International Journal on Semantic Web and Information Systems
Volume 19 • Issue 1

A Lightweight Real-Time System for Object Detection in Enterprise Information Systems for Frequency-Based Feature Separation

YiHeng Wu, China University of Mining and Technology, China*
JianXin Chen, China University of Mining and Technology, China

ABSTRACT

In the domain of target detection in mobile and embedded devices, neural network model inference speed is a crucial metric. This paper introduces YOLO-FLNet, a lightweight algorithm for detecting people in open scenes. The model utilizes the DFEM structure to capture and process high-frequency and low-frequency information in the feature map. Additionally, the VoV-DFEM structure, based on the concept of one-shot aggregation, enhances feature aggregation from different scales and frequencies in the backbone network. To validate its performance, experiments were conducted using publicly available datasets on a computer with dedicated GPUs. As a result, compared to YOLOv7-tiny, YOLO-FLNet achieved a 0.3% mAP@0.5 improvement, reduced parameter size by 52.9%, and increased inference speed by 30.2%. These characteristics make it valuable for person detection in engineering domains, providing theoretical guidance for lightweight models in edge computing.

KEYWORDS

Feature Fusion, Frequency-Based Information, Lightweight Network, Personnel Detection

The popularity of mobile phones and other devices in people's daily lives and socialization in the 21st century has led to the generation of a significant amount of data. Correspondingly, many enterprises can now efficiently distribute and process this information due to the rapid development of cloud computing technology in recent years (Vano et al., 2023). However, cloud computing will have to bear inevitable delays, additional power consumption, and corresponding costs (Peñalvo et al., 2022), and the IoT system relying on it also faces challenges in software and hardware security (Gaurav et al., 2023; Memos et al., 2018) as well as the rationality of computing logic (Guebli & Belkhir, 2021). As mobile chips and embedded platforms continue to advance, offloading data to edge computing platforms for processing has become a viable solution (Kang, 2023). Edge computing technology offers real-time processing capabilities and improved security, partially mitigating the challenges

DOI: 10.4018/IJSWIS.330015 *Corresponding Author

faced by cloud computing (Al-Qerem et al., 2020; Li & Huang, 2021; Stergiou et al., 2022). As computing power becomes more reliable on target platforms, researchers have shifted their focus toward edge algorithms. In particular, object detection algorithms applied in real-world scenarios have gained significant attention.

Currently, machine learning has been widely applied by researchers in various fields, including information security (Almomani et al., 2022; Li et al., 2022; Sahoo et al., 2021), autonomous driving (Arikumar et al., 2022; Prathiba et al., 2022), and agricultural production (Kumar et al., 2021; Dan, 2022), and has demonstrated excellent results. Object detection algorithms, in particular, are closely related to these areas of study. Object detection, an essential research direction in computer vision, aims to identify specific objects in digital images. When designing relevant algorithms, researchers must consider classification accuracy, localization precision, and processing speed as the primary evaluation metrics. These metrics determine the suitability of algorithms for specific task scenarios. Early object detection algorithms relied on manual construction and clever feature analysis, resulting in unsatisfactory accuracy, running speed, and generalizability (Zaidi et al., 2022). With the birth and development of convolutional neural networks (CNNs) that can learn more robust and abstract features, explosive growth has occurred in fields such as natural language processing (Zhang et al., 2023) and image processing (Al Sobbahi & Tekli, 2022). Based on this, when the region-based convolutional neural network (RCNN) appeared, the field of target detection entered a new stage of rapid development (Zou et al., 2023). Recent years have witnessed the proposal of various detection algorithms using different methods, such as anchor-based detectors, which have demonstrated excellent performance in detection tasks (Amjoud & Amrouch, 2023).

In object detection, classical two-stage detection algorithms, like Faster R-CNN, employ complex network models that can use the dataset of the input model directly for training. These algorithms first identify region proposals and then perform classification and position regression within these regions. This facilitates a more operable model training process and results in significantly improved precision and recall compared to traditional object detection algorithms. On the other hand, detection methods based on the one-stage concept, such as You Only Look Once (YOLO), achieve end-to-end object detection using a single CNN model. The core idea is to use the entire image as the input and directly regress the position of bounding boxes and their corresponding categories at the output. One-stage detection methods are generally faster than two-stage algorithms in terms of detection speed, making them preferable for scenarios with limited computing power. The YOLO algorithm has made significant contributions to the implementation of enterprise information systems and has demonstrated its commercial value across diverse fields, as evident in the findings presented in Table 1.

While the YOLO series detectors have shown promising potential in these domains, implementing lightweight enhancements tailored explicitly for detecting human targets on edge devices is imperative. Furthermore, enterprises often utilize edge sensors, such as closed-circuit cameras, in practical applications to collect personnel data in relatively open scenes. In light of these requirements, this paper proposes YOLO-FLNet, a lightweight model designed for edge computing, aiming to improve precision and inference performance. The model utilizes the diverse frequency extract module (DFEM)

Table 1. Different improvement schemes

Involved Detector	Properties of the Standard Model (Input 640×640)		Application Area
	params(M)	FLOPs	
YOLOv3	61.9	156	Traffic management (Al-qaness et al., 2021; Rudregowda et al., 2021).
YOLOv5s	7.3	16.4	Infrastructure management (Inam et al., 2023)
YOLOv7	36.5	103	Breeding industry (Ranjan et al., 2023)

and its corresponding VoV-DFEM structure, introduced in this study, for extracting and aggregating features with differential frequency information. Additionally, the slim spatial pyramid pooling (SSPP) structure and light feature pyramid networks (light-FPN) with a compact design are incorporated to enhance the model's recognition performance for multi-scale objects. The experiments used a subset from the MS COCO dataset (Lin et al., 2014) and PASCAL VOC (Everingham et al., 2010), which consists of personnel targets in open scenes. The results demonstrate that these innovations enable YOLO-FLNet to outperform other lightweight models in terms of mAP@0.5 and inference efficiency.

RELATED RESEARCH

Backbone

In recent years, researchers have made significant advancements in the field of lightweight backbones aimed at reducing the parameter count and improving the detection accuracy and inference efficiency of neural network models. Important considerations among these advancements include model size reduction, memory access cost (MAC), and the computational efficiency of the GPU. Taking the impact of MAC on model performance into account, Lee et al. (2019) proposed VoVNet, which adopts the OSA concept. VoVNet achieves a speed twice as fast as densely connected convolutional networks and reduces energy consumption by over 1.5 times. Han et al. (2020) observed the feature maps generated by residual structures and found that many of them are redundant and can be obtained through linear transformations. Consequently, they proposed the ghost convolution module, which can selectively process intrinsic feature maps and ghost feature maps differently. By partitioning the features along the channel dimension into different branches, the ghost convolution module effectively reduces the parameter size, improves computational efficiency, and maintains the module's feature extraction capability. It has been proven to achieve outstanding performance in model lightweighting and runtime efficiency (Ma et al., 2023; Xu et al., 2022; Zhao et al., 2021). Jiang et al. (2022) discovered that when the spatial information of high-resolution feature maps interacts effectively with the semantic information in low-resolution feature maps, even models with extremely lightweight backbone networks can achieve good detection performance on the COCO dataset. To strike a balance between more robust learning capability during training and higher efficiency during inference, Ding et al. (2021) applied the re-parameterization technique in RepVGG to adjust the model structure. This technique allows RepVGG to maintain higher accuracy while achieving an inference speed that is 83% faster than the classic deep residual network. To lightweight the model and enhance its robustness against interference, Zeng et al. (2022) designed the improved dense dilated convolution (IDDC) block in the network structure. Their proposed LDSNet limits the parameter size to within one million while maintaining high accuracy. In their research, Huang et al. (2022) proposed the lightweight oriented object detector (LO-Det) and dynamic receptive field (DRF) to improve the detection performance of the model. The CSA-DRF component exhibited good efficiency and accuracy in their experiments. Mehta and Rastegari (2021) introduced the MobileViT block, a lightweight universal transformer structure suitable for mobile devices. As a backbone, it reduces the parameter size by over 90% compared to ResNet-101. These creative efforts have yielded practical achievements in various fields. However, the decrease in feature extraction capability during the construction of lightweight models is a problem that cannot be ignored. Additionally, when feature maps are propagated in network models, they always carry specific frequency information. However, none of the abovementioned methods have specialized structures to extract such frequency-based information. In this research, a method is proposed to fuse features with different information frequencies in the feature maps to minimize the loss in precision.

Feature Pyramid Networks

In practical object detection tasks, the targets may have different sizes and be occluded, among other situations. Early algorithms that only used a single feature map for prediction had limited receptive fields, and the generated regions were only associated with fixed areas of vague features, making it difficult for them to handle these challenging tasks. Fortunately, researchers have proposed the feature pyramid network (FPN) and path aggregation network (PANet) methods to enhance learning multi-scale feature representations with rich semantic information. These approaches have undergone extensive experimental validation, confirming their effectiveness. Considering the self-optimization ability of neural networks, Ghiasi et al. (2019) proposed the NAS-FPN, where the RNN structure in this research can construct and select merging cells in the model. This modular search makes the pyramid architecture easier to manage, and models adopting it have achieved state-of-the-art performance in various tasks (Ghiasi et al., 2019; Wang et al., 2021; Yu et al., 2021). Furthermore, Li et al. (2021) proposed AutoDet, which can enhance the efficiency of FPN and achieve an AP of 47.3 on the COCO dataset. To better utilize contextual feature information, Wang and Zhong (2021) introduced an adaptive feature pyramid network, which primarily involves adaptive feature upsampling and fusion. After incorporating it into Faster R-CNN, it achieved a performance improvement of 1.0 average precision (AP) on the COCO dataset. Additionally, researchers in EfficientDet (Tan et al., 2019) proposed Bi-FPN with a weighted bi-directional design, which achieved a new state-of-the-art COCO AP of 55.1% while only using 77M parameters. To further improve the performance of cross-scale connection (CSC), Wang et al. (2020) introduced an implicit feature pyramid network (i-FPN) based on fixed point iteration to generate balanced features with global receptive fields directly. This structure increased the AP of Faster R-CNN on the MS COCO dataset by 3.2%. To make the model adapt to more complex scenes of target categories, Kim and Chi (2021) proposed SAFFNet with self-attention, which contributed to improving classification accuracy. To optimize the detection of small volumes and high-density objects, Sun et al. (2022) proposed an FPN structure that can better fuse local and global features, achieving a 3.4% improvement in mAP@0.5 compared to YOLOv3. However, previous research has found it challenging to find effective methods for carefully fusing features with different frequency information in the backbone network. Therefore, this study proposes a lightweight FPN structure with a simplified design, which is used to efficiently aggregate complex features from different stages of the backbone network.

YOLO Series Algorithms

Considering the demand for lightweight deployment, one-stage detectors are naturally considered first at the algorithm level. As a classic algorithm in one-stage object detection, YOLO plays a significant role in developing real-time detection. Compared to the classical two-stage models, YOLO does not require a region proposal process and only needs one forward computation of the convolutional neural network to perform object detection and classification simultaneously. It can achieve a frame rate of 45 FPS on GPUs. Viewed from the standpoint of the YOLO algorithm family's evolution, the incorporation of pioneering elements, including anchor boxes, batch normalization, and FPN, has resulted in remarkable advancements in accuracy and inference speed with the introduction of the third generation. Based on these advancements, subsequent models, such as YOLOv4 (Bochkovskiy et al., 2020), YOLOv5 (Jocher, 2021), and YOLOX (Ge et al., 2021), introduced different improvements in the input, backbone network, multi-scale feature aggregation, and output, resulting in significant performance improvements in various aspects.

In recent studies, researchers have introduced novel designs in YOLOv7 (Wang et al., 2022). YOLOv7 offers models of varying scales, such as YOLOv7-W6 and YOLOv7D6, which have different sizes and depths for various usage scenarios. In larger-scale models, such as YOLOv7-E6E, an extended efficient layer aggregation network (E-ELAN) is introduced. Compared to the efficient layer aggregation network (ELAN) used in smaller-scale models in YOLOv7, E-ELAN only changes the architecture of the computation block, allowing for grouped parallel computing in the calculation

process without modifying the architecture of the transition layer. Based on this module, researchers also proposed using model scaling for concatenation-based models, which matches the depth and width scaling factors with ELAN or E-ELAN structures to adapt to different scenarios. On the other hand, YOLOv7 redesigns an application strategy related to re-parameterization. Additionally, the model employs RepConv at the end to improve detection precision without significantly increasing the inference time. In the training process, larger YOLOv7 models use lead head predictions as guidance to generate hierarchical labels from coarser to finer levels, which are used for both auxiliary head and lead head learning. With the introduction of these innovative designs, YOLOv7 surpasses the target detectors in speed and precision within the range of 5–160 FPS. Particularly on the COCO dataset, YOLOv7-E6E achieves a maximum AP improvement of 13.7% compared to Meituan/YOLOv6-s (Li et al., 2022).

This paper introduces the YOLO-FLNet model to optimize person recognition in open scenes and platforms with limited computing resources, utilizing the YOLO series algorithms framework. Firstly, the paper introduces the diverse frequency extraction module, which extracts and blends the information of different frequencies in the features through two similar stages with double branches and cascades and outputs them along the channel direction at the end of the branches. Based on this structured design, the VoV-DFEM module can effectively fuse the feature information from different stages of DFEM through a single pathway. The VoV-DFEM module is applied in the lightweight-designed light feature pyramid networks to efficiently combine multi-scale, multi-frequency features from the backbone network. Additionally, the paper employs SSPP at the end of the backbone network to extract more abstract information at different resolutions, complementing the use of DFEM.

Lastly, compared to the lightweight YOLOv7-tiny model, the YOLO-FLNet with these novel modules achieves a slight improvement in mAP@0.5 while reducing the parameter size and inference time to 3.93M and 6.3ms, which account for approximately 52.9% and 30.2% optimization, respectively. The main contributions of this work can be summarized as follows:

1) This paper introduces the DFEM, SSPP, and VoV-DFEM. Experimental results demonstrate their capability to effectively separate and aggregate robust features from input information of different sizes and frequencies in neural networks.
2) The paper presents a lightweight backbone network structure and light-FPNs. Empirical evidence suggests that these structures can demonstrate good performance in tasks with a more streamlined architecture.
3) The proposed YOLO-FLNet represents a feasible solution for target detection applications with limited computational power. It combines computational efficiency, storage efficiency, and detection precision, as confirmed by experimental validation.

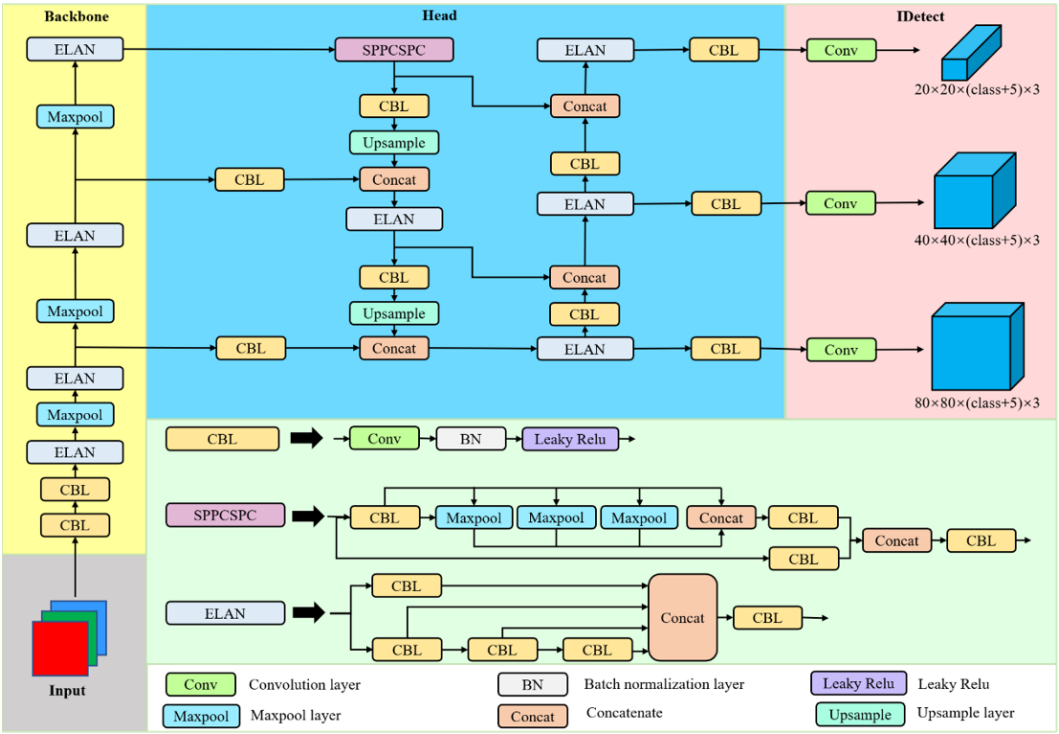
IMPLEMENTATION OF THE NETWORK

Network Architecture of YOLO-FLNet

Among the various models in the YOLOv7 series, YOLOv7-tiny is a lightweight model suitable for edge computing scenarios (see Figure 1). Regarding the ELAN, spatial pyramid pooling, and convolutional spatial pyramid pooling (SPPCSPC) structures in YOLOv7, YOLOv7-tiny has downscaled them. After inputting a 640x640 image, the model first goes through processing by its backbone network. The generated features at different resolutions are then fused through the head structure and subjected to final regression prediction after being processed by the IDetect detection head. On the COCO dataset, the YOLOv7-tiny model achieves a 25% improvement in AP and a 0.2% improvement in inference speed compared to the Meituan/YOLOv6-n model.

To better adapt to applications in edge computing scenarios, this paper further improves the model's backbone and head based on YOLOv7-tiny. Through steps such as network structure analysis,

Figure 1. The structure of the YOLOv7-tiny



module replacement, model training, and model testing, a lightweight network model called YOLO-FLNet is proposed for applications with limited computing power. This model utilizes an efficient and lightweight backbone and a head structure capable of efficiently aggregating complex features (see Figure 2).

Taking a 640x640 input tensor as an example, in the backbone of YOLO-FLNet, after downsampling through the second CBL block (comprising a convolutional layer, BN layer, and Leaky ReLU) with a stride of 2, a feature map with a size of 160x160 is obtained. Subsequently, the DFEM structures and CBL blocks are alternatively used to further extract features of different scales.

Diverse Frequency Extract Module

DFEM is a core component in the backbone network of YOLO-FLNet. Its crucial functionality involves using Maxpool and convolution to reduce the model's parameter size and computational cost. After passing through a certain depth of the neural network, the output features from convolution and other computational units are already capable of differentiating target information from background or foreground information to some extent. At this stage, Maxpool can extract information with drastic changes in image or semantics in the form of extreme values within a region, thereby extracting high-frequency features. As shown in Figure 3, this unit can be divided into two stages by the channel shuffle operation, with each stage containing two branches for extracting different frequency features.

When the feature map is passed into the first stage of this module, it is divided into two parts in the depth direction by the channel split operation. Then, these are processed separately in different branches for feature extraction. In the high-frequency information extraction branch, a Maxpool layer with a kernel size of 3x3 and a stride of 1 is used to extract the maximum value within the feature range, thus extracting the high-frequency information from the feature map. In the other branch, a

Figure 2. The structure of the YOLO-FLNet

Note. DEFM is the equal-scale sampling unit, and VoV-DFEM is the subsampling unit based on DFEM.

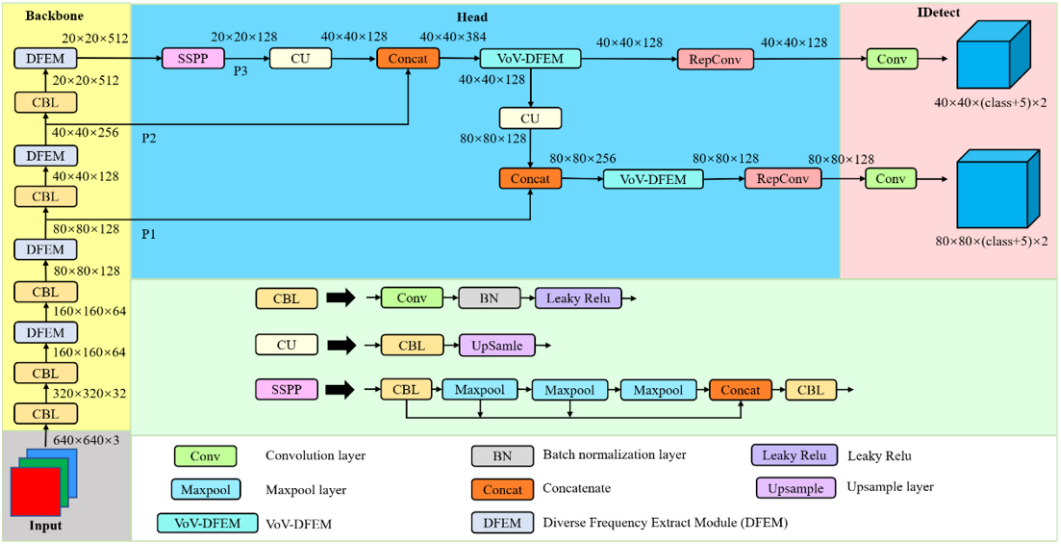
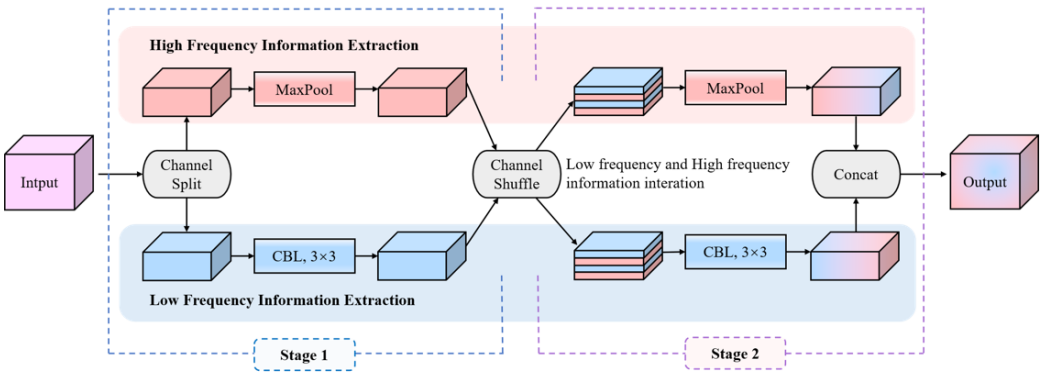


Figure 3. The structure of the diverse frequency extract module



convolution with the same kernel size and stride is used to process the input feature map and extract relatively lower-frequency information. The information of different frequencies extracted by these two branches is stacked and interleaved in the depth direction through channel shuffle, allowing the information to be recombined to obtain mixed-frequency features, which serve as input for the second stage. The same operation is performed in the branches of the second stage, and the separated features will be stacked pointwise to obtain the final output of this module.

In each stage of DFEM processing, the feature map is partitioned into equal scales based on the number of channels. Subsequently, it undergoes processing without altering the channel number, followed by merging. The resulting feature map maintains the same size and channel number as the input. This method guarantees that half of the branches in DFEM have no additional parameters, while the input channels of the convolutional layers in the other half are halved from the initial input channel number. This design achieves a lightweight structure while extracting features based

on differences in information distribution. In other words, the DFEM module can separate different frequency features, like person and background, with fewer parameters.

Slim Spatial Pyramid Pooling

The variations in these backbone networks necessitate corresponding adjustments to the model's head section to ensure adaptability. To obtain feature maps with multiple resolutions at the end of the backbone network, this research proposes a slim spatial pyramid pooling (SSPP) structure, taking inspiration from ELAN and the SPPF structure in YOLOv5. This structure is connected to the end of the backbone in the head (see Figure 5). If we let the input feature map of this module have C channels, after passing through the point-wise CBL block, the channels are compressed to 1/4C. Then, they undergo three consecutive Maxpool layers. The output results of each structure are stacked along the channel direction and processed by the CBL block to produce an output with 1/4C channels.

This design effectively reduces the width of the SPP structure through the concatenation of Maxpool layers. Adopting the OSA design improves the computational efficiency of the model. At the same time, SSPP ensures the structure's receptive field while effectively limiting the module's parameter size. In summary, SSPP enables the processing of multi-scale features with reduced parameter and computational requirements.

VoV-DFEM

In addition to SSPP, the head section of the model incorporates innovative computational modules to further enhance its capabilities. To better handle complex information with different resolutions and frequencies, VoV-DFEM was designed based on DFEM. The main computational part of this unit, as shown in Figure 4, includes one DFEM and two CBL blocks. The input feature map in this unit is first compressed using a point-wise operation by a CBL block, and the resulting output is passed into DFEM to extract features of different frequencies. Subsequently, the outputs of the convolutional and DFEM stages in this processing pipeline are concatenated along the channel dimension, and the output of the CBL block serves as the final result of this unit.

In this structure, inspired by the OSA design approach mentioned in VoVNet (Lee et al., 2019), the features extracted at each stage of the process are directly aggregated in the channel dimension at the end of VoV-DFEM. This effectively separates the features required for subsequent processing from the rich information extracted by each stage. Moreover, by applying OSA, the MAC of this structure is reduced, leading to improved computational efficiency. Based on this, it can be concluded that this module combines the feature separation function of the DFEM module for different frequencies while also possessing computational efficiency in its structure.

The Designed Backbone Network and FPN

To apply the abovementioned modules more effectively, it is necessary to make corresponding modifications to different model components. In the backbone network of YOLO-FLNet, we replaced

Figure 4. The structure of slim spatial pyramid pooling

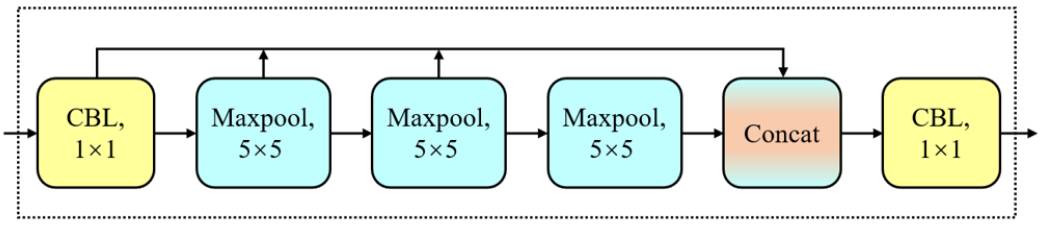
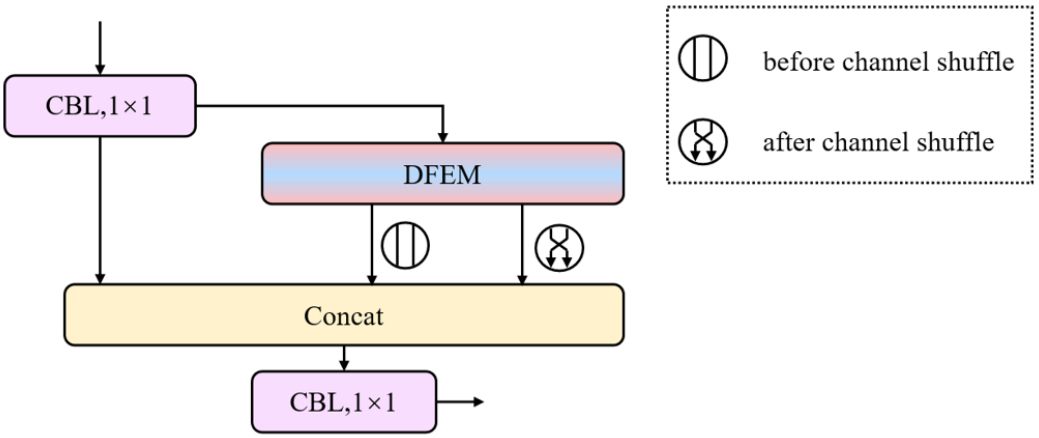


Figure 5. The structure of VoV-DFEM



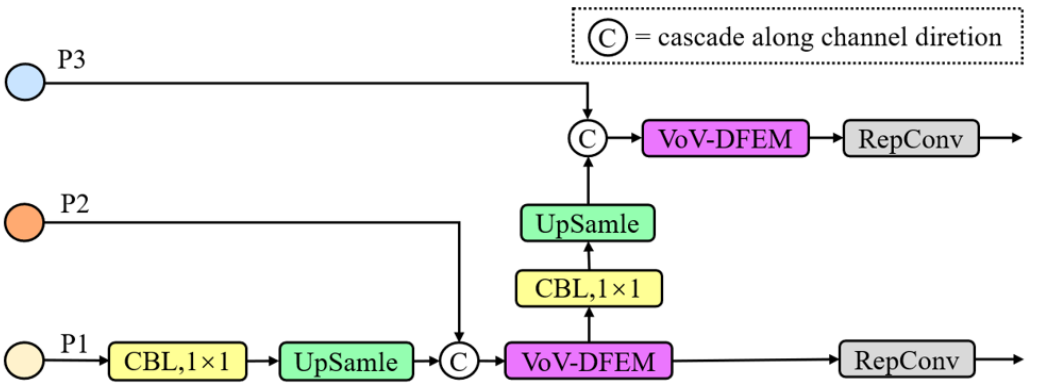
the ELAN module from the original YOLOv7-tiny backbone network with DFEM. They are connected by CBL blocks, enabling efficient extraction, separation, fusion, and activation of information of different frequencies in the input image. The output of the second DFEM (P1), the third DFEM (P2), and the end of SSPP (P3) of the backbone network are sequentially passed into the light-FPN, as shown in Figure 6.

In light-FPN, the features from SSPP are first stacked with the larger-sized features received from the backbone after CU block processing. Subsequently, these stacked features are processed by VoV-DFEM. The result of this processing is then subjected to the same treatment by another VoV-DFEM module. The outputs of these two VoV-DFEM modules undergo reparameterized convolution and finally pass through IDetect for model output. This pyramid network architecture efficiently handles multi-resolution features, including frequency information, from the backbone network.

EXPERIMENTAL DESIGN

This paper trains YOLO-FLNet on publicly available datasets and evaluates its performance in detecting human targets within specific scenarios. First, to demonstrate the performance of the

Figure 6. The structure of light feature pyramid networks



final lightweight person object detection model, a comparison is made with state-of-the-art object detection algorithms. Second, ablation experiments are conducted to assess and explain the impact of the lightweight modules designed in this paper on model detection precision, inference speed, and model size.

Experimental Environment

The implementation of the solution utilized a software environment comprising the Windows 10 operating system, PyTorch 1.9.0 based on Python 3.9 as the deep learning framework, CUDA 11.1, and cuDNN 8.0.5 for computational architecture. The hardware conditions included an NVIDIA GeForce RTX 3090 GPU with 24GB memory, Intel i7 12700KF CPU, and 32GB RAM. During the experiments, the model was trained on the GPU using an optimizer. In order to ensure experiment reproducibility, the methods and hyperparameters used for model training remained consistent across different model training setups. The experimental settings were as follows: no pre-trained weights were utilized for training; the stochastic gradient descent (SGD) optimizer with an initial learning rate of 1e-2 was chosen, with optimization momentum and corresponding weight decay set to 0.937 and 5e-4, respectively; training was conducted for 300 epochs with a warm-up period of three initial training epochs; and random translation, horizontal flipping, HSV color space transformation, and mosaic augmentation were applied to the input images to enhance the robustness of the features learned from the data. During testing, the confidence and IOU thresholds were set to 1e-3 and 0.5, respectively.

Dataset

In enterprise information systems, the personnel targets captured by edge devices, such as monitors in closed-circuit surveillance systems, are commonly situated a significant distance from the camera, resulting in a wide field of view. To simulate the object data in open scenes more accurately, this experiment utilized a dataset comprising 5,000 images sampled from MS COCO and PASCAL VOC. The dataset primarily includes open scenes typically encountered in people's daily lives, such as streets, shopping malls, parking lots, and natural environments. No relatively complex patterns appear in these background scenes, and the size of the human objects does not dominate the overall image composition. Moreover, the dataset includes human targets from different age groups, focusing primarily on young adults. These human targets exhibit various states, including walking, standing, sitting, and dancing. The corresponding labels for these images consist of 13,259 ground truth bounding boxes. To maintain the aspect ratio, the images were resized proportionally to have a maximum side length of 640 pixels. They were padded with zero values along the shorter side, resulting in 640×640 images. By analyzing the ratios of the longer side of each scaled bounding box to the image side length, the label distribution of the dataset was obtained, as presented in Table 2.

In the processed dataset, most of the ground truth bounding boxes have a longer side that occupies up to 60% of the image side length, indicating medium-sized or slightly smaller targets. Additionally, the dataset includes a smaller proportion of large and extremely small object boxes, which more closely resembles the image acquisition scenarios in the practical deployment of the algorithm. To ensure the normal operation of the experiments, a random selection of 3,000 images from the dataset was allocated for model training, while the remaining 2,000 images were divided equally, with 1,000 images used for validation and 1,000 images used for testing.

Table 2. The size statistics of the ground true box in 640×640 image after processing

Proportion of the longest edge of the box (%)	0–20	20–40	40–60	60–80	80–100
Box quantity ratio (%)	36.4	45.4	15.2	2.4	0.6

Evaluation Criteria

The experiments employed commonly used performance analysis metrics in the field of object detection as evaluation indicators for the models: precision and recall. True positive (TP) predictions, false positive (FP) predictions, and false negative (FN) predictions were utilized for calculating these metrics. Precision represents the proportion of accurately detected targets among all positive predictions, with a higher value indicating fewer errors in the predicted results. Recall represents the proportion of correctly detected positive instances among the actual positive instances, with a higher value indicating fewer missed positive instances. During the calculation of precision and recall, the confidence threshold and intersection over union (IOU) threshold had to be considered as they significantly affected the computation results. The calculation formulas for precision and recall are represented by Equations 1 and 2, respectively.

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

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

The precision–recall (P–R) curve can be plotted based on precision and recall statistics. The average precision (AP) is calculated based on the area enclosed by the curve and the recall axis. Introducing AP allows for a comprehensive evaluation of the object detection performance of the model. For the experimental task in this paper, as only human targets are detected, the mAP value is equal to the AP value, and its calculation follows Equation 3. Additionally, the number of parameters and inference time are essential metrics for evaluating the model's performance in this experiment.

$$AP = \int_0^1 P\left(R\right) dR \tag{3}$$

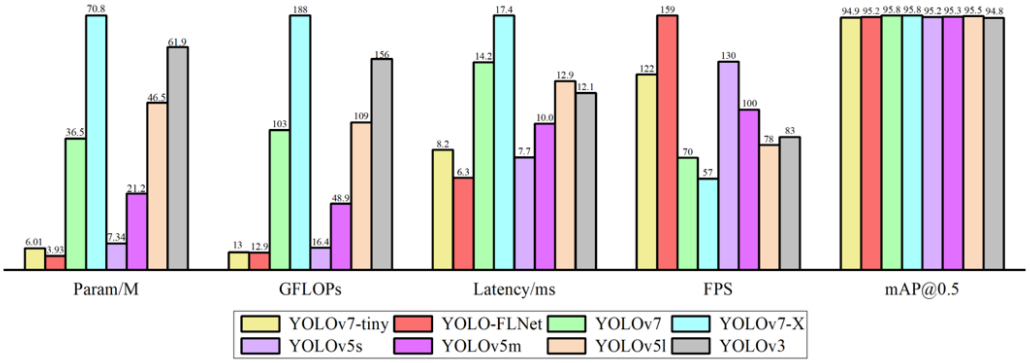
Comparison with State-of-the-Art Models

To establish the superiority of YOLO-FLNet in the field of detection tasks on the dataset, the experiment involved comparing its performance with several one-stage algorithms that have exhibited exceptional results. The experimental results are presented in Figure 7. Based on these results, the following conclusions can be drawn: YOLO-FLNet outperforms the other networks regarding model size, computational complexity, inference latency, and frames per second (FPS). It showcases significant parameter count and inference speed optimizations, even when compared to lightweight models such as YOLOv7-tiny and YOLOv5s. Specifically, compared to the larger-scale YOLOv5s, YOLO-FLNet achieves comparable mAP@0.5 performance while reducing parameter count by 46.5% and improving inference speed by 22.3%. The precision improvement resulting from models with more parameters, such as YOLOv7-X and YOLOv5l, should not be ignored; nevertheless, YOLO-FLNet, being the most compact model, possesses merely 5.6% of the parameters in YOLOv7-X and 8.5% of YOLOv5l, alongside significantly improved inference speed, providing it with a distinct advantage for the given task.

Ablation Experiments

To further explore the impact of the DFEM, SSPP, and Light-FPN on the model's performance in this task, different improvement schemes were tested on the data mentioned above, and the combinations of these schemes are shown in Table 3. In the FL-CBL scheme, the CBL block replaced the Maxpool layer connected to ELAN in the backbone network, compared to YOLOv7-tiny. In the FL-DFEM scheme, the DFEM replaced the ELAN structure in the original backbone network with CBL blocks

Figure 7. Comparison of object detection algorithms on the test set



used to connect modules. In the FL-SSPP scheme, only the SSPP was used to replace the SPPCSPC structure at the end of the backbone network. Based on FL-DFEM, the FL-DFEM-SSPP scheme applied the SSPP module in the head, while the FL-DFEM-LFPN scheme adopted the light-FPN structure. The FL-DFEM-SSPP-LFPN scheme utilized all the optimization modules designed in this study.

Stability Testing

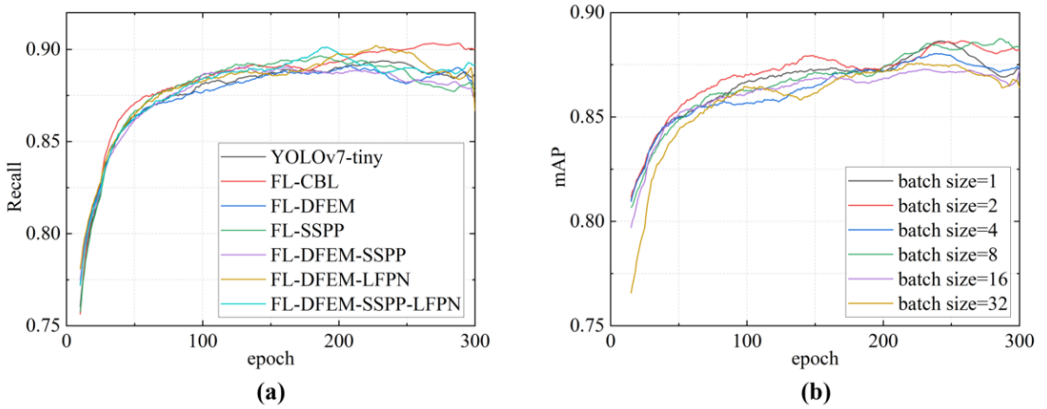
Before proceeding with further experiments, validating the stability and rationality of the proposed approach used in the experiments was essential. Figure 8(a) shows each scheme's recall–epoch relationship during the training process, where the x-axis represents the training epochs, and the y-axis represents the corresponding recall at each epoch. It can be observed that with the increase in epochs, the recall of models using each scheme tends to stabilize after a rapid increase, reaching a relatively stable state at around 200 epochs. This indicates that the abovementioned optimization schemes have almost no negative impact on the learning performance of the model while adjusting its structure and can effectively optimize the model's performance in a smooth manner. Among them, the model obtained from the FL-DFEM-SSPP-LFPN scheme achieves a 1.5% increase in recall compared to YOLOv7-tiny, demonstrating its more reliable object detection capability.

Based on the above performance, this study further investigated the influence of different batch sizes on the precision of the FL-DFEM-SSPP-LFPN scheme, as shown in Figure 8(b). The trend of this scheme remained consistent across different batch sizes, fully demonstrating the stability and reliability of the proposed improvement points in this paper. In addition, the model obtained with a

Table 3. Different improvement schemes

Scheme	CBL	DFEM	SSPP	Light-FPN
FL-CBL	✓			
FL-DFEM	✓	✓		
FL-SSPP			✓	
FL-DFEM-SSPP	✓	✓	✓	
FL-DFEM-LFPN	✓	✓		✓
FL-DFEM-SSPP-LFPN	✓	✓	✓	✓

Figure 8. (a) Recall–epoch curves of improvement schemes, (b) mAP–epoch curves of scheme FL-DFEM-SSPP-LFPN with different batch sizes



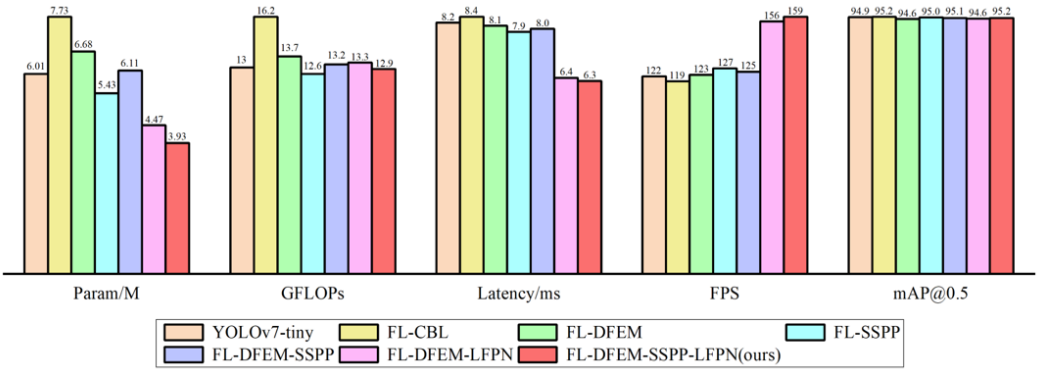
batch size of eight had a relatively high mAP of 90.1, with differences of within one percentage point compared to other training strategies, indicating the rationality of the batch size value set in this study.

Performance Testing

The results of the ablation experiments are shown in Figure 9. These indicate that introducing different computational units in each scheme can impact the model's parameter size, FLOPs, latency, and precision. In the FL-CBL scheme, although introducing the relatively simple CBL structure results in a 0.3% improvement in mAP@0.5 compared to YOLOv7-tiny, the cost is an increase in model size and inference time. By contrast, in the FL-DFEM scheme, the precision decreases, indicating that the original feature pyramid structure cannot effectively aggregate information of different frequencies and sizes extracted by DFEM. The results of the FL-SSPP scheme indicate that SSPP contributes to reducing the model's size and computational complexity, with an optimization of approximately 0.58M parameters and 0.4 GFLOPs, respectively, leading to a slight reduction in inference time. Similarly, in the FL-DFEM-SSPP scheme, the improvement in computational efficiency leads to a slight reduction in latency of approximately 0.1ms and an increase of 0.2% in mAP@0.5. In the FL-DFEM-LFPN scheme, the fusion of complex features extracted by the light-FPN backbone network resulted in significant reductions of 1.81M parameters and 0.8 GFLOPs, leading to an effective reduction of 1.7ms in inference time. This indicates that applying VoV-DFEM to the designed head part can improve the model's inference performance. For the FL-DFEM-SSPP-LFPN scheme, both the model size and computational complexity were significantly optimized compared to YOLOv7-tiny: the reductions in parameters and inference time were approximately 52.9% and 30.2%, respectively, and there was a 0.3% improvement in mAP@0.5 compared to YOLOv7-tiny. Finally, this study aimed to strike a balance between model performance and inference speed during the actual deployment process by considering performance indicators such as mAP@0.5, storage capacity, and computational requirements while addressing the task of detecting human subjects in open scenes with YOLO-FLNet. The proposed solution adopted in this research to address this challenge is FL-DFEM-SSPP-LFPN.

Edge computing platforms, such as smartphones and embedded development platforms, are limited by power consumption, architecture, and heat dissipation design. They have strict requirements for various performance aspects of algorithms. Therefore, when considering the actual application scenarios of the model, YOLO-FLNet, which efficiently improves the model size and inference speed with an advantage in precision, is a reasonable candidate. In user-operated applications specifically, the optimization of inference speed by 30.2% significantly enhances the user experience by delivering a

Figure 9. Ablation results of different schemes on the test set



smoother frame rate. From a functional perspective, this level of improvement leaves more performance headroom to extend other algorithmic functionalities, including object tracking, based on the detector.

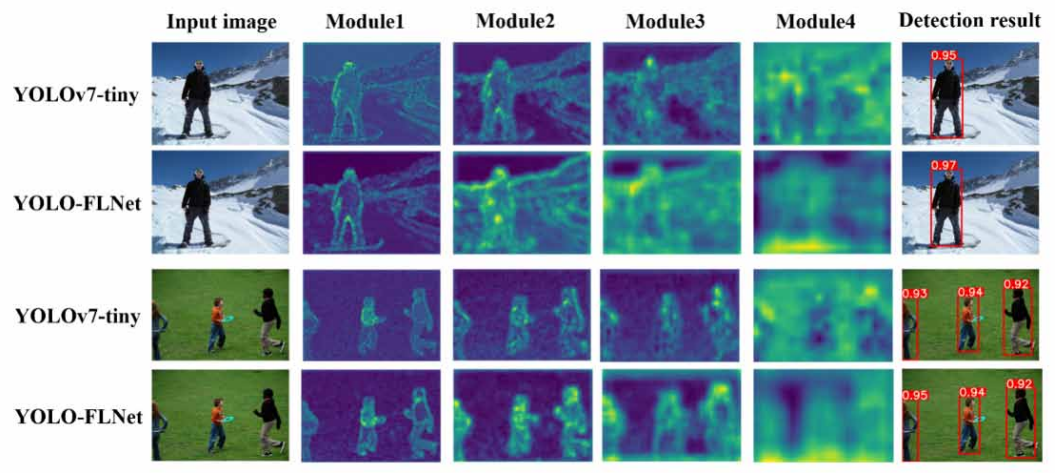
Feature Visualization

In order to visually validate the rationality of DFEM as the core module, the visualization effects of the backbone networks of YOLO-FLNet and YOLOv7-tiny are presented in Figure 10. Specifically, the visualized feature maps in each row correspond to the ELAN module and DFEM of YOLOv7-tiny and YOLO-FLNet, respectively, at corresponding positions in the backbone network. In the Module 1 stage, the boundaries between the person and the surrounding scene and the internal regions of the person are more distinct in YOLO-FLNet, indicating that the DFEM efficiently separates regions with drastic pixel value changes. In the Module 2 and Module 3 stages, the target contours are filled with more light colors (yellow), indicating that the DFEM provides additional high-level semantic information in those regions. In the Module 3 stage, near the end of the backbone, information abstracted from the deep network manifests as light colors in the target positions. At the same time, the background appears dark (blue), which is opposite to YOLOv7-tiny. Finally, both models yield similar detection results, suggesting that the DFEM can effectively extract features from input images based on the frequency of information in pixel features, with greater sensitivity towards high-frequency information.

CONCLUSION

Anticipating future trends, the continuous enhancement of edge platform performance, coupled with their cost advantages, will lead to an increasing adoption in commercial procurements and a broadened application in enterprise information systems. This, in turn, will generate a growing demand for lightweight detection algorithms to meet evolving requirements. Handling human image data captured in open scenarios occurs regularly during these systems' operational processes. However, computing power is severely limited on terminals such as smartphones, embedded devices, and other platforms, making image detection challenging. To enhance the detection performance of the model in such scenarios, this paper introduces several innovative designs: a lightweight computational module including DFEM, SSPP, and VoV-DFEM; a simplified backbone network structure; and light-FPN. Based on these innovations, the study demonstrates the effectiveness of YOLO-FLNet, which incorporates these advancements through comparative and ablation experiments. YOLO-FLNet demonstrates outstanding performance in terms of model size, speed, and precision. Simultaneously, its ability to efficiently capture personnel locations in real time enables easy functionality expansion by defining output interfaces for other applications.

Figure 10. Feature visualization of part of the network layer in YOLOv7-tiny and YOLO-FLNet



Due to limitations in experimental conditions and time constraints, this paper only validates particular types of targets in specific scenes using a limited dataset. Future studies should test more diverse datasets and categories. These datasets should contain images captured in complex scenarios with challenging foreground and background conditions, such as contamination and occlusion. In terms of applications, deploying this model on edge devices such as embedded development boards will unlock its full potential in industrial production and daily life contexts. Furthermore, by integrating additional technologies, such as facial recognition and multi-object tracking algorithms, enterprise information systems can provide enhanced practical solutions for personnel management, encompassing security control and personnel tracking analysis.

AUTHOR NOTE

The authors of this publication declare there are no competing interests.

The data used to support the findings of this study are included in the article.

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. Funding for this research was covered by the authors of the article.

REFERENCES

Al-qaness, M. A. A., Abbasi, A. A., Fan, H., Ibrahim, R. A., Alsamhi, S. H., & Hawbani, A. (2021). An improved YOLO-based road traffic monitoring system. *Computing*, *103*(2), 211–230. doi:10.1007/s00607-020-00869-8

Al-Qerem, A., Alauthman, M., Almomani, A., & Gupta, B. B. (2020). IoT transaction processing through cooperative concurrency control on fog-cloud computing environment. *Soft Computing*, *24*(8), 5695–5711. doi:10.1007/s00500-019-04220-y

Al Sobbahi, R., & Tekli, J. (2022). Comparing deep learning models for low-light natural scene image enhancement and their impact on object detection and classification: Overview, empirical evaluation, and challenges. *Signal Processing Image Communication*, *109*, 109. doi:10.1016/j.image.2022.116848

Almomani, A., Alauthman, M., Shatnawi, M. T., Alweshah, M., Alrosan, A., Alomoush, W., Gupta, B. B., Gupta, B. B., & Gupta, B. B. (2022). Phishing website detection with semantic features based on machine learning classifiers: A comparative study. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–24. doi:10.4018/IJSWIS.297032

Amjoud, A. B., & Amrouch, M. (2023). Object detection using deep learning, CNNs and vision transformers: A review. *IEEE Access : Practical Innovations, Open Solutions*, *11*, 35479–35516. doi:10.1109/ACCESS.2023.3266093

Arikumar, K. S., Kumar, A. D., Gadekallu, T. R., Prathiba, S. B., & Tamilarasi, K. (2022). Real-time 3D object detection and classification in autonomous driving environment using 3D LiDAR and camera sensors. *Electronics (Basel)*, *11*(24), 4203. doi:10.3390/electronics11244203

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal speed and accuracy of object detection.* arXiv:2004.10934 DOI: 10.48550/arXiv.2004.10934

Dan, S. J. (2022). NIR spectroscopy oranges origin identification framework based on machine learning. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–16. doi:10.4018/IJSWIS.297039

Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., & Sun, J. (2021). RepVGG: Making VGG-style ConvNets great again. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13728–13737. doi:10.1109/CVPR46437.2021.01352

Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. (2010). The PASCAL visual object classes (VOC) challenge. *International Journal of Computer Vision*, *88*(2), 303–338. doi:10.1007/s11263-009-0275-4

Gaurav, A., Gupta, B. B., & Panigrahi, P. K. (2023). A comprehensive survey on machine learning approaches for malware detection in IoT-based enterprise information system. *Enterprise Information Systems*, *17*(3), 2023764. doi:10.1080/17517575.2021.2023764

Ge, Z., Liu, S., Wang, F., Li, Z., & Sun, J. (2021). *YOLOX: Exceeding YOLO series in 2021*. arXiv:2107.08430 DOI: 10.48550/arXiv.2107.08430

Ghiasi, G., Lin, T.-Y., Pang, R., & Le, Q. V. (2019). NAS-FPN: Learning scalable feature pyramid architecture for object detection. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7029–7038. doi:10.1109/CVPR.2019.00720

Guebli, W., & Belkhir, A. (2021). Inconsistency detection-based LOD in smart homes. *International Journal on Semantic Web and Information Systems*, *17*(4), 56–75. doi:10.4018/IJSWIS.2021100104

Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., & Xu, C. (2020). GhostNet: More features from cheap operations. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1577–1586.

Huang, Z., Li, W., Xia, X.-G., Wang, H., Jie, F., & Tao, R. (2022). LO-Det: Lightweight oriented object detection in remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 60. doi:10.1109/TGRS.2021.3067470

Inam, H., Ul Islam, N., Akram, M. U., & Ullah, F. (2023). Smart and automated infrastructure management: A deep learning approach for crack detection in bridge images. *Sustainability (Basel)*, *15*(3), 1866. doi:10.3390/su15031866

Jiang, Y., Tian, Z., Wang, J., Sun, X., Lin, M., & Li, H. (2022). *GiraffeDet: A heavy-neck paradigm for object detection.* arXiv:2202.04256 DOI: 10.48550/arXiv.2202.04256

Jocher, G. (2021 Apr). *YOLOv5*. https://github.com/ultralytics/yolov5

Kang, P. (2023). Programming for high-performance computing on edge accelerators. *Mathematics*, *11*(4), 1055. doi:10.3390/math11041055

Kim, J., & Chi, M. (2021). SAFFNet: Self-attention-based feature fusion network for remote sensing few-shot scene classification. *Remote Sensing (Basel)*, *13*(13), 2532. doi:10.3390/rs13132532

Kumar, N., Poonia, V., Gupta, B. B., & Goyal, M. K. (2021). A novel framework for risk assessment and resilience of critical infrastructure towards climate change. *Technological Forecasting and Social Change*, *165*, 165. doi:10.1016/j.techfore.2020.120532

Lee, Y., Hwang, J.-W., Lee, S., Bae, Y., & Park, J. (2019). An energy and GPU-computation efficient backbone network for real-time object detection. *IEEE/Cvf Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 752–760.

Li, C., Li, L., Jiang, H., Wang, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., & Wei, X. (2022). *YOLOv6: A single-stage object detection framework for industrial applications.* arXiv:2209.02976 DOI: 10.48550/arXiv.2209.02976

Li, J. P., & Huang, H. (2021). Research study on edge computing. *IEEE 6th International Conference on Smart Cloud, Smartcloud*, 26–32. doi:10.1109/SmartCloud52277.2021.00012

Li, S., Qin, D., Wu, X., Li, J., Li, B., & Han, W. (2022). False alert detection based on deep learning and machine learning. *International Journal on Semantic Web and Information Systems*, *18*(1), 1–21. doi:10.4018/IJSWIS.297035

Li, Z., Xi, T., Zhang, G., Liu, J., & He, R. (2021). AutoDet: Pyramid network architecture search for object detection. *International Journal of Computer Vision*, *129*(6), 1087–1105. doi:10.1007/s11263-020-01415-x

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *Computer Vision—ECCV, V*(8693), 740–755.

Ma, Z., Li, Y., Huang, M., Huang, Q., Cheng, J., & Tang, S. (2023). Automated real-time detection of surface defects in manufacturing processes of aluminum alloy strip using a lightweight network architecture. *Journal of Intelligent Manufacturing*, *34*(8), 2431–2447. doi:10.1007/s10845-022-01930-3

Mehta, S., & Rastegari, M. (2021). *MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer*. arXiv:2110.02178 DOI: 10.48550/arXiv.2110.02178

Memos, V. A., Psannis, K. E., Ishibashi, Y., Kim, B.-G., & Gupta, B. B. (2018). An efficient algorithm for media-based surveillance system (EAMSuS) in IoT smart city framework. *Future Generation Computer Systems*, *83*, 619–628. doi:10.1016/j.future.2017.04.039

Peñalvo, F. J. G., Sharma, A., Chhabra, A., Singh, S., Kumar, S., Arya, V., & Gaurav, A. (2022). Mobile cloud computing and sustainable development: Opportunities, challenges, and future directions. *International Journal of Cloud Applications and Computing*, *12*(1), 1–20. doi:10.4018/IJCAC.312583

Prathiba, S. B., Raja, G., Bashir, A. K., Alzubi, A. A., & Gupta, B. (2022). SDN-assisted safety message dissemination framework for vehicular critical energy infrastructure. *IEEE Transactions on Industrial Informatics*, *18*(5), 3510–3518. doi:10.1109/TII.2021.3113130

Ranjan, R., Sharrer, K., Tsukuda, S., & Good, C. (2023). MortCam: An artificial intelligence-aided fish mortality detection and alert system for recirculating aquaculture. *Aquacultural Engineering*, *102*, 102341. doi:10.1016/j.aquaeng.2023.102341

Rudregowda, S., Manjunath, A. S., Kumar, R. S., Roopa, M., & Puneeth, S. B. (2021). Vehicle number plate detection and recognition using YOLO-V3 and OCR method. *IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC)*.

Sahoo, S. R., & Gupta, B. B. (2021). Multiple features based approach for automatic fake news detection on social networks using deep learning. *Applied Soft Computing*, *100*, 106983. doi:10.1016/j.asoc.2020.106983

Stergiou, C. L., Psannis, K. E., & Gupta, B. B. (2022). InFeMo: Flexible big data management through a federated cloud system. *ACM Transactions on Internet Technology*, *22*(2), 1–22. doi:10.1145/3426972

Sun, W., Dai, L., Zhang, X., Chang, P., & He, X. (2022). RSOD: Real-time small object detection algorithm in UAV-based traffic monitoring. *Applied Intelligence*, *52*(6), 8448–8463. doi:10.1007/s10489-021-02893-3

Tan, M., Pang, R., & Le, Q. V. (2019). *EfficientDet: Scalable and efficient object detection.* arXiv:1911.09070 DOI: 10.48550/arXiv.1911.09070

Vano, R., Lacalle, I., Sowinski, P., S-Julián, R., & Palau, C. E. (2023). Cloud-native workload orchestration at the edge: A deployment review and future directions. *Sensors (Basel)*, *23*(4), 2215. doi:10.3390/s23042215 PMID:36850813

Wang, C., & Zhong, C. (2021). Adaptive feature pyramid networks for object detection. *IEEE Access : Practical Innovations, Open Solutions*, *9*, 107024–107032. doi:10.1109/ACCESS.2021.3100369

Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2022). *YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors*. arXiv:2207.02696 DOI: 10.48550/arXiv.2207.02696

Wang, J., Huang, R., Guo, S., Li, L., Zhu, M., Yang, S., & Jiao, L. (2021). NAS-guided lightweight multiscale attention fusion network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, *59*(10), 8754–8767. doi:10.1109/TGRS.2021.3049377

Wang, T., Zhang, X., & Sun, J. (2020). *Implicit feature pyramid network for object detection.* arXiv:2012.13563 DOI: 10.48550/arXiv.2012.13563

Xu, H., Guo, M. T., Nedjah, N., Zhang, J., & Li, P. (2022). Vehicle and pedestrian detection algorithm based on lightweight YOLOv3-promote and semi-precision acceleration. *IEEE Transactions on Intelligent Transportation Systems*, *23*(10), 19760–19771. doi:10.1109/TITS.2021.3137253

Yu, Z., Wan, J., Qin, Y., Li, X., Li, S. Z., & Zhao, G. (2021). NAS-FAS: Static-dynamic central difference network search for face anti-spoofing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(9), 3005–3023. doi:10.1109/TPAMI.2020.3036338 PMID:33166249

Zaidi, S. S. A., Ansari, M. S., Aslam, A., Kanwal, N., Asghar, M., & Lee, B. (2022). A survey of modern deep learning based object detection models. *Digital Signal Processing*, *126*(11), 103514. doi:10.1016/j.dsp.2022.103514

Zeng, W., Li, H., Hu, G., & Liang, D. (2022). Lightweight dense-scale network (LDSNet) for corn leaf disease identification. *Computers and Electronics in Agriculture*, *197*, 106943. doi:10.1016/j.compag.2022.106943

Zhang, Q., Guo, Z., Zhu, Y., Vijayakumar, P., Castiglioni, A., & Gupta, B. B. (2023). A deep learning-based fast fake news detection model for cyber-physical social services. *Pattern Recognition Letters*, *168*, 31–38. doi:10.1016/j.patrec.2023.02.026

Zhao, Z., Yang, X., Zhou, Y., Sun, Q., Ge, Z., & Liu, D. (2021). Real-time detection of particleboard surface defects based on improved YOLOV5 target detection. *Scientific Reports*, *11*(1), 21777. doi:10.1038/s41598-021-01084-x PMID:34741057

Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object detection in 20 years: A survey. *Proceedings of the IEEE*, *111*(3), 257–276. doi:10.1109/JPROC.2023.3238524

YiHeng Wu is an algorithm software engineer. A postgraduate, he graduated from China University of Mining and Technology in 2022. His research interests include computer vision and refrigeration equipment design and control.

JianXin Chen is a thermal design engineer. A postgraduate, he graduated from China University of Mining and Technology in 2022. His research interests include thermal design and numerical simulation.