

Article

Not peer-reviewed version

DLCAS: A Deep Learning-Based CPR Action Standardization Method

Yongyuan Li , Mingjie Yin , Wenxiang Wu , Jiahuan Lu , [Shangdong Liu](#) , [Yimu Ji](#) *

Posted Date: 31 May 2024

doi: 10.20944/preprints202405.2102.v1

Keywords: CPR; AED; depth measurement; object detection; OpenPose; edge computing



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

DLCAS: A Deep Learning-Based CPR Action Standardization Method

Yongyuan Li ¹, Mingjie Yin ², Wenxiang Wu ², Jiahuan Lu ³, Shangdong Liu ² and Yimu Ji ^{2,*}

¹ Jiangsu Tuoyou Information Intelligent Technology Research Institute Co., Ltd., Nanjing City 210012, China; lyy573303122@163.com (Y.L.)

² School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing City 210023, China; 1023040815@njupt.edu.cn (M.Y.); 1223045119@njupt.edu.cn (W.W.); lsd@njupt.edu.cn (S.L.); jiyim@njupt.edu.cn (Y.J.)

³ School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing City 210023, China; 2023070801@njupt.edu.cn (J.L.)

* Correspondence: jiyim@njupt.edu.cn; Tel.: +86-189-189-6436

Abstract: In emergency situations, ensuring standardized Cardiopulmonary Resuscitation (CPR) actions is crucial. However, current Automated External Defibrillators (AEDs) lack methods to determine whether CPR actions are performed correctly, leading to inconsistent CPR quality. To address this issue, we introduce a novel method called Deep Learning-based CPR Action Standardization (DLCAS). This method involves three parts. First, it detects correct posture using OpenPose to recognize skeletal points. Second, it identifies a marker wristband with our CPR-Detection algorithm and measures compression depth, count and frequency using a depth algorithm. Finally, we optimize the algorithm for edge devices to enhance real-time processing speed. Extensive experiments on our custom dataset have shown that the CPR-Detection algorithm achieves a mAP_{0.5} of 97.04%, while reducing parameters to 0.20M and FLOPs to 132.15K. In a complete CPR operation procedure, the depth measurement solution achieves an accuracy of 90% with a margin of error less than 1 cm, while the count and frequency measurements achieve 98% accuracy with a margin of error less than 2 counts. Our method meets the real-time requirements in medical scenarios, and the processing speed on edge devices have increased from 8fps to 25fps.

Keywords: CPR; AED; depth measurement; object detection; OpenPose; edge computing

1. Introduction

Out-of-hospital cardiac arrest (OHCA) is a critical medical emergency with a substantial impact on public health, exhibiting annual incidence rates of approximately 55 per 100,000 people in North America and 59 per 100,000 in Asia. Without timely intervention, OHCA can lead to irreversible death within 10 minutes [1]. Studies have demonstrated that CPR and AED defibrillation performed by nearby volunteers or citizens significantly improve survival rates [1–3]. Standard CPR procedures are known to enhance survival outcomes in cardiac arrest patients [3]. However, the dissemination of CPR skills remains limited in many countries, primarily relying on mannequins and instructors, leading to high costs and inefficiencies. Traditional AED devices also lack the capability to prevent harm caused by improper operation [4,5].

Research has highlighted the limitations of traditional CPR training methods and the potential of AI to transform CPR training and execution. Wang et al. developed a vision-based system for recognizing incorrect actions and assessing skills in CPR training, marking progress in fine-grained medical behavior analysis. However, this approach is restricted to training scenarios [5]. Furthermore, systematic reviews have shown that while technologies like virtual reality (VR) and augmented reality (AR) are being explored to enhance CPR training, these innovations are primarily focused on educational settings and have not yet been widely integrated into real-time emergency applications [6,7]. Additionally, mainstream CPR methods have not incorporated AI assistance, as most advancements in CPR technologies focus on mechanical devices and VR/AR for training purposes, rather than real-time AI-based interventions during actual emergencies [8,9]. This paper proposes the first application of pose estimation and object detection algorithms on AEDs to assist in real-time CPR action standardization, extending the application to actual emergency rescue. This innovative approach

addresses the gaps in existing CPR methods, which have not utilized AI assistance for real-time interventions, thereby improving the accuracy and effectiveness of life-saving measures during OHCA events. By integrating AI technologies into AED devices, we aim to provide immediate feedback and corrective actions during CPR, potentially increasing survival rates and reducing the risks associated with improper CPR techniques. This approach represents a significant advancement over traditional methods, which lack the capability to dynamically adapt and guide rescuers in real-time.

To enhance real-time medical interventions, advanced pose estimation techniques like OpenPose are highly beneficial. Developed by the Perceptual Computing Lab at Carnegie Mellon University, OpenPose is a pioneering open-source library for real-time multi-person pose estimation. It detects human body, hand, facial, and foot keypoints simultaneously [10]. Initially, OpenPose used a dual-branch CNN architecture to produce confidence maps and Part Affinity Fields (PAFs) for associating body parts into a coherent skeletal structure. Subsequent improvements focused on refining PAFs, integrating foot keypoint detection, and introducing multi-stage CNNs for iterative prediction refinement [11,12]. Supported by continuous research and updates, OpenPose remains robust and efficient for edge computing and real-time applications [13], solidifying its status as a leading tool in diverse and complex scenarios.

In addition, deploying neural network models on AED edge devices to recognize and standardize rescuers' CPR actions can effectively improve the survival rate of cardiac arrest patients. However, deploying neural network models on embedded systems faces challenges such as high weight, insufficient computational power, and low running speed [14]. Most early lightweight object detection models were based on MobileNet-SSD (Single Shot Multibox Detector) [15]. Installing these models on some high-end smartphones can achieve sufficiently high running speeds [16]. However, due to insufficient ARM cores for running neural networks, model execution speed is slow on low-cost advanced RISC machine (ARM) devices [17].

In recent years, various lightweight object detection networks have been proposed and widely applied in traffic management [18–21], fire warning systems [22] anomaly detection [23–25], and facial recognition [26–28]. Redmon et al. [29] introduced an end-to-end object detection model based on deep learning, using Darknet53 as the backbone network. This network, built on Darknet-19 and residual modules, is a novel feature extraction network. In this model, they applied the k-means clustering algorithm to determine anchor boxes and used multi-label classification to predict class probabilities, while employing a feature pyramid network to predict bounding boxes at three different scales. Wong et al. [30] combined human-machine collaboration design strategies with the Yolo architecture to develop Yolo Nano, a compact network tailored for embedded object detection, featuring custom module-level macro and micro architectures. The proposed Yolo Nano model is approximately 4.0MB in size. Hu et al. [31] improved the Yolov3-tiny network by replacing its convolutional layers with depthwise distributed convolutions with squeeze and excitation blocks, and introduced Micro-Yolo, a progressive channel pruning algorithm designed to reduce the number of parameters and optimize detection performance. This significantly reduces the number of parameters and computational cost while maintaining detection performance. Lyu [32] proposed NanoDet, a single-stage anchor-free object detection model in the FCOS style, using generalized focal loss for classification and regression. NanoDet-Plus introduced a new label assignment strategy, equipped with a simple assignment guidance module (AGM) and a dynamic soft label assigner (DSL), aiming to optimize the training efficiency of lightweight models. It also introduced a lightweight feature pyramid named Ghost-PAN to enhance multi-layer feature fusion. These improvements increased NanoDet's detection accuracy on the COCO dataset by 7% mAP (mean Average Precision). Ge et al. [33] modified the Yolo detector to an anchor-free mode and introduced a decoupled head and leading label assignment strategy SimOTA, significantly enhancing model performance. For example, Yolo Nano achieved 25.3% AP on the COCO dataset with only 0.91M parameters and 1.08G FLOPs, surpassing NanoDet by 1.8%. Similarly, the improved Yolov3 AP increased to 47.3%, exceeding the current best practice by 3.0%. Yolov5 Lite [34] added shuffle channels and pruned the head channels on the original Yolov5,

optimizing inference speed under the same hardware conditions while maintaining high accuracy. Dog-qiuqiu [35] developed the Yolo-Fastest series, emphasizing single-core real-time inference performance and maintaining real-time performance while reducing CPU usage. Yolo-FastestV2 adopted the lightweight ShufflenetV2 backbone network, decoupled the detection head, reduced parameters, and improved the anchor matching mechanism. On this basis, Dog-qiuqiu [36] further proposed FastestDet, which simplified the model to a single detection head, transitioned from anchor-based to anchor-free, and increased the number of candidate objects across grids, adapting to resource-constrained ARM platforms. However, for our dataset, FastestDet underperformed mainly due to its single detection head design limiting the utilization of features with different receptive fields and lacking sufficient feature fusion, resulting in insufficient accuracy in locating small objects.

This paper proposes a standard detection method for CPR actions based on AED, utilizing skeletal points to assist in posture estimation. The method identifies the rescuer's marked wristband to measure compression depth, frequency, and count. Considering the limitations of object detection networks on edge devices, we developed the CPRDetection algorithm based on Yolo-FastestV2. This algorithm not only improves detection accuracy but also simplifies the model structure. Building on this algorithm, we designed a novel compression depth calculation method, which maps actual depth by analyzing the wristband's displacement. We also optimized the network to enhance speed and accuracy on edge devices, ensuring precise compression depth measurement to protect the safety of the rescued individual. Furthermore, we optimized the computation for edge devices. The main contributions of this paper include:

- (1) We introduced a novel method called Deep Learning-based CPR Action Standardization (DLCAS) and developed a custom CPR action dataset. Additionally, we incorporated OpenPose for pose estimation of rescuers.
- (2) We proposed an object detection model called CPR-Detection and introduced various methods to optimize its structure. Based on this, we developed a new method for measuring compression depth by analyzing wristband displacement data.
- (3) An optimized deployment method for Automated External Defibrillator (AED) edge devices is proposed. This method addresses the issues of long model inference time and low accuracy that exist in current edge device deployments of deep learning algorithms.
- (4) Conducting extensive experimental validation to confirm the effectiveness of the improved algorithm and the feasibility of the compression depth measurement scheme.

The following is an outline of this study. Section 2 discusses the modules and algorithms we use. In Section 3, the introduction of the data set and the introduction of data preprocessing are carried out. Section 4 provides the results of the experiments. Section 5 discusses our conclusions.

2. Methods

As shown in Figure 1, the overall workflow of this study is divided into three parts. The first part involves the experimental preparation phase, which includes dataset collection, image preprocessing and augmentation, dataset splitting, training, and then testing the trained model to obtain performance metrics. The second part presents the flowchart of the DLCAS, covering pose estimation, object detection network, and depth measurement, ultimately yielding depth, compression count, and frequency. The third part describes the model's inference and application. The captured images, processed through the optimized AED edge devices, eventually become CPR images with easily assessable metrics.

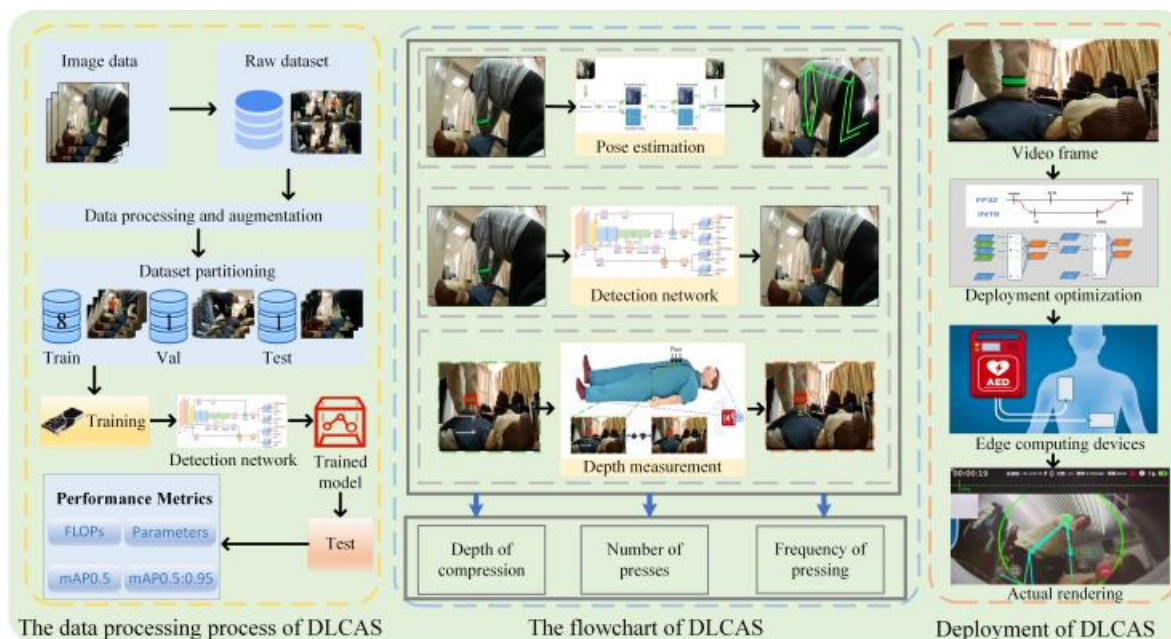


Figure 1. The overall working flow chart.

In this section, we first introduce the principles of OpenPose, followed by the design details of CPR-Detection. Next, we explain the depth measurement scheme based on object detection algorithms. Finally, we discuss the optimization of computational methods for edge devices.

2.1. OpenPose

In edge computing devices for medical posture assessment, processing speed and real-time performance are crucial. Therefore, we chose OpenPose for skeletal point detection due to its efficiency and accuracy. OpenPose employs a dual-branch architecture that generates confidence maps for body part detection and Part Affinity Fields (PAFs) to assemble these parts into a coherent skeletal structure. This method enables precise and real-time posture analysis, which is essential for medical applications. Traditional pose estimation algorithms often involve complex computations that delay processing. OpenPose optimizes this process by focusing on key points and their connections, significantly reducing computational load and improving speed. It detects body parts independently before associating them, enhancing accuracy and efficiency by minimizing redundant computations. Overall, OpenPose allows for accurate and swift identification and assessment of human postures, making it ideal for real-time medical applications. Its efficient processing and reduced computational overhead make it suitable for deployment in edge computing devices used in emergency medical care, ensuring both reliability and speed in critical situations.

As shown in Figure 2, the workflow of OpenPose starts with feature extraction through a backbone network. These features pass through Stage 0, producing keypoint heatmaps and PAFs. Keypoint heatmaps indicate confidence scores for the presence of keypoints at each location, while PAFs encode the associations between pairs of keypoints, capturing spatial relationships between different body parts. These outputs are refined in subsequent stages, iteratively improving accuracy. Finally, the keypoint heatmaps and PAFs are processed to generate the final skeletal structure, combining keypoints according to the PAFs to form a coherent and accurate representation of the human pose. This method ensures precise and real-time posture analysis, making it highly effective for applications in medical posture assessment, particularly in edge computing devices used in emergency medical care, ensuring both reliability and speed in critical situations. [11]

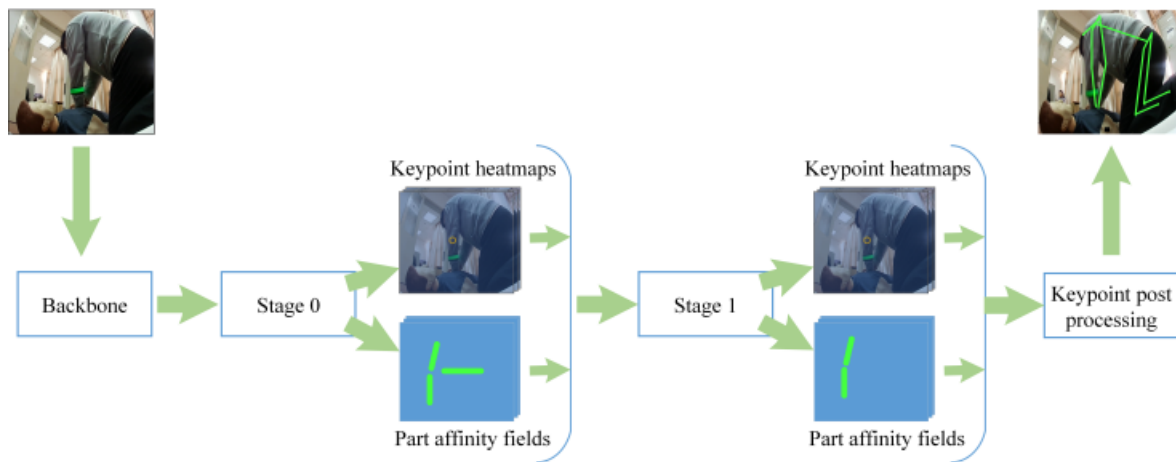


Figure 2. Overall framework of OpenPose.

2.2. CPR-Detection

In this study, we provide a detailed explanation of CPR-Detection. As illustrated in Figure 3, the model consists of three components: the backbone network ShuffleNetV2, the STD-FPN feature fusion module, and the detection head. The STD-FPN feature fusion module incorporates the MLCA attention mechanism, and the detection head integrates PConv position-enhanced convolution.

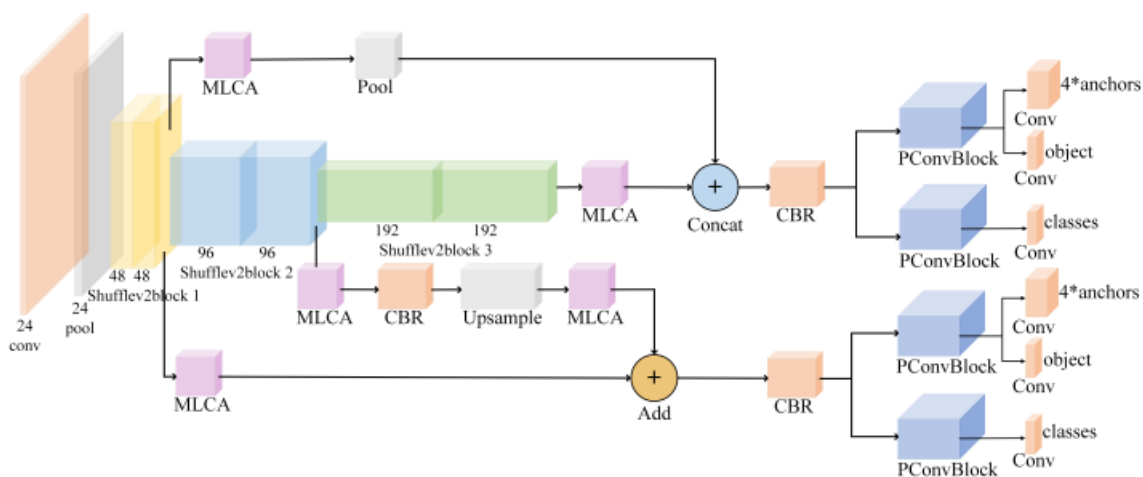


Figure 3. Overall framework of CPR-Detection.

2.2.1. PConv

In edge computing devices for medical emergency care, we need to prioritize processing speed due to performance and real-time processing requirements. Therefore, we chose Partial Convolution (PConv) to replace Depthwise Separable Convolution (DWSCConv) in Yolo-FastestV2. PConv offers higher efficiency while maintaining performance, meeting the needs for real-time processing [37].

As shown in Figure 4a, DWSCConv works by first performing depthwise convolution on the input feature map, grouping by channels, and then using 1x1 convolution to integrate all channel information. However, this depthwise convolution can lead to computational redundancy in practical applications. The principle of PConv, illustrated in Figure 4b, involves performing regular convolution operations on a portion of the input channels while leaving the other channels unchanged. This design significantly reduces computational load and memory access requirements because it processes only a subset of feature channels. PConv only performs convolution on a specific proportion of the input features, resulting in lower FLOPs compared to DWSCConv, thereby reducing computational overhead

and improving model efficiency. In summary, PConv enhances the network's feature representation capability by focusing on crucial spatial information without sacrificing detection performance.

This strategy not only improves the network's processing speed but also enhances the extraction and focus on key feature channels, making it essential for real-time object detection systems. Additionally, by reducing redundant computations, the application of PConv lowers model complexity and increases model generalization, ensuring robustness and efficiency in complex medical emergency scenarios. Therefore, PConv is an ideal convolution method for medical emergency devices, enabling real-time object detection while ensuring reliability and efficiency on edge computing devices.

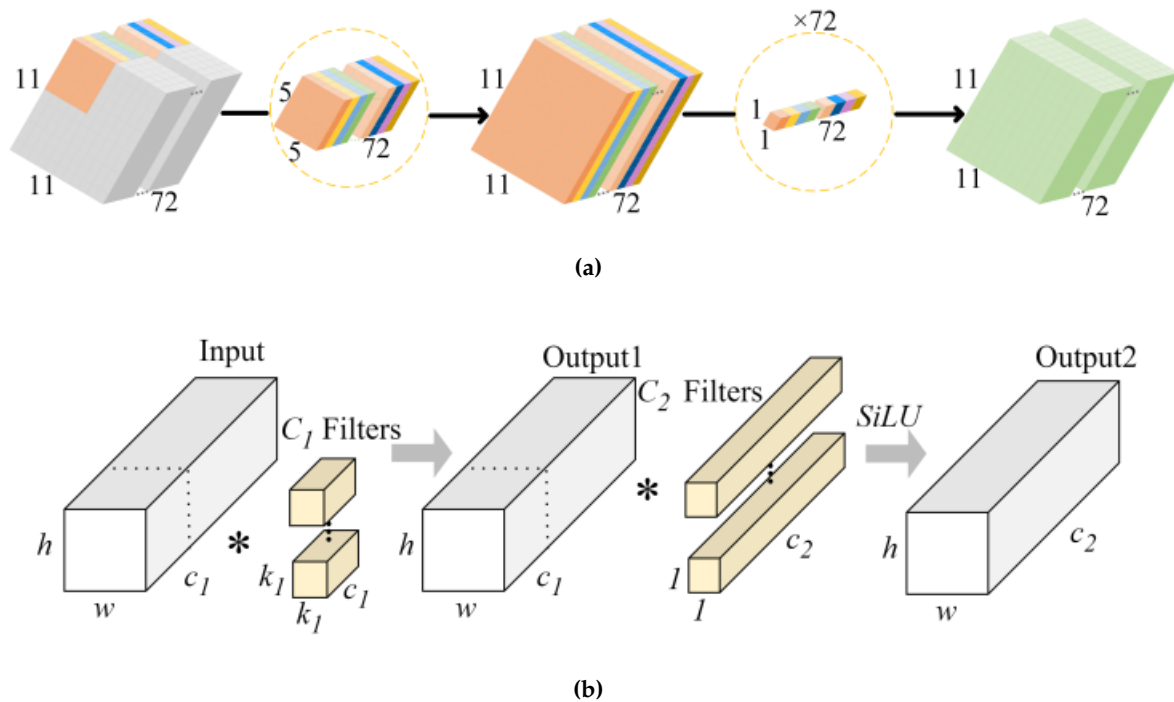


Figure 4. (a) DWSCConv. (b) PConv.

2.2.2. MLCA

In emergency medical scenarios, complex backgrounds can interfere with the effective detection of wristbands. To address this, we introduced the Mixed Local Channel Attention (MLCA) module to enhance the model's performance in processing channel-level and spatial-level information. As illustrated in Figure 5, MLCA combines local and global context information to improve the network's feature representation capabilities. This focus on critical features enhances both the accuracy and efficiency of target detection [38].

The core of MLCA lies in its ability to process and integrate both local and global feature information simultaneously. Specifically, MLCA first performs two types of pooling operations on the input feature vector: local pooling, which captures fine-grained spatial details, and global pooling, which extracts broader contextual information. These pooled features are then sent to separate branches for detailed analysis. Each branch output is further processed by convolutional layers to extract cross-channel interaction information. Finally, the pooled features are restored to their original resolution through an unpooling operation and fused using an addition operation, achieving comprehensive attention modulation. Compared to traditional attention mechanisms such as SENet [39] or CBAM [40], MLCA offers the advantage of considering both global dependencies and local feature sensitivity. This is particularly important for accurately locating small-sized targets. Moreover, the design of MLCA emphasizes computational efficiency. Despite introducing a complex context fusion strategy, its implementation ensures that it does not significantly increase the network's computational burden, making it well-suited for integration into resource-constrained edge devices. In performance evaluations,

MLCA demonstrated significant advantages. Experimental results showed that models incorporating MLCA achieved a notable percentage increase in mAP0.5 compared to the original models while maintaining low computational complexity.

Overall, MLCA is an efficient and practical attention module ideal for target detection tasks in emergency medical scenarios requiring high accuracy and real-time processing.

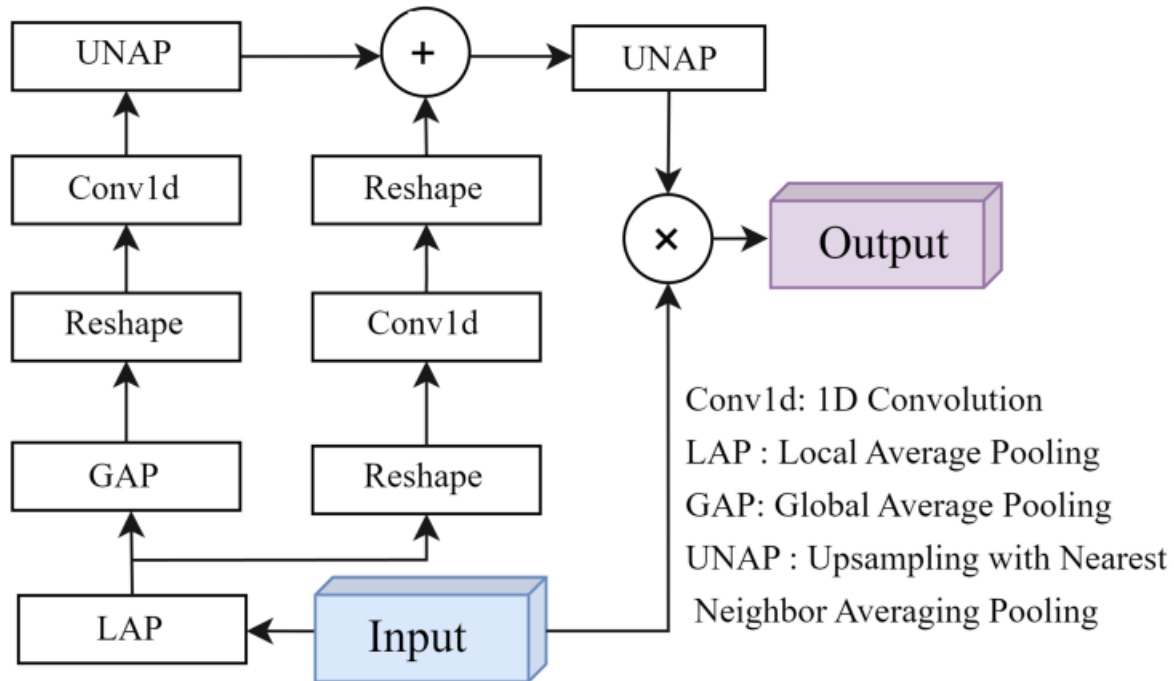


Figure 5. Mixed Local Channel Attention (MLCA).

2.2.3. STD-FPN

In recent years, ShuffleNetV2 [41] has emerged as a leading network for lightweight feature extraction, incorporating innovative channel split and channel shuffle designs that significantly reduce computational load and the number of parameters while maintaining high accuracy. Compared to its predecessor, ShuffleNetV1 [42], ShuffleNetV2 demonstrates greater efficiency and scalability, with substantial innovations and improvements in its structural design and complexity management. The network is divided into three main stages, each containing multiple ShuffleV2Blocks. Data first passes through an initial convolution layer and a max pooling layer, progressively moving through the stages, and ultimately outputs feature maps of three different dimensions. The entire network optimizes feature extraction performance by minimizing memory access.

As shown in Figure 6a, the FPN structure of Yolo-FastestV2 utilizes the feature map from the third ShuffleV2Block in ShuffleNetV2, combined with 1×1 convolution to predict large objects. These feature maps are then upsampled and fused with the feature maps from the second ShuffleV2Block to predict smaller objects. However, Yolo-FastestV2's FPN only uses two layers of shallow feature maps, limiting the acquisition of rich positional information and affecting the semantic information extraction and precise localization of small objects. Considering that AED devices are typically placed within 50cm to 75cm from the patient, and the wristband is a small-scale target, we propose an improved FPN structure named STD-FPN (see Figure 6b), which effectively merges shallow and deep feature maps from ShuffleV2Block, focusing on small object detection. Each output from the ShuffleV2Block is defined as S_i , $i \in [1, 3]$. After processing through the MLCA module, it becomes C_i . First, C_1 is globally pooled to reduce its size by a factor of four to get C'_1 , which is then concatenated with C_3 . This concatenated feature undergoes Convolution-BatchNormalization-ReLU(CBR), forming the input for the first detection head. The second detection head, designed for small objects, processes C_2 through

CBR operations to match the channel count of C_1 and then upsamples C_2' along all dimensions using a specified scaling factor. C_2' is element-wise added to C_1 , followed by the CBR operation.

After each feature fusion step, a 1×1 convolution is applied. During the entire model training process, convolution helps extract effective features from previous feature maps and reduces the impact of noise. By using additive feature fusion, shallow and deep features are fully integrated, producing fused feature maps rich in object positional information, thus enhancing the original model's localization capability.

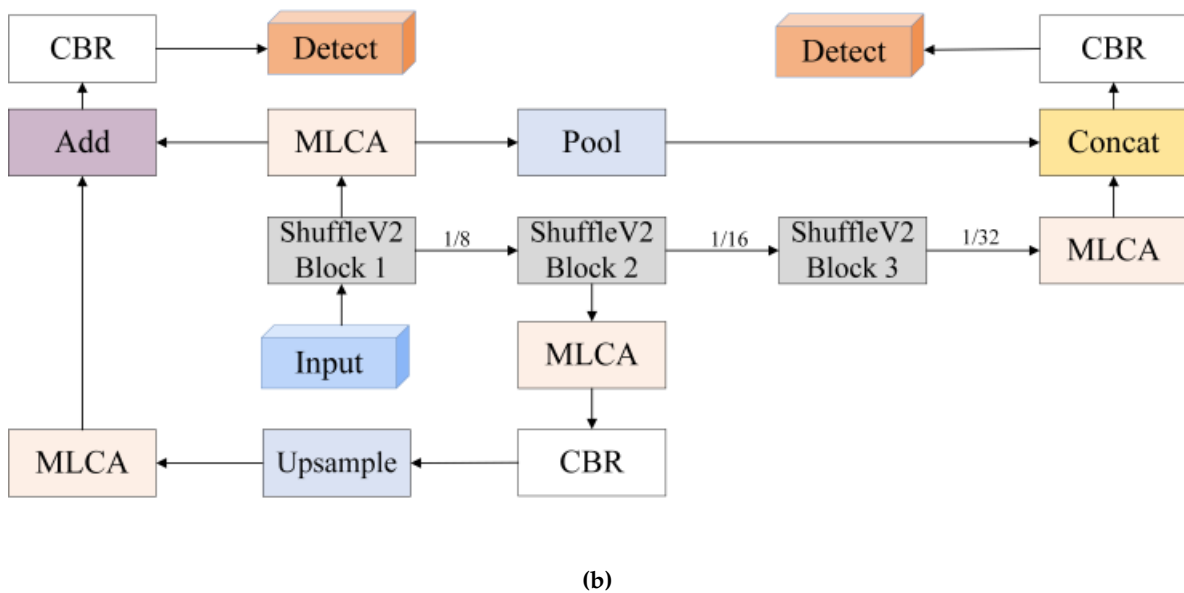
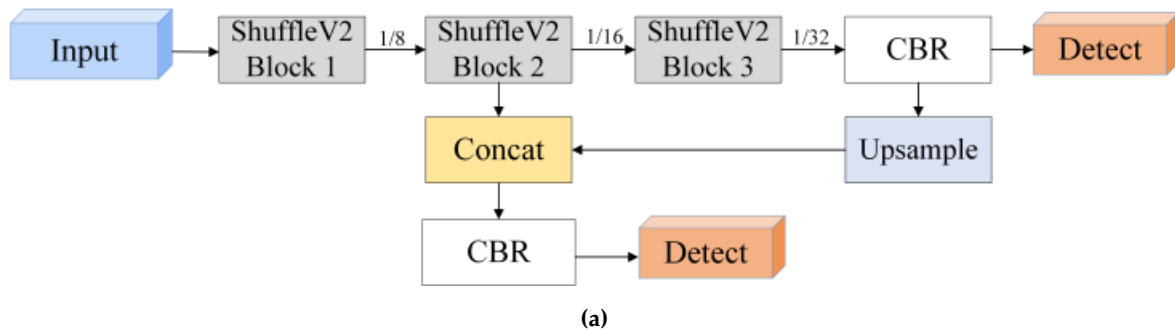


Figure 6. (a) The FPN of Yolo-FastestV2. (b) Small Target Detection-Feature Pyramid Network.

2.3. Depth Measurement Method

Image processing often involves four coordinate systems: the world coordinate system, the camera coordinate system, the image coordinate system, and the pixel coordinate system. Typically, the transformation process starts from the world coordinate system, passes through the camera coordinate system and the image coordinate system, and finally reaches the pixel coordinate system [43]. Assume world coordinate point $P_w = (x_w, y_w, z_w)^T$, camera coordinate point $P_c = (x_c, y_c, z_c)^T$, image coordinate point $m = (x_p, y_p, 1)^T$, and pixel coordinate point $P_{ix} = (u, v, 1)^T$. From the world coordinate point $P_w = (x_w, y_w, z_w)^T$ to the camera coordinate point $P_c = (x_c, y_c, z_c)^T$, it is transformed by formula (1).

$$\begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & T \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (1)$$

Among them, the orthogonal rotation matrix $R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$ and the translation matrix $T = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T$. Assume the center O of the projective transformation as the origin of the camera coordinate system, and the distance from this point to the imaging plane, the focal length is f . According to the principle of similar triangles, formula (2) can be obtained to transform from the camera coordinate point $P_c = (x_c, y_c, z_c)^T$ to the image coordinate point $m = (x_p, y_p, 1)^T$:

$$z_c m = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} P_c \quad (2)$$

Assume that the length and width of a pixel are d_x, d_y respectively. Pixel coordinate point $P_{ix} = (u, v, 1)^T$, then

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1/d_x & 0 & 0 \\ 0 & 1/d_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} \quad (3)$$

In summary, combining formulas (1) (2) (3), the transformation matrix K from the camera coordinate point $P_c = (x_c, y_c, z_c)^T$ to the pixel coordinate point $P_{ix} = (u, v, 1)^T$ can be obtained:

$$K = \begin{bmatrix} 1/d_x & 0 & 0 \\ 0 & 1/d_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

Among them, $f_x = f/d_x, f_y = f/d_y$, are called the scale factors of the camera in the u -axis and v -axis directions:

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \quad (5)$$

Equation (5) represents the transformation from world coordinates to pixel coordinates. The above explanation covers the principles of camera imaging. Building on this foundation, we propose a new depth measurement method.

In conventional monocular camera distance measurement, directly measuring depth is challenging because it lacks stereoscopic information. To address this issue, this study employs an innovative approach, as shown in Figure 7, using a fixed-length marker wristband as a depth calibration tool. By applying the principles of camera imaging, we can accurately calculate the distance between the camera and the marker wristband. Ultimately, by comparing the known length of the marker with the image captured by the camera, we achieve precise mapping calculations of real-world compression depth.

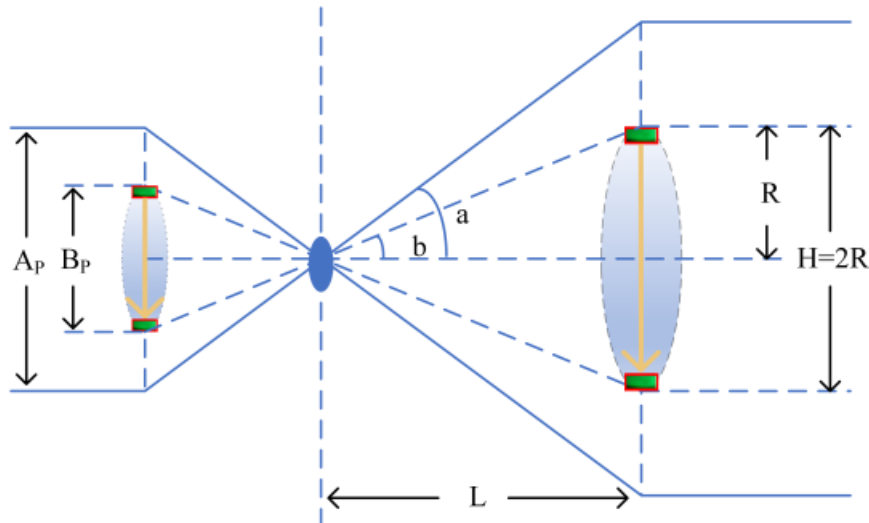


Figure 7. Depth ranging schematic.

During the execution of the program, it is necessary to read the detection frame displacement, denoted by B_0 , at the current window resolution. The resolution conversion function f : converts the detection frame displacement B_0 at the current window resolution to the pixel height B_p at the ideal camera resolution, i.e.:

$$B_p = f(B_0) \quad (6)$$

From Figure 6, B_p is the vertical displacement of the marker captured by the camera, L' is the focal length of the camera, L is the horizontal distance between the marker and the camera, R is half of the vertical displacement of the marker, H is the vertical displacement of the marker, and the following equation is obtained:

$$\begin{cases} \tan(a) = \frac{A_p}{2L'} \\ \tan(b) = \frac{B_p}{2L'} \\ \tan(b) = \frac{R}{L} \end{cases} \quad (7)$$

It is obtained from Equation (7): It is obtained from Equation (7):

$$\frac{\tan(a)}{\tan(b)} = \frac{A_p}{B_p} \quad (8)$$

Substituting $\tan(b) = \frac{R}{L}$ from Equation (8) yields:

$$L = \frac{R \times A_p}{B_p \times \tan(a)} \quad (9)$$

In summary:

$$H = 2\tan(b) \times L = \frac{B_p \times L}{L'} \quad (10)$$

H is the realistic depth of compression that we seek.

2.4. Edge Device Algorithm Optimization

Given the limited computational power of existing edge devices, a special optimization method is needed to enhance the timeliness of CPR action recognition, which requires high accuracy and real-time

processing. As illustrated in Figure 8, the deep learning algorithm model is first converted into weights compatible with the corresponding NPU. During this conversion process, MMSE algorithms and lossless pruning are employed to obtain more lightweight weights. Next, a multithreading scheme is designed. Two threads on the CPU handle the algorithm's pre-processing and post-processing, while one thread on the NPU handles the inference phase. The RGA method is applied to image processing during both the pre- and post-processing stages. Finally, NEON instructions are used during the algorithm's compilation phase.

By using the MMSE algorithm for weight quantization and applying RGA and NEON acceleration, the algorithm's size is reduced, computational overhead is minimized, and inference speed is increased. Lossless pruning during model quantization effectively prevents accuracy degradation. The multithreading design enables asynchronous processing between the CPU and NPU, significantly improving the model's performance on edge devices.

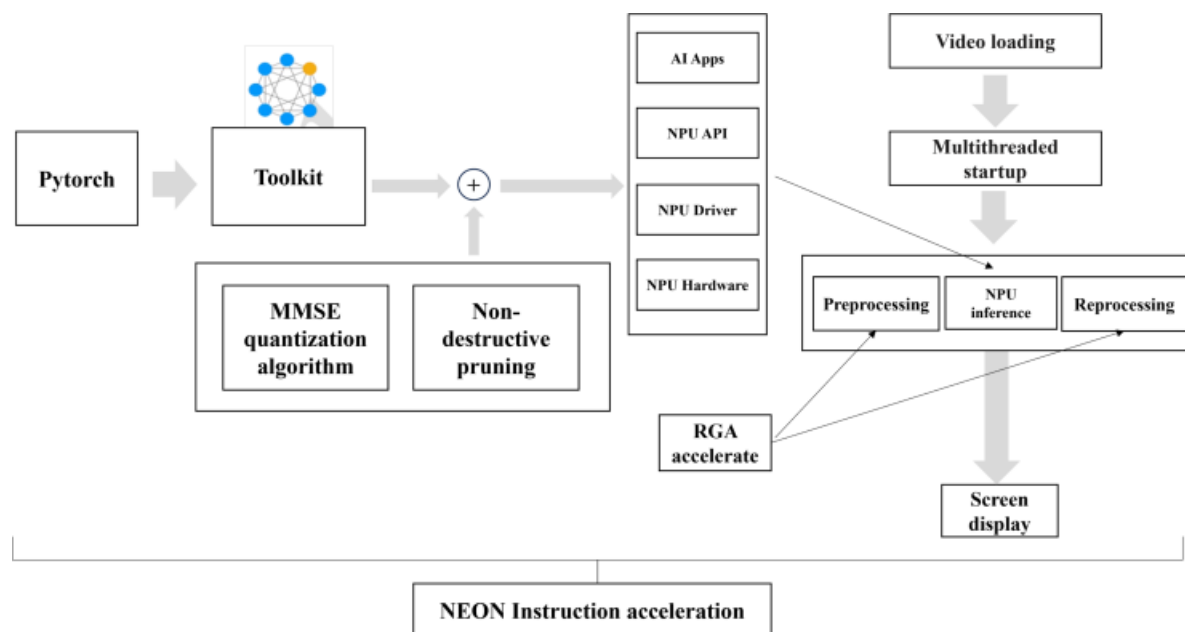


Figure 8. Edge device computing optimization flow chart.

3. Results

3.1. Datasets

The dataset used in this study consists of video frames of CPR actions captured by student volunteers from Nanjing University of Posts and Telecommunications in various scenarios. These videos encompass different indoor and outdoor environments and lighting conditions. The environments include objects with colors similar to the marker wristbands. The volunteer group comprised students with and without first aid knowledge to ensure data diversity and broad applicability. Videos were selected based on clarity, shooting angle, and visibility of the marker wristbands. Videos with low image quality due to blurriness, overexposure, or unclear markers were excluded to maintain high quality and consistency in the dataset. The original dataset contained 1,479 images, which were augmented to 8,874 images. To ensure the model's robustness and generalizability, the dataset was divided into training, testing, and validation sets in an 8:1:1 ratio, comprising 7,081, 897, and 896 images, respectively. The experiments focused on a single object type—the marker wristband—ensuring the model specifically targeted this object.

3.2. Data Pre-Processing

The marker wristband used in the experiments is 33.40 cm long, 3.80 cm wide, and fluorescent green. The experiments were conducted on an NVIDIA GEFORCE RTX 6000 GPU with 24 GB of memory to ensure efficient training. The model was trained without using pre-trained weights. Image processing and data augmentation techniques were employed to reduce overfitting and improve recognition accuracy. The training parameters were set as follows: image resolution of 352×352, 300 epochs, a learning rate of 0.001, and a batch size of 512. To ensure annotation accuracy and consistency, professionally trained volunteers used the LabelMe tool to annotate images, accurately marking each wristband within the boundary boxes to avoid unnecessary noise. During the training phase, we implemented basic image quality control measures, including checking image clarity, brightness, and contrast. All images were cropped and scaled to a uniform 352×352 pixels to standardize the input data format. To enhance the model's generalization ability and reduce overfitting, various data augmentation techniques were applied. These included random rotation, horizontal and vertical flipping, random scaling, and slight color transformations (such as hue and saturation adjustments) to simulate different lighting conditions. These steps ensured the dataset's quality, making the model more robust and reliable. The training process of the dataset is illustrated in Figure 9a, showing batch 0, while Figure 9b shows the testing of batch 0 using the dataset labels.



Figure 9. (a) Train batch 0 with datasets. (b) Test batch 0 labels with Datasets.

True Positives (TP) refer to the number of instances where the actual condition is "yes" and the model also predicts "yes." True Negatives (TN) refer to the number of instances where the actual condition is "no" and the model correctly predicts "no." False Positives (FP) occur when the model incorrectly predicts "yes" for an actual "no" scenario, leading to false alarms. Conversely, False Negatives (FN) occur when the model incorrectly predicts "no" for an actual "yes" scenario [44]. Precision and recall are calculated using Equations (11) and (12) respectively [45,46].

$$\text{Precision}(P) = \frac{TP}{TP + FP} \quad (11)$$

$$\text{Recall}(R) = \frac{TP}{TP + FN} \quad (12)$$

4. Discussion

This study used video frames of rescuers wearing marker wristbands during CPR as the dataset. First, we trained the model. Then, through ablation experiments, we evaluated the effects of PConv, MLCA, and STD-FPN to determine how these improvements impacted the original network. Next, we compared the improved model with other mainstream lightweight object detection models to validate the effectiveness of our approach. Finally, the optimized model was deployed on hardware and achieved real-time and accurate detection of compression depth, count, and frequency on AED

edge devices with low computational power. This allows rescuers to adjust their actions promptly, ensuring proper CPR performance.

4.1. OpenPose for CPR Recognition

During the process of performing CPR with an AED device, some errors may be difficult to detect through direct observation by a physician. Therefore, it is necessary to use OpenPose to draw skeletal points. As shown in Figure ??, three common incorrect CPR scenarios are identified: obscured arm movements due to dark clothing, kneeling on one knee, and non-vertical compressions. In the first scenario, dark clothing reduces the contrast with the background, making it difficult to clearly distinguish the edges of the arms. This issue is exacerbated in low-light conditions, making arm movements even more blurred and harder to identify. In the second scenario, kneeling on one knee causes the rescuer's body to be unstable, affecting the stability and effectiveness of the compressions. In the third scenario, non-vertical compressions cause the force to be dispersed, preventing it from being effectively concentrated on the patient's chest, thereby affecting the depth and effectiveness of the compressions. These issues can all be addressed using OpenPose. After posture recognition, physicians can remotely provide voice reminders, allowing for the immediate correction of these otherwise difficult-to-detect incorrect postures.

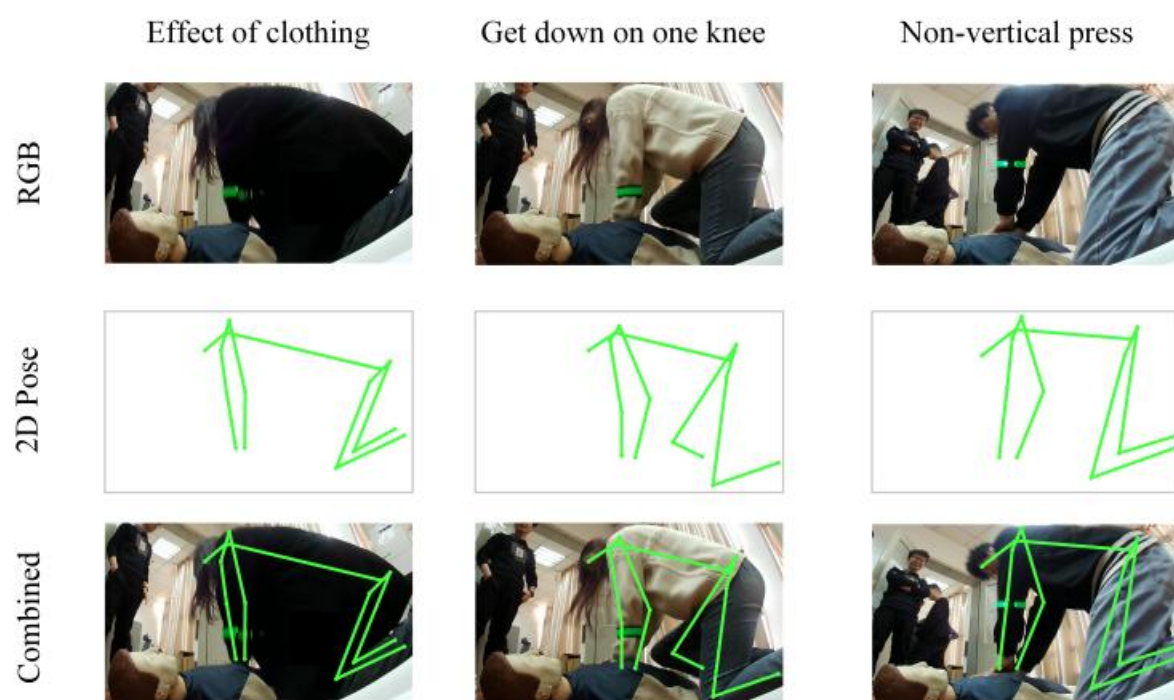


Figure 10. Common incorrect posture images (including RGB, 2D Pose, Combined).

4.2. Ablation Experiment

CPR-Detection is an improved object detection model designed to optimize recognition accuracy and speed. In medical CPR scenarios, due to the limited computational power of edge devices, smaller image inputs (352x352 pixels) are typically used to achieve the highest possible mAP0.5. To assess the specific impact of the new method on mAP0.5, ablation experiments were conducted on Yolo-FastestV2. The study independently and jointly tested the effects of the PConv, MLCA, and STD-FPN modules on model performance. The results, as shown in Table 1, clearly demonstrate that these modules, whether applied alone or in combination, enhance the model's mAP0.5: Introducing PConv improved mAP0.5 by 0.44%, optimizing the extraction and representation of positional features. Using MLCA increased mAP0.5 by 0.44%, effectively enhancing the model's ability to process channel-level and spatial-level

information. Applying the STD-FPN structure resulted in a 0.11% mAP0.5 improvement, optimizing feature fusion and positional enhancement. Combining PConv and MLCA boosted mAP0.5 to 96.87%, achieving a 0.83% increase. The combination of PConv and STD-FPN raised mAP0.5 by 0.95%, better integrating local and global features. The combined use of all three modules increased mAP0.5 by 1.00%, slightly increasing FLOPs but reducing the number of parameters.

These improvements significantly enhance the model's ability to recognize small targets in CPR scenarios, ensuring higher accuracy while maintaining real-time detection, and demonstrating the superiority of the CPR-Detection model. The combined use of the three modules fully leverages their unique advantages, enabling the model to adapt flexibly to different input sizes and application scenarios, providing an ideal object detection solution for medical emergency scenarios that demand high accuracy and speed.

Table 1. Validation of the Proposed Method on Yolo-FastestV2

Index	BASE	PConv	MLCA	STD-FPN	FLOPs	Parameters	mAP0.5	mAP0.5:0.95
1	✓	✗	✗	✗	114.12K	238.50K	96.04	72.55
2	✓	✓	✗	✗	105.98K	213.30K	96.48	73.89
3	✓	✗	✓	✗	114.36K	238.52K	96.48	75.09
4	✓	✗	✗	✓	159.53K	229.38K	96.15	71.12
5	✓	✓	✓	✗	131.83K	204.18K	96.99	75.16
6	✓	✓	✗	✓	106.22K	213.32K	96.87	76.57
7	✓	✓	✓	✓	132.15K	204.20K	97.04	75.13

4.3. Compared With State-of-the-Art Models

To evaluate the impact of the proposed method on the model's feature extraction capabilities, the CPR-Detection model was compared with six state-of-the-art lightweight object detection models, including FastestDet and Yolo-FastestV2 based on the YoloV5 architecture, as well as other official lightweight models. This comparison aimed to demonstrate the effectiveness of the new method in improving model performance. Compared to Yolo-FastestV2, the improved CPR-Detection model significantly enhanced feature extraction capabilities. Table II presents a quantitative comparison of these models in terms of FLOPs, parameter count, mAP0.5, and mAP0.5:0.95.

As shown in Table 2, the comparison results of CPR-Detection with other models in terms of mAP0.5 are as follows: CPR-Detection's mAP0.5 improved by 1.02% compared to YoloV7-Tiny; by 6.84% compared to NanoDet-m; by 11.46% compared to FastestDet; and by 1.00% compared to Yolo-FastestV2. Although CPR-Detection's mAP0.5 is slightly lower than YoloV3-Tiny and YoloV5-Lite (1.45% and 1.16% lower, respectively), it has fewer parameters and lower computational costs compared to these models. This balance strikes an optimal point between speed and accuracy, making it an ideal choice for medical emergency scenarios with limited computational resources.

Table 2. Model Comparison.

Method	Size	FLOPs	Parameters	mAP0.5	mAP0.5:0.95
YoloV3-Tiny	352×352	1.97G	8.66M	98.49	80.42
YoloV7-Tiny	352×352	13.2G	6.01M	96.02	66.05
NanoDet-m	352×352	0.87G	0.96M	90.20	65.70
Yolo-FastestV2	352×352	0.11G	0.23M	96.04	72.55
FastestDet	352×352	0.13G	0.23M	85.58	52.90
YoloV5-Lite	352×352	3.70G	1.54M	98.20	77.20
CPR-Detection	352×352	0.13G	0.20M	97.04	75.13

4.4. Measurement Results

One of the key parameters in CPR is the number and frequency of compressions. In this study, we identified each effective compression by analyzing the peaks and troughs of hand movements in the video, with each complete peak-trough cycle representing one compression. The frequency was calculated based on the number of effective compressions occurring per unit of time. Extensive testing showed that the accuracy of compression count and frequency exceeds 98%, with depth accuracy over 90% and errors generally within 1 cm. The errors in count and frequency were mainly due to initial fluctuations of the marker, while depth errors were often caused by inconsistencies in marker performance under different experimental conditions, such as camera angle and lighting changes. The video analysis-based method for measuring CPR compression count, frequency, and depth proposed in this study is highly accurate and practical. It is crucial for guiding first responders in performing standardized CPR, significantly enhancing the effectiveness of emergency care. Although there are some errors, further optimization of the algorithm and improvements in data collection methods are expected to enhance measurement accuracy.

Figure 11a shows the depth variance distribution for 100 compressions. Most data points have depth errors within ± 1 cm, meeting CPR operational standards and demonstrating the high accuracy of the measurement system. However, a few data points exceed a 1 cm depth error, likely due to changes in experimental conditions, such as slight adjustments in camera angle or lighting intensity, which can affect the visual recognition accuracy of the wristband. Figure 11b illustrates the accuracy for each of the 100 measurement tests conducted. A 90% accuracy threshold was set to evaluate the system's performance. Results indicate that the vast majority of measurements exceed this threshold, confirming the system's high reliability in most cases. However, there are a few instances where accuracy falls below 90%, highlighting potential weaknesses in the system, such as improper actions, insufficient device calibration, or environmental interference. Future work will focus on diagnosing and addressing these issues to improve the overall performance and reliability of the system.

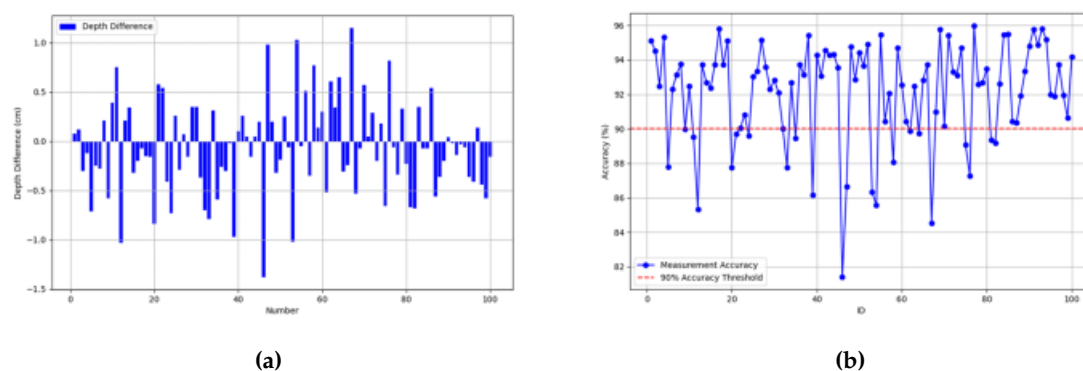


Figure 11. (a) Depth difference between actual depth and measured. (b) Measurement accuracy over time.

4.5. AED Application for CPR

When using the AED edge device, the user should wear the wristband on their arm and prepare for CPR. The usage process is as follows: After activating the AED edge device, the data collection unit starts automatically. Once the intelligent emergency function is initiated, the device automatically activates the AI recognition module, capturing real-time images of the emergency scene and collecting data for AI image recognition. During CPR, the AI recognition module uses multiple algorithms to assess whether the procedure meets standards. The voice playback and video display modules provide corrective prompts based on AI processing feedback. The storage module continuously records device operation, emergency events, detection, and AI recognition feedback. Medical emergency personnel can view real-time audio-visual information, location data, AED data, and AI recognition feedback sent by the intelligent module via the emergency platform server. The server also transmits this

data back to the device. The intelligent module connects to the emergency platform server through the communication module, retrieves the server's audio-visual data, and plays it through the voice playback and video display modules. As illustrated in the Figure 12, our algorithm's effectiveness in practical applications is demonstrated. We captured two frames from the AED edge device video after activation, showing the displayed activation time, compression count, frequency, and depth. Additionally, we used OpenPose to visualize skeletal points, capturing the arm's local motion trajectory during compressions [11]. This helps doctors assess the correctness of the posture via the emergency platform server.



Figure 12. Application scenario flow chart.

As shown in the Figure 13, after optimizing the algorithm on the edge device, the initial frame rate of 8 FPS was significantly improved. By applying quantization methods, the frame rate increased by 5 FPS. Pruning techniques added another 2 FPS, and the asynchronous method contributed an additional 7 FPS. Further enhancements were achieved with RGA and NEON, which improved the frame rate by 1 FPS and 2 FPS, respectively. Overall, the frame rate increased from 8 FPS to 25 FPS, validating the feasibility of these optimization methods.

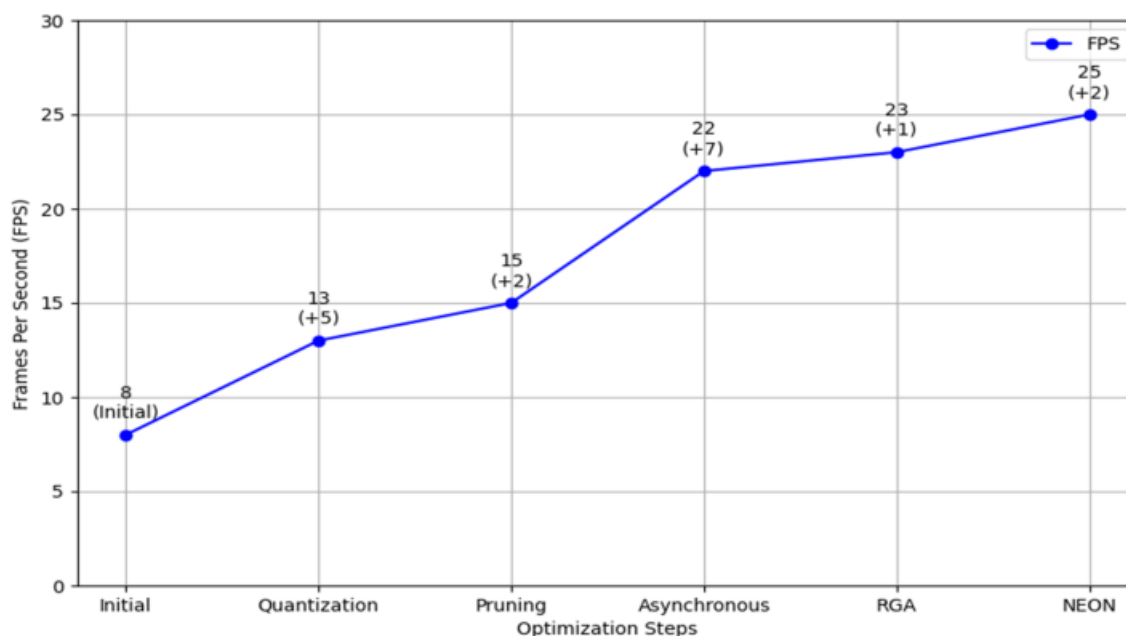


Figure 13. FPS Improvement through Various Optimazation Steps.

5. Conclusions

In this paper, we aim to address the issue related to lacking of standardized Cardiopulmonary Resuscitation (CPR) actions in Automated External Defibrillators (AEDs). We propose the Deep Learning-based CPR Action Standardization (DLCAS) method. The first part of DLCAS utilizes

OpenPose to identify skeletal points, enabling remote doctors to correct rescuers' posture through networked AED devices. In the second part of DLCAS, we design the CPR-Detection network. This network uses Partial Convolution (PConv) to enhance feature representation by focusing on critical spatial information. Additionally, we employed Mixed Local Channel Attention (MLCA) on our custom Small Target Detection-Feature Pyramid Network (STD-FPN). MLCA combines local and global contextual information, improving detection accuracy and efficiency. STD-FPN effectively merges shallow and deep image features, enhancing the model's localization capability. Based on CPR-Detection, we introduced a new depth algorithm to measure the rescuers' compression depth, count and frequency. In the third part of DLCAS, we applied computational optimization methods, including multi-threaded CPU and NPU asynchronous design, RGA and NEON acceleration, significantly boosting real-time processing efficiency. Extensive experiments on our custom dataset have shown that our method effectively addresses the issue of AED devices' inability to standardize CPR actions. Furthermore, our method improves the stability and speed of edge devices, validating the applicability of the DLCAS method in current medical scenarios through performance testing.

Author Contributions: Conceptualization, Y.L. and M.Y.; methodology, Y.L. and M.Y.; software, Y.L. and M.Y.; validation, Y.L., M.Y., and W.W.; formal analysis, Y.L., M.Y., W.W. and J.L.; investigation, Y.L., M.Y. and J.L.; resources, Y.L., M.Y., and W.W.; data curation, Y.L., M.Y., and W.W.; writing—original draft preparation, Y.L.; writing—review and editing, M.Y.; visualization, M.Y.; supervision, Y.L.; project administration, S.L. and Y.J.; funding acquisition, S.L. and Y.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key R&D Program of China (2023YFB2904000, 2023YFB2904004), the Jiangsu Key Development Planning Project (BE2023004-2), the Natural Science Foundation of Jiangsu Province (Higher Education Institutions) (20KJA520001), the 14th Five-Year Plan project of Equipment Development Department (315107402), the Jiangsu Hongxin Information Technology Co., Ltd. Project (JSSGS2301022EGN00), the Postgraduate Research & Practice Innovation Program of the Jiangsu Province (KYCX24_1204) and the Future Network Scientific Research Fund Project (No. FNSRFP-2021-YB-15).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Due to privacy protection for student volunteers, the data supporting the reported results will be made available upon request after acceptance and following privacy protection inquiries.

Acknowledgments: We would like to express our gratitude to Yuwell for their financial support and the provision of equipment used in our experiments. Their generous contributions were instrumental in the successful completion of this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Berdowski, J.; Berg, R.A.; Tijssen, J.G.; Koster, R.W. Global incidences of out-of-hospital cardiac arrest and survival rates: Systematic review of 67 prospective studies. *Resuscitation* **2010**, pp. 1479–1487.
2. Yan, S.; Gan, Y.; Jiang, N.; Wang, R.; Chen, Y.; Luo, Z.; Zong, Q.; Chen, S.; Lv, C. The global survival rate among adult out-of-hospital cardiac arrest patients who received cardiopulmonary resuscitation: a systematic review and meta-analysis. *Critical Care* **2020**, pp. 1–13.
3. Song, J.; Guo, W.; Lu, X.; Kang, X.; Song, Y.; Gong, D. The effect of bystander cardiopulmonary resuscitation on the survival of out-of-hospital cardiac arrests: a systematic review and meta-analysis. *SCANDINAVIAN JOURNAL OF TRAUMA RESUSCITATION & EMERGENCY MEDICINE* **2018**, pp. 1–10.
4. GrÅd'sner, J.T.; Wnent, J.; Herlitz, J.; Perkins, G.D.; Lefering, R.; Tjelmeland, I.; Koster, R.W.; Masterson, S.; Rossell-Ortiz, F.; Maurer, H.; et al. Survival after out-of-hospital cardiac arrest in Europe - Results of the EuReCa TWO study. *Resuscitation* **2020**, pp. 218–226.
5. Wang, S.; Yu, Q.; Wang, S.; Yang, D.; Su, L.; Zhao, X.; Kuang, H.; Zhang, P.; Zhai, P.; Zhang, L. CPR-Coach: Recognizing Composite Error Actions based on Single-class Training **2023**.
6. Everson, T.; Joordens, M.; Forbes, H.; Horan, B. Virtual reality and haptic cardiopulmonary resuscitation training approaches: A review. *IEEE Systems Journal* **2021**, *16*, 1391–1399.

7. Pivač, S.; Gradišek, P.; Skela-Savič, B. The impact of cardiopulmonary resuscitation (CPR) training on schoolchildren and their CPR knowledge, attitudes toward CPR, and willingness to help others and to perform CPR: mixed methods research design. *BMC Public Health* **2020**, *20*, 1–11.
8. Yin, J.; Ngiam, K.Y.; Teo, H.H. Role of artificial intelligence applications in real-life clinical practice: systematic review. *Journal of medical Internet research* **2021**, *23*, e25759.
9. Okonkwo, C.W.; Ade-Ibijola, A. Chatbots applications in education: A systematic review. *Computers and Education: Artificial Intelligence* **2021**, *2*, 100033.
10. Cao, Z.; Simon, T.; Wei, S.E.; Sheikh, Y. Realtime multi-person 2d pose estimation using part affinity fields. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 7291–7299.
11. Cao, Z.; Martinez, G.H.; Simon, T.; Wei, S.; Sheikh, Y.A. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2019**.
12. Simon, T.; Joo, H.; Matthews, I.; Sheikh, Y. Hand keypoint detection in single images using multiview bootstrapping. In Proceedings of the Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 1145–1153.
13. Joo, H.; Liu, H.; Tan, L.; Gui, L.; Nabbe, B.; Matthews, I.; Kanade, T.; Nobuhara, S.; Sheikh, Y. Panoptic studio: A massively multiview system for social motion capture. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2015, pp. 3334–3342.
14. Gholami, A.; Kwon, K.; Wu, B.; Tai, Z.; Yue, X.; Jin, P.; Zhao, S.; Keutzer, K. SqueezeNext: Hardware-Aware Neural Network Design. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018.
15. Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications [arXiv]. *arXiv* **2017**, p. 9.
16. Zeng, T.; Li, S.; Song, Q.; Zhong, F.; Wei, X. Lightweight tomato real-time detection method based on improved YOLO and mobile deployment. *Computers & Electronics in Agriculture* **2023**.
17. Cong, S.; Zhou, Y. A review of convolutional neural network architectures and their optimizations. *ARTIFICIAL INTELLIGENCE REVIEW* **2023**, pp. 1905–1969.
18. Wei, Y.; Zhao, L.; Zheng, W.; Zhu, Z.; Zhou, J.; Lu, J. SurroundOcc: Multi-Camera 3D Occupancy Prediction for Autonomous Driving. In Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 2023.
19. Wu, D.; Liao, M.W.; Zhang, W.T.; Wang, X.G.; Bai, X.; Cheng, W.Q.; Liu, W.Y. Correction to: YOLOP: You Only Look Once for Panoptic Driving Perception. *Machine Intelligence Research* **2023**, p. 952.
20. Xu, M.; Wang, X.; Zhang, S.; Wan, R.; Zhao, F. Detection algorithm of aerial vehicle target based on improved YOLOv3. *Journal of Physics: Conference Series* **2022**, p. 012022.
21. Jamiya, S.S.; Rani, P.E. An Efficient Method for Moving Vehicle Detection in Real-Time Video Surveillance. In Proceedings of the Advances in Smart System Technologies, 2020.
22. Wu, S.; Zhang, L. Using Popular Object Detection Methods for Real Time Forest Fire Detection. In Proceedings of the 2018 11th International Symposium on Computational Intelligence and Design (ISCID), 2018.
23. Mishra, S.; Jabin, S. Anomaly detection in surveillance videos using deep autoencoder. *International Journal of Information Technology (Singapore)* **2024**, pp. 1111–1122.
24. Ali, M.M. Real-time video anomaly detection for smart surveillance. *IET Image Processing (Wiley-Blackwell)* **2023**, pp. 1375–1388.
25. Sun, S.; Xu, Z. Large kernel convolution YOLO for ship detection in surveillance video. *Mathematical Biosciences and Engineering* **2023**, pp. 15018–15043.
26. Zhang, X.; Xuan, C.; Xue, J.; Chen, B.; Ma, Y. LSR-YOLO: A High-Precision, Lightweight Model for Sheep Face Recognition on the Mobile End. *Animals* **2023**, p. 1824.
27. Yu, F.; Zhang, G.; Zhao, F.; Wang, X.; Liu, H.; Lin, P.; Chen, Y. Improved YOLO-v5 model for boosting face mask recognition accuracy on heterogeneous IoT computing platforms. *INTERNET OF THINGS* **2023**, p. 100881.
28. Sun, F. Face Recognition Analysis Based on the YOLO Algorithm. In Proceedings of the The 4th International Conference on Computing and Data Science (CONF-CDS 2022), 2022.

29. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement **2018**.
30. Wong, A.; Famuori, M.; Shafiee, M.J.; Li, F.; Chwyl, B.; Chung, J. YOLO Nano: a Highly Compact You Only Look Once Convolutional Neural Network for Object Detection **2019**.
31. Hu, L.; Li, Y. Micro-YOLO: Exploring Efficient Methods to Compress CNN based Object Detection Model. In Proceedings of the International Conference on Agents and Artificial Intelligence, 2021.
32. Lyu, R. Nanodet-plus: Super fast and high accuracy lightweight anchor-free object detection model. URL: <https://github.com/RangiLyu/nanodet> **2021**.
33. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021 **2021**.
34. Jocher, G.; Nishimura, K.; Mineeva, T.; Vilarino, R. yolov5. *Code repository* **2020**, p. 9.
35. DOG-QIUQIU, A. dog-qiuqiu/Yolo-Fastest: Yolo-fastest-v1. 1.0 **2021**.
36. Ma, X. Fastestdet: Ultra lightweight anchor-free realtime object detection algorithm, 2022.
37. Chen, J.; hong Kao, S.; He, H.; Zhuo, W.; Wen, S.; Lee, C.H.; Chan, S.H.G. Run, Don't walk: Chasing higher FLOPS for faster neural networks. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12021–12031.
38. Wan, D.; Lu, R.; Shen, S.; Xu, T.; Lang, X.; Ren, Z. Mixed local channel attention for object detection. *Engineering Applications of Artificial Intelligence* **2023**, *123*, 106442.
39. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132–7141.
40. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), 2018, pp. 3–19.
41. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), 2018, pp. 116–131.
42. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 6848–6856.
43. You, Z.; Luan, Z.; Wei, X. General lens distortion model expressed by image pixel coordinate. *Optical Technique* **2015**, *41*, 265–269.
44. Dewi, C.; Chen, R.C. Random forest and support vector machine on features selection for regression analysis. *Int. J. Innov. Comput. Inf. Control* **2019**, *15*, 2027–2037.
45. Yuan, Y.; Xiong, Z.; Wang, Q. An incremental framework for video-based traffic sign detection, tracking, and recognition. *IEEE Transactions on Intelligent Transportation Systems* **2016**, *18*, 1918–1929.
46. Dewi, C.; Chen, R.C.; Tai, S.K. Evaluation of robust spatial pyramid pooling based on convolutional neural network for traffic sign recognition system. *Electronics* **2020**, *9*, 889.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.