Enhancing Water Meter Digit Recognition Using YOLO-OCR

Fadhila Alya Syahfahlevi
Informatics Engineering Study Program, Telkom University, Purwokerto, Indonesia
fadhilaalya@student.telkomuniversity.ac.id

Sudianto Sudianto
Informatics Engineering Study Program, Telkom University, Purwokerto, Indonesia
sudianto@telkomuniversity.ac.id (corresponding author)

Aminatus Sa'adah
Informatics Engineering Study Program, Telkom University, Purwokerto, Indonesia
aminatuss@telkomuniversity.ac.id

Received: 1 January 2026 | Revised: 22 February 2026, 7 March 2026, and 11 March 2026 | Accepted: 15 March 2026

 DOI: https://doi.org/10.48084/etasr.17291

ABSTRACT

This study presents a deep learning-based framework for automatic water meter digit recognition by combining You Only Look Once version 8 (YOLOv8) for digit region detection and PP-OCRv4 for Optical Character Recognition (OCR). The model was developed using a dataset of 3,250 annotated water meter images collected under various lighting and visibility conditions, including ideal and non-ideal scenarios. The detection module was fine-tuned using different optimizers, with Adam achieving the highest detection accuracy (F1-score, precision, and recall all at 100% and a Mean Average Precision at Intersection over Union (IoU) thresholds ranging from 0.50 to 0.90 (mAP50–90) of 0.95831). The recognition model was trained on both the full dataset and a subset of 1,000 non-ideal images. Interestingly, the best performance was observed in the model trained solely on non-ideal data using AdamW, achieving a recognition accuracy of 90.90% and a Character Error Rate (CER) of 0.0151 in end-to-end testing across 55 benchmark images. The system demonstrated near real-time inference, averaging 0.5 s per image. These results validate the robustness and practical applicability of the proposed YOLO-OCR pipeline and highlight the advantage of training on visually challenging data to improve robustness in real-world deployment conditions.

Keywords-YOLOv8; PP-OCRv4; water meter; detection; recognition

I. INTRODUCTION

Water is a vital natural resource that must be managed efficiently to ensure sustainability. The implementation of smart technologies in water management correlates with Sustainable Development Goal (SDG) 9, which emphasizes innovation, infrastructure development, and industrial advancement [1]. Among these technologies, the Smart Water Meter enables real-time water consumption monitoring through integrated communication and data analytics, improving efficiency, reducing wastage, and increasing user awareness. Research on Smart Water Meter systems has grown rapidly since 2000, driven by advances in engineering and data science [2].

Despite this progress, many water utilities worldwide—particularly in developing and transitional regions—still rely on manual meter reading practices. These conventional approaches are time-consuming, labor-intensive, prone to human error, and lack scalability, ultimately limiting operational efficiency and data reliability. Such constraints highlight a global need for automated, accurate, and cost-effective meter-reading solutions that can be deployed in real operational environments without extensive infrastructure upgrades. Computer vision–based approaches offer a promising alternative, enabling non-intrusive digit recognition from meter images using object detection and text recognition techniques.

Recent advances in deep learning have demonstrated strong potential for automated utility-meter reading. Object detection models such as You Only Look Once (YOLO) are capable of detecting meter regions and digit areas in a single inference pass with high accuracy and low latency [3]. The effectiveness of YOLO-based detectors for real-time object localization in practical environments has also been extensively validated, where optimized YOLO architectures were shown to significantly improve detection accuracy while maintaining computational efficiency for real-world applications [4, 5],

whereas Optical Character Recognition (OCR) techniques convert visual digit information into machine-readable text [6]. Prior studies have investigated using deep learning for object detection along with digit recognition to automate utility-meter reading. A low-cost smart-metering system integrating YOLOv8 as meter output detection and PaddleOCR as digit recognition achieved 96.92% recognition accuracy and 97.8% end-to-end performance on 8,044 images of water, gas, and electricity meters [7]. The current research adopts a similar concept but focuses specifically on water meters and utilizes the improved PP-OCRv4 for enhanced robustness. Another study implemented a two-stage water-meter reading method using YOLOv8 for digit-region detection and OCR architectures such as PP-OCRv3, TrOCR, and Robust Scanner on 3,672 water meter images, where PP-OCRv3 achieved approximately 93% accuracy under low-light conditions [8].

Further validation of deep learning–based object detection combined with OCR has been reported across various metering and digit-recognition domains. Recent studies employing YOLO-based architectures integrated with OCR demonstrate high robustness in digit localization and recognition tasks. For instance, an end-to-end YOLO-OCR framework achieved detection accuracy exceeding 99% and character recognition accuracy above 98% in real-world numeric recognition scenarios, confirming the effectiveness of unified detection–recognition pipelines for structured digit sequences [9]. Although these studies were conducted in domains such as automotive or industrial monitoring, the underlying visual characteristics—namely fixed-layout numeric displays and sequential digits—closely resemble those found in water-meter imagery, indicating strong cross-domain transferability. Similarly, automated meter-reading systems utilizing earlier YOLO variants (e.g., YOLOv3) have reported recognition accuracies approaching 98% on large-scale datasets of energy-meter images [10]. However, such approaches generally rely on separated or simplified OCR stages, which limits their capability to accurately extract sequential digits under challenging conditions such as reflections, occlusions, or non-uniform illumination—conditions that are common in water-meter environments.

Building on this work, the current study aims to improve water meter digit recognition with an optimized YOLO-OCR framework designed for practical use. By using YOLOv8 for accurate digit-region detection and the enhanced PP-OCRv4 for strong text recognition, this research seeks to solve the global problem of automating water meter readings in real-world settings, helping achieve more efficient, scalable, and sustainable water management systems.

Unlike many previous water meter reading systems that primarily focus on improving model architectures, this study investigates the effect of dataset conditioning on the robustness of an integrated detection–recognition pipeline. Specifically, we analyze how training the OCR model on visually challenging (non-ideal) samples influences end-to-end recognition performance. This experimental perspective provides insight into the role of dataset composition in improving robustness under real-world acquisition conditions, where reflections, occlusions, and digit transitions frequently occur.

II. METHODOLOGY

This research consists of two main stages, namely developing a water meter digit detection model using YOLOv8 and developing a digit extraction model using PP-OCRv4. Figure 1 illustrates both development processes, including data acquisition, preprocessing, annotation, data splitting, building a model, fine-tuning, and evaluation.

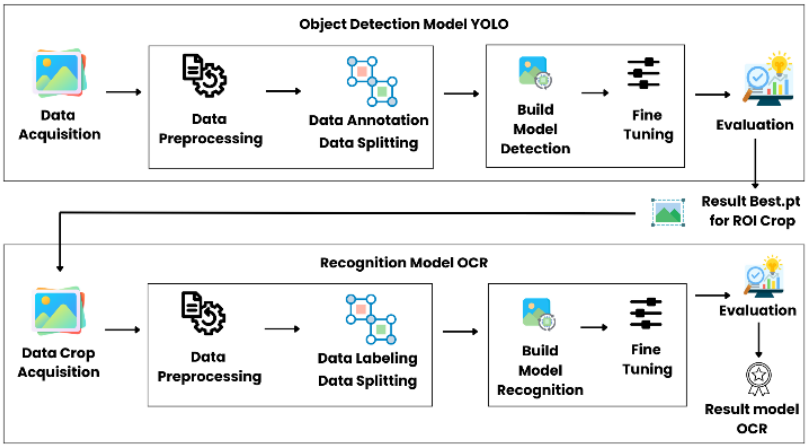


Fig. 1. Workflow of the proposed system.

The YOLOv8 model represents a significant advancement over its predecessors by employing an anchor-free methodology, which enhances network efficiency by removing the requirement for preset anchor boxes [11, 12]. This modification facilitates direct predictions of object centers and bounding box dimensions, thereby streamlining the detection process and accelerating inference speed [11]. The three primary parts of the architecture are the head, neck, and backbone, as shown in Figure 2 [11]. The backbone extracts visual characteristics from initial images by convolutional layers and C2F modules, which enhance feature representation across various scales. The neck combines multiscale features

using a Feature Pyramid Network (FPN) and Path Aggregation Network (PANet), enhancing the model capacity to identify objects of various sizes. This feature integration is visually represented by structured flow in the diagram. The head is composed of numerous detection layers that function at varying feature resolutions and employ a decoupled head strategy for classification, objectness scoring, and bounding box regression, resulting in heightened accuracy with minimal computational overhead.

PP-OCRv4 is one of the high-performance OCR frameworks developed by Baidu as part of the PaddleOCR ecosystem [13]. It is intended to handle multi-language text recognition with increased speed and accuracy. The architecture is comprised of a preprocessing stage, a backbone, a neck, and a head, as shown in Figure 3. While the neck includes a Simple Visual Transformer (SVTR) block to capture long-range relationships and improve spatial representation, the backbone employs PP-LCNetV3 to extract effective visual features. Flexible decoding is made possible by the head's combination of a Neural Machine Translation Recognizer (NRTR) head for direct sequence prediction and a Connectionist Temporal Classification (CTC) head for logit prediction. CTC decoding is used in a postprocessing phase to provide final digit outputs. To increase generalization under various visual situations, methods like TextConAug augmentation and Guided CTC training are used.

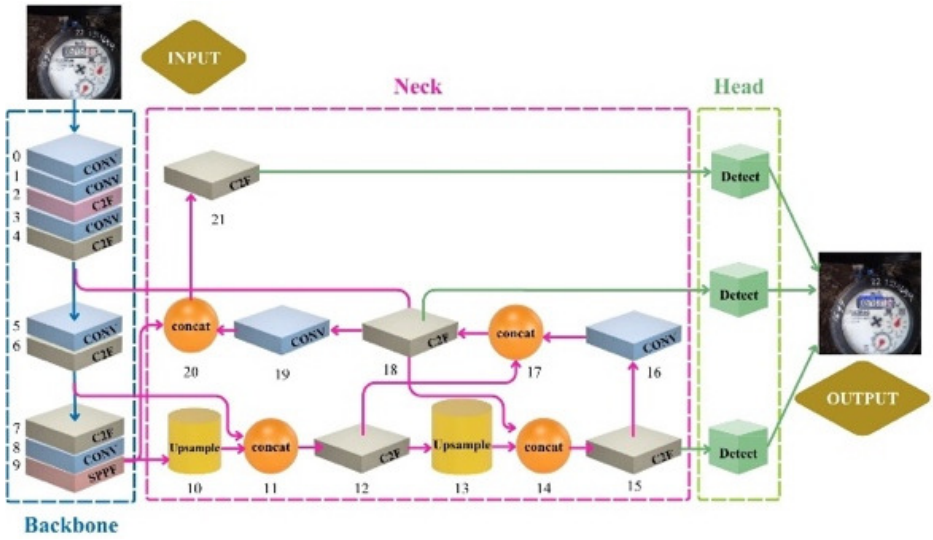


Fig. 2. YOLOv8 object detection architecture.

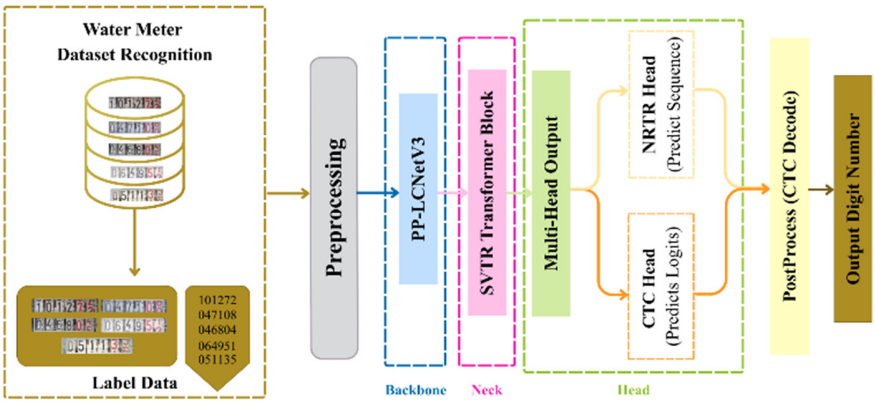


Fig. 3. PP-OCRv4 recognition architecture.

A. Data Acquisition

This study utilizes secondary data from the Tirtawijaya Cilacap Regional Water Company, consisting of 3,250 monthly water meter images (.jpg) captured under varied lighting conditions and representing multiple 6-digit meter types. Red digits indicate liters and black digits indicate cubic meters (Figure 4). Images are categorized as ideal when all six digit windows are fully visible and each numeral appears completely formed without overlap or truncation. Conversely, images are classified as non-ideal when at least one digit is partially visible, exhibits transitional overlap between consecutive numbers (e.g., 1–2, …, 9–0), or is affected by occlusion, glare, or perspective distortion, as shown in Figure 5. Based on this operational definition, 2,250 images are labeled as ideal and 1,000 as non-ideal. An additional 55 images were included for end-to-end evaluation.

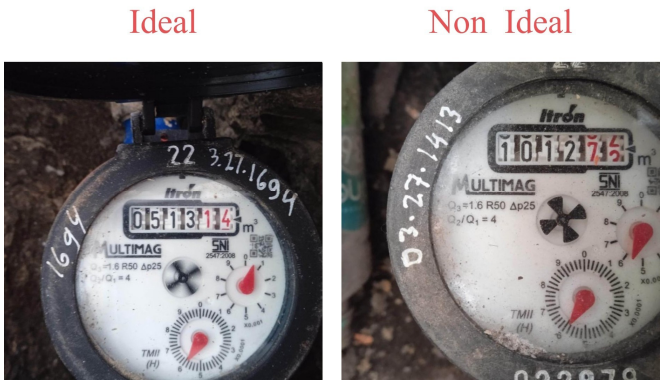


Fig. 4. Examples of ideal and non-ideal water meter conditions.

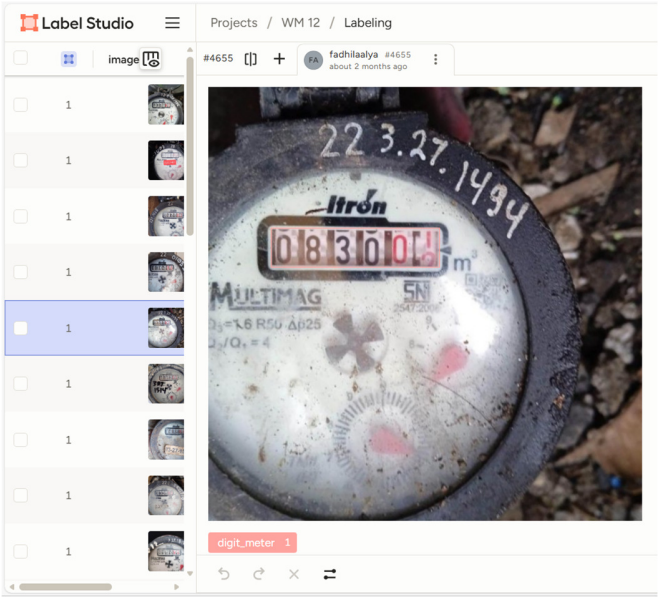


Fig. 5. Water meter dataset annotation using object detection with a bounding box in Label Studio.

B. Data Preprocessing

The detection dataset, which included 3,250 images, was separated into training and validation subsets using an 80:20 ratio. Before training, all photos were scaled to 640 × 640 pixels and visually improved with the Contrast Limited Adaptive Histogram Equalization (CLAHE) approach. As shown in Figure 5, bounding box annotations were created using the Label Studio platform with a single annotation class (digit_meter) and the annotations were produced in .txt format. The recognition dataset was also created using the same 3,250 images, with two subgroups dependent on image quality: 2,250 images in ideal conditions and 1,000 images in non-ideal conditions, such as partial digit visibility. To evaluate model robustness under different data conditions, two training scenarios were designed, as shown in Figure 6:

- Scenario A: The full dataset of 3,250 images (ideal + non-ideal).
- Scenario B: Only the 1,000 non-ideal images.

Keterangan	Dataset Non Ideal	Dataset Ideal + Non Ideal
Train	800 data	2600 data
Validation	200 data	650 data
Total	1000 data	3250 data

Fig. 6. Dataset splitting into training and validation.

Each scenario was split into 80% training and 20% validation. In order to match the PP-OCRv4 architecture's input requirements, recognition samples were scaled to 48 × 320 pixels prior to model training. Additionally, to evaluate their impact on performance, each scenario was trained under three fine-tuned sub-scenarios with different hyperparameter selections.

C. YOLO-OCR Model Building and Fine-Tuning

The detection dataset was used to train the YOLOv8m model with several hyperparameter configurations to produce the best results for the water meter number area detection model. In addition, PP-OCRv4 was also trained and fine-tuned to produce the best model for character recognition. Evaluation was carried out using a validation dataset to ensure model performance, as shown in Tables I and II.

TABLE I. DETECTION HYPERPARAMETER CONFIGURATION

Parameter	Scenario 1	Scenario 2	Scenario 3
Epoch	150	150	150
Batch size	16	16	16
Learning rate	0.001	0.001	0.001
Optimizer	Adam	AdamW	SGD

SGD = Stochastic Gradient Descent.

TABLE II. RECOGNITION HYPERPARAMETER CONFIGURATION

Parameter	Scenario 1	Scenario 2	Scenario 3
Epoch	150	150	150
Batch size	64	64	64
Learning rate	0.001	0.001	0.001
Optimizer	Adam	AdamW	Momentum
Workers	4	4	4

The hyperparameters were determined through controlled preliminary experiments analyzing convergence behavior under different configurations. A learning rate of 0.001 was selected as it ensured stable convergence across optimizers, whereas lower values slowed training and higher values caused validation instability. Training was conducted for 150 epochs because validation loss and mean Average Precision (mAP) plateaued between epochs 120–140, indicating convergence without additional performance gain.

Batch sizes were defined according to task-specific computational demands and GPU memory constraints: 16 for detection due to the higher memory footprint of 640 × 640 inputs and model complexity, and 64 for recognition since cropped digit regions required fewer computational resources. While multiple optimization scenarios were evaluated, exhaustive grid-search tuning was not performed, as the study primarily aimed to assess robustness under varying dataset conditions rather than to maximize architectural performance.

In the system workflow, the meter image was first taken as input and then went through the detection preprocessing stage. If the confidence level of the meter screen detection was below 0.8, the image was sent back to the beginning to be processed again. If the confidence level was above 0.8, the detected area was cut out and sent to the OCR preprocessing stage. The final result was considered valid only if the character recognition process gave a confidence level higher than 0.5. If not, the image was sent back into the detection stage until the model achieved the required level of confidence. Figure 7 displays the system's block diagram.

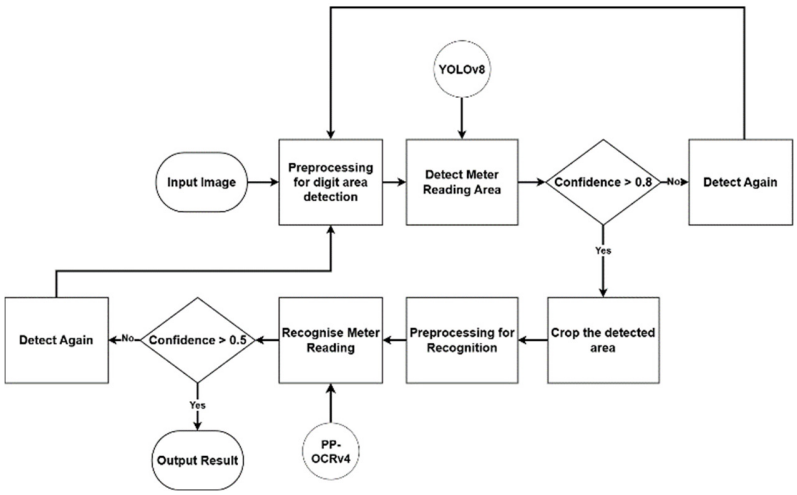


Fig. 7. Block diagram of the proposed framework.

D. Model Evaluation

The system overall performance was assessed using a set of task-specific evaluation indicators. Precision, recall, mAP at an Intersection over Union (IoU) threshold of 0.50 (mAP50), mAP at IoU thresholds ranging from 0.50 to 0.90 (mAP50–90), and the F1-score were used to evaluate the detection component's effectiveness. The Character Error Rate (CER), Recognition Accuracy (RA), and Character Accuracy (CA) were used to assess the OCR and recognition processes. CER is better suited for concise, single-term outputs similar to meter readings than Word Error Rate (WER), which is designed for longer phrases. Metrics such as precision, recall, and F1-score are unsuitable here because recognition outputs may be incomplete or contain extra characters [14, 15].

Precision (P) measures the proportion of correctly detected meter digit regions among all detected digit regions. Precision is defined as the ratio of true positives (TP) to the sum of true positives (TP) and false positives (FP), as shown in (1):

$$P = \frac{TP}{TP+FP} \quad (1)$$

Recall (R) measures the proportion of correctly detected meter digit regions relative to all ground-truth digit regions. It is calculated as the ratio of true positives (TP) to the sum of true positives (TP) and false negatives (FN), as defined in (2):

$$R = \frac{TP}{TP+FN} \quad (2)$$

The F1-score (F1) is used to balance precision and recall and is computed as the harmonic mean of both metrics, as given in (3):

$$F1 = 2 * \frac{P*R}{P+R} \quad (3)$$

For detection evaluation, mAP is employed. The average precision (AvgP) is first computed for each class based on the ranked list of detections at a given IoU threshold. The final mAP at an IoU threshold of 0.50, denoted as mAP50, is obtained by averaging the average precision values over all $|X|$ classes, as defined in (4):

$$mAP50 = \frac{1}{|X|}\sum_{i=1}^{|X|} AvgP(Z_i) \quad (4)$$

In (4), $|X|$ denotes the total number of digit classes, and $AvgP(Z_i)$ represents the average precision of class Z_i.

Similarly, taking into consideration multiple IoU thresholds between 0.5 and 0.9, mAP50–90 is calculated as the mean of the average precision scores over the assessment queries q_i, as shown in (5):

$$mAP50-90 = \frac{1}{|Q|}\sum_{i=1}^{|Q|} AvgP(q_i) \quad (5)$$

CER calculates the character-level discrepancy between the ground-truth text and the OCR output, as defined in (6):

$$CER = \frac{S+D+I}{C} * 100 \quad (6)$$

where S, D, I, and C denote substitutions, deletions, insertions, and the total number of reference characters, respectively.

The percentage of meter readings that are entirely accurate is indicated by recognition accuracy, as defined in (7):

$$RA = \frac{N}{T} * 100 \quad (7)$$

where T denotes the total number of readings and N denotes the number of correctly identified readings.

The percentage of successfully identified characters is represented by character accuracy, as shown in (8)

$$CA = \frac{CC}{TC} * 100 \quad (8)$$

where CC denotes correct characters and TC denotes total characters.

III. RESULTS AND ANALYSIS

A. Fine-Tuning Results

Three distinct optimizers were used to refine the object detection model, YOLOv8: Adam, AdamW, and standard Stochastic Gradient Descent (SGD). Each training configuration employed a fixed learning rate of 0.001, a batch size of 16, and a total of 150 epochs. The dataset consisted of 3,250 annotated water meter images, divided into 80% training and 20% validation sets. Based on the evaluation metrics presented in Table III, the Adam optimizer yielded the most optimal detection performance, attaining perfect scores for F1-score, precision, and recall, alongside the highest mAP50–90 value of 0.95831, as shown in Figure 8. While the models trained with AdamW and SGD demonstrated slightly lower classification losses, their corresponding mAP50–90 scores were recorded at 0.95806 and 0.95777, indicating a marginal decline in detection precision. The results confirm that Adam provides the most effective balance between optimization stability and detection accuracy for the water meter localization task using YOLOv8. Therefore, Adam demonstrated the optimal trade-off between loss reduction and detection precision in the YOLOv8 water meter localization task.

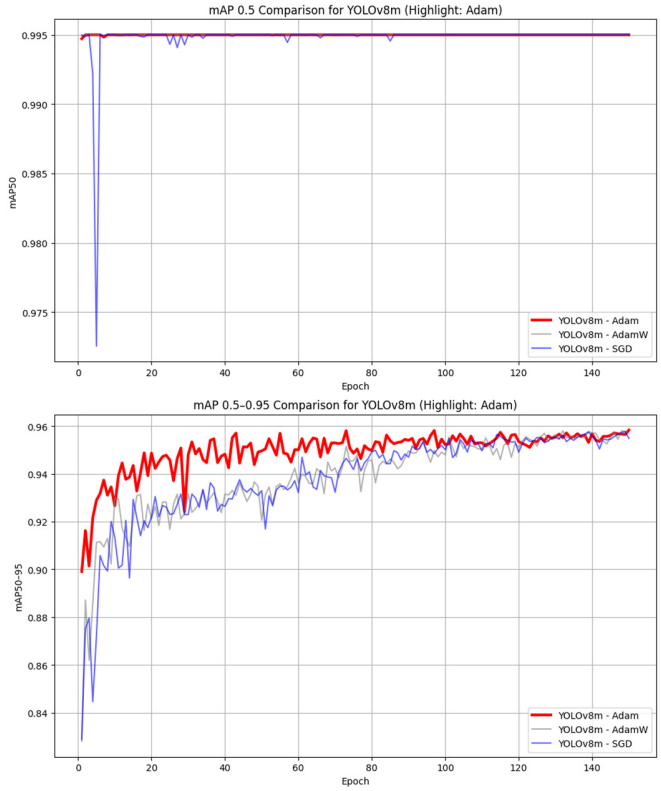


Fig. 8. Fine-tuning results of the detection model for the three scenarios.

The detection model achieved nearly perfect precision and recall, which is partly influenced by the single-class detection setup where YOLOv8 is trained to detect only the water meter digit display region, resulting in a relatively less complex detection task.

TABLE III. DETECTION PERFORMANCE RESULTS FOR FINE-TUNING SCENARIOS

Scenario	F1-score (%)	Precision (%)	Recall (%)	mAP50	mAP50–90
1	100	100	100	0.995	0.95831
2	99.994	99.989	100	0.995	0.95806
3	99.995	99.992	100	0.995	0.95777

The recognition model, PP-OCRv4, was fine-tuned under the same number of epochs and learning rate, using a batch size of 64 and three optimization scenarios: Adam, AdamW, and Momentum-SGD. Two training scenarios were considered, comprising a subset of 1,000 non-ideal images and the complete set of 3,250 images. Table IV displays the best performance on the 1,000-image subset that was achieved using the AdamW optimizer, with a training accuracy of 96.49%, a normalized edit distance of 0.9909, and an inference speed of 382.09 frames per second (FPS). The full dataset configuration, as detailed in Table V, also indicated AdamW as the top-performing optimizer, achieving a training accuracy of 95.53% with a normalized edit distance of 0.9906 and the highest inference throughput of 940.21 FPS. Despite the Momentum-SGD optimizer reaching faster convergence and yielding competitive edit distances, its training accuracy remained below that of AdamW. The results across both dataset scales consistently demonstrate the capability of AdamW to enhance both recognition precision and computational efficiency in the PP-OCRv4 model.

TABLE IV. RECOGNITION PERFORMANCE RESULTS FOR FINE-TUNING SCENARIOS (1,000 IMAGES)

Scenario	Training accuracy (%)	Normalized edit distance	FPS
1	95.99	0.9872	333.05
2	96.49	0.9909	382.09
3	95.99	0.9894	347.99

TABLE V. RECOGNITION PERFORMANCE RESULTS FOR FINE-TUNING SCENARIOS (3,250 IMAGES)

Scenario	Training accuracy (%)	Normalized edit distance	FPS
1	95.23	0.9911	611.67
2	95.53	0.9906	940.21
3	92.92	0.9863	648.38

During training, the recognition model showed stable convergence, with loss values stabilizing and recognition accuracy approaching 1.0 in later epochs. The best validation performance was achieved at epoch 56 with an accuracy of 0.9649, indicating effective learning without significant overfitting.

To validate these findings under operational conditions, the recognition models trained with AdamW were further evaluated using 55 end-to-end benchmark images. The

performance results summarized in Table VI indicate that the model trained on the 1,000 non-ideal images produced a recognition accuracy of 90.90%, a character accuracy of 98.48%, and a CER of 0.0151. The model trained on the full dataset produced lower recognition accuracy at 81.81% and a higher CER of 0.0303. A detailed breakdown of recognition performance under ideal and non-ideal conditions is presented in Table VII. These results highlight that exposure to challenging image conditions during training improves generalization capabilities, allowing the model to better accommodate variations commonly found in real-world scenarios. The AdamW optimized model trained on non-ideal data therefore demonstrates higher reliability for deployment in practical water meter reading applications.

TABLE VI. RECOGNITION PERFORMANCE ON THE BENCHMARK DATASET

Training dataset	Recognition accuracy (%)	CA (%)	CER	Training accuracy (%)
1,000	90.90	98.48	0.0151	96.49
3,250	81.81	96.96	0.0303	95.53

TABLE VII. RECOGNITION ACCURACY UNDER IDEAL AND NON-IDEAL CONDITIONS

Training dataset	Ideal images (40)	Non-ideal images (15)	Overall accuracy (55)
1,000	92.50% (37/40)	86.67% (13/15)	90.90% (50/55)
3,250	82.50% (33/40)	80.00% (12/15)	81.81% (45/55)

To evaluate the statistical reliability of the results obtained from 55 test images, a 95% confidence interval was calculated using the binomial proportion method. The model trained with 1,000 non-ideal images achieved a recognition accuracy of 90.90% with a 95% confidence interval of [83.3%, 98.5%], whereas the model trained with 3,250 images achieved an accuracy of 81.82% with a confidence interval of [71.5%, 92.1%]. These results suggest that training with visually challenging samples improves model generalization, as the model trained on non-ideal images demonstrates stronger robustness in end-to-end recognition. Furthermore, unlike previous water meter reading systems that rely on earlier OCR frameworks such as PaddleOCR or PP-OCRv3, this study integrates PP-OCRv4 within a YOLOv8-based detection pipeline. The improved backbone (SVTR_LCNetV3) and optimized training strategies in PP-OCRv4 enable more effective feature extraction and improved recognition accuracy under real-world conditions.

Exposure to visually degraded data encourages the model to rely on more stable and semantically meaningful features rather than ideal visual patterns. Similar observations have been reported in studies on natural distribution shifts and domain generalization, where models trained on diverse or imperfect data distributions exhibit reduced performance degradation during deployment [16, 17]. Furthermore, increased visual variability has been shown to improve robustness and generalization by mitigating overfitting to ideal training conditions, as discussed in robustness studies on common visual corruptions [18].

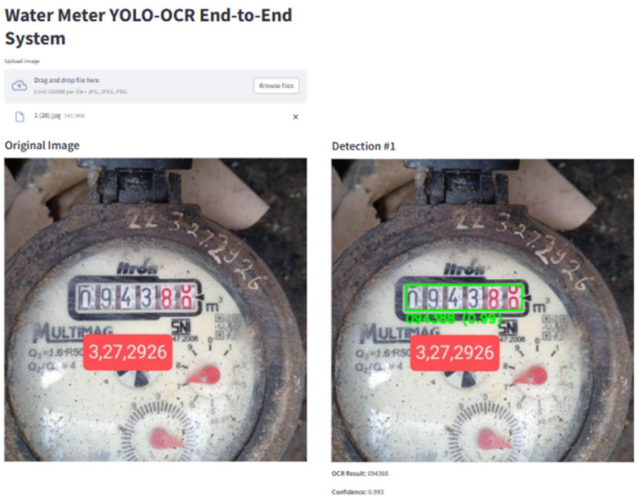
Compared with previous studies reporting recognition accuracies between approximately 93% and 97% on larger datasets, the recognition accuracy obtained in this study (90.90%) is slightly lower. However, the evaluation setting in this work intentionally includes visually challenging samples and transitional digit states that frequently occur in real meter reading scenarios. These conditions increase task difficulty but better reflect practical deployment environments.

B. Proposed System Performance

The proposed system combines a fine-tuned YOLOv8 detector and a PP-OCRv4 recognizer. The performance was evaluated using an end-to-end dataset of 55 images. With an accuracy of 90.90% and a CER of 0.0151, the model trained using only 1,000 non-ideal images outperformed the model trained on the full 3,250-image dataset. Additionally, near real-time performance was attained, with an average inference time per image of approximately 0.5 s. This measurement was obtained using an NVIDIA A100-SXM4 GPU (40 GB) on Google Colab, indicating server-level performance rather than real-time execution on embedded devices. Furthermore, Figure 9 illustrates the practical performance of the system. Figure 9(a) presents the end-to-end model visualization, showing the detection outputs and the recognized meter digits, whereas Figure 9(b) displays the user interface of the developed YOLO-OCR system, which demonstrates how the framework operates in real-world usage. Together, these results highlight the system's reliability, practical applicability, and robustness in real-world water meter digit recognition scenarios.

These results empirically indicate that exposure to visually challenging samples during training plays a critical role in improving model generalization. Specifically, the OCR model trained exclusively on non-ideal images demonstrates superior robustness and end-to-end recognition performance compared to models trained on larger but visually homogeneous datasets.

(a)



(b)

Fig. 9. (a) End-to-end model output visualization, (b) user interface of the developed YOLO-OCR water meter reading system.

IV. CONCLUSION

This study presents an automated water meter reading system that integrates You Only Look Once version 8 (YOLOv8) for digit display detection and PP-OCRv4 for Optical Character Recognition (OCR). Experimental results demonstrate that training on visually challenging data leads to stronger model generalization compared to larger datasets dominated by ideal conditions. After fine-tuning, the Adam-optimized YOLOv8 achieved the best detection performance, whereas the AdamW-optimized PP-OCRv4 delivered the highest recognition accuracy. Notably, the model trained on 1,000 non-ideal images outperformed the model trained on the full 3,250-image dataset, achieving 90.90% recognition accuracy with a Character Error Rate (CER) of 0.0151, highlighting the robustness of the proposed framework. A brief error analysis further indicates that most recognition errors occur under visually challenging conditions, particularly during partial digit transitions, lighting variations, or when visually similar digits such as 6 and 8 appear in overlapping regions.

Despite the promising results, several limitations remain. The dataset primarily consists of mechanical 6-digit water meters, which may limit generalization to digital or smart meter displays. Although robustness was evaluated under various real-world conditions, extreme environmental variations were not extensively represented. Additionally, inference time was measured on server-class GPU hardware, and performance may differ in mobile or edge deployment environments. Future work will explore lightweight model variants for embedded systems and incorporate temporal consistency mechanisms for video-based meter reading to further improve robustness and practical applicability.

DECLARATION OF COMPETING INTERESTS

The authors declare no conflict of interest.

ACKNOWLEDGMENT

The author would like to thank Perumdam Tirta Wijaya Cilacap for providing access to customer water consumption data used in this study.

DATA AVAILABILITY

The dataset used in this study were provided by an industry partner and are not publicly available due to confidentiality restrictions. Processed data may be available from the corresponding author upon reasonable request and with permission from the data provider.

AI USE AND DECLARATION OF GENERATIVE AI USE

The authors used generative AI tools (e.g., ChatGPT and Grammarly) to assist in language refinement and grammar improvement during manuscript preparation. All scientific content, analysis, and conclusions were developed and verified by the authors.

REFERENCES

[1] K. Satija and M. Chhatlani, "Infrastructure & Industrialization: An SDG-9 Study of Gujarat," *International Journal of Science and Social Science Research*, vol. 3, no. 1, pp. 310–315, June 2025, https://doi.org/10.63671/ijsssr.v3i1.430.

[2] A. J. Zapata-Sierra, E. Salmerón-Manzano, A. Alcayde, M. L. Zapata-Castillo, and F. Manzano-Agugliaro, "The Scientific Landscape of Smart Water Meters: A Comprehensive Review," *Water*, vol. 16, no. 1, Dec. 2023, Art. no. 113, https://doi.org/10.3390/w16010113.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 779–788, https://doi.org/10.1109/CVPR.2016.91.

[4] Sudianto, Y. Herdiyeni, A. Haristu, and M. Hardhienata, "Chilli Quality Classification using Deep Learning," in *2020 International Conference on Computer Science and Its Application in Agriculture*, Bogor, Indonesia, 2020, pp. 1–5, https://doi.org/10.1109/ICOSICA49951.2020.9243176.

[5] B. Satya, D. Manongga, Hendry, and A. Aminuddin, "Optimized YOLOv8 for Automatic License Plate Recognition on Resource Constrained Devices," *Engineering, Technology & Applied Science Research*, vol. 15, no. 2, pp. 21976–21981, Apr. 2025, https://doi.org/10.48084/etasr.9983.

[6] Y. Du *et al.*, "PP-OCR: A Practical Ultra Lightweight OCR System." arXiv, Oct. 15, 2020, https://doi.org/10.48550/arXiv.2009.09941.

[7] F. Khan, S. Rafique, and G. M. Khan, "Low-Cost Smart Metering Using Deep Learning," *International Journal of Innovations in Science & Technology*, vol. 6, no. 5, pp. 93–104, May 2024.

[8] B. Nguyen van *et al.*, "Water Meter Reading Based on Text Recognition Techniques and Deep Learning," *IEEE Access*, vol. 13, pp. 41422–41434, 2025, https://doi.org/10.1109/ACCESS.2025.3547225.

[9] H. Moussaoui *et al.*, "Enhancing automated vehicle identification by integrating YOLO v8 and OCR techniques for high-precision license plate detection and recognition," *Scientific Reports*, vol. 14, no. 1, June 2024, Art. no. 14389, https://doi.org/10.1038/s41598-024-65272-1.

[10] M. Imran, H. Anwar, M. Tufail, A. Khan, M. Khan, and D. Ramli, "Image-Based Automatic Energy Meter Reading Using Deep Learning," *Computers, Materials & Continua*, vol. 74, no. 1, pp. 203–216, Sept. 2022, https://doi.org/10.32604/cmc.2023.029834.

[11] M. Yaseen, "What is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector." arXiv, Aug. 28, 2024, https://doi.org/10.48550/arXiv.2408.15857.

[12] V. Afifah and S. Erniwati, "YOLOv8 for Object Detection: A Comprehensive Review of Advances,Techniques, and Applications," *IJACI : International Journal of Advanced Computing and Informatics*,

vol. 2, no. 1, pp. 53–61, Jan. 2026, https://doi.org/10.71129/ijaci.v2i1.pp53-61.

[13] C. Cui *et al.*, "PaddleOCR 3.0 Technical Report." arXiv, July 08, 2025, https://doi.org/10.48550/arXiv.2507.05595.

[14] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context." arXiv, Feb. 21, 2015, https://doi.org/10.48550/arXiv.1405.0312.

[15] H. Zhang and X. Yang, "A Survey on Algorithms and Performance Metrics for Object Detection," *International Core Journal of Engineering*, vol. 9, no. 9, pp. 133–145, Sept. 2023, https://doi.org/10.6919/ICJE.202309_9(9).0017.

[16] R. Taori, A. Dave, V. Shankar, N. Carlini, B. Recht, and L. Schmidt, "Measuring Robustness to Natural Distribution Shifts in Image Classification," in *34th Conference on Neural Information Processing Systems*, Vancouver, Canada, 2020, pp. 18583–18599.

[17] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain Generalization: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4396–4415, Apr. 2023, https://doi.org/10.1109/TPAMI.2022.3195549.

[18] S. Wang, R. Veldhuis, C. Brune, and N. Strisciuglio, "A Survey on the Robustness of Computer Vision Models against Common Corruptions." arXiv, Sept. 14, 2024, https://doi.org/10.48550/arXiv.2305.06024.