electronics

Article

A Face Detector with Adaptive Feature Fusion in Classroom Environment

Cheng Sun [1], Pei Wen [2], Shiwen Zhang [2], Xingjin Wu [2], Jin Zhang [2,3,*] and Hongfang Gong [4]

[1] School of Mathematics and Statistics, Hunan Normal University, Changsha 410081, China
[2] College of Information Science and Engineering, Hunan Normal University, Changsha 410081, China
[3] School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China
[4] School of Mathematics and Statistics, Changsha University of Science and Technology, Changsha 410114, China
* Correspondence: mail_zhangjin@163.com

Abstract: Face detection in the classroom environment is the basis for student face recognition, sensorless attendance, and concentration analysis. Due to equipment, lighting, and the uncontrollability of students in an unconstrained environment, images include many moving faces, occluded faces, and extremely small faces in a classroom environment. Since the image sent to the detector will be resized to a smaller size, the face information extracted by the detector is very limited. This seriously affects the accuracy of face detection. Therefore, this paper proposes an adaptive fusion-based YOLOv5 method for face detection in classroom environments. First, a very small face detection layer in YOLOv5 is added to enhance the YOLOv5 baseline, and an adaptive fusion backbone network based on multi-scale features is proposed, which has the ability to feature fusion and rich feature information. Second, the adaptive spatial feature fusion strategy is applied to the network, considering the face location information and semantic information. Finally, a face dataset Classroom-Face in the classroom environment is creatively proposed, and it is verified with our method. The experimental results show that, compared with YOLOv5 or other traditional algorithms, our algorithm portrays better performance in WIDER-FACE Dataset and Classroom-Face dataset.

Keywords: face detection; adaptive fusion; Classroom-Face

Citation: Sun, C.; Wen, P.; Zhang, S.; Wu, X.; Zhang, J.; Gong, H. A Face Detector with Adaptive Feature Fusion in Classroom Environment. *Electronics* **2023**, *12*, 1738. https://doi.org/10.3390/electronics12071738

Academic Editor: Chiman Kwan

Received: 3 March 2023
Revised: 1 April 2023
Accepted: 3 April 2023
Published: 6 April 2023

1. Introduction

Face detection is a major research project in the field of image processing. Generally, face position localization is the first step of face recognition, face attribute analysis, and human-computer interaction; thus, the accuracy of face detection directly affects the performance of face-related systems. Face detection in a classroom environment is the basis of face recognition and student attention detection. Effectively detecting students' facial information in the classroom environment helps to analyze students' attendance and the efficiency of listening accurately and quickly.

The human face is a special object, and to better solve the problems in face detection, convolutional neural networks (CNNs) have grown in popularity in recent years. Further, scholars applied object detection algorithms [1–4] to detect faces and made progress. Meanwhile, the birth of FDDB [5], VOC FACE, and WIDER-FACE [6] benchmark datasets, which provide face location information in real-world scenes with high variability in terms of scale, pose, and occlusion, makes face detection ever more widely applied in real complex scenes.

In recent years, face detection has attracted the attention of many scholars as well as industries. Thanks to the use of deep learning, face detection accuracy has risen over time, and significant technological advancements have been achieved. However, there are still many challenges in the technique of accurately locating the face position due to different shooting angles, differences in lighting conditions, background noise interference,

crowding of people, and changes in the expressions of people's postures. The classroom environment is a type of unconstrained environment. In the classroom, there necessitates the ability to detect multiple faces in an image. The image has the following characteristics: faces of different sizes and postures, small-scale faces far away from the camera, and moving and occluded faces that occupy fewer pixels. The features are not obvious, and the recall rate is low compared to the large-scale face detection that is closer to the camera. Therefore, one of the most important problems in face detection technology is how to increase the accuracy of tiny face detection with several faces in the classroom.

Although the accuracy of face detection algorithms on large public datasets has been refreshed to over 90%, most of the public datasets are celebrity images or crawled from the internet. There are fewer face datasets and less related research in unconstrained environments. Currently, face detection algorithms are primarily applied in non-specific environments. However, in classroom settings, face detection faces several challenges, including low resolution, variation in scale, crowding, severe occlusion, and motion blur. The existing face detection algorithms cannot effectively extract features from small faces, cannot detect faces of different scales well in a single image, rely on keypoint detection, creating relevant datasets is a complex task, and the algorithm may not generalize well to complex real-world scenarios. YOLOv5 has good detection accuracy on VOC [7] and COCO [8] datasets; therefore, we propose an improved YOLOV5 architecture for detecting tiny faces in classroom environments. The emphasis of this research is to propose a new face detection algorithm based on the adaptive fusion of multi-scale features in classroom environments, which is related to face detection methods in non-specific environments as well as face detection algorithms in classroom environments. These are the primary contributions:

(1) A small-scale face detection layer is added to address the problem that it is difficult to detect small-scale student faces that are far from the camera.
(2) To address the issue of changeable scale and inadequate identification of fuzzy faces, a multi-scale adaptive backbone network is presented. Firstly, the feature map is convolutionally concatenated at different scales to generate a new multi-scale feature map; secondly, the attention weights of different channels of the multi-scale feature map are extracted by SENet and rescaled by softmax to obtain an adaptive multi-scale fusion feature map. Finally, the original feature map is concatenated with the adaptive multi-scale fusion feature map to ensure the new features are obtained based on the original features.
(3) To address the problem of denser concentrations of students in the classroom environment and the mismatch between location information and semantic information, the pyramid feature fusion strategy is applied to the network detection in part to reduce the missed detection rate of students' faces in the classroom environment through adaptive fusion between different scales of detection layers.
(4) This paper establishes a publicly available face dataset in the classroom environment (Classroom-Face) and divides it into three subsets for testing.

2. Related Work

2.1. Face Detection

Since 1990, face detection has become increasingly important as the first step in face analysis tasks. Currently, CNNs have been most wildly applied for face detection, face recognition, expression analysis, and other face feature extraction-related fields. According to the network architecture and training strategy, face detection algorithms can be classified into cascade detection algorithms, one-stage detection algorithms, two-stage detection algorithms, and feature pyramid-based algorithms.

H. Li et al. [9] were the first to propose a CNN combined with a V-J cascade structure for face detection, which contains six subnetworks, three of which are used to determine whether a face is a face and the other three are used for face rectangle frame correction. Zhang et al. [10] detected faces and face alignment by a third-order cascaded CNN, with

each order of the network containing three tasks: face determination, border regression, and localization of facial feature points. In addition, the works of Yang, Zhang, and Tomar [11–13] also rely on cascaded convolutional networks for face detection.

Wang et al. [14] combined multi-task loss function and multi-scale training with R-CNN to improve the speed and accuracy of face detection. To better use Faster R-CNN for face detection tasks, CMS-RCNN [15] fused contextual information and proposed multi-scale suggestions to fuse features after RoI Pooling on a multilayer feature map. Wu, Zhang, and Cakiroglu [16–18] performed face detection based on Fast R-CNN, Faster RCNN, and Mask RCN.

Face detection is a specific object detection task and thus, is usually derived and extended from generic object detection algorithms. Single-stage object detection algorithms [19,20] have emerged as a new and important study area as object detection algorithms have gotten faster and more accurate as deep learning becomes more common. The algorithm did not rely on FCN but detected faces at different scales simultaneously in the network and achieved the advancement results in [5,6] and VOC face datasets. Faceboxes [21] proposed "Rapidly Digested Convolutional Layers (RDCL)" and the "Multiple Scale Convolutional Layers (MSCL)", which can combine both detection speed and accuracy of face detection at different scales. Hu [22] solved the accuracy of small face detection by investigating three aspects of scale invariance, image resolution, and contextual inference in combination with a single-stage detector. Deng et al. [23] designed a more accurate face detector, RetinaFace, based on RetinaNet [24], manually labeling five feature points in the WIDER-FACE dataset and introducing a self-supervised network to predict three-dimensional facial information with the supervised network.

Tang et al. [25] proposed the addition of feature fusion in a regional full convolutional network R-FCN for detecting small-scale faces. PyramidBox [26] proposed to use the contextual anchor to detect occluded faces by learning contextual information about the human head and body, combining low-level and high-level semantic information to detect faces of various scales. Saha et al. [27] introduced recurrent neural networks into computer vision tasks to efficiently aggregate feature information and rapidly reduce activation map sampling, experimentally demonstrating that the RNNPool layer could effectively replace feature extraction sub-networks such as MobileNets [28].

2.2. Face Detection in Classroom

In recent years, face detection has developed by leaps and bounds; advanced algorithms have achieved more than 90% accuracy on all face benchmark datasets. Face detection algorithms in unrestricted situations are less reliable because of factors like a paucity of real-world datasets and their inconsistent quality. The classroom environment is one of the unconstrained environments; lighting conditions, face pose, motion, and the relative distance between the face and the camera are all difficult issues for face detection in the classroom. With the promotion of smart education, face detection in the classroom helps the subsequent research of face recognition, sensorless attendance, and student concentration detection. Increasingly greater numbers of scholars have started to study face detection in classroom environments.

Phakjira [29] combined generic instance segmentation with a robust face detection algorithm to detect faces and segment students as objects. Karnalim [30] proposed a publicly available dataset for face detection in classroom environments and quantified the dataset using four pre-trained models, but the dataset had clear and scattered faces, which were not suitable for student face detection with more than ten faces. Hu [31] proposed a non-public classroom environment face detection dataset of 0–15 students and detected student faces by MTCNN combined with the HOG algorithm. Gu [32] achieved the detection of student faces in a real classroom environment by simplifying MTCNN and introducing a residual generation feature module [33–36], first by implementing face detection in the classroom environment using face detection algorithms and then using the detected face feature information for student classroom concentration analysis.

3. Methods

In this section, we instantiate the face detection algorithm based on feature fusion to show how it works on YOLOv5. First, we add a small face detection layer to introduce a stronger baseline than the original YOLOv5. Then, we propose a new backbone network for small face detection, drawing from the pyramid fusion algorithm. Finally, we show the details of training, testing, and modeling. The overall framework of our proposed network is shown in Figure 1.

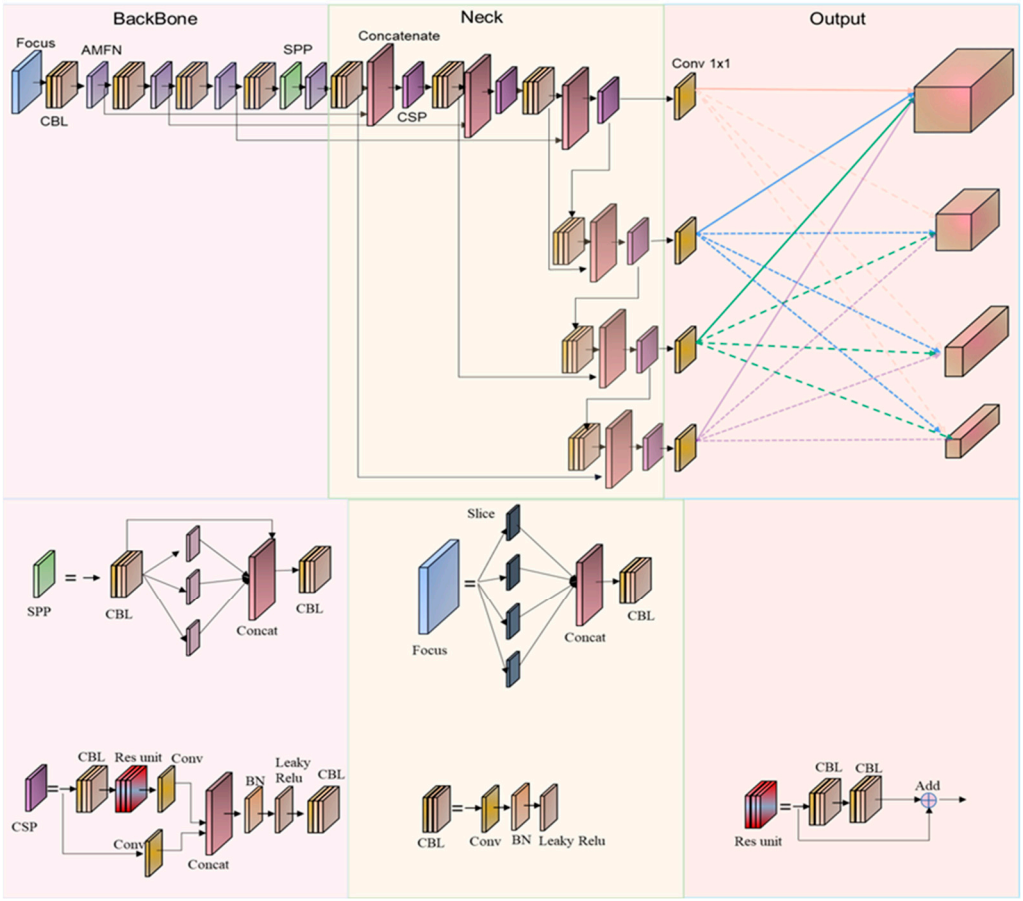


Figure 1. The architecture of AMFN-YOLOv5 (among them, the parts we optimized are the design of the new backbone network part AMFN and the adaptive pyramid fusion strategy in the output. AMFN is in Section 3.2.1, and the adaptive pyramid fusion strategy is in Section 3.2.2. SPP is spatial pyramid pooling, CSP is used for feature extraction, CBL is a component module of resnet, and Focus is an image slice operation).

3.1. Strong Baseline

The face detection dataset collected from the real classroom environment has different sizes of students' faces due to their different distances from the camera. In the dataset, the faces of students in the front row are clearer and occupy more pixels, while the faces of students in the back row are smaller. The initial YOLOv5 algorithm cannot solve the problem of detecting smaller faces due to the varying face size caused by the shooting distance. To make our face detection algorithm more generalizable, we added a new detection layer to the original YOLOv5 algorithm framework based on the respective data characteristics of the classroom environment face detection dataset and the large public face dataset--WIDER-FACE dataset.

The output part of the YOLOv5 detector uses PANet. PANet is based on FPN with the addition of a bottom-up feature pyramid and fused with the features in FPN. In YOLOV5, PANet has three output layers; each layer detects different scales, corresponding to 80 $\times$ 80,

40 × 40, and 20 × 20, which are used to detect 8 × 8, 16 × 16, and 32 × 32 objects, respectively. However, in the public dataset WIDERFACE and the face images in the real classroom environment crawled through the web, some face images are smaller than 8 × 8, so the existing anchor boxes cannot effectively detect smaller faces. In this paper, we add an output layer with a 160 × 160 feature map, which helps to detect small faces above 4 × 4. Larger faces are easier to detect in various face detection tasks, but the detection of very small faces is a persistent issue in face detection. We solve this problem by increasing the detection layer.

3.2. The Two-Stage Spatial Feature Fusion

To effectively detect small faces as well as blurred faces in images, a new backbone network architecture is designed. In this section, we illustrate our adaptive multi-scale feature fusion network (AMFN), showing how it makes YOLOv5 more powerful in face detection. Our method is based on the YOLOv5 framework but employs feature fusion techniques in the backbone network as well as in the head prediction part. In the dataset, faces appear at different scales due to their different distances from the camera, so we have designed a new backbone network for generalizing face detection at various scales using a multi-scale fusion technique.

3.2.1. Adaptive Muti-Scale Fusion Network

The mechanism of this work is to build a backbone network with different channels of adaptive multi-scale convolution. As shown in Figure 2, it is implemented by the following steps: firstly, the parameters are reduced by 1 × 1 convolution, and secondly, the feature maps are passed through convolution kernels of different scales to obtain multi-scale feature maps and are re-concatenated. Then, the multi-scale feature maps are concatenated by extracting the attention weight vectors of feature information at different scales through the SE attention mechanism module, and the attention vectors in the channel direction are calibrated with the softmax function to obtain the new weights. Finally, the new weights are matched with the multi-scale feature maps to obtain feature maps with more refined multi-scale feature information.

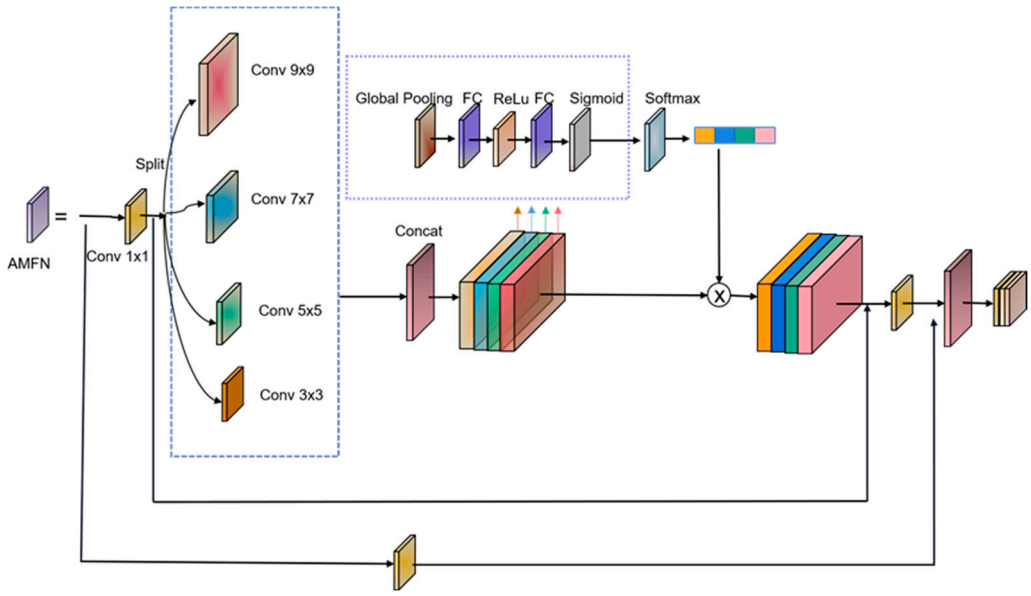


Figure 2. The architecture of AMFN (The calculation process of the AMFN backbone network. The computation process of the AMFN backbone network involves extracting feature importance through attention mechanisms, followed by weight calibration of feature maps that have undergone different convolutions and concatenation fusion using the softmax function).

According to Figure 2, it can be seen that we recalibrate the attention weights of the extracted feature information on different channels using the softmax function to obtain feature maps with richer multi-scale information expressiveness. The spatial information of the features is extracted by separating the channels, and the multi-scale features are obtained by different convolutions. This operation allows us to obtain the tensor information of different channels as well as the possibility of parallelizing the processing of multi-scale features. Firstly, 1×1 convolution is used to reduce the number of parameters, and then the convolved feature map channels are divided into four groups, and the feature maps of different channels are convolved by using 3×3, 5×5, 7×7, and 9×9 multi-scale convolution kernels.

$$C_{out} = C_{in}/4 \tag{1}$$

Four different sets of spatial features with different convolution kernels can learn spatial information at different scales and achieve cross-channel fusion. Different spatial multi-scale features $F' \in R^{C_{in} \times W \times H}$ are obtained by concatenation.

$$F' = Concat(F_1, F_2, F_3, F_4) \tag{2}$$

Attention weight vectors for each scale of multi-scale features are extracted using *SEWeight*:

$$Z_i = SEWeight(F_i), i = 0, 1, 2, 3 \tag{3}$$

where $Z_i \in R^{C_{out} \times 1 \times 1}$ are the attention weights extracted by the SE module in the feature maps at different scales, and the multi-scale attention weights are fused to obtain the overall attention vector.

$$Z = Concat(Z_0, Z_1, Z_2, Z_3) \tag{4}$$

To establish long-term channel attention dependence and to enable information interaction before multi-scale attention in different spaces, the channel weights are rescaled using *softmax*.

$$z_i' = softmax(Z_i) = \frac{\exp(Z_i)}{\sum_{i=0}^{3} \exp(Z_i)}, i = 0, 1, 2, 3 \tag{5}$$

The rescaled feature vectors are multiplied with the multi-scale feature maps of the corresponding channels to obtain the weighted feature maps.

$$Y_i = F_i \cdot z_i', i = 0, 1, 2, 3 \tag{6}$$

The dimensionality of the weighted feature maps in different channels is concatenated together to obtain a richer feature map in multi-scale information as well as more refined.

$$Out = shorcut(F, Concat(Y_0, Y_1, Y_2, Y_3)) \tag{7}$$

The parameters are reduced by 1×1 convolution, and the initial features are further concatenated with the weighted features in dimension, thus ensuring that the original feature information is not lost while having more refined features.

$$Out' = Conv(Concat(out, conv(in))) \tag{8}$$

3.2.2. f-Adaptive Spatial Future Fusion

Normally, stacking CNNs is used to extract the features in the object detection methods. This operation is easy to cause the object-semantic information in the shallow features to be weak but has rich position information, and the object-semantic information in the deep features is strong, though the position information is weak, as shown in Figure 3. PANet mainly solves the problem of target scale variation. PANet outputs different scales to detect objects of various sizes, but when PANet outputs a feature map to match a certain object, the feature information of other layers will be disregarded. There is a problem of feature

inconsistency between different scales. In the face dataset, all the objects to be detected are faces. When the feature maps between different scales match the face information, the face information on the feature maps of other layers is ignored, which easily leads to the problem of face misdetection. In this paper, we combine the pyramid fusion strategy to YOLOv5 with PANet, which makes the feature size uniform and the features between different scales fused.

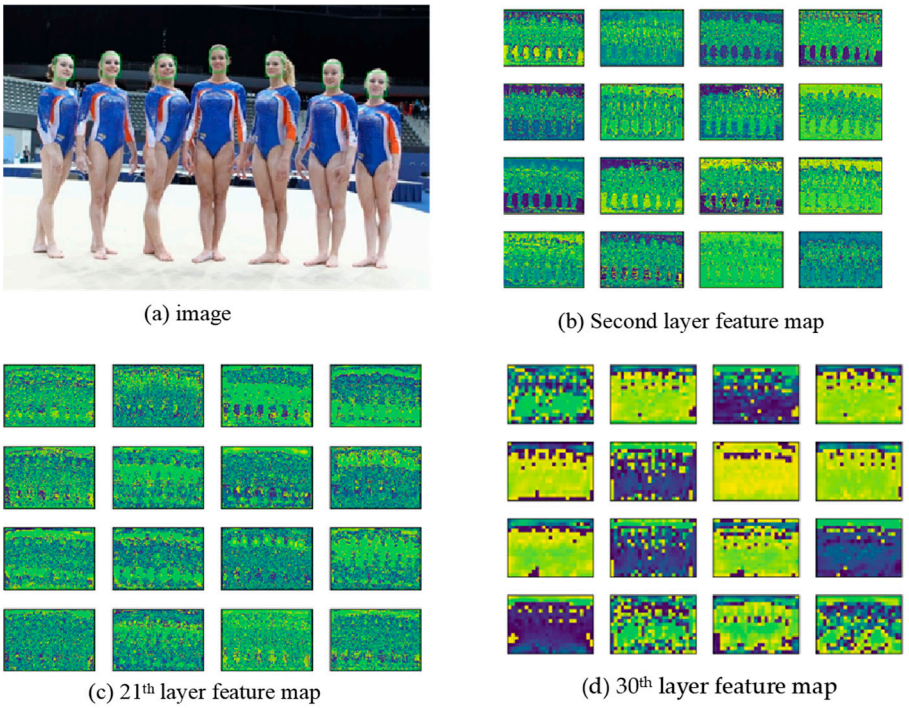


Figure 3. Shallow features and deep features ((**a**) is the input image with labels, (**b**) is the feature map of the second layer of feature extraction, (**c**) is the feature map of the 21st layer in the feature extractor, and (**d**) is the feature map of the 30th layer of feature extraction).

Since the four layers of PANet have different sizes and a different number of channels, the first step is to modify the feature maps that need to be fused to a uniform size by up-sampling and down-sampling algorithms. For example, if the feature maps of layers 1, 3, and 4 need to be fused with the layer 2 feature maps, the size of the layer 1, 3, and 4 feature maps and the number of channels need to be modified accordingly to be consistent with the features of the layer 2 feature maps, as in Equation (9), where $x_{i,j}^{n}$ denotes the n$^{\text{th}}$ layer feature:

$$F_2 = Upsample(F_3)\&Upsample(F_4)\&Downsample(F_1) \tag{9}$$

By setting α, β, γ, and φ to represent different weights in the fused feature maps. $x_{i,j}^{n\to l}$ denotes the feature mapping vector of the n-layer adjusted to the (i, j) position on the l-layer feature map. $y_{i,j}^{l}$ denotes the feature vector of the (i, j) position of the n-layer weighted fusion with the l-layer.

$$y_{i,j}^{l} = \alpha_{i,j}^{l} x_{i,j}^{1\to l} + \beta_{i,j}^{l} x_{i,j}^{2\to l} + \gamma_{i,j}^{l} x_{i,j}^{3\to l} + \varphi_{i,j}^{l} x_{i,j}^{4\to l} \tag{10}$$

$\alpha_{i,j}^{l}$, $\beta_{i,j}^{l}$, $\gamma_{i,j}^{l}$, and $\varphi_{i,j}^{l}$ are the weights of each of the four different layers of the network fused to the l-layer by softmax adaptive learning.

$$\alpha_{ij}^{l} = \frac{e^{\lambda_{\alpha_{ij}}^{l}}}{e^{\lambda_{\alpha_{ij}}^{l}} + e^{\lambda_{\beta_{ij}}^{l}} + e^{\lambda_{\gamma_{ij}}^{l}} + e^{\lambda_{\varphi_{ij}}^{l}}} \tag{11}$$

$\lambda^l_{\alpha_{ij}}$ is the control parameter that makes the features at (i, j) position of each $n \rightarrow l$ layer calculated by 1×1 convolution. The above equation satisfies.

$$\alpha^l_{ij} + \beta^l_{ij} + \gamma^l_{ij} + \rho^l_{ij} = 1, \ \alpha^l_{ij}, \ \beta^l_{ij}, \ \gamma^l_{ij}, \ \rho^l_{ij} \in [0, 1] \tag{12}$$

4. Experiment and Analysis

In this part, we assess the AMFN-YOLOv5 algorithm's performance using both a large public dataset and a private dataset. The datasets used in the experiment are the WIDER-FACE and Classroon-Face datasets. The training dataset is a mixture of the WIDER-FACE dataset and the Classroom-Face dataset. To verify the model's efficacy and resilience, extensive face-testing experiments are conducted using the large publicly face dataset WIDER-FACE; student face detection in a classroom environment is conducted by the self-built dataset Classroom-Face dataset.

The most popular face detection dataset currently created by the Chinese University of Hong Kong is WIDER-FACE. The dataset contains 32,203 images with 393,703 face data labeled, which are divided into 61 scenes according to image types, but not including classroom scenes. The mAP(mean Average Precision) is calculated separately by WIDER-FACE evaluation following the VOC dataset evaluation method, which is divided into Hard/Medium/Easy.

4.1. Classroom-Face Dataset

As the accuracy of face detection algorithms is improving, there are more and more challenging face datasets. At present, only Oscar [30] has proposed a public face dataset in a classroom environment, but it is a constrained environment with dispersed personnel, clear and unobstructed faces, and no moving faces. The dataset includes 90 students participating with only 194 images in a computer lab environment, which does not correspond to a non-constrained classroom environment close enough to a real classroom. We propose the Classroom-Face dataset, which will be used by scholars around the world for research related to the face tasks of the classroom environment, such as sensorless attendance of students in classroom environments and concentration detection. To maximize the use of the dataset, we have asked for consent from participating students in the dataset and made the dataset publicly accessible on GitHub. We plan to expand and update our dataset by continuously collecting suggestions from scholars and face data from real classroom environments. Figure 4 illustrates our dataset production process.

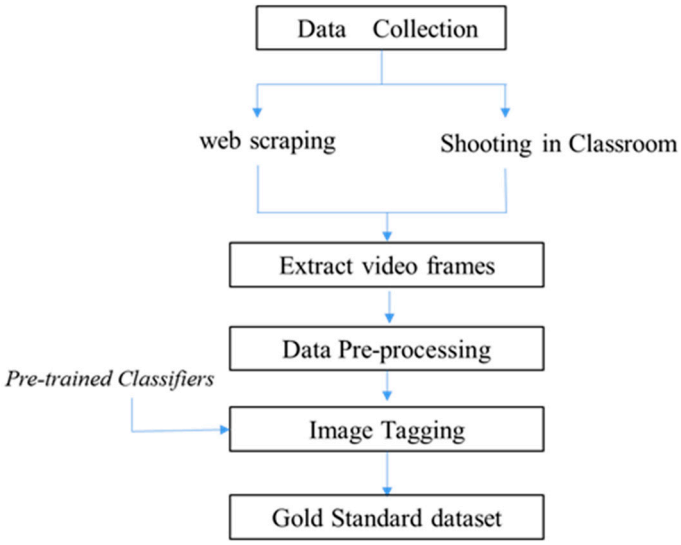


Figure 4. The process of dataset production (visually showing the collection and data processing process of the Classroom-Face dataset).

The images in the Classroom-Face dataset were obtained by the following steps: (1) web crawling using an engine search images, which have clear faces and high resolution, roughly 1000 images; and (2) video of real classroom environments. Using a tripod to place a high-definition camera in front of the classroom, we shot videos of students in real classrooms, taking a 1 s time interval for one frame to intercept the image. We chose five different classrooms containing 120 students and took a total of 13 videos under night lighting and natural daytime light, obtaining 6900 images. This part of the dataset is characterized by low resolution and moving faces, which makes detection difficult.

Data annotation: due to a large amount of image data, we use a semi-automated annotation method to mark the boundaries of the faces in all images using rectangular boxes to save labor and time. In order for researchers to better use our data set to detect faces and apply them to face recognition or student attention detection, we use the PASCAL VOC data set cleaning method to organize our dataset. We remove faces that are difficult for the human eye to recognize and to maintain the format consistent with the PASCAL VOC dataset.

We use the re-trained model from the open-source project MTCNN to perform preliminary face detection on the original images in the classroom environment and save the results in XML format, which is convenient for workers to adjust using LableImage and obtain the final face detection dataset in the classroom environment.

To the best of our knowledge, Classroom-Face is the most realistic and unconstrained face dataset in the classroom environment. We divide the datasets for training and testing in an 8:2 ratio. The test set contains 1582 images in the classroom environment and is divided into three subsets: Easy\Medium\Hard, each of which contains natural light and dim light (daylight/dimlight). Among them, the easy subset contains 1–20 students; the Medium subset is a crowded classroom environment containing 20–30 students; and the Hard subset has several students, ranging from 30–50, with more crowded seats and poor pixels. The specific values are shown in Table 1. We name each part of the dataset based on the above characteristics as shown in Figure 5: Easy-daylight, Easy-dimlight, Medium-daylight. Medium-dimlight, Hard-daylight, Hard-dimlight.

Table 1. Classroom-Face.

Train	**Test**					
	Easy (1–20)		Medium (20–30)		Hard (30–50)	
7910	daylight 462	dimlight 109	daylight 388	dimlight 169	dimlight 168	daylight 286

Some subsets of the dataset are shown in Figure 5:

(**a**) Easy_daylight

(**b**) Easy_dimlight

Figure 5. *Cont.*

(**c**) Medium_daylight

(**d**) Medium_dimlight

(**e**) Hard_daylight

(**f**) Hard_dimlight

Figure 5. Partial image in Classroom-Face. According to the number of people, the data set is divided into Easy, Medium, and Hard. According to the light, the data set is divided into dimlight and daylight.

4.2. Train Details

During training, we used the bounding box regression loss combined with the face loss as our final loss. The loss function is as follows:

$$Loss_{box} = L_{GIoU} = 1 - \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} + \frac{|C - B \cup B^{gt}|}{|C|} \tag{13}$$

In Equation (13), we choose the *GIoU* loss function as the bounding *box* regression loss function, where C represents the smallest rectangle containing B and B^{gt}. B is the prediction box. B^{gt} refers to the ground-truth box.

$$Loss_{face} = \frac{1}{N}\sum_{n}^{N} -[y_n * logx_n + (1 - y_n) * \log(1 - x_n)] \tag{14}$$

In Equation (14), the *face* prediction loss is the BCE loss, y_n is the true label, and x_n is whether the model predicts a face or not. Equation (15) represents the overall loss value of the model.

$$Loss = Loss_{box} + Loss_{obj} \tag{15}$$

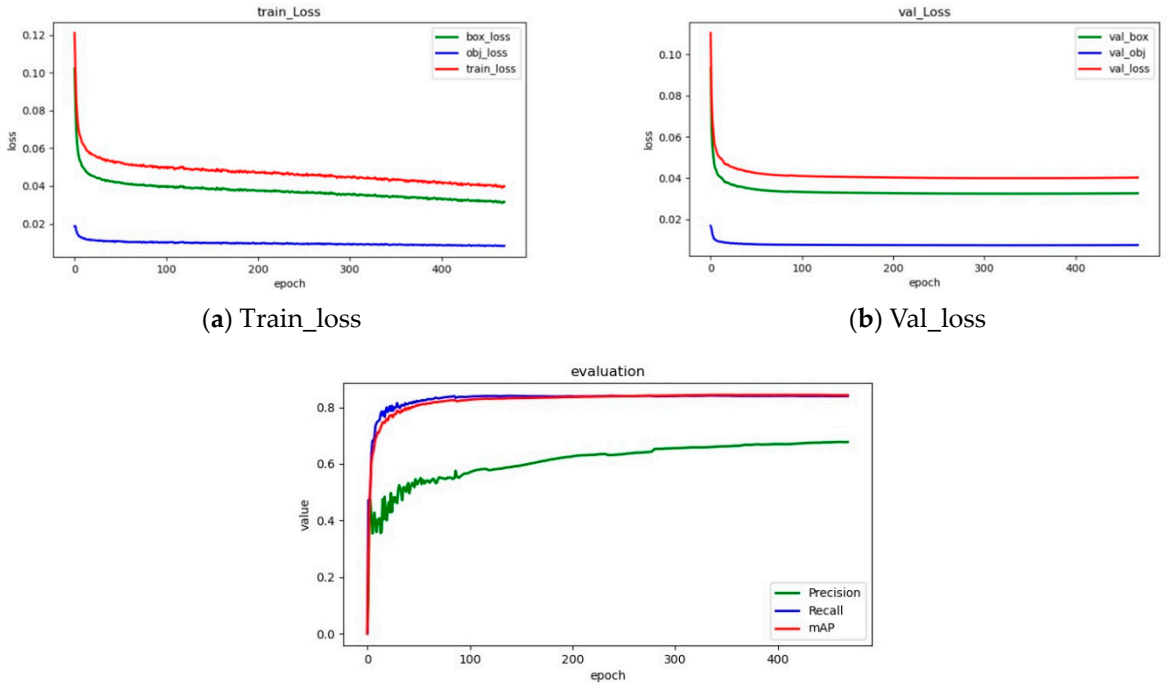
During AMFN-YOLOv5 training, to make the model perform at its optimum, we set the epochs to 500, the weight decay coefficient to 0.001, the learning rate momentum to 0.94, and the learning rate to 0.01 to prevent overfitting. The batch_size is set to 16. To avoid the disappearance of features caused by reducing the image size and the training difficulty, the image's size is set to 800 × 800. Figure 6 shows the change process of each loss and the change curve of Precision, recall, and mAP in training and verification. The loss tends to be flat at about 400 epochs, and the map is optimal.

4.3. Comparison with Analysis on WIDER-FACE Dataset

4.3.1. Comparison with SOTA Algorithms

For fairness, many face detection algorithms are not open-sourced on GitHub and cannot be compared individually. Therefore, we compare our algorithm with the SOTA methods on the WIDER-FACE dataset. The face detection algorithms of the WIDER-FACE dataset are categorized into keypoint-based training and rectangle-box-based training methods. In this paper, considering the manpower and time issues for subsequent self-

built datasets, the face key points are not annotated in the dataset. Consequently, the face detector proposed in this paper is compared with other classical face detector algorithms based on rectangular boxes on the WIDER-FACE dataset, and the results are shown in Table 2.



(**a**) Train_loss (**b**) Val_loss

(**c**) Evaluation

Figure 6. Changes in loss and evaluation index values.

Table 2. Comparison with the typical method based on the rectangle box.

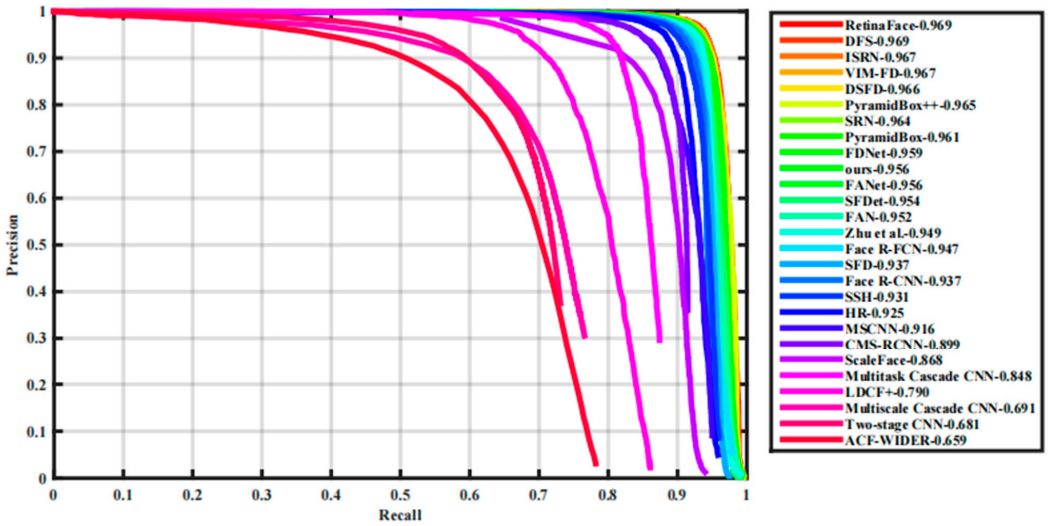
Methods	Easy	Medium	Hard
Multi-scale Cascade CNN	69.1	66.4	42.4
LDCF+	79.0	76.9	52.2
ScaleFace	86.8	86.7	77.2
CMS-RCNN	89.9	87.4	62.4
Chaowei Tang	91.3	89.6	79.4
YOLOv5	95.4	93.7	84.5
Our Method	95.6	94.7	89.1

The PR curves for our algorithm and other SOTA methods on the WIDER-FACE dataset are displayed in Figure 7. According to the figure, our face detector achieves 95.6%, 94.7%, and 89.1% on the Easy, Medium, and Hard subsets, respectively. Although, our method falls short of the Retinaface, which completes face detection through key points. Our method is slightly lower than Retinaface on the Easy, Medium, and Hard subsets of the WIDER-FACE dataset without training with the keypoint annotation dataset, but from Figure 7, the accuracy of our algorithm in face detection is still higher than most of the classical algorithms at this stage.

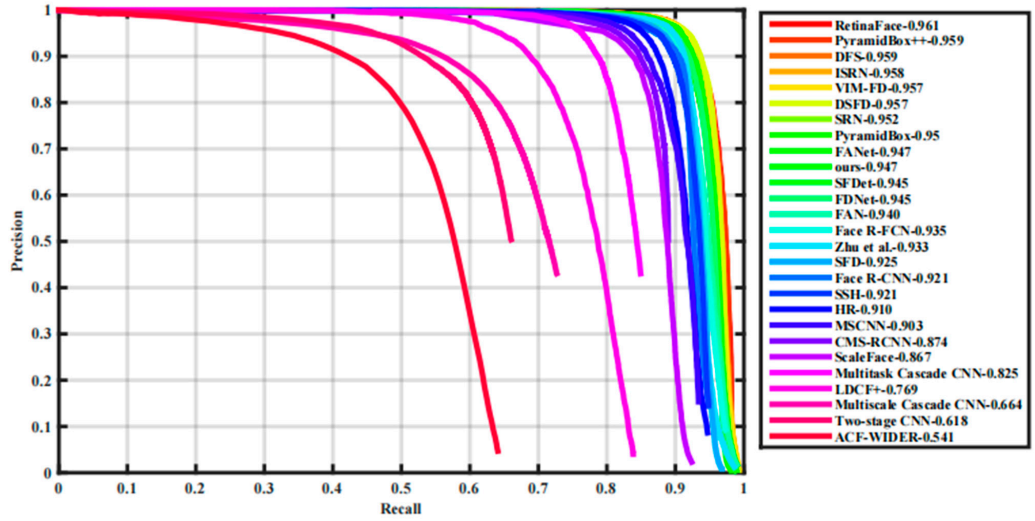
4.3.2. Comparison with YOLOv5

We select images from the WIDER-FACE dataset with the same characteristics as the classroom environment and compare our algorithm with YOLOv5. In Figure 8, the green rectangle is the result of YOLOv5, and the red rectangle represents the result of AMFN-YOLOv5. The figure demonstrates that YOLOv5 has a higher missed detection

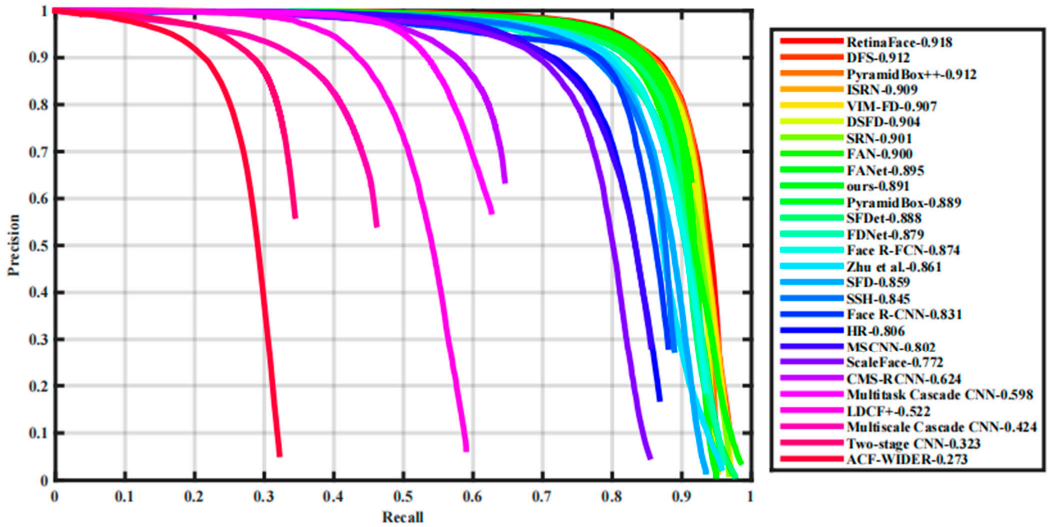
rate when the faces are denser, and AMFN-YOLOv5 is effective for small face detection in dense scenes.



(**a**) Validation−Easy



(**b**) Validation−Medium



(**c**) Validation−Hard

Figure 7. PR curves of the proposed and SOTA method (This is a comparison experiment with the advanced algorithm respectively on the easy, medium and hard data of the validation set).

Figure 8. Face detection on the WIDER-FACE dataset.

4.4. Experiment and Analysis on Classroom-Face Dataset

4.4.1. Comparison with YOLOv5

The previous sections demonstrate that our algorithm has good performance. In this summary, we experimentally test the performance of the original YOLOV5 and our AMFN-YOLOV5 on the Classroom-Face dataset. As shown in Table 3, our algorithm shows excellent performance regardless of Recall, Precision, and mAP. Among them, the mAP of

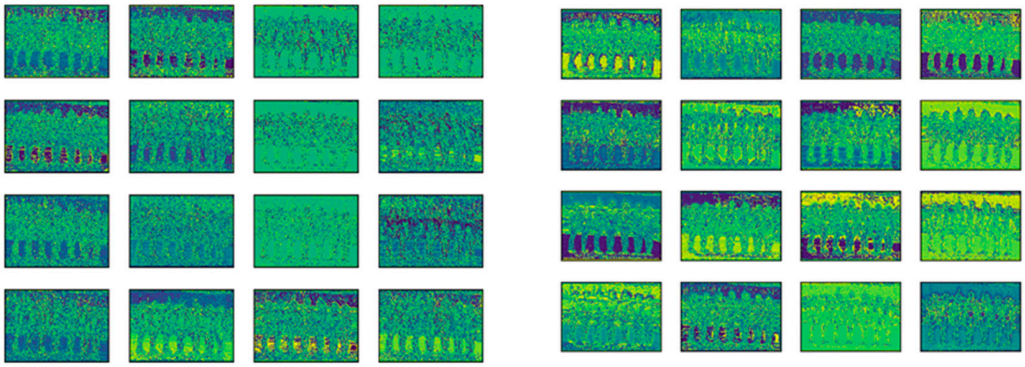
our model increased by 1.7% and 1.69% in natural light and dim light in the easy dataset. In the medium dataset, the mAP increased by 0.64% and 1.67% in natural light and dim light, respectively. In the hard dataset, mAP increased by 1.08% and 4.5%, respectively. This shows that our algorithm is better at detecting small faces in the dimly lit classrooms.

Table 3. Comparison YOLOV5 with AMFN-YOLOv5.

Ours(AMFN-YOLOv5)(%)						
	Easy		**Medium**		**Hard**	
	Daylight	**Dimlight**	**Daylight**	**Dimlight**	**Daylight**	**Dimlight**
Precision	86.40	93.72	87.87	88.66	90.34	89.07
Recall	98.90	96.46	96.53	99.16	96.36	73.99
mAP	96.8	96.28	95.42	98.10	95.36	72.8
YOLOv5						
	Easy		**Medium**		**Hard**	
	Daylight	**Dimlight**	**Daylight**	**Dimlight**	**Daylight**	**Dimlight**
Precision	84.52	89.41	84.63	86.36	83.70	95.92
Recall	97.20	94.77	95.97	97.43	94.48	68.67
mAP	95.10	94.59	94.78	96.43	93.28	68.30

4.4.2. Fusion Experiment

Figure 9 is the feature map extracted by the original backbone network CSP network of YOLOv5 and the improved AMFN network. The feature extraction ability of the person in AMFN is obviously stronger than that inCSP, especially the head feature. The figure demonstrates that the features extracted by AMFN are more abundant and important, which greatly avoids the interference of the background. A cleaner environment is created for subsequent feature fusion, making face detection more robust.



(**a**) CSP_feature (**b**) AMFN_feature

Figure 9. Comparison of feature extraction between old and new backbone networks. Under the feature extraction of the same layer, obviously, the features extracted by AMFN are more clear than those extracted by CSP (CSP_feature refers to the features extracted from CSP network, and AMFN _feature refers to the features extracted from AMFN network).

The results of ablation tests between AMFN-YOLOv5 and YOLOv5 are shown in Table 4. It clearly illustrates that increasing the detection layer can effectively improve the detection accuracy of small faces far away from the camera. Difficult faces in the classroom environment improved accuracy by 1.5% in dim light and 0.75% in natural light. By constructing the AMFN backbone network, the detection accuracy in the dim light environment of the hard subset in the classroom environment is improved by 1.97% based on adding anchors, and the mAP of face detection in natural light is improved by 0.96%. The reason is that through different scales of convolution operations and multi-channel

adaptive fusion, the backbone network can more effectively extract smaller facial feature information. After adding the adaptive pyramid feature fusion strategy, compared with adding the anchor detection layer and optimizing the backbone network, the mAP of each subset is significantly improved. This shows that the network after the fusion of location information and deep feature information is more effective in detecting crowded faces in an unconstrained environment.

Table 4. Comparison with the fusion experiment.

	Add_Anchor (%)					
	Easy		**Medium**		**Hard**	
	Daylight	**Dimlight**	**Daylight**	**Dimlight**	**Daylight**	**Dimlight**
Precision	86.55	92.43	89.48	89.34	91.20	94.47
Recall	97.28	95.26	95.92	97.60	95.03	70.39
mAP	95.29	95.0	94.85	96.80	94.03	69.8
	Add_Anchor+AMFN (%)					
	Easy		**Medium**		**Hard**	
	Daylight	**Dimlight**	**Daylight**	**Dimlight**	**daylight**	**Dimlight**
Precision	86.71	92.68	87.25	87.36	90.07	93.45
Recall	98.84	96.03	96.55	87.22	95.88	72.49
mAP	96.6	95.57	95.25	97.41	94.99	71.77
	Add_Anchor+AMFN+f-ASFF(AMFN-YOLOv5) (%)					
	Easy		**Medium**		**Hard**	
	Daylight	**Dimlight**	**Daylight**	**Dimlight**	**Daylight**	**Dimlight**
Precision	86.40	93.72	87.87	88.66	90.34	89.07
Recall	98.90	96.46	96.53	99.16	96.36	73.99
mAP	96.8	96.28	95.42	98.10	95.36	72.8

4.4.3. Comparison with Open-Source Algorithms

This paper compares our algorithm with the existing open-source algorithms Pymidbox and Facebox for frame-based face detection on GitHub. For fairness, the confidence is set at 0.6 and the IoU threshold at 0.5. Table 5 shows that our algorithm achieves better performance on the three subsets of Easy/Medium/Hard.

Table 5. Comparison with open-source algorithms.

Method	**Easy (%)**		**Medium (%)**		**Hard (%)**	
	Daylight	**Dimlight**	**Daylight**	**Dimlight**	**Daylight**	**Dimlight**
Ours	**96.8**	**96.28**	**95.42**	**98.10**	**95.36**	**72.8**
Pyramidbox [26]	95.78	94.71	95.19	98.19	91.99	61.44
Facebox [21]	82.74	66.32	81.87	82.64	55.49	32.8
Centerface	94.39	95.6	87.71	95.63	89.94	11.8
S3FD	95.24	94.0	93.48	97.83	90.42	55.91
MTCNN	85.76	77.57	79.85	86.98	58.21	10.95
Retinaface	95.01	95.16	94.19	98.08	92.49	58.81

Figure 10 is the result of AMFN-YOLOv5 on the Classroom-Face dataset. Our algorithm can accurately locate and identify students' faces in the classroom environment regardless of whether there is sufficient light in the classroom or whether the scales of students' faces are consistent.

(**a**) Easy_daylight (**b**) Easy_dimlight

(**c**) Medium_daylight (**d**) Medium_dimlight

(**e**) Hard_daylight (**f**) Hard_dimlight

Figure 10. Face detection on Classroom-Face dataset (We name each part of the dataset based on the above characteristics: Easy-daylight, Easy-dimlight, Medium-daylight. Medium-dimlight, Hard-daylight, Hard-dimlight).

5. Conclusions

Face detection in the real world is still very challenging due to the influence of light, human head pose, etc. To address the issue of tiny face detection in low light in the classroom, this research suggests a new face detection technique due to YOLOv5. First, the anchor frame for small face detection is added. Second, the AMFN backbone network is proposed to adaptively fuse multi-scale face information and improve the feature expression ability of small faces. Finally, a four-layer adaptive spatial feature fusion technique is suggested to increase the detection rate of small faces by considering both the semantic content of deep features and the position information of shallow features. According to experimental results, our method performs better in tiny face detection on the WIDER-FACE dataset and the Classroom-Face dataset as compared to other methods.

6. Discussion

Compared with the existing face detection-related research, our work makes up for the vacancy of classroom face detection, creatively proposes a Classroom-Face dataset, and proposes an adaptive fusion face feature for the classroom environment. This work solves the problem of the fusion of facial features at different scales. However, there are still shortcomings in our research. The next research direction is to reduce the size of the model and apply it to mobile devices.

Author Contributions: Writing—original draft, C.S.; Validation, P.W.; Writing—review & editing, S.Z. and X.W., Investigation, J.Z. and H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Natural Science Foundation of Hunan Province (2021JJ30456, 2021JJ30734), the Open Research Project of the State Key Laboratory of Industrial Control Technology (No. ICT2022B60), and the National Defense Science and Technology Key Laboratory Fund Project (2021-KJWPDL-17), the National Natural Science Foundation of China (61972055).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: http://shuoyang1213.me/WIDERFACE/ (accessed on 30 October 2020).

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

References

1. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
2. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
3. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Cham, Switzerland, 2016; pp. 21–37.
4. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
5. Jain, V.; Learned-Miller, E. *FDDB: A Benchmark for Face Detection in Unconstrained Settings*; UMass Amherst Technical Report; Bepress: Berkeley, CA, USA, 2010; Volume 2, Available online: https://works.bepress.com/erik_learned_miller/55/ (accessed on 1 March 2023).
6. Yang, S.; Luo, P.; Loy, C.C.; Tang, X. Wider face: A face detection benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 5525–5533.
7. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
8. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Springer International Publishing: Cham, Switzerland, 2014; pp. 740–755.
9. Li, H.; Lin, Z.; Shen, X.; Brandt, J.; Hua, G. A convolutional neural network cascade for face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 5325–5334.
10. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
11. Yang, X.B.; Zhang, W. Heterogeneous face detection based on multi-task cascaded convolutional neural network. *IET Image Process.* **2022**, *16*, 207–215. [CrossRef]
12. Zhang, L.; Wang, H.; Chen, Z. A Multi-task Cascaded Algorithm with Optimized Convolution Neural Network for Face Detection. In Proceedings of the Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS), Shenyang, China, 22–24 January 2021.
13. Tomar, P.; Singh, R.C. Cascade-based Multimodal Biometric Recognition System with Fingerprint and Face. *Macromol. Symp.* **2021**, *397*, 2000271. [CrossRef]
14. Wang, H.; Li, Z.; Ji, X.; Wang, Y. Face r-cnn. *arXiv* **2017**, arXiv:1706.01061.
15. Zhu, C.; Zheng, Y.; Luu, K.; Savvides, M. Cms-rcnn: Contextual multi-scale region-based cnn for unconstrained face detection. In *Deep Learning for Biometrics*; Springer: Cham, Switzerland, 2017; pp. 57–79.
16. Wu, W.; Yin, Y.; Wang, X.; Xu, D. Face detection with different scales based on faster R-CNN. *IEEE Trans. Cybern.* **2018**, *49*, 4017–4028. [CrossRef] [PubMed]
17. Zhang, C.; Xu, X.; Tu, D. Face detection using improved faster rcnn. *arXiv* **2018**, arXiv:1802.02142.
18. Cakiroglu, O.; Ozer, C.; Gunsel, B. Design of a deep face detector by mask R-CNN. In Proceedings of the 2019 27th Signal Processing and Communications Applications Conference (SIU), Sivas, Turkey, 24–26 April 2019; pp. 1–4.
19. Najibi, M.; Samangouei, P.; Chellappa, R.; Davis, L.S. Ssh: Single stage headless face detector. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4875–4884.
20. Zhang, S.; Zhu, X.; Lei, Z.; Shi, H.; Wang, X.; Li, S.Z. S3fd: Single shot scale-invariant face detecto. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 192–201.
21. Zhang, S.; Zhu, X.; Lei, Z.; Shi, H.; Wang, X.; Li, S.Z. Faceboxes: A CPU real-time face detector with high accuracy. In Proceedings of the 2017 IEEE International Joint Conference on Biometrics (IJCB), Denver, CO, USA, 1–4 October 2017; pp. 1–9.

22. Hu, P.; Ramanan, D. Finding tiny faces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 951–959.
23. Deng, J.; Guo, J.; Ververas, E.; Kotsia, I.; Zafeiriou, S. Retinaface: Single-shot multi-level face localisation in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2020, Seattle, WA, USA, 13–19 June 2020; pp. 5203–5212.
24. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
25. Tang, C.; Chen, S.; Zhou, X.; Ruan, S.; Wen, H. Small-scale face detection based on improved R-FCN. *Appl. Sci.* **2020**, *10*, 4177. [CrossRef]
26. Tang, X.; Du, D.K.; He, Z.; Liu, J. Pyramidbox: A context-assisted single shot face detector. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 797–813.
27. Saha, O.; Kusupati, A.; Simhadri, H.V.; Varma, M.; Jain, P. RNNPool: Efficient non-linear pooling for RAM constrained inference. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 20473–20484.
28. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
29. Sombatpiboonporn, P.; Tian, F.; Zhang, J.; Liu, X.; Jing, W. Human Segmentation for Classroom Video: Dealing with the small size overlapped and distorted human. In Proceedings of the 2021 IEEE International Conference on e-Business Engineering (ICEBE), Guangzhou, China, 12–14 November 2021; pp. 27–34. [CrossRef]
30. Karnalim, O.; Budi, S.; Santoso, S.; Handoyo, E.D.; Toba, H.; Nguyen, H.; Malhotra, V. Face-face at classroom environment: Dataset and exploration. In Proceedings of the 2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA), Xi'an, China, 7–10 November 2018; pp. 1–6.
31. Hu, J.H. *Research of Student Position Detection on Face Recognition in Video Streaming*; Central China Normal University: Wuhan, China, 2019.
32. Gu, M.; Liu, X.; Feng, J. Classroom face detection algorithm based on improved MTCNN. *Signal Image Video Process.* **2022**, *16*, 1355–1362. [CrossRef]
33. kumar Pandey, R.; Faridi, A.A.; Shrivastava, G. SAttentiveness Measure in Classroom Environment using Face Detection. In Proceedings of the 2021 6th International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 20–22 January 2021; pp. 1053–1058.
34. Alexander, A.D.; Salkiawati, R.; Lubis, H.; Rahman, F.; Herlawati, H.; Handayanto, R.T. Local Binary Pattern Histogram for Face Recognition in Student Attendance System. In Proceedings of the 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), Yogyakarta, Indonesia, 15–16 September 2020; pp. 152–156.
35. Chen, L. Evaluation technology of classroom students' learning state based on deep learning. *Computat. Intell. Neurosci.* **2021**, *2021*, 6999347. [CrossRef] [PubMed]
36. Wu, B.; Wang, C.; Huang, W.; Huang, D.; Peng, H. Recognition of Student Classroom Behaviors Based on Moving Target Detection. *Traitement Signal* **2021**, *38*, 215–220. [CrossRef]