An Integrated Approach to Mask Wearing Classification and Crowd Counting in Public Spaces Using YOLOv5s and Deep SORT

*Shyang-En Weng, Ying-Cheng Lin, Ming-Yao Liang, Shaou-Gang Miaou**

Department of Electronic Engineering, Chung Yuan Christian University, Taiwan
shyangen104@gmail.com, ycman911@gmail.com, x581014@kimo.com, miaou@cycu.edu.tw

Abstract

For safety and health reasons, we often need to monitor the flow of people in some public places. Object occlusion is a long-term, challenging problem for counting people based on the whole body of a pedestrian. In addition, counting approaches based on face object detection may perform poorly because facemasks obscure some important facial features. Thus, we propose to integrate face object detection based on mask wearing classification and people flow estimation. Through our research, we have identified YOLOv5s and Deep SORT as the optimal combination for this integration among various alternatives, and our system has been demonstrated to be effective across diverse population flow densities. Furthermore, we found out that using the face part as the tracking target performs better than using the whole body of a pedestrian for people flow estimation, especially in dense crowd cases. These findings make our approach highly feasible for real-world crowd monitoring applications, ensuring effective and reliable crowd control while considering safety and health measures.

Keywords: Mask wearing classification, People flow estimation, Deep learning neural networks, Object detection and tracking, Crowd monitoring

1 Introduction

On the evening of October 29, 2022, a crowd crush occurred during Halloween festivities near Itaewon, Seoul, South Korea, causing hundreds of deaths and injuries [1]. Although there are many factors contributing to the occurrence of this incident, the most important is the lack of effective crowd control. If there had been appropriate control at that time, this incident and other similar tragedies should have been avoided.

With the limited manpower and material resources of local governments, the cost and efficiency of such control are important considerations. The first step in efficient crowd control is crowd monitoring, because the control cost can be precisely adjusted according to the flow density. Crowd monitoring is not a new issue. In the past, we may have monitored the flow of people in some public places due to concerns such as air quality, safety, or comfort [2-3]. In recent years, people flow monitoring has added a new meaningful function: monitoring whether social distance is maintained to reduce the spread of epidemics [4].

At present, the mainstream of people flow monitoring technology is the visual image system [5]. One advantage of using image technology is that many existing surveillance cameras can be combined with the Internet to form a very effective IoT (Internet of Things) for people flow monitoring. People flow monitoring is highly dependent on people counting. Image-based people counting technology usually regards the entire pedestrian or at least the face part as the counting basis; the former is more suitable for situations where the shooting distance is far and the whole body is not occluded, and the latter is more suitable for situations where the torso is easily occluded. People counting approaches that used pure faces as object features (such as [6] and [7]) in the past must be modified because after years of epidemics, it is more and more common to encounter situations where people wear masks to cover part of the facial features. The most convenient way of making the modification is to derive the counting method of people flow based on the current techniques of mask wearing recognition, not the other way around. Based on this thinking, this study proposes a framework that integrates mask-wearing recognition and people counting.

In summary, the conventional people counting technique based on entire pedestrian body analysis is prone to body occlusion problems that undermine its effectiveness. Moreover, the emergence of epidemic prevention requirements has introduced a new challenge involving the automatic identification of mask-wearing within crowd control settings. Therefore, how to efficiently combine the face-based people counting method (to mitigate occlusion problems) with the additional mask-wearing recognition function has become the primary issue discussed in this study.

As shown in Figure 1, the proposed framework features a combination of general-purpose object detectors and people flow estimators. First, an object detection network is used to find the face position and perform mask-wearing classification for each pedestrian in the image sequences of the surveillance video. Then the resulting information is integrated into the object tracking technology to determine whether the pedestrian passes through a preset area, and the passers are counted to obtain the number of people

*Corresponding Author: Shaou-Gang Miaou; Email: miaou@cycu.edu.tw
DOI: https://doi.org/10.70003/160792642025072604001

wearing or not wearing masks and the total number of people passing. Compared with the traditional people counting method that only provides people flow data, this framework not only provides people flow data but also creates a value-added function: mask-wearing conditions such as wearing or not wearing a mask for special occasions that require this function. Furthermore, this framework is open in the sense that the object detection part and the people tracking and counting part can be implemented by any suitable method. In this study, we demonstrate the feasibility of the framework by choosing YOLOv5s and Deep SORT as the implementation methods of these two parts, respectively.

The main contributions of this study are as follows:

- Propose the first framework that combines mask wearing classification and people flow estimation.
- For people flow estimation, we show the advantage of using only the face as the tracking target instead of the whole body at an entrance or exit.
- Comparing various combinations of detection and people flow estimation, our analysis reveals that YOLOv5s and Deep SORT consistently demonstrate superior performance.
- The modular system design enables each functional block to be used individually or collectively according to different needs, which facilitates the development of any extended application.

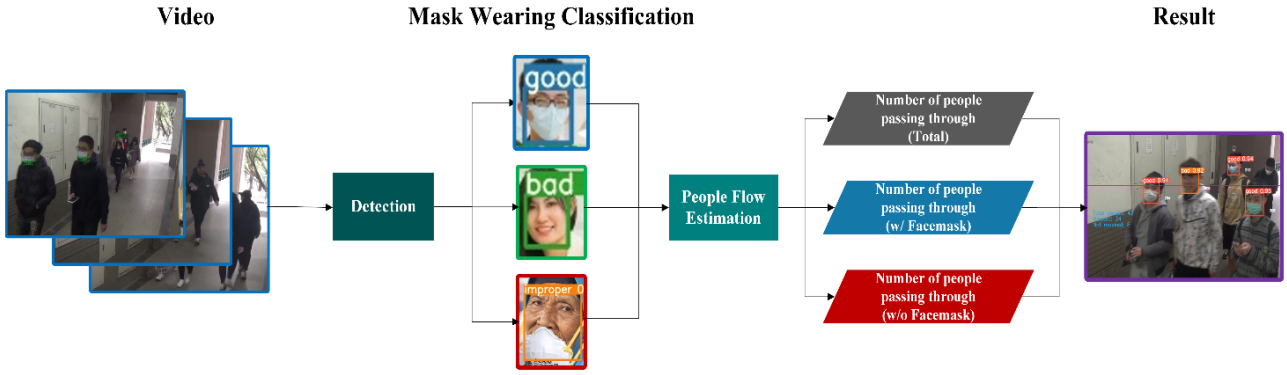


Figure. 1. An automatic detection system with both mask-wearing classification and people flow estimation functions (Initially, object detection on the input video extracts bounding box and mask class information. This data is then integrated into the people flow estimation subsystem to ascertain if targets enter a predetermined area, ultimately providing an overall count of passing individuals in that designated space.)

2 Related Works

In the following, we review the related works on face detection, masking wearing classification, people flow estimation, and some combinations of the above.

Mask wearing classification is mostly based on a face detection technique that detects the face location in an image, and face detection is highly dependent on facial features. However, in the case of wearing a mask, facial features are largely obscured and difficult to detect. It also increases the difficulty of automatically finding faces in an image. In 2017, Ge et al. [8] trained a deep convolutional neural network (CNN) for detecting occluded faces, and the network can simultaneously classify and identify real faces and output their accurate locations and scales in an image. In 2020, Li et al. [9] proposed a multi-angle head pose classification method that was verified by the MAFA (Masked Face) dataset they collected, achieving detection accuracy of 93.6% and 87.2% for frontal and profile faces, respectively. In 2022, Wang et al. [10] realized the classification of mask wearing through an object detection network and conducted a field test at a busy station exit, demonstrating its application potential in public places. In 2021, Yang et al. [11] used Mask R-CNN, a two-stage object detection model, to develop a system for facial mask detection. While they mentioned potential applications of crowd monitoring with their system, they did not suggest or provide any specific practices or experiments.

There is always a potential need to count people automatically at public places, such as tourist attractions, government units, stations, department stores, and amusement parks. Hara et al. [12] used R-CNN, an early two-stage object detection model, to find the head locations of pedestrians in an image. Then, given the moving speeds of vehicles and pedestrians, the trajectories of the detected bounding boxes are estimated based on location and color similarities. For the dash cam videos taken in an urban area of Osaka, the mean error rate of people counting for two-way flow was about ±13.6%.

In the estimation of people flow, the better we can monitor the action and position changes of all targets, the more accurately we can estimate the number of people passing, and multi-target tracking is an effective way to do so. Wang [13] conducted an in-depth investigation on the methods of face recognition and multi-target tracking and combined the two to achieve real-time and high-quality tracking results. In [14], Mohaghegh and Pang implemented a multifunctional system, including pedestrian sensing, identity discrimination, face detection, and people flow estimation. For people flow estimation, they proposed an object tracking model called a discriminative correlation filter to extract the ID information of each pedestrian and then determine whether the pedestrian with a particular ID

passes a counting baseline or not.

In [15], Militante and Dionisio trained a VGG-16 CNN to discriminate between wearing a mask and not wearing one with 96% accuracy, and the trained model was imported on edge computing devices, which demonstrates the feasibility of using deep learning for facemask detection on edge computing devices and facilitates the deployment of mask wearing classification systems in practice. Sethi et al. [16] combined the functions of pedestrian identity and mask-wearing recognition under the epidemic. In their approach, one-stage and two-stage object detectors for mask wearing recognition were integrated with a backbone neural network that performs face recognition for those who do not wear masks.

Given the ideas from the literature above, especially [10] and [13], this study proposes to conduct mask wearing classification and use the main byproduct of the classification task, i.e., face-bounding boxes, as the tracking target for people flow estimation. To the best of our knowledge, it is the first modular integrated framework that combines mask wearing classification and people flow estimation.

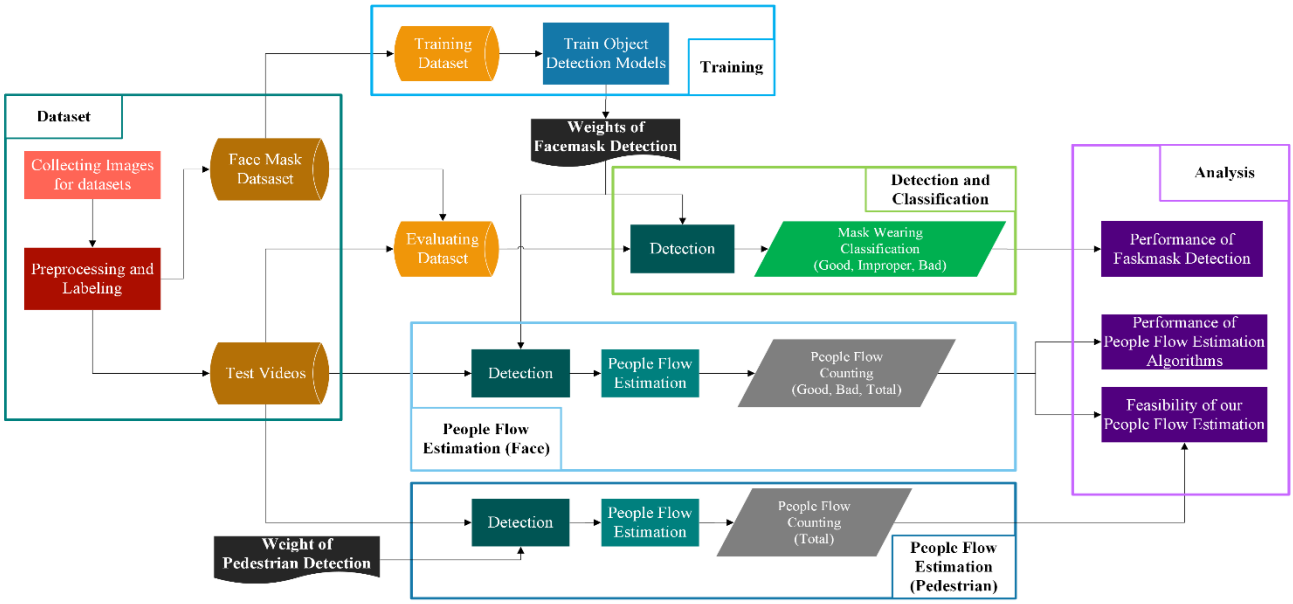


Figure 2. A flow chart of the experiment for the proposed approach
(**Dataset:** Collect, label, and pre-process mask images. **Training and Detection:** Train networks for mask classification and face detection. **People Flow Estimation:** Use face bounding boxes to determine pedestrian movement. **Analysis:** Evaluate facemask detection and people flow estimation methods for final system verification.)

3 Proposed Approach

The main objective of this study is to develop a system that can effectively perform object detection and tracking tasks. The two tasks of the system are highly dependent in a way that the overall system fails if one of the tasks fails to work properly. Therefore, we not only need a powerful people flow estimation method to count the number of people passing by, but we also need an object detection method with high enough recognition accuracy to deal with the large flow of people at the entrances and exits of public places.

To meet the design objective above, we developed an experiment process as shown in Figure 2, which includes five parts as follows:

(1) Collecting images and building datasets: this study attempts to identify three mask-wearing classes: Good (wearing a mask correctly), Improper (wearing a mask incorrectly, such as exposing the nose), and Bad (not wearing a mask). Therefore, we prepared the labeled images of these three mask-wearing classes and built the dataset for network training and verification. In addition, we took the video at the entrance and exit of public places as a test set to show the practicability of the proposed system.

(2) Training object detection models: building the model of deep learning object detection and setting up relevant experimental environments. Besides the YOLOv5s model chosen in this study, we also consider two more general-purpose deep learning object detection networks, namely SSD and YOLOv4, to see if the proposed framework is indeed open to any suitable method chosen.

(3) Object detection subsystem: for each neural network, the trained network weights are provided for the model to do face location finding and mask wearing classification.

(4) People flow estimation subsystem: count the number of people passing based on the classification result from an object detection network and the tracking result from an object tracking method. The chosen method of object tracking in this study is Deep SORT.

(5) Performance analysis: analyze the performance of mask wearing classification and people flow estimation for the proposed system and evaluate its feasibility in real-world applications.

More details on the proposed approach and the corresponding experiments will be given next.

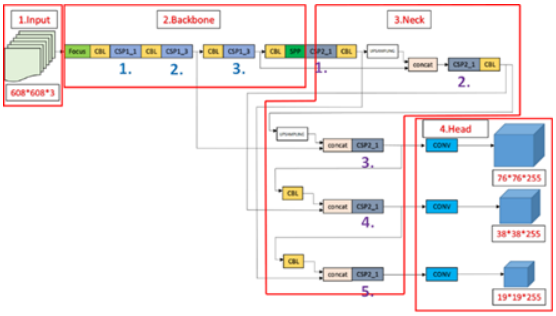


Figure. 3. The architecture of YOLOv5s [25]

3.1 Object Detection

Recently, CNN-based object detection networks, such as R-CNN [17], Faster R-CNN [18], SSD (Single Shot Multi-Box Detector) [19], and YOLO (You Only Look Once) [20], have drawn great attention. The first two belong to the two-stage object detector, which will first do the preliminary detection of objects and then do the classification and positioning. If there are too many objects detected initially, it will take a lot of time to classify and locate these objects. The latter two are one-stage object detectors, which will simultaneously do preliminary detection, classification, and localization. Generally speaking, the one-stage method has relatively high object detection efficiency, but the detection accuracy is slightly lower than that of the two-stage method.

The network output of YOLO [20] includes the position of the bounding box as well as the category and probability of each bounding box. The entire image is divided into several small images, and bounding box detection and sub-image classification are performed for each sub-image. The bounding boxes with low confidence would be discarded. The entire network is end-to-end, with the advantage of being easy to train and fast. YOLOv2 [21] optimized YOLO in 2017 through methods such as the new feature extractor DarkNet-19, anchor boxes, and batch normalization. In the following year, YOLOv3 [22] was proposed, using DarkNet-53 and citing the concept of FPN [23] to strengthen information between different scales to obtain higher precision and faster real-time object detection. At the same resolution and precision, it is three times faster than an SSD. In April 2020, YOLOv4 [24] was proposed based on a new CNN design called Cross-Stage Partial Network (CSPNet). CSPNet is a processing idea that can be incorporated into different backbone networks, such as ResNet, ResNeXt, and DenseNet, to reduce the computational cost significantly and maintain or even slightly improve the detection accuracy. Following YOLOv3, which uses DarkNet-53 as the backbone network, YOLOv4 adopts the backbone network called CSPDarknet53, which is the combination of CSPNet and DarkNet-53. Besides the change in network architecture mentioned above, YOLOv4 also incorporated some network training strategies for further performance improvement [24].

In June of the same year, a company released YOLOv5 [25] and claimed that: it outperforms the EfficientDet proposed by the Google Brain team; it has an extremely fast speed similar to YOLOv4; and a weight file size much smaller than that of YOLOv4. However, due to the lack of formal papers, it has aroused doubts in academic circles and social groups. Nevertheless, the results in [10] show that a version of YOLOv5 called YOLOv5s has good performance in mask wearing classification and thus was chosen for this study.

YOLOv5 presents two aspects of novel improvement: preprocessing and model structure, respectively. The preprocessing includes Mosaic augmentation with random scaling and auto-learning anchors for the bounding box prediction, which is fixed in YOLOv3 by the distribution from K-means. For the architecture, taking YOLOv5s as an example, as shown in Figure 3, the backbone of YOLOv5s improves the CSP-DarkNet with Focus structure and then connects the CSP block and Convolution block in series. The Neck part of YOLOv5s introduces SPPF (Spatial Pyramid Pooling Fusion), the optimized version of the neck of YOLOv3, updating its structure by using CSP-Block to replace the convolutional layer after concatenation, and removing the convolution block of the feature map output by FPN before concatenation to simplify the model and reduce the network size, enhance feature extraction and fusion capabilities, and reduce the computational cost of using a large number of convolutional layers. The head component follows the head of YOLOv3, using three distinct resolutions for prediction, aiming to combine multi-scale predictions for different viewpoints. Each prediction comprises coordinates, anchors, and class probabilities. These predictions are then consolidated through non-max suppression (NMS) to derive the ultimate outputs. In short, YOLOv5s is fast, light weight, and high precision. For a more comprehensive understanding and detailed explanation of the concepts discussed, we recommend consulting the source provided in [25].

3.2 People Flow Estimation and People Counting

Object tracking is a technique for finding the trajectory over time for each moving object with a unique ID. Specifically, the technique is used to determine whether the object in the current frame has appeared in the previous frame. If so, keep the same ID assignment; otherwise, the object will be assigned a new ID. There are two main trends in tracking technology today. One is to predict the position of the object in the next frame according to the momentum change, and the other is to analyze the similarity of the predicted pre-selected boxes. Object tracking algorithms can be divided into single-target tracking and multi-target tracking; the former focuses on correctly marking the same target at different times and is often used for cross-shot targets, and the latter focuses on detecting the position and motion of multiple targets

at the same time. We use multi-target tracking because the proposed system is to be deployed in crowded public places.

In addition, since we usually only control whether people entering a particular place wear masks and not people leaving, this study only counts the number of people entering. In other words, we are only interested in targets facing the camera. However, this unidirectional approach can easily be extended bi-directionally by setting up another camera in the opposite direction to capture images. As for the tracking target, we mainly consider the face part; we also consider the whole pedestrian, but only for comparison.

In this study, we will use Deep SORT (Simple Online Real-Time) [26] as the main method for estimating people flow since it is one of the powerful tracking techniques that consider both object momentum and similarity. Next, we discuss Deep SORT and describe how it can be used for people flow estimation.

People Counting Based on Multi-Object Tracking

Before explaining Deep SORT, we have to introduce SORT first because SORT is the prototype of Deep SORT.

SORT [27] uses a Kalman filter and a Hungarian algorithm to perform momentum prediction of a bounding box based on seven variables, including center point coordinates (u, v), area scale (s), and aspect ratio (r) of the bounding box, as well as the variation rates of u, v, and s, respectively, as shown in Eq. (1):

$$\mathbf{x} = [u, v, s, r, \dot{u}, \dot{v}, \dot{s}]^T \tag{1}$$

The next step is to compare the IOU (intersection over union) distance of the bounding box with the predicted position of the box on the current frame, and select the target with the same identity. This algorithm can track objects very efficiently. However, since it does not consider the occlusion problem that often occurs in many real-world applications, frequent ID switching makes it undesirable for long-term tracking counting.

Deep SORT [26] is one of the SOTA (state-of-the-art) online multi-object tracking algorithms. In addition to predicting with a Kalman filter like SORT, Deep SORT also uses a deep CNN to analyze the appearance association of appearance features. For architectural details of deep CNN, see [26]. The momentum change of the object is expressed by the Mahalanobis distance $d^{(1)}$ between the Kalman filter predicted and the current detection result, as shown in Eq. (2):

$$d^{(1)}(i, j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i) \tag{2}$$

where d_j represents the bounding box position of the j^{th} detected object, y_i represents the i^{th} position predicted by the momentum of the Kalman filter, and S represents the covariance matrix of the momentum prediction. A deep CNN is then used to obtain the minimum cosine distance $d^{(2)}$ between the object appearance and the object appearance in the gallery set, as shown in Eq. (3):

$$d^{(2)}(i, j) = \min\{1 - r_j^T r_k^{(i)} \,\big|\, r_k^{(i)} \in \mathcal{R}_i\} \tag{3}$$

Figure. 4. An illustration for facilitating the explanation of people counting based on multi-object tracking

where r_j represents the appearance descriptor of the j^{th} object, $\mathcal{R}_i$ denotes the gallery set of the i^{th} object, and $r_k^{(i)}$ denotes the k^{th} appearance descriptor of the ith object. Next, the association metric with tracking is used as a weighting factor to integrate momentum prediction and apparent features, as shown in Eq. (4):

$$c_{i,j} = \lambda d^{(1)}(i, j) + (1 - \lambda) d^{(2)}(i, j) \tag{4}$$

where λ denotes a hyperparameter that determines the proportion of associations between momentum predictions and the tracked apparent features. Finally, possible tracking targets are obtained through cascade trajectory matching based on the Hungarian algorithm. The key innovation of Deep SORT is the use of deep feature embeddings to match detected objects over time, which successfully alleviates the problems of large changes and mis-predictions caused by obvious mutations or partial occlusions and also solves the problem of ID switching determination in SORT.

As shown in Figure 4, a counting baseline is defined. The ID of each target in different frames can be obtained by an object tracking algorithm. Targets are counted after they pass the baseline, and targets with the same ID are counted only once. In this way, the exact number of times the baseline was passed can be found. The proposed counting procedure is briefly described as follows:

(1) Set a counting baseline, with which two separate zones are defined.
(2) Record the object ID that exists in the upper zone.
(3) If the center of a bounding box appears in the lower zone, check whether the ID associated with that bounding box has ever been recorded in the upper zone.
(4) If so, count it and record the mask-wearing class of the target.

3.3 Integrated System

Mask wearing classification and people flow estimation can be two independently designed tasks, but this study effectively combines the two to improve the overall

efficiency, which is one of the special features of this study. In this combination, the people flow estimation can also be regarded as an added value for mask wearing classification. We obtain the bounding box of the face and the mask-wearing category of the face through the object detection network. Given a bounding box, analyzing the positional change of the bounding box over time is a way to estimate the flow of people. In addition to object detection methods that can accurately predict bounding boxes for objects, robust object tracking algorithms are also required to have a perfectly integrated system. Traditionally, the whole-body bounding boxes of pedestrians are used as tracking targets, while we propose to use facial bounding boxes in this study. Therefore, we will compare the tracking performance of these two approaches. Specifically, we shall evaluate whether tracking with facial bounding boxes is of sufficient value for people flow estimation.

A pseudocode of our system is given below:

Given: MW dataset, Test Video,
Initialize: *Upper and Lower Zone*,
ID Set // FIFO (First in, first out) queue,
Passing Infos = empty Set,
$Threshold_{passing}$
// Training and Preparing Step
Detector Train YOLOv5s on MW dataset
Tracker Pretrained DeepSORT
// Inferencing Step
for each *frame* in Test Video **do**
BBox, Conf Detector (*frame*)
IDs Tracker (*BBox*)
IDs w/ Conf Match *IDs* with each *Conf* by *BBox*
for each *id* in *IDs w/ Conf* **do**
IF *id* in *Upper Zone* **do**
Update *ID Set* with *id*
IF *id* in *Upper Zone* && *id* in *Lower Zone* **do**
Update *Passing Infos* with *IDs w/ Conf*
IF length of *IDs* > $Threshold_{passing}$ **do**
Remove the top *IDs*
Ouput the *Passing Infos*
Summarize and Analyize the *Passing Infos*

4 Experiments

4.1 Research Equipment and Specifications

The computer hardware specifications used in this study are: CPU Intel i7-8700 3.2GHz, RAM 16GB, and GPU NVIDIA RTX 2080 8GB. The operating system is Windows 10 (64-bit). The software virtual environment is set to CUDA 10.2 and cuDNN v8.0.4, using Anaconda as the virtual environment manager. For deep learning frameworks, we use Tensorflow, Pytorch, and Darknet. The camera model is Sony HDR-PJ380, the image size of the video captured by the camera is 1920×1080, and the video playback rate is 30 frames per second (FPS).

4.2 Datasets

This study uses an open and freely available dataset [28] to train and validate object detection networks. For convenience, this dataset is referred to as mask-wearing (MW) in this study. The images in the MW dataset are labeled by the Digital Data Processing Sheltered Workshop belonging to the Eden Foundation in Taiwan.

(a) Good (*G*) (b) Improper (*I*) (c) Bad (*B*)

Figure. 5. Typical samples from the MW dataset [24]
Note: black mosaic patches are added artificially due to the consideration of personal privacy.

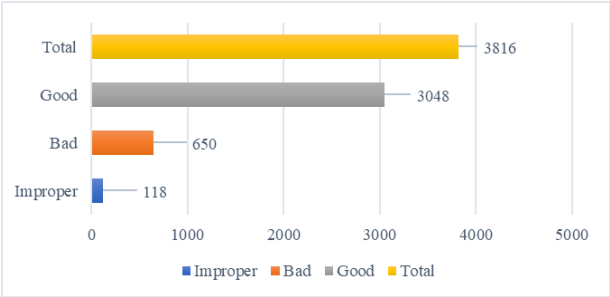


Figure. 6. Statistics of the data samples used for training the mask-wearing classifier in this study

Furthermore, we test object detection networks and people counting methods using videos recorded by ourselves at Chung Yuan Christian University (CYCU) in Taiwan and by the authors of [10] at the Zhongli Railway Station in Taiwan. For convenience, they are referred to as the CYCU dataset and the ZS dataset, respectively.

4.2.1 MW Dataset for Training Mask Wearing Classifier

The samples in the MW dataset were divided into three labeled categories, namely Good (*G*), Improper (*I*), and Bad (*B*), as shown in Figure 5.

After sample tagging, sorting, and selection, the statistical results of the data samples used in this study are shown in Figure 6. Only a very small fraction of images (approximately 2.7%) in the original MW dataset were discarded due to poor image quality. It clearly shows that the resulting datasets for each category are extremely unbalanced. To reduce this unbalanced impact on the performance evaluation of object detection, an unequal weighted metric for each category will be introduced and discussed in Section 4.3.

4.2.2 Test Videos

The CYCU dataset was produced in March 2021 at a gateway on the first floor of Chen Chih Hall in CYCU. The video was recorded around the beginning of two class sections at 12:10 and 13:10. Specifically, two videos were shot from 11:55 to 12:15 and from 12:55 to 13:15, when the crowd was relatively huge and the flow of people was mainly in an inbound direction into the hall. The ZS dataset contains two of the videos taken at the exit of Zhongli Railway Station in Taoyuan [10]. The recording time is at

2:00 pm in October 2020, and the weather is sunny. There will be a large-scale movement of people in the short term. Moreover, only movement of people in the same direction will occur. However, the system must deal with numerous pedestrian occlusions and high flow density.

Table 1. Confusion matrix for detection performance evaluation (Imp: Improper)

		Ground truth			
		Good	Imp	Bad	Precision
Prediction	Good	a	b	c	$\frac{a}{a+b+c}$
	Imp	d	e	f	$\frac{e}{d+e+f}$
	Bad	g	h	i	$\frac{i}{g+h+i}$
	Recall	$\frac{a}{a+d+g}$	$\frac{e}{b+e+h}$	$\frac{i}{c+f+i}$	

Both test datasets were recorded live at 30 FPS without any post-production. All videos were recorded before the Taiwan government enforced a level-3 alert for COVID-19. At that time, the control measures were relatively loose, and there was no mandatory requirement to wear a mask, so the videos can reflect various interesting situations such as wearing a mask correctly, not wearing a mask, and not wearing a mask properly. For object detection, 150 images in the datasets are randomly selected to evaluate the performance of mask-wearing classification. For object tracking, all videos in the dataset will be used to evaluate the performance of people-counting methods.

4.3 Evaluation Metrics

4.3.1 Object Detection Metrics

It is assumed in [10] that every face in an image can be detected since the same face will appear in multiple frames, and the probability of missing detecting all frames containing that face is negligible. In the case that every face in the image can be detected, four performance indicators of object detection are directly or indirectly derived from the confusion matrix shown in Table 1, including Accuracy, Precision, Recall, and F1-Score. As explained in Section 4.2.1, these metrics will be weighted and averaged to compensate for the impact of unbalanced dataset and ensure fairness for each mask-wearing category. Furthermore, here we remove the optimistic assumption given in [10] by treating the above performance metrics as conditional probabilities, where the condition or premise is that each face has been detected and displayed in a bounding box. So at the end, multiply by the a priori probability of a detected face to get a more realistic result. The estimate of this a priori probability, called the Face Detection Rate, will be presented and explained later. In addition to the above performance metrics, this study also considers object detection speed in terms of FPS.

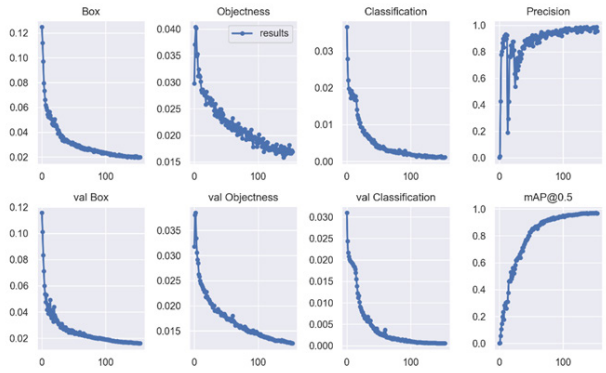


Figure 7. Training loss of YOLOv5s with the MW dataset (During each epoch of training and validation, three types of losses are observed: box regression loss, objectness loss, and classification loss. These losses exhibit a smooth decrease until convergence, indicating the model's progress. Additionally, the precision and mAP@0.5 curves demonstrate that the model achieves favorable results without overfitting.)

Accuracy is defined as the ratio of the total number of correct predictions on a mask-wearing class to the total number of predictions, as shown in Eq. (5):

$$\text{Accuracy} = \frac{a+e+i}{a+b+c+d+e+f+g+h+i} \tag{5}$$

Precision is defined as the ratio of the total number of correctly predicted cases in Class k of mask-wearing to that of all the predicting cases falling in Class k of mask-wearing, where $k \in \{G, I, B\}$, as shown in Eq. (6):

$$\begin{aligned} \text{Precision}_G &= \frac{a}{a+b+c}, \\ \text{Precision}_I &= \frac{e}{d+e+f}, \\ \text{Precision}_B &= \frac{i}{g+h+i} \end{aligned} \tag{6}$$

Recall is defined as the ratio of the total number of correctly predicted mask-wearing cases of Class k to that of all mask-wearing cases that actually belong to Class k, as shown in Eq. (7):

$$\begin{aligned} \text{Recall}_G &= \frac{a}{a+d+g}, \\ \text{Recall}_I &= \frac{e}{b+e+h}, \\ \text{Recall}_B &= \frac{i}{c+f+i} \end{aligned} \tag{7}$$

To avoid biased interpretation of the results due to dataset size unbalances for the mask-wearing category,

each equation in Eqs. (5) to (7) will be transformed into a weighted average (WA) form, where the weight w_k for Category k is proportional to the number of actual cases in that category to the total number of cases for all categories, resulting in the following WA forms:

$$\overline{\text{Accuracy}} = \frac{w_G \times a + w_I \times e + w_B \times i}{a+b+c+d+e+f+g+h+i} \tag{8}$$

$$\overline{\text{Precision}} = \sum\nolimits_{k\in\{G,I,B\}} w_k \text{Precision}_k \tag{9}$$

$$\overline{\text{Recall}} = \sum\nolimits_{k\in\{G,I,B\}} w_k \text{Recall}_k \tag{10}$$

Table 2. Performance comparison of the facemask detectors considered in this study (Best one in bold; second best underlined)

	Evaluation metrics	SSD @45 FPS	YOLOv4 @33 FPS	YOLOv5 @**62** FPS
Validation set	mAP	0.9079	**0.9969**	0.9920
Test videos @30 FPS	$\overline{\text{Accuracy}}$	68.32%	65.53%	**68.83%**
	$\overline{\text{Precision}}$	**99.45%**	97.92%	97.28%
	$\overline{\text{Recall}}$	**94.32%**	85.28%	84.41%
	Face detection rate	73.88%	**93.65%**	89.51%
	$\overline{\text{F}_1\text{-Score}}$	0.7153	**0.8537**	0.8091

F1-Score is a composite metric that combines precision and recall for each category. Following the same definition of F1-score, except precision and recall are replaced by corresponding weighted averages, $\overline{\text{F}_1\text{-Score}}$ is defined as

$$\overline{\text{F}_1\text{-Score}} = \frac{2\times\overline{\text{Precision}}\times\overline{\text{Recall}}}{\overline{\text{Precision}}+\overline{\text{Recall}}} \tag{11}$$

It can be treated as a weighted version of the F1-Score.

The Face Detection Rate is an estimate of the probability of successfully detecting a face in a video frame. This probability estimate is given by

$$\text{Face Detection Rate} = \frac{m}{m+\tilde{m}} \tag{12}$$

where m and $\tilde{m}$ denote the number of correctly recognized and undetected faces in a frame sequence, respectively. We multiply $\overline{\text{F}_1\text{-Score}}$ by the Face Detection Rate to obtain the overall performance for mask wearing classification in a more realistic manner for practical applications, as shown in Eq. (13):

$$\overline{\text{F}_1\text{-Score}} = \text{Face Dectection Rate} * \overline{\text{F}_1\text{-Score}} \tag{13}$$

The inference speed of object detection networks is measured in terms of the number of pictures processed by each neural network per second, or frames per second (FPS). The larger the FPS value, the faster the running speed; the smaller the FPS value, the slower the running speed.

4.3.2 People Flow Estimation Metrics

People flow estimators have different evaluation criteria, but they are all related to the difference between the estimated number $\hat{n}$ and the actual number n of people passing through a film shooting area within a fixed period. Referring to [12] and [14], we use Error Rate ε, to be defined, as the evaluation metric for the people flow estimator. This is the normalized error between the estimated number and the actual number of pedestrians passing in one direction, as shown in Eq. (14), where the number of people counted is further divided into three situations: wearing a facemask (f), not wearing a facemask ($\tilde{f}$), and the total number (T).

$$\varepsilon_c = \frac{\hat{n}_c - n_c}{n_c}\times 100\%,\ c\in\{f,\tilde{f},T\}. \tag{14}$$

A counting method with a lower error rate represents a closer approximation to actual flow, which is a better estimate of people's flow. In [14], the absolute value operation is implemented on the numerator of Eq. (14). However, considering that there are double counting and omissions in the estimation of people flow, the sign associated with Eq. (14) is retained in this study.

4.4 Analysis of Object Detection Results

4.4.1. Training Object Detection Networks

The training data is the MW dataset. The image size is set to 320×320. The batch size is 16, and the number of epochs is 500. The model type chosen for YOLOv5 is YOLOv5s because it has the fastest detection speed and the smallest number of parameters among all model types of YOLOv5. For YOLOv5s, the convergence curves obtained during the training phase are shown in Figure 7. For performance comparison, we also consider YOLOv4 and VGG-SSD, which are single-stage SOTA detectors that can run in real-time.

4.4.2 Performance of Object Detection Networks

Experimental results for the object detection networks considered in this study are summarized and shown in Table 2, followed by a discussion.

According to the results in Table 2, SSD has the best performance on $\overline{\text{Precision}}$ and $\overline{\text{Recall}}$, so it must have the best $\overline{\text{F}_1\text{-Score}}$, but because its Face Detection Rate is too low, the final overall indicator $\overline{\overline{\text{F}_1\text{-Score}}}$ is the worst. YOLOv5 has the best $\overline{\text{Accuracy}}$ and medium $\overline{\overline{\text{F}_1\text{-Score}}}$,

while YOLO4 has the best mAP, Face Detection Rate and $\overline{\overline{\text{F}_1\text{-Score}}}$. YOLO4 and YOLO5 perform similarly: the former performs better on Face Detection Rate, and the latter wins on $\overline{\text{Accuracy}}$. In terms of the overall performance metric $\overline{\overline{\text{F}_1\text{-Score}}}$, YOLOv4, YOLOv5, and SSD rank from the first to the third, respectively.

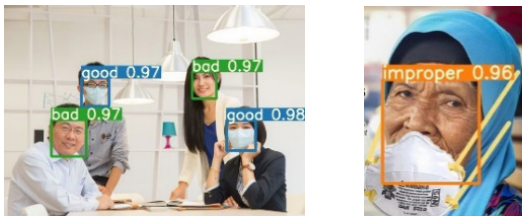
As shown in Table 2, the detection speeds of SSD, YOLOv4, and YOLOv5 are all higher than 30 FPS of the normal playback speed of the test video, indicating that they all have real-time processing capabilities. The actual processing speed is YOLOv5 > SSD > YOLOv4, which shows that YOLOv5 is more than enough for real-time implementation. Some typical detection results are shown in Figure 8 and Figure 9.

4.5 Results of the People Flow Estimation

Table 3 shows the results of people flow estimation based on three object detectors and the Deep SORT tracker. The best performance is shown in bold, and the next best performance is underlined. As can be seen from Table 3, no matter which data set is used, whether wearing a mask or not, YOLOv5s ranks first or second in error rate performance. The time required for the system to estimate people flow includes the object detection time, the time to track the movement of the bounding box containing the object, and the time to determine whether the bounding box entered a particular hot zone. So in terms of execution speed, people flow estimation will run much slower than pure object detection, but still mostly around 20 FPS. For YOLOv5s, the average FPS on both datasets corresponds to about 26.5 count updates per second, which should be enough for most applications. In terms of execution speed, YOLOv5s performs best on the CYCU dataset and ranks second on the ZS dataset. Considering the performance of Error Rate and execution speed, YOLOv5s and Deep SORT are the best combination.

Table 3. Performance comparison of people flow estimation systems

Test videos	Detector	People-counting estimator	Count (Error rate ε_f (w/ facemask)	Count (Error rate ε_f (w/o facemask)	Count (Error rate ε_f (total)	FPS
CYCU dataset (Sparse crowd)	SSD	Deep SORT	292 (-9.60%)	55 (-32.10%)	347 (-14.11%)	28.63
	YOLOv4		288 (-10.84%)	71 (**-12.35%**)	**359 (-11.14%)**	23.94
	YOLOv5 (Proposed)		**299 (-7.43%)**	57 (-29.63%)	357 (-11.63%)	**37.10**
	Ground Truth		323 (N/A)	81 (N/A)	404 (N/A)	N/A
ZS dataset (Dense crowd)	SSD	Deep SORT	242 (-28.19%)	10 (**-28.57%**)	252 (-28.21%)	**26.11**
	YOLOv4		288 (-14.54%)	6 (-57.14%)	294 (-16.24%)	14.85
	YOLOv5 (Proposed)		302 (**-10.39%**)	10 (**-28.57%**)	312 (**-11.11%**)	16.01
	Ground Truth		337 (N/A)	14 (N/A)	351 (N/A)	N/A



(a) Good and Bad (b) Improper

Figure 8. A typical detection result of YOLOv5 for the validation set of the MW dataset

(a) CYCU dataset (b) ZS dataset

Figure 9. A typical detection result of YOLOv5 for test videos

4.5.1 The Influence of Object Detection on People Flow Estimation

Since all the people flow estimation systems in this study are based on the results of object detection, their computational performance and counting performance are affected by object detection. Table 3 shows that object detection has a non-negligible impact on people-flow estimation. Specifically, under the same people counting estimator, the accuracy of object detection is negatively correlated with the counting error, indicating that the performance of object detection is positively correlated with the performance of the people flow estimation system; that is, the object detection capability is one of the important factors in estimating the overall performance of the people flow estimation system.

Most of the counting errors come from undetected bounding boxes or wrong detection of mask-wearing types, which also explains why most of the counting results are lower than the actual results. Interestingly, the execution speed increases when there are more undetected bounding boxes, which increases the estimation error of people flow. The reason should be that when the number of detected objects is small, the number of times to initiate the tracking procedure will naturally decrease. For example, SSD has the largest counting error in the ZS dataset, but its FPS performance is the best.

4.5.2 Practical Issues for People Flow Estimation

For practical reasons, the following also need to be considered:

a) Height of the target person: The target will be counted after passing the selected baseline. If the baseline or its associated hot zones are not set properly, the system may misjudge detection targets at different heights. Therefore, it is necessary to set an appropriate baseline or adjust the shooting angle to obtain better results.

b) Occlusion problem: When multiple people pass through the hot zone at the same time, some targets may be occluded, resulting in counting errors. Therefore, being able to set the camera at an angle or height that is less likely to cause targets to be occluded can significantly improve the accuracy of people's counting.

4.6 Performance Comparison of Integrated Systems

This section discusses the performance of people flow estimation for systems that target the whole body of pedestrians and systems that target only faces.

First, since the target of face detection is a human face, it naturally faces the same direction at the entrance or exit, and the target moving in the opposite direction will be ignored, reducing the interference from non-interesting directions. This is the advantage of face detection over pedestrian detection, as shown in Figure 10(a) and Figure 10(b). Secondly, the bounding boxes obtained by pedestrian detection may change the detectable area over time due to factors such as body occlusion, which will greatly affect the apparent discrimination ability of the tracking algorithm; using face detection can obviously avoid the occurrence of this situation, as shown in Figure 10(c) and Figure 10(d). Table 4 shows that face detection performs slightly better than pedestrian detection, although both count fewer people than actually pass by due to the influence of the people count estimator. To sum up, using the face as the detection target has certain reliability and robustness for this application scenario, and it is no worse than the traditional tracking approach that uses the whole pedestrian body as the detection target.

(a) (b) (c) (d)

(a) Moving targets in the opposite direction appear in pedestrian detection; (b) The same targets in (a) do not appear in face detection; (c) Large body occlusion for pedestrian detection; (d) No occlusion for face detection.

Figure 10. Pedestrian detection vs. face detection

Table 4. Comparison of people flow integrated systems

Method	Count (Error rate ε_f (total)	
	CYCU dataset	ZS dataset
Face det. + Count estimation	**357 (-11.63%)**	**312 (-11.11%)**
Pedestrian det. + Count estimation	355 (-13.37%)	513 (46.15%)
Ground truth	404	351

5 Conclusions and Future Work

5.1 Conclusions

By integrating object detection and tracking techniques, we develop an automatic recognition system for both mask-wearing classification and people count estimation. Specifically, the efficient object detection network model YOLOv5s and the SOTA tracking model Deep SORT are used for these two tasks, respectively. Experimental results show that the proposed system has good performance.

We compare the proposed YOLOv5s and two other SOTA object detection models in terms of accuracy and speed for mask-wearing classification performance. Furthermore, this study combines the results of object detection with the Deep SORT-based people count estimator to form a people flow estimation system for estimating people flow through an entrance. The experimental results show that YOLOv5 has the best

comprehensive performance in terms of mask-wearing classification and processing speed.

This study also compares the performance of traditional pedestrian detection and the face detection proposed in this study on tracking and people flow estimation, and the results show that face detection is better than pedestrian detection in terms of people counting accuracy. The system proposed in this study achieves a practical level in terms of execution speed and counting accuracy.

5.2 Future Work

The system architecture proposed in this study has powerful and diverse performance and modular options and can act as a multi-functional auxiliary tool according to the needs of users. Although the requirement to wear masks will be relaxed or even lifted in the future, the people flow estimation system based on face counting is still effective. In the future, we can consider porting computing platforms from general-purpose computers to edge computing devices to facilitate the use of portable devices and more convenient use in actual public places. Furthermore, when multiple systems are deployed in a larger area with multiple entrances, a cloud platform can be designed to integrate and organize the data from these systems and use mask-wearing information as the basis for analyzing the flow of people across the region, providing relevant real-time information to decision-makers.

References

[1] Seoul Halloween crowd crush, https://en.wikipedia.org/wiki/Seoul_Halloween_crowd_crush, retrieved on Feb. 8, 2023.

[2] Occupational Safety and Health Administration, U.S. Department of Labor, *Indoor Air Quality in Commercial and Institutional Buildings*, OSHA 3430-04, 2011.

[3] C. Martani, S. Stent, S. Acikgoz, K. Soga, D. Bain, Y. Jin, Pedestrian Monitoring Techniques for Crowd-Flow Prediction, *Proceedings of the Institution of Civil Engineers-Smart Infrastructure and Construction*, Vol. 170, No. 2, pp. 17-27, June, 2017. https://doi.org/10.1680/jsmic.17.00001

[4] Y. H. Chen, C. T. Fang, Combined Interventions to Suppress R0 and Border Quarantine to Contain COVID-19 in Taiwan, *Journal of the Formosan Medical Association*, Vol. 120, No. 2, pp. 903-905, February, 2021. https://doi.org/10.1016/j.jfma.2020.08.003

[5] G. S. Soares, R. C. Machado, R. A. Lotufo, People-Flow Counting Using Depth Images for Embedded Processing, *International Conference on Image Analysis and Recognition*, Montreal, QC, Canada, 2017, pp. 239-246. https://doi.org/10.1007/978-3-319-59876-5_27

[6] X. Zhao, E. Delleandrea, L. Chen, A People Counting System Based on Face Detection and Tracking in a Video, *IEEE International Conference on Advanced Video and Signal Based Surveillance*, Genova, Italy, 2009, pp. 67-72. https://doi.org/10.1109/AVSS.2009.45

[7] T. Y. Chen, C. H. Chen, D. J. Wang, Y. L. Kuo, A People Counting System Based on Face-Detection, *International Conference on Genetic and Evolutionary Computing*, Shenzhen, China, 2010, pp. 699-702. https://doi.org/10.1109/ICGEC.2010.178

[8] S. Ge, Q. Ye, Z. Luo, S. Zhao, Try Everything: Detecting Occluded Faces by Cascading Outrageous Proposal Generation and Deep Convolutional Neural Network, *IEEE International Conference on Multimedia Big Data*, Laguna Hills, CA, USA, 2017, pp. 193-196. https://doi.org/10.1109/BigMM.2017.40

[9] S. Li, X. Ning, L. Yu, L. Zhang, X. Dong, Y. Shi, Multi-angle Head Pose Classification when Wearing the Mask for Face Recognition under the COVID-19 Coronavirus Epidemic, *International Conference on High Performance Big Data and Intelligent Systems*, Shenzhen, China, 2020, pp. 1-5. https://doi.org/10.1109/HPBDIS49115.2020.9130585

[10] X. Wang, S. G. Miaou, Y. C. Lin, Mask Wearing Identification based on Deep Learning Object Detection Networks, *Journal of Technology*, Vol. 37, No. 1, pp. 53-63, 2022.

[11] C. Y. Yang, H. Samani, N. Ji, C. Li, D. B. Chen, M. Qi, Deep Learning Based Real-Time Facial Mask Detection and Crowd Monitoring, *Journal of Computing and Informatics*, Vol. 40, No. 6, pp. 1263-1294, 2021. https://doi.org/10.31577/cai_2021_6_1263

[12] Y. Hara, A. Uchiyama, T. Umedu, T. Higashino, Sidewalk-level People Flow Estimation Using Dashboard Cameras Based on Deep Learning, *International Conference on Mobile Computing and Ubiquitous Network*, Auckland, New Zealand, 2018, pp. 1-6. https://doi.org/10.23919/ICMU.2018.8653595

[13] J. Z. Wang, *Evaluation of Face Detectors and Feature Association Metrics for Real-Rime Multi-Face Tracking*, Ph.D. Thesis, University of Ottawa, Ottawa, Canada, 2020.

[14] M. Mohaghegh, Z. Pang, A Four-Component People Identification and Counting System Using Deep Neural Network, *Asia-Pacific World Congress on Computer Science and Engineering*, Nadi, Fiji, 2018, pp. 10-17. https://doi.org/10.1109/APWConCSE.2018.00011

[15] S. V. Militante, N. V. Dionisio, Real-Time Facemask Recognition with Alarm System Using Deep Learning, *IEEE Control and System Graduate Research Colloquium*, Shah Alam, Malaysia, 2020, pp. 106-110. https://doi.org/10.1109/ICSGRC49013.2020.9232610

[16] S. Sethi, M. Kathuria, T. Kaushik, Face Mask Detection Using Deep Learning: An Approach to Reduce Risk of Coronavirus Spread, *Journal of Biomedical Informatics*, Vol. 120, Article No. 103848, August, 2021. https://doi.org/10.1016/j.jbi.2021.103848

[17] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation, *IEEE Conf. on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014, pp. 580-587. https://doi.org/10.1109/CVPR.2014.81

[18] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, pp. 1137-1149, June, 2017. https://doi.org/10.1109/TPAMI.2016.2577031

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, A. C. Berg, SSD: Single Shot MultiBox Detector, *European Conference on Computer Vision*, Amsterdam, The Netherlands, 2016, pp. 21-37. https://doi.org/10.1007/978-3-319-46448-0_2

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

[21] J. Redmon, A. Farhadi, YOLO9000: Better, Faster, Stronger, *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 6517-6525. https://doi.org/10.1109/CVPR.2017.690
[22] J. Redmon, A. Farhadi, *YOLOv3: An Incremental Improvement*, arXiv preprint arXiv:1804.02767v1, April, 2018. https://arxiv.org/abs/1804.02767
[23] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature Pyramid Networks for Object Detection, *IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 936-944. https://doi.ieeecomputersociety.org/10.1109/CVPR.2017.106
[24] A. Bochkovskiy, C. Y. Wang, H. Y. M. Liao, *Yolov4: Optimal Speed and Accuracy of Object Detection*, arXiv preprint arXiv:2004.10934, April, 2020. https://arxiv.org/abs/2004.10934
[25] G. Jocher, *Ultralytics YOLOv5, version 6.0*, 2020. [Online]. Available: https://github.com/ultralytics/yolov5
[26] N. Wojke, A. Bewley, D. Paulus, Simple Online and Realtime Tracking with a Deep Association Metric, *IEEE International Conference on Image Processing*, Beijing, China, 2017, pp. 3645-3649. https://doi.org/10.1109/ICIP.2017.8296962
[27] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and Realtime Tracking, *IEEE International Conference on Image Processing*, Phoenix, AZ, USA, 2016, pp. 3464-3468. https://doi.org/10.1109/ICIP.2016.7533003
[28] GitHub, *Eden social welfare foundation, Dataset for mask wearing*, 2020. https://github.com/ch-tseng/Dataset_for_Mask_Wearing

Biographies

Shyang-En Weng received a B.S. degree in Electronic Engineering from CYCU in 2022. He is pursuing a dual M.S. degree in Electronic Engineering at CYCU and in Electrical Engineering at the University of Wisconsin-Milwaukee. His research interests include computer vision and pattern recognition.

Ying-Cheng Lin was born in Nantou City, Taiwan, in 1991. He received the B.S., M.S., and Ph.D. degrees from the Department of Electronic Engineering, CYCU, in 2013, 2016, and 2023 respectively. His research interests include image processing and deep learning.

Ming-Yao Liang received a B.S. degree in Electronic Engineering from CYCU in 2022. His research interests include image processing and deep learning.

Shaou-Gang Miaou received the Ph.D. degree in EE from the University of Florida, USA, in 1993. He is a professor with the Department of Electronic Engineering, CYCU. He served as the Dean of College of EE&CS, CYCU, from 2017 to 2022. His research interests include image processing and pattern recognition.