Intelligent Mobile-Based Detection of Shrimp Weight Anomalies Using Random Forest Regression

Kanjana Eiamsaard[1], Kathawach Satianpakiranakorn[2], Anan Banharnsakun[3], and Jirawat Jitprasutwit[4*]

[1,2,3,4]Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University Sriracha Campus, Thailand

Email: kanjana@eng.src.ku.ac.th, kathawach@eng.src.ku.ac.th, ananb@ieee.org, jirawat@eng.src.ku.ac.th[*]

Received: December 11, 2024 / Revised: July 28, 2025 / Accepted: August 13, 2025

***Abstract*—Shrimp is one of the most widely consumed seafood items globally, yet consumers frequently encounter fraud, such as weight manipulation through adulteration injections, which poses significant health and economic risks. This research presents a practical system for detecting anomalies in shrimp weight. A cross-platform mobile application has been developed to classify shrimp as either normal or abnormal in weight. The application integrates a shrimp segmentation model, developed using Mask R-CNN, and a weight prediction model based on the random forest algorithm, utilizing features such as area, perimeter, length, and width of the shrimp image. The weight prediction model achieves a value of 0.821 and a Mean Absolute Error (MAE) of 1.786 grams, which is less than 10% of the average shrimp weight in the dataset. Final classification is performed by comparing the predicted weight with the actual weight, measured using a 7-segment digit recognition module. The developed mobile application represents a novel integration of machine learning with mobile technology to address both non-adulterated and adulterated shrimp scenarios. It offers a reliable, accessible tool for consumers to detect weight-based adulteration, thereby helping to mitigate health risks and economic losses in the seafood supply chain.**

***Index Terms*—Anomaly Detection, Weight Prediction, Shrimp Fraud, Authentic Food, Food Engineering, Random Forest**

I. Introduction

Authentic food plays a vital role in human well-being. Today, various efforts, both digital and non-digital [1]-[3], are being employed to enhance the seafood supply chain. These efforts encompass improvements in farming [4]-[7], classification [8]-[10], and most importantly, the elimination of fraud utilizing both technology [11]-[14] and policy [15]-[17]. There are nine recognized types of fraud, commonly referred to as the 'nine sins of seafood'. Such fraudulent practices undermine food integrity, pose risks to public health, and have adverse economic impacts.

Undeclared product extension is a type of fraud that involves using technology to increase the perceived weight of seafood. For example, injecting gelatin-like substances, derived from animal skins and bones, into shrimp can increase their weight by 20-30% [18] and also make them appear larger. This practice poses a serious threat to international standards by endangering consumer health through various side effects.

Shrimp is the most-consumed seafood product in the United States. U.S. citizens face the problem of shrimp fraud, which led to the founding of the Southern Shrimp Alliance, an organization that works to protect millions of U.S. shrimp consumers from such fraud. This problem is not limited to the U.S. but also occurs in Japan and many other countries. Since ASEAN is one of the biggest shrimp exporters, its member nations have been experiencing incidents of fraud and adulteration for the past 20 years [19]-[21].

Detecting adulteration that increases shrimp weight is challenging without scientific laboratory tools. This issue could potentially be addressed through shrimp weight prediction or estimation research. In studies [22], [23], a specialized setup was utilized to capture the shadow of the shrimp area, with weight predictions made using power and forced power equations. Another study [24] proposed a method for predicting the weight of shelled shrimp using machine vision. This approach involved predicting weight through a polynomial equation based on the shrimp's area and perimeter pixel features, which were extracted using image processing methods. However, this system is not user-friendly due to the prerequisite hardware and the software being developed for personal computers. In study [25], a computer program was proposed for estimating the weight of Vannamei shrimp. The program utilized the number of pixels in conjunction with a non-linear regression equation for weight estimation. However, the system has limitations, as shrimp with the same number of pixels can have significantly different weights. While previous studies contribute valuable techniques, they are often constrained by limited accuracy in adulterated cases, reliance on specialized equipment, and lack of real-time, mobile-ready deployment. Most existing systems are designed for desktop platforms, which limits their usability for smartphone consumers. Currently, few solutions fully provide an integrated, consumer-accessible system capable of detecting shrimp weight anomalies in practical environments.

To provide a solution tailored for real-life conditions, it is essential to have convenient tools for detecting anomalies in shrimp weight, especially tools that are easy to use. The primary benefit of such a system is its accessibility, as it eliminates the need for specialized laboratory equipment, making it suitable for both shrimp consumers and businesses. By accurately detecting weight-based adulteration in shrimp, the system mitigates the risks associated with consuming products injected with gelatin-like substances, thereby protecting public health. Additionally, it promotes economic integrity by preventing fraudulent practices in the seafood supply chain, which can lead to financial losses for consumers and businesses alike. The adoption of this tool enhances consumer trust and promotes adherence to international food safety standards, contributing to a healthier and more transparent seafood industry.

Given these advantages, this research proposes a real-time system for detecting shrimp weight anomalies, thereby bridging the gap between research innovations and practical consumer needs. The developed system enables users to determine whether a shrimp's weight is normal or abnormal by capturing an image of the shrimp placed on a digital weighing scale, with the output simply indicating either 'Normal' or 'Abnormal'. The system consists of a cross-platform mobile application that integrates shrimp detection, morphological feature extraction, 7-segment digit recognition, and weight prediction.

II. Objective

This study aims to develop an intelligent and practical mobile application for shrimp consumers that can detect weight-based adulteration in shrimp.

III. Materials and Methods

The proposed system consists of a cross-platform mobile application, Shrimp, a digital weighing scale, and an Application Programming Interface (API) server. The process begins with the user capturing a top-view photograph of a shrimp placed on the digital weighing scale using the developed mobile application. This image is then transmitted to the API server over a wireless network. Upon receiving the image, the API server processes it to classify the result of shrimp weight anomaly detection, as illustrated in Fig. 1.

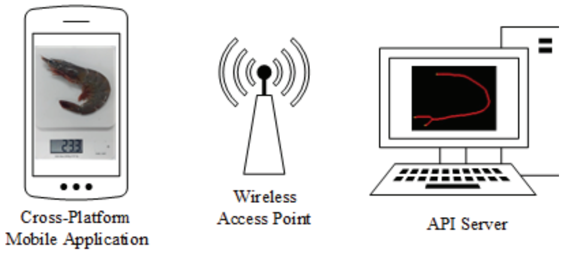


Fig. 1. Proposed system overview

The system workflow starts with the user capturing a photo of the shrimp placed on a digital weighing scale using the mobile application. The system then separates the shrimp image from the scale display to recognize the digits on the 7-segment display. Simultaneously, it generates a shrimp mask to extract the necessary features. The shrimp weight is predicted using a machine learning model. These two results—the actual weight from the scale and the predicted weight—are then compared to classify the shrimp as either 'normal' or 'abnormal'. Finally, all input and output data are stored in a cloud database, and relevant information is presented to the user via the mobile application, as shown in Fig. 2. The proposed system was developed across seven modules, which are described in the following subsections.

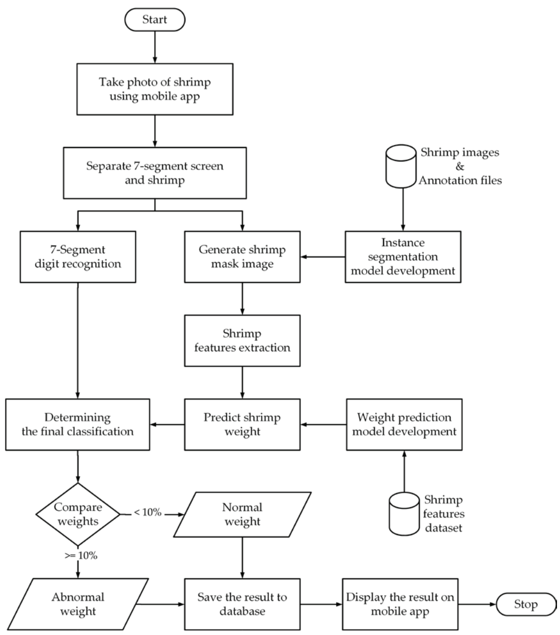


Fig. 2. The proposed system workflow

1) Cross-Platform Mobile Application Development

The mobile application was developed in two parts: the frontend and the backend. Flutter was chosen as the frontend framework due to its ability to build mobile, web, and desktop applications from a single codebase. In this research, which aims to provide a practical approach for identifying anomalies affecting shrimp weight, the mobile application serves multiple functions: Capturing images of shrimp on a digital weighing scale, detecting the 7-segment display, and providing the user interface for interacting with the instance segmentation and weight prediction model.

As illustrated in Fig. 1, the proposed system utilizes an API server to handle high-computation tasks, a common approach in research within this field [26]. FastAPI was chosen as the backend development framework due to its high performance in API management [27]. By employing this technique, the system avoids the challenges of deploying machine learning models directly on mobile devices [28].

Two API endpoints were implemented: the first utilizes the HTTP POST method for preprocessing and uploading shrimp images to cloud storage, along with storing related data in a cloud database. The second endpoint employs the HTTP GET method to return shrimp weight predictions, incorporating both feature extraction and the execution of a shrimp weight prediction model.

The authentication system, including sign-in and sign-up modules, seamlessly integrates with cloud-based authentication services. Additionally, data such as shrimp images, user locations, and prediction results are securely stored in a NoSQL database, leveraging appropriate cloud infrastructure.

2) 7-Segment Digit Recognition

In this research, digit recognition from 7-segment displays was facilitated using a point-by-point color comparison method. This technique analyzes the color at seven key points of the 7-segment display, converting this data into digital values, as shown in Fig. 3. This method ensures accurate and rapid digit recognition under varied lighting conditions, outperforming traditional OCR methods, which often struggle with segmented displays [29]. As shown in Fig. 4 (a-c), this process involves converting the original image to binary and setting the reading points of the 7 segments.

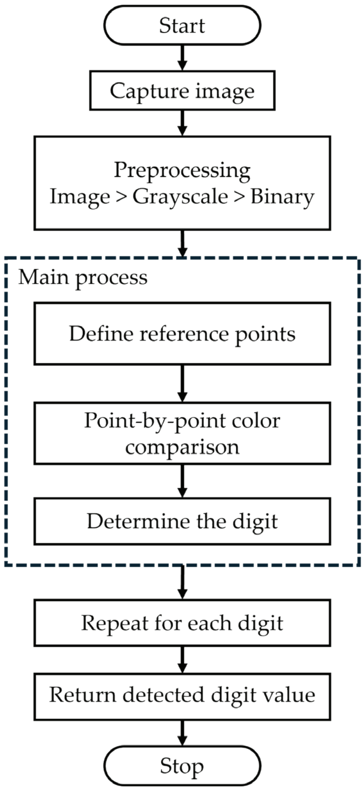


Fig. 3. The 7-Segment digit recognition workflow

(a)
(b)
(c)

Fig. 4. The 7-Segment digit recognition process: (a) Original image of numbers from a digital scale, (b) Image converted to binary, (c) Setting the reading points of the 7 segments

A notable limitation of this method is the need to define a precise frame for capturing images, which can be challenging and lead to errors if not done correctly. Ensuring that the picture frame matches the reference in the application can be difficult, potentially affecting the accuracy of digit recognition. To address this, integrating LiDAR technology in smartphones was considered. LiDAR can dynamically adjust the focus and framing based on distance, facilitating optimal image capture. However, the limited availability of LiDAR in high-end models restricts widespread use, indicating a need for future research on more accessible solutions [29].

3) Shrimp Dataset Collection

The shrimp dataset used in this research was gathered using four different mobile devices. Each captured image showed a shrimp placed on a digital weighing scale. As shown in Fig. 5, the dataset included variations in the digital weighing scale, the shrimp's pose, and the mobile device used for image capture. The dataset contained a total of 1,286 images.

Fig. 5. Examples of images in the dataset

4) Instance Segmentation Model Development

A shrimp instance segmentation model was developed using the Mask R-CNN, utilizing MMDetection, a comprehensive toolbox for object detection and instance segmentation. Built on PyTorch and distributed under an open-source license, MMDetection supports numerous well-known models such as Mask R-CNN, YOLO, and Cascade R-CNN. It has been benchmarked using standard datasets, including COCO, PASCAL VOC, and Cityscapes, and is optimized for both speed and memory efficiency during training and inference. Additionally, MMDetection's modular architecture allows developers to easily modify the toolbox to suit specific requirements [31].

This process involved both dataset annotation and model development. The VGG Image Annotator was used to annotate the dataset and establish ground truth. Model development focused on tuning dataset-specific, model-specific, and training-specific hyperparameters.

During annotation, each shrimp image was marked with a polygon outlining the shrimp's edges, and the entire shrimp was labeled as 'white_shrimp'. The annotated dataset was then formatted according to the COCO standard.

The developed model was evaluated using the CocoMetric class, which supports various standard metrics, including Average Precision and Average Recall at different Intersection over Union (IoU) thresholds. These metrics assess both bounding box and instance segmentation performance. Once the desired metrics were achieved, the final model produced a mask image as output. In this mask image, black pixels represent the shrimp, and white pixels represent the background.

5) Shrimp Features Extraction

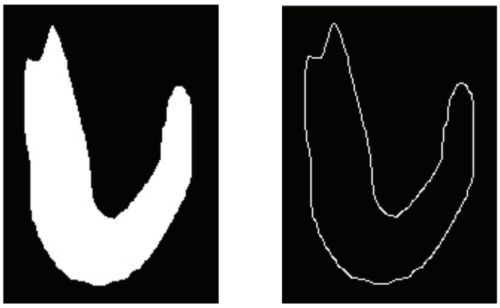
Regression analysis is the primary method for developing shrimp weight prediction models [32], [33]. While various imagery techniques can extract predictive features, including statistical, textural, and color-based features [34], [35] morphological features have shown the strongest correlation with shrimp weight. This is because shrimp have a distinctive morphology, with curved bodies exhibiting a weight directly proportional to their size. Four key morphological features are extracted from shrimp images: Total area, perimeter, head-to-tail length, and maximum width. These measurements are expressed in pixels and derived using pixel counting, contour analysis, skeletonization, and circular base measurement, respectively [36].

This investigation employed a four-step feature extraction process to derive these morphological traits from shrimp images, which are crucial for accurate weight prediction. Each step is detailed below.

First, the shrimp's area was quantified. This was achieved by counting the black pixels within the mask image, with each pixel representing a part of the shrimp to provide a total area. Fig. 6 (a) shows the area inside the shrimp.

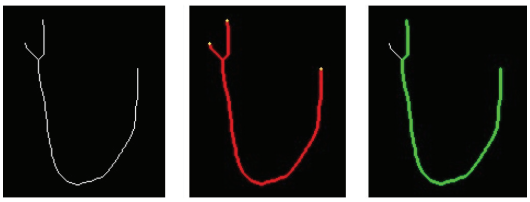
Second, the perimeter was calculated. The shrimp's contour was detected in the mask image, and the number of pixels along the contour was counted. Fig. 6 (b) shows the perimeter of the shrimp.

Third, the shrimp's length was determined. The mask image was skeletonized, as depicted in Fig. 7 (a), and endpoints were identified on the resulting structure, as shown in Fig. 7 (b). Distances between all pairs of endpoints were calculated. The endpoints with the maximum distance were used to denote the shrimp's head-to-tail length, as illustrated in Fig. 7 (c).



(a) (b)

Fig. 6. The process of determining shrimp area and perimeter: (a) The area of the shrimp, (b) The perimeter of the shrimp



(a) (b) (c)

Fig. 7. The process of determining shrimp length: (a) Skeleton from the mask image, (b) Endpoints on the skeleton, (c) Calculated length between endpoints

Finally, the shrimp's width was measured. This was achieved by iteratively placing increasingly larger circles over the mask image until the shrimp region was fully encompassed. The diameter of the encompassing circle represents the shrimp's width. To enhance efficiency, the Multiscale Approximation (MSA) technique was employed, downscaling the image to quickly identify the optimal placement for this circle. Fig. 8 shows the width determination results, which were verified using 16 points by checking the number of white pixels within the circle.

The extracted features and their corresponding weights in grams were compiled into a structured dataset. This dataset was used for the development of the weight prediction model, facilitating further analysis and model training.

6) Weight Prediction Model Development

The weight prediction model developed in this research is a supervised learning model, implemented using both linear and non-linear regression algorithms. For non-linear regression, the Random Forest algorithm, which leverages the bagging ensemble learning method, was designed and utilized. A Random Forest consists of multiple base learners, denoted as decision trees, each trained using a sample from feature randomness. Once each learner has produced a continuous value, all values are averaged to produce the final output, as shown in Fig. 9.

By using the bagging ensemble algorithm, Random Forests address the high variance of decision tree models without increasing bias. Moreover, bagging requires less computation time than most machine learning algorithms when training on large datasets [37], and it has been utilized across several research domains [38], [39].

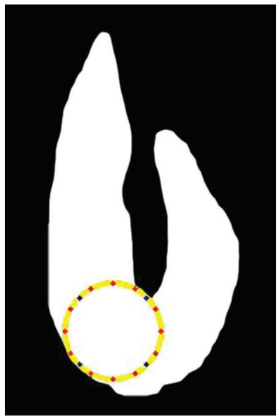


Fig. 8. The shrimp's width is measured by iteratively placing a 16-point circle entirely encompassing the shrimp region

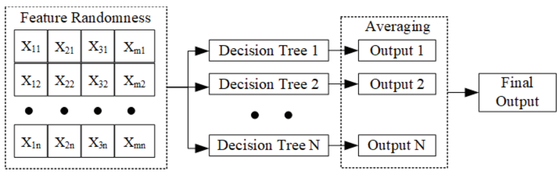


Fig. 9. Bagging ensemble algorithm in random forest regression.

The developed model utilizes features extracted from the shrimp feature extraction process. Specifically, the features denoted as area, perimeter, length, and width are considered as independent variables, while the weight is treated as the dependent variable. Before model development, the entire dataset was visualized to identify and eliminate outliers. The dataset was then randomly split into training and testing sets. All features in the dataset were standardized as in (1), where z represents the standardized samples, is the original sample, x is the mean of the sample, μ and is the standard deviation of the training samples.

$$z = \frac{x-\mu}{s} \tag{1}$$

The model and dataset-specific hyperparameters were tuned during the development of the random forest model. This included the number of trees in the forest, the maximum depth of the trees, the minimum number of samples required at a leaf node and to split an internal node, and the number of features to consider when determining the best split.

To evaluate the model's performance and robustness, the coefficient of determination (R^2) and the Mean Absolute Error (MAE) were used. A higher indicates a better fit between the model and the data, while a lower MAE indicates a smaller difference between the predicted and actual values. Additionally, the model evaluation was repeated 30 times, and the average and MAE were recorded as the results.

7) Determining the Final Classification

The proposed system produces two possible classifications: 'Normal' and 'Abnormal'. 'Normal' indicates that the shrimp is not subject to weight-increasing fraud, while 'Abnormal' signals the detection of an anomaly in the shrimp's weight.

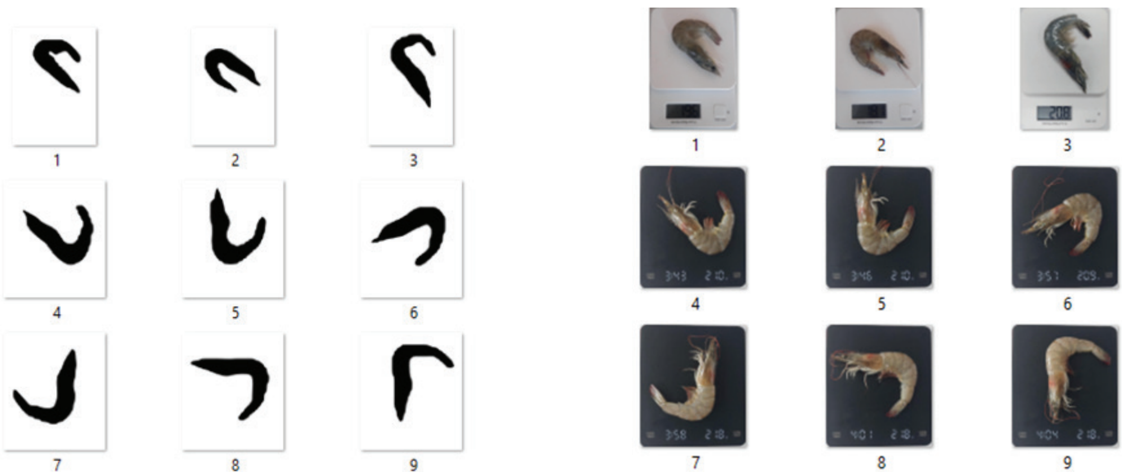
To determine the final classification, the system integrates recognized digits from the 7-segment digit recognition module with the predicted shrimp weight generated by the weight prediction model. If the variation between these two numbers does not exceed 10% of the average weight in the dataset (1.816 grams), the system classifies the shrimp as 'Normal'. Conversely, if the variation exceeds 10%, the shrimp is classified as 'Abnormal'.

IV. Experiment Setup

Three experiments were set up to align with the proposed method described in the previous section. These experiments are detailed in the following subsections.

1) Generate a Shrimp Mask Image

The shrimp mask images, as illustrated in Fig. 10 (a), were generated using the instance segmentation model proposed in subsection 2. 4. The corresponding shrimp dataset, depicted in Fig. 10 (b), was then divided into training, validation, and testing sets with a ratio of 70:20:10. To further enhance the diversity of the training set, data augmentation was applied using the random flip technique.



(a) (b)

Fig. 10. The comparison of: (a) shrimp mask images obtained from the instance segmentation model, (b) shrimp images captured from a mobile application

A pre-trained ResNet-50 model was utilized as the backbone for development. The model was trained for 12 epochs using the SGD optimizer, with a learning rate of 0.0025, a momentum of 0.9, and a weight decay factor of 0.0001. The evaluation results, reported using CocoMetric, include a segmentation mean Average Precision (mAP) of 0.601 at IoU thresholds ranging from 0.5 to 0.95 in increments of 0.05, and a segmentation mean Average Recall (mAR) of 0.648 at the same IoU thresholds.

2) Feature Analysis And Visualization

The extracted features from the proposed method were analyzed using descriptive statistics, as shown in Table I, to summarize the dataset. This analysis provides insights into the central tendency and variability of each variable. Due to differences in units between the dependent variable (weight), measured in grams, and the independent variables, measured in pixels, a log scale transformation was applied to all variables. The results were visualized using a box-and-whisker plot, as shown in Fig. 11, to better illustrate variability. The plot reveals that "Weight" exhibits high variance, with "Area" displaying the highest variance among the independent variables. Additionally, the other independent variables also demonstrate significant variability.

TABLE I
DESCRIPTIVE STATISTICS OF THE COLLECTED DATASET

Variable (Unit)	Range (max-min)	Mean	S.D.
Area (pixels)	2220	1864.872	426.384
Perimeter (pixels)	177	287.461	33.311
Length (pixels)	72	115.739	13.323
Width (pixels)	24	19.076	3.122
Weight (grams)	25.5	18.160	5.357

This analysis suggests that the selected variables are well-suited for regression model development, as high variance in both independent and dependent variables improves the model's ability to effectively capture diverse patterns and relationships within the data [40], [41].

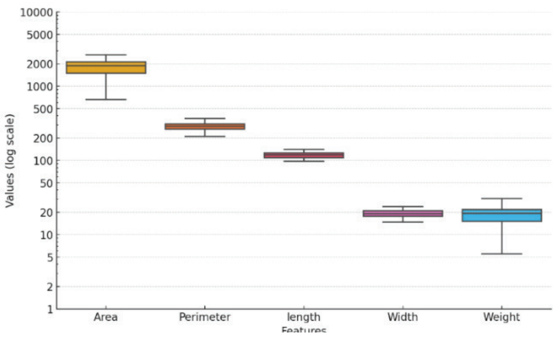


Fig. 11. Log-scaled box-and-whisker plot of variables in the dataset

Additionally, the scatterplot matrix and correlation coefficients, as illustrated in Fig. 12 and Fig. 13, were visualized to understand the strength of the relationships between the independent variables and the dependent variable. This analysis aids in the model and feature selection process by identifying which variables are most relevant for predicting the target outcome.

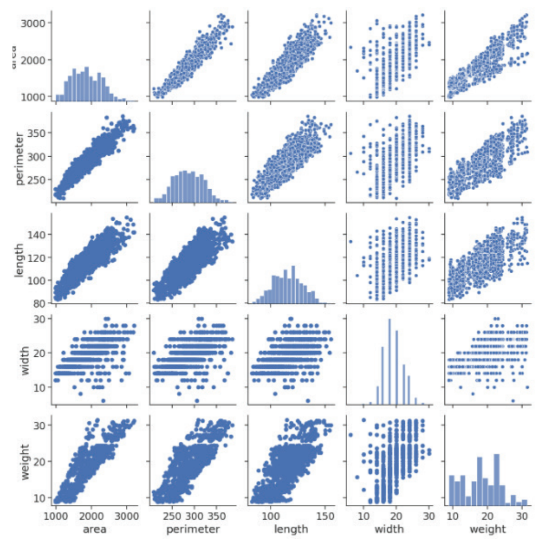


Fig. 12. Scatterplot matrix of variables in the shrimp dataset

Fig. 13. Correlation coefficient heat map of the extracted shrimp features

3) Weight Prediction Model Training

This experiment focuses on the model and feature selection process. Both linear and non-linear algorithms were explored based on the patterns observed between the independent and dependent variables. Initially, a linear regression algorithm was employed to develop a weight prediction model using shrimp area, as this feature exhibited the highest correlation coefficient with the dependent variable. The dataset was divided into training and testing sets, with 80% used for training and 20% for testing, to evaluate model performance. The experimental results showed that shrimp weight can be predicted using (2), where y denotes the predicted weight and $area$ represents the number of black pixels in the shrimp mask image. The model evaluation yielded a R^2 value of 0.777 and an MAE of 2.021.

$$y = -1.9634 + 0.0108 \times area \quad (2)$$

As the performance of the initial linear model was unsatisfactory, a multiple linear regression model was developed. To avoid multicollinearity, the correlation coefficients in Fig. 13 were used for feature selection, with a correlation threshold set at 0.8 [42]. This process led to the selection of shrimp area and shrimp width as features. The resulting model is presented in (3), where y represents the predicted weight, $area$ refers to the number of black pixels in the shrimp mask image, and $width$ denotes the diameter of the largest circle that fits within the shrimp mask. The model evaluation yielded an R^2 value of 0.779 and an MAE of 2.01.

$$y = -2.4994 + (0.0105 \times area) + (0.0569 \times width) \quad (3)$$

The two linear models developed earlier were still unsatisfactory. Therefore, an ensemble model, specifically a Random Forest, was employed. This model was developed using GridSearchCV to perform an exhaustive search through a specified hyperparameter grid, as shown in Table II.

Additionally, 10-fold cross-validation was applied during this process to improve the model's generalization.

TABLE II
SPECIFIED HYPERPARAMETERS

Hyperparameter	Value	Purpose
n_estimators	300	To specify the number of trees in the forest, which can improve model accuracy.
max_depth	5	To specify the maximum depth of each tree, which can prevent model overfitting.
min_samples_leaf	5	To specify the minimum number of samples required to be at a leaf node, which can improve the model's generalization.
min_samples_split	1%	To specify the minimum fraction of samples required to split an internal node, which can prevent model overfitting.
max_features	Square Root	To specify the number of features considered at each split as the square root of the total number of features, which can help reduce model variance.

V. RESULTS

1) *Weight Prediction Model*

The proposed model in this study was developed using Random Forest Regression and trained on the collected shrimp dataset, with 80% of the data used for training and 20% reserved for model evaluation. The training phase included hyperparameter tuning and 10-fold cross-validation to improve generalization. To enhance the reliability of the results, the evaluation process was repeated 30 times, and the average performance metrics were recorded. The evaluation results for the weight prediction model are presented in Table III.

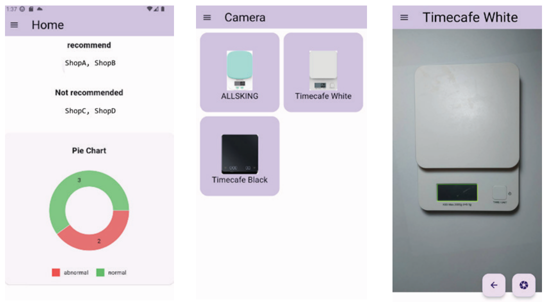
The results indicate that the weight prediction model developed using random forest best fits the observed data, as reflected by the value. Additionally, the average magnitude of the errors, represented by the MAE, is the lowest among the compared algorithms.

TABLE III
WEIGHT PREDICTION MODELS PERFORMANCE

Regression Model	Feature	R^2	MAE
Linear	Area	0.777	2.021
Multiple Linear	Area & Width	0.779	2.010
Random Forest	All Features	0.821	1.786

2) *Mobile Application*

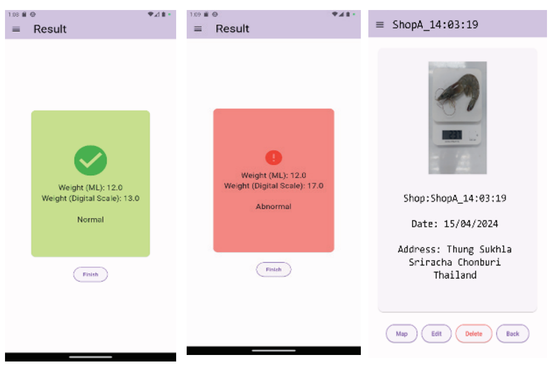
Upon opening the developed application, users can create their accounts or log in using third-party authentication. On the home page, users can view summarized information, including recommendations on places to buy shrimp and a pie chart illustrating the proportion of normal and abnormal shrimp weights they have predicted. To predict shrimp weight, users can select the Camera menu from the drawer menu in the top left corner of the screen. The application will then display images of various digital weighing scales, allowing users to select the one that matches their own. Subsequently, the mobile camera is activated, and a small green square appears to manually specify the location of the 7-segment display. The mentioned features of the application are illustrated in Fig. 14 (a)-(c).



(a) (b) (c)

Fig. 14. The developed mobile application: (a) Home page, (b) Camera page, and (c) Green square on 7-segment display

To detect anomalies in shrimp weight, the user must place the suspicious shrimp on a digital weighing scale while simultaneously capturing an image. Then, they press the prediction button to send the entire image to the API server. If the predicted weight does not differ from the weight displayed on the 7-segment display by more than two grams, the system classifies the shrimp as 'Normal', and the result shown in Fig. 15 (a) will be displayed to the user. Otherwise, Fig. 15 (b) will be shown, indicating an 'Abnormal' classification. Moreover, a prediction history is provided within the developed application, as shown in Fig. 15 (c), to help users identify which stores are safe for purchasing shrimp.



(a) (b) (c)

Fig. 15. The developed system: (a) Normal shrimp weight result, (b) Abnormal shrimp weight result, (c) Shrimp weight prediction history

To verify the developed application, both a non-adulterated shrimp detection scenario and an adulterated shrimp detection scenario were tested. In the first scenario, a non-adulterated shrimp, with an actual weight of 24.2 grams, was placed on a digital weighing scale. The developed mobile application captured an image of the shrimp, which was then sent to the API server. The server generated a shrimp mask, extracted all morphological features, predicted the weight from the obtained features, and determined the final classification. The feedback from the API server was 'Normal', indicating that the developed system successfully passed this test.

In the second scenario, an adulterant weighing 3.3 grams was injected into the same shrimp used in the first scenario, resulting in a total weight of 27.5 grams. The testing process was identical to the first scenario. However, the result from the API server was 'Abnormal', indicating that the developed system also successfully passed this test.

VI. Discussion

The experimental results show that the Random Forest Regression algorithm, utilizing all extracted features, significantly outperformed both linear and multiple linear regression models within the scope of this experiment's dataset. This superior performance comes from the ensemble approach of Random Forests, where the predicted weight is an aggregation of numerous decision trees. Each tree is constructed using feature randomness and the bagging ensemble learning technique, which enhances the model's robustness. Additionally, the diverse conditions under which the individual trees are created contribute to the model's ability to generalize effectively. The final prediction, being an average of all tree outputs, results in a more reliable and generalized weight prediction model. It's worth noting that the success of the Random Forest model is due not only to the algorithm itself but also to the careful selection of relevant features, which played a crucial role in optimizing model performance.

The mobile application for detecting anomalies in shrimp weight was developed using the Dart programming language. It interfaces with image processing and weight prediction components through APIs implemented in Python. Python was chosen for image processing because it's more efficient than Dart for those tasks. While machine learning models could be deployed directly on mobile devices, APIs were used for the weight prediction component to ensure consistent performance. This server-side approach guarantees uniform model accuracy, regardless of users' varying hardware capabilities, making the system more reliable and scalable. Furthermore, the application's features are specifically designed to detect undeclared product modifications that increase the perceived weight of shrimp. This information is valuable for consumers, helping them avoid buying from sources previously associated with adulterated shrimp.

VII. Conclusion and Future Work

This research aimed to develop a practical approach for individuals seeking to avoid shrimp weight fraud. The proposed method focuses on daily usability and feasibility. Recognizing the pervasive use of mobile devices and their synergy with AI, a mobile application was developed as the user interface for the anomaly detection module. This module ingeniously combines an instance segmentation model with a weight prediction model.

The system identifies weight anomalies through sophisticated image analysis. It starts by accurately segmenting shrimp from their background using Mask R-CNN, chosen for its excellent object segmentation capabilities. This method proved highly precise and efficient, yielding clear shrimp outlines crucial for the developed application.

After segmentation, vital morphological features, including area, perimeter, length, and width, were extracted. While area and perimeter were simple to calculate, determining length and width demanded more complex, computationally intensive methods. To address this, a shortest path algorithm and a circle-fitting method with reduced point frequency were implemented, allowing for efficient extraction of the necessary numerical data.

These extracted features are then fed into the shrimp weight prediction model. Experimental results show that the model, built with the Random Forest algorithm and leveraging all four features (area, perimeter, length, and width), significantly outperformed other models. It achieved an value of 0.821 and a Mean Absolute Error (MAE) of 1.786 grams, successfully meeting the proposed system's criteria.

However, this research could be further improved by developing an automated 7-segment screen detector. This improvement would allow the system to adapt to a wider variety of weighing scales, greatly increasing its flexibility and real-life applicability.

Acknowledgments

This research received no external funding. The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author on reasonable request. The authors would like to thank the Faculty of Engineering at Sriracha, Kasetsart University, for providing the facilities for experimentation. The authors declare no conflicts of interest.

REFERENCES

[1] H. Zhang and F. Gui, “The application and research of new digital technology in marine aquaculture,” *J. Mar. Sci. Eng.*, vol. 11, no. 2, p. 401, Feb. 2023.

[2] S. A. Jasmin, P. Ramesh, and M. Tanveer, “An intelligent framework for prediction and forecasting of dissolved oxygen level and biofloc amount in a shrimp culture system using machine learning techniques,” *Expert Syst. Appl.*, vol. 199, p. 117160, Aug. 2022.

[3] A. Steinkruger et al., “Seafood traceability program design: Examination of the United States’ Seafood Import Monitoring Program,” *Ambio*, vol. 54, no. 2, pp. 168-174, Feb. 2025, https://doi.org/10.1007/s13280-024-02075-8

[4] L. C. Navarro, A. Azevedo, A. Matos, A. Rocha, and R. Ozório, “Predicting weight dispersion in seabass aquaculture using discrete event system simulation and machine learning modeling,” *Aquac. Rep.*, vol. 38, p. 102315, Oct. 2024.

[5] L. Chen, X. Yang, C. Sun, Y. Wang, D. Xu, and C. Zhou, “Feed intake prediction model for group fish using the MEA-BP neural network in intensive aquaculture,” *Inf. Process. Agric.*, vol. 7, no. 2, pp. 261-271, Jun. 2020.

[6] N. Chirdchoo, S. Mukviboonchai, and W. Cheunta, “A deep learning model for estimating body weight of live Pacific white shrimp in a clay pond shrimp aquaculture,” *Intell. Syst. Appl.*, vol. 24, p. 200434, Dec. 2024.

[7] X. Chen, I. N’ Doye, F. Aljehani, and T. M. Laleg-Kirati, “Fish weight prediction using empirical and data-driven models in aquaculture systems,” in *Proc. 2024 IEEE Conf. Control Technol. Appl. (CCTA)*, 2024, pp. 369-374.

[8] Y. Jin, L. Meng, and T. Shi, “An effective feature learning approach using genetic programming for crab age classification,” *Fish. Res.*, vol. 281, p. 107197, Jan. 2025.

[9] J. H. Qu, J. H. Cheng, D. W. Sun, H. Pu, Q. J. Wang, and J. Ma, “Discrimination of shelled shrimp (Metapenaeus ensis) among fresh, frozen-thawed and cold-stored by hyperspectral imaging technique,” *LWT-Food Sci. Technol.*, vol. 62, no. 1, pp. 202-209, Jun. 2015.

[10] Z. Liu, M. Huang, Q. Zhu, J. Qin, and M. S. Kim, “Nondestructive freshness evaluation of intact prawns (Fenneropenaeus Chinensis) using line-scan spatially offset Raman spectroscopy,” *Food Control*, vol. 126, p. 108054, Aug. 2021.

[11] T. Valeeprakhon, K. Orkphol, and P. Chaihuadjaroen, “Deep convolutional neural networks based on VGG-16 transfer learning for abnormalities peeled shrimp classification,” *Int. Sci. J. Eng. Technol. (ISJET)*, vol. 6, no. 2, pp. 13-23, Dec. 2022.

[12] D. A. Willette, K. Andrade, B. Fitzpatrick, and K. Wilson, “Outreach & DNA-based monitoring facilitate 3-fold reduction in seafood mislabeling in Los Angeles over 10 years,” *Food Control*, vol. 168, p. 110913, Feb. 2024.

[13] L. Peruzza et al., “Preventing illegal seafood trade using machine-learning assisted microbiome analysis,” *BMC Biol.*, vol. 22, no. 1, p. 202, Sep. 2024, https://doi.org/10.1186/s12915-024-02005-w

[14] E. Basdeki and T. N. Tsironi, “Novel packaging technologies for shrimp and shrimp products,” in *Postharvest Technologies and Quality Control of Shrimp, Amsterdam*, NL: Elsevier, 2025, pp. 295-321.

[15] S. de la Puente and V. Christensen, “Linking catch reconstructions with downstream supply chain nodes can help strengthen management actions in favour of just, sustainable and resilient futures,” *Mar. Policy*, vol. 170, p. 106387, Sep. 2024.

[16] L. Lorusso et al., “Mismanagement and poor transparency in the European processed seafood supply revealed by DNA metabarcoding,” *Food Res. Int.*, vol. 194, p. 114901, Oct. 2024.

[17] W. Woraprayote, C. Kongsawat, and W. Visessanguan, “Safety and regulation of different food additives for shrimp and shrimp products: Comprehensive analysis in Thailand,” in *Postharvest Technologies and Quality Control of Shrimp*, Amsterdam, NL: Elsevier, 2025, ch. 12, pp. 323-346.

[18] M. Fox, M. Mitchell, M. Dean, C. Elliott, and K. Campbell, “The seafood supply chain from a fraudulent perspective,” *Food Secur.*, vol. 10, no. 4, pp. 939-963, Aug. 2018, https://doi.org/10.1007/s12571-018-0826-z

[19] M. C. Rivers, A. B. Campbell, C. H. Lee, P. Kapoor, and R. S. Hellberg, “Short-weighting, species authentication, and labeling compliance of prepackaged frozen shrimp sold in grocery stores in Southern California,” *Food Control*, vol. 155, p. 110101, Jan. 2024.

[20] P. Ganapathiraju, T. J. Pitcher, and G. Mantha, “Estimates of illegal and unreported seafood imports to Japan,” *Mar. Policy*, vol. 108, p. 103439, Oct. 2019.

[21] I. O. Owolabi and J. A. Olayinka, “Incidence of fraud and adulterations in ASEAN food/feed exports: A 20-year analysis of RASFF’s notifications,” *PLoS One*, vol. 16, no. 11, p. e0259298, Nov. 2021.

[22] M. O. Balaban, S. Yeralan, and Y. Bergmann, “Determination of count and uniformity ratio of shrimp by machine vision,” *J. Aquat. Food Prod. Technol.*, vol. 3, no. 3, pp. 43-58, May. 1995.

[23] D. A. Luzuriaga, M. O. Balaban, and S. Yeralan, “Analysis of visual quality attributes of white shrimp by machine vision,” *J. Food Sci.*, vol. 62, no. 1, pp. 113-118, Jan. 1997, https://doi.org/10.1111/j.13652621.1997.tb04379.x

[24] P. Pan, J. Li, G. Lv, H. Yang, S. Zhu, and J. Lou, “Prediction of shelled shrimp weight by machine vision,” *J. Zhejiang Univ. Sci. B*, vol. 10, pp. 589-594, Aug. 2009.

[25] H. Mubarak, Z. Zainuddin, and M. Niswar, “Pixel-based weight estimation of Vannamei shrimp using digital image processing: A solution for precise feeding management in aquaculture,” in *Proc. 2023 Int. Seminar Intell. Technol. Appl. (ISITIA)*, 2023, pp. 115-118.

[26] M. Spreitzenbarth, T. Schreck, F. Echtler, D. Arp, and J. Hoffmann, “Mobile-sandbox: Combining static and dynamic analysis with machine-learning techniques,” *Int. J. Inf. Secur.*, vol. 14, no. 2, pp. 141-153, Apr. 2015, https://doi.org/10.1007/s10207-014-0250-0

[27] M. Lathkar, “Getting started with FastAPI,” in *High-Performance Web Apps with FastAPI, Berkeley*, CA: Apress, 2023, pp. 29-64. [Online]. Available: https://doi.org/10.1007/978-1-4842-9178-8_2

[28] Z. Chen et al., “An empirical study on deployment faults of deep learning-based mobile applications,” in *Proc. 2021 IEEE/ACM 43rd Int. Conf. Softw. Eng. (ICSE)*, 2021, pp. 674-685.

[29] S. Patil et al., “Enhancing optical character recognition on images with mixed text using semantic segmentation,” *J. Sens. Actuator Netw.*, vol. 11, no. 4, p. 63, Oct. 2022.

[30] A. Spreafico, F. Chiabrando, L. Teppati Losè, and F. G. Tonolo, “The iPad Pro built-in LiDAR sensor: 3D rapid mapping tests and quality assessment,” *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. 43, pp. 63-69, Jun. 2021.

[31] K. Chen et al., “MMDetection: Open MMLab detection toolbox and benchmark,” *arXiv*, 2019. [Online]. Available: https://arxiv.org/abs/1906.07155 [Accessed: Dec. 15, 2024].

[32] X. Yu, J. Wang, S. Wen, J. Yang, and F. Zhang, “A deep learning based feature extraction method on hyperspectral images for nondestructive prediction of TVB-N content in Pacific white shrimp (Litopenaeus Vannamei),” *Biosyst. Eng.*, vol. 178, pp. 244-255, Feb. 2019.

[33] M. Sun, X. Yang, and Y. Xie, “Deep learning in aquaculture: A review,” *J. Comput.*, vol. 31, no. 1, pp. 294-319, Jan. 2020, https://doi.org/10.3966/199115992020023101028

[34] W. K. Mutlag, S. K. Ali, Z. M. Aydam, and B. H. Taher, “Feature extraction methods: A review,” *J. Phys. Conf. Ser.*, vol. 1591, no. 1, p. 012028, Jul. 2020, https://doi.org/10.1088/1742-6596/1591/1/012028

[35] Z. Liu, X. Jia, and X. Xu, "Study of shrimp recognition methods using smart networks," *Comput. Electron. Agric.*, vol. 165, p. 104926, 2019.

[36] A. Saleh, M. M. Hasan, H. W. Raadsma, M. S. Khatkar, D. R. Jerry, and M. R. Azghadi, "Prawn morphometrics and weight estimation from images using deep learning for landmark localization," *Aquac. Eng.*, vol. 106, p. 102391, Oct. 2024.

[37] I. D. Mienye and Y. Sun, "A survey of ensemble learning: Concepts, algorithms, applications, and prospects," *IEEE Access*, vol. 10, pp. 99129-99149, Oct. 2022.

[38] D. E. P. Moghaddam, A. Muguli, M. Razavi, and B. Aazhang, "A graph-based cardiac arrhythmia classification methodology using one-lead ECG recordings," *Intell. Syst. Appl.*, vol. 22, p. 200385, Jun. 2024.

[39] T. Khater, H. Tawfik, and B. Singh, "Explainable Artificial Intelligence for investigating the effect of lifestyle factors on obesity," *Intell. Syst. Appl.*, vol. 23, p. 200427, Sep. 2024.

[40] X. Yang, Y. Wang, W. Yan, and J. Li, "Variance estimation based on blocked 3×2 cross-validation in high-dimensional linear regression," *J. Appl. Stat.*, vol. 48, no. 11, pp. 1934-1947, Aug. 2021, https://doi.org/10.1080/02664763.2020.1780571

[41] Y. Fujikoshi and T. Sakurai, "High-dimensional consistencies of KOO methods for the selection of variables in multivariate linear regression models with covariance structures," *Mathematics*, vol. 11, no. 3, p. 671, Jan. 2023.

[42] J. H. Kim, "Multicollinearity and misleading statistical results," *Korean J. Anesthesiol.*, vol. 72, no. 6, pp. 558-569, Jul. 2019.

Kanjana Eiamsaard is an Assistant Professor in the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University Sriracha Campus, Thailand. She received an M.S. in Software Engineering from Chulalongkorn University, Thailand. Her research interests include Artificial Intelligence, machine learning, and software engineering.

Kathawach Satianpakiranakorn is a lecturer in the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University Sriracha Campus, Thailand. He received M.Eng. in Computer Applied Technology from Harbin Engineering University, P.R. Harbin, China. His research field includes Artificial Intelligence, Computer Vision, and Digital Image Processing.

Anan Banharnsakun is an Associate Professor in the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University Sriracha Campus, Thailand. He received a Ph.D. degree in Electrical and Computer Engineering from King Mongkut's University of Technology Thonburi (KMUTT),Thailand. His research interests include biologically inspired computational intelligence, robust optimization, pattern recognition, and high-performance computing.

Jirawat Jitprasutwit is a lecturer in the Department of Computer Engineering, Faculty of Engineering at Sriracha, Kasetsart University Sriracha Campus, Thailand. He received an M.Eng. in Electrical Engineering from Mahidol University, Thailand His research interests include Artificial Intelligence, Internet of Things, embedded systems , and smart intelligent systems.