ICIC Express Letters
Part B: Applications
Volume **14**, Number **1**, January **2023**

ICIC International ©2023 ISSN 2185-2766
pp. **65–72**

CATTLE FACE DETECTION WITH EAR TAGS USING YOLOV5 MODEL

Wai Hnin Eaindrar Mg[1] and Thi Thi Zin[2,*]

[1]Interdisciplinary Graduate School of Agriculture and Engineering
[2]Graduate School of Engineering
University of Miyazaki
1-1, Gakuen Kibanadai-Nishi, Miyazaki 889-2192, Japan
nc22003@student.miyazaki-u.ac.jp; *Corresponding author: thithi@cc.miyazaki-u.ac.jp

Received May 2022; accepted July 2022

Abstract. *With the increasing population's need for meat and requirements for high food quality, the livestock industry is developing from small-scale and subsistence farming towards intensive and specialized grazing. Cattle monitoring and management system is crucial to be registered for breeding association, food quality tracing, disease prevention and control and fake insurance claims. This research presents cattle face detection with their ear tags' names by applying light-weight YOLOv5 (You Only Look Once) model. This research is intent to the farmers who can not only monitor and manage the cattle conditions at the farm. The proposed system was trained to get the best accuracy model. The accuracy of the proposed model achieves up to 99.4% for four surveillance cameras.*
Keywords: Object detection, YOLOv5, Face detection, Ear tags, Cattle face detection

1. **Introduction.** Cattle face detection is a particular application of object detection that accurately finds the target face and its location in images [3]. Object detection is currently a very active research field in computer vision that facilitates high-level tasks such as automatic individual identification and intelligent image recognition. At present, individual cattle's identification methods can be divided into contact based identification technology and non-contact based identification technology. The contact cow identification method requires external tools to leave permanent marks on the cow's body or wear identity information devices. This type of identification methods is harmful and causes irreparable damage to cows. It is not only time-consuming and laborious but also having a poor recognition effect [4]. The contact identification method includes permanent identification method, temporary identification method, and electronic label method. Permanent identification methods include ear prints, hot-iron branding, and freeze marking. This type of methods directly causes irreversible damage to individual cows. The temporary identification method is an ear tag identification technology, which needs to be read manually. The ear tag is easy to fail due to damage, stain, falling off, and loss. The piercing ear tags will cause physical damage to the cow's body, and improper installation will even tear the eardrum. The identification method is the same as the biological characteristics used when the animal performs individual identification, without direct contact with cows, which is not easy to make cows uneasy [5]. After solving the accuracy problem of ear tags identification in a complex environment, cattle ear tags identification is expected to become the mainstream identification technology in the market.

To be able to build a real-time cattle monitoring and management system, it is necessary to implement good detection, recognition, identification and tracking methods. The present work relies on a YOLOv5 (You Only Look Once) deep neural network, to implement the cattle face detection with their ear tags' names. To properly train the YOLOv5

DOI: 10.24507/icicelb.14.01.65

model, it is essential to have a good dataset. The larger the dataset, the higher the chances of training the network to have good performance.

YOLOv5 is popular as a single-stage object detector [2] known for its performance and speed with a clear and flexible structure that can be broken down, adjusted and built on a very widely accessible platform. Many of the systems apply this architecture and attempt to optimize it; however, they mainly rely on adjusting specific parameters or augmenting their training set to improve performance, without much consideration for structural changes. The details of the proposed architecture system are presented in Section 2 and the experimental results are shown in Section 3. In Section 4, we conclude our paper and discuss the future work.

2. Proposed Architecture and Methodology.

In this section, the detailed explanations of the flowchart of the model, dataset preparation, dataset annotation and model training procedure for the proposed system are presented.

2.1. YOLO deep neural network.

YOLO is one of the most popular deep convolution neural models for object detection, due to its good performance and short time requirements. YOLOv5 is based on the PyTorch framework [1]. The present implementation uses YOLOv5s, which is the smallest model, and YOLOv5m, which is the next model in size. The other models available are YOLOv5l and YOLOv5x, the latter being the largest of all [7]. As the network size increases, its performance may also increase, at the cost of additional processing times. Therefore, the larger models may only be useful for complex problems where large datasets are available [8].

There are many other deep neural networks that can be used to detect objects. One of them is the mask-RCNN (recurrent neural network), which aims to solve the instance segmentation problem in machine learning or computer vision [6]. The mask-RCNN is therefore, in theory, more precise, at the cost of additional processing time. In a literature study, for the task of detecting multi cows in a farm, both YOLO and mask R-CNN with pre-trained weights show good precision and recall. Because it is a good and faster detector, with high levels of performance, it was decided to choose the YOLOv5 network for the present project [2]. However, the YOLO speed is a great advantage for real-time operation of the cattle monitoring.

YOLO was the object detector of predicting bounding boxes with class labels [9]. It can divide images into a grid system and each cell in the grid is responsible for detecting

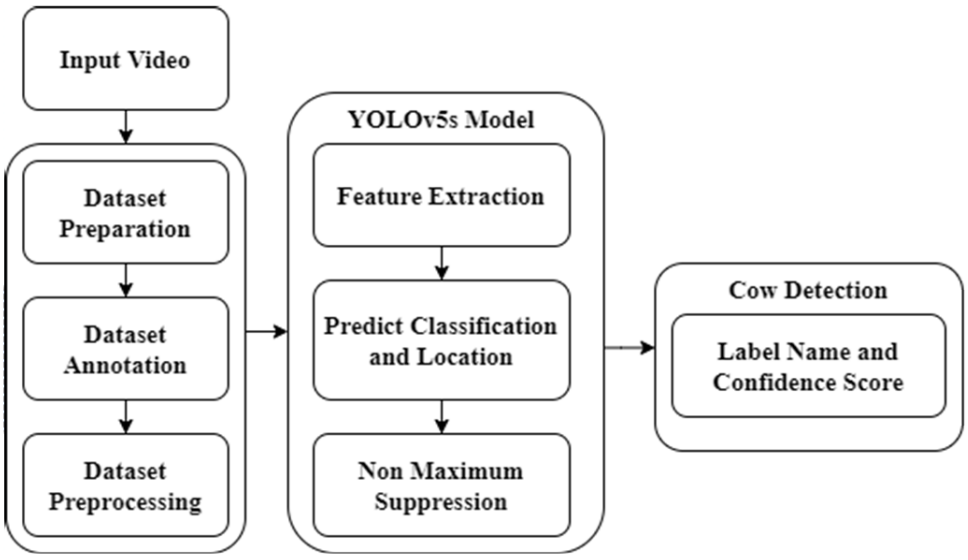


FIGURE 1. Flowchart of cattle detection with ear tags using YOLOv5s model

objects. Firstly, this model is designed to create features from input images and then to feed these features through a prediction system to draw boxes around the objects and predict their classes. The results of estimated class label and cattle face location are conducted to non-maximum value suppression to give the prediction result [10]. So, it can predict categories, generate bounding boxes, and their confidence score.

2.2. **Dataset preparation.** The datasets used in this study are from a large-scale dairy farm in Obihiro City, Hokkaido Prefecture, Japan. The dataset was captured by the surveillance camera in the base, with a resolution of (1920 × 1080) pixels, which was located on the frontal view of the cattle farm. Frames were extracted based on 2 images per 1 second intervals. Most of the images in the dataset are clear, though some cows in a few images were blurry when static images were captured due to movement. In this study, such images were also added to the dataset in order to increase its robustness. The dataset marked the position of each cow whose posture could be clearly judged, as well as the positions of any cows visible at the edges of the frame.

2.3. **Dataset annotation.** Makesenseai is the annotation tool that was used to label the ground truth for cattle faces using RectBox for training datasets. For labeling, the region of every cattle face was selected and annotated using the RectBox in the image. Then, the class label named cattle face needed to be marked on the bubble pop up on the screen. For the present research, the eighteen numbers of cows were chosen as targets. The total 5 cows are included in Cam1. Afterwards, we have to rename the class label according to their ear tags' names. For Cam2, there also include total 5 cows. After that, we have to rename the class label according to their ear tags' names just like the same process as video1. For Cam3 and Cam4, the total 4 cows are included and continue to rename the class label according to their ear tags' names.

2.4. **Dataset preprocessing.** For preprocessing stage, the experiment was tested under various scenes such as different illumination, overlapping, occlusion variation and postures changes without human intervention. In this system, blurry images were also added to the dataset.

This work aims to facilitate the detection of cattle face by using surveillance cameras, and it is common to collect the images where the cattle faces occupy large areas. The experimental dataset class labels translation is shown in Table 1. The total 688 images are prepared for this research. As can be seen in Table 2, the experimental dataset consists of two parts, training (80%), and verification (20%), which contains 548, and 140 images, respectively.

Table 1. The experimental dataset class labels translation

Videos	**#images**	**#cows**	**Class labels (ear tags)**
Cam1	172	5	3197, 3197, 0661, 1896, 0693
Cam2	172	5	8644, 7231, 1799, 0782, 1249
Cam3	172	4	1361, 8558, 1859, 1928
Cam4	172	4	0564, 2130, 1757, 1512
Total	688	18	18

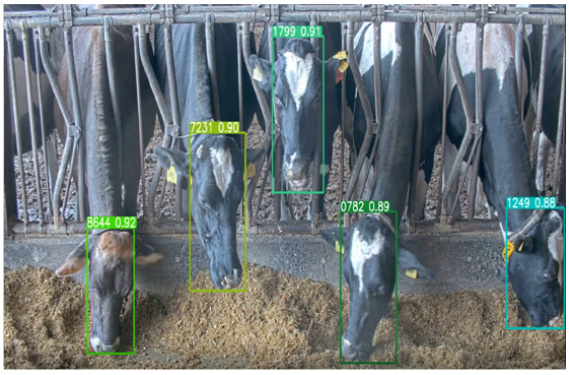
Table 2. The experimental dataset specifications

Dataset	**Images**	**Image size**	**Proportion**
Train	548	(640 × 640)	80%
Validation	140	(640 × 640)	20%
Total	688	(640 × 640)	100%

2.5. **Model training.** This system used the YOLOv5 network to detect multi-cow images according to their ear tags as quickly as possible. This system tested YOLOv5 on the constructed dataset. YOLOv5s showed the fastest speed and relatively high accuracy, so it has been chosen as the model for this step. The multi-angle multi-cattle image size was adjusted to (640×640) through adaptive image scaling, and it was sent to the network. The original image ($640 \times 640 \times 3$) was input into the newly added focus structure; first, it was turned into a feature map of ($320 \times 320 \times 12$) by using the slicing operation, and then, it became a feature map of ($320 \times 320 \times 32$) after a convolution operation with 32 convolution kernels. After the input into the structure of feature pyramid networks (FPN) combined with the path aggregation network (PAN), GIoU_Loss was used as the loss function of the bounding box frame to perform the final target detection.

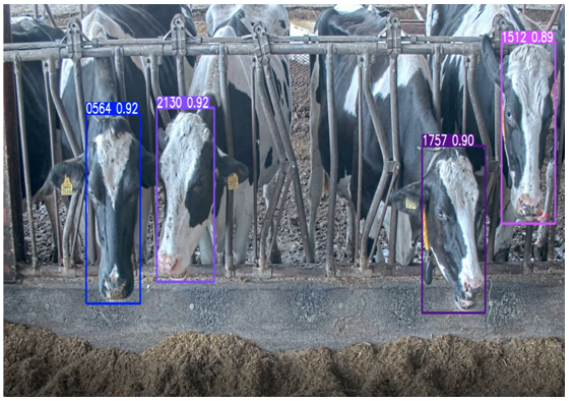
The training process was completed by running the model through Google Colab GPU. YOLOv5s is significantly better than YOLOv5x in terms of performance and speed. The mean average precision (mAP) is quite similar for both the processes, but when the processing speed is considered, YOLOv5s is slightly superior to YOLOv5x [6]. This system uses the pre-trained YOLOv5x model to determine the initial weights from which it started the training. The rest of the configuration settings are mostly preset: 50 epochs, 640 px image size for training including test set and batch size of 16. Each iteration took about 44.2 seconds in training. We trained our model up to 50 iterations and average loss was found to be 0.32 for using the batch size of 16. When the class labels score is less than 25%, the threshold will eliminate. This framework extracted features for face detection, and conducted final classification.

(a) Face detection for Cam1



(b) Face detection for Cam2

(c) Face detection for Cam3



(d) Face detection for Cam4

FIGURE 2. Cattle face detection results for all Cams with YOLOv5

3. **Experimental Results.** To evaluate the performance of an object detector, it is crucial to use appropriate metrics for each problem. Object detection is a very challenging problem because it is necessary to draw a bounding box around each detected object in the image. To evaluate the detection performance, some of the most common metrics are shown in Equations (1) to (3): precision, recall, and mAP.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{1}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2}$$

$$\text{mAP} = \frac{1}{N} \sum_{N}^{i=1} AP_i \tag{3}$$

True positive (TP) is a correct detection of an object that actually exists in the picture. False positive (FP) is an incorrect detection of an object, i.e., the network marks an object that is not there in the picture. False negative (FN) is an object that actually exists in the picture but is not detected by the network. In object detection, the intersection over union (IoU) measures the overlap area between the predicted bounding box and the ground truth bounding box of the actual object. Comparing the IoU with a given threshold, detection can be classified as correct or incorrect. Each value of the IoU threshold provides a different average precision (AP) metric, so it is necessary to specify this value. Table 3 shows the results of those metrics for all classes, obtained on the dataset with model YOLOv5s. The table shows the performance for each of the eighteen classes and for the whole validation set. The third column shows the number of known targets to be detected. The fourth and fifth columns show the precision and recall of the detector. The sixth and seventh columns show the mean average precision for the IoU specified. The proposed system performs 99.4% accuracy with 30 frames per second (fps). For mean average precision over different IoU thresholds, from 0.5 to 0.95, each class is above 80% accuracy.

Table 3. Performance of the 18 categories dataset (688 images)

Class labels	**Images**	**#detected targets**	**Precision**	**Recall**	**mAP@0.5**	**mAP@0.5:0.95**
all	140	629	0.994	0.998	0.994	0.85
3197	140	34	0.969	1	0.995	0.856
1007	140	35	0.997	1	0.995	0.846
0661	140	35	0.997	1	0.995	0.837
1896	140	35	0.996	1	0.995	0.812
0693	140	35	0.999	1	0.995	0.796
8644	140	35	1	1	0.995	0.837
7231	140	35	0.969	1	0.994	0.851
1799	140	35	0.995	0.971	0.995	0.84
0782	140	35	0.996	1	0.995	0.878
1249	140	35	0.998	1	0.995	0.757
1361	140	35	0.998	1	0.995	0.899
8558	140	35	0.997	1	0.995	0.839
1859	140	35	0.997	1	0.995	0.861
1928	140	35	0.997	1	0.995	0.874
0564	140	35	0.997	1	0.995	0.896
2130	140	35	0.996	1	0.995	0.879
1757	140	35	0.997	1	0.995	0.871
1512	140	35	0.997	1	0.995	0.871

The precision and recall curves are used for evaluating the performance of binary classification algorithms. It is often used in situations where classes are heavily imbalanced. The plots show that the models are progressively learning through every epoch because the performance is increasing and 50 epochs are enough training, considering that the curves are stable at that point. The precision-recall curve is constructed by calculating and plotting the precision against the recall for a single classifier at a variety of thresholds. The threshold would be the predicted probability of an observation belonging to the positive class. In this system, if an observation is predicted to belong to the positive class at probability > 0.7, it is labeled as positive. The result is detected cattle face with their ear tags. A precision-recall curve helps to visualize how the choice of threshold affects classifier performance, and can even help to select the best threshold for a specific problem.

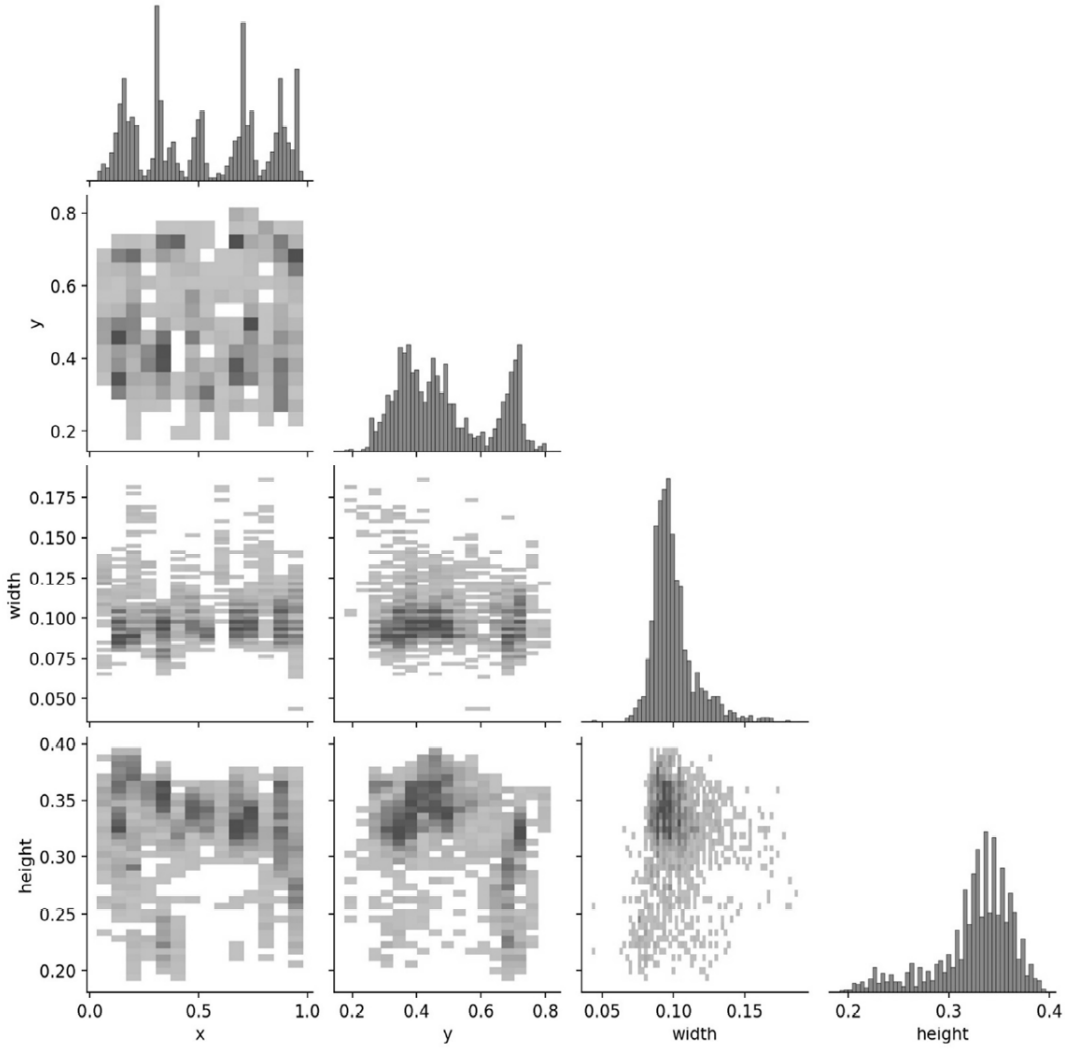


FIGURE 3. Graphical representation of labels correlogram

In this Figure 3 plot, correlation coefficient is colored with gray according to the value. Correlation matrix can be also reordered according to the degree of association between class labels.

The loss function shows the performance of a given predictor in classifying the input data points in a dataset. The smaller the loss is, the better the classifier is at modeling the relationship between the input data and the output targets. There are two different types of loss shown in Figure 4. The loss represented at the top is related to both the predicted bounding box and the loss related to the given cell containing an object during

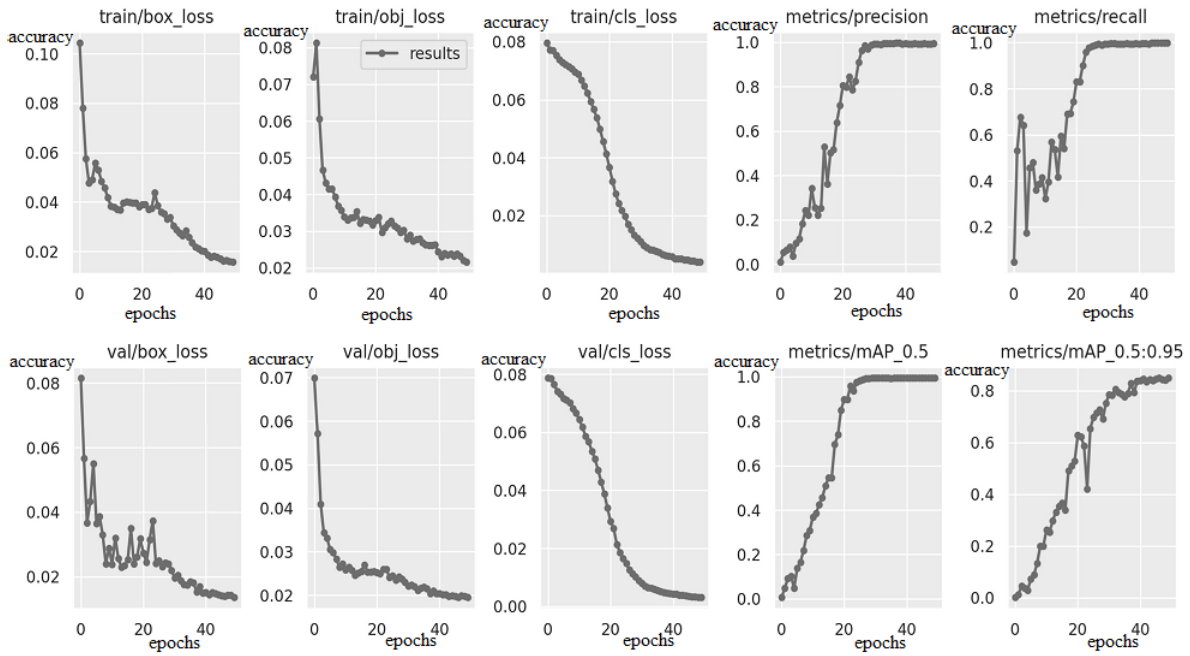


FIGURE 4. Loss during training, related to both the predicted bounding box and the loss related to the given cell containing an object, as well as their validation scores displayed as Box and Object

the training. The graphs of validation Box and validation Object represent their validation scores. Training loss is measured during each epoch while validation loss is measured after each epoch.

4. **Conclusions and Future Work.** The goal of the present work was to train a neural network for deep learning that can serve as a basis for recognizing cattle ear tags that helps professionals in the field of monitoring system. Relatively high identification result can be observed using the designed framework with the mAP of 99.4% but the best processing time of 8.52 minutes and execution time of 30 fps. This work only achieves the offline face identification in cattle farm. Future work includes a real-time system for cattle monitoring and management system using by identifying the cattle ear tags and examining the cattle behaviors, so that the farmer is guided through the procedures of the working order in real time. Moreover, future work will also concentrate on building an autonomous livestock individual identification system using facial features.

Acknowledgement. We gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga and A. Desmaison, PyTorch: An imperative style, high-performance deep learning library, *Advances in Neural Information Processing Systems*, vol.32, 2019.

[2] W. Xia, F. Yu, H. Wang and R. Hong, A high-precision lightweight smoke detection model based on SE attention mechanism, *2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pp.941-944, 2022.

[3] S. Kumar, S. Tiwari and S. K. Singh, Face recognition of cattle: Can it be done, *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, vol.86, no.2, pp.137-148, 2016.

[4] A. Malta, M. Mendes and T. Farinha, Augmented reality maintenance assistant using YOLOv5, *Applied Sciences*, vol.11, no.11, 4758, 2021.

[5] D. Snegireva and A. Perkova, Traffic sign recognition application using YOLOv5 architecture, *2021 International Russian Automation Conference (RusAutoCon)*, pp.1002-1007, 2021.

[6] Z. Li, X. Tian, X. Liu, Y. Liu and X. Shi, A two-stage industrial defect detection framework based on improved-YOLOv5 and optimized-Inception-ResnetV2 models, *Applied Sciences*, vol.12, no.2, 834, 2022.

[7] S. M. Noe, T. T. Zin, P. Tin and I. Kobayashi, Automatic detection and tracking of mounting behavior in cattle using a deep learning-based instance segmentation model, *International Journal of Innovative Computing, Information and Control*, vol.18, no.1, pp.211-220, 2022.

[8] U. Nepal and H. Eslamiat, Comparing YOLOv3, YOLOv4 and YOLOv5 for autonomous landing spot detection in faulty UAVs, *Sensors*, vol.22, no.2, 464, 2022.

[9] J. Ieamsaard, S. N. Charoensook and S. Yammen, Deep learning-based face mask detection using YOLOv5, *2021 9th International Electrical Engineering Congress (iEECON)*, pp.428-431, 2021.

[10] C. Li, R. Wang, J. Li and L. Fei, Face detection based on YOLOv3, *Recent Trends in Intelligent Computing, Communication and Devices*, pp.277-284, 2020.