Lightweight Neural Network for Real-Time Crack Detection on Concrete Surface in Fog

Gang Yao, Yujia Sun, Yang Yang* and Gang Liao

School of Civil Engineering, Chongqing University, Chongqing, China

Cracks are one of the most common factors that affect the quality of concrete surfaces, so it is necessary to detect concrete surface cracks. However, the current method of manual crack detection is labor-intensive and time-consuming. This study implements a novel lightweight neural network based on the YOLOv4 algorithm to detect cracks on a concrete surface in fog. Using the computer vision algorithm and the GhostNet Module concept for reference, the backbone network architecture of YOLOv4 is improved. The feature redundancy between networks is reduced and the entire network is compressed. The multi-scale fusion method is adopted to effectively detect cracks on concrete surfaces. In addition, the detection of concrete surface cracks is seriously affected by the frequent occurrence of fog. In view of a series of degradation phenomena in image acquisition in fog and the low accuracy of crack detection, the network model is integrated with the dark channel prior concept and the Inception module. The image crack features are extracted at multiple scales, and BReLU bilateral constraints are adopted to maintain local linearity. The improved model for crack detection in fog achieved an mAP of 96.50% with 132 M and 2.24 GMacs. The experimental results show that the detection performance of the proposed model has been improved in both subjective vision and objective evaluation metrics. This performs better in terms of detecting concrete surface cracks in fog.

Keywords: crack detection, deep learning, concrete surface, improved YOLOv4, ghostnet, dark channel prior

INTRODUCTION

Controlling concrete surface quality is one of the main challenges facing the concrete industry. High quality concrete surfaces leave an aesthetically pleasing impression, so architects and building owners are getting stricter about concrete surface quality (Chen et al., 2019; Wei et al., 2019). Crack, one of the most common affecting factors for concrete surface quality, has a significance impact on the safety and sustainability of concrete buildings. Therefore, crack detection plays an essential role in maintaining buildings.

Traditionally, human visual inspection was often used to assess defects on concrete surfaces (Peng et al., 2020). Nevertheless, the judgment conclusions drawn by different people are diverse under the identical concrete surface conditions (Laofor and Peansupap, 2012). Furthermore, the above method generally requires more labor force and time, and it does not produce consistent quantitative objective results. Hence, automatic defect inspection is extremely feasible to assess defects more efficiently and objectively.

In comparison with the deficiencies of the traditional human visual identification methods, there has been extensive research on the computer-based methods. Scholars have proposed a mass of

damage detection methods based on image processing techniques (IPT) have been proposed. Obviously, IPT is preponderant in identifying various surface defects. Yeum et al. once applied IPT to detect cracks (Yeum and Dyke, 2015), while integrating with sliding window technology. In the present study, the potentials of IPT are embodied distinctly. In recent years, many studies (Nishikawa et al., 2012; Choi et al., 2017) based on IPT have been carried out to replace human visual inspection. However, the detection performance is severally weakened in case the intensities of some noisy pixels are lower than those of crack pixels. Given edges and cracks are morphologically similar to a large extent, many researchers (Salman et al., 2013; Zalama et al., 2014) adopt filter-based methods special for edge detection to detect pavement cracks. The IPT-based method is effective and fast, but its robustness is still far from enough in the event of noises (mainly generated by lighting and distortion), which seriously affects the results (Koziarski and Cyganek, 2017). Denoising technology can overcome these problems desirably. Total variation image denoising is a prevailing method to reduce the noises of image data, thereby enhancing the edge detectability of images (Beck and Teboulle, 2009). Owing to the significant changes in the image data captured in real engineering, the application of transcendental knowledge in IPT is restricted. These traditional crack detection methods are obviously defective: each method is designed for a specific database or setting. The crack detectors often do not work, once the setting or database is changed. Moreover, it is difficult to extract semantic information (width and location of cracks, etc.) from images. In order to help inspectors detect defects, image processing algorithms are usually used. But the final results are still obtained relying on manual judgment (Oh et al., 2009).

At present, image acquisition equipment and computing capabilities are increasing improved, a host of machine learning algorithms (such as deep learning) have been used to recognize objects with acceptable results (Ciresan et al., 2012; He et al., 2015; Krizhevsky et al., 2017; Zhang et al., 2021). Deep learning techniques are data-driven approaches which do not require manually-designed rules. When building the model, it is just necessary to select a proper network structure for model output evaluation and a reasonable optimization algorithm. Wide attention has been attracted to the Convolutional Neural Network (CNN), as an effective recognition method (Lecun et al., 2015). In addition, it has been highlighted in image classification and object detection (Ren et al., 2017). A deep-learning-based method was developed to detect concrete bugholes (Wei et al., 2019; Yao et al., 2019; Wei et al., 2021), concrete cracks (Chen and Jahanshahi, 2018; Dung and Anh, 2019; Sun et al., 2021; Tang et al., 2021a; Yao et al., 2021), pavement cracks (Ji et al., 2020; Mei and Gül, 2020; Guan et al., 2021), and other defects (Lin et al., 2017; Cha et al., 2018; Li et al., 2019; Tang et al., 2021b; Jiang et al., 2021). The existing crack detection methods based on CNN generally have problems, such as complex network structures and excessive training parameters. One crucial technical problem is balancing the efficiency and accuracy of the detection.

At present, the mainstream framework of object detection based on deep learning is mainly divided into two categories: two-stage, based on the idea of target region proposal, and one-stage,

based on the idea of regression. Target region proposals in the two-stage category are extracted first, and the detection model is then trained based on them (such as RCNN (Girshick et al., 2013), Fast-RCNN (Girshick, 2015) and Faster-RCNN) (Ren et al., 2017). The one-stage category does not have the extraction operation of the target region proposal, and the target category and location information are directly generated by the detection network (such as SSD (Liu et al., 2016), YOLOv3 (Redmon and Ali, 2018) and YOLOv4 (Bochkovskiy et al., 2020)). Two-stage has higher task accuracy but slower speed, while one-stage can achieve real-time performance at the expense of accuracy. Therefore, in order to balance the detection efficiency and accuracy of the concrete surface cracks, the GhostNet Module concept (Han et al., 2020) is used for reference, and the backbone network architecture of YOLOv4 is improved so as to use fewer parameters to generate more features. The feature redundancy between networks is reduced, and the whole network is compressed. The multi-scale fusion method is adopted to effectively detect cracks on concrete surfaces, greatly overcoming the difficulty of manual detection.

At present, most relevant studies are carried out under conditions wherein the image information is clear and obvious. With the frequent occurrence of adverse weather (such as haze), the collected images will be degraded by the loss of detailed information, color distortion and image resolution. The recognition method of the network model in the clear image scene appears to be slightly lacking in practicality. When the target features of the collected images in fog are unclear and the resolution is problematic, the lightweight YOLOv4 network model is directly used for crack detection, and the detection performance will be reduced. Consequently, in this paper, the network model is integrated with the dark channel prior concept (Kaiming et al., 2011) and the Inception module (Szegedy et al., 2016). The image crack features are extracted at multiple scales, and BReLU bilateral constraints are adopted to maintain local linearity. Based on the atmospheric scattering model (Nayar and Narasimhan, 1999) and the improved YOLOv4 model, cracks on concrete surfaces can be detected effectively in fog. The proposed network model is compared with the YOLOv3 algorithm in subjective vision and objective evaluation metrics. The results show that the proposed method has better detection performance, and the model parameters and amount of calculations are greatly reduced.

THE LIGHTWEIGHT MODEL FOR CONCRETE CRACK DETECTION

The Principles of YOLOv4

The backbone network CSPDarknet53 of YOLOv4 is the core of the algorithm and is used to extract the target features. CSPNet can maintain accuracy and reduce computing bottlenecks and memory costs while being simplified. Drawing from the experience of CSPNet, YOLOv4 adds CSP to each large residual block of Darknet53. It divides the feature mapping of the base layer into two parts, then merges them through a cross-stage hierarchical structure to reduce the amount of calculations

FIGURE 1 | YOLOv4 network architecture.

FIGURE 2 | ResBlock module structure.
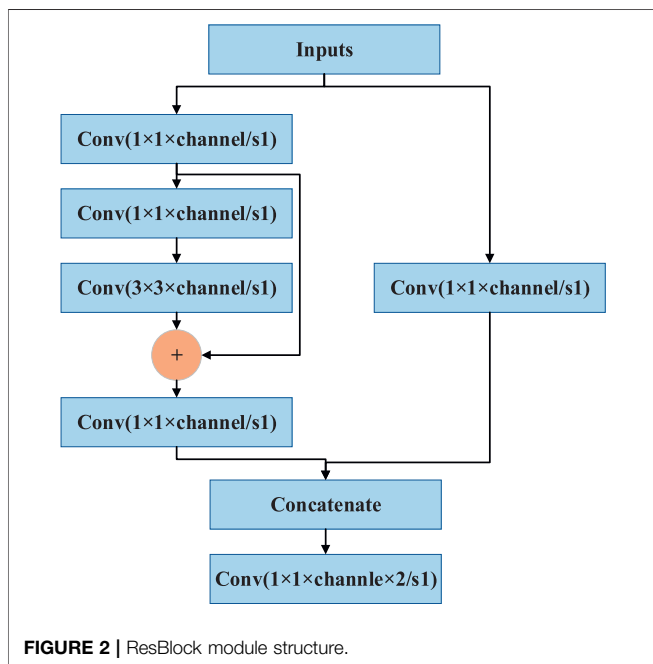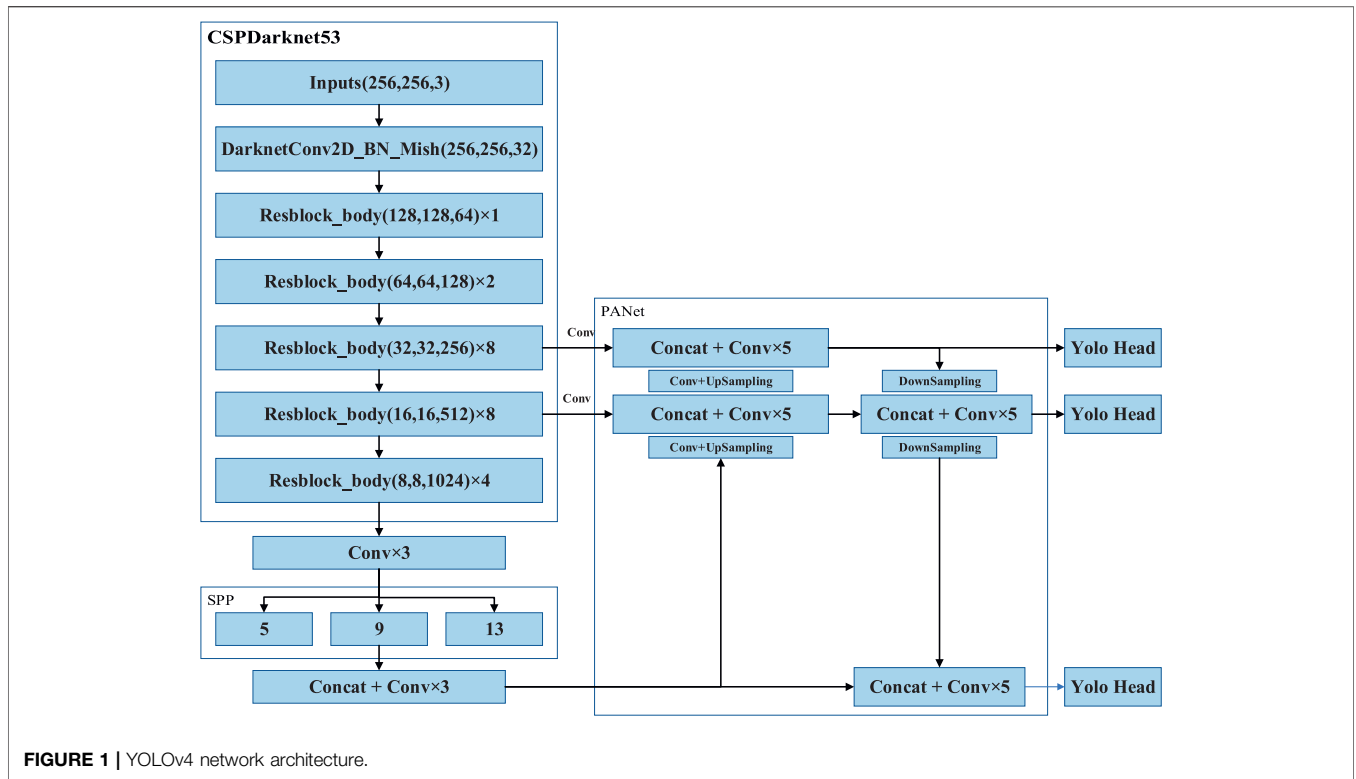
while ensuring accuracy. The activation function of CSPDarknet53 uses the Mish activation function, and the subsequent network uses the Leaky ReLU function. The experiments demonstrated that this setting had higher accuracy in object detection. Unlike the YOLOv3 algorithm, which uses FPN for upsampling, YOLOv4 draws on the idea

of information circulation in the PANet network. Firstly, the semantic information of the layer features is propagated to the low-level network by upsampling, and is then fused with the high-resolution information of the underlying features to improve the small target detection effect. Next, the information transmission path from the bottom to the top is increased, and the feature pyramid is enhanced through downsampling. Finally, the feature maps of different layers are fused to make predictions. The specific network framework is shown in **Figure 1** (Bochkovskiy et al., 2020). The ResBlock_body is the residual block of CSPDarknet53, which can extract the target features of the image and reduce the computational bottleneck and the memory cost. The specific internal Module architecture is shown in **Figure 2** (Yao et al., 2021).

The Principles of the Ghost Module

The Ghost Module can use fewer parameters to generate more feature maps. Specifically, in view of the large amount of redundancy in the intermediate feature maps calculated by mainstream CNN, the deep neural network divides the ordinary convolutional layer into two parts. In the first part, the number of convolutions is strictly controlled, and the inherent feature maps are extracted by ordinary convolution operations. In the second part, a series of simple linear operations are used to generate more feature maps. Compared with the ordinary convolutional neural network, the total number of parameters and computational complexity required in the Ghost module are reduced without changing the size of the output feature map. The specific implementation is shown in **Figure 3** (Han et al., 2020).
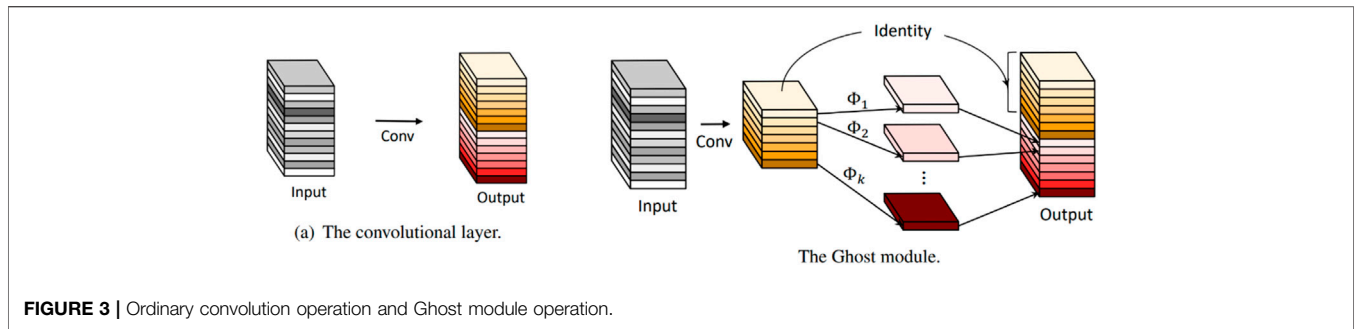
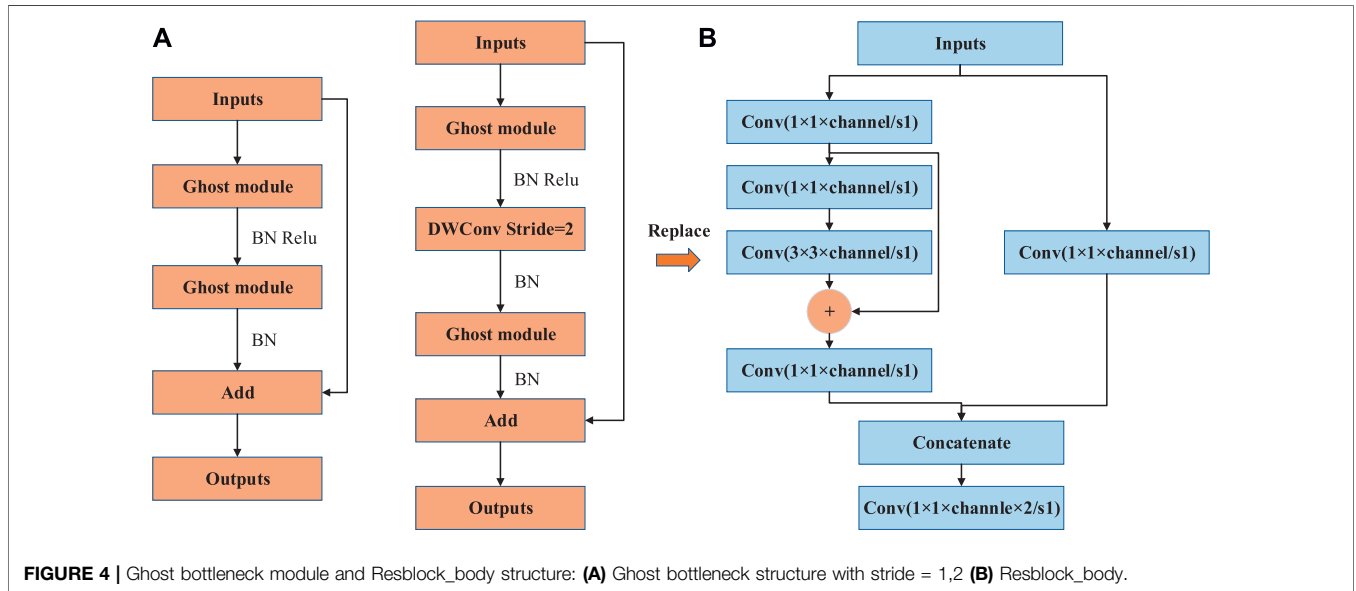**FIGURE 3 |** Ordinary convolution operation and Ghost module operation.

FIGURE 4 | Ghost bottleneck module and Resblock_body structure: **(A)** Ghost bottleneck structure with stride = 1,2 **(B)** Resblock_body.

The memory and computation benefits achieved using the Ghost module are analyzed. The input data is $X \in R^{c \times h \times w}$, where c is the number of channels, w is the width and h is the height. $Y \in R^{n \times h \times w}$ is the output feature map with n channels, and $f \in R^{c \times k \times k \times n}$ is the convolutional kernel size. The FLOPs required by the ordinary convolution process are $n \times h \times w \times c \times k \times k$.

The Ghost module believes that the output feature maps are transformed by a few original feature maps through some cheap operations such as linear transformation. The size of the ordinary convolution kernel is $f' \in R^{c \times k \times k \times m}$, where $m \leq n$, and the output feature map is $Y' \in R^{h' \times w' \times m}$. In this way, in order to further obtain the required n feature maps, a series of cheap linear operations are used to generate s duplicate feature maps for the original map in Y', and the average kernel size of each linear operation is $d \times d$. The theoretical speedup ratio when using the Ghost module to upgrade ordinary convolution is computed by **Eq. 1** The magnitude of $d \times d$ is similar to $k \times k$ and $s \ll c$. The compression ratio of the parameter is computed by **Eq. 2** The compression ratio of the parameter is approximately equal to the speedup ratio.

$$r_s = \frac{n \times h' \times w' \times c \times k \times k}{\frac{n}{s} \times h' \times w' \times c \times k \times k + (s-1) \times \frac{n}{s} \times h' \times w' \times d \times d}$$

$$= \frac{c \times k \times k}{\frac{1}{s} \times c \times k \times k + \frac{s-1}{s} \times d \times d} \approx \frac{s \times c}{s+c-1} \approx s \qquad (1)$$

$$r_c = \frac{n \times c \times k \times k}{\frac{n}{s} \times c \times k \times k + (s-1) \times \frac{n}{s} \times d \times d} \approx \frac{s \times c}{s+c-1} \approx s \qquad (2)$$

The Lightweight YOLOv4 Model Structure
The Improvement of the YOLOv4 Model Framework
The Ghost module is applied to generate the same number of feature maps as the ordinary convolutional layer. It can easily replace the convolutional layer and integrate it into the existing designed neural network structure to reduce the computational cost. The Ghost module is utilized to build a Ghost bottleneck structure to replace the Resblock_body bottleneck structure in the YOLOv4 network model, which can further eliminate feature redundancy.

The Ghost bottleneck constructed using the Ghost module concept is shown in **Figure 4**. Similar to the basic residual block in ResNet, it integrates multiple convolutional layers and shortcuts.
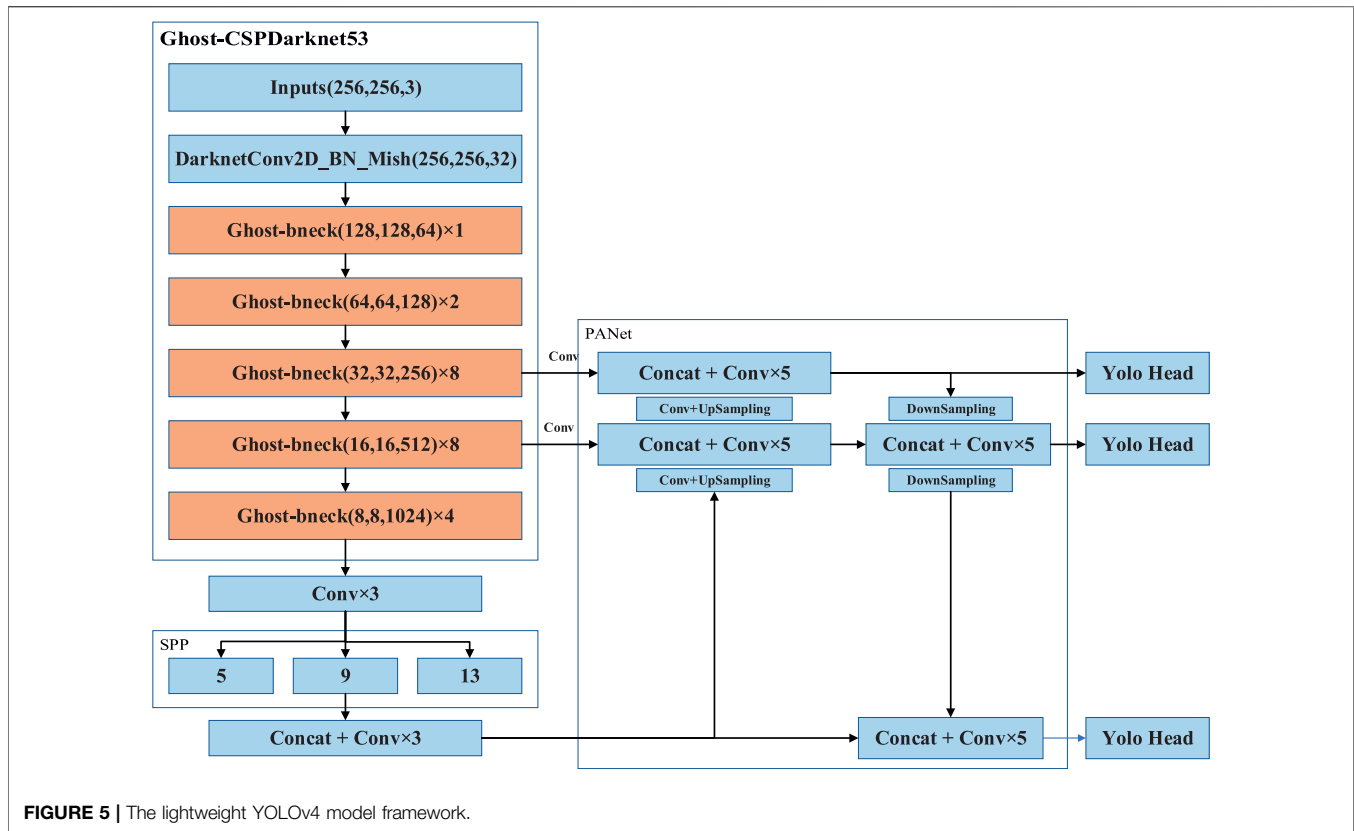
FIGURE 5 | The lightweight YOLOv4 model framework.

The Ghost bottleneck structure is mainly constructed by two stacked Ghost modules. The first Ghost module is used as an extension layer which increases the number of channels. The second one reduces the number of channels to match the shortcut path. The shortcut is then used to connect the input and output of the two Ghost modules, and the MobileNetv2 module concept is used for reference. The ReLU structure is not used after the second Ghost module. For the Ghost module with stride = 2, the shortcut path is implemented by the separable convolution of the downsampling layer and stride = 2.

In this section, following the advantages of the basic SPP and PANet architecture in the YOLOv4 module, the Ghost bottleneck is used to replace the Resblock_body structure in the YOLOv4 module. The specific replacement network structure is shown in **Figure 5**.

The Fast Non-Maximum Suppression Algorithm

The traditional Non-Maximum Suppression (NMS) (Neubeck and Gool, 2006) algorithm arranges items in descending order according to the confidence scores of the detected target boxes. It sets an Intersection over Union (IoU) threshold and removes bounding boxes larger than the threshold. Until all the prediction boxes are traversed, the remaining bounding boxes are taken as the final target detection result. Since it is a sequential traversal, each category needs to be sorted and filtered, which will result in a loss of algorithm speed. In this study, considering the image characteristics of the concrete surface cracks, parallel processing is adopted to screen and retain each boundary box in parallel.

Firstly, all unfiltered network prediction boxes are input, and are arranged in descending order according to the confidence scores. Only the first N detection results are selected. The IoU for the crack prediction results is then calculated to obtain the IoU matrix. The diagonal elements and the lower triangle are self-intersecting and recalculating, and are set to 0. The maximum value of the IoU matrix is calculated, and the boxes outside the threshold limit are filtered out. As a result, the final screening result is the final recognition result.

THE CONCRETE CRACK DETECTION MODEL IN FOG

With the frequent occurrence of adverse weather such as haze, the collected images will be degraded by low image resolution and the loss of detailed information. The YOLOv4 network model lacks some practicability when applied to crack detection in clear image scenes, and its detection performance will decrease. This section focuses on a series of degradation phenomena in image acquisition and the low detection rate of cracks in adverse weather. Dark channel prior and the Inception module are used for reference to integrate into the network model. The image crack features are extracted at multiple scales, and BReLU bilateral constraints are adopted to maintain local linearity. The clear crack scene is restored based on the atmospheric scattering model, and the crack structure is effectively detected by combining with the improved YOLOv4 network model.

FIGURE 6 | Atmospheric scattering imaging link.

FIGURE 7 | Clear image and dark channel image.
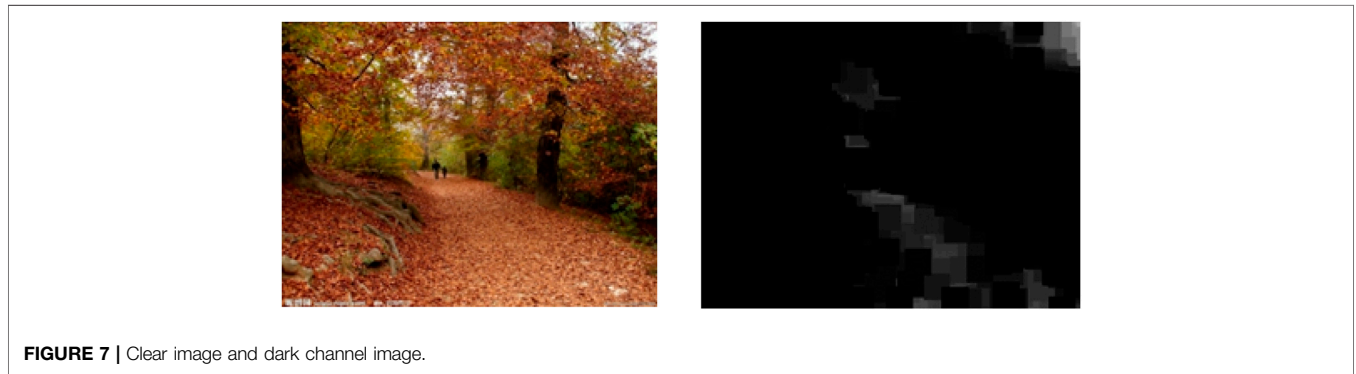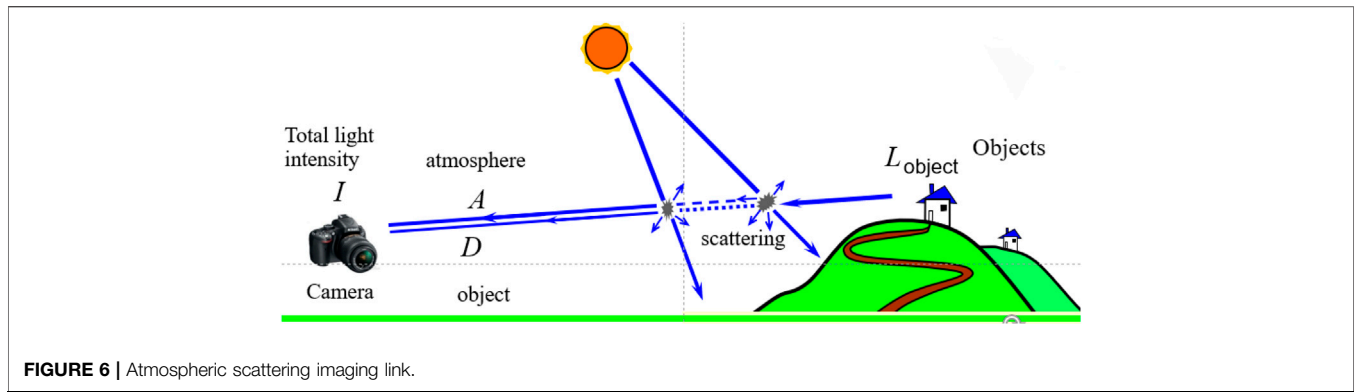
Atmospheric Scattering Model

There are two main factors leading to image degradation in hazy weather. One is that the atmospheric light is scattered by the atmospheric haze particles to produce stray light, which affects the image resolution. The other is that the light reflected from the atmosphere to the target will cause light attenuation through the absorption and scattering of suspended particles, which usually results in blurred image details and decreased contrast. Based on these two factors, Nayar and Narasimhan proposed an atmospheric scattering model (Nayar and Narasimhan, 2002). The imaging link is shown in **Figure 6**. The mathematical expression is shown in **Eq. 3**.

$$\mathbf{I} = L^{object} \cdot t + A = L^{object} \cdot e^{-\beta d} + A_{\infty}\left(1 - e^{-\beta d}\right) \qquad (3)$$

where I and L^{object} represent the light intensity of foggy images and sunny images, respectively; t represents the atmospheric transmittance; A represents the atmospheric light intensity; A_{∞} represents the atmospheric light intensity at infinity; β represents the extinction coefficient of scattering and absorption; d represents the distance between the target and the imaging system. **Eq. 3** reveals that two parameters must be reconstructed if a clear image is to be restored: the atmospheric transmittance t and the atmospheric light intensity at infinity A_{∞}.

Dark Channel Prior

Dark channel prior (Kaiming et al., 2011) is a prior theory obtained from the statistics of a large number of fog-free images. In most local areas of fog-free images, there are always one or more pixels whose gray value is close to 0. The mathematical expression is shown in **Eq. 4**.

$$J^{object}_{dark} = \min_{y \in \Omega(x)} \left\{ \min_{c \in \{r,g,b\}} \left[L^{object}_c \right] \right\} \to 0 \qquad (4)$$

$$t(i,j) \to 1 - \min_c \left(\min_{y \in \Omega(x)} \left(\frac{I(i,j)}{A_{\infty}} \right) \right) \qquad (5)$$

where $\Omega(x)$ is the local area of the fog-free image; L^{object}_c and J^{object}_{dark} represent the original image and the image processed by dark channel, respectively. **Figure 7** presents a dark channel image of a fog-free image. The atmospheric transmittance t can be obtained quickly by substituting the dark channel concept of **Eq. 4** into the atmospheric scattering model, after which the clear image can be restored. The mathematical expression is presented as **Eq. 5** Drawing lessons from the characteristics of the dark channel concept and integrating them into the neural network model is helpful for restoring clear images, and provides a higher detection rate for detecting cracks on concrete surfaces.

Framework of Crack Detection Model
The Improvement of the Model Framework
In this section, the YOLOv4 model architecture is improved to complete the crack detection on concrete surfaces in fog. The input data structure is the RGB concrete fog image, and the transmittance map of the corresponding fog image is expected to
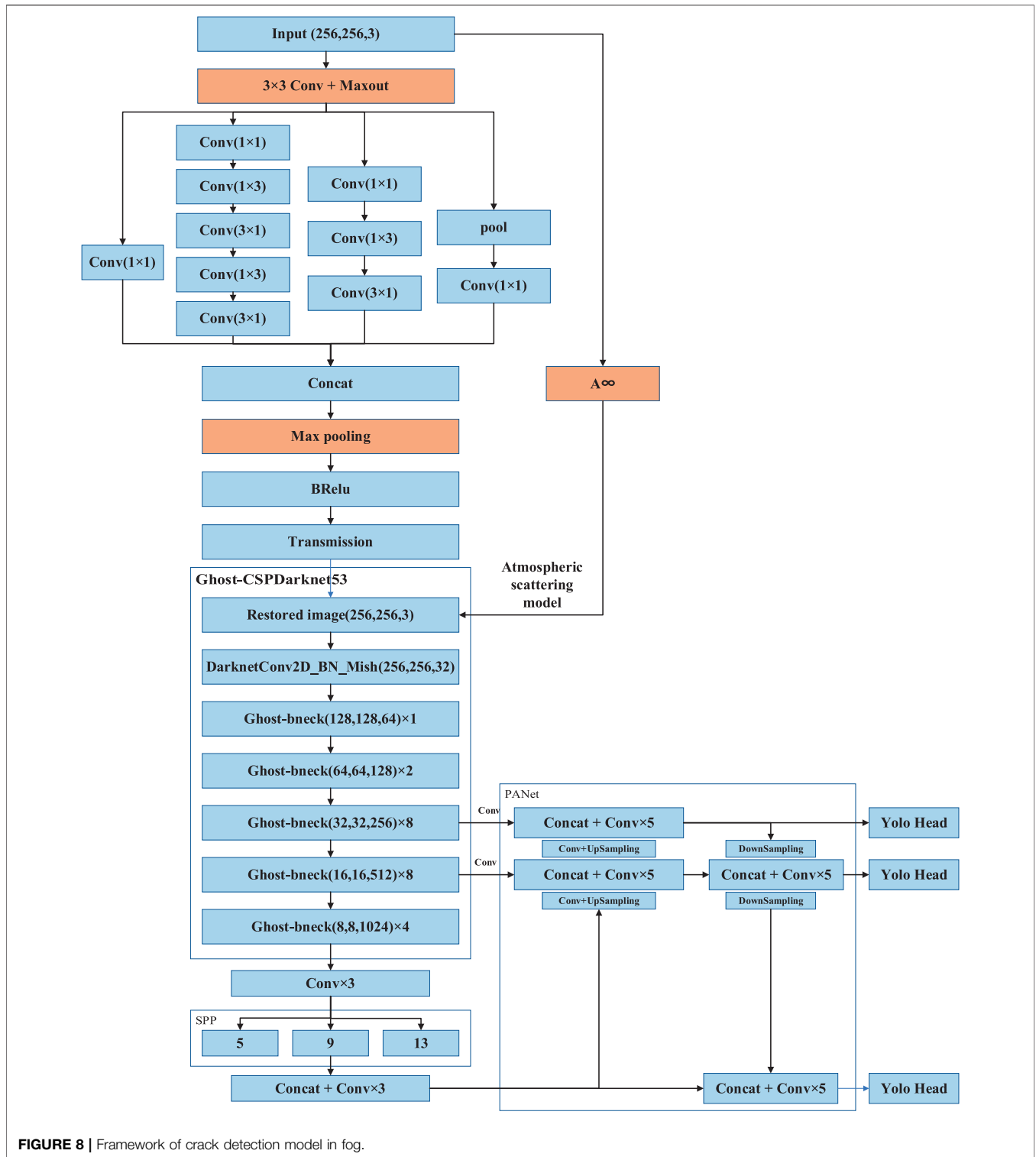
FIGURE 8 | Framework of crack detection model in fog.

be output in the middle. The position information of the image pixels remains unchanged, and the atmospheric light intensity at infinity is then estimated. The atmospheric scattering model is used to restore the image, and the cracks on the concrete surface are detected in combination with the lightweight YOLOv4 model

structure. The specific improved network model is shown in **Figure 8**.

The first layer of the network is the feature extraction layer, which can effectively extract the features of foggy images. Combined with the dark channel prior method, the activation
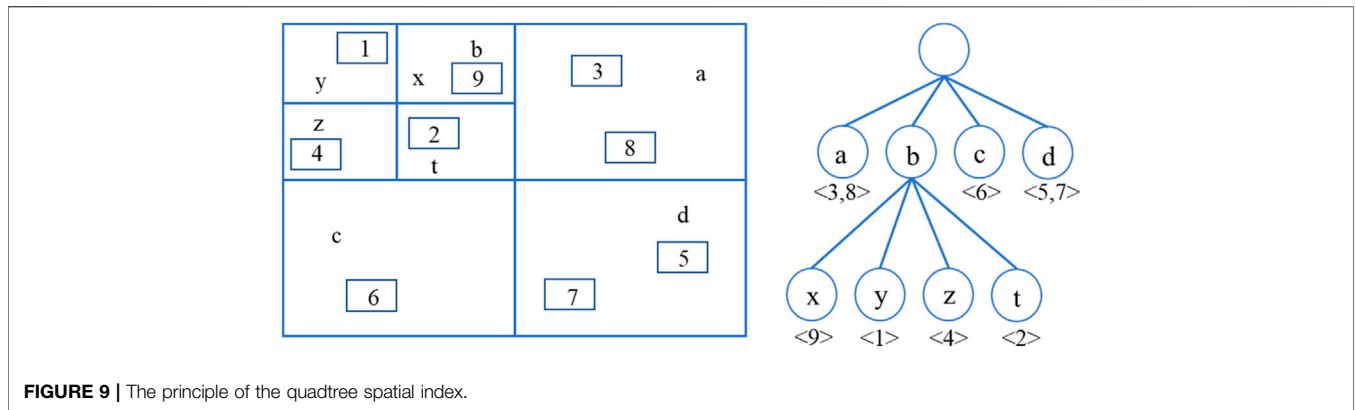
FIGURE 9 | The principle of the quadtree spatial index.

TABLE 1 | The method for estimating atmospheric light intensity at infinity.

Input image：Original foggy image *I* and Dark channel image I_{dark}

(1) Perform median filtering on *I*: $I_{median} = median(I_{dark})$

(2) Perform quadtree segmentation on I_{median}. The image area information is stored in four leaf nodes

(3) The leaf node score is defined as:$I_{score}(\Omega) = ave\{I_{median}(\Omega)\}$;

(4) Select the leaf node with the largest score Ω

(5) Loop 2–5 steps until the area Ω is smaller than the threshold area size ψ of the adaptive image resolution. The threshold area is defined as: $\psi = 0.04 \times min(raw, col)$

(6) Calculate the average value of the corresponding coordinate area in the original foggy image *I* as the atmospheric light intensity at infinity: $A_{\infty} = ave\{I(\Omega)\}$

function of convolution and Maxout is used as the first layer of the network. Firstly, the input foggy image is composed of 16 filters with a convolution kernel size of 3 × 3. Subsequently, based on the idea of image dark channel, the Maxout nonlinear activation function is selected to realize the local minimum filtering function, and the extracted feature map is output.

Maxout divides the feature map *z* extracted by convolution into groups with *k* values, after which the Maxout unit outputs the largest element among them. It is defined as $h = z_j$. It can be seen that the Maxout activation function achieves the fitting of the convex function by separating the *k* terminal and taking its maximum value. On the basis of the Maxout unit, the first layer of the network is designed as shown in **Eq. 6**.

$$F_1^i(x) = \max_{j \in [1,k]} f_1^{i,j}(x), \quad f_1^{i,j} = W_1^{i,j} \times I + B_1^{i,j} \qquad (6)$$

where $W_1^{i,j}$ is an inverse filter with a center value of −1 and a size of 3 × 3. This means that the maximum output after the Maxout activation function is the minimum color channel value, which is the idea of the dark channel. The automatic learning of dark channel features is realized through feature extraction of the first layer.

In the second layer, a multi-scale convolution neural network is used to extract the features of the target. To improve the robustness of the feature extraction under different resolutions, the multi-scale extraction capability of the Inceptionv3 module structure is utilized, and the adaptability to the network width

and depth is increased. It can be seen that the spatial filters (5 × 5, 7 × 7) with larger computing power are replaced by a convolution kernel (3 × 3) with smaller computing power in the second layer network structure, which not only reduces the number of parameters, but also speeds up the computations. Cross-channel information integration can be realized by designing the dimension reduction structure of a 1 × 1 convolution kernel followed by a 3 × 3 convolution kernel. The outputs of the adjacent activation responses are highly correlated, and the local representation ability is not reduced when reducing the number of these activation effects before aggregation. At the same time, the 3 × 3 convolution kernel is decomposed into two one-dimensional convolution kernels 1 × 3 and 3 × 1 by the convolution kernel decomposition design. Not only can this speed up the computations, but it can also increase the depth and nonlinearity of the network.

In the third layer, space invariance is achieved by selecting the maximum value of the neighborhood. Moreover, the local extremum is also consistent with the assumption that the medium transmission is locally constant in foggy weather, and the noise in the transmission image can be suppressed. In the fourth layer, inspired by the ReLU and Sigmoid activation functions, the BReLU activation function is used for nonlinear activation. The range of atmospheric transmittance *t* is 0–1, which cannot be infinite or infinitesimal. Both local linearity and bilateral restrictions are maintained through a 3 × 3 convolution kernel. The transmittance image of the atmosphere can be mapped end to end. Ultimately, the input fog image is used to estimate the atmospheric light intensity at infinity, and the clear image can be restored. The lightweight YOLOv4 network model is utilized to detect concrete surface cracks.

The Estimation of Atmospheric Light Intensity at Infinity A_{∞}

According to the atmospheric scattering model of **Eq. 3**, the accuracy of the atmospheric light intensity estimation at infinity is analyzed. It directly determines the clarity of the restored target. Consequently, the atmospheric light intensity at infinity is directly related to the image clarity. In this paper, based on the dark channel of the haze image and combined with the quadtree spatial index principle, the atmospheric light

TABLE 2 | The initial parameters of the first stage.

Parameter	Value
Base_LR	10^{-3}
Batch_Size	16
Train_Epoch	100
Weight_decay	5×10^{-4}
Lr_scheduler_Max_iterations	5
Lr_scheduler_Minimum_lr	10^{-5}

TABLE 3 | The initial parameters of the second stage.

Parameter	Value
Base_LR	10^{-4}
Batch_Size	16
Train_Epoch	500
Weight_decay	5×10^{-4}
Lr_scheduler_Max_iterations	5
Lr_scheduler_Minimum_lr	10^{-5}

intensity at infinity is estimated. **Figure 9** illustrates the principle of the quadtree spatial index. The specific method of estimation is shown in **Table 1**.

The atmospheric light intensity at infinity is estimated by the above method, and the transmission image is reconstructed with the neural network. Through combination with the atmospheric scattering model of **Eq. 3**, a clear concrete crack image can be recovered, and by combination with the lightweight YOLOv4 network model framework, the crack training and detection can then be carried out.

The Construction of the Network Loss Function

The loss function of the network model is composed of the following parts: the transmittance estimation loss, the regression loss, the confidence loss and the classification loss. The regression loss of the prediction box adopts the CIOU loss function of the YOLOv4 network model. The mathematical expression is shown in **Eqs. 7–9**.

$$L_{CIOU} = 1 - IoU + \frac{d^2}{c^2} + \alpha v \qquad (7)$$

$$v = \frac{4}{\pi^2} \left(arc\,tan\,\frac{w^{gt}}{h^{gt}} - arc\,tan\,\frac{w}{h} \right)^2 \qquad (8)$$

$$\alpha = \frac{v}{1 - IoU + v} \qquad (9)$$

where d represents the Euclidean distance between the center points of the two prediction boxes and c represents the diagonal distance of the closed area of the prediction box.

The confidence loss function adopts cross entropy and is divided into two parts: obj and noobj. In order to reduce the contribution weight of the noobj calculation part, the loss of noobj increases the weight coefficient λ_{noobj}. It is shown in **Eq. 10** The classification loss function adopts cross entropy. When the jth anchor box of the ith network is responsible for a real target, the bounding box generated by this anchor box will

calculate the classification loss function. This is shown in **Eq. 11** The loss of transmittance estimation learns the mapping relationship between the input image and transmittance by minimizing the loss function between the estimated transmittance and the actual transmittance. The mean square error between actual transmittance and predicted transmittance is defined as the loss function. The loss function is defined as **Eq. 12**.

$$\mathbf{L_{conf}} = \sum_{i=0}^{K \times K} \sum_{j=0}^{M} \mathbf{I}_{ij}^{obj} \left[\widehat{\mathbf{C}}_{\mathbf{i}} \log(\mathbf{C}_{\mathbf{i}}) + \left(1 - \widehat{\mathbf{C}}_{\mathbf{i}} \right) \log(1 - \mathbf{C}_{\mathbf{i}}) \right] -$$
$$\boldsymbol{\lambda_{noobj}} \sum_{i=0}^{K \times K} \sum_{j=0}^{M} \mathbf{I}_{ij}^{noobj} \left[\widehat{\mathbf{C}}_{\mathbf{i}} \log(\mathbf{C}_{\mathbf{i}}) + \left(1 - \widehat{\mathbf{C}}_{\mathbf{i}} \right) \log(1 - \mathbf{C}_{\mathbf{i}}) \right] \quad (10)$$

$$L_{classes} = \sum_{i=0}^{K \times K} I_{ij}^{obj} \sum_{c \in classes} \left[\widehat{p}_i(c) log(p_i(c)) + \left(1 - \widehat{p}_i(c) \right) log(1 - p_i(c)) \right] \quad (11)$$

$$L_{trans} = \left[\frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{2} \parallel h_{w,b}(x) - y \parallel^2 \right) \right] + \frac{\lambda}{2} \sum_{l=1}^{n_l-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} \left(w_{ji}^{(l)} \right)^2 \quad (12)$$

In **Eq. 12**, The first term on the right side of the equation is the mean square error term, and the second term is the regular term. The regular term has nothing to do with the bias $b_i^{(l)}$ and can only control the size of the weight, so it is also called the weight attenuation term. The weight decay parameter λ in the weight decay term can be used to determine the proportion of the two items in the loss function. The key of training is to obtain the minimum cost function by continuously adjusting the weight $w_{ij}^{(l)}$ and bias $b_i^{(l)}$.

In this section, the loss function of the crack detection model framework in fog is defined as:

$$L_{total} = L_{CIOU} + L_{conf} + L_{classes} + L_{trans} \qquad (13)$$

EXPERIMENTS

Image Database Creation

A smartphone is used for image acquisition. For the purpose of collecting images of small cracks on a concrete surface, all images are taken from a distance of 0.1 m between the smartphone and concrete surface. 2000 original images (3,024 × 3,024 pixels) are extracted from the surfaces of concrete buildings. Each original image can be cropped to generate 139 images (256 × 256 pixels). However, some cropped images do not include cracks. As a result, the images with cracks are meticulously selected from the cropped image set. Finally, 10,000 images conforming to requirements are selected to create the database.

The actual transmittance of the defogging images needs to be obtained as a label for training. Since the shooting target in actual fog needs to be aligned with the pixel position of the shooting target on a sunny day, there can be no shooting error, and the construction is too arduous. Accordingly, in this paper, images taken on sunny days are manually fogged to build the database.
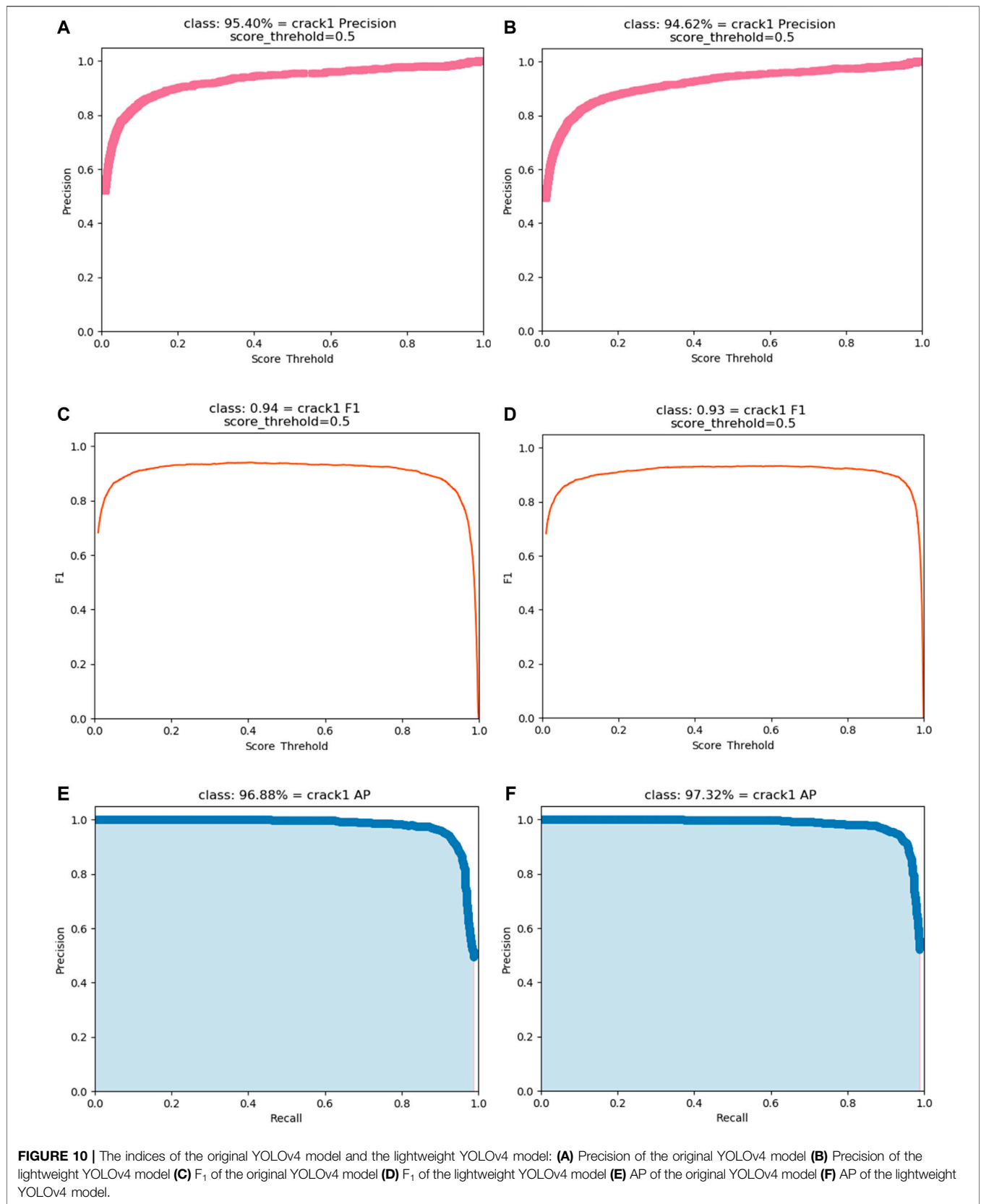
FIGURE 10 | The indices of the original YOLOv4 model and the lightweight YOLOv4 model: **(A)** Precision of the original YOLOv4 model **(B)** Precision of the lightweight YOLOv4 model **(C)** F_1 of the original YOLOv4 model **(D)** F_1 of the lightweight YOLOv4 model **(E)** AP of the original YOLOv4 model **(F)** AP of the lightweight YOLOv4 model.

TABLE 4 | The performance comparison of the lightweight YOLOv4 model with the original YOLOv4 model and the YOLOv3 model.

Algorithm	mAP (%)	Parameters (M)	FLOPs (GMacs)
The YOLOv3 model	93.13	234	32.8
The YOLOv4 model	97.32	244	29.9
The lightweight YOLOv4 model	96.88	121	1.95
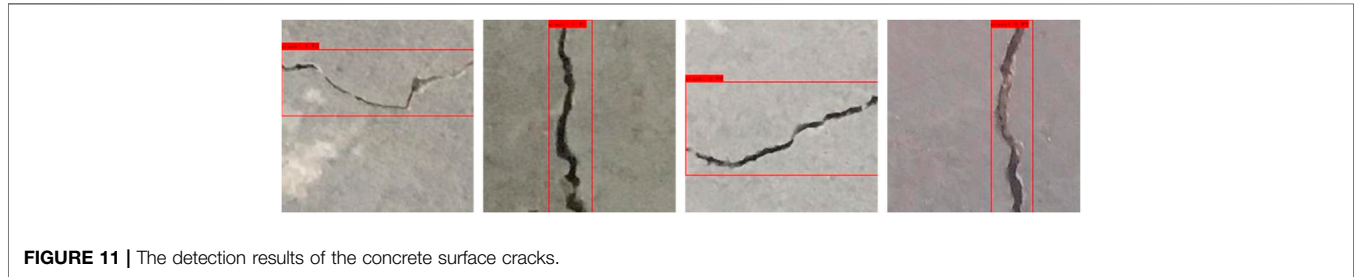
FIGURE 11 | The detection results of the concrete surface cracks.

Matlab is used to realize by adding different degrees of white noise to the image pixel by pixel. Furthermore, with reference to the Mosaic data enhancement method of YOLOv4, four images are randomly selected from the database, randomly scaled, and then randomly distributed for stitching. Not only can this greatly expand the original database, but it can also enrich the background of the images. When performing random cropping, if a part of the label box in the sample is cropped, it will be discarded and the intact label box will be retained after cropping. In the process of random scaling, many small targets are added to balance the scale problem of the original database, and the robustness of the network is better. Another benefit of Mosaic data enhancement is that the data of four pictures can be directly calculated during training, and the batch size can be improved in disguise. Therefore, the Mini-batch size set during training does not need to be large, which reduces the training difficulty of the model.

For the purpose of assessing the generalization ability of the proposed model, 10,000 images are divided into five parts based on the fivefold cross-validation principle, among which 80% are used to train and validate the model and the remaining 20% are used to test. More precisely, 8,000 images are randomly selected from the 10,000 images, among which 7,000 images are used to generate a training set and 1,000 images are used to create a validation set. The remaining 2,000 images not selected for training and validation are used to build a testing set.

Model Initialization

In the process of network training, in order to improve efficiency and better save computing resources and time, this paper adopts the training strategy of freezing some layers. The whole training process is divided into two stages. In the first stage, only the Backbone network structure is trained; in the second stage, the overall network structure is trained. In the training process, the Cosine Annealing learning rate strategy is adopted, and hyperparameters are optimized according to the genetic

algorithm. The initial parameter settings of the first stage and the second stage are shown in **Tables 2**, **3**, respectively.

Evaluation Metrics of Accuracy

For the purpose of assessing the accuracy of any object detection technique, many evaluation criteria are proposed and adopted. The most frequently-used metric for object detection is the mean Average Precision (mAP) which is currently used to measure how the labeling methods perform on a task. Before introducing the mAP, it is necessary to introduce the commonly-used metrics in the field of object detection, such as Intersection over Union (IoU), Precision and Recall. IoU is the ratio of the intersection and union of the candidate bound and the ground truth bound, which is also called the Jaccard index. The classification problem generally sorts the concerned classes into positive classes and other classes into negative classes. The prediction results of the testing set may be correct or wrong, and these results can be divided into four categories: True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). The accuracy refers to the number of correct recognitions of all samples predicted to be positive. According to the above classification, **Eqs. 14–15** define the precision and recall, respectively.

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{P} \tag{15}$$

where P is the number of positive samples in the testing set. The Recall reflects the missing rate of the model. The Precision and Recall are independent of each other. High precision means that the FP rate is low, which can lead to a high missing rate. **Eq. 16** defines the comprehensive evaluation value F_1. Taking the recall as the horizontal axis and the precision as the vertical axis, the P-R curve can be obtained and the Average Precision (AP) can be calculated. In
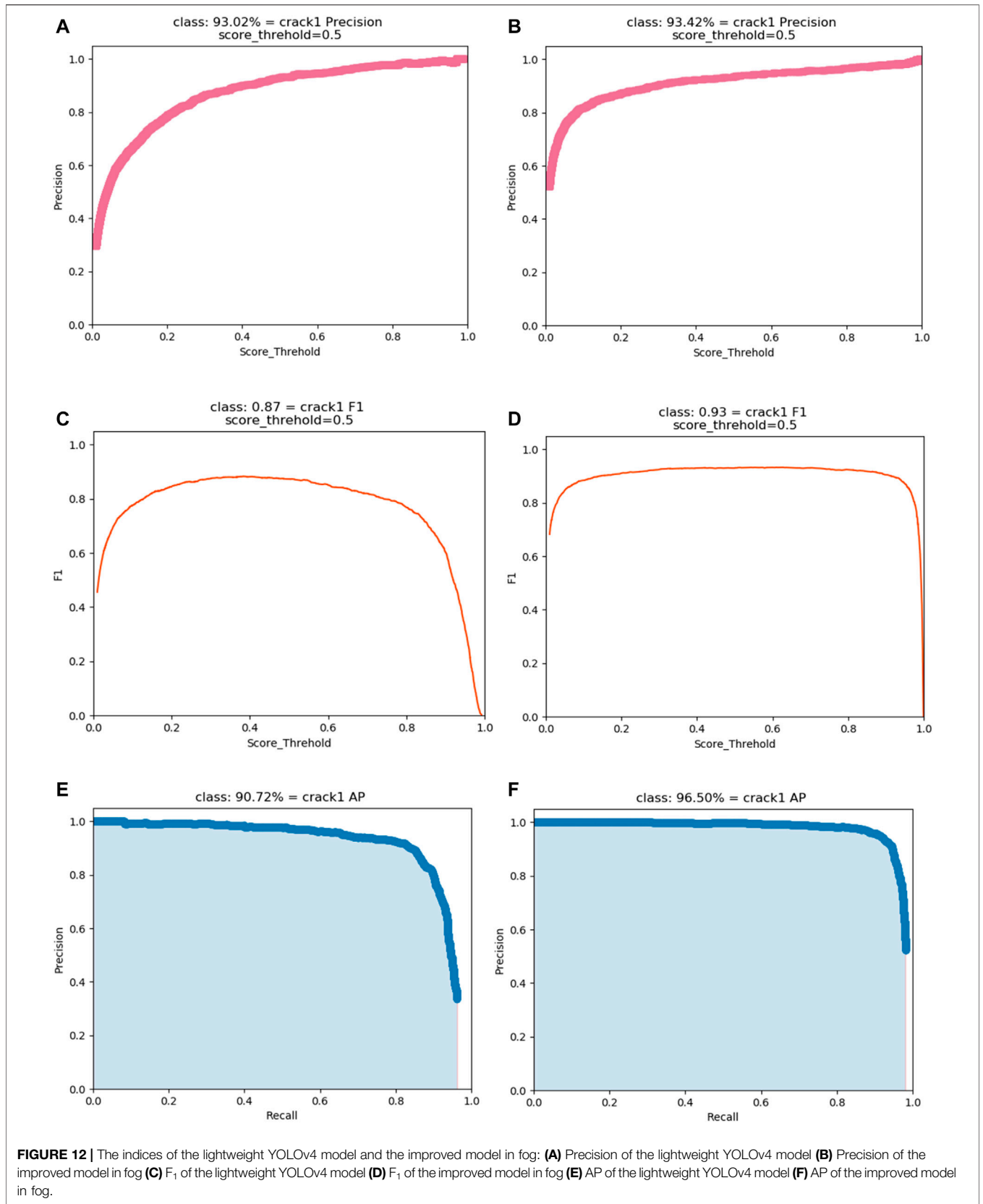
FIGURE 12 | The indices of the lightweight YOLOv4 model and the improved model in fog: (A) Precision of the lightweight YOLOv4 model (B) Precision of the improved model in fog (C) F$_1$ of the lightweight YOLOv4 model (D) F$_1$ of the improved model in fog (E) AP of the lightweight YOLOv4 model (F) AP of the improved model in fog.

TABLE 5 | The performance comparison between the lightweight YOLOv4 model and the improved model in fog.

Algorithm	mAP (%)	Parameters (M)	FLOPs (GMacs)
The lightweight YOLOv4 model	90.72	121	1.95
The improved model in fog	96.50	132	2.24

TABLE 6 | The input and output of the lightweight YOLOv4 model and the improved model in fog in concrete surface crack detection.

Original image	Prediction by the lightweight YOLOv4 model	Prediction by the improved model in fog
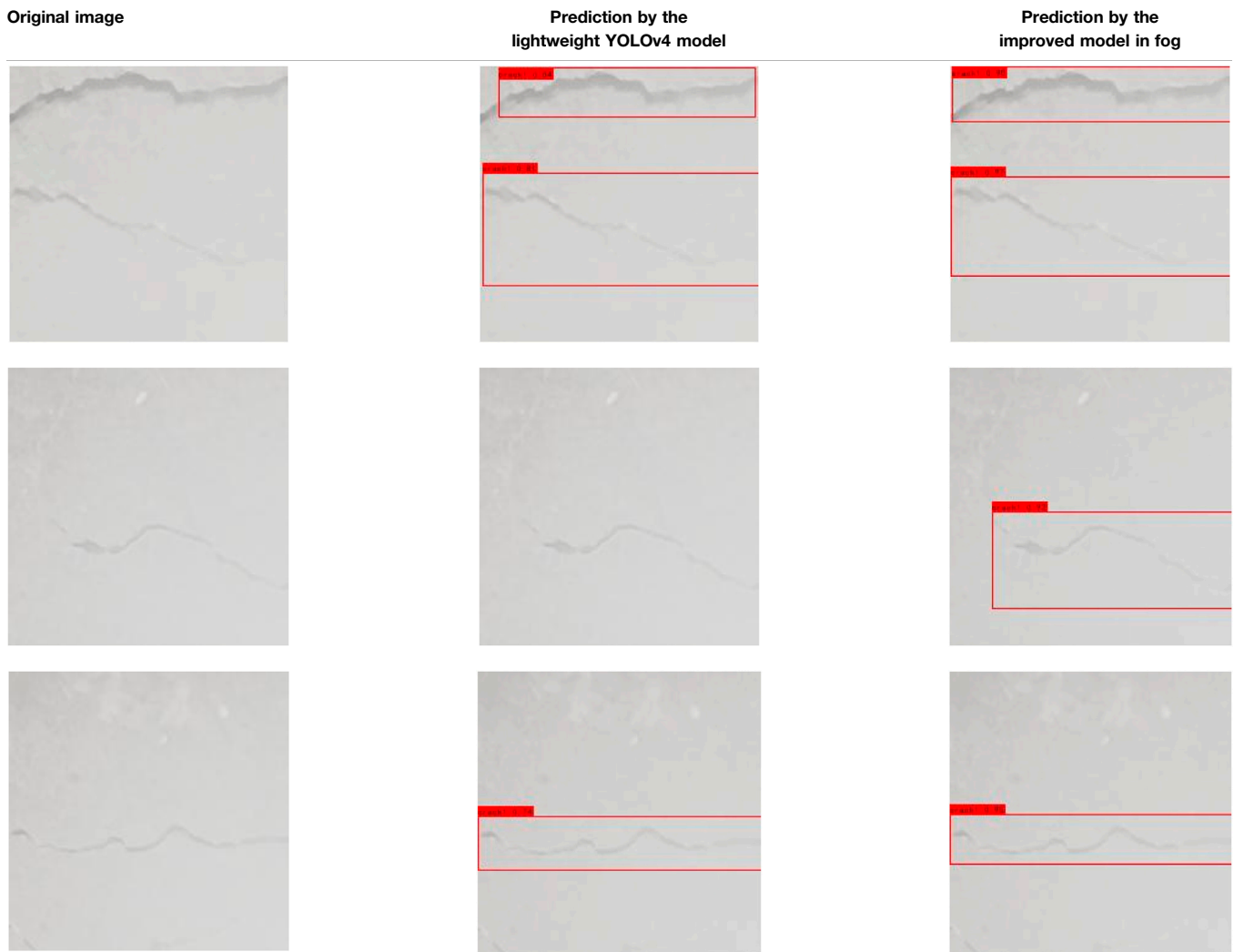
short, this is for calculating the mean value of precision on the P-R curve. **Eqs. 17**, **18** are the formulas for AP and mAP, respectively.
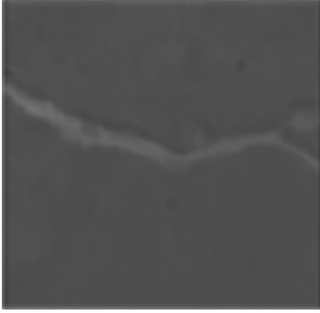
$$F_1 = \frac{2 \times P \times R}{P + R} \tag{16}$$

$$AP = \int_0^1 p(r)dr \tag{17}$$

$$mAP = \frac{1}{|Q_R|} \sum_{q=1}^{Q_R} AP(q) \tag{18}$$

In this paper, in addition to the mAP, the model size and computational complexity FLOPs are used to evaluate the model compression algorithm. The model's size is closely related to its parameters, which can be used to measure the simplification of the YOLOv4 model. FLOPs reflects the calculation amount of the algorithm. The unit of FLOPs is GMacs, which is short for Giga

TABLE 7 | The haze removal results of the concrete surface cracks.

Original image	Transmittance image	Restoration image

Multiply-Accumulation operations per second. It represents the floating-point operations per second, which can reflect the algorithm's calculation performance.

RESULTS AND DISCUSSIONS

The Lightweight YOLOv4 Model

To verify the concrete surface crack detection performance of the lightweight YOLOv4 model proposed in this paper, the experimental results are compared with those of the original YOLOv4 model. The detection rate (precision), comprehensive evaluation value F_1 and AP are used for evaluation, as shown in **Figure 10**.

Figure 10 shows that the detection performance of the lightweight YOLOv4 model is basically consistent with that of the original YOLOv4 model, and the curves of the precision, F_1 and AP are similar. It can be concluded that there is basically no loss in performance after the model is simplified. The model size and FLOPs are used to verify the simplification effect of the model. **Table 4** shows the performance comparison of the lightweight YOLOv4 model with the original YOLOv4 model and the YOLOv3 model. **Table 4** shows that in the lightweight YOLOv4 model, the weight is reduced by 50% and the FLOPs is reduced by 93.5%. Compared with the YOLOv3 model, the performance of the lightweight YOLOv4 model has certain advantages in terms of the mAP, the weight and the FLOPs. To demonstrate the detection performance of the proposed model more intuitively, some images shown in **Figure 11** were randomly selected from the database for testing.

The Crack Detection Model in Fog

To verify the performance of the improved YOLOv4 model when detecting concrete surface cracks in fog, the lightweight YOLOv4 model and the improved model in fog were trained with the same experimental conditions, the same foggy database and the same optimization algorithm. As shown in **Figure 12**, the precision, F_1 and AP are used for evaluation.

Figure 12 illustrates that the lightweight YOLOv4 model does not perform well in fog, while the improved model performs better. It can be seen that the AP has a higher improvement, which proves that the performance of the improved network model in detecting concrete surface cracks in fog exhibits a certain improvement. The model size and FLOPs are used to verify the performance of the improved model. **Table 5** shows the performance comparison between the lightweight YOLOv4 model and the improved model in fog.

Table 5 shows that the improved model in fog is slightly higher than the lightweight YOLOv4 model in terms of weight and FLOPs, but has a greater benefit in the mAP than the lightweight model. In order to more intuitively show the performance of the improved model in detecting concrete surface cracks in fog, some images were randomly selected from the database for testing, as shown in **Table 6**.

It can be seen intuitively from **Table 6** that the detection result of the lightweight YOLOv4 model in the first image is not accurate, and some of the crack features are not recognized. The improved model in fog has a better recognition effect. In the second image, due to the heavy fog, the thin crack structure and the inconspicuous features, the lightweight YOLOv4 model fails to make a correct identification, while the improved network model is accurate. Both algorithms can correctly detect cracks in the third image. In order to intuitively show

the effect of crack scene restoration in the middle layer of the improved model, several test images are selected to illustrate the effect of haze removal, as shown in **Table 7**. It can be seen that the results of restoring the crack structure using this paper's improved model are clearer and more obvious than the original fog image, which is more conducive to feature extraction by the subsequent lightweight YOLOv4 model.

CONCLUSION

A crack detection method based on the YOLOv4 algorithm is proposed, which provides a more accurate, efficient and intelligent method for the detection of cracks on concrete surfaces. A smartphone is used for collecting 2000 raw images (3,024 × 3,024 pixels) from the surfaces of concrete buildings. To reduce the computation of the training process, the collected images are cropped to 256 × 256 pixels. 7,000, 1,000 and 2,000 images are used for training, validation and testing, respectively. The YOLOv4 architecture described in detail in **Section 2.3** was simplified for crack detection. The lightweight YOLOv4 model achieved an mAP of 96.88% with 121 M and 1.95 GMacs. The results showed that the proposed method can provide good crack detection results with a lower trained model weight. In this paper, images taken on sunny days are manually fogged to build the database. The lightweight YOLOv4 model was modified to have better performance for crack detection in fog, which is described in detail in **Section 3.3**. The improved model for crack detection in fog achieved an mAP of 96.50% with 132 M and 2.24 GMacs. The results showed that the improved method can provide better crack detection results with only a slightly higher trained model weight. The detection performance of the proposed model has been improved in both subjective vision and objective evaluation metrics, and is more effective at detecting concrete surface cracks in fog.

Though the proposed method in this paper exhibits good performance, there is still a long way to go for engineering applications. In the experiment, there are several directions we found that may be tried and improved. Firstly, to reduce the computation of the training process, the model only uses small pixel images (256 × 256 pixels). If a large number of images must be processed, the images shall be cropped or scaled. It is absolutely not a fundamental solution to the problem. It is worth exploring approaches to improve the algorithm to adapt it to larger image inputs. Secondly, many hyper-parameters need to be artificially adjusted when applying the method. We must still carry out plenty of experiments to thoroughly explore the impact of these hyper-parameters on the performance of the model. Finally, more types of defect images across more complex backgrounds should be collected to enlarge the database, thus improving the accuracy and robustness of the proposed method.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

FUNDING

REFERENCES

Beck, A., and Teboulle, M. (2009). Fast Gradient-Based Algorithms for Constrained Total Variation Image Denoising and Deblurring Problems. *IEEE Trans. Image Process.* 18, 2419–2434. doi:10.1109/TIP.2009.2028250

Bochkovskiy, A., Wang, C. Y., and Liao, H. (2020). "YOLOv4: Optimal Speed and Accuracy of Object Detection," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, June 14-19 (Seattle, WA: arXiv:2004.10934v1).

Cha, Y.-J., Choi, W., Suh, G., Mahmoudkhani, S., and Büyüköztürk, O. (2018). Autonomous Structural Visual Inspection Using Region-Based Deep Learning for Detecting Multiple Damage Types. *Computer-Aided Civil Infrastructure Eng.* 33, 731–747. doi:10.1111/mice.12334

Chen, F.-C., and Jahanshahi, M. R. (2018). NB-CNN: Deep Learning-Based Crack Detection Using Convolutional Neural Network and Naïve Bayes Data Fusion. *IEEE Trans. Ind. Electron.* 65, 4392–4400. doi:10.1109/tie.2017.2764844

Chen, M., Tang, Y., Zou, X., Huang, K., Li, L., and He, Y. (2019). High-accuracy Multi-Camera Reconstruction Enhanced by Adaptive point Cloud Correction Algorithm. *Opt. Lasers Eng.* 122, 170–183. doi:10.1016/j.optlaseng.2019.06.011

Choi, J.-I., Lee, Y., Kim, Y. Y., and Lee, B. Y. (2017). Image-processing Technique to Detect Carbonation Regions of concrete Sprayed with a Phenolphthalein Solution. *Construction Building Mater.* 154, 451–461. doi:10.1016/j.conbuildmat.2017.07.205

Cireşan, D., Meier, U., Masci, J., and Schmidhuber, J. (2012). Multi-column Deep Neural Network for Traffic Sign Classification. *Neural Networks* 32, 333–338. doi:10.1016/j.neunet.2012.02.023

Dung, C. V., and Anh, L. D. (2019). Autonomous concrete Crack Detection Using Deep Fully Convolutional Neural Network. *Automation in Construction* 99, 52–58. doi:10.1016/j.autcon.2018.11.028

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE Comp. Soc.* doi:10.1109/CVPR.2014.81

Girshick, R. (2015). "Fast R-CNN," in IEEE International Conference on Computer Vision (ICCV), Santiago, CHILE, December 11-18 (IEEE), 1440–1448. doi:10.1109/ICCV.2015.169

Guan, J., Yang, X., Ding, L., Cheng, X., Lee, V. C. S., and Jin, C. (2021). Automated Pixel-Level Pavement Distress Detection Based on Stereo Vision and Deep Learning. *Automation in Construction* 129, 103788. doi:10.1016/j.autcon.2021.103788

Han, K., Wang, Y., Tian, Q., Guo, J., and Xu, C. (2020). "GhostNet: More Features from Cheap Operations," in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, June 14-19 (Seattle, WA: arXiv:1911.11907), 1577–1586. doi:10.1109/cvpr42600.2020.00165

He, K., Sun, J., and Tang, X. (2011). Single Image Haze Removal Using Dark Channel Prior. *IEEE Trans. Pattern Anal. Mach. Intell.* 33 (12), 2341–2353. doi:10.1109/TPAMI.2010.168

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 1904–1916. doi:10.1109/TPAMI.2015.2389824

Ji, A., Xue, X., Wang, Y., Luo, X., and Xue, W. (2020). An Integrated Approach to Automatic Pixel-Level Crack Detection and Quantification of Asphalt Pavement. *Automation in Construction* 114, 103176. doi:10.1016/j.autcon.2020.103176

Jiang, Y., Pang, D., and Li, C. (2021). A Deep Learning Approach for Fast Detection and Classification of concrete Damage. *Automation in Construction* 128, 103785. doi:10.1016/j.autcon.2021.103785

Koziarski, M., and Cyganek, B. (2017). Image Recognition with Deep Neural Networks in Presence of Noise - Dealing with and Taking Advantage of Distortions. *Ica* 24, 337–349. doi:10.3233/ica-170551

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* 60, 84–90. doi:10.1145/3065386

Laofor, C., and Peansupap, V. (2012). Defect Detection and Quantification System to Support Subjective Visual Quality Inspection via a Digital Image Processing: A Tiling Work Case Study. *Automation in Construction* 24, 160–174. doi:10.1016/j.autcon.2012.02.012

Lecun, Y., Bengio, Y., and Hinton, G. (2015). Deep Learning. *Nature* 521, 436–444. doi:10.1038/nature14539

Li, S., Zhao, X., and Zhou, G. (2019). Automatic Pixel-Level Multiple Damage Detection of concrete Structure Using Fully Convolutional Network. *Computer-Aided Civil Infrastructure Eng.* 34, 616–634. doi:10.1111/mice.12433

Lin, Y.-Z., Nie, Z.-H., and Ma, H.-W. (2017). Structural Damage Detection with Automatic Feature-Extraction through Deep Learning. *Computer-Aided Civil Infrastructure Eng.* 32, 1025–1046. doi:10.1111/mice.12313

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "SSD: Single Shot MultiBox Detector," in 2016 14th European Conference on Computer Vision, Amsterdam, The Netherlands, October 8-10, 21–37. doi:10.1007/978-3-319-46448-0_2

Mei, Q., and Gül, M. (2020). A Cost Effective Solution for Pavement Crack Inspection Using Cameras and Deep Neural Networks. *Construction Building Mater.* 256, 119397. doi:10.1016/j.conbuildmat.2020.119397

Nayar, S. K., and Narasimhan, S. G. (1999). "Vision in Bad Weather," in Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece, September 20-27 (IEEE), 820–827. doi:10.1109/ICCV.1999.790306

Neubeck, A., and Van Gool, L. (2006). "Efficient Non-maximum Suppression," in International Conference on Pattern Recognition, Hong Kong, China, August 20-24, 850–855. doi:10.1109/icpr.2006.479

Nishikawa, T., Yoshida, J., Sugiyama, T., and Fujino, Y. (2012). Concrete Crack Detection by Multiple Sequential Image Filtering. *Computer-Aided Civil Infrastructure Eng.* 27, 29–47. doi:10.1111/j.1467-8667.2011.00716.x

Oh, J.-K., Jang, G., Oh, S., Lee, J. H., Yi, B.-J., Moon, Y. S., et al. (2009). Bridge Inspection Robot System with Machine Vision. *Automation in Construction* 18, 929–941. doi:10.1016/j.autcon.2009.04.003

Peng, C., Yang, M., Zheng, Q., Zhang, J., Wang, D., Yan, R., et al. (2020). A Triple-Thresholds Pavement Crack Detection Method Leveraging Random Structured forest. *Construction Building Mater.* 263, 120080. doi:10.1016/j.conbuildmat.2020.120080

Redmon, J., and Ali, F. (2018). "YOLOv3: An Incremental Improvement," in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, June 19 - 21 (Salt Lake City, UT: arXiv:1804.02767).

Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans.*

Pattern Anal. Mach. Intell. 39, 1137–1149. doi:10.1109/TPAMI.2016.2577031

Salman, M., Mathavan, S., Kamal, K., and Rahman, M. (2013). "Pavement Crack Detection Using the Gabor Filter," in 2013 16th International Ieee Conference on Intelligent Transportation Systems, The Hague, NETHERLANDS, 6-9 Oct. 2013 (IEEE), 2039–2044. doi:10.1109/ITSC.2013.6728529

Sun, Y., Yang, Y., Yao, G., Wei, F., and Wong, M. (2021). Autonomous Crack and Bughole Detection for Concrete Surface Image Based on Deep Learning. *IEEE Access* 9, 85709–85720. doi:10.1109/access.2021.3088292

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). "Rethinking the Inception Architecture for Computer Vision," in 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, June 27-30 (IEEE), 2818–2826. doi:10.1109/CVPR.2016.308

Tang, Y., Feng, W., Chen, Z., Nong, Y., Guan, S., and Sun, J. (2021b). Fracture Behavior of a Sustainable Material: Recycled concrete with Waste Crumb Rubber Subjected to Elevated Temperatures. *J. Clean. Prod.* 318, 128553. doi:10.1016/j.jclepro.2021.128553

Wei, F., Yao, G., Yang, Y., and Sun, Y. (2019). Instance-level Recognition and Quantification for concrete Surface Bughole Based on Deep Learning. *Automation in Construction* 107, 102920. doi:10.1016/j.autcon.2019.102920

Wei, W., Ding, L., Luo, H., Li, C., and Li, G. (2021). Automated Bughole Detection and Quality Performance Assessment of concrete Using Image Processing and Deep Convolutional Neural Networks. *Construction Building Mater.* 281, 122576. doi:10.1016/j.conbuildmat.2021.122576

Yao, G., Sun, Y., Wong, M., and Lv, X. (2021). A Real-Time Detection Method for Concrete Surface Cracks Based on Improved YOLOv4. *Symmetry* 13, 1716. doi:10.3390/sym13091716

Yao, G., Wei, F., Yang, Y., and Sun, Y. (2019). Deep-Learning-Based Bughole Detection for Concrete Surface Image. *Adv. Civil Eng.* 2019, 1–12. doi:10.1155/2019/8582963

Yeum, C. M., and Dyke, S. J. (2015). Vision-Based Automated Crack Detection for Bridge Inspection. *Computer-Aided Civil Infrastructure Eng.* 30, 759–770. doi:10.1111/mice.12141

Yunchao, T., Zheng, C., Wanhui, F., Yumei, N., Cong, L., and Jieming, C. (2021a). Combined Effects of Nano-Silica and Silica Fume on the Mechanical Behavior of Recycled Aggregate concrete. *Nanotechnology Rev.* 10, 819–838. doi:10.1515/ntrev-2021-0058

Zalama, E., Gómez-García-Bermejo, J., Medina, R., and Llamas, J. (2014). Road Crack Detection Using Visual Features Extracted by Gabor Filters. *Computer-Aided Civil Infrastructure Eng.* 29, 342–358. doi:10.1111/mice.12042

Zhang, W., Li, H., Li, Y., Liu, H., Chen, Y., and Ding, X. (2021). Application of Deep Learning Algorithms in Geotechnical Engineering: a Short Critical Review. *Artif. Intell. Rev.* 54, 5633–5673. doi:10.1007/s10462-021-09967-1

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.