frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 23 May 2024
DOI 10.3389/fphy.2024.1404448

Check for updates

OPEN ACCESS

EDITED BY
Zhenqiu Shu,
Kunming University of Science and Technology, China

REVIEWED BY
Jiaxu Leng,
Chongqing University of Posts and Telecommunications, China
Teng Sun,
Kunming University of Science and Technology, China

*CORRESPONDENCE
Zheng Xu,
✉ zheng.xu@giat.ac.cn

[†]These authors share first authorship

RECEIVED 21 March 2024
ACCEPTED 29 April 2024
PUBLISHED 23 May 2024

CITATION
Chen J, Huang Z, Jiang X, Yuan H, Wang W, Wang J, Wang X and Xu Z (2024), Authenticity identification method for calligraphy regular script based on improved YOLOv7 algorithm.
Front. Phys. 12:1404448.
doi: 10.3389/fphy.2024.1404448

Authenticity identification method for calligraphy regular script based on improved YOLOv7 algorithm

Jinyuan Chen[1†], Zucheng Huang[1†], Xuyao Jiang[1], Hai Yuan[1], Weijun Wang[1,2], Jian Wang[1], Xintong Wang[1] and Zheng Xu[1*]

[1]Guangzhou Institute of Advanced Technology, Guangzhou, China, [2]Shenzhen Cas Derui Intelligent Technology Co., Ltd., Shenzhen, China

A regular calligraphy script of each calligrapher has unique strokes, and a script's authenticity can be identified by comparing them. Hence, this study introduces a method for identifying the authenticity of regular script calligraphy works based on the improved YOLOv7 algorithm. The proposed method evaluates the authenticity of calligraphy works by detecting and comparing the number of single-character features in each regular script calligraphy work. Specifically, first, we collected regular script calligraphy works from a well-known domestic calligrapher and divided each work into a single-character dataset. Then, we introduced the PConv module in FasterNet, the DyHead dynamic detection header network, and the MPDiou bounding box loss function to optimize the accuracy of the YOLOv7 algorithm. Thus, we constructed an improved algorithm named YOLOv7-PDM, which is used for regular script calligraphy identification. The proposed YOLOv7-PDM model was trained and tested using a prepared regular script single-character dataset. Through experimental results, we confirmed the practicality and feasibility of the proposed method and demonstrated that the YOLOv7-PDM algorithm model achieves 94.19% accuracy (mAP50) in detecting regular script font features, with a single-image detection time of 3.1 m and 31.67M parameters. The improved YOLOv7 algorithm model offers greater advantages in detection speed, accuracy, and model complexity compared to current mainstream detection algorithms. This demonstrates that the developed approach effectively extracts stroke features of regular script calligraphy and provides guidance for future studies on authenticity identification.

KEYWORDS

calligraphy works identification, YOLOv7 algorithm, PConv module, DyHead dynamic detection head network, MPDiou loss function

1 Introduction

Calligraphy, as a unique form of artistic expression, has a long history in China and stands out in the progression of human civilization [1]. Due to their significant collection value and potential for appreciation, calligraphy works are highly sought after by collectors both domestically and internationally, particularly those created by renowned ancient calligraphers [2]. However, genuine works by master calligraphers are becoming increasingly scarce, leading to abundant forgeries in the market. Consequently, there is an urgent need for calligraphy authenticity identification.

Traditional methods of calligraphy identification mainly involve three approaches [3]. One relies on experienced calligraphy experts with solid skills and substantial experience for empirical identification [4]. However, subjective factors often influence this method, biasing the identification results. An alternative approach utilizes physical techniques to determine authenticity by examining the presence of seals and analyzing the composition of paper used in the calligraphy work. Nevertheless, as technology advances, forgery techniques have become increasingly sophisticated, with the ability to replicate seals and paper, resulting in identification biases [5]. The third method uses computer-assisted techniques to detect the authenticity of calligraphic works. With the further development of computer science and technology in recent years, many researchers have employed computer-assisted methods to detect the authenticity of calligraphic works. However, computer-assisted methods can be further categorized into two types: one is based on traditional image processing algorithms, such as the calligraphic work authentication method proposed by Zeng [6] based on image recognition and the computer-assisted calligraphy authenticity identification proposed by Pang [7]. The other type employs novel image processing methods based on deep learning, such as Li's [8] evaluation and detection of calligraphic copying based on deep learning.

To address the challenge of the identification bias, this study develops an authenticity identification method for calligraphy regular script based on an improved YOLOv7 algorithm. Specifically, first, we manually annotate the features of individual characters in authentic calligraphy regular script works, followed by feature extraction using deep learning networks. The authenticity of calligraphy works is determined by comparing the number of extracted features from genuine works with the forged ones. This method aims to enhance the accuracy and reliability of calligraphy regular script authenticity identification by combining manual annotation and deep learning techniques.

The traditional algorithmic approach involves image processing, and after conducting feature extraction on the works of a single calligrapher, this approach exhibits relatively high detection accuracy. However, the detection algorithm cannot be directly applied to the works of another calligrapher, thus posing significant limitations. Unlike simplistic image processing schemes, deep learning can automatically learn features and exhibits strong robustness and adaptability, enabling accurate detection and recognition in complex environments. Furthermore, deep learning approaches demonstrate high generalization and are suitable for detecting the works of most calligraphers using the same font style [9]. Deep learning has experienced extensive application and has recently advanced significantly in diverse domains. For instance, Wang [10] employed an improved EfficientNet algorithm to authenticate calligraphic works, efficiently categorizing genuine from fake calligraphic pieces using the two-class classification property of the EfficientNet algorithm. The corresponding experimental results demonstrated significant effectiveness. Xu [11] proposed an improved YOLOv4-Tiny algorithm that effectively detects boats on rivers and lakes, ensuring waterway safety. Hu [12] applied the improved YOLOX algorithm to rapidly detect surface hole defects on aluminum castings, enhancing casting efficiency. Mai made a breakthrough in calligraphy font recognition using DenseNet networks [13]. The advantages of deep learning methods lie in their ability to learn features automatically and possess strong robustness and adaptability in accurate detection and recognition in complex environments. Therefore, utilizing deep learning methods for calligraphy regular script authenticity identification holds great potential and feasibility. Hence, building upon these successful research achievements, we leverage deep learning methods to authenticate calligraphy regular script works. Indeed, by constructing a deep learning model suitable for regular calligraphy script works, we extract and analyze the features of each character and compare these features with those of authentic works to determine the degree of authenticity. However, additional datasets and annotations may be required for training and validating the algorithm model. The proposed authenticity identification method is based on the improved YOLOv7 algorithm evaluating the authenticity of regular calligraphy scripts by detecting and comparing the features in each character.

2 Calligraphy regular script stroke feature detection algorithm based on YOLOv7-PDM

2.1 YOLOv7 algorithm

The YOLOv7 algorithm [14], introduced by the YOLOv4 [15] team, is another significant breakthrough in the YOLO series. Since its proposal at the end of 2022, the YOLOv7 algorithm has received considerable attention from the academic community, as it demonstrates excellent performance with a detection speed ranging from 5 to 160 FPS and exhibits higher detection accuracy and speed levels than current mainstream object detection algorithms. Figure 1 illustrates the structure of the YOLOv7 model [16].

The YOLOv7 algorithm comprises four main components: Input, the feature extraction network known as Backbone, the feature fusion network identified as Neck, and the detection head network referred to as YOLO-Head. Compared to prior YOLO algorithms, YOLOv7 presents innovative improvements in its Backbone, Neck, and YOLO head. The feature extraction network comprises CBS, ELAN, and MP1 convolution modules. The CBS module is a conventional convolution module consisting of regularization and activation functions, whereas the ELAN module is a layer aggregation network that improves efficiency. Additionally, dilation and transformation methods are used to enhance the learning performance of the algorithm model, boosting the model's computational capability while maintaining the original gradient path intact. The MP1 convolution module is formed by adding a Maxpool layer after the CBS module, which forms two branches combined with a Concat module to integrate the characteristics of both branches and enhance the network's ability to extract features. YOLOv7 has modified the SPP module in the Neck to the SPPCSPC module, a revised adaptation of Spatial Pyramid Pooling, to accommodate inputs of varying sizes. This modification reduces the image distortion caused by image processing and overcomes the feature re-extraction problem during convolution. In 2021, Megvii Technology published the PAFPN model, which incorporates the same feature pyramid network structure as YOLOX. Feature fusion between layers is

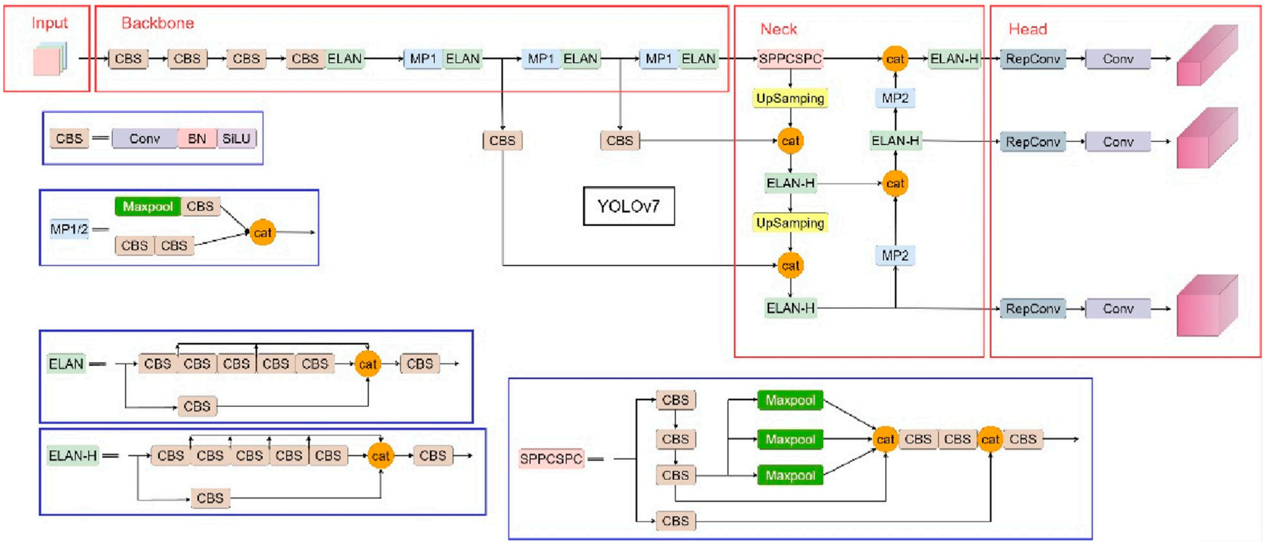

FIGURE 1
Structure of the YOLOv7 algorithm model.

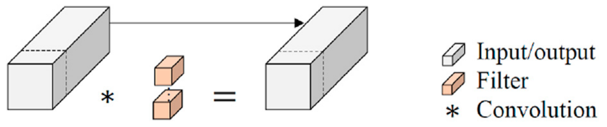

FIGURE 2
Structure of the PConv module.

achieved by passing deep features from bottom to top. Additionally, the Neck network includes the ELAN-H and MP2 modules, where the ELAN-H module aggregates more layers than the ELAN module. The only variation between the MP1 and MP2 modules is the number of channels. In the YOLO-Head, YOLOv7 combines the RepConv module's re-parameterized convolutions with the network structure, balancing speed and accuracy during training.

2.2 PConv module

To enhance the detection accuracy of the YOLOv7 algorithm, we replace the Conv layer in the CBS module with the PConv module from FasterNet [17]. The modified module has been renamed the PBS module. The PConv module plays a vital role in FasterNet, a novel image classification algorithm introduced in CVPR2023, which attains an exceptional TOP-1 accuracy of 83.5% on ImageNet-1k. The structure of the PConv module is illustrated in Figure 2.

PConv addresses higher memory access and reduces the overall computational complexity caused by depthwise separable convolution (DWConv), particularly on I/O-bound devices. DWConv can reduce the computational complexity of Conv by a factor of (number of channels), but the detection accuracy decreases as a result of the cost incurred. To mitigate the accuracy loss, the channel width must be increased to compensate for the decrease in parameter quantity. However, when DWConv is applied with an increased channel width, it introduces higher memory access and generates more computational redundancy. Considering these limitations, PConv performs regular Convolution on a specific group of input channels to extract spatial features while keeping the rest unaltered. The first or last consecutive channels represent the entire feature map for computation with consecutive or regular memory access. Without any loss of generality, it is assumed that the input and output feature maps have the same number of channels. Therefore, Eq. 1 defines the FLOPs of PConv, while Eq. 2 depicts the memory access.

$$FLOPs = h \times w \times k^2 \times c_p^2, \tag{1}$$

$$h \times w \times 2c_p + k^2 \times c_p^2 \approx \mathrm{h} \times w \times 2c_p, \tag{2}$$

In this case, the width and height of the feature map are represented by h and w, respectively. The size of the convolution kernel is denoted by k, and c_p indicates the number of channels affected by regular convolution. This cp value is equivalent to the change from c_{in} to c_p in conventional convolution. However, in practical scenarios, PConv uses only one-fourth of the channels present in c_p which leads to a reduction of FLOPs by 1/16 and memory access by 1/4 compared to conventional convolution.

2.3 DyHead dynamic detection head

The DyHead dynamic detection head network proposed by Microsoft [18] aims to enhance the detection accuracy of the YOLOv7 algorithm. DyHead is a dynamic detection network that introduces attention mechanisms to consolidate different object detection heads innovatively. The core idea of this method is to leverage attention mechanisms to enable interaction among scales (referred to as π_L), spatial (referred to as π_S), and task (referred to as π_C) awareness based on a given feature tensor, denoted as $F \in R^{L \times S \times C}$. Specifically, the π_L attention mechanism facilitates

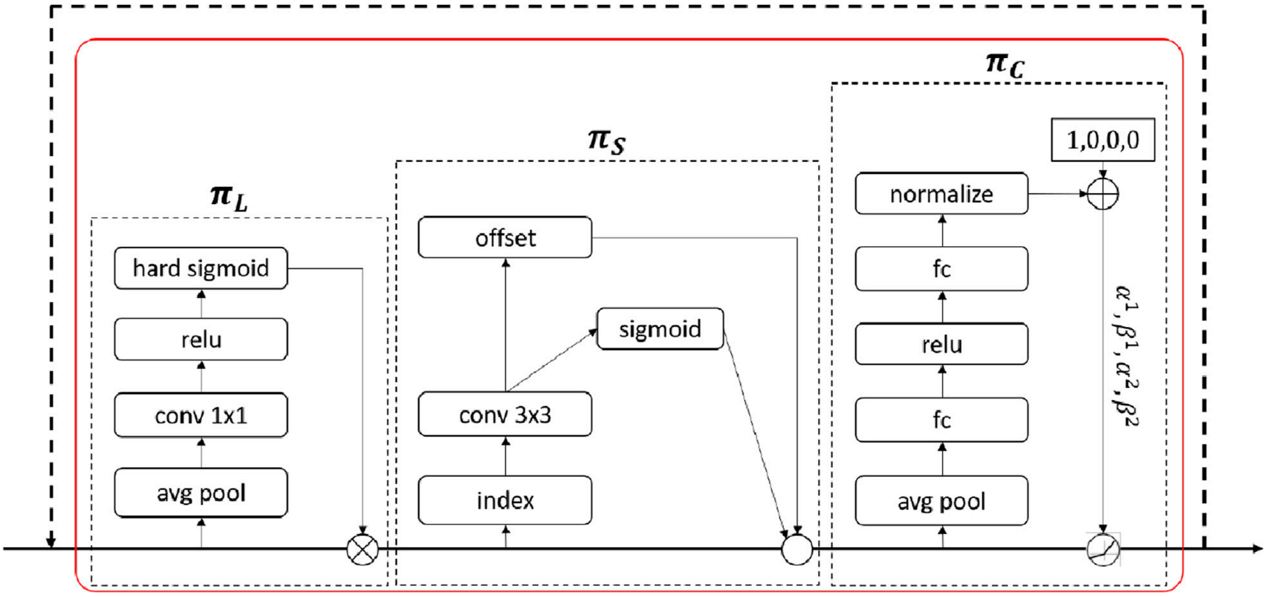

FIGURE 3
Structure of the DyHead module.

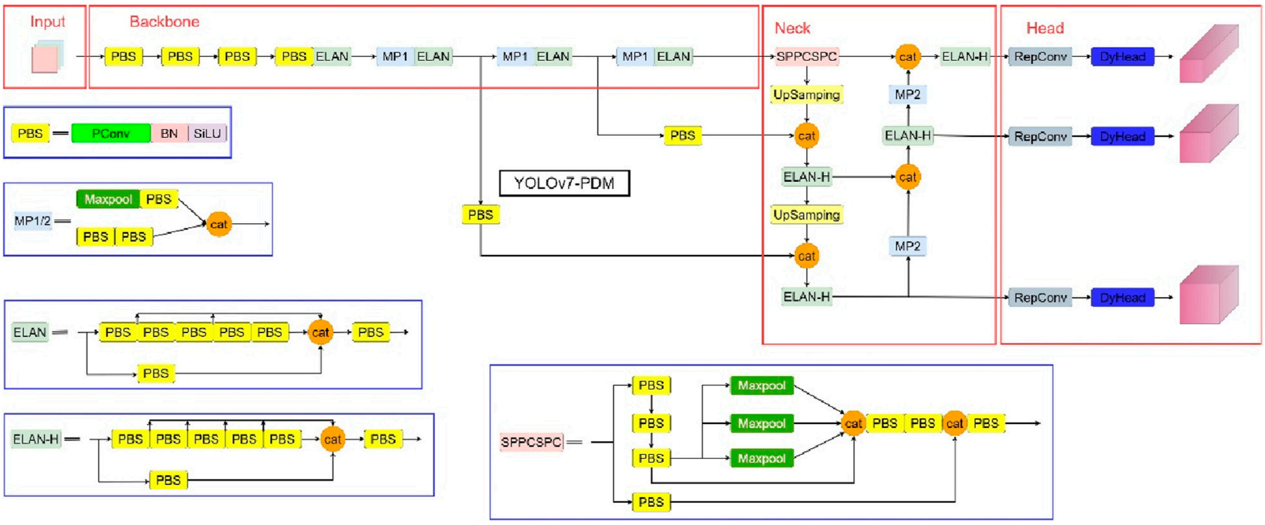

FIGURE 4
Model structure diagram of YOLOv7-PDM.

scale awareness between different feature levels, the π_S attention mechanism enables spatial awareness between spatial positions and the π_C attention mechanism promotes task awareness within the output channels. These π_L, π_S, and π_C attention mechanisms are combined to form the DyHead dynamic detection head module, as illustrated in Figure 3. By introducing the DyHead dynamic detection head module, we effectively utilize attention mechanisms to improve the performance and accuracy of object detection. The novelty of this method lies in applying attention mechanisms to different levels of perception and achieving a unified object detection head network through modular design.

The general form of self-attention is presented by Eq. 3.

$$W(F) = \pi(F) \cdot F, \tag{3}$$

This form has many parameters and directly learns the attention function through fully connected layers across all dimensions. In order to enhance efficiency and reduce the number of parameters, we transformed this attention function into three separate attentions, each concentrating on a specific dimension, as presented in Eq. 4.

$$W(F) = \pi_C(\pi_S(\pi_L(F) \cdot F) \cdot F) \cdot F, \tag{4}$$

where π_L combines the characteristics from various scales while considering their semantic significance, π_S focuses on the discriminative capacity between different spatial positions and π_C

promotes joint learning and generalizability of target representation by dynamically switching feature channels to assist different tasks. Stacking the DyHead dynamic detection head module multiple times yields better performance improvement, which peaks after stacking more than six modules. By introducing the DyHead dynamic detection head module, the expressiveness of the YOLOv7 algorithm's YOLO-Head is significantly enhanced without substantially increasing the computational complexity.

2.4 MPDioU bounding box loss function

As an improvement, we introduce the MPDIoU bounding box loss function [19] to address the instability in expressing the aspect ratio penalty of the CIoU loss function when the aspect ratio of the predicted bounding box matches that of the ground truth bounding box in the original YOLOv7 algorithm. The latter bounding box initially utilizes the CIoU loss function for bounding box regression. The proposed MPDioU bounding box loss function, which relies on the minimum point distance, assesses the similarity between predicted and ground truth bounding boxes, acting as a criterion for comparison. It should be noted that the YOLOv7 algorithm's convergence speed and detection accuracy are constrained because the CIoU and EIoU lose their effectiveness when the predicted and ground truth bounding boxes have varying width and height values but the same aspect ratio. This issue is overcome by combining the benefits of CIoU and EIoU. Besides, MPDIoU takes inspiration from the geometric characteristics of bounding boxes by directly minimizing the distances between the top left and bottom right points of the predicted and ground truth bounding boxes. The specific implementation is presented in Eq. 5.

$$MPDioU = \frac{A \cap B}{A \cup B} - \frac{d_1^2}{w^2 + h^2} - \frac{d_2^2}{w^2 + h^2}, \tag{5}$$

where A and B are two bounding boxes, d_1 is the distance between their top left points, d_2 is the distance between their bottom right points, and w and h represent the width and height of the input image. This design simplifies the similarity comparison between two bounding boxes and applies to overlapping and non-overlapping bounding box regression. Consequently, by leveraging the benefits of the MPDIoU bounding box loss function, the accuracy of the YOLOv7 algorithm in detecting objects is enhanced.

2.5 YOLOv7-PDM algorithm model

Figure 4 overviews the structure diagram of the YOLOv7-PDM algorithm, which has been optimized by incorporating the PConv module, DyHead dynamic detection head, and MPDioU bounding box loss function.

3 Experiment

3.1 Dataset creation

Due to a shortage of publicly accessible datasets for regular script characters in calligraphy, this research meticulously compiled an exclusive dataset by utilizing genuine works from Shen, a renowned calligrapher and member of the China Calligraphers Association, provided by the Sanpin Art Gallery in Shenzhen City. Regular script characters in calligraphic works typically exhibit single color and high contrast characteristics, with most presenting a consistent and neat writing style. Therefore, the works were first scanned using a line-scan camera in the data preprocessing stage. Subsequently, traditional binarization techniques effectively separated the acquired images into foreground and background. Additionally, we obtained individual regular script characters by batch cropping, utilizing fixed spacing between the characters. As a result, 2,782 black-and-white image samples of regular script characters were obtained, as depicted in Figure 5.

Following the research on Chinese digital calligraphy retrieval and authenticity identification by Zhang et al. [20], the stroke features of regular script characters were categorized into three basic features: start (qi), turn (zhuan), and end (shou). The start and end features were further divided into horizontal start (hengqi), vertical start (shuqi), horizontal end (hengshou), and vertical end (shushou). The turning feature was classified into a right-angle turn (zhijiaoze) and an acute-angle turn (ruijiaoze). Therefore, six-stroke features were extracted from regular script characters. After obtaining the images of regular script characters, we annotated them using the DLtools (MVTec Deep Learning Tool) annotation software. The annotation process requires careful alignment with every feature of regular script calligraphy characters. Besides, the selection of feature boxes should be neither too large nor too small, and it is necessary to conduct repeated inspections to ensure the absence of missed annotations, as omitting a single feature could potentially impact the accuracy of subsequent model training. The specific annotation quality is illustrated in Figure 6, representing a favorable annotation standard. Among them are 5,343 characters with a horizontal starting stroke, 4,545 with a horizontal ending stroke, 7,542 with a vertical starting stroke, 3,991 with a vertical ending stroke, 1,658 with right-angle turns, and 3,074 with acute-angle turns. The number of characters with right-angle and acute-angle turns is small, as not every character contains these types of turns.

Each calligrapher's characters exhibit a unique style, with the most distinctive characteristics being evident in the three fundamental aspects of "start," "turn," and "end," Where "start" refers to the starting point of the stroke, signifying the moment the brush touches the paper. The pressure and angle of initiation vary among calligraphers. Additionally, "turn" involves the rotation of the brush, with some characters requiring a subtle adjustment while others may demand a more pronounced rotation. Finally, "end" marks the stroke's conclusion, representing the character's completion. Some calligraphers execute the termination process, while others incorporate personal stylistic elements to showcase individuality. Therefore, using these six brushstroke features can effectively encapsulate the unique stylistic characteristics of a calligrapher's regular script.

In order to effectively mitigate the overfitting phenomenon during the algorithm model training process, this study employed data augmentation techniques on the 2,782 black-and-white images of individual regular script characters, including spatial transformation methods and noise addition methods, to expand

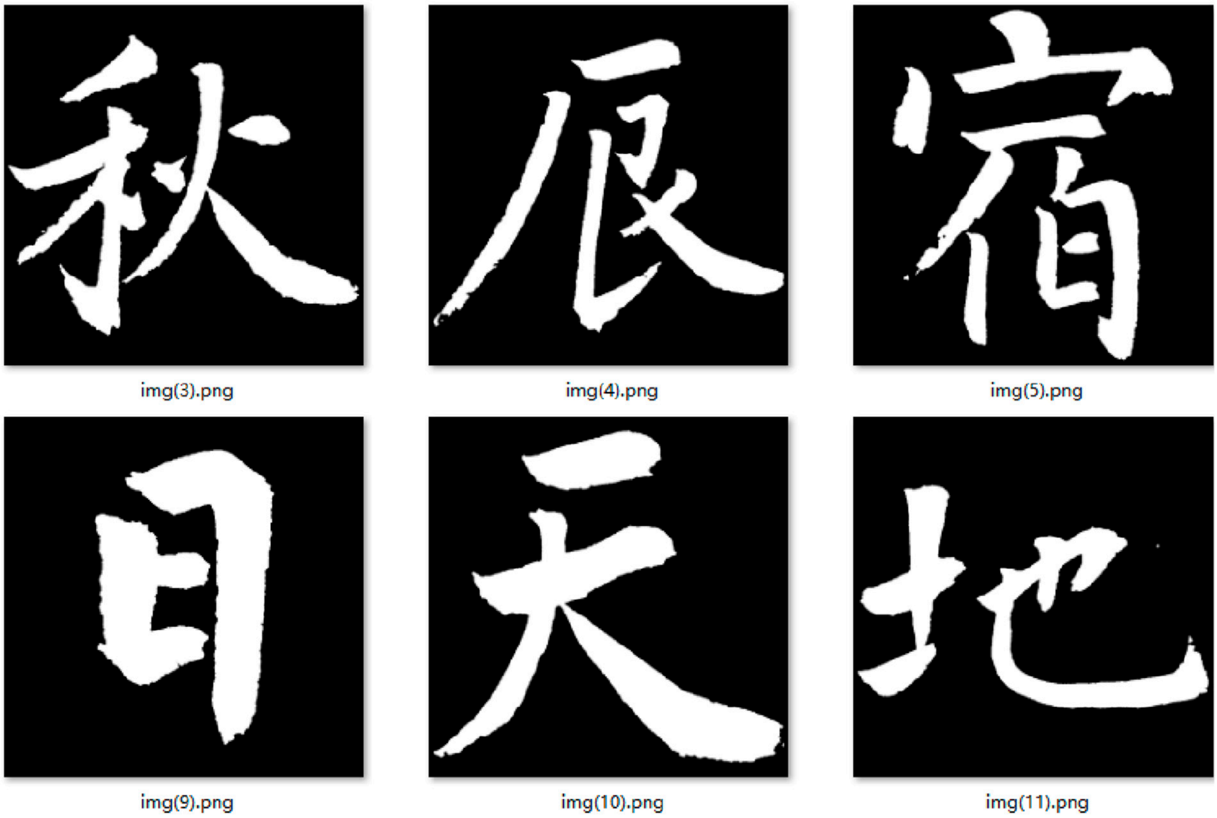

FIGURE 5
Segmented grayscale images of regular script characters.

the dataset. Through these methods, the original images of individual regular script characters were augmented to a total of 5,687 images, significantly enlarging the dataset. Data augmentation not only significantly enhanced the generalization capability of the algorithm model but also optimized the training performance of the model. Concurrently, following the format required by the YOLOv7 algorithm for training datasets, this study meticulously constructed the dataset of individual regular script characters for calligraphy. In order to ensure the scientific and practical validity of the dataset, we rigorously divided the dataset into training, validation, and testing sets in a 7:2:1 ratio to guarantee the reliability and effectiveness of model evaluation. Through the comprehensive implementation of the steps mentioned above, the construction of the dataset of individual regular script characters for calligraphy has been completed, providing a solid data foundation for the subsequent training and evaluation of algorithm models.

3.2 Experimental setup

The experimental setup for this research involved an Intel i9-13900K CPU, 128 GB of RAM, and two NVIDIA RTX4090 GPU cards with 24 GB of VRAM each. We set up the appropriate operating system (Ubuntu 20.04), Python 3.9, CUDA 11.8, PyTorch 2.0.0, and related dependencies on the training machine to conduct training and simulation experiments. By utilizing such hardware configuration and software environment, we ensured the smooth progress of the experiments and obtained accurate and reliable results. Furthermore, these configurations provided sufficient computational resources and performance to support the training and evaluation.

3.3 Training parameters and evaluation metrics

Before training the model, it is necessary to set the evaluation metrics and initialize the training parameters. This study employed four metrics to evaluate the model's performance: Mean Average Precision (mAP) with an IoU of 0.5, detection speed per image, parameter quantity, and computational complexity (FLOPs). The evaluation metrics were selected based on a comprehensive algorithm performance and efficiency consideration. Specifically, evaluating the accuracy and precision of the object detection algorithm relies on using mAP with an IoU of 0.5, while the detection speed per image measures the algorithm's efficiency in processing. Additionally, the complexity and computational requirements of the model are indicated by the parameter quantity and computational complexity (FLOPs). The selection of these four evaluation metrics is based on the fact that the authenticity detection of regular script characters only pursues

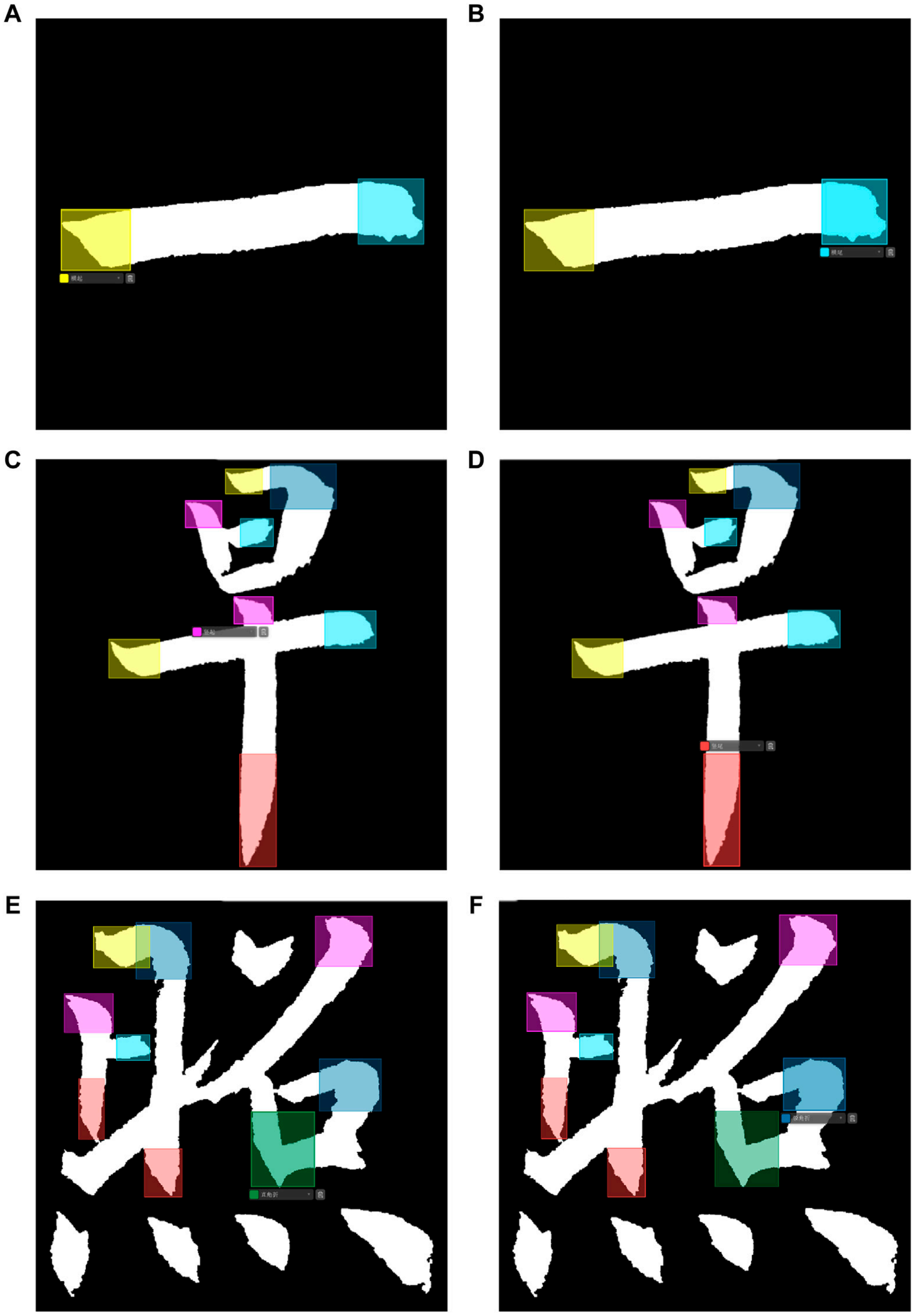

FIGURE 6
Six calligraphic character stroke types with their characteristic classification diagram highlighted: **(A)** horizontal start (yellow); **(B)** horizontal end (light blue); **(C)** vertical start (purple); **(D)** vertical end (red); **(E)** right angle turn (green); **(F)** acute angle turn (blue).

detection accuracy. Thus, the choice of these four evaluation metrics already satisfies the requirements.

Table 1 reports the precise configurations used to initialize the training parameters. Setting the hyperparameters is an important task impacting the model's performance and effectiveness. Moreover, ensuring hyperparameter setting consistency is crucial for enhancing the YOLOv7 algorithm model. On the one hand, preserving consistency in hyperparameter settings ensures effective algorithmic improvements while maintaining consistent hyperparameters, allowing for accurate evaluation of the

TABLE 1 Training parameters for the algorithm model.

Parameter	Value
Initial Learning Rate(Init_Lr)	0.02
Minimum Learning Rate(Min_Lr)	0.0002
Total Training Epochs (Total_Epochs)	1,000
Learning Rate Decay Type (Lr_Decay_Type)	cos
Batch Size of Each Training (Batch_Size)	48
Optimizer Type of Network Architecture (Optimzer_Type)	SGD
Momentum of Optimization Function (Momentum)	0.937
Weight Decay Coefficient (Weight_Decay)	0.0002

enhancements' effectiveness by comparing the algorithm's performance before and after the improvements. However, it is challenging to differentiate between the improvement effect of the algorithm itself and the performance changes caused by modifications to the hyperparameters when adjustments are made to the hyperparameters during the improvement process. Hence, the proposed method adopts the hyperparameters of YOLOv7.

In order to prevent overfitting of the regular script character dataset during the YOLOv7 algorithm training process, we measured the loss values of both the validation and training sets. After analysis, we found that the training set had a loss value (Loss) of 0.03, while the validation set had a loss value of 0.026, resulting in a minimal difference of only 0.004. This small difference suggests the absence of overfitting.

4 Experimental results

4.1 YOLOv7 algorithm with PConv module

We enhanced the Backbone and Neck sections of the YOLOv7 algorithm while considering the attributes of PConv. Specifically, the convolutions with a kernel size of 3 × 3 in the three feature output layers were replaced with PBS modules. The modified algorithm models in the SPPCSPC module, ELAN-H module, and the improved MP2 module in the Neck were labeled as YOLOv7-P, respectively. In order to guarantee the reliability of the experiments, this study conducted no less than 10 repeated experimental verifications on the YOLOv7 algorithm and YOLOv7-P algorithm on the proposed dataset. Table 2 reports the experimental results obtained by calculating the average of the experimental values when excluding the best and worst outcomes.

The experimental results in Table 2 highlight that the YOLOv7-P algorithm demonstrated a performance increase of almost 2.5% in mAP0.5 compared to the YOLOv7 algorithm. Additionally, the YOLOv7-P algorithm reduced the parameter quantity by 4.5M and FLOPs by one-fifth. Moreover, the single detection time remained almost unchanged between the two algorithms. By incorporating the PConv module into the YOLOv7 algorithm, the experimental results present enhanced detection accuracy and reduce the model's parameter quantity and computational complexity. This demonstrates the positive impact of the PConv module in the YOLOv7 algorithm without affecting the single detection time.

4.2 YOLOv7 algorithm with DyHead dynamic detection head

In order to evaluate the performance of integrating the DyHead dynamic detection head into the YOLOv7 algorithm (referred to as YOLO-Head) and to determine the optimal number of layers to embed the DyHead module, this study conducted no less than 10 repeated experimental verifications on the YOLOv7 algorithm and YOLOv7-D algorithm. To guarantee the reliability of the experiments, the experimental results were obtained by excluding the best and worst outcomes and averaging the remaining values. The detailed experimental results are presented in Table 3.

Table 3 infers that including four DyHead modules in the YOLOv7-D algorithm results in a performance enhancement of around 3.1% in mAP0.5 compared to the YOLOv7 algorithm. Additionally, the parameter quantity of the YOLOv7-D algorithm increases by 13M, while the FLOPs computational load shows a slight decrease. Furthermore, the detection time per image remains almost unchanged between the two algorithms. These experimental results demonstrate that although including the DyHead dynamic detection head in the YOLO-Head of the YOLOv7 algorithm leads to a relatively significant increase in parameter quantity, the FLOPs' computational load and detection time per image experience have insignificant changes. Moreover, the YOLOv7-D algorithm exhibits certain improvements in detection accuracy compared to the YOLOv7 algorithm. Thus, these findings substantiate the efficacy of integrating the DyHead dynamic detection head with the YOLOv7 algorithm.

4.3 YOLOv7 algorithm with MPDioU boundary box loss function

To assess the impact of replacing the CIoU bounding box loss function with the MPDioU bounding box loss function on the YOLOv7 algorithm's performance, we compared the training

TABLE 2 Experimental verification of PConv module.

Algorithm	mAP0.5 (%)	Parameter quantity(M)	FLOPs(G)	Detection time per Image (ms)
YOLOv7	90.19	36.50	105.20	3.1
YOLOv7-P	92.53	32.00	82.96	3.2

TABLE 3 Experimental verification of DyHead dynamic detection head.

Algorithm	Number of DyHead	mAP0.5 (%)	Parameter quantity(M)	FLOPs(G)	Detection time per Image (ms)
YOLOv7	\	90.19	36.50	105.20	3.1
YOLOv7-D	1	90.88	39.27	84.73	3.1
	2	91.72	42.44	91.19	3.1
	3	92.53	46.03	97.65	3.2
	4	93.21	49.51	104.56	3.3
	5	93.03	54.14	110.68	3.4
	6	92.88	57.82	118.92	3.4

TABLE 4 Experimental verification of MPDioU boundary box loss function.

Algorithm	mAP0.5 (%)	Parameter quantity(M)	FLOPs(G)	Detection time per Image (ms)
YOLOv7	90.19	36.50	105.20	3.1
YOLOv7-M	92.85	37.22	105.20	3.4

losses of both the regular YOLOv7 and the modified YOLOv7-M algorithms. The results reveal that the training loss of the YOLOv7-M algorithm is 0.02, while the training loss of the YOLOv7 algorithm is 0.03. This indicates that the MPDioU bounding box loss function is superior to the CIoU bounding box loss function. Furthermore, to ensure the validity of the experiments, we conducted no less than 10 repeated experiments on the YOLOv7 algorithm and the YOLOv7-M algorithm using the developed dataset. We calculated the average of the remaining experimental values after excluding the best and worst results to obtain the experimental results, with Table 4 presenting the experimental results.

Table 4 reveals that the YOLOv7-M algorithm achieves a boost of approximately 2.7% in mAP0.5 compared to the YOLOv7 algorithm. Additionally, the parameter quantity of the YOLOv7-M algorithm increases by nearly 1M, but there is no change in the FLOPs computational complexity, while the single detection time slightly increased. Considering these results, the YOLOv7-M algorithm model has higher detection accuracy under almost unchanged FLOPs computational complexity and single detection time. These results prove that the MPDioU boundary box loss function significantly enhances the performance of the YOLOv7 algorithm model.

4.4 Overall experiment analysis

4.4.1 Ablation experiment

This paper proposes three improvement methods, namely, the PConv module (YOLOv7-P), the DyHead dynamic detection head (YOLOv7-D), and the MPDioU bounding box loss function (YOLOv7-M). To ascertain the efficacy and enhancements of these three methods, comparative experiments were carried out under the same experimental settings to evaluate the performance disparities between the YOLOv7 algorithm and the YOLOv7 algorithm equipped with one, two, and three enhancement methods. To guarantee the experiments' validity, we repeated each experiment 10 times and excluded the most extreme results. The remaining values from the experiments were averaged to obtain the experimental outcome, as presented in Table 5.

Based on the findings in Table 5, the YOLOv7-PDM algorithm exhibits a 4% enhancement in mAP0.5 compared to the YOLOv7 algorithm. Furthermore, the YOLOv7-PDM algorithm has nearly 5M fewer parameters and approximately 27G FLOPs while maintaining the same detection time for individual images. These results suggest that the YOLOv7-PDM algorithm model surpasses the YOLOv7 algorithm model, considering operational and spatial complexity. Besides, the YOLOv7-PDM algorithm model, which integrates three enhancement methods, exhibits the highest performance, as it enhances detection accuracy (mAP0.5 improvement) and significantly reduces the parameter count and computational workload without impacting the time required for single-image detection.

4.4.2 Comparison with other mainstream object detection models

In order to assess the effectiveness of the YOLOv7-PDM algorithm, we carried out comparative experiments involving eight popular detection models: YOLOv7, YOLOv6 [21], YOLOv8 [22], Deformable-DETR [23], RT-DETR [24], Faster-RCNN [25], SSD [26], and DETR [27] under the same experimental configuration. To ascertain the experiment's validity, a minimum of 10 repetitions of experimental training and validation were conducted on all data results. The optimal and worst outcomes were disregarded, and the remaining experimental values were averaged to derive the final result. The experimental results are reported in Table 6.

TABLE 5 Ablation experiment comparison of three improvement methods.

Algorithm	mAP0.5 (%)	Parameter quantity(M)	FLOPs(G)	Detection time per Image (ms)
YOLOv7	90.19	36.50	105.20	3.1
YOLOv7-P	92.53	32.00	82.96	3.2
YOLOv7-D	93.21	49.51	105.22	3.3
YOLOv7-M	92.85	37.22	105.20	3.4
YOLOv7-PM	92.78	32.00	82.96	2.4
YOLOv7-PD	93.62	31.67	78.20	4.3
YOLOv7-DM	93.47	36.18	98.47	3.6
YOLOv7-PDM	94.19	31.67	78.20	3.1

TABLE 6 Performance comparison of nine detection models.

Algorithm	mAP0.5 (%)	Parameter quantity(M)	FLOPs(G)	Detection time per Image (ms)
YOLOv7	90.19	36.50	105.20	3.1
YOLOv6	89.87	34.80	85.64	3.4
YOLOv8	89.72	25.80	78.70	4.0
DETR	88.37	36.74	223.62	35.7
Deformable-DETR	87.98	39.83	157.35	36.2
RT-DETR	89.89	32.00	110.53	13.3
Faster-RCNN	85.33	41.38	269.03	27.3
SSD	53.94	13.69	30.71	1.3
YOLOv7-PDM	94.19	31.67	78.20	3.1

According to the results in Table 6, the YOLOv7-PDM algorithm outperforms the other eight mainstream algorithms in terms of mAP0.5, achieving 94.19%. The YOLOv7-PDM performs better in detection accuracy, showing a nearly 41% improvement in mAP0.5 while having fewer parameters, computational FLOPs, and detection time per image than the SSD algorithm. This indicates a significant advantage for YOLOv7-PDM. Compared with the Faster-RCNN algorithm, the YOLOv7-PDM outperforms it in all aspects, including mAP0.5, the number of parameters, computational FLOPs, and detection time per image. Moreover, relative to other YOLO series models, the YOLOv7-PDM achieves the highest levels of performance in mAP0.5, FLOPs, and detection time per image, with a slight disadvantage in the number of parameters compared to YOLOv8 but superior to YOLOv6. Compared with the DETR series models, the YOLOv7-PDM performs better in mAP0.5, the number of parameters, computational FLOPs, and detection time per image, validating the superiority of the proposed YOLOv7-PDM. In the detection of regular script characters, detection accuracy is more critical. Compared to the YOLOv7 algorithm, the YOLOv7-PDM algorithm maintains the same single-image detection time but substantially improves detection accuracy, parameter quantity, and computational FLOPs. This further validates the superiority of the proposed YOLOv7-PDM algorithm model in this study. Figure 7 illustrates the detection results of the nine models.

Comparing the graphs in Figure 7 reveals that when the pen stroke feature is small, the eight algorithm models fail to detect it correctly. It should be noted that in this paper, the size of the target is defined as follows, taking the commonly used dataset COCO object definition in the field of object detection as an example: small targets refer to objects smaller than 32 × 32 pixels, medium targets refer to objects ranging from 32 × 32 to 96 × 96 pixels, and large targets refer to objects larger than 96 × 96 pixels. When a single character has many strokes, leading to smaller pen stroke features, the SSD algorithm model fails to detect it. When there is a partial overlap in the pen stroke features, the YOLOv6, Faster-RCNN, and DETR algorithm models fail to detect it accurately. On the other hand, the proposed YOLOv7-PDM algorithm model can accurately detect and recognize most of the pen stroke features, demonstrating superior performance in bounding box regression and higher confidence levels compared to the YOLOv7 algorithm model. This further proves that the YOLOv7-PDM algorithm model is the most suitable for detecting calligraphy Kai-style characters' pen stroke features.

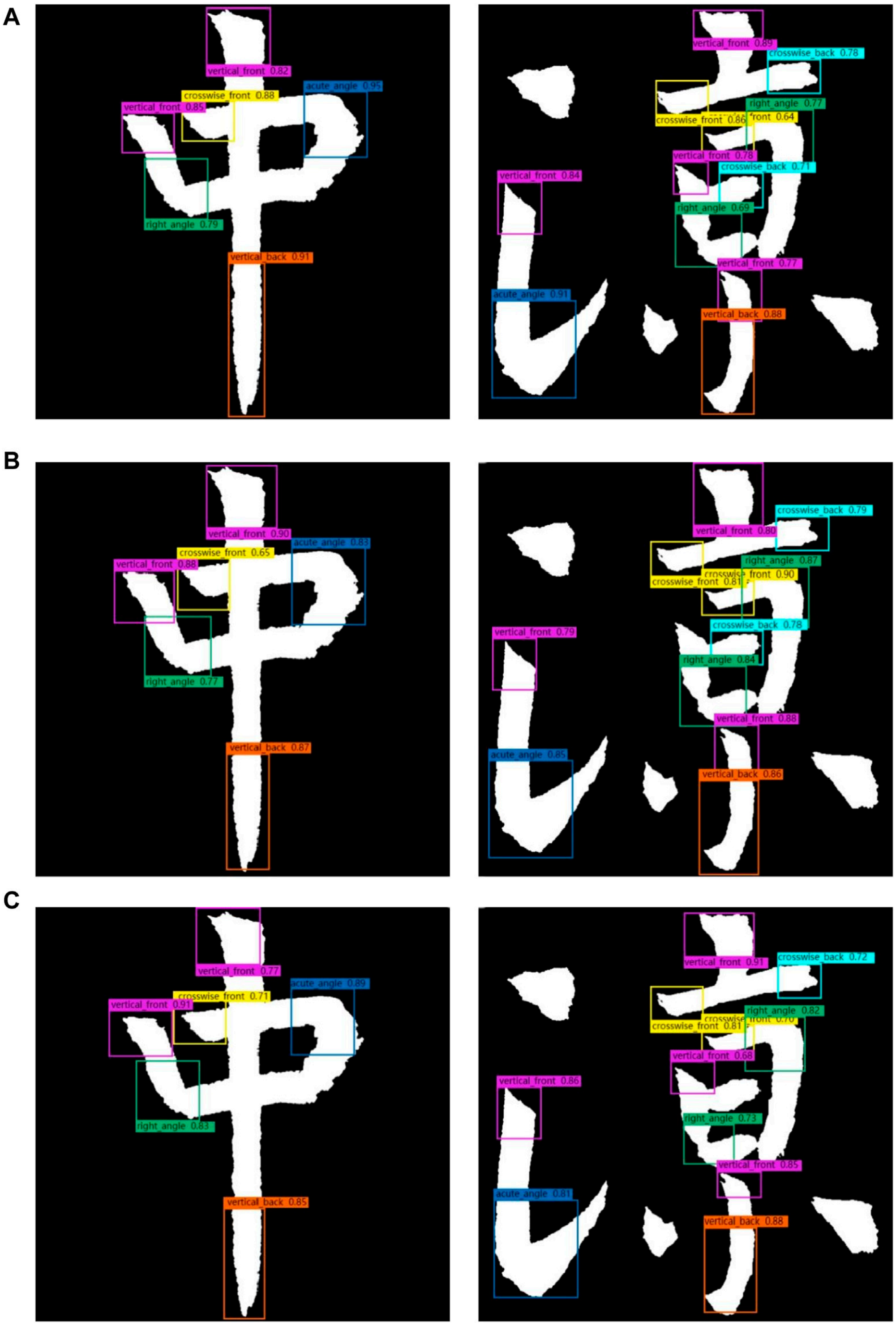

FIGURE 7
(Continued).

4.4.3 Test of replica calligraphy regular script works

To further confirm the effectiveness of the proposed method, tests were carried out using two genuine copies of regular script characters and their corresponding imitations by the same calligrapher. The testing procedure involved extracting individual characters from the two authentic copies and two imitations separately, following the method mentioned above of creating the dataset. As a result, four sets of character datasets were obtained for detection. Subsequently, the regular script pen-

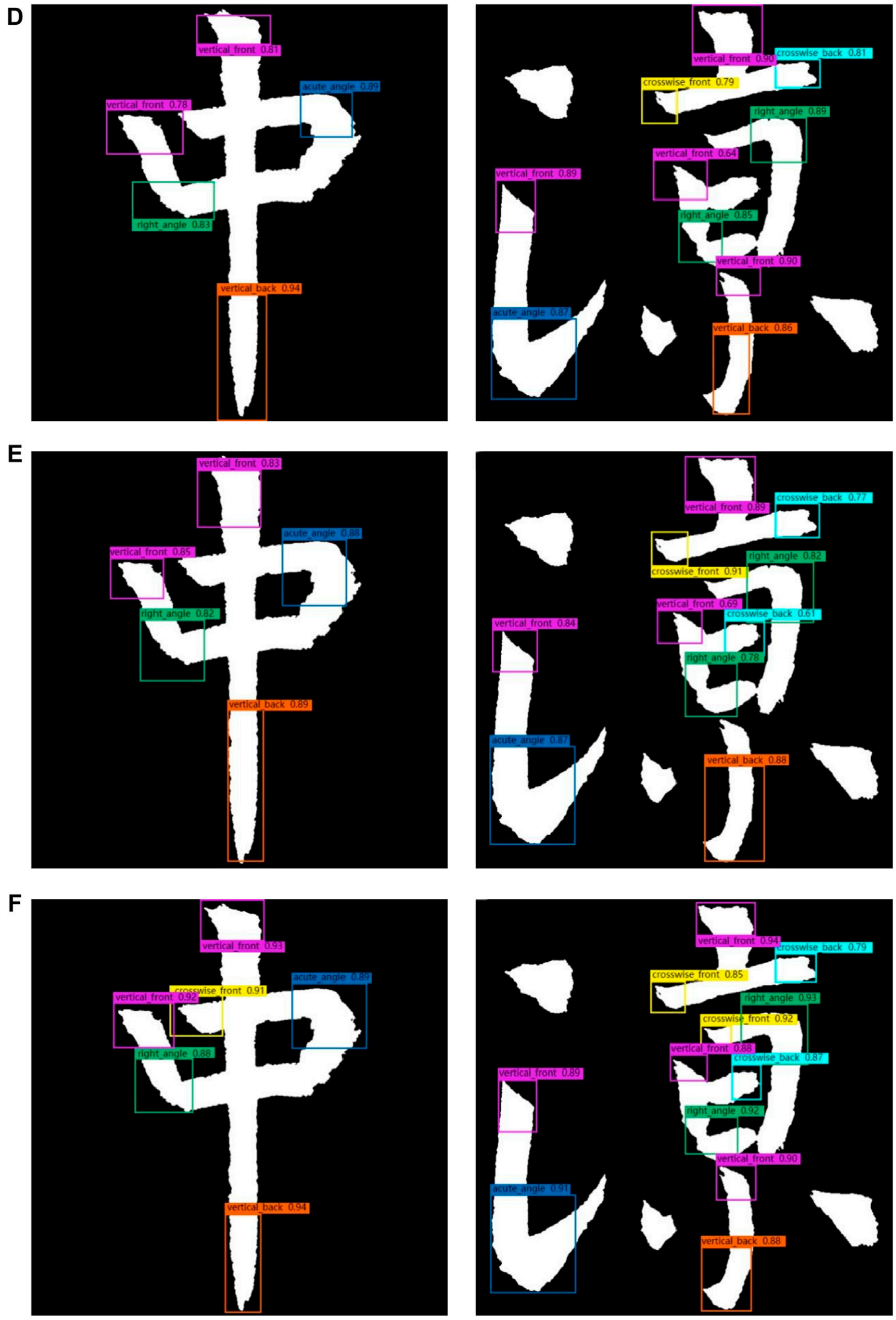

FIGURE 7
(Continued).

pressure feature detection was performed on each of the four sets of character datasets. Finally, the total number of pen-pressure features for each category of regular script characters in the four datasets was recorded, and the detection results are presented in Table 7.

Table 7 highlights a significant difference in the total stroke feature count of different categories of regular script characters detected using the YOLOv7-PDM algorithm for the authentic and imitation works of Shen in works one and 2. The total stroke feature count for each category in the two authentic works is generally above

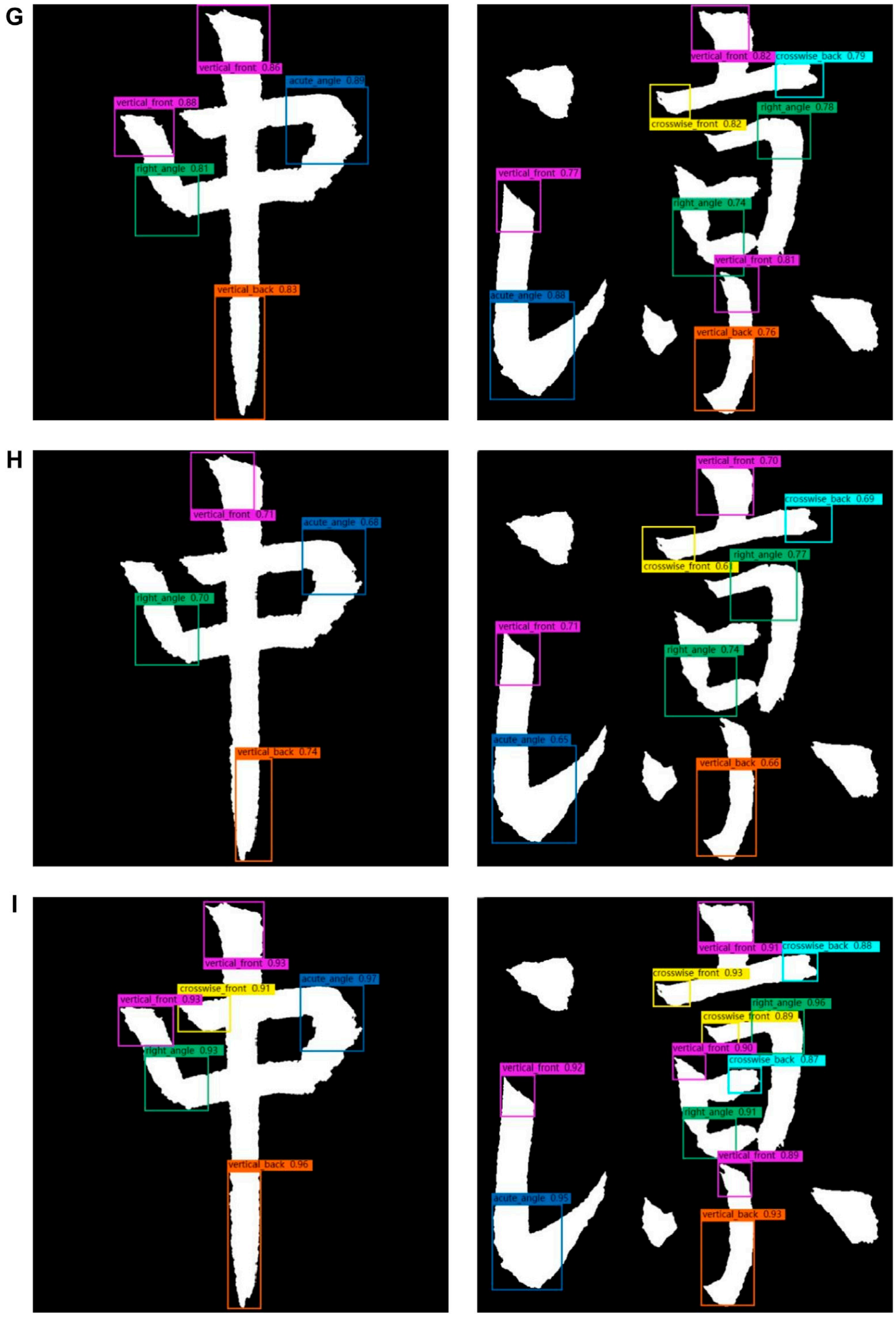

FIGURE 7
(Continued). Comparison of the effects of nine algorithm models: **(A)** YOLOv7; **(B)** YOLOv6; **(C)** YOLOv8; **(D)** DETR; **(E)** Deformable-DETR; **(F)** RT-DETR; **(G)** Faster-RCNN; **(H)** SSD; **(I)** YOLOv7-PDM.

200, while the total for each category in the two imitation works is below 25. This demonstrates that the developed method efficiently differentiates between genuine and counterfeit works of Shen's regular script characters. Moreover, this serves as additional evidence supporting the efficacy of the identification method introduced in this paper.

TABLE 7 Test results of Shen's regular script works identification.

Calligraphy	Work type	Horizontal start	Horizontal end	Vertical start	Vertical end	Straight angle	Fold angle
Works 1	Genuine	256	354	229	273	215	261
	Replica	11	18	10	13	19	20
Works 2	Genuine	266	314	329	203	315	191
	Replica	13	19	21	8	17	12

5 Conclusion

This paper presents an enhanced YOLOv7-PDM algorithm model for verifying regular calligraphy script works built upon the YOLOv7 algorithm. Specifically, to avoid the increased complexity of the improved YOLOv7 algorithm, we replaced the convolutional layers in the Backbone part with the PConv module. Reducing the model's parameter count and computational cost (FLOPs) enhanced the algorithm's mAP0.5 and maintained the same single-image detection time. Furthermore, the DyHead dynamic detection head was introduced to enhance the detection accuracy of the YOLOv7 algorithm as much as possible. This improvement increased the algorithm's recognition accuracy without affecting the inference speed. Additionally, to improve the regression capability of the bounding boxes in the YOLOv7 algorithm, we incorporated the MPDioU bounding box loss function. By further improving the overall mAP0.5 value, a recognition accuracy of 94.19% was achieved. By comparing the YOLOv7-PDM algorithm model with eight mainstream algorithms including YOLOv7, YOLOv6, YOLOv8, Deformable-DETR, RT-DETR, Faster-RCNN, SSD, and DETR, we demonstrated that the YOLOv7-PDM algorithm achieved the best performance in terms of mAP0.5 and single-image detection time, accomplishing the improvement goals of the algorithm.

When applying the YOLOv7-PDM algorithm to the authentication of calligraphy regular script works, the genuine works and replicas can be distinguished by comparing the detected feature quantities. Nevertheless, there is scope for enhancing our algorithm as we overlooked special cases like overlapping and intersecting characters in the later stages of calligraphy cursive script works, which directly impacted the accuracy of the model's detection. In upcoming studies, our main goal will be to refine the algorithm and enhance the model's resilience.

Data availability statement

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

Author contributions

JC: Conceptualization, Data curation, Investigation, Methodology, Writing–original draft. ZH: Conceptualization, Data curation, Investigation, Methodology, Writing–original draft. XJ: Conceptualization, Data curation, Formal Analysis, Methodology, Validation, Writing–original draft. HY: Funding acquisition, Project administration, Supervision, Validation, Writing–review and editing. WW: Formal Analysis, Funding acquisition, Project administration, Supervision, Writing–review and editing. JW: Funding acquisition, Investigation, Resources, Supervision, Visualization, Writing–review and editing. XW: Formal Analysis, Resources, Software, Validation, Visualization, Writing–review and editing. ZX: Project administration, Resources, Software, Visualization, Writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This research was funded in part by the National Key Research and Development Project of China (grant number: 2018YFA0902900), the Basic Research Program of Guangzhou City of China (grant number 202201011692), and the Guangdong Water Conservancy Science and Technology Innovation Project (grant number 2023-03).

Acknowledgments

The authors would like to express their thanks to the Guangzhou Institute of Advanced Technology for helping them with the experimental characterization. We want to express our gratitude to the Shenzhen Sanpin Art Museum for providing the calligraphy artwork data.

Conflict of interest

Author WW was employed by Shenzhen Cas Derui Intelligent Technology Co, Ltd.

The remaining 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.

References

1. Yuan BI. Calligraphy. *Calligraphy and Painting Art* (2023) 2023(06):16–29+98.

2. Chen H, Xu B. *Chen Hui and Xu Bangda's calligraphy and painting appraisal research master's thesis*. Qufu: Qufu Normal University (2023).

3. Zhao R. *Research on the authenticity identification of Chinese calligraphy based on style characteristics master's thesis*. Xi'an: Xi'an University of Architecture and Technology (2016).

4. Zhang Y. On the identification of dong qichang's calligraphy——taking the cultural relics in the collection of shandong Museum as an example. *Artwork* (2021) 2021(06):8–15.

5. Lu Q, Liu C. Discussion on the application of handwriting theory to calligraphy forensics. *Leg Syst Soc* (2021) 2021(05):163–5. doi:10.19387/j.cnki.1009-0592.2021.02.166

6. Zeng B. *Research on the authenticity identification method of Chinese calligraphy based on image recognition, master's thesis*. Xi'an: Xi'an University of Architecture and Technology (2015).

7. Chen SC. *Computer-aided authentication of Chinese calligraphy*. Xi'an: Xi'an University of Architecture and Technology (2017).

8. Xiao L. *Evaluation and detection of calligraphy copying characters based on deep learning*. Shanghai: East China University of Science and Technology (2023).

9. Wang H, Huang Y, Cai B. Comparison of image similarity algorithms based on traditional methods and deep learning methods. *Comp Syst Appl* (2024) 33(02):253–64. doi:10.15888/j.cnki.csa.009413

10. Wang W, Jiang X, Yuan H, Chen J, Wang X, Huang Z Research on algorithm for authenticating the authenticity of calligraphy works based on improved EfficientNet network. *Appl Sci* (2024) 14:295. doi:10.3390/app14010295

11. Xu X, Chen B, Wang J. Target detection algorithm for river and lake ships based on improved YOLOv4-Tiny. *Yangtze River* (2023) 54(09):264–71. doi:10.16232/j.cnki.1001-4179.2023.09.035

12. Hu J, Wang C, Yang C. Research on surface hole formation defect detection of aluminum castings based on improved YOLOX algorithm. *Special-cast and Non-ferrous Alloys* (2023) 43(09):1205–9. doi:10.15980/j.tzzz.2023.09.008

13. Mai G, Liang Y, Pan J. Calligraphy font recognition algorithm based on improved DenseNet network. *Comp Syst Appl* (2022) 31(02):253–9. doi:10.15888/j.cnki.csa.008326

14. Wang C-Y, Bochkovskiy A, Liao H-YM. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; June 18-22, 2023; Vancouver, Canada (2023). p. 7464–75.

15. Bochkovskiy A, Wang C-Y, Liao H-YM. Yolov4: optimal speed and accuracy of object detection (2020). Available from: https://arxiv.org/abs/2004.10934 (Accessed August 9, 2023).

16. Wang W, Chen J, Huang Z, Yuan H, Peng L, Jiang X, et al. Improved YOLOv7-based algorithm for detecting foreign objects on the roof of a subway vehicle. *Sensors* (2023) 23(23):9440. doi:10.3390/s23239440

17. Chen J, Kao S-hong, He H, Zhuo W, Wen S, Lee C-H, et al. Run, don't walk: chasing higher FLOPS for faster neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition; June 18-22, 2023; Vancouver, Canada (2023). p. 12021–31.

18. Dai X, Chen Y, Xiao B, Chen D, Liu M, Lu Y, et al. Dynamic head: unifying object detection heads with attentions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; Jun 19-25, 2021; Virtual (2021). p. 7373–82.

19. Siliang M, Yong X. MPDIoU: a loss for efficient and accurate bounding box regression (2023). Available from: https://arxiv.org/abs/2307.07662 (Accessed September 12, 2023).

20. Zhang X A *study on the retrieval of Chinese digital calligraphy and the identification of the authenticity of works*. Hangzhou: Zhejiang University (2006). Master's thesis.

21. Li C, Li L, Jiang H, Weng K, Geng Y, Liang L, et al. YOLOv6: a single-stage object detection framework for industrial applications (2022). Available from: https://arxiv.org/abs/2209.02976 (Accessed August 13, 2023).

22. Oh G, Lim S. One-stage brake light status detection based on YOLOv8. *Sensors* (2023) 23(17):7436. doi:10.3390/s23177436

23. Zhu X, Su W, Lu L, Li B, Wang X, Dai J. Deformable detr: Deformable transformers for end-to-end object detection (2020). Available from: https://arxiv.org/abs/2010.04159 (Accessed June 8, 2023).

24. Lv W, Xu S, Zhao Y, Wang G, Wei J, Cui C, et al. Detrs beat yolos on real-time object detection (2023). Available from: https://arxiv.org/abs/2304.08069 (Accessed July 25, 2023).

25. Ren S, He K, Girshick R, Sun J. Faster r-cnn: towards real-time object detection with region proposal networks. In: Advances in neural information processing systems 28 IEEE Trans Pattern Anal Mach Intell; Dec 7-12, 2015; Montreal, Quebec, Canada, 39 (2015). p. 1137–49.

26. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, et al. Ssd: single shot multibox detector. In: Computer Vision–ECCV 2016: 14th European Conference, Proceedings, Part I 14; October 11–14, 2016; Amsterdam, The Netherlands. Springer International Publishing (2016). p. 21–37. Available from: https://arxiv.org/pdf/1512.02325.pdf. (Accessed July 25, 2023)

27. Carion N, Massa F, Synnaeve G, Usunier N, Kirillov A, Zagoruyko S. End-to-end object detection with transformers. In: European conference on computer vision; Aug 23-28, 2020; SEC, Glasgow (2020). p. 213–29. Available from: https://arxiv.org/pdf/2005.12872.pdf (Accessed July 26, 2023).