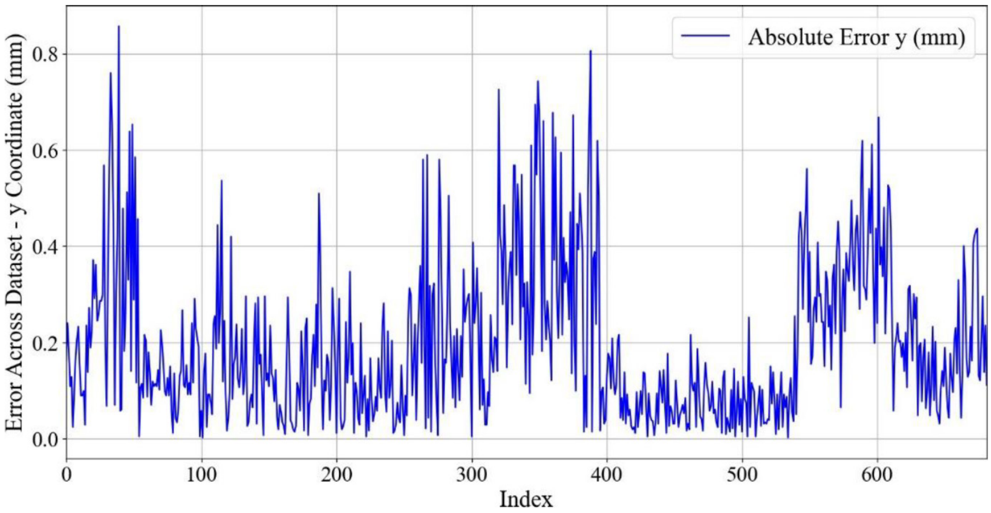


Fig. B8 Absolute error of the y-coordinate of the object in the global coordinate system