Journal of Computer Science and Artificial Intelligence
ISSN: 3078-8242 (Print), ISSN: 3078-2309 (Online) | Vol. 3, No. 1, 2025

Lightweight Model-Based Intrusion Detection in Construction Scenes

Haoyang Zhao

School of Mechatronics and Vehicle Engineering, East China Jiaotong University, Jiangxi, 330013, China
Zhy18612713011@163.com

Abstract: Intrusion detection in construction scenes can effectively reduce the occurrence of hazardous incidents. Current detection methods, while effective, are often too complex. This paper proposes a lightweight monitoring model based on convolutional neural networks (CNNs). First, the model is trained using a dataset to achieve high accuracy. Then, the model is lightweighted using CNNs. Simulation results show that the model can maintain accuracy while occupying a smaller volume.

Keywords: Safety helmet detection, Yolov5, Convolutional Neural Network.

1. Introduction

Risk detection technology is a key technology in intelligent manufacturing and quality control, widely used in various fields such as production manufacturing, medical diagnosis, and autonomous driving. Its main purpose is to ensure product quality, reduce safety accidents, and improve production efficiency. In the industrial sector, defect detection technology plays a crucial role in modern industrial production. With the advancement of industrialization, many large and dangerous machines lack efficient warning systems, making it difficult to monitor construction site risks and implement corresponding strategies. For example, when a person enters a hazardous area without wearing a helmet, timely warnings are often not issued. This paper designs an optimized algorithm that can quickly identify whether a person entering a certain area is wearing a safety helmet and issue a warning.

2. Research Background

In China, infrastructure serves as a pillar of the national economy. With the rapid development of infrastructure projects, associated risks have surged. Thanks to technological advancements, many construction sites are now equipped with hazard detection systems to prevent casualties. These systems trigger alarms when personnel enter hazardous zones, effectively mitigating potential accidents. The convolutional neural network (CNN) used in this study is a type of deep feedforward neural network incorporating convolutional operations. As a representative algorithm in deep learning, CNNs possess representation learning capabilities, enabling translation-invariant classification of input information through their hierarchical structure. This characteristic has earned them the name "translation-invariant artificial neural networks."

CNNs, with their powerful feature learning capabilities, revolutionized computer vision by outperforming traditional methods in the ImageNet competition. From AlexNet to GoogLeNet and ResNet, a series of milestone architectures have emerged, continuously pushing the boundaries of image recognition performance. However, in complex environments, CNNs still face significant challenges. Variations in lighting distort image brightness and contrast, viewpoint changes cause object shape and pose deformation, background clutter complicates target separation, and occlusions directly compromise object integrity. These factors rigorously test the adaptability and robustness of image recognition algorithms.

A Convolutional Neural Network is a type of deep feedforward neural network with representation learning capabilities, structurally analogous to artificial neural networks and Backpropagation neural networks (multi-layer feedforward networks trained via error backpropagation algorithms). Unlike BP neural networks, CNNs can perform translation-invariant classification of input information based on their hierarchical architecture. Additionally, CNNs are trained using error backpropagation algorithms, enabling them to approximate any continuous function with arbitrary precision while achieving fast convergence speeds [1].

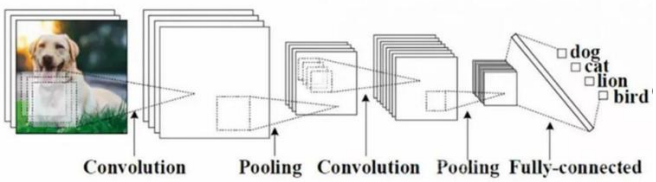

Figure 1. Convolutional Neural Network (Source: Baidu Encyclopedia)

3. Experimental Design and Application

Convolutional Neural Networks (CNNs), after years of development, have been extensively studied for intrusion detection. Zou Guohong and Gai Weixu trained the YOLOv5s model using a training dataset.

During training, the model parameters were iteratively adjusted through forward propagation and backpropagation of the training data to minimize the loss function. Subsequently, hyperparameters such as learning rate, batch size, and regularization parameters were optimized based on validation set performance.

The trained YOLOv5s model was evaluated on a test dataset to calculate critical metrics for object detection tasks, including precision, recall, and mean Average Precision (mAP). Finally, the optimized model was deployed in real-world scenarios to perform inference for target detection tasks, enabling accurate identification and localization of objects in images or videos [2].

Lian Jingmin implemented accurate face detection under complex conditions such as small targets, blurring, and partial occlusion using the PyramidBox target detection algorithm and deep learning techniques. Based on a pre-trained weight model, the algorithm network was trained using a large open-source face detection dataset. The Dlib tool library and Euclidean distance threshold discrimination method was used to detect and classify key facial feature points, and the feasibility of the face recognition algorithm was verified using a self-made face database [3].

Building on previous research, this paper proposes using the YOLO model for training to enable fast and accurate image recognition. Then, through CNNs, the YOLO model is pruned to reduce its volume, thereby optimizing the model. Calculations show that this method is feasible.

4. Dataset Collection

The SHWD dataset was found online. The SHWD dataset is a dataset for safety helmet wearing and personnel detection, containing 7,581 images. In the dataset, 9,044 people are wearing safety helmets, while 111,514 are not. The images in the dataset were collected from Google or Baidu and annotated using LabelImg. The data can be directly loaded in the Pascal VOC format.

5. Principles of Convolutional Neural Networks

5.1. Input

The input is the initial stage of the CNN workflow. At this stage, various types of data (such as images, audio, etc.) are introduced into the network. For image data, it exists in the form of a pixel matrix. If it is a color image (e.g., RGB mode), each pixel contains red, green, and blue channel information. This data is transmitted to the first layer of the network in a specific dimension, providing raw material for subsequent processing.

5.2. Convolutional Layer

The convolutional layer is the core component of a CNN. This layer contains multiple convolutional kernels, which are matrices of specific sizes. During operation, the convolutional kernel slides over the input data with a certain stride and performs convolution operations on the covered areas. The specific operation involves multiplying the elements of the convolutional kernel with the corresponding input data elements and summing them to generate output values. Different convolutional kernels are used to extract different features from the input data, and the output matrices generated are called feature maps. Multiple convolutional kernels generate multiple feature maps, which together constitute the feature extraction results of this layer.

5.3. Activation Function

The activation function introduces nonlinearity into the CNN. Without an activation function, a multi-layer neural network would only be a linear combination of inputs and could not handle complex nonlinear relationships. Activation functions act on the output of the convolutional layer. Common activation functions (such as ReLU) perform nonlinear transformations on the output, enhancing the network's ability to express complex data patterns.

5.4. Pooling Layer

The pooling layer performs downsampling on the feature maps to reduce data dimensionality. Common pooling methods include max pooling and average pooling. Max pooling selects the maximum value in a specified small area as the output, while average pooling calculates the average value in the area as the output. Pooling reduces the amount of data, lowers computational costs, and helps prevent overfitting.

5.5. Fully Connected Layer

After processing through the previous layers, the data enters the fully connected layer. Each neuron in the fully connected layer is connected to all neurons in the previous layer, and its main function is to integrate the features extracted earlier. In the fully connected layer, matrix multiplication and bias addition are used to generate outputs. For different tasks (such as classification or regression), different functions are used in the last layer of the fully connected layer. For example, in classification tasks, the Softmax function is often used to convert the output into a probability distribution of categories.

5.6. Output Layer

The output layer is the final stage of the CNN workflow. The output result depends on the task type. For classification tasks, the output is the predicted class label; for regression tasks, the output is a specific value. This completes the network's processing of the input data and provides the final output.

6. Experimental Results and Analysis

6.1. Training Dataset

When training the SHWD dataset with YOLOv5, data preprocessing is first performed, including converting the data annotation format to the format required by YOLOv5, dividing the dataset into training, validation, and test sets in a certain ratio, and using data augmentation techniques such as random flipping, rotation, scaling, and cropping to expand data diversity. Next, model configuration and parameter adjustment are performed. Based on the characteristics of the dataset and computational resources, an appropriate network architecture version (such as YOLOv5-s, m, l, x) is selected, and hyperparameters such as learning rate, batch size, and training epochs are adjusted. Then, the loss function is defined, using cross-entropy loss to measure classification accuracy and CIoU loss to ensure precise detection box localization. Finally, GPU acceleration is used to speed up the training process and improve training efficiency.

In object detection tasks, the experimental results mainly focus on detection accuracy and recall rate. The mAP (mean Average Precision) comprehensively considers the detection accuracy of different categories of objects. The higher the mAP value, the higher the accuracy of the model in detecting various target objects. The experimental results can be displayed visually, such as marking the location and category labels of detected target objects on the image. This allows for an intuitive view of where the model accurately detected targets and whether there are any false positives or missed detections. For example, in traffic scene object detection, it can be seen whether vehicles, pedestrians, and other targets are correctly detected, and whether the bounding boxes accurately enclose the target objects.

Figure 2. Model Detection Results

Figure 3. Model Detection Results

6.2. Pruning the Model

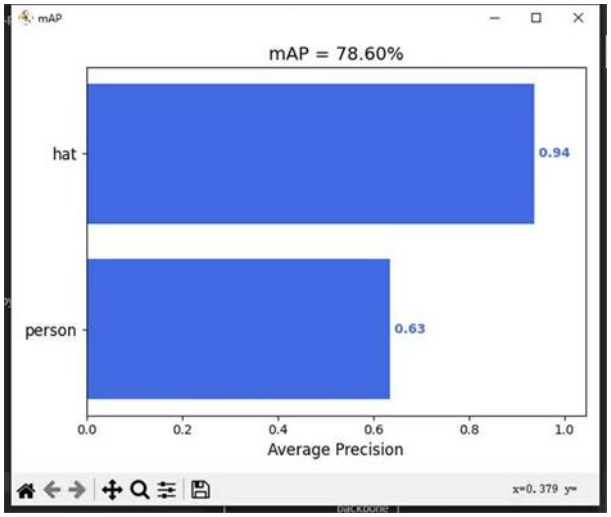

Figure 4. mAP Values

In this paper, after model training, pruning is performed, which significantly optimizes resource utilization and improves operational efficiency. Pruning can significantly reduce the number of model parameters, thereby effectively reducing storage requirements. Taking a neural network model as an example, after pruning many redundant parameters, the volume of the model file is greatly reduced. This feature makes the model easier to deploy on devices with limited storage resources (such as mobile smart terminals and embedded IoT chips). At the same time, during network transmission, the transmission speed can be significantly accelerated, greatly improving resource utilization efficiency. In addition, due to the reduction in model parameters, the computational load during inference is also reduced.

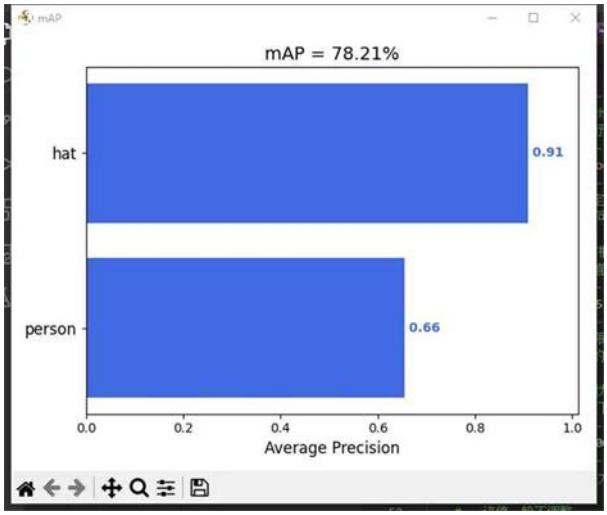

Figure 5. mAP Values

7. Conclusion

Model pruning after training is a highly valuable technical approach that plays an important role in multiple key aspects. In terms of resource utilization and operational efficiency, pruning can significantly reduce model parameters, shrink the model file size, and lower storage requirements, making the model easier to deploy on devices with limited storage resources (such as mobile terminals and IoT chips). At the same time, it speeds up model transmission and improves resource utilization efficiency. Moreover, the reduction in parameters decreases the computational load during inference. In tasks such as image recognition and speech recognition, this can significantly speed up data processing, reduce reliance on hardware computing resources, and allow devices with limited computing power to run models efficiently.

References

[1] Sun, Z. (2024). Research on Early Warning of Unsafe Behaviors of Construction Workers Based on Convolutional Neural Networks. Jianzhu yu Yusuan (Architecture and Budget), (5), 37–39.

[2] Zou, G., & Gai, W. (2024). Research on Camera Intrusion Detection Technology Based on Deep Learning. Longdong Xueyuan Xuebao (Journal of Longdong University), 35(2), 13–18.

[3] Lian, J. (2021). Research on Intelligent Personnel Security Monitoring Technology Based on Images. Harbin Engineering University, 03.