symmetry

MDPI

Article

Symmetry-Based Fusion Algorithm for Bone Age Detection with YOLOv5 and ResNet34

Wenshun Sheng *, Jiahui Shen, Qiming Huang, Zhixuan Liu, Jiayan Lin, Qi Zhu and Lan Zhou

Pujiang Institute, Nanjing Tech University, Nanjing 211200, China
* Correspondence: sws@njpji.edu.cn

Abstract: Bone age is the chronological age of human bones, which serves as a key indicator of the maturity of bone development and can more objectively reflect the extent of human growth and development. The prevalent viewpoint and research development direction now favor the employment of deep learning-based bone age detection algorithms to determine bone age. Although bone age detection accuracy has increased when compared to more established methods, more work needs to be conducted to raise it because bone age detection is primarily used in clinical medicine, forensic identification, and other critical and rigorous fields. Due to the symmetry of human hand bones, bone age detection can be performed on either the left hand or the right hand, and the results are the same. In other words, the bone age detection results of both hands are universal. In this regard, the left hand is chosen as the target of bone age detection in this paper. To accomplish this, the You Only Look Once-v5 (YOLOv5) and Residual Network-34 (ResNet34) integration techniques are combined in this paper to create an innovative bone age detection model (YARN), which is then combined with the RUS-CHN scoring method that applies to Chinese adolescent children to comprehensively assess bone age at multiple levels. In this study, the images in the hand bone dataset are first preprocessed with number enhancement, then YOLOv5 is used to train the hand bone dataset to identify and filter out the main 13 joints in the hand bone, and finally, ResNet34 is used to complete the classification of local joints and achieve the determination of the developmental level of the detected region, followed by the calculation of the bone age by combining with the RUS-CHN method. The bone age detection model based on YOLOv5 and ResNet34 can significantly improve the accuracy and efficiency of bone age detection, and the model has significant advantages in the deep feature extraction of key regions of hand bone joints, which can efficiently complete the task of bone age detection. This was discovered through experiments on the public dataset of Flying Paddle AI Studio.

Keywords: bone age detection; deep learning; YOLOv5; ResNet34; data augmentation

Citation: Sheng, W.; Shen, J.; Huang, Q.; Liu, Z.; Lin, J.; Zhu, Q.; Zhou, L. Symmetry-Based Fusion Algorithm for Bone Age Detection with YOLOv5 and ResNet34. *Symmetry* **2023**, *15*, 1377. https://doi.org/10.3390/sym15071377

Academic Editor: Zhixun Su

Received: 13 June 2023
Revised: 2 July 2023
Accepted: 5 July 2023
Published: 6 July 2023

1. Introduction

The degree of human growth and development can be more accurately reflected by bone age since the development of the human skeleton occurs in phases and is continuous. Additionally, the physical properties of the bones at different stages vary. This is why forensics, sports, and clinical medicine all frequently employ bone age. By taking X-rays of the hand and wrist and then having the doctor review and interpret them, bone age is traditionally determined. This process is time-consuming and subject to the doctor's subjective judgment. This work conducts bone age detection on the left-hand bone based on the symmetry of the human hand bone, considerably reducing the time of bone age detection. Additionally, significant progress has been made with the usage of convolutional neural networks as artificial intelligence and deep learning in connection with this study. There is still room for improvement. However, the final bone age data were separated and evaluated using a computer, considerably improving the bone age detection accuracy.

TW [1] is one of the latest international detection methods, which is proved by the example that it does not apply to human bone age detection in China. YOLOv5 [2] is a single-stage target detection algorithm for target identification and localization, which is used to identify the major joints of hand bones in this paper. ResNet34 [3] is the 34th layer of the ResNet network, which is a deep feed-forward neural network for completing information transfer and circulation and is used to achieve the classification of minor joints of hand bones in this paper. In this paper, we propose a bone age detection framework (YARN) combining YOLOv5 and ResNet34, and we choose the RUS-CHN scoring method [4], which is more suitable for the physiology of domestic adolescent children, as the evaluation standard for bone age detection.

The "left-hand bone age X-ray" and "left-hand small joint X-ray classification" datasets from Flying Paddle AI Studio [5] are used in this study as the research object because of the symmetry of human bones. After preprocessing the left-hand bone data, YOLOv5 was trained to detect 21 hand-bone joints and filter out 13 significant hand-bone joints. Following the classification of the joints using the enhanced ResNet34 network, the RUS-CHN approach was used to determine bone age.

The rest of this paper is organized as follows: Section 2 introduces the related work; Section 3 describes the design and implementation of the model; Section 4 describes how to train and detect the data using both YOLOv5 and ResNet34 algorithms; Section 5 concludes this study with outlooks for the future and also points out the shortcomings and areas for improvement in this paper.

2. Related Works

Since deep learning and bone age detection were combined, many scholars have been exploring the value of deep learning in the field of bone age detection, and all of them have achieved good results.

Davis et al. [6] proposed a model combining image pre-processing and feature extraction algorithms to automate bone age detection by building a predictive model, i.e., estimating bone age directly from radiographs without the need for a manual procedure. The work achieved a leap from manual to automated, greatly saving time in bone age detection and improving the accuracy of bone age detection.

In Ref. [7], Lee et al. proposed a fully automated bone age detection model based on deep learning. The model incorporates ImageNet and a fine-tuned CNN with a fully automated deep learning pipeline to segment the region of interest, normalize and preprocess the X-ray images, and finally, in the test, obtained an accuracy of 57.32 and 61.40% for the female and male groups, respectively. However, the experimental results are still somewhat different from the actual values due to the Greulich–Pyle (GP) atlas's choice to provide time points only every 1 year and the lack of richness in the dataset.

Zhan et al. [8] proposed an improved structural model of the AlexNet network to detect bone age, which improves the recognition rate of the network for hand bone image features by expanding the size of the original image and the resolution of the image during training. In addition, the rectified linear unit (ReLU) in the activation layer of the original network was replaced with a parametric rectified linear unit (PreLU) to improve the fitting ability of the model. Finally, the mean absolute error (MAE) between the predicted and actual ages through an experiment for females and males was 0.72 and 0.90 years, respectively, but there was still a degree of error in the experimental results because of the small sample size.

In Ref. [9], Ari et al. proposed a region-based feature connectivity layer (RB-FCL) deep learning model that uses Faster R-CNN to automatically segment important parts of skeletal radiographs and sequentially selected DenseNet121, InceptionV3, and Inception-ResNetV2 for the training of the critical regions. According to the evaluation results, the experiments produced an MAE result of 6.97, which is much better than the standard deep learning model.

Zhang et al. [10] proposed an improved Xception regression network by introducing deep separation convolution and residual connectivity, embedding a convolutional block attention module (CBAM) in the Xception network to enhance attention in channel and spatial supervision, extracting important features required for bone age assessment while using gender as a distinction and detecting men and women separately, which resulted in a reduction in MAE of 3.23 months.

In Ref. [11], Wang et al. used deep CNN for feature learning, fused low-level and high-level features of hand bone images, and removed the Softmax layer in the Inception ResNet V2 network to optimize the Inception ResNet V2 network structure. They then compared with the bone age detection method using the BoNet network, and the MAE between predicted and actual bone age was reduced by 0.4230 years.

Ding et al. [12] proposed a new bone age detection model different from the traditional bilinear convolutional neural network, applied the fine-grained image recognition method to bone age image recognition, integrated the attention mechanism and residual module in each feature extraction sub-network, and used Resenet-50 to replace the visual geometry group (VGG) network in the original model, which effectively improved the accuracy of bone age assessment. The experimental results show that the MAE between the assessed bone age and the true bone age is 0.57 years in the bone age stage from 9.0 to 13.0 years, but due to the uneven distribution of samples in the dataset across age groups, the bone age detection effect still needs to be improved.

In Ref. [13], Lee et al. proposed a deep learning-based bone age detection model consisting of three steps: ROI detection, regional maturity classification, and integrated bone age. First, the model automatically detects seven regions of the hand bone using a CNN algorithm. Then, the maturity of each ROI and the overall hand image is automatically classified using the CNN algorithm. Finally, the bone age results are objectively derived by combining the physician scores. Experiments showed that the error between the model and the reference standard was less than 0.5 years.

Mao et al. [14] added Harris features and a convolutional attention module to the AlexNet network to assess the developmental stage of each reference bone, and they added a normalization layer and ReLU activation function after the convolutional layer in the model, which greatly improved the training speed of the bone age detection network and, combined with the optimized CHN method, could eventually obtain an error within $\pm$0.5 years of 94.6% of accuracy.

The aforementioned experimental methods, to a certain extent, boost the precision and effectiveness of bone age detection, although they do not fully take into account the impact of the type of optimizer on the training results during model training. As a result, it is essential to train the common optimizers independently to achieve a more ideal bone age detection effect, and the experiments in this paper compare the training effects of two different optimizers, stochastic gradient descent (SGD) [15] and adaptive moment estimation (Adam) [16], before selecting an SGD optimizer for ResNet34 training a small joint dataset, which can produce training weight files that are both higher than 90%.

The main contributions of this work are as follows:

- Using the adaptable anchor frame in YOLOv5 makes it easier for the network to learn the features of hand bone images, improves the efficiency of image pre-processing work before training, and the selection of a suitable anchor frame also provides great help to improve the accuracy of subsequent image feature extraction.
- ResNet34 is used to train the small joint dataset, which ensures that the accuracy of each classification reaches more than 90%.
- A bone age detection framework based on YOLOv5 and ResNet34 combined with the RUS-CHN scoring method (YARN) was proposed and implemented to construct a targeted detection model for bone age detection in Chinese adolescent children.

3. Model Design and Implementation

To perform bone age detection through target detection and classification regression in the field of deep learning, a novel bone age detection model (YARN) combining YOLOv5 and ResNet34 is proposed in this paper. The RUS-CHN scoring method, which is appropriate for the physiology of Chinese adolescent children, is chosen as the evaluation criterion for bone age detection. The following are the design concepts:

(1) To improve the quality of the dataset used in model training and to speed up the target detection, the hand bone images will be subjected to contrast-limited adaptive histogram equalization defogging, rotation augmentation, and adaptive scaling expansion square operations as a way to enhance the data of the images themselves.
(2) To improve the accuracy and efficiency of hand bone joint recognition, the study uses the YOLOv5 network structure for training, and 21 joints are identified based on the dataset labels, and then 13 joints are obtained via screening.
(3) To solve the problems of gradient disappearance, gradient explosion, overfitting, and degradation that often accompany deep networks, the study uses the ResNet34 network [3], which is more effective in solving the degradation problem, to weaken the connection between layers of the network through its residual structure.
(4) To further improve the model accuracy, the SGD optimizer is selected for the training of joint classification based on ResNet34 through experimental comparison.
(5) To more closely match the physiological characteristics of Chinese adolescent children, the RUS-CHN scoring method is used as the evaluation marker for bone age detection.

The model was trained and tested on the public dataset of Flying Paddle AI Studio.

3.1. YOLOv5 Network

The structure of the YOLOv5 network is shown in Figure 1, which consists of five main parts, namely the Input; the Backbone network for extracting image features; the Neck layer; and the Prediction detection layer for outputting results [17].

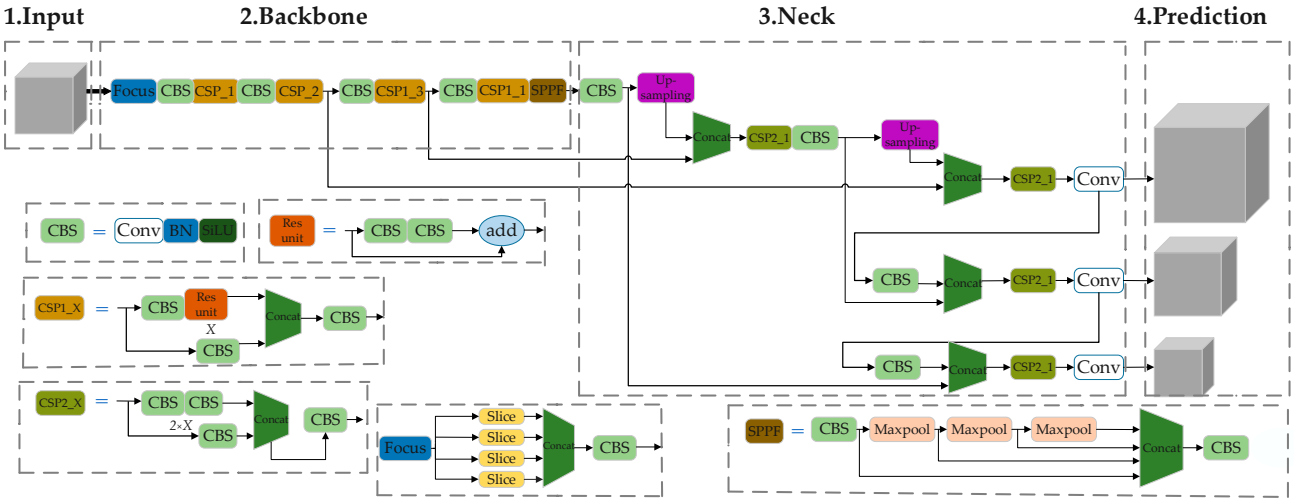

Figure 1. YOLOv5 structure diagram.

3.1.1. Input

The input side is mainly for image processing, including adaptive scaling of image size, Mosai data enhancement [18], and adaptive anchor frame calculation; uniform image size will make the speed of inference improve, and adaptive scaling ensures the stability and integrity of the image. The effect is shown in Figure 2. After normal scaling and filling, it is difficult to minimize the width of the black edges at the ends of the image, and once too much is filled, there will be redundant information, which will greatly affect the speed of algorithmic reasoning. The YOLOv5 adaptive scaling of the image, on the

other hand, automatically adds black edges of minimum width according to the size of the original image, which ensures a significant reduction in computation when the algorithm is reasoning, thus speeding up target detection.

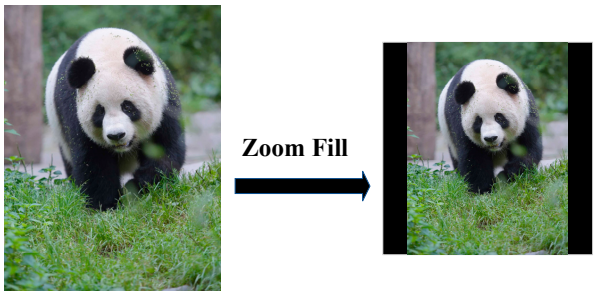

Figure 2. The effect after adaptive scaling.

As demonstrated in Figure 3, mosaic enhancement randomly chooses four photos from the training set and then randomly scales and crops the images before pasting them into a new image in a clockwise orientation. This method expands the dataset, strengthens the network's robustness, and increases the network's sensitivity to the identification of small targets. The anchor frame concept is carried over from the previous iteration, with the main difference being the use of the K-means clustering algorithm [19] to increase the distance between different classes and obtain the best anchor frame value before training, which will make it simpler for the following network to learn the image features during training. As opposed to the previous iteration of YOLOv5, this one embeds the calculation of anchor frames into the code and automatically determines the recall on the dataset's label file to obtain the proper anchor frames before training [20]. The training and detection stages of YOLOv5 can also choose whether to calculate anchor frames automatically by adjusting the parameters.

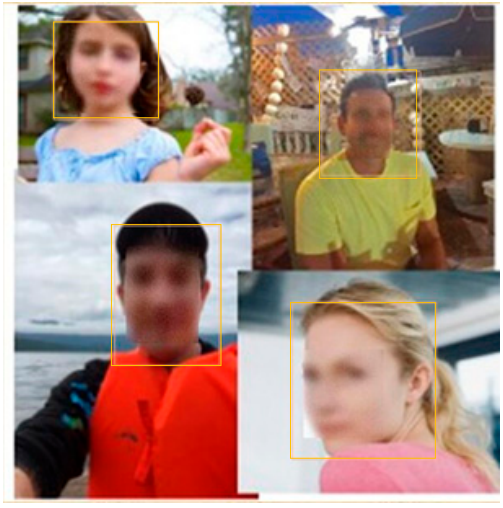

Figure 3. Mosaic's enhanced image effect.

3.1.2. Backbone Network

Four modules comprise the Backbone network of YOLOv5: Focus, CBL, CSP, and SPP. The Focus module is a new addition to YOLOv5 and is primarily used to broaden the perceptual field of view, much like null convolution. As seen in Figure 4, the Focus module enables the images to be "sliced" with a value for each pixel before entering the Backbone network. Four pictures are chopped out of the picture since it has an NCHW [21] structure with four channels. The four pictures complement one another, and the information from the W and H channels is concentrated on the C channel without any information being lost,

and the channel is increased four times. In YOLOv5, the CSP module is utilized in both the Backbone and Neck layers, as illustrated in Figure 1. CSP1_X is used in the Backbone layer, while CSP2_X is used in the Neck layer, whereas in YOLOv4, the CSP module was only used in the Backbone layer.

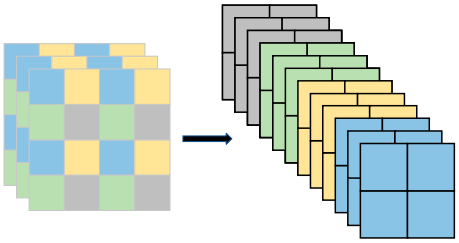

Figure 4. Slicing operation of focus.

3.1.3. Neck Layer

To achieve multi-scale fusion, the Neck portion primarily employs the feature pyramid network (FPN) and path aggregation network (PAN), two opposing procedures, as depicted in Figure 5. PAN samples from the bottom up so that the top feature contains picture location information and FPN samples from the top down so that the bottom feature map contains stronger feature information. The two features are combined to improve the network's capacity to anticipate events.

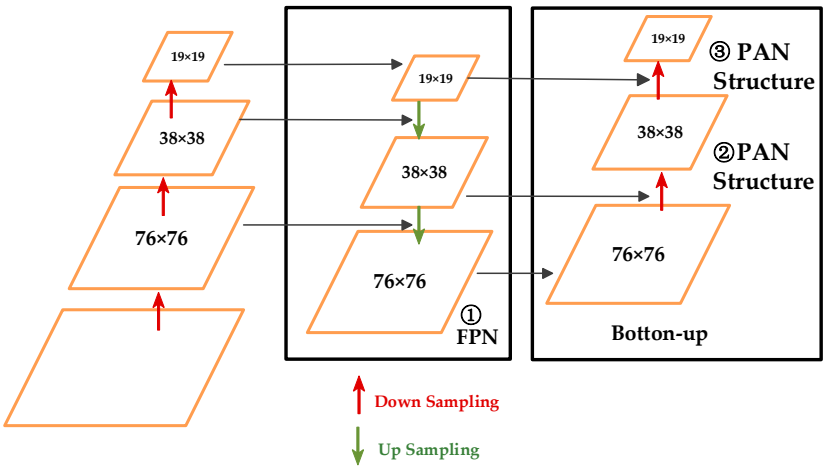

Figure 5. FPN and PAN implementation process.

3.1.4. Prediction Detection Layer

Non-maximum suppression (NMS) and loss function are mainly used in the detection. When a target is selected by several boxes at the same time, the box with the highest NMS confidence is selected as the result. The loss function consists of three functions, namely loss of classification, loss of localization, and loss of confidence.

To date, there are five versions of YOLOv5, namely YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. These five models are becoming larger and more accurate but slower in turn, and for different projects, researchers can use different versions, as shown in Figure 6, which shows the scales corresponding to the five different versions. Figure 7 shows the ratio of velocities for different versions of YOLOv5 on the COCO dataset [22]. Based on these two figures, it can be seen that YOLOv5l has a significant advantage in terms of mAP values on the COCO dataset compared to YOLOv5n, YOLOv5s, and YOLOv5m, and that YOLOv5l has a faster detection speed compared to YOLOv5x. Therefore, in this paper, the relatively balanced YOLOv5l was chosen for experiments based on accuracy and speed, two important metrics involving target detection.

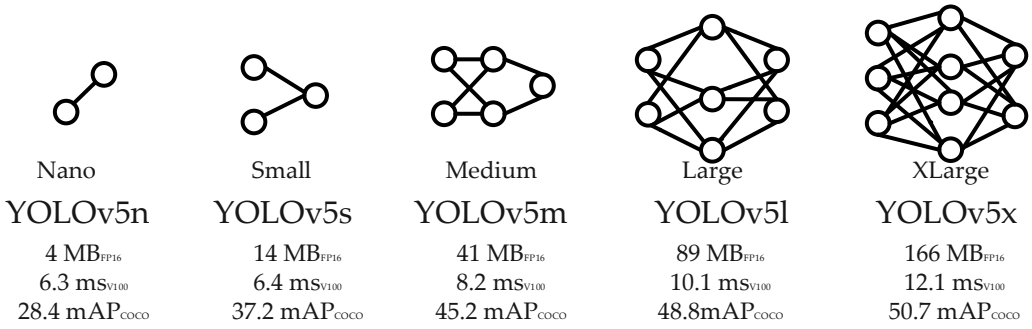

Figure 6. Depth and width comparison of YOLOv5 versions.

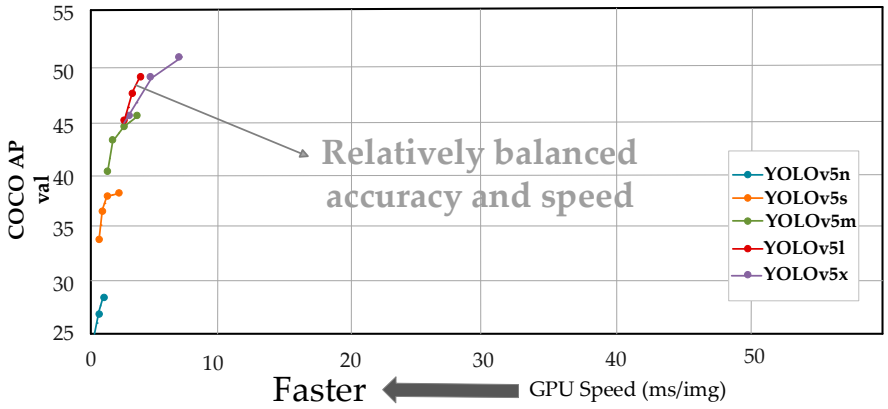

Figure 7. Comparison of speed and accuracy of YOLOv5 versions in the COCO dataset.

3.2. ResNet34 Network

A variety of breakthroughs in image processing have been rendered available by deep convolutional neural networks. Deeper networks should, in theory, be better at extracting features and obtaining better training outcomes, but they are also more prone to degradation, as seen in Figure 8, along with gradient disappearance, gradient explosion, and overfitting. The results at layer 56 are worse than those at layer 20 while training on the Cifar-10 dataset.

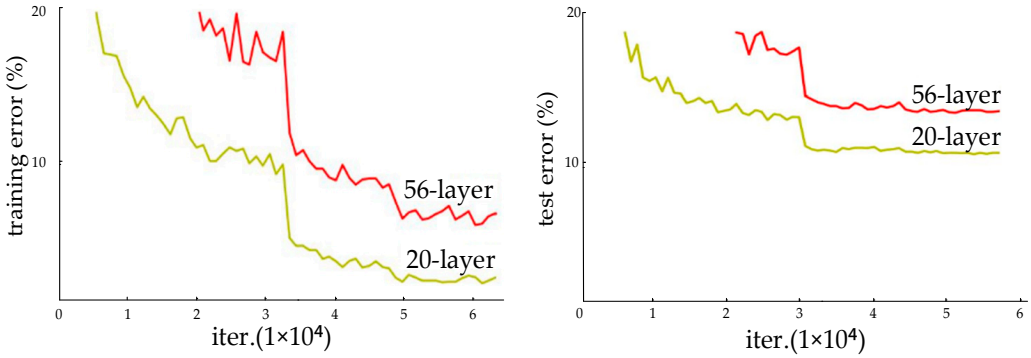

Figure 8. Comparison of training and testing results between layer 20 and layer 56 of the Cifar-10 dataset.

Furthermore, the layered structure of the proposed ResNet deepens the network layers and increases detection accuracy while also resolving the issue of gradient deterioration and disappearance. In the original publication of ResNet [3], it is suggested to employ the Batch Normalization layer via data preprocessing as well as in the network to address the issue of gradient disappearance or explosion in deep networks. Regarding the degradation issue, ResNet suggests a residual structure (residual), which makes use of layer hopping and

prevents information loss by weakening the connection between network levels through short-cut connections [23]. The residual structure is formed by stacking multiple residual blocks, and a residual block consists of two convolutional blocks. The convolutional block in ResNet does not have only a single convolution but a layer of convolutional layers + batch normalization layer (BN) + ReLU activation function together, and by setting the step size of the convolution, the change in feature map size is realized, and downsampling feature extraction is performed, which greatly improves the network performance.

As Figure 9 shows the two residual structures proposed in the original paper, the structure on the left is for networks with fewer layers, such as ResNet18 and ResNet34, and the one on the right is for networks with more layers, such as ResNet101 and ResNet152.

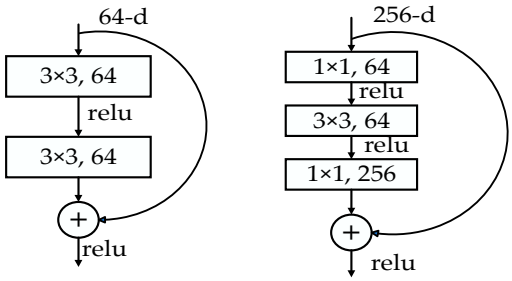

Figure 9. Structure of residuals.

A comparison experiment between ResNet18, ResNet34, and Plain18 was conducted in the ResNet paper, and it was found that ResNet34 solves the degradation problem better than the other two, as Table 1 shows the network structure of ResNet34, which contains 33 convolutional layers, one maximum pooling layer and one average pooling layer, and a fully connected layer at the end of the network.

Table 1. ResNet34 network structure.

Network Layer Name	Output Size	Network Layer Structure
Conv1	112 × 112	7 × 7, 64, stride 2 3 × 3 Maxpooling, stride 2
Conv2_x	56 × 56	$\begin{bmatrix} 3\times3,64 \\ 3\times3,63 \end{bmatrix}$
Conv3_x	28 × 28	$\begin{bmatrix} 3\times3,128 \\ 3\times3,128 \end{bmatrix} \times 3$
Conv4_x	14 × 14	$\begin{bmatrix} 3\times3,256 \\ 3\times3,256 \end{bmatrix} \times 3$
Conv5_x	7 × 7	$\begin{bmatrix} 3\times3,512 \\ 3\times3,512 \end{bmatrix} \times 3$
	1 × 1	Average pooling, 1000-d fc, softmax

3.3. SGD-ResNet34 Optimization Network

Through experimental comparison based on techniques, such as data augmentation and uniform image size, we select an SGD optimizer to aid ResNet34 network training to increase the model's accuracy, and we ultimately achieve a weight file with more accuracy. The premise of gradient descent is to apply the iterative notion to acquire the minimized loss function and model parameter values by finding the minimum of the loss function for it [24]. SGD is a stochastic gradient descent technique for altering weights to minimize the loss. Additionally, the "back and forth oscillation" feature of the SGD optimization path efficiently prohibits the model from stabilizing into local optimal solutions when the loss function has numerous local minima. One sample gradient is used for each parameter update in the model, which allows it to effectively handle enormous datasets. This gradient is given by (as in Equation (1)):

$$\theta_j = \theta_j + \alpha(y^{(i)} - h_\theta(x^{(i)}))x_j^{(i)} \quad (1)$$

3.4. Bone Age Detection Model Based on YOLOv5 and ResNet34

Despite the fact that the YOLOv5 algorithm has the benefits of quick execution, high accuracy, and a lightweight model structure, there are still a number of drawbacks, including the following: The initial clustering centers of the K-means clustering algorithm are frequently selected randomly by hand, which makes it extremely simple to select noisy data and isolated points, which easily makes the network fall into the local optimal solution dilemma [19], which results in a significant decrease in detection accuracy. The K-means clustering algorithm is used at the input side of YOLOv5 to realize the adaptive calculation of the anchor frame. Moreover, as the number of network layers increases during the convolution operation, the YOLOv5 algorithm gradually begins to deteriorate network performance, which is particularly unfavorable to the feature extraction of smaller targets and, consequently, affects the detection accuracy of small targets. In contrast to dynamic neural networks like the recurrent neural network (RNN), static neural networks like ResNet34 may be trained quickly and accurately [25]. In the original ResNet paper [3], it was shown that ResNet34 outperforms ResNet18 and Plain18 in terms of solving network degradation, and that ResNet34's modest layer count decreases the risk of overfitting. The shortcoming of ResNet34, on the other hand, is that both training and inference require a substantial amount of data. We fused the optimized ResNet34 network to create a bone age detection model (YARN) based on YOLOv5 and ResNet34 due to the fact that it is challenging to accurately recognize and estimate the age of hand bone joints using just one algorithm (YOLOv5) alone. In order to prevent YOLOv5 from entering a locally optimal solution state, the model uses the SGD in the optimized ResNet34 network. It also makes use of the residual structure within ResNet34 to reduce the phenomenon of YOLOv5's network degradation brought on by having too many network layers, and ultimately to increase the accuracy of YOLOv5 for small target recognition and detection. The two networks' strengths are successfully combined through their merging. Last but not least, *MAE* (as in Equation (2)) demonstrates that the model considerably enhances the precision and efficacy of bone age detection.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_j - \hat{y}_j| \tag{2}$$

The specific pseudo-code for this model is shown in Algorithm 1.

Algorithm 1 YARN Bone Age Detection Algorithm

```
def opt, label, yolov5(structure), resnet34(structure), model
opt = torch.randn(opt_name[0], opt_name[1])
batch reading and data enhancement based on bath_size
if opt.update: # update all models (to fix sourcechangewarning)
    for opt.weights in ['yolov5l.pt']:
        detect()
        get(opt.weights)
else:
        detect()
for key, value in all_labels:
    if opt_name == key:
        get(opt)
    else:
        next opt
for epoch in range(epochs):
    resnet34.train()
for opt, label in enumerate(train_bar):
        for outputs = model(opt):
            get(outputs)
```

The following are the specific steps of bone age detection using YOLOv5 and ResNet34 model (YARN), as shown in Figure 10.

(1) Input hand bone X-ray images for image pre-processing operation;
(2) Use the YOLOv5 network to first identify the 21 joints of the hand bone, and then filter the 13 major joints that were focused by the RUS-CHN method based on the testing parameters of YOLOv5;
(3) Train the minor joint dataset using ResNet34, finding the corresponding class of each minor joint by weighting, and then classify these 13 minor joints into 9 major classes precisely;
(4) Calculate the age of hand bones using the RUS-CHN method.

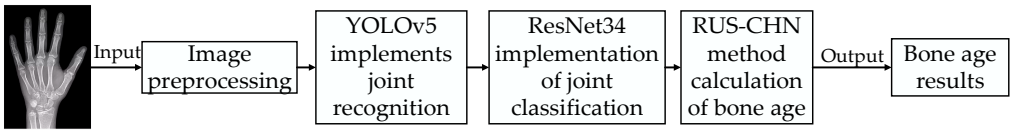

Figure 10. Bone age detection model based on YOLOv5 and ResNet34 (YARN).

4. Experimental Results and Analysis

This experiment is based on the target detection algorithm YOLOv5 and the deep residual network ResNet34. The programming language is Python 3.10. CPU configuration is Intel(R) Core (TM) i3-10110U CPU @ 2.10 GHz 2.59 GHz, and the hardware is RTX3060.

4.1. Public Dataset Preparation

We performed joint recognition and joint classification experiments on the hand bone dataset and the ResNet34 training small joint dataset, respectively, to verify the efficacy of YOLOv5 and ResNet34 in bone age detection.

The Flying Paddle AI Studio open-source dataset, which has 881 original photos and has been data upgraded and increased by a factor of 5, contains all of the hand X-ray images utilized in this paper. High-quality data are needed for the medical project of bone age detection. According to our observations, the images in this dataset varied in size and clarity (as shown in Figure 11, and the quantity of each classification was not constant (as shown in Table 2 and Figure 12), necessitating a number of image pre-processing actions. In this study, we apply rotation enhancement, adaptive scaling extended square operations, and adaptive histogram equalization with restricted contrast to data augmentation.

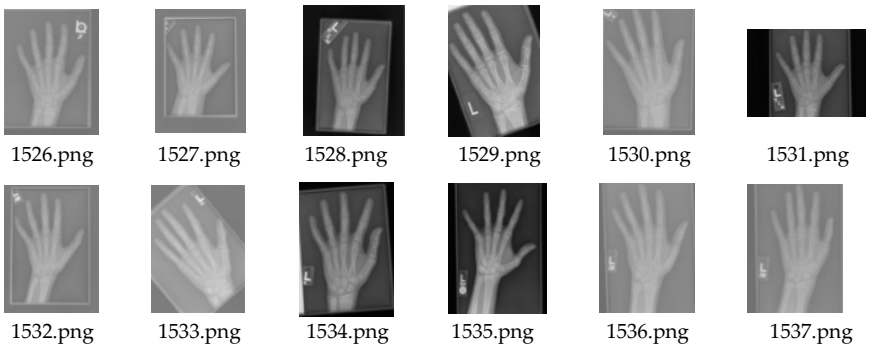

Figure 11. Hand bone dataset.

Table 2. Summary of small joint data.

Joint Name	DIP	DIP First	MCP	MCP First	MIP	PIP	PIP First	Radius	Ulan
Number of Levels	11	11	10	11	12	12	12	14	12

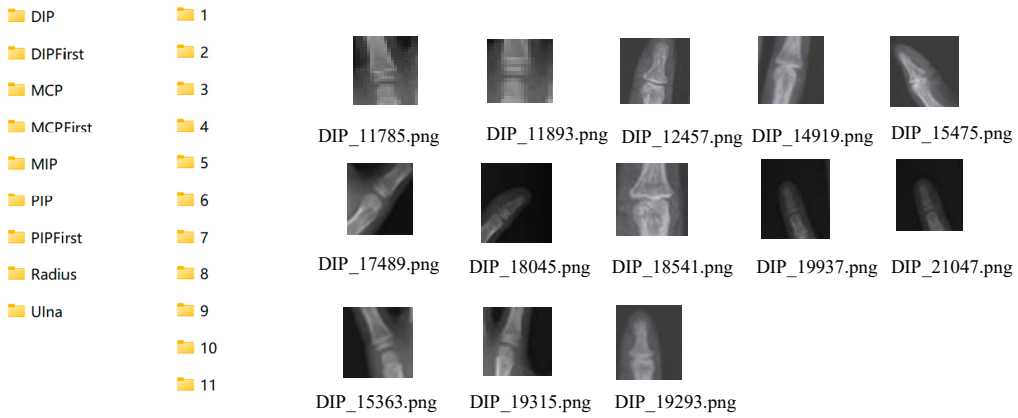

Figure 12. Small joint dataset.

Contrast-limited adaptive histogram equalization, which differs from the frequently used spatial domain image enhancement method, and histogram equalization integrate the location and grayscale information of the pixel points, and the grayscale values of the pixel points are expanded and reconstructed using the algorithm before the algorithm is trained. The output image's gray value is calculated using the bilinear interpolation method [26], which prevents the gray value from being reduced or even losing image details after histogram equalization, as illustrated in Figure 13. In this study, we executed equal restriction equalization for the hand bone dataset and the tiny joint dataset. The results are displayed in Figure 14 which significantly increased the contrast of the hand bone X-ray images.

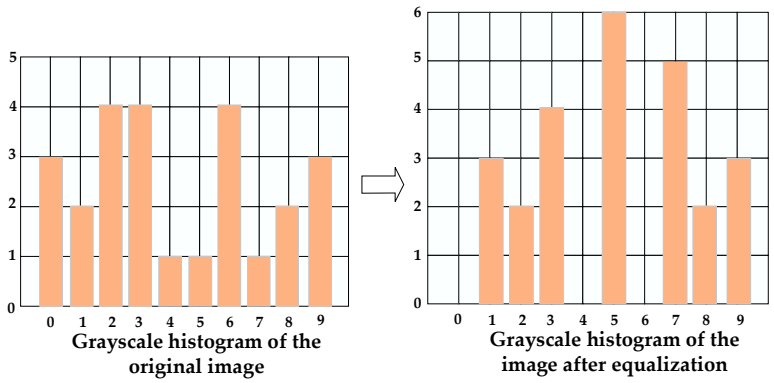

Figure 13. Reduction in grayscale values after histogram equalization.

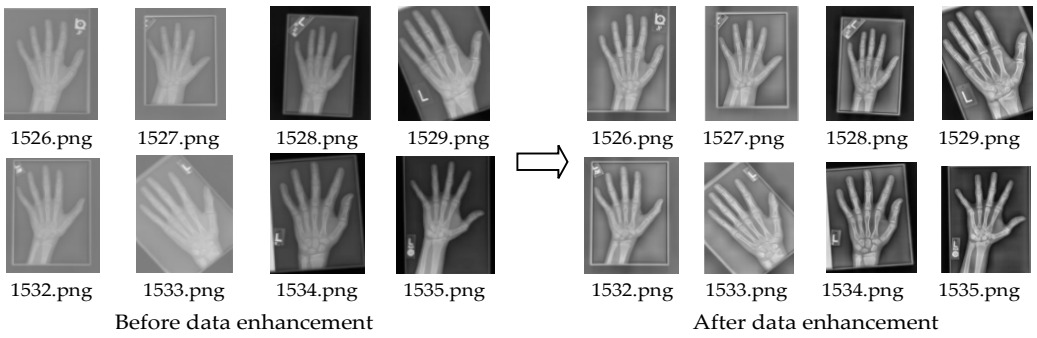

Figure 14. Effect before dataset enhancement.

Each image in the small joint dataset is randomly rotated by 45 degrees and performed five times to increase the number of datasets and enhance generalization, taking into account that there are always variations in the placement of the hand when the detector

performs X-ray detection. This makes it easier for the ResNet34 network model to learn more information about the data.

The experiments in this paper use black pixels to fill the empty part of the transformation, which significantly increases the accuracy of image recognition. The method used in this paper also applies a square transformation to the small joint dataset to prevent distortion of the image data.

The outcomes of the hand bone X-ray images following the three image pre-processing techniques mentioned above are displayed in Figure 15.

DIP_9116.png DIP_9116_0.png DIP_9116_1.png DIP_9116_2.png DIP_9116_3.png DIP_9116_4.png

Figure 15. Pre-processing effect of small joint dataset.

4.2. YOLOv5 Implementation of Joint Recognition

4.2.1. Model Training

The open-source YOLOv5 code is implemented based on the COCO dataset, which has a total of 80 classifications. In this experiment, we use YOLOv5 to train our dataset. In addition to modifying some training parameters, we also need to develop the following series of preparatory operations.

(1) Divide the training set, validation set, and test set. The training set is used to estimate the parameters in the model so that the model can learn the laws that are close to the real environment and make predictions for real situations; the validation set is used to make a preliminary assessment of the hyperparameters of the network and the ability of the model to prevent overfitting while training; and the test set is used to evaluate the prediction performance of the model. Among them, the data in the training set and the test set should not overlap, and the amount of data in the training set should be much larger than those in the test set.
(2) Modify the data path to prevent a situation where the training cannot be performed due to the non-existence of the directory and the non-existence of the data. The paths of the relevant datasets should be modified.
(3) Modify the categories. There are 80 categories in the COCO dataset, and 7 categories are used to identify joints in this experiment.
(4) Write Yaml files to write these seven categories in Yaml file format.
(5) Adjust the hyperparameters for training. YOLOv5 defines many parameters, which can be modified as needed during training and testing, and the parameters for this experiment are shown in Table 3.

Table 3. YOLOv5l training parameters.

Parameter Name	Value	Role
weights	YOLOv5l.pt	Version weights used by YOLOv5
data	'data/my_data.yaml'	Yaml files on categories and category numbers
epochs	200	Training rounds
batch-size	−1	Automatic calculation of batches per round

4.2.2. Detection Results of YOLOv5

After running all rounds, YOLOv5 saves the results in the runs/train folder in the root directory of the project, which contains the evaluation of training, training weights, and training results, as shown in Figure 16. Regarding several files of major concern in it, as shown in Table 4, the parameters to be modified during testing are shown in Table 5.

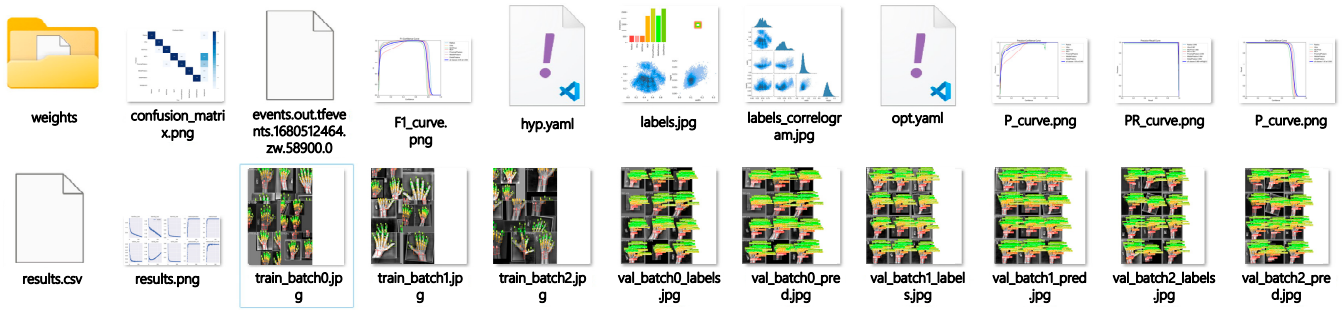

Figure 16. YOLOv5 training results.

Table 4. Description of the main files for YOLOv5 training.

Files/Folders	Instructions
weights	Best/worst weights for training
Confusion_matrix.png	Confusion Matrix
F1_curve	The relationship between the harmonic mean function of precision and recall and the confidence level
train_batch	Training results
val_bathch	Validation results

Table 5. YOLOv5 test parameters.

Parameter Name	Value	Role
weights	runs/train/exp/weights/best.pt	The best weights obtained from YOLOv5 training
source	'E:/code/Bone/Img'	Path of the dataset to detect
data	'data/my_data.yaml'	Yaml files on categories and category numbers
conf-three	0.5	Confidence threshold
you-three	0.25	IOU threshold
save-txt	true	Retain target information txt file

The final 21 joints detected are obtained according to the dataset labels. Since only 13 joints are used in the RUS-CHN method, the experiment obtains the coordinates, classification, and confidence of the upper-left and lower-right corners of each box via YOLOv5 detection of the saved txt files, and 13 joints are obtained through screening, as shown in Figure 17. It is worth noting that both the number of training rounds and the model selected may have an impact on the results, resulting in multiple or missed detections. In addition, the confidence and IOU thresholds can be adjusted appropriately to ensure that the 21 joints are detected correctly.

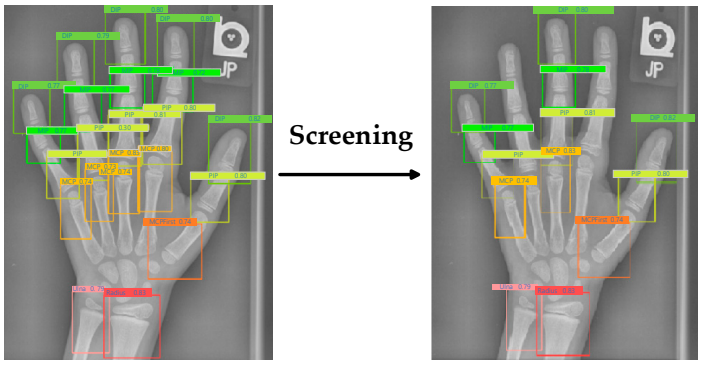

Figure 17. YOLOv5 detection results.

4.3. ResNet34 Implementation of Joint Classification

After 13 joints are obtained from the above operation. It is necessary to classify these 13 joints into nine categories, as shown in Table 6. Since the medical project requires very high results for the detection, this experiment uses ResNet34 to train the small joint dataset so that each classification has an accuracy of more than 90% for the joints to be detected.

Table 6. Table showing 13 joints corresponding to 9 classifications.

13 Joints	Joint Name 9 Categories	Classifying Grades
MCPFirst	Metacarpal bone I	MCPFirst→11
MCPThird	Metacarpal bone III	MCP→10
MCPFifth	Metacarpal bone V	MCP→10
DIPFirst	Distal phalange I	DIPFirst→11
DIPThird	Distal phalange III	DIP→11
DIPFifth	Distal phalange V	DIP→11
PIPFirst	Proximal phalange I	PIPFirst→12
PIPThird	Proximal phalange III	PIP→12
PIPFifth	Proximal phalange V	PIP→12
MIPThird	Middle phalange III	MAP→12
MIPFifth	Middle phalanges V	MAP→12
Radius	Radius	Radius→14
Ulna	Ulnar	Ulna→12

4.4. Algorithm Comparison

To improve the model accuracy, in addition to the methods, such as data enhancement and uniform image size, already used in this paper, the optimizers for training can be selected according to the merits of the results. In this experiment, two optimizers, SGD and Adam, are used for comparison. As Table 7 shows the implementation codes of the two optimizers, SGD adds the learning rate, momentum, and weight decay parameters; Adam adds the learning rate and exponential decay rate, and Table 8 shows a comparison of the effect of training the two optimization algorithms. Based on the effect comparison, the SGD optimizer is selected for training in this experiment.

Table 7. Optimizer implementation code.

SGD	opt = torch.optim.SGD(model.parameters(),lr = 0.001,moment um = 0.9, weight_decay = 0.005)
Adam	opt = torch.optim.Adam(model.parameters(),lr = 0.001,be tas = (0.9,0.999))

Table 8. Comparison of the effects of the two optimizers.

SGD		**Adam**	
MCPFirst	0.952	MCPFirst	0.832
MCP	0.928	MCP	0.804
DIPFirst	0.917	DIPFirst	0.804
DIP	0.945	DIP	0.859
PIPFirst	0.974	PIPFirst	0.888
PIP	0.958	PIP	0.828
MIP	0.962	MIP	0.856
Radius	0.929	Radius	0.838
Ulna	0.931	Ulna	0.835

To further illustrate the superiority of the new model, Table 9 compares the accuracy of bone age detection using only YOLOv5, ResNet34 alone, and YARN.

Table 9. Comparison of accuracy of bone age detection by the algorithm.

Joint Bone	YOLOv5	ResNet34	YARN
MCPFirst	0.942	0.905	0.951
MCP	0.934	0.912	0.960
DIPFirst	0.915	0.910	0.942
DIP	0.935	0.923	0.953
PIPFirst	0.927	0.916	0.948
PIP	0.956	0.914	0.965
MIP	0.961	0.907	0.962
Radius	0.931	0.911	0.939
Ulna	0.940	0.929	0.944
Average Accuracy Rate	0.938	0.914	0.952

4.5. Bone Age Detection Results

Figure 18 shows the specific calculation process. Firstly, 13 joints were classified, and then the corresponding grade of each small joint could be found through the weight of ResNet34 training. Then, the age of hand bones was calculated using the RUS-CHN method. Figure 19 shows the RUS-CHN bone maturity score scale (percentile curve), in which 3rd, 10th, 25th, 50th, 75th, 90th, and 97th represent the percentiles of bone age scores in normal subjects of the same age, and then the corresponding bone age is mapped together with the grade score table.

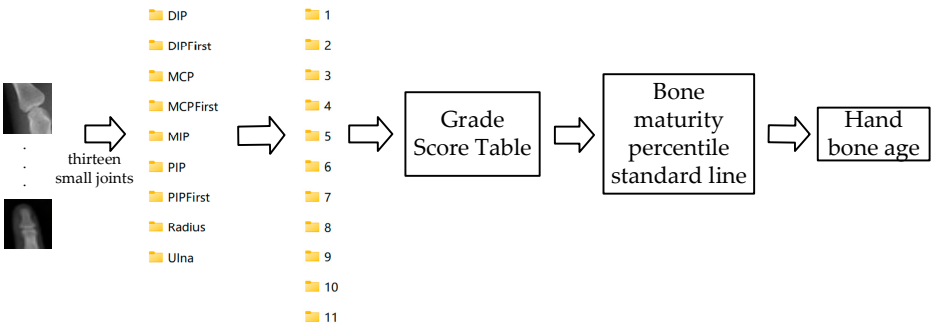

Figure 18. Bone age calculation process.

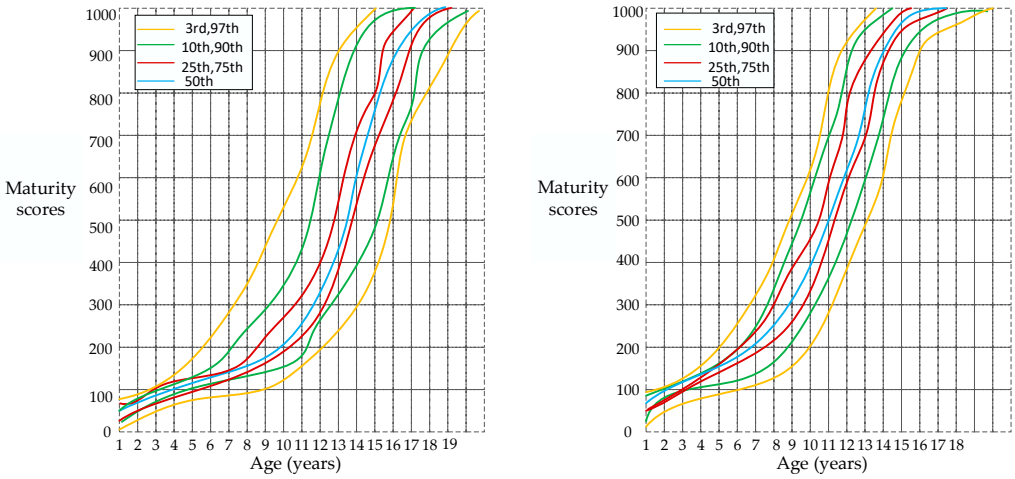

Figure 19. RUS-CHN bone maturity score criterion (left male, right female).

The final results of this experiment are shown by Pyqt5, and the steps are: (1) click "Open Pictures", (2) select "Gender", and (3) click "Start Detection"; Figure 20 shows the final detection results.

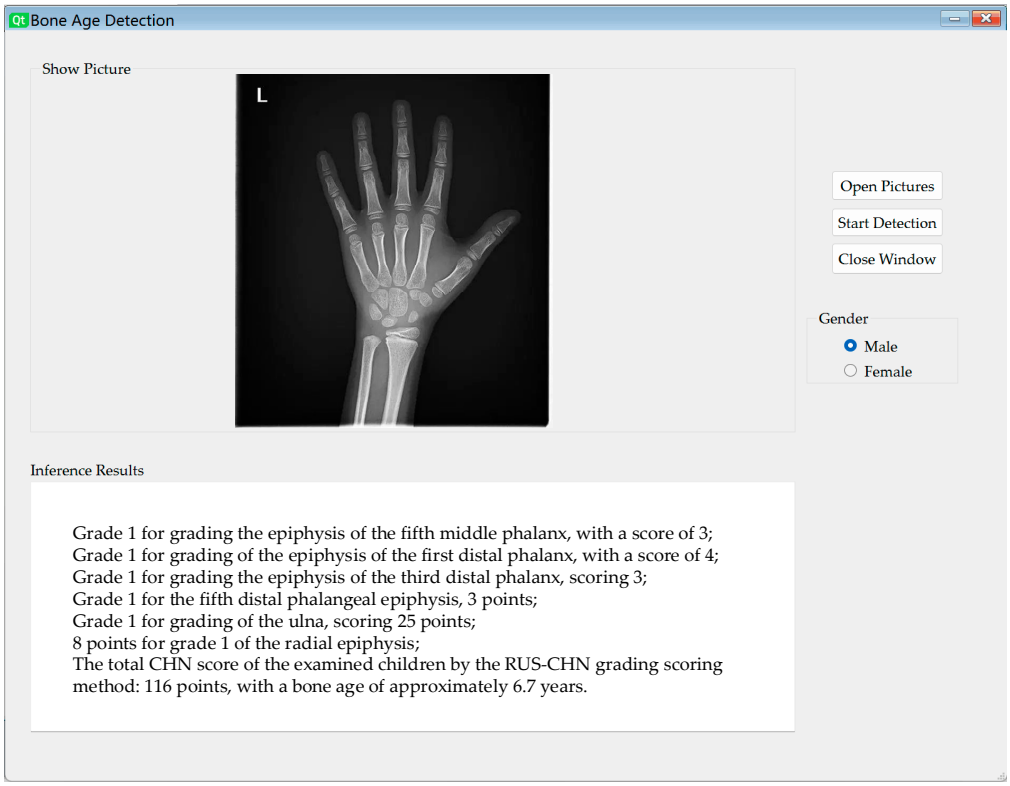

Figure 20. Detection results.

From Figure 20, it can be seen that the bone age predicted by the model is 6.7 years, which is equivalent to 80.4 months, while the bone age on the original hand bone image is labelled as 84.6 months, i.e., the error between the predicted bone age and the actual bone age is 4.2 months. In the experiment, 881 samples yielded an MAE of 4.18 months using Equation (2), and the accuracy of bone age detection reached 95.2%. Comparing the experimental results of this paper with those of [8,12,14], we found that the prediction error of this paper was 6.62 months smaller than that of [8] and 2.66 months smaller than that of [12], and the prediction accuracy was 0.6% higher than that of [14], indicating that the YARN model can accurately detect bone age.

5. Conclusions

In this study, the RUS-CHN standard compatible with Chinese adolescent children is adopted and is based on the two cutting-edge deep learning algorithms YOLOv5 and ResNet34. The ResNet34 network is, among them, optimized, and a novel bone age detection model (YARN), including YOLOv5 and ResNet34, is presented, which enhances the precision and effectiveness of small target recognition to some extent. The RUS-CHN method is only used in this paper to identify and detect 13 joints of the hand bones; however, the identification method of each joint or some joints that have a greater impact on the score can be optimized separately. These shortcomings call for additional research and improvement in the future. This research does not go into extensive detail on the relationship between the amount of YOLOv5l architectural nodes and the number of prominent features extracted in normal skeleton vision. The benefits of YOLOv5l over other versions can be further explained if this relationship is examined. The results of the error of bone age detection for adolescents worldwide remain uncertain since the dataset used in this study only comprised samples of Chinese adolescents. ResNet34 has a large computational complexity; consequently, this can be optimized in the future to be better. This study employs the less useful RUS-CHN scale. It is possible to investigate the existence of a scale with wider applicability. We will address these problems in our forthcoming study and believe they merit more research.

Author Contributions: Conceptualization, W.S. and J.S.; methodology, W.S. and Q.Z.; investigation, Q.H. and Z.L.; software, Q.Z. and L.Z.; supervision, W.S. and J.S.; writing—original draft preparation, J.S., Q.H., Z.L. and Q.Z.; writing—review and editing, W.S. and J.S.; project administration, W.S., J.S. and J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was sponsored by the Qing Lan Project of Jiangsu Province (Su Teacher's Letter [2021] No. 11), the Natural Science Research Program of Higher Education Jiangsu Province (19KJD520005) and the Young Teacher Development Fund of Pujiang Institute Nanjing Tech University ([2021] No. 73).

Data Availability Statement: Data available on request due to restrictions, e.g., privacy or ethics.

Acknowledgments: The authors would like to thank the editor and the anonymous reviewer whose constructive comments will help to improve the presentation of this paper.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

Abbreviations

TW	Tanner–Whitehouse
YOLOv5	You Only Look Once-v5
ResNet34	Residual Network-34
YARN	A bone age detection framework combining YOLOv5 and ResNet34
RUS-CHN	China 05 RUS–CHN
CNN	Convolutional Neural Network
BAA	Bone Age Assessment
GP	Greulich–Pyle
ReLU	Rectified Linear Units
PreLU	Parametric Rectified Linear Units
MAE	Mean Absolute Error
RB-FCL	A region-based feature connectivity layer
R-CNN	A region-based convolutional neural network
CBAM	Convolutional Block Attention Module
VGG	Visual Geometry Group
ROI	Region Of Interest
CHN	the standards of skeletal maturity of hand and wrist for Chinese method
SGD	Stochastic Gradient Descent
Adam	Adaptive Moment Estimation
CBL	Conv Bn Leakyrelu
CPS	Cyber Physical Systems
SPP	Spatial Pyramid Pooling
NCHW	Non-Coronal Hole Solar Wind
FPN	Feature Pyramid Network
PAN	Path Aggregation Network
NMS	Non-Maximum Suppression
BN	Batch Normalization
RNN	Recurrent Neural Network

References

1. Ning, G.; Qu, H.B.; Liu, G.J.; Wu, K.M.; Xie, S.X. Diagnostic Test of TW System Bone Age of the Radius Ulna and Short of Bones in Chinese Girls With Idiopathic Precocious Puberty. *Chin. J. Obs./Gyne Pediatr. (Electron. Version)* **2008**, *4*, 16–20.
2. Nepal, U.; Eslamiat, H. Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors* **2022**, *22*, 464. [CrossRef] [PubMed]
3. He, K.M.; Zhang, X.Y.; Ren, S.Q.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
4. Peng, L.Y.; Duan, Y.M.; Wang, Y.Q.; Peng, T.; Niu, X.K. The value of novel artificial intelligence in the determination of bone age in regional population. *China Med. Equip.* **2021**, *18*, 113–117.
5. Zhang, X.Y. Baid Paddlepaddle: Independent AI based deep learning accelerates industrial upgrading. *In-Depth Interview* **2022**, *11*, 26–37.

6. Davis, L.M.; Theobald, B.J.; Bagnall, A. Automated Bone Age Assessment Using Feature Extraction. *Lect. Notes Comput. Sci.* **2012**, *7435*, 43–51.
7. Lee, H.; Tajmir, S.; Lee, J.; Zissen, M.; Yeshiwas, B.A.; Alkasab, T.K.; Choy, G.; Do, S. Fully Automated Deep Learning System for Bone Age Assessment. *J. Digit. Imaging* **2017**, *4*, 427–441. [CrossRef]
8. Zhan, M.J.; Zhang, S.J.; Liu, L.; Liu, H.; Bai, J.; Tian, X.M.; Ning, G.; Li, Y.; Zhang, K.; Chen, H.; et al. Automated bone age assessment of left hand and wrist in Sichuan Han adolescents based on deep learning. *Chin. J. Forensic Med.* **2019**, *34*, 427–432.
9. Wibisono, A.; Mursanto, P. Multi Region-Based Feature Connected Layer (RB-FCL) of deep learning models for bone age assessment. *J. Big Data* **2020**, *7*, 67. [CrossRef]
10. Zhang, S.; Zhang, J.H. Bone Age Assessment Method on X-ray Images of Pediatric Hand Bone Based on Deep Learning. *Space Med. Med. Eng.* **2021**, *34*, 252–259.
11. Wang, J.Q.; Mei, L.Y.; Zhang, J.H. Bone Age Assessment for X-ray Images of Hand Bone Based on Deep Learning. *Comput. Eng.* **2021**, *47*, 291–297.
12. Ding, W.L.; Yu, J.; Li, T.; Ding, X. Bone age assessment of the carpal region based on improved bilinear network. *J. Zhe Jiang Univ. Technol.* **2021**, *49*, 511–519.
13. Lee, K.C.; Lee, K.H.; Kang, C.H.; Ahn, K.S.; Chung, L.Y.; Lee, J.J.; Hong, S.J.; Kim, B.H.; Shim, E. Clinical Validation of a Deep Learning-Based Hybrid (Greulich-Pyle and Modified Tanner-Whitehouse) Method for Bone Age Assessment. *Korean J. Radiol.* **2021**, *22*, 2017–2025. [CrossRef] [PubMed]
14. Mao, K.J.; Wu, K.X.; Lu, W.; Chen, L.J.; Mao, J.F. A Study of the CHN Intelligent Bone Age Assessment Method concerning Atlas Developmental Indication. *J. Electron. Inf. Technol.* **2023**, *45*, 958–967.
15. Kleinberg, R.; Li, Y.Z.; Yuan, Y. An Alternative View: When Does SGD Escape Local Minima? In Proceedings of the 35th International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; Volume 80, pp. 2698–2707.
16. Tang, S.H.; Teng, Z.S.; Sun, B.; Hu, Q.; Pan, X.F. Improved BP neural network with ADAM optimizer and the application of dynamic weighing. *J. Electron. Meas. Instrum.* **2021**, *35*, 127–135.
17. Li, X.B.; Li, Y.G.; Guo, N.; Fan, Z. Mask detection algorithm based on YOLOv5 integrating attention mechanism. *J. Graph.* **2023**, *44*, 16–25.
18. Chen, C.Q.; Fan, Y.C.; Wang, L. Logo Detection Based on Improved Mosaic Data Enhancement and Feature Fusion. *Comput. Meas. Control* **2022**, *30*, 188–194+201.
19. Fang, S.Q.; Hu, P.L.; Huang, Y.Y.; Zhang, X. Optimization and Application of K-means Algorithm. *Mod. Inf. Technol.* **2023**, *7*, 111–115.
20. Wu, L.Z.; Wang, X.L.; Zhang, Q.; Wang, W.H.; Li, C. An object detection method of a falling person based on optimized YOLOv5s. *J. Graph.* **2022**, *43*, 791–802.
21. Huang, C.; Jiang, H.; Quan, Z.; Zuo, K.; He, N.; Liu, W.C. Design and Implementation of Batched GEMM for Deep Learning. *Chin. J. Comput.* **2022**, *45*, 225–239.
22. Veit, A.; Matera, T.; Neumann, L.; Matas, J.; Belongie, S. COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images. *arXiv* **2016**, arXiv:1601.07140.
23. Xu, X.P.; Kou, J.C.; Su, L.J.; Liu, G.J. Classification Method of Crystalline Silicon Wafer Based on Residual Network and Attention Mechanism. *Math. Pract. Theory* **2023**, *53*, 1–11.
24. Li, Z.H.; Li, R.G.; Li, X.F. Improved LFM-SGD collaborative filtering recommendation algorithm based on implicit data. *Intell. Comput. Appl.* **2023**, *13*, 52–57.
25. Sharkawy, A.N. Principle of neural network and its main types. *J. Adv. Appl. Comput. Math.* **2020**, *7*, 8–19. [CrossRef]
26. Wang, J.; Pang, Y.W. X-Ray Luggage Image Enhancement Based on CLAHE. *J. Tianjin Univ.* **2010**, *43*, 194–198.