mathematics

MDPI

Article

Improving YOLOv4-Tiny's Construction Machinery and Material Identification Method by Incorporating Attention Mechanism

Jiale Yao [1], Dengsheng Cai [2,3], Xiangsuo Fan [1,4,*] and Bing Li [5]

[1] School of Electrical Electronics and Computer Science, Guangxi University of Science and Technology, Liuzhou 545006, China; 221055224@stdmail.gxust.edu.cn
[2] School of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China; caids@liugong.com
[3] Intelligent Technology Research, Institute of Global Research and Development Center, Guangxi LiuGong Machinery Company Limited, Liuzhou 545007, China
[4] School of Resources and Environment, University of Electronic Science and Technology of China, Chengdu 611731, China
[5] Guangxi Collaborative Innovation Centre for Earthmoving Machinery, Guangxi University of Science and Technology, Liuzhou 545006, China; 100001548@gxust.edu.cn
* Correspondence: 100002085@gxust.edu.cn

Abstract: To facilitate the development of intelligent unmanned loaders and improve the recognition accuracy of loaders in complex scenes, we propose a construction machinery and material target detection algorithm incorporating an attention mechanism (AM) to improve YOLOv4-Tiny. First, to ensure the robustness of the proposed algorithm, we adopt style migration and sliding window segmentation to increase the underlying dataset's diversity. Second, to address the problem that YOLOv4-Tiny's (the base network) framework only adopts a layer-by-layer connection form, which demonstrates an insufficient feature extraction ability, we adopt a multilayer cascaded residual module to deeply connect low- and high-level information. Finally, to filter redundant feature information and make the proposed algorithm focus more on important feature information, a channel AM is added to the base network to perform a secondary screening of feature information in the region of interest, which effectively improves the detection accuracy. In addition, to achieve small-scale object detection, a multiscale feature pyramid network structure is employed in the prediction module of the proposed algorithm to output two prediction networks with different scale sizes. The experimental results show that, compared with the traditional network structure, the proposed algorithm fully incorporates the advantages of residual networks and AM, which effectively improves its feature extraction ability and recognition accuracy of targets at different scales. The final proposed algorithm exhibits the features of high recognition accuracy and fast recognition speed, with mean average precision and detection speed reaching 96.82% and 134.4 fps, respectively.

Keywords: intelligent loader; style transfer; machine vision; CAM; YOLOv4-Tiny

MSC: 68T10

Citation: Yao, J.; Cai, D.; Fan, X.; Li, B. Improving YOLOv4-Tiny's Construction Machinery and Material Identification Method by Incorporating Attention Mechanism. *Mathematics* **2022**, *10*, 1453. https://doi.org/10.3390/math10091453

Academic Editor: Stelios Papadakis

Received: 16 March 2022
Accepted: 21 April 2022
Published: 26 April 2022

1. Introduction

Recently, with the rapid development of computer technology, fifth-generation technology, communication technology, artificial intelligence, and other high-tech, various industries' intelligence has been improved to different degrees. Among them, the intelligent development of construction machinery (CM) has attracted considerable attention [1]. The development strategy of "Made in China 2025" indicated the direction for the future development of various industries at present as well as the pathway for the intelligent development of CM. CM covers national defense construction, transportation construction, energy industry construction, mining, and other important national fields, and its

intelligent development has become one of the hot research directions. As the pioneer of modern construction, CM is mainly used in the construction sites of large-scale projects. However, the working conditions are often accompanied by vibration, high temperatures, dust, volatile gas, bad smells, radiation, and other harsh environments; these harsh environments pose severe threats to the personal safety and health of machine operators. Furthermore, longterm exposure to these environments is extremely harmful to workers. Therefore, the development of intelligent and unmanned engineering machinery and equipment is crucial. In this study, we address the current application needs of harsh working conditions and improve the ability of a loader to independently determine the category of CM and materials by exploring material intelligence recognition technology under complex scenes to realize the intelligent level of the loader. As such, the loader can sense the surrounding environment and make its own decisions as well as operate under the condition of protecting the personal safety of workers, thereby minimizing risk while improving work efficiency.

At present, mainly two types of target detection methods exist: traditional manual feature extraction (FE)- and convolutional neural network (CNN)-based visual detection methods. The traditional manual FE-based detection method has been mainly used for classifying and recognizing images in recent years. For example, Haralick et al. [2] proposed a classification technique based on texture features using a grayscale co-occurrence matrix to describe image textures and hence classify them. Lazebnik et al. [3] proposed a classification technique based on spatial relationship features using a spatial pyramid model to enhance the feature description capability and achieve target classification. Chai et al. [4] used a sliding window detector to obtain local features and then fed them into a support vector machine classifier for classification; this algorithm obtained 67% classification accuracy.

In the last decade, the target detection field developed with the development of deep learning techniques. In particular, target detection techniques within the deep learning field developed rapidly after the advent of AlexNet [5]. AlexNet is a network proposed by Hinton and his student Alex Krizhevsky in 2012, which won the ILSVRC competition for classification. In 2014, Simonyan et al. [6] proposed a deeper network model, VGGNet. Szegedy et al. [7] also proposed GoogleNet in 2014 and won the championship of ILSVRC-2014. K. He [8] proposed ResNet in 2015, which uses a residual module to achieve constant mapping to solve the degradation problem of networks as the number of layers increases. Girshick et al. [9] proposed the region-based CNN (RCNN) model to apply CNN to the target detection field; afterward, the Fast RCNN [10] and Faster RCNN [11] models were developed to improve the detection accuracy of targets. The emergence of you only look once (YOLO) [12] realized end-to-end real-time detection, and the detection speed was significantly improved; however, accuracy was sacrificed.

Currently, applying improved YOLO versions to specific topics has become a mainstream idea. Wang et al. [13] applied YOLO version 4 (YOLOv4) to the multitarget detection of aerial images to improve the generalizability of YOLO for problems of many small and easily obscured targets in unmanned aerial vehicle (UAV) images, complex detection scenes, and low detection accuracy due to large-scale variability. Yu et al. [14] employed the YOLOv4-Tiny algorithm for target detection of pigs; compared with YOLOv4, the algorithm compressed the model size by 80% and improved the detection speed by 11 frames/s without sacrificing accuracy. Huang et al. [15] applied the YOLOv4 algorithm to remote sensing target detection, thereby improving the detection accuracy for remote sensing image targets. Li et al. [16] employed an improved YOLOv4 algorithm for surface defect detection; compared with the original YOLOv4 algorithm, the improved YOLOv4 algorithm increased the mean average precision (mAP) and detection speed on a test set by 2.17% and 1.58 fps, respectively, solving the problems of low accuracy and slow detection speed of surface defects of aero-engine components to a certain extent. Guo et al. [17] applied the improved YOLOv4 algorithm to detect mixed pedestrian–vehicle traffic at complex intersections; compared with YOLOv4, the detection accuracy of pedestrians and vehicles with severe occlusion and overlap improved in complex intersection scenarios.

Wang et al. [18] applied the improved YOLOv4-Tiny algorithm to recognize and detect blueberry ripeness; the algorithm could achieve an average accuracy of 96.24% in complex scenes, such as occlusion and uneven illumination. Jing et al. [19] applied the improved YOLOv4-Tiny algorithm to UAV photography target detection using a bottom-up fusion of deep and shallow semantic information to enrich the feature information about small targets; the average accuracy rate improved by 5.09% compared with the original YOLOv4-Tiny algorithm, with better comprehensive performance. Andriyanov et al. designed an apple position estimation system based on YOLOv3 and depth cameras [20], which can calculate the relative distance of all coordinates based on the distance of the target and the positioning of the target in the image. Kuznetsova et al. used the YOLOv3 algorithm to develop a machine vision system for orchard apple detection [21], which can be used not only on an apple harvesting robot, but also on an orange harvesting robot. The approach from multilayer artificial neural networks to pattern recognition and CNN designed by Kamyshova et al. is inspiring [22]. Osipov et al. used a deep learning approach to solve the problem of recognition and classification under severe weather conditions [23], which has significant advantages over the classical convolutional neural network approach. Yan et al. applied the improved YOLOv5 algorithm to apple detection [24], which met the requirements of realtime apple detection. Zhao et al. applied the improved YOLOv5 algorithm to wheat spike detection [25] with a mAP of 94.1%, which is 10.8% better than the standard YOLOv5. In recent years, an image annotation algorithm using convolutional features from the middle layer of deep learning proposed by Chen et al. [26] and the idea of processing images based on the attention mechanism (AM) [27] motivated this study; we believe incorporating the AM into CNN can be helpful for accuracy improvement. Li et al. incorporated the AM into the YOLOv3 algorithm to solve the imbalance in the distribution of detection frames in the edge region problem [28]. Mo et al. improved an image restoration algorithm using multiscale adversarial networks and neighborhood models to improve the accuracy of image coloring and to apply to many types of images [29].

To promote the rapid development of intelligent unmanned loaders, studying the intelligent recognition algorithms of loaders is essential. Inspired by the above studies, we propose a target detection algorithm for problems faced by loaders in daily operations, such as variable environments, complex working conditions, and large jitter amplitude. The algorithm enables the loader to detect and classify materials and CM accurately in real time in both normal and complex environments and demonstrates high real-time performance while ensuring high recognition accuracy.

The main contributions of this study are as follows.

(1) For the complex and variable environments faced by loader operations, we employ the technical means of style migration and sliding window segmentation to expand the dataset, which significantly improves the recognition rate of the loader in a low-contrast environment and vehicle jittering situation;

(2) We use VGG19, a network with strong FE ability, and a YOLOv4-Tiny streamlined network to realize a high recognition rate for loaders under complex working conditions, making the proposed algorithm exhibit a high recognition rate under various working conditions;

(3) In this study, an improved FE network is integrated with an AM to effectively balance the relationship between detection speed and accuracy. Compared with the previous network, the recognition accuracy increased to a certain extent while ensuring almost constant detection speed and training time;

(4) The final target detection algorithm is significantly improved compared with the traditional algorithm, and the final proposed network demonstrates the characteristics of high recognition accuracy and fast recognition speed, with mAP and detection speed reaching 96.82% and 134.4 fps, respectively.

2. Dataset

CM operating in the construction site scene, along with vehicle movement, environmental changes, and other circumstances, will cause image blurring problems. Similarly,

extreme environments and weather will also affect images to a certain extent. Therefore, we need to enhance the adaptability of the detection models to complex working conditions and reduce the impact of external factors on recognition accuracy by enhancing the underlying dataset.

According to our research objectives, the proposed target detection algorithm needs to demonstrate a high recognition rate and fast recognition speed in low-contrast environments; also, the algorithm should not be computationally intensive. During the image acquisition, we captured images under different angles, distances, clarities, weathers, times, etc. To increase the ability to realize a high recognition rate in extreme environments, we use the technical means of style migration to migrate the feature information about extreme weather, such as fog, rain, and snow, and enhance the image by replacing the original image background. In addition, we use the technique of sliding window to convert a single image into multiple images, including part of the target image, to expand the sample size and enhance the detail features of the target simultaneously. At this point, we acquired a total of 14,349 images.

2.1. Image Acquisition

In this study, we used a Canon SX280 HS camera to capture images of seven types of targets, including loaders, excavators, trucks, stones, rocks, loess, and fine sand, in the early morning, midday, and evening under rainy, clear, and high sunlight weather conditions, respectively. The images captured under normal weather are shown in Figure 1. The images were collected during the shooting process for targets with different rotation angles, different distances, different clarities, and different exposure values. These images were captured at the R&D experiment site of Guangxi LiuGong Machinery Co. The images were taken over a large time span in order to acquire images under different weather conditions [30].

Figure 1. Target images under normal environment.

When capturing images, the exposure value of the camera and the shooting angle were artificially adjusted to acquire low-contrast images. This image enhancement method not only expands the variety of images, but also increases the adaptability and robustness of the proposed target detection algorithm in complex environments, which makes the model's algorithm exhibit a high recognition rate in low-contrast environments and effectively solves the interference problem faced by loaders in actual operations [31,32]. The schematic diagram is shown in Figure 2.

Figure 2. Exposure value transformation image enhancement schematic.

2.2. Data Augmentation

To ensure the proposed algorithm is robust and adaptive, we use image enhancement to augment the dataset. In addition, an overfitting problem caused by insufficient samples is avoided to some extent due to an increase in the sample size of the dataset. We mainly use style migration and sliding window segmentation for the image enhancement process.

2.2.1. Style Transformation

Style migration is a method to make the original image acquire the style of other images. The specific process is to make one image the style image and the other image the content image. After preprocessing the two images, a Gaussian white noise image is generated, and then, after VGG19 or VGG16 is employed, the deep features of the two images are extracted, and the feature maps of each convolution layer are obtained. The covariance matrix of the feature maps obtained after the images have been convolved can well characterize the texture features of the images [33].

Because VGG19 exhibits a stronger FE capability than VGG16, we choose VGG19 as our FE network. To achieve the best results for the final style-migrated image, a loss function is defined to specify the desired goal and minimize loss (see Equations (1)–(4)). The content loss not only captures the global and abstract image content in the convolution layer but also calculates the L2 parity between the target and generated images at the same layer's activation function, which ensures that the generated image looks similar to the original target image. Style loss refers to maintaining the internal correlation between different layers between the stylized reference image and the generated stylized feature image, which effectively ensures that the texture styles of the reference image and the generated image do not differ significantly on different scale spaces. The loss functions of the content and style images are assigned with different weights, which constitute the final style migration image loss, and a gradient descent process is set to minimize this loss function to ensure the accuracy of VGG19 in extracting features from the content and style images. Finally, the image texture is captured using feature correlation, and the texture features of the content and style images are fused to form the final migrated image. The flowchart of style migration implementation is shown in Figure 3.

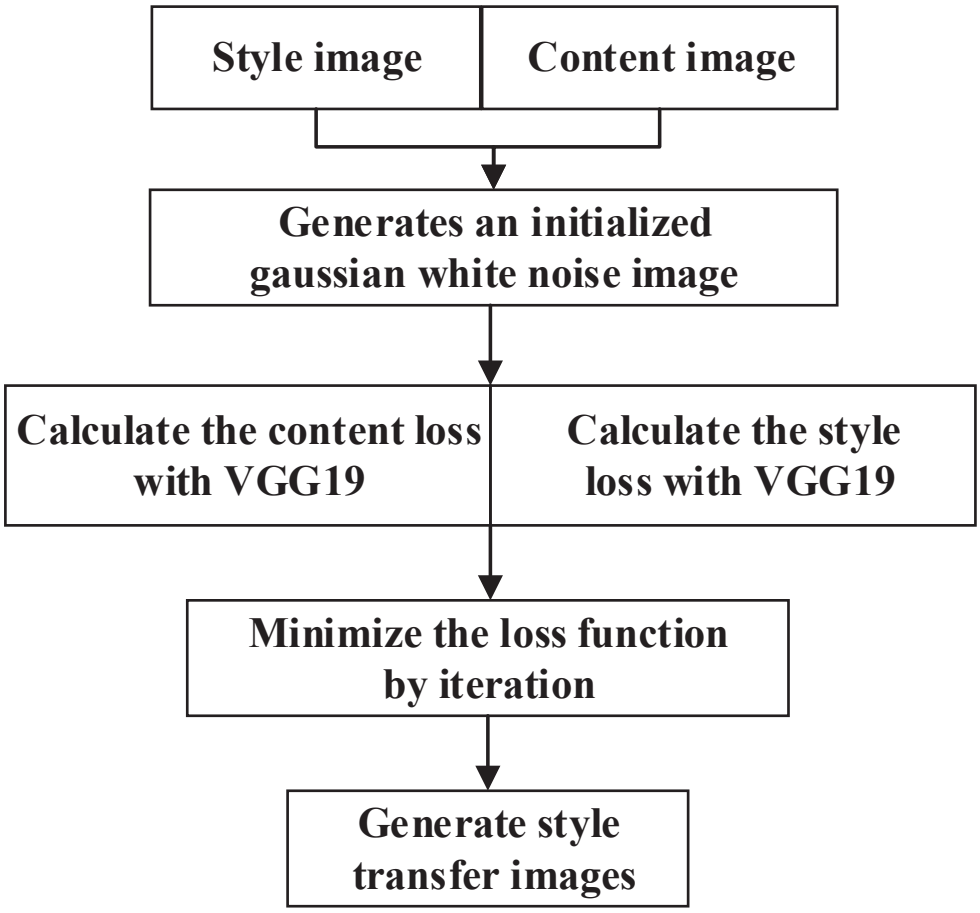

Figure 3. Image style migration implementation flowchart.

The loss calculation formula for style migration is as follows:
Content loss:

$$L_{content}(\vec{p}, \vec{a}, r) = \frac{1}{2} \sum_{i,j} \left(F_{i,j}^{r} - A_{i,j}^{r} \right)^2 \tag{1}$$

Style loss:

$$G_{i,j}^{r} = \sum_{k=1}^{M_r} N_{ik}^{r} N_{jk}^{r} \tag{2}$$

$$L_{style} = \sum_{r} \omega_r \left(\frac{1}{4N_r^2 M_r^2} \sum_{i=1}^{N_r} \sum_{j=1}^{N_r} \left(G_{ij}^{r} - A_{ij}^{r} \right)^2 \right) \tag{3}$$

Generate style migration images:

$$L_{total}(\vec{p}, \vec{a}, \vec{x},) = \alpha L_{content}(\vec{p}, \vec{a}) + \beta L_{style}(\vec{a}, \vec{x}) \tag{4}$$

where $L_{total}(\vec{p}, \vec{a}, \vec{x},)$ denotes the content loss at layer r; $\vec{p}$ and $\vec{a}$ denote the content and generated images, respectively; $F_{i,j}^{r}$ and $A_{i,j}^{r}$ denote the content and background images, respectively, at layer r; N_{jk}^{r} and N_{ik}^{r} denote the k-th element of the i-th and j-th channels, respectively, at layer r of the feature map; ω_r is the weighting factor of each layer's contribution to the total loss; M_r denotes the product of the length and width of the feature map (i.e., the feature size); and G_{ij}^{r} is the Gram matrix of the style migration image.

In this study, the expansion of the dataset's sample size and that of the types of complex scene images are achieved using an image style migration method. The content

map is mainly the seven types of target images captured above, and the style map uses three types of backgrounds: rainy, foggy, and snowy days. By this method, the problem of not easily obtaining meteorological conditions, such as snowy and foggy days, in southern China is solved, and the dataset's diversity is increased. The results are presented with an excavator as the representative of CM and loess as the representative of material (Figure 4).

Figure 4. Schematic diagram of the style migration image of excavator and loess.

2.2.2. Sliding Window Segmentation

Sliding window processing is a technique to capture a complete image at a certain size and cut the image at a fixed step from the beginning to the end of the original image. The sliding window processing allows us to obtain a partial feature map of the original image, and we select the generated image with distinct features in the generated image to expand the dataset. The dataset obtained as such can enhance the adaptability of the training model so that the model can obtain more information, demonstrate higher detection accuracy, and better adapt to the real CM operating environment. The calculation formula is shown in Equation (5), and the graph of the sliding window processing results is shown in Figure 5.

$$\begin{cases} [m, n, d] = size(f) \\ h = round(m/2) \\ w = round(n/2) \\ i = h : step : m - h \\ j = w : step : n - w \\ f_1 = f((i - (h-1)) : (i + h), (j - (w-1)) : (i + w), :) \end{cases} \tag{5}$$

where f, m, n and d represent the original image, its height, width, and bit depth, respectively; h and w represent the height and width of the sliding window; $step$ represents the step size; i and j represent the rows and columns of the original image, respectively; $round$ represents rounding; and f_1 represents the generated image.

2.3. Image Annotation and Dataset Production

We use professional labeling software, LabelImg, to create image labels and produce the dataset. To ensure a more accurate recognition rate, we balanced the number of images in each category more. Each category includes both captured original images and image enhanced images. The breakdown of the dataset categories and quantities is shown in Table 1.

Table 1. Breakdown of dataset categories and quantities.

Classes	Excavator	Loader	Truck	Loess	Stone	Gravel	Sand	Total
Samples	2480	1797	2002	2113	2000	2000	1957	14,349

Figure 5. Sliding window segmentation processing results graph; the first column is the original image, and columns 2–4 are the generated images.

3. Method

3.1. YOLO (You Only Look Once)

YOLO is a one-stage detection and recognition integrated algorithm, which can obtain the location and category of the target directly from the image and achieve two major tasks of classification and localization simultaneously, with high real-time performance. Several versions of YOLO exist, including YOLOv1, YOLOv2 [34], YOLOv3 [35], and YOLOv4 [36]. YOLOv4 is the best comprehensive algorithm among the YOLO series algorithms; particularly, YOLOv4 adds more tricks to YOLOv3, making YOLOv4 exhibit better accuracy in handling small targets. YOLOv4 exhibits better accuracy for small targets. In the design of the CNN, YOLOv4 adds CSPNet to the Darknet53 structure. CSPNet can enhance the learning ability of the CNN, which can increase the accuracy, reduce the computational bottleneck, and lower the memory cost while maintaining its lightweight qualities.

YOLOv4-Tiny is a simplified version of YOLOv4, which removes some feature layers based on YOLOv4 and keeps only two independent prediction branches of sizes 13 × 13 and 26 × 26 for predicting large- and medium-sized objects, respectively, and the accuracy is reduced compared with YOLOv4. The reason for the low detection accuracy is that the YOLOv4-Tiny backbone network is relatively shallow and cannot extract higher-level semantic features. The proposed target detection algorithm requires high detection speed and detection accuracy. After the speed and detection accuracy tradeoff, we improve YOLOv4-Tiny. The principle of YOLOv4 is described as follows.

The image is divided into grid cells of size $S \times S$ by YOLOv4 before entering the neural network in which grid the center of the object is in the image, and the corresponding mesh

is only responsible for predicting the object. Predict B bounding boxes per grid and give the confidence of that box [37]; each box contains five variables, as defined in Equation (6).

$$T = [\mathrm{x}, \mathrm{y}, \mathrm{w}, \mathrm{h}, \mathrm{confidence}] \tag{6}$$

where x and y are the target prediction frame centroid position; w and h are the prediction frame's width and height, respectively; confidence is the confidence of the prediction category, and its calculation formula is given by Equation (7).

$$\mathrm{Confidence} = \mathrm{P}_r(\mathrm{Object}) \times IOU_{pred}^{truth}, \mathrm{P}_r(\mathrm{Object}) \in \{0, 1\} \tag{7}$$

In addition, there is Class information; *Class* denotes the class of its own dataset, which is represented in Equation (8).

$$Class = [Class_1, Class_2, \cdots, Class_C] \tag{8}$$

This yields the final tensor output, which is calculated as shown in Equation (9).

$$Final\,output\,tensor = S \times S \times (5 \times B + Class) \tag{9}$$

where B is generally taken as 2, and $Class$ is the number of categories in the dataset.

YOLOv4 characterizes the corresponding loss function based on the squared sum of the errors between the predicted and true borders and the crossentropy of the probabilities between the target prediction and true categories. In YOLOv4, the loss is divided into three parts: one is the error brought by the x, y, w and h, which is the loss brought by the bounding box location; another is the error brought by the category; the third is the error brought by the confidence level. Unlike YOLOv3, YOLOv4 uses CIOU. CIOUconsiders the scale information on the overlap, center distance, and aspect ratio of borders based on IOU; the formula of the loss function is given by (10)–(14).

$$Loss = L_{CIOU} + L_{conf} + L_{cls} \tag{10}$$

$$\begin{aligned} L_{conf} = &\sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\hat{O}_i \log(O_i) + \left(1 - \hat{O}_i\right) \log(1 - O_i)\right] - \\ &\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left[\hat{O}_i \log(O_i) + \left(1 - \hat{O}_i\right) \log(1 - O_i)\right] \end{aligned} \tag{11}$$

$$L_{cls} = -\sum_{i=0}^{S^2} 1_{ij}^{obj} \sum_{o \in class} \left[\hat{P}_i(o) \log(p_i(o)) + \left(1 - \hat{P}_i(o)\right) \log(1 - P_i(o))\right] \tag{12}$$

$$L_{IOU} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(2 - w^{gt} \times h^{gt}\right) [1 - CIOU(X, Y)] \tag{13}$$

$$CIOU(X, Y) = IOU(X, Y) - \frac{\rho^2(X_{ctr} Y_{ctr})}{m^2} - uv \tag{14}$$

where S represents the grid size; S^2 represents the detection box size; B denotes the number of bounding boxes; 1_{ij}^{obj} indicates that if a target is found in the bounding box at ij, the value is 1, and it is 0 otherwise; and 1_{ij}^{noobj} indicates that if no target is found in the bounding box at ij, the value is 1, and it is 0 otherwise. To balance the influence of too many "nonobject" bounding boxes, the penalty weight coefficient λ_{noobj} is added, which is generally taken as 0.5, to balance too many "nonobject" cells; $\hat{O}_i$ is the IOU (confidence score) of the predicted and labeled boxes; O_i is the confidence score generated by the network; w^{gt} and h^{gt} are the width and height of the real frame, respectively; $IOU(X, Y)$ is the intersection ratio between

the predicted frame X and the real frame Y; $\rho^2(X_{ctr}Y_{ctr})$ is the Euclidean distance between the center point of the predicted frame and real frames; m is the diagonal distance of the minimum closed region containing both the predicted and real frames; u is the balance adjustment parameter; and v is the parameter measuring the consistency of the aspect ratio [38].

With the development of target detection algorithms, the latest algorithm in the YOLO family is currently YOLOv5. Studies have shown that YOLOv5 has faster execution efficiency and is higher than YOLOv4 in terms of training speed. YOLOv4 uses the Darknet-based framework to acquire targets with higher accuracy and execution efficiency than other target detection algorithms [39]. To further evaluate the advantages of both, the two algorithms are analyzed from the activation function and loss perspectives. YOLOv4 uses the Mish activation function with higher complexity and outperforms the activation function of YOLOv5 in the benchmark test. Comparing the two algorithms from the loss perspective, the bounding box loss in YOLOv5 uses GIoU in the early stage and CIoU in the later stage, and CIoU is used in YOLOv4. Compared with other methods, CIoU brings faster convergence and better performance, and the prediction box is more in line with the real box [40]. In general, YOLOv4 is stronger in performance and YOLOv5 is more efficient in execution. In this paper, we are more concerned with the recognition accuracy of targets in complex scenarios, and we choose YOLO v4 for improvement and optimization from the perspective of compromise between accuracy and execution efficiency.

3.2. Attention Model

The AM in deep learning is similar to the attention mechanism in human vision, which is to focus attention on the target area that needs to be focused in a large amount of information, and the core goal is to select the information that is more critical to the current task goal from a large amount of information to obtain more detailed information. This enables humans to use limited attentional resources to quickly obtain high-value information from a large amount of information, which greatly improves the efficiency of the brain in processing information. Algorithmically, we can analogize the attention mechanism to pooling, i.e., pooling in convolutional neural networks is seen as a special average weighted attention mechanism, or the attention mechanism is a general pooling method with preferences for input assignment. The attention mechanism is the attention to the input weight assignment, which was first used in encoder-decoder, where the attention mechanism obtains the input variables of the next layer by weighting the hidden states of the encoder over all time steps. The addition of the attention model allows neural networks to automatically learn the region of interest in images, thereby achieving the goal of saving resources and quickly obtaining the most effective information, improving recognition accuracy without significantly increasing the training difficulty [41]. Currently, the common attention model is the channel, spatial, and convolutional block attention models. The channel attention model generates and scores the mask for the channel, represented by SE-Net and CAM; the spatial attention model generates and scores the mask for the space, represented by SAM; the hybrid domain attention model evaluates and scores both the channel and spatial attention, represented by BAM and CBAM [42].

3.3. The Proposed Algorithm

YOLOv4 is not perfectly adapted to our dataset because the CNN of YOLOv4 is more complex and the training time is longer; in addition, the prediction module of YOLOv4 targets three sizes of objects, whereas our targets are mainly CM and materials as the detection targets, which are large and small in size, respectively, and the features are distinct. We choose the corresponding simplified version for improvement, aiming to make the trained model achieve a higher recognition rate, faster recognition speed, and shorter training time.The CNN designed in this paper for CM and materials in low contrast is derived from the improved fusion of VGG19, CAM, and YOLOV4-Tiny improvements with the following details.

In the selection of the FE network, we choose to optimize and improve the network structure of VGG19. VGG19's network structure is deep, robust to extract deep features of a target, and mature, with a high recognition accuracy, which meets the improvement requirements of this study. In addition, we incorporate the idea of a residual network to suppress the phenomenon of network degradation and use jump connections in the residual blocks inside the network to alleviate the problem of gradient disappearance brought by increasing depth in deep neural networks.

In order to make the trained model with high robustness, this paper adopts style migration and sliding window segmentation to increase the diversity of the dataset; firstly, considering that the YOLOv4-Tiny base network framework only adopts the form of layer-by-layer connection, which has insufficient feature extraction ability, the paper makes use of the VGG19 network idea to deeply connect the low-level information with the high-level information. Meanwhile, in order to achieve recognition of targets at different scales, the detection head of YOLOv4-Tiny and the backbone FE layer of VGG19 are selected and fused as the prediction module. The detection head size of YOLOv4-Tiny is changed from 13 × 13 and 26 × 26 to 8 × 8 and 16 × 16, respectively, for our dataset. The 8 × 8 and 16 × 16 detection heads are used to detect large- and medium-sized targets, respectively. Finally, to make the designed network demonstrate higher accuracy and efficiency, we add CAM between the backbone and prediction module of CNN. The correlation of features between different channels is learned by assigning weights to the features of each channel to enhance the transmission of deep information of the network structure, thereby reducing the interference of a complex background on target recognition. The detection network exhibits fewer network layers, occupies low memory, and demonstrates little impact on the training speed, which can effectively reduce the interference of low-contrast environmental background information on the target and improve the recognition accuracy of the target.

We optimize the network structure by changing the size of some convolutional layers and adding the residual mesh structure, so that the three network structures of the backbone, CAM, and prediction module are smoothly connected. Through the experimental comparison, the effect of this step can increase the training speed and detection accuracy of the network to some extent. Because the anchor size of YOLOv4 is obtained by clustering on the COCO dataset, which is not consistent with the target material size needed in this study, to improve the model recognition accuracy and make the proposed algorithm suitable for this dataset, the anchor box size needs to be reclustered. We use the K-means++ clustering algorithm to recluster the dataset, and the anchor sizes obtained are 94, 115, 250, 69, 172, 157, 311, 133, 170, 319, 373, and 322. The block diagram of the proposed improved algorithm is shown in Figure 6.

An image is automatically scaled to 512 × 512 × 3 in size before entering the algorithm; it first passes through a convolutional layer for size transformation and initial FE, and then, it passes through six residual modules for depth FE, cascading the low- and high-level information. After the second residual module, the maximum pooling layer is added between each interval to enhance the extraction effect on image features. The final extracted result enters CAM for secondary filtering of feature information in the region of interest at each scale, filtering the redundant feature information and retaining the important feature information. Finally, the information is further extracted into the prediction module, and the detection results of large targets are output through an 8 × 8 × 24 convolution kernel, whereas part of the feature information is upsampled through a 16 × 16 × 24 convolution kernel, and the detection results of medium targets are output. The algorithm pseudocode is shown in Table 2.

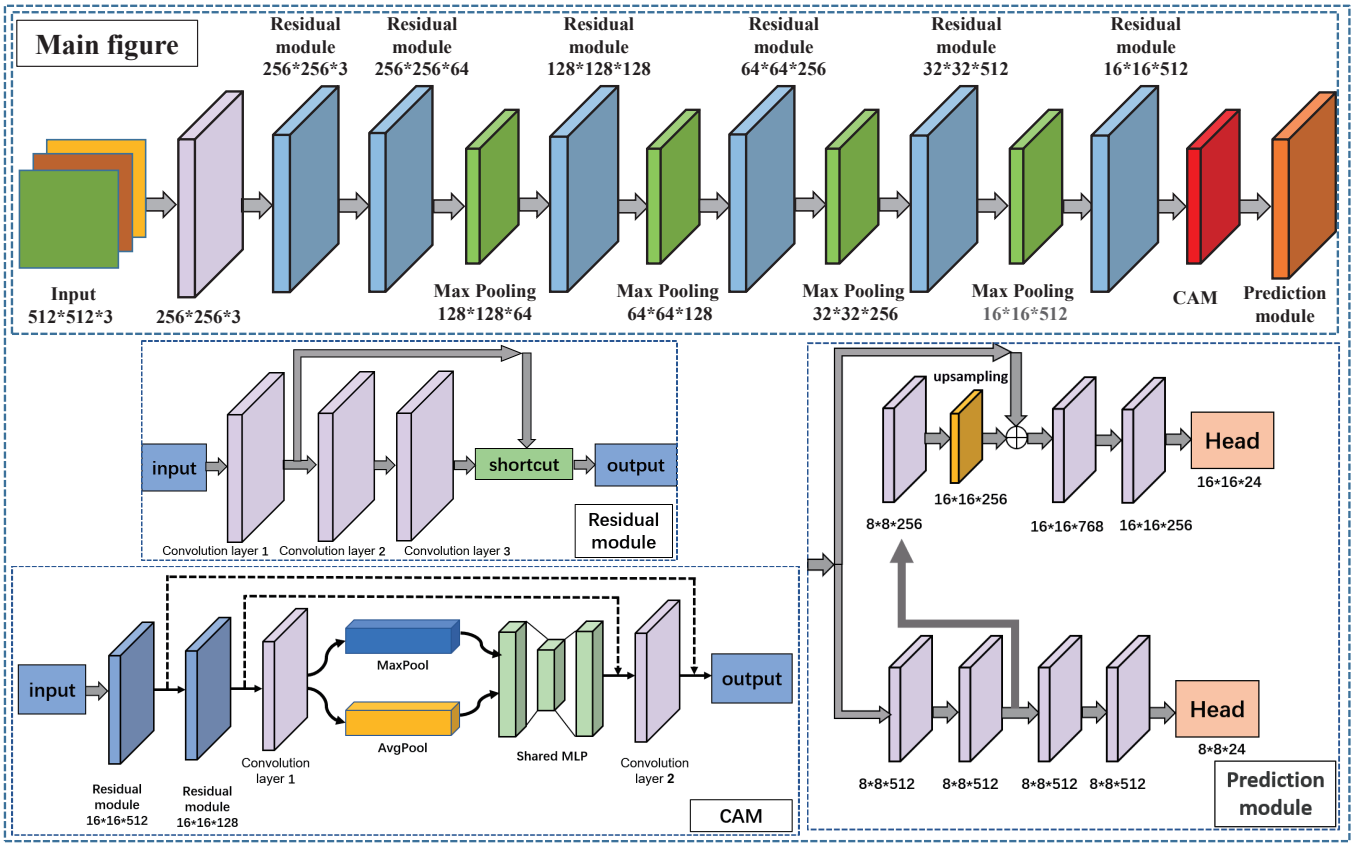

Figure 6. Block diagram of the improved algorithm.

Table 2. Pseudo code of algorithm.

Input: image 1. Hyperparameter setting: batch = 128, subdivisions = 8, momentum = 0.9, learning_rate = 0.00261, max batches = 20,000, momentum = 0.9, decay = 0.0005. 2. Data set preprocessing (2.1) In order to improve the model recognition accuracy and make the designed network more suitable for this dataset, the Kmeans++ algorithm was used to re-anchor box sizes, and the anchor sizes were obtained as 94,115, 250, 69, 172,157, 311,133, 170,319, 373 and 322. (2.2) To increase the diversity of the dataset, Equations (1)–(4) are used for style migration of images, while Equation (5) sliding window segmentation is used to achieve sample diversity. 3. The network input image size is 512 × 512 × 3, combined with Equations (10)–(14), and the modified YOLOv4-Tiny model is used to train the preprocessed sample data. The model is tested using validation and test sets to obtain the class of the target.
Output: target recognition results.

3.4. Algorithm Summary

To improve the recognition accuracy of the loader for CM and materials in normal and low-contrast environments, a YOLOv4-Tiny improved CM and material recognition algorithm is proposed. In this study, first, a diverse dataset is constructed by adopting manual shooting and image enhancement; second, a CNN is constructed by fusing VGG19 with YOLOv4-Tiny and incorporating CAM into it. We use the K-means++ algorithm to recluster on the labeled sample set to obtain the new anchor box size. Then, the constructed network is used to train its sample dataset until the loss function does not converge. Finally, the trained model is used to predict the static or dynamic target image, and the class and

location of the target are labeled in the recognition image. The flowchart of the proposed algorithm is shown in Figure 7.

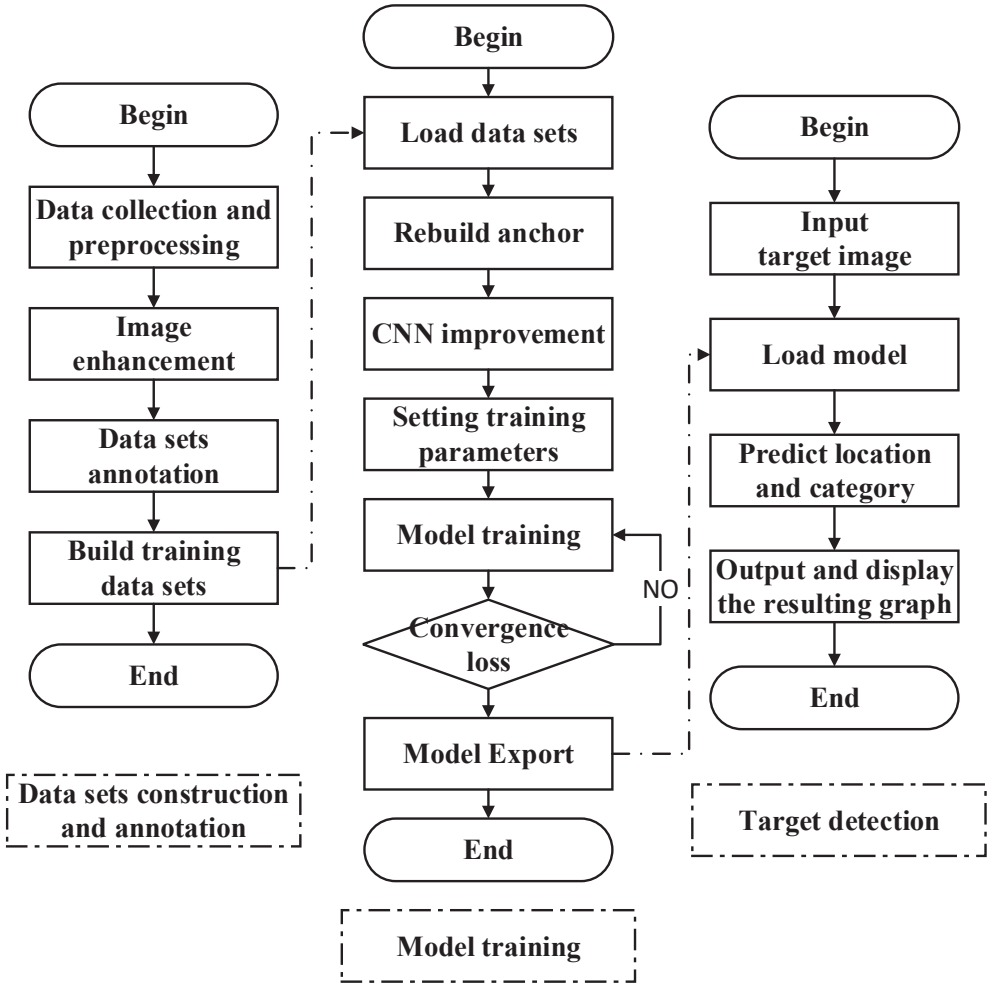

Figure 7. Algorithm summary flow chart.

4. Experiments and Discussion

4.1. Evaluation Criterion

To evaluate the performance of the trained network models, we employed precision, recall, F1-score, and mAP as evaluation metrics. In addition, to compare and analyze the performance of different algorithms, precision–recall (PR) curves are used to visualize and compare different algorithms.

Precision describes how many positive cases predicted by the binary classifier are true positive cases, i.e., how many of the positive cases predicted by the binary classifier are accurate; the calculation method is shown in Equations (15) and (16). The F1-score is a statistical measure of the accuracy of a binary classification model. The F1-score can be considered a weighted average of the model accuracy and recall and is defined as the summed average of the accuracy and recall [43], calculated as in Equations (17)–(19). P in the PR curve represents the precision, and R represents the recall; the PR curve represents the relationship between precision and recall. In general, the recall and precision are set as the horizontal and vertical coordinates, respectively. The area level of a PR curve is an indicator of the strength and weakness of the algorithm comparison [44].

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

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{16}$$

$$F1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{17}$$

$$\text{AP}_c = \frac{1}{11} \sum_{c \in \{0.0,\ldots,1.0\}} \rho interp(c) \tag{18}$$

$$pinterp(c) = max\rho(\tilde{c}) \tag{19}$$

4.2. Comparison of Different Algorithms

Using the same dataset, we compare four target detection algorithms: Faster-RCNN, single-shot detection (SSD), YOLOv4-Tiny, and the proposed algorithm. The evaluation metrics, i.e., precision, recall, F1-score, and mAP, are compared and analyzed, and data are finally visualized using PR curves. The development platform information is shown in Table 3.

Table 3. Development platform information.

Name	Parameter
Operating System	Ubuntu20.04
CPU	I7-11700F
GPU	RTX3060 Dual Graphics Card
Memory	32G
CUDA	11.1

During experiments, several algorithms were set to the same parameters for training. The model hyperparameters were set to a batch size of 128, subdivisions of 8, max batches of 20,000, and momentum and decay of weights of 0.9 and 0.0005, respectively. The initial learning rate was set to 0.00261, and the learning rate was reduced by a factor of 10 at 80% of the maximum number of iterations; at 90% of the maximum number of iterations, the learning rate was reduced by a factor of 10 again, and the weights were saved every 500 training sessions.

The evaluation metrics of Faster-RCNN, SSD, YOLOv4-Tiny, and the proposed algorithm were compared at IOUs of 0.5 and 0.75; from Table 4, the overall evaluation metrics show that the proposed algorithm demonstrates the best performance with an mAP of 96.82% and detection speed of 134.4 fps.

In this paper, we compare Faster-RCNN, SSD, YOLOv4-Tiny and our method evaluation metrics at an IOU of 0.5 and 0.75, respectively. Meanwhile, we also compare YOLOv5 data at an IOU of 0.5. Considering the balanced performance of the YOLOv5 algorithm, we finally chose to use the YOLOv5s model training dataset. From Table 4, we can see that based on the overall evaluation metrics, our method with YOLOv5 algorithm model performance is optimal, although the YOLOv5 algorithm is slightly higher than our algorithm in terms of execution efficiency. However, considering factors such as accuracy requirements and customizability of the loader in low contrast environments, this project still chose to improve on the YOLOv4-Tiny algorithm.

Table 4. Comparison of evaluation indexes of different algorithms.

Methods	Iou Thresh	Precision	Recall	F1	mAP	FPS
Faster RCNN	0.5	0.59	0.98	0.74	92.30%	12.11
	0.75	0.42	0.68	0.51	53.51%	
SSD	0.5	0.94	0.93	0.94	96.31%	78.22
	0.75	0.75	0.74	0.75	66.08%	
YOLOv4-Tiny	0.5	0.97	0.91	0.94	93.85%	414.7
	0.75	0.87	0.81	0.84	76.04%	
YOLOv5	0.5	0.97	0.95	0.96	95.8%	142.9
Our method	0.5	**0.99**	**0.95**	**0.97**	**96.82%**	**134.4**
	0.75	**0.95**	**0.91**	**0.93**	**90.16%**	

4.3. Experimental Comparison and Analysis

Because the proposed algorithm was improved on the basis of YOLOv4-Tiny, we performed a longitudinal comparison with YOLOv4-Tiny, and a horizontal comparison for the YOLOv5 algorithm. From the summary plot of the PR curves of YOLOv4-Tiny and the proposed algorithm (Figure 8), overall, the proposed algorithm is better than YOLOv4-Tiny, with higher recognition accuracy and better robustness. From the figure, the PR curve of the loader category results in the worst effect. Analyzing the reason, the number of loader samples in our dataset is the lowest (1797). In the sample of loaders, multiple loader targets exist in one image. Moreover, some images exist in which the overall outline of the loader is incomplete, which increases the difficulty of recognizing the loader category. Meanwhile, in the data preprocessing stage, we use the sliding window segmentation method to divide one image into multiple images containing partial feature information, which in turn increases the number of incomplete images of the loader category. This part of the images demonstrates inconspicuous features, which will cause a certain bias and eventually lead to a low mAP and poor PR curve performance for the loader category as a whole. From the PR graph of YOLOv5, we can see that the category of loader is less effective, and the recognition of truck is equally poor. Considered together, the improved algorithm of this paper based on YOLOv4-Tiny is more suitable for the practical application of the project.

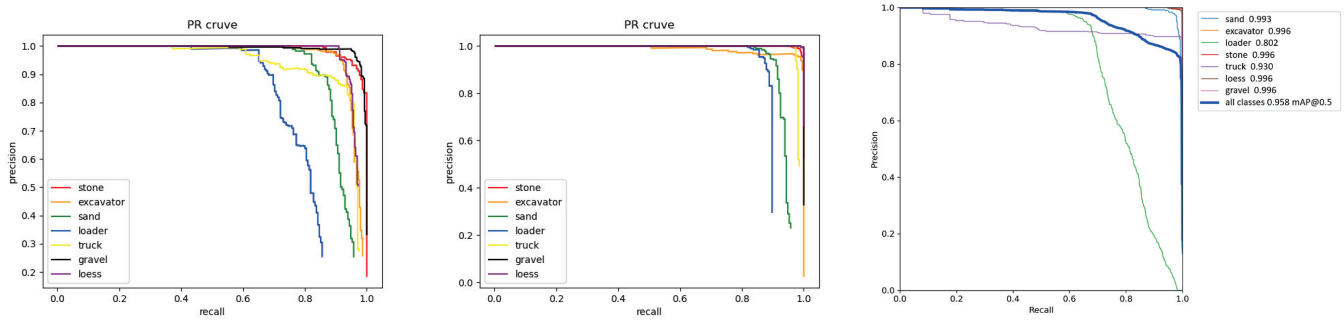

Figure 8. Summary of PR curves for YOLOv4-Tiny, our method and YOLOv5.

By selecting four categories of targets in our dataset, namely, stone, loader, truck, and fine sand, the PR curves of the four algorithms for these four categories were compared horizontally (Figure 9). From the analysis of the PR curve results, Faster-RCNN, SSD, and YOLOv4-Tiny demonstrated good results in individual categories, but their overall performance was poor, whereas the proposed algorithm demonstrated good results in each

category. The PR curve produced by the proposed algorithm is the most stable and exhibits the best performance.

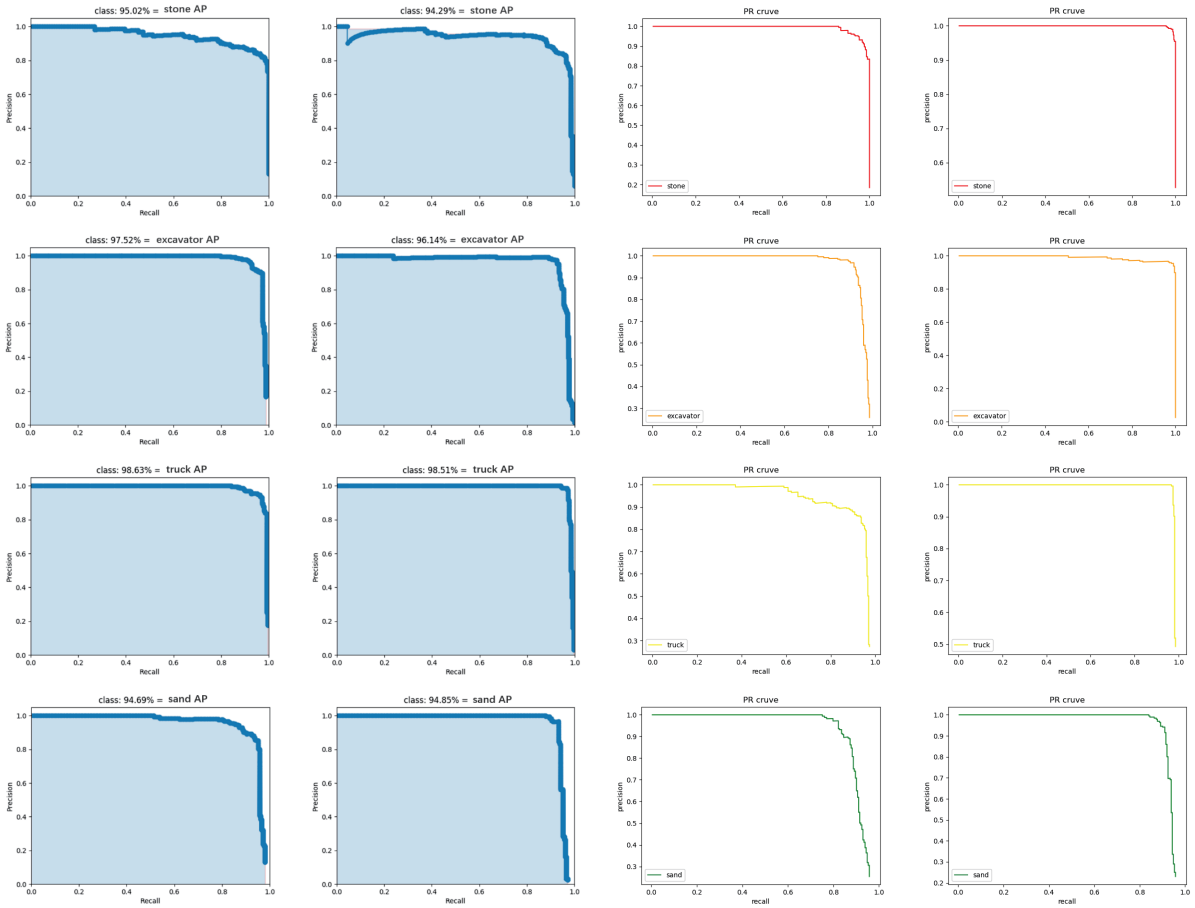

Figure 9. Comparison of PR curves of Faster-RCNN, SSD, YOLOv4-Tiny, and the proposed algorithm for four categories of targets.

To compare the improvement degree of the proposed algorithm with YOLOv4-Tiny in more detail, we compare each category of the two algorithms separately. PR curve comparison results are shown in Figure 10; the PR curve results of the proposed algorithm are significantly better than that of YOLOv4-Tiny, so the conclusion exists that the proposed algorithm achieves the research purpose.

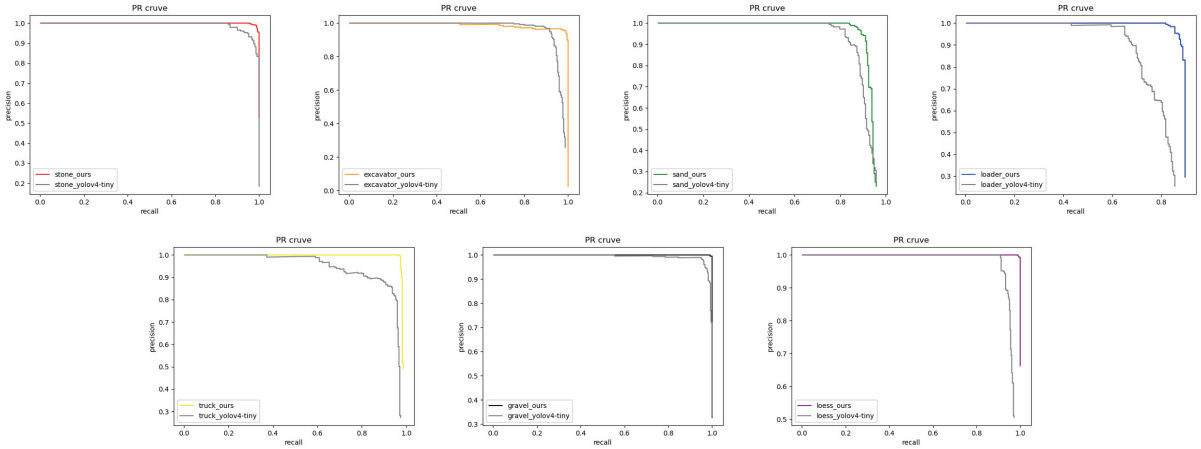

Figure 10. Summary of PR curves for YOLOv4-Tiny and our method.

4.4. Experimental Results Graph

The image recognition results generated by migration learning are shown in Figure 11; the images of an excavator and loess migrating in rain, snow, and fog are recognized, but the highest recognition rate is only 0.74. Because the percentage of such images is not very high, it demonstrates a certain impact on the recognition accuracy.

Figure 11. Migration learning image recognition result graph.

From the high-exposure and low-contrast image recognition result plots in Figure 12, the recognition rate reaches a relatively high level due to the sufficient samples in the dataset. From the recognition result graph, the proposed algorithm still exhibits a high recognition rate under poor external conditions.

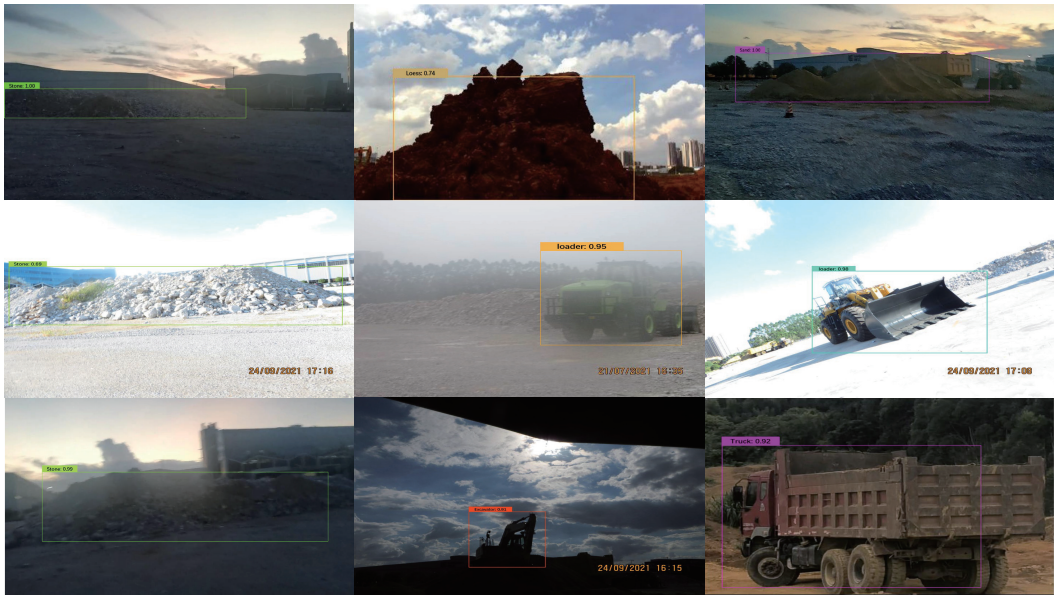

Figure 12. High-exposure and low-contrast image recognition results.

5. Conclusions

In summary, we focus on a recognition algorithm for CM and materials in normal and low-contrast environments for intelligent unmanned loaders. First, we adopt manual shooting and technical enhancement to acquire images and perform image enhancement, use different shooting techniques for manual enhancement, and use style migration and sliding window segmentation to enhance the sample dataset by technical means. Then, the backbone network structure of VGG19 is combined with the detection and prediction module of YOLOv4-Tiny using migration learning, and CAM is incorporated to make the network smoothly connected by changing the structure of some convolutional lay-

ers. Finally, the network details are modified and optimized for the employed dataset. By comparing Faster-RCNN, SSD, YOLOv4-Tiny, YOLOv5 and the proposed algorithm, it is obvious in the PR curve and evaluation metric comparison that the proposed algorithm demonstrates the best performance, with 96.82% mAP and 134.4-fps detection speed, which are ideal. In the experimental result graph, the proposed algorithm achieves better results not only in image recognition with style migration but also in low-contrast image recognition. Overall, the proposed target detection algorithm mitigates the problem of the recognition accuracy of loaders in different environments and different contrast scenes. This study lays a good foundation for developing intelligent unmanned loaders, which is conducive to improving the intelligence level of CM, and achieves the research purpose.

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

Funding: This work was supported by the Guangxi Natural Science Foundation (2021GXNSFBA075029) and Innovation Project of Guangxi Graduate Education (YCSW2021309).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are included within the article.

Conflicts of Interest: The authors declare that they have no conflict of interest with this work.

Abbreviations

AM	Attention mechanism
CAM	Channel attention module
CM	Construction machinery
FE	Feature extraction
CNN	Convolutional neural network
RCNN	Region-based CNN
YOLOv4	YOLO version 4
UAV	Unmanned aerial vehicle
mAP	Mean average precision
SE	Squeeze and Excitation
PR	Precision–recall
SSD	Single-shot detection

References

1. Li, H.J. A "5G remote control + semi-intelligent" driverless control scheme for loaders. *Port Handl.* **2021**, *1*, 51–53.
2. Manjunath, B.S.; Ma, W.Y. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell.* **1996**, *18*, 837–842. [CrossRef]
3. Lazebnik, S.; Schmid, C.; Ponce, J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, NY, USA, 17–22 June 2006 ; pp. 2169–2178.
4. Chai, Y.; Lempitsky, V.; Zisserman, A. Symbiotic segmentation and part localization for fine-grained categorization. In Proceedings of the IEEE European Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; pp. 321–328.
5. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
6. Karen, S.; Andrew, Z. Very Deep Convolutional Networks for Large-scale Image Recongnition. In Proceedings of the International Conference of Learning Representation, San Diego, CA, USA, 7–9 May 2015.
7. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 27–30.
8. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

9. Girshick, R.B.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR* **2013**, *abs*, 1311–2524.
10. Girshick, R. Fast R-CNN. In Proceedings of the International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; IEEE Press: Piscataway, NJ, USA, 2015; pp. 1440–1448.
11. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural lnformation Processing Systems, Montreal, QC, Canada, 7–12 December 2015; MIT Press: Cambridge, MA, USA, 2015; pp. 91–99.
12. 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 (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
13. Wang, H.X.; Cao, J.; Qiu, C.; Liu, Y.H. A multi-target detection method for aerial images based on improved YOLOv4. *Electro-Opt. Control* **2022**, 1–6.
14. Yu, Q.D.; Yang, M.; Yuan, H.; Liang, K. Lightweight YOLOv4-based target detection algorithm for pigs. *J. China Agric. Univ.* **2022**, *27*, 183–192.
15. Huang, H.X.; Tang, X.D. An improved YOLOv4 algorithm for remote sensing target detection. *Electron. World* **2021**, *22*, 34–35.
16. Li, B.; Wang, C.; Ding, X.Y.; Ju, H.J.; Guo, Z.P.; Li, J.Y. Improved surface defect detection algorithm for YOLOv4. *J. Beijing Univ. Aeronaut. Astronaut.* **2022**, 1–10.
17. Guo, Z.Y.; Gao, G.F. Research on algorithm for detecting mixed pedestrian-vehicle traffic under complex intersections based on YOLO v4. *Inf. Technol. Informatiz.* **2021**, *2*, 236–240.
18. Wang, L.S.; Qin, M.X.; Lei, J.Y.; Wang, X.F.; Tan, K.Z. An improved YOLOv4-Tiny based method for blueberry ripeness identification. *J. Agric. Eng.* **2021**, *37*, 170–178.
19. Wu, J.; Han, L.X.; Shen, Y.; Wang, S.; Huang, F. Improved YOLOv4-Tiny based UAV aerial target detection. *Electro-Opt. Control* **2022**, 1–8.
20. Andriyanov, N.; Khasanshin, I.; Utkin, D.; Gataullin, T.; Ignar, S.; Shumaev, V.; Soloviev, V. Intelligent System for Estimation of the Spatial Position of Apples Based on YOLOv3 and Real Sense Depth Camera D415. *Symmetry* **2022**, *14*, 148. [CrossRef]
21. Kuznetsova, A.; Maleva, T.; Soloviev, V. Using YOLOv3 algorithm with pre-and post-processing for apple detection in fruit-harvesting robot. *Agronomy* **2020**, *10*, 1016. [CrossRef]
22. Kamyshova, G.; Osipov, A.; Gataullin, S.; Korchagin, S.; Ignar, S.; Gataullin, T.; Terekhova, N.; Suvorov, S. Artificial Neural Networks and Computer Vision's—Based Phytoindication Systems for Variable Rate Irrigation Improving. *IEEE Access* **2022**, *10*, 8577–8589. [CrossRef]
23. Osipov, A.; Pleshakova, E.; Gataullin, S.; Korchagin, S.; Ivanov, M.; Finogeev, A.; Yadav, V. Deep Learning Method for Recognition and Classification of Images from Video Recorders in Difficult Weather Conditions. *Sustainability* **2022**, *14*, 2420. [CrossRef]
24. Yan, B.; Fan, P.; Lei, X.; Liu, Z.; Yang, F. A real-time apple targets detection method for picking robot based on improved YOLOv5. *Remote Sens.* **2021**, *13*, 1619. [CrossRef]
25. Zhao, J.; Zhang, X.; Yan, J.; Qiu, X.; Yao, X.; Tian, Y.; Zhu, Y.; Cao, W. A wheat spike detection method in UAV images based on improved YOLOv5. *Remote Sens.* **2021**, *13*, 3095. [CrossRef]
26. Chen, Y.; Liu, L.; Tao, J.; Chen, X.; Xia, R.; Zhang, Q.; Xiong, J.; Yang, K.; Xie, J. The image annotation algorithm using convolutional features from intermediate layer of deep learning. *Multimed. Tools Appl.* **2021**, *80*, 4237–4261. [CrossRef]
27. Chen, Y.; Liu, L.; Phonevilay, V.; Gu, K.; Xia, R.; Xie, J.; Zhang, Q.; Yang, K. Image super-resolution reconstruction based on feature map attention mechanism. *Appl. Intell.* **2021**, *51*, 4367–4380. [CrossRef]
28. Li, D.; Huang, C.; Liu, Y. YOLOv3 Target Detection Algorithm Based on Channel Attention Mechanism. In Proceedings of the 2021 3rd International Conference on Natural Language Processing (ICNLP), Beijing, China, 26–28 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 179–183.
29. Mo, J.; Zhou, Y. The image inpainting algorithm used on multi-scale generative adversarial networks and neighbourhood. *Autom. Časopis Za Autom. Mjer. Elektron. Računarstvo I Komun.* **2020**, *61*, 704–713. [CrossRef]
30. Huang, R.; Gu, J.; Sun, X.; Hou, Y.; Uddin, S. A Rapid Recognition Method for Electronic Components Based on the Improved YOLO-V3 Network. *Electronics* **2019**, *8*, 825. [CrossRef]
31. Liu, T.T.; Zhang, Y.J.; Xiong, S.T. A double-exposure fusion processing algorithm for low-light image enhancement. *Electron. Sci. Technol.* **2021**, *34*, 34–39.
32. Chen, H.; Lai, H.C.; Gao, G.X.; Wu, H.; Qian, X.Z. Sand and dust image enhancement based on multi-exposure image fusion. *J. Photonics* **2021**, *50*, 0910003. [CrossRef]
33. Ren, J.J.; Zhang, W.Z.; Zhang, W.W.; Wang, Y.F.; Cui, J.J.; Li, C.L.; Liu, Y.; Liu, X.Q. A study on the artistic style migration of blue and white porcelain decoration. *J. Light Ind.* **2021**, 1–11.
34. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
35. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
36. Jiang, Z.C.; Zhao, L.Q.; Li, S.Y.; Jia, Y.F. Real-time object detection method based on improved YOLOv4-Tiny. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022.

37. Zhao, Y.; Chen, Q.; Cao, W.; Yang, J.; Xiong, J.; Gui, G. Deep learning for risk detection and trajectory tracking at construction sites. *IEEE Access* **2019**, *7*, 30905–30912. [CrossRef]
38. Li, X.; Pan, J.; Xie, F.; Zeng, J.; Li, Q.; Huang, X.; Liu, D.; Wang, X. Fast and accurate green pepper detection in complex backgrounds via an improved Yolov4-Tiny model. *Comput. Electron. Agric.* **2021**, *191*, 106503. [CrossRef]
39. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
40. Misra, D. Mish: A self regularized non-monotonic activation function. *arXiv* **2021**, arXiv:1908.08681.
41. Park, J.; Woo, S.; Lee, J.-Y.; Kweon, I.S. BAM: Bottleneck Attention Module. *arXiv* **2018**, arXiv:1807.06514.
42. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the European conference on computer vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
43. Yang, K.; Song, Z. Deep Learning-Based Object Detection Improvement for Fine-Grained Birds. *IEEE Access* **2021**, *9*, 67901–67915. [CrossRef]
44. Hao, W.; Xiao, N. Research on Underwater Object Detection Based on Improved YOLOv4. In Proceedings of the 2021 8th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS), Beijing, China, 10–12 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 166–171.