ICIC Express Letters
Part B: Applications
Volume **17**, Number **1**, January **2026**

ICIC International ©2026 ISSN 2185-2766
pp. 31–39

MODIFIED PARTICLE SWARM OPTIMIZATION AND SUPPORT VECTOR MACHINE FOR SOLVING WATER LEVEL MEASUREMENT

Chiabwoot Ratanavilisagul[1] and Somkiat Kosolsombat[2]

[1]Department of Computer and Information Sciences
Faculty of Applied Science
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pracharat 1 Road, Wongsawang, Bangsue, Bangkok 10800, Thailand
chaibwoot.r@sci.kmutnb.ac.th

[2]Data Science and Innovation
College of Interdisciplinary Studies
Thammasat University
99 Moo 18 Paholyothin Road, Klong Nueng, Klong Luang, Pathumthani 12121, Thailand
somkiatk@tu.ac.th

Received March 2025; accepted May 2025

Abstract. *Nowadays, flooding frequently occurs in many areas. To warn the population before this issue arises, cameras are used to monitor water levels on gauges in flood-prone areas. Recently, many research studies have attempted to accurately determine the water level line. A well-known technique for detecting the water level line with high performance is YOLO. However, a key aspect of monitoring water levels is reading the characters (numbers and symbols) above the water level line on the staff gauge. Reading these characters can utilize the same techniques used to solve the problem of Handwritten Character Recognition (HCR). To achieve a higher accuracy rate in reading these characters, this research proposes feature extraction using the Histogram of Oriented Gradients (HOG) technique, classification with a Support Vector Machine (SVM), and the application of Particle Swarm Optimization (PSO) to optimize SVM parameters. Experiments are conducted on a widely distributed dataset consisting of 1,200 images. The results demonstrate that the proposed method achieves superior accuracy compared to similar methods, underscoring its effectiveness in reading characters on the staff gauge.*
Keywords: Water level measurement, Particle swarm optimization, Support vector machine, Machine learning, Object detection, Histogram of oriented gradients

1. **Introduction.** Global warming has led to flooding, which threatens lives and property [1]. Water level measurement is essential for flood warnings and water resource management. To measure water levels, numerous studies have proposed various methods, such as contact measurement systems like capacitive sensors [2], and non-contact measurement systems like ultrasonic ranging systems [3]. However, both methods have disadvantages, including high capital investment and difficulties in practical implementation [1,4]. Due to these disadvantages, some researchers have proposed using surveillance cameras [5]. The advantages of this technique include low cost, easy installation, real-time capability, high accuracy, improved reliability, and simplified maintenance [1,4]. Many researchers employed image processing and machine learning techniques for water level detection. For instance, [1,6,7] use YOLO to detect the position of the water level line on the staff gauge. Experimental results show that YOLO is highly effective in detecting water level positions. Many researchers focus on accurately determining the water level line by calculating water level values based on the image height with a fixed ratio [1,7-9], or by transforming the original image into a virtual image and calculating water level values

DOI: 10.24507/icicelb.17.01.31

from the height of the virtual image [4,9,10]. These methods are highly effective, but they have a weakness: if the original image changes, the results may not be accurate, making them unsuitable for practical applications or general use [4]. If techniques are to be applied in practical applications, they should read the numbers above the water level, like the technique in [6]. Moreover, in practice, water level readings are often estimated to determine whether the water level exceeds a specified threshold. If this occurs, an alert can be issued. Therefore, precise water level measurement is rarely important. To develop a water level measurement technique that can be applied in practice, this research proposes using YOLO to detect the water level line position, and then reading the characters above the water level line on the staff gauge. The main concept of reading characters utilizes the same algorithm as HCR. The characters on the staff gauge are italicized and have unclear patterns, resembling handwriting [11]. This research focuses on improving the algorithm to enhance the accuracy rate of reading characters on the staff gauge. Recently, many researchers have been trying to solve HCR, employing various techniques such as PSO and SVM as follows. Ratanavilisagul [12] proposed a feature extraction method using HOG [13]. Ghadekar et al. [14] presented SVM classifiers and achieved good accuracy rates on standard datasets. This research aims to compare the effectiveness of reading characters on the staff gauge using CNN [16], YOLO [15], SVM [17], Extreme Gradient Boosting (XGBoost) [18], K-Nearest Neighbor (KNN) [19], Random Forest (RF) [20], and Logistic Regression (LR) [21], and to determine the most effective method for water level reading. The experimental results from this research show that SVM achieves the best performance among these methods, indicating its suitability for reading characters on the staff gauge. However, SVM has a drawback that is parameter adjustment [22]. To solve this problem, this research proposes using PSO to optimize SVM parameters. Consequently, this paper proposes an algorithm for reading characters on the staff gauge using HOG for feature extraction and SVM with parameter optimization through PSO for classification. Experiments are conducted on a widely distributed dataset consisting of 1,200 images. The results indicate that the proposed method outperforms other similar approaches in water level reading. The rest of this document is organized as follows: Section 2 reviews related work, Section 3 presents the proposed method, Section 4 discusses the experimental results, and Section 5 provides conclusions along with future research directions.

2. **Related Work.**

2.1. **Particle swarm optimization.** PSO was introduced by [23]. The concept of PSO is inspired by the movement of animal groups, such as flocks of birds, as they coordinate to locate the optimal food source. PSO is a stochastic algorithm used to solve optimization problems. PSO has been successfully applied to optimizing parameters in various machine learning models, including SVM. By finding optimal parameters, PSO can enhance the performance of the classifiers, leading to better recognition accuracy.

2.2. **Support vector machine.** SVM is a supervised learning model employed for both classification and regression tasks. Its primary goal is to identify the hyperplane that optimally divides the data into distinct classes. This hyperplane maximizes the margin between the classes. The decision function for SVM can be written as

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b \tag{1}$$

where α_i is the Lagrange multipliers, y_i is the class labels, x_i is the support vectors, $K(x_i, x)$ is the kernel function, and b is the bias term. The Radial Basis Function (RBF) kernel is a commonly used kernel in SVM, defined as

$$K(x_i, x) = \exp\left(-\gamma \|x_i - x\|^2\right) \tag{2}$$

where γ is a parameter that determines the width of the Gaussian function. In this research, the RBF kernel is employed for experiments involving PSO to optimize SVM parameters. The RBF kernel can handle non-linear relationships by mapping data into higher dimensions. The parameters of SVM using the RBF kernel include two key parameters [22]. The first parameter is "C" which is a hyper parameter in SVM that controls the error. A low value of C indicates low error, while a high value of C indicates high error. However, a low error does not necessarily mean that it has a good model. It depends on the dataset. The parameter γ (Gamma) is used with the Gaussian RBF kernel. If SVM uses a linear or polynomial kernel, it does not use gamma. γ determines how much curvature. A high gamma results in more curvature, while a low gamma results in less curvature. The decision to tune for high or low gamma depends on the dataset.

In this research, we utilize the scikit-learn library [24] to develop the algorithm. An example source code using the scikit-learn library is provided in Figure 1. According to [25], the default values for SVM are C = 1.0, kernel = 'RBF', and gamma that is defined as

$$\gamma = \frac{1}{\textit{the number of features} \times \textit{the variance of the input data}} \tag{3}$$

```
from sklearn.svm import SVC
Define the variable as follows:
X_train is features used to train the model.
y_train is target labels for the training data.
X_test is features used to evaluate the model's performance.
y_test is true labels for the testing data, used to assess the accuracy of the model's predictions.
# Fit the SVM model to the training data
model = SVC().fit(X_train, y_train)
# Use the trained model to make predictions on the test data
output = model.predict(X_test)
```

Figure 1. An example source code using the scikit-learn library

2.3. **You only look once.** YOLO, introduced by [15], is an algorithm that uses neural networks for real-time object detection. Currently, YOLO is widely used in various applications for object detection due to its speed and high accuracy in processing. The process of YOLO can be summarized as follows: The image is divided into equally sized sub-grids. Each grid is responsible for detecting objects within its boundaries, and each bounding box contains multiple features. Next, overlapping and incorrect bounding boxes are eliminated by evaluating the overlap between predicted and actual object locations.

For detecting the water level position, YOLO can accurately and efficiently identify the water level line. However, when applying YOLO to classifying characters on the staff gauge, it is often ineffective. This is because most images, when cropped to contain only the characters, become small in size. One major drawback of YOLO is its inefficiency in classifying small images [26]. As a result, YOLO is unsuitable for reading characters on the staff gauge.

2.4. **Convolutional neural network.** CNN is developed from artificial neural networks. CNN has proven effective in classifying data in tasks involving images and visual data [16]. CNN automatically learns spatial hierarchies of features through the application of convolutional layers, pooling layers, the fully connected layer, dropout layer, and output layer. The convolutional layer extracts various features of an image using a filter matrix (kernel). The pooling layer functions similarly to the convolutional layer; it is used to reduce the data size while preserving important image features. The fully connected layer is used for feature extraction and transforming the data into a one-dimensional vector (flatten). The dropout layer is used to prevent overfitting. The dropout technique randomly deactivates some processing units during training, reducing the model's dependence on specific neurons. The output layer applies activation functions such as softmax in the case of multi-class classification to generating the desired output. As mentioned, the design of the CNN architecture is complex. Defining the structure of the CNN is crucial for the performance of the model. If not designed appropriately, it can result in a poor model [27]. The complexity of designing the model means that CNN may be unsuitable for application in classifying characters on the staff gauge.

2.5. **A water level measurement approach based on YOLO.** The algorithm introduced by [6] can be summarized as follows. First, numbers (0 to 9) are used to train a model with CNN, and images of the water level line on the staff gauge are used to train a model with YOLO. Next, the YOLO model is used to identify the position of the water level line on the staff gauge, and the first series of characters above the water level line are classified by the CNN model. The proposed method is validated using images and videos from a monitoring station on a river in Beijing. The results demonstrate that the proposed method performs well. This research is called YOLO+CNN. However, this method has some weaknesses. The CNN structure must be designed appropriately; if the structure is not properly designed, it does not yield the expected results. Therefore, applying this technique in general is difficult.

3. **Proposal Work.** As previously mentioned, YOLO has proven to be effective in detecting water levels on the staff gauge. However, it is rarely efficient in reading the characters on the staff gauge because the characters are small. Meanwhile, the problem with CNN is that its structure must be designed to suit the images. Figure 2 illustrates that a standard staff gauge is typically 100 centimeters (cm) tall, with the letter 'E' or '∃' aligned with a numerical mark. The dash or space from the character 'E' or '∃' is 2 centimeters apart. In most cases, images captured by cameras show characters that appear slightly tilted, small in size, and inconsistent, as shown in Figure 3. The complexity of

FIGURE 2. The standard staff gauge

FIGURE 3. The original image

the characters on the staff gauge is similar to the complexity of handwritten character recognition. So, this research proposes applying HCR techniques to reading characters on the staff gauge. Recently, a technique combining HOG for feature extraction and SVM for classification has been introduced to effectively solve the HCR problem. So, this research proposes applying both techniques to reading characters on the staff gauge. Moreover, the drawback of SVM is the need for careful parameters adjustment. Adjusting parameters to build an effective model can impact model performance. To enhance the efficiency of model building with SVM, this research proposes using PSO to optimize SVM parameters. Techniques like PSO can further improve classifier accuracy by finding the optimal settings for the model, leading to a model with a higher accuracy rate. Additionally, the researchers experimented with other machine learning techniques such as XGBoost, LR, RF, KNN, CNN, YOLO, and SVM without PSO to compare with the proposed method and to demonstrate that PSO can enhance the performance of SVM. The proposed technique for reading characters on the staff gauge includes preprocessing, feature extraction using HOG, classification using SVM, and optimization of SVM parameters using PSO. This technique is called SVM+PSO. The proposed method consists of two phases: the training phase and the testing phase.

The training phase consists of the following steps.

1) The training of YOLO involves annotating images by marking the water level position using bounding boxes, as shown in Figure 4 (red rectangle). This annotated data is used to train the YOLO model to detect the water level position on the staff gauge.

FIGURE 4. The marking of the water level position

2) For the characters on the staff gauge, the original image is cropped to isolate a single character within a frame. Subsequently, image segmentation is used to trim the surrounding white space, ensuring that the characters are centered. The image is then resized to 20×20 pixels, as shown in Figure 5. The characters on the staff gauge can be divided into 14 classes as follows: '.', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '-', 'E', and '∃'. Then, thinning is a technique used to simplify the input data by removing any dots or the outer layer of a character until it becomes a single pixel in width. Following this, the HOG technique is employed to extract crucial features, which are then rescaled using Min-Max Normalization.

FIGURE 5. The characters used to train the SVM model

3) The results from step 2 are used to train the model using SVM, while PSO optimizes the SVM parameters, specifically the regularization parameter C and the kernel parameter γ.

The test phase consists of the following steps.

Step 1: Use the model from YOLO to detect the water level position. The lower boundary detected by YOLO, as shown in Figure 6, is then expanded upwards until the character 'E' or '∃' is found. The area from the lower boundary until it is expanded upwards is the copper, as shown in Figure 7.

Figure 6. The result from YOLO detecting the water level position

Figure 7. The result from cropping the image

Step 2: K-means clustering is used with the image in Figure 7 to classify the objects into segments, as shown in Figure 8. Each segment is processed as in step 2 of the training phase.

Figure 8. The image segmented by K-means clustering

Step 3: Use the SVM model from the training phase to classify all characters from step 2 and consider only the numbers. For example, a value of 30∃ is considered as 30, indicating that the water level is between 28 cm and 30 cm.

4. **Experiments and Results.** The datasets used in this research are obtained from public websites, and the image augmentation process is applied. The dataset can be downloaded from [28]. It is divided into two groups: the training set contains 1,000 images, while the testing set contains 200 images. In the testing set, only images with clearly visible characters on the staff gauge are selected, and these images do not overlap with those in the training dataset. These datasets are used to train and test the YOLOv8n model to detect the water level. For training the model using XGBoost, LR, RF, KNN, CNN, YOLO, SVM without PSO, and SVM with PSO, the characters on the staff gauge are cropped from images in the training set. Each character is trained with 100 characters per class. The performance metrics used in the experiments are described as follows. The Recognition Rate is calculated as the percentage of correctly identified predictions divided by the total number of data points. The Recognition Rate from the test phase serves as an indicator of the algorithm's classification performance, with higher Recognition Rate values corresponding to better classification performance. The parameters used in all experiments are as follows. For the PSO parameters, the PySwarms Library [29] is utilized. All PSO parameters use the default settings from this library, except for the

Table 1. Comparative the Recognition Rate of LR, RF, XGBoost, KNN, YOLO, SVM, YOLO+CNN, and SVM+PSO

Technique	**LR**	**RF**	**XGBoost**	**KNN**	**YOLO**	**YOLO+ CNN**	**SVM**	**SVM+ PSO**
Recognition Rate	48.50	75.50	74.00	76.50	50.00	69.50	78.50	98.50

swarm size, which is set to 20, and the maximum number of iterations, which is also set to 40. These values are selected to balance the trade-off between computational cost and optimization effectiveness. A smaller swarm size and less iteration reduce computation time while still allowing PSO to effectively search the parameter space. The CNN or YOLO+CNN parameters and structure are configured based on the recommendations outlined in the original paper [8], while HOG, RF, XGBoost, LR, and SVM parameters are configured based on standard values from the scikit-learn library. For the SVM+PSO parameters, the bounds for C and γ are set to $[1, 100]$ and $[0, 10]$, respectively. These bounds are chosen to provide a reasonable range for the SVM parameters, ensuring that the search space is neither too broad nor too narrow, which helps in efficiently finding the optimal parameters. This research is conducted on a personal computer with an Intel i7-14700F CPU running at 2.1 GHz, 16 GB of RAM, and Python 3 as the programming language.

The experimental results in Table 1 show that SVM surpasses LR, RF, KNN, YOLO, and XGBoost in terms of classification performance. The experiment used the same water level line detection and image cropping techniques, but different models for classifying the characters, as shown in the algorithms in Table 1. The results indicate that YOLO performs quite poorly in classification, likely because the cropped images are small, reducing its effectiveness. Meanwhile, CNN's internal structure may be unsuitable for this case, leading to poor performance. In the case where images are obtained from public websites, they come from multiple water level monitoring stations and various camera positions. Therefore, designing a CNN model suitable for solving this problem is difficult. Moreover, SVM with PSO, which uses the same feature extraction method, outperforms SVM without PSO, as demonstrated by its superior Recognition Rate. This suggests that optimizing parameters with PSO can improve character recognition performance on the staff gauge.

5. **Conclusion.** This research proposes a method for reading characters on the water level gauge using HOG for feature extraction, SVM for classification, and PSO for optimizing SVM parameters. The performance of the proposed method is evaluated on datasets obtained from multiple water level monitoring stations and various camera positions. Experimental results revealed that the proposed algorithm outperforms other algorithms, such as LR, RF, KNN, CNN, YOLO, XGBoost, and SVM. Furthermore, PSO enhances the performance of SVM. Therefore, the combination of HOG, SVM, and PSO provides the best classification performance for practical use. The disadvantage of the proposed method is that it lacks an image enhancement step before the process of detecting the water level position and the copper image. If the input image does not clearly display numbers or the water level, the proposed method will be unable to read the water level. For example, at night, the image may be too dark to detect numbers or the water level. Hence, the proposed method requires further development in the future. In practice, images captured by cameras are affected by various factors, such as water stains, lighting conditions, weather, daytime and nighttime variations, and other environmental influences. These factors can lead to errors in character recognition. Therefore, it is necessary

to develop preprocessing techniques to enhance image quality before applying the proposed method.

Acknowledgment. This research was funded by King Mongkut's University of Technology North Bangkok, Contract no. KMUTNB-68-BASIC-46.

REFERENCES

[1] J. Li, C. Tong, H. Yuan and W. Huang, A complex environmental water-level detection method based on improved YOLOv5m, *Sensors*, vol.24, no.16, pp.1-16, 2024.
[2] R. Katman and A. Kaewpoonsuk, An experimental kit of water level measurement with low-cost capacitive sensors, *ICIC Express Letters, Part B: Applications*, vol.13, no.6, pp.623-630, 2022.
[3] S. Ben-Yaakov, C. Golan and S. Kessler, A stand-alone ultrasonic ranging system for hydrological water stage measurements, *IEEE Transactions on Instrumentation and Measurement*, vol.41, no.5, pp.699-702, 1992.
[4] L. C. Kuo and C. C. Tai, Implementation of inverse perspective mapping for camera-vision water-level measurements, *Proc. of the 2020 International Computer Symposium (ICS 2020)*, pp.348-351, 2020.
[5] N. H. Jafari, X. Li, Q. Chen, C. Y. Le, L. P. Betzer and Y. Liang, Real-time water level monitoring using live cameras and computer vision techniques, *Computers & Geosciences*, vol.147, 104642, 2021.
[6] G. Qiao, M. Yang and H. Wang, A water level measurement approach based on YOLOv5s, *Sensors*, vol.22, 3714, 2022.
[7] M. Liu, C. Wang and W. Huang, Research on water level measurement technology based on the residual length ratio of image characters, *Signal, Image and Video Processing (SIViP)*, vol.18, pp.57-70, 2024.
[8] G. R. d. O. Fleury, D. V. do Nascimento, A. R. Galvão Filho, F. d. S. L. Ribeiro, R. V. de Carvalho and C. J. Coelho, Image-based river water level estimation for redundancy information using deep neural network, *Energies*, vol.13, 6706, 2020.
[9] D. Zhang and J. Tong, Robust water level measurement method based on computer vision, *Journal of Hydrology*, vol.620, Part B, 129456, 2023.
[10] Z. Xie, J. Jin, J. Wang, R. Zhang and S. Li, Application of deep learning techniques in water level measurement: Combining improved SegFormer-UNet model with virtual water gauge, *Applied Sciences*, vol.13, 5614, 2023.
[11] M. A. Mohamad, H. Hassan, D. Nasien and H. Haron, A review on feature extraction and feature selection for handwritten character recognition, *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol.6, no.2, 2015.
[12] C. Ratanavilisagul, Modified particle swarm optimization with resetting and jumping techniques for solving handwritten digits, *ICIC Express Letters, Part B: Applications*, vol.15, no.12, pp.1227-1234, 2024.
[13] N. Dalal and B. Triggs, Histograms of oriented gradients for human detection, *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, San Diego, CA, USA, vol.1, pp.886-893, 2005.
[14] P. Ghadekar, S. Ingole and D. Sonone, Handwritten digit and letter recognition using hybrid DWT-DCT with KNN and SVM classifier, *Proc. of the 2018 4th International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, pp.1-6, 2018.
[15] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, You only look once: Unified, real-time object detection, *Proc. of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.779-788, 2016.
[16] W. Liu, J. Wei and Q. Meng, Comparisons on KNN, SVM, BP, and the CNN for handwritten digit recognition, *Proc. of the 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, pp.587-590, 2020.
[17] S. Ding, B. Qi and H. Tan, Review of support vector machine theory and algorithm research, *Journal of University of Electronic Science and Technology of China*, vol.40, pp.2-10, 2011.
[18] T. Q. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, *Proc. of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, pp.785-794, 2016.
[19] S. G. Kulkarni and M. V. Babu, Introspection of various K-nearest neighbor techniques, *UACEE International Journal of Advanced Computer Science and Applications*, vol.3, no.2, pp.103-106, 2013.

[20] S. Bernard, S. Adam and L. Heutte, Using random forests for handwritten digit recognition, *Proc. of the 9th International Conference on Document Analysis and Recognition (ICDAR 2007)*, Curitiba, Brazil, pp.1043-1047, 2007.
[21] D. K. Kiran, P. A. Chandawale and A. A. Sonawane, Handwriting recognition by logistic regression and neural networks, *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol.8, no.4, pp.245-251, 2022.
[22] S. Karamizadeh, S. M. Abdullah, M. Halimi, J. Shayan and M. J. Rajabi, Advantage and drawback of support vector machine functionality, *2014 International Conference on Computer, Communications, and Control Technology (I4CT)*, Langkawi, Malaysia, pp.63-65, 2014.
[23] J. Kennedy and R. Eberhart, Particle swarm optimization, *Proceedings of ICNN'95 – International Conference on Neural Networks*, vol.4, pp.1942-1948, 1995.
[24] *Scikit-Learn Library*, https://scikit-learn.org/stable, Accessed on April 14, 2025.
[25] *Default Parameter of SVM*, https://scikit-learn.org/dev/modules/generated/sklearn.svm.SVC.html, Accessed on April 14, 2025.
[26] J. Wang, S. Jiang, W. Song and Y. Yang, A comparative study of small object detection algorithms, *2019 Chinese Control Conference (CCC)*, Guangzhou, China, pp.8507-8512, 2019.
[27] R. A. Ravikumar, H. Sriraman, P. M. Sai Saketh, S. Lokesh and A. Karanam, Effect of neural network structure in accelerating performance and accuracy of a convolutional neural network with GPU/TPU for image analytics, *PeerJ Computer Science*, 2022.
[28] *Water Level Gauges Dataset*, http://chaibwoot.no-ip.org/Dataset/Water Level Gauges Dataset.zip, Accessed on April 14, 2025.
[29] *PySwarms Library*, https://pyswarms.readthedocs.io/en/latest/api/pyswarms.base.html, Accessed on April 14, 2025.