applied sciences

Article

Complex Indoor Human Detection with You Only Look Once: An Improved Network Designed for Human Detection in Complex Indoor Scenes

Yufeng Xu and Yan Fu *

School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China
* Correspondence: laura_fy@mail.hust.edu.cn

Abstract: Indoor human detection based on artificial intelligence helps to monitor the safety status and abnormal activities of the human body at any time. However, the complex indoor environment and background pose challenges to the detection task. The YOLOv8 algorithm is a cutting-edge technology in the field of object detection, but it is still affected by indoor low-light environments and large changes in human scale. To address these issues, this article proposes a novel method based on YOLOv8 called CIHD-YOLO, which is specifically designed for indoor human detection. The method proposed in this article combines the spatial pyramid pooling of the backbone with an efficient partial self-attention, enabling the network to effectively capture long-range dependencies and establish global correlations between features, obtaining feature information at different scales. At the same time, the GSEAM module and GSCConv were introduced into the neck network to compensate for the loss caused by differences in lighting levels by combining depth-wise separable convolution and residual connections, enabling it to extract effective features from visual data with poor illumination levels. A dataset specifically designed for indoor human detection, the HCIE dataset, was constructed and used to evaluate the model proposed in this paper. The research results show that compared with the original YOLOv8s framework, the detection accuracy has been improved by 2.67%, and the required floating-point operations have been reduced. The comprehensive case analysis and comparative evaluation highlight the superiority and effectiveness of this method in complex indoor human detection tasks.

Keywords: human detection; CNN; indoor scene; YOLOv8

check for updates

Citation: Xu, Y.; Fu, Y. Complex Indoor Human Detection with You Only Look Once: An Improved Network Designed for Human Detection in Complex Indoor Scenes. *Appl. Sci.* **2024**, *14*, 10713. https://doi.org/10.3390/app142210713

Academic Editors: João M. F. Rodrigues and Pedro Couto

Received: 20 September 2024
Revised: 28 October 2024
Accepted: 18 November 2024
Published: 19 November 2024

1. Introduction

With the continuous development of computer vision, human detection technology plays a crucial role in many practical applications. Especially in indoor environments, human detection has broad application prospects in fields such as security monitoring [1,2], smart homes [3,4], human–computer interaction [5], and abnormal behavior detection [6,7], among others. However, due to the complexity of indoor scenes, factors such as changes in lighting, occlusion, background interference, and scale variations pose some challenges [8]. Although advances in deep learning technology have made some breakthroughs in this issue [9,10], there is still a need for targeted improvements to existing technologies based on human detection in complex indoor environments.

Traditional detection algorithms use manual feature detection, and the quality of manually designed features determines the accuracy of the detection algorithm [11]. The emergence of deep learning has the potential to address some of the limitations of traditional techniques [12]. Deep learning technology has become the main method of feature extraction due to its powerful learning ability and ability to express features, and it has been gradually integrated into object detection algorithms, thereby greatly improving the accuracy and efficiency of detection and enabling real-time monitoring [13–16]. The ideal algorithm for human detection technology is one that achieves both high accuracy and high efficiency. The detection algorithm must be able to accurately locate and recognize the targets in every frame of an image or video while remaining unaffected by external environmental factors and inherent variability. The YOLO series has the advantages of faster detection speed and better real-time performance by transforming object detection problems into regression problems while determining the category and location of the target [13].

However, the complexity of indoor environments poses a challenge to improving the accuracy of human detection. Firstly, indoor environments typically contain various objects, furniture, and structures that may obstruct or partially hide objects of interest, resulting in obstructed and chaotic scenes. At the same time, indoor scenes also have various backgrounds, such as walls, furniture, and decorations, making the color and contour features in the image cluttered and increasing the difficulty of neural networks distinguishing effective features of human targets from the surrounding environment [8]. Secondly, the position of the camera in the indoor space may result in different perspectives and angles, affecting the appearance of the target. This can lead to significant differences and variations in the scale and size of objects in the indoor environment, from small details to large furniture, which increases the complexity of detection and recognition tasks [9,17]. Although some progress has been made on these issues, there are still shortcomings. Thirdly, the variability of lighting in different indoor spaces may pose challenges to object detection systems, including differences in natural light, artificial lighting, shadows, and reflections [9,18,19]. Uneven lighting creates distinct shadows and highlights in different parts of the human body, increasing the variability of appearance. Dark lighting can blur human details, reducing the accuracy and stability of detection algorithms for human targets. Due to the combined effect of these factors, the algorithm needs to have strong light adaptability and robustness for indoor human detection tasks under complex lighting conditions.

Based on the above issues, this paper proposes a novel deep learning method specifically designed for indoor human detection using the YOLOv8 architecture, called complex indoor human detection YOLO (CIHD-YOLO). In order to effectively extract the required features for human detection from occlusion and differential illumination, an optimized network structure called the generalized separated and enhancement aggregation network (GSEAN) was designed to replace the C2f module in the YOLOv8 neck network. A lightweight convolution called global spatial and channel reconstruction convolution (GSCConv) was also added to the neck network to compress spatial and channel redundancy. Then, the spatial pyramid pool was combined with effective partial self-attention (SPPEPSA) to ensure the network's ability to extract features at different scales. Finally, the proposed neural network was trained using a self-built dataset consisting entirely of indoor human body images (Human in Complex Indoor Environments Dataset, HCIE dataset). The experimental results show that the model significantly improves the detection accuracy of indoor human bodies without increasing floating-point operations. It can detect targets of different scales and recognize human bodies under low illumination.

The main contributions of this paper are as follows:

1. A new method for human detection through vision named CIHD-YOLO has been developed based on deep learning. The method is optimized and adapted based on the YOLOv8 architecture, which can significantly improve detection accuracy in complex indoor environments.
2. Due to the lack of a dedicated dataset for indoor human detection, the HCIE dataset was created. The new dataset combines multiple dimensions, such as different camera angles, subtle differences in lighting, indoor obstacles, and diverse populations composed of different age groups, forming a comprehensive resource.
3. The combination of spatial pyramid pooling and the efficient partial self-attention mechanism (SPPEPSA) allows the network to extract features at different scales and aggregate them locally, enhancing the model's ability to capture critical information. This improves the model's detection capability for human subjects at various scales.

4. An optimized network architecture GSEAM was proposed, which compensates for the losses caused by occlusion and illumination level differences by combining depthwise separable convolutions and residual connections, enabling it to extract effective features from visual data with poor illumination levels in indoor environments.

2. Related Works

The algorithms for human detection in computer vision can be categorized into traditional machine learning-based methods and novel deep learning-based methods [20]. Traditional machine learning methods rely on detection algorithms that manually extract image features to detect targets [11]. Schwartz et al. enhanced widely used edge-based features through texture and color information and employed a partial least squares (PLS) analysis to achieve human localization and tracking [21]. However, this enhancement resulted in an extremely high-dimensional feature space. Ahmed et al. applied the rotating histogram of oriented gradient (RHOG) algorithm and machine learning-based SVM classifier in a top-down view to significantly improve detection performance [22]. However, this method has the drawbacks of traditional machine learning, as its multi-level operations cannot achieve good real-time speed and detection accuracy, making it difficult to apply in practical scenarios.

Deep learning-based methods are categorized into two-stage and one-stage algorithms, which are known for their real-time performance and detection accuracy. In multi-stage detectors like the R-CNN series [15,23–25], one model identifies object regions while another model classifies and detects object positions [23]. Although these methods yield high accuracy, they suffer from increased computational costs and longer processing times. In contrast, one-stage algorithms like YOLO [13,26–29] and SSD series [14,30] segment images into grids, predicting object categories and bounding box coordinates directly without intermediate tasks. They excel in agility and real-time capabilities. Fu et al. integrated the Residual-101 classifier with SSD to create DSSD [30], thereby enhancing context in object detection. Similarly, Wang et al. combined efficient training tools with YOLOv7 [27], thereby achieving superior speed and accuracy (5FPS to 120FPS) compared to existing detectors. While one-stage algorithms may slightly lag in accuracy compared to two-stage algorithms, they offer notable advantages in speed and real-time performance.

Launched by Ultralytics in 2023, YOLOv8 is currently a research hotspot and widely used in the field of indoor object detection. The network structure of YOLOv8 is shown in Figure 1. Aoki et al. applied CNN to extract features from the detection objects cropped by the YOLOv8 object detection algorithm and then integrated these features into a single feature vector using LSTM to achieve position localization of multiple indoor targets [31]. Safaldin et al. enhanced the YOLOv8 model's ability to detect small targets and specific motion detection in various visual environments by focusing on large-sized feature maps and introducing Bi-PAN-FPN [32]. Although this method has generalization in various visual environments, it does not take into account that indoor scenes often contain dense elements and small-scale features of different scales. Han et al. proposed an intelligent monitoring method for the real-time distribution of indoor pedestrians based on deep learning and spatial partitioning [33]. Enhanced YOLOv8 and DeepSORT models were employed to intelligently generate pedestrian IDs and location data, facilitating evacuation count and direction determination. This approach surpasses alternative algorithms in detection accuracy and efficiency. However, further experiments and evaluations are required to assess its performance adequately, considering factors like camera resolution, monitoring distance, passenger occlusion, and environmental brightness that impact detection accuracy.

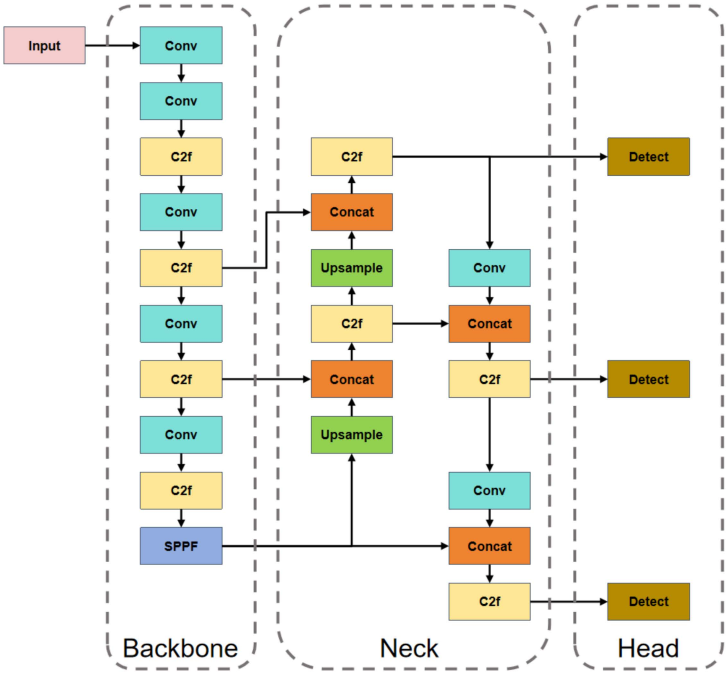

Figure 1. Network architecture of YOLOv8.

Unlike the outdoors, indoor environments are characterized by different lighting conditions, potential obstacles, such as furniture, and enclosed spaces, all of which can seriously affect the accuracy and quality of human detection [34]. Yu et al. proposed a novel real-time face detector, YOLO-FacedV2, based on the YOLOv5 architecture [9]. YOLO-FacedV2 compensates for the response loss of occluded faces and enhances the response of unobstructed faces through the attention network module SEAM to improve the effectiveness of object occlusion detection. This method is designed to solve facial occlusion, so its drawback is that it has limited effectiveness when facing multi-scale occlusion problems. Cao et al. designed a multi-scale small object detection structure for MCS-YOLO to improve the recognition sensitivity of dense multi-scale objects [35]. This method has effectiveness and superiority in autonomous driving object detection tasks, but it has not been extended to indoor environments. In summary, there is an urgent need for a human detection method specifically designed for indoor scenes to solve the complex detection caused by factors such as lighting conditions and scale changes in indoor environments. Inspired by these methods [9,28,29,36,37], this article proposes a new approach based on YOLOv8 that combines GSEAM, SPPEPSA, and GSCConv to address these challenges.

3. Methods

3.1. CIHD-YOLO Network

This article introduces the CIHD-YOLO model, which is based on YOLOv8s [38], for indoor human detection. Improving upon YOLOv8, this model addresses the limitations faced in detecting indoor human bodies amidst complex backgrounds and varying object scales. The CIHD-YOLO network architecture, which is depicted in Figure 2, highlights the enhanced components of the algorithm with red dashed boxes.

In the upgraded network design, the SPPEPSA module replaces SPPF in YOLOv8's backbone network. SPPEPSA combines SPP's multi-scale feature extraction with self-attention (PSA) for global modeling. It enhances global dependency capture by locally aggregating features with SPP and calculating correlation weights with PSA.

A refined architecture, GSEAM, replaces YOLOv8's C2f module, thereby addressing illumination variations with depth-wise separable convolutions and residual connections for challenging indoor settings.

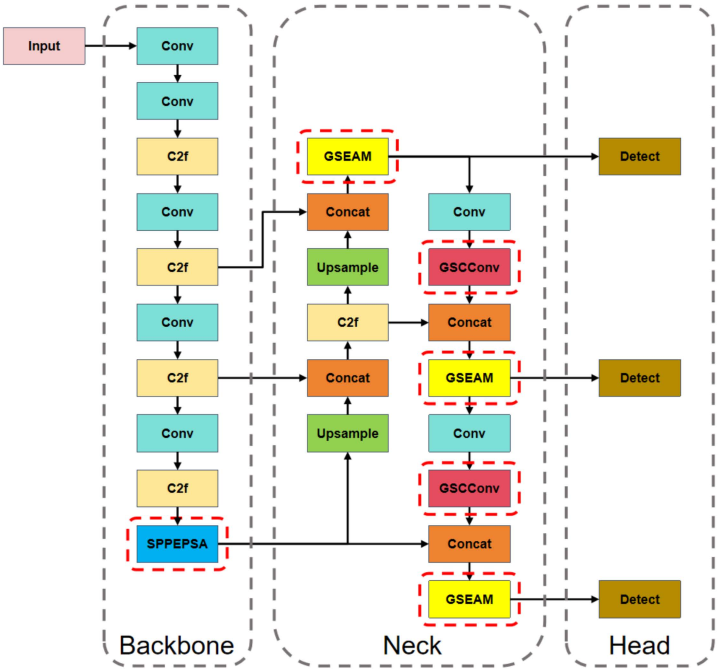

Figure 2. Network architecture of CIHD-YOLO.

GSCConv, a global spatial and channel reconstruction convolution, is integrated into the neck network following standard convolution. It leverages SCConv techniques to minimize redundancy, enhance efficiency, and maintain feature information integrity.

The resulting CIHD-YOLO network excels at capturing human features globally across scales, ensuring accurate detection in poorly lit and occluded environments.

3.2. Spatial Pyramid Pooling with Effective Partial Self-Attention (SPPEPSA)

In the human detection task of indoor scenes with limited space and shooting angles, the proportion of targets of different scales is relatively high. Therefore, a well-designed spatial pyramid pooling can significantly improve the detection performance of the model. We suggested a spatial pyramid pooling method called SPPEPSA that combines partial self-attention in the backbone network of the model.

Spatial pyramid pooling (SPP) divides feature maps into grids of varying scales, pooling features across scales to create a multi-scale feature representation [39]. By utilizing MaxPooling with different window sizes, like 1 × 1, 3 × 3, and 5 × 5, SPP enables the network to extract features at diverse scales, facilitating the detection of targets of different sizes. This pooling technique segments images into regions of different sizes, pooling features within each region to capture information across scales. MaxPooling, a common method in SPP, operates on inputs (N, C, $\mathrm{H_{in}}$, $\mathrm{W_{in}}$) and outputs (N, C, $\mathrm{H_{out}}$, $\mathrm{W_{out}}$), per Equations (1) and (2) as follows, for output shape calculation:

$$H_{out} = \left\lfloor \frac{H_{in} + 2 \times P[0] - D[0] \times (K[0] - 1) - 1}{S[0]} + 1 \right\rfloor \tag{1}$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2 \times P[1] - D[1] \times (K[1] - 1) - 1}{S[1]} + 1 \right\rfloor \tag{2}$$

In these equations, the following parameters are defined: S[i] represents the stride, K[i] denotes the kernel size, P[i] stands for padding, and D[i] refers to dilation. Maxpooling involves sliding fixed-sized windows over input feature maps, selecting the maximum value within each window to generate the output. This method preserves significant features, maintaining spatial key feature positions in the image and enhancing model invariance and robustness.

While spatial pyramid pooling algorithms have excelled in multi-scale object detection tasks [26,27,40], distinguishing human bodies from complex backgrounds in indoor settings remains a challenge. The SPPEPSA model combines spatial pyramid pooling with self-attention mechanisms to enhance multi-scale human detection in indoor environments, as depicted in Figure 3. Spatial pyramid pooling captures multi-scale features, while self-attention mechanisms enable better feature utilization, enhancing human target recognition accuracy.

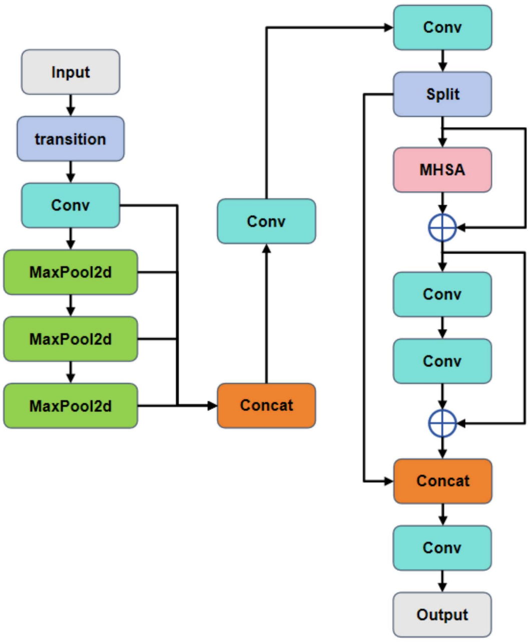

Figure 3. Network architecture of spatial pyramid pooling with effective partial self-attention (SPPEPSA).

Partial self-attention dynamically assigns weights to each feature by learning the interrelationships between features, so that the model can focus on the most relevant features, thus greatly improving the multi-scale detection capability.

Self-attention can capture the dependency relationships between different elements in the input sequence by learning the correlations between them, which helps improve the performance of the model in sequence tasks. It first calculates the attention distribution of all input information and then calculates the attention value based on the attention distribution. For the input feature information **X**, the key–value pair can be used to represent the N pixel information it contains, as follows: **X** = (K, V) = [(k_1, v_1), (k_2, v_2), . . . , (k_N, v_N)]. A query vector **q** is now given to calculate the attention distribution for all input information. The attention score of input image features on each pixel can be expressed as **s(k$_i$, q)**, which is shown in Equation (3), as follows, where **d** is the dimension of the input information:

$$s(k_i, q) = \frac{k_i^T q}{\sqrt{d}} \tag{3}$$

If an attention variable **t** is defined to represent the index position of the input information, then the formula for calculating the probability $\boldsymbol{\alpha_i}$ of the input information at the i-th pixel can be expressed using Equations (4) and (5), as follows:

$$\alpha_i = p(t = i|\mathbf{X}, q), t \in [1, N] \tag{4}$$

$$\alpha_i = \frac{\exp(s(k_i, q))}{\sum_{j=1}^{N} \exp(s(k_j, q))} = \frac{e^{\frac{k_i^T q}{\sqrt{d}}}}{\sum_{j=1}^{N} e^{\frac{k_j^T q}{\sqrt{d}}}} \tag{5}$$

The function of Equation (5) is to normalize and obtain a probability distribution where the sum of all weight coefficients is 1 and to highlight the weights of important elements. The vector ($\boldsymbol{\alpha}_\mathbf{i}$) composed of the obtained results is the attention distribution of the input feature information. Finally, the value is weighted and summed based on the weight coefficients, as shown in Equation (6), as follows:

$$attention(X) = \sum\nolimits_{i=1}^{N} \alpha_i v_i = \sum\nolimits_{i=1}^{N} \frac{v_i e^{\frac{k_i^T q}{\sqrt{d}}}}{\sum_{j=1}^{N} e^{\frac{k_j^T q}{\sqrt{d}}}} \quad (6)$$

In the described approach (Figure 3), the feature map undergoes spatial pyramid pooling initially, merging features of various scales post-pooling. Subsequently, the post-1 × 1 convolution channel features are split evenly into two segments, with only one part entering the multi-head self-attention module. Finally, these two segments are fused via 1 × 1 convolution. This enhancement weights multi-scale feature representations using self-attention mechanisms, thereby facilitating the adjustment and integration of features based on their significance.

3.3. Generalized Separated and Enhancement Aggregation Network (GSEAM)

The generalized separated and enhancement aggregation network (GSEAM) is introduced to improve feature focus in low-light scenarios. Initially, feature maps at different scales are split and later recombined after Rep-NCSP module processing. Depth-wise separable convolutions with residual connections are utilized to address illumination discrepancies. A two-layer fully connected network is then used to project features into a reduced-dimensional space for better detection. GSEAM replaces the C2f module by merging different-scale feature maps in the neck network to enhance feature extraction efficiency.

The RepNCSP module enhances human body feature identification by extracting spatial and semantic information through convolutional structures and residual connections. Its three layers manage feature extraction, channel fusion, and transformation, capturing intricate image details and improving performance in low-light conditions. Illustrated in Figure 4, RepConvN conducts feature extraction across scales through iterative convolutions. RepNBottleneck combines multi-layer convolutions with residuals to merge information from various levels, allowing the model to address global and local features simultaneously.

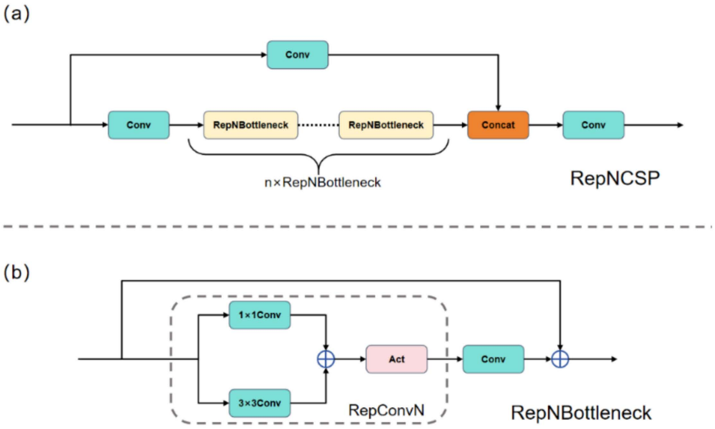

Figure 4. Network architecture of RepNCSP and RepNBottleneck. (**a**) The network architecture of RepNCSP; (**b**) The network architecture of RepNBottleneck.

Depth-wise separable convolution enhances the processing of feature maps under varying illumination by decreasing the parameter count and enhancing feature extraction efficiency [41]. To combat potential feature degradation and ambiguity stemming from uneven lighting, this paper introduces depth-separable convolution with residual connections within the GSEAM module, combining depth-wise and pointwise convolutions. The computational formulation for this process is outlined in Equations (7)–(9), as follows:

$$Conv^{P}_{(i,j)} = \sum_{m}^{M} W_P \cdot y_{(i,j,m)} \tag{7}$$

$$Conv^{D}_{(ij)} = \sum_{k,l}^{K,L} W_D \odot y_{(i+k,j+l)} \tag{8}$$

$$DSConv_{(i,j)} = \sum_{m}^{M} W_P \cdot \left(\sum_{k,l}^{K,L} W_D \odot y_{(i+k,j+l)} \right) \tag{9}$$

Among them, **k**, **l**, and **m** represent offsets in the depth, width, and height dimensions, respectively, and $y_{(i,\,j)}$ denotes the pixel position index on the feature map, with W as the trainable weight matrix.

In this process, the input features undergo depth-wise separable convolution followed by GELU activation and BatchNorm operations. The resulting output is then added to the initial input, allowing the network to learn a residual function rather than a complete input-to-output mapping. This approach eases the learning burden by focusing on the discrepancy between the input and the desired output. Additionally, unlike the SiLU function in YOLOv8, the GELU activation function, which is detailed in Equations (10) and (11) as follows, is utilized in this part:

$$GELU(x) = x \cdot P(X \le x) = x \int_{-\infty}^{x} \frac{e^{-\frac{(X-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma}} dX \tag{10}$$

$$GELU(x) = 0.5x \left[1 + tanh\left(\sqrt{\frac{2}{\pi}} \left(x + 0.047715x^3 \right) \right) \right] \tag{11}$$

Equation (10) involves the cumulative function of the Gaussian normal distribution, which is denoted as P(X ≤ x), where μ and σ represent the mean and standard deviation of the distribution. Since Equation (10) cannot be directly calculated, the calculation formula for the GELU activation function can also be approximated as Equation (11). After average pooling and weighting of the output results, the network architecture of the entire GSEAM module is shown in Figure 5.

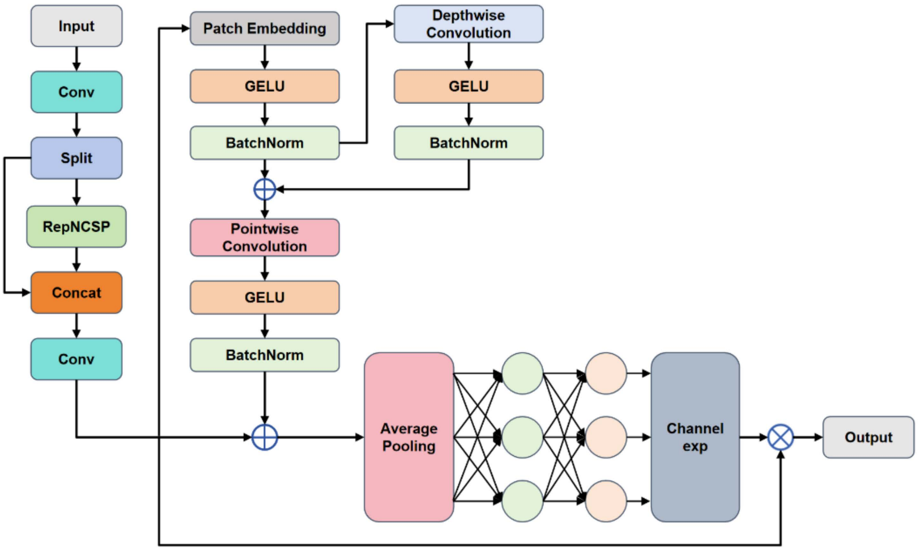

Figure 5. Network architecture of generalized separated and enhancement aggregation network (GSEAM).

For the input feature map **y**, a global average pooling operation is performed to obtain a feature vector **z**, where each channel of **z** corresponds to the average value of the corresponding channel of the input feature map. This can be represented by Equation (12), as follows:

$$z_i = \frac{1}{h \times w} \sum\nolimits_{x=1}^{h} \sum\nolimits_{y=1}^{w} y_i(x, y) \tag{12}$$

In this context, $\mathrm{z_i}$ represents the i-th element of the feature vector **z**, while $\mathrm{y_i}$(x, y) signifies the pixel value of the i-th channel of the input feature map **y** at position (x, y). Here, h and w denote the height and width of the input feature map, respectively.

The fully connected layer maps the feature vector **z** to the attention weight vector **a**. It then leverages this attention weight to weight the input feature map **y**, resulting in a weighted feature map, as depicted in Equations (13) and (14) as follows:

$$a_i = Sigmoid(Wz_i + b) = \frac{1}{1 + e^{-\left(\frac{\mathrm{W}}{h \times w} \sum_{x=1}^{h} \sum_{y=1}^{w} y_i(x,y) + b\right)}} \tag{13}$$

$$y\prime : y\prime_i(x, y) = y_i(x, y) \times a_i = \frac{y_i(x, y)}{1 + e^{-\left(\frac{\mathrm{W}}{h \times w} \sum_{x=1}^{h} \sum_{y=1}^{w} y_i(x,y) + b\right)}} \tag{14}$$

Here, **W** represents the weight matrix of the fully connected layer, **b** is the bias term, and $\mathrm{y'_i}$(x, y) denotes the pixel value of the i-th channel of the output feature map $\mathbf{y'}$ at position (x, y).

The GSEAM module splits feature maps into two segments, recombining one after processing through the RepNCSP module to enhance sensitivity to subtle features in low-light conditions and prevent the loss of dark details. Depth-wise separable and pointwise convolutions enhance feature representations by dissecting feature maps spatially and in depth. A two-layer fully connected network then consolidates information from each channel, addressing information loss from uneven illumination by learning relationships between adjacent human target areas. Finally, the resulting output is multiplied with the original features to amplify feature representation, thereby effectively resolving illumination discrepancies.

3.4. Global Spatial and Channel Reconstruction Convolution (GSCConv)

In object detection networks, the neck network plays a crucial role in extracting pertinent features from the backbone, processing this data, and forwarding it to the recognition head. To enhance both accuracy and efficiency in handling complex images without increasing computational demands, this paper introduces a novel convolution module named global spatial and channel reconstruction convolution (GSCConv), which is based on the SCConv module [36].

Drawing inspiration from the GSConv module [37], the GSCConv module incorporates the Conv, SCConv, Concat, and shuffle modules. The mathematical formulation is provided in Equation (15), as follows:

$$X_{out} = f_{shuffle}(f_{concat}(f_{SCConv}(f_{Conv}(X_{in})), f_{Conv}(X_{in}))) \tag{15}$$

As shown in Figure 6, in GSCConv, data undergo a flow from standard convolution to SCConv, followed by concatenation with standard convolution output. The shuffle module then processes the concatenated data, ensuring a random mixture that evenly blends information from standard convolutions into the SCConv output, promoting the exchange of local feature details across channels.

The SCConv module within the GSCConv framework, as presented in this article, is composed of two integral units: the spatial reconstruction unit (SRU) and the channel reconstruction unit (CRU), which are depicted in Figure 7.

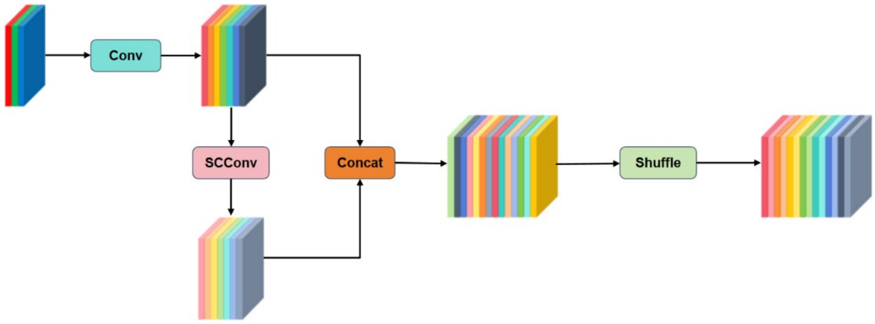

Figure 6. Network architecture of global spatial and channel reconstruction convolution (GSCConv).

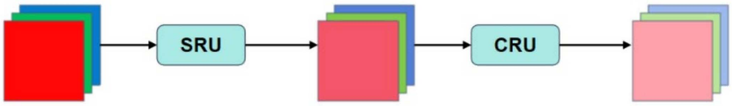

Figure 7. Network architecture of spatial and channel reconstruction convolution (SCConv).

SRU focuses on addressing spatial redundancy through separation and reconstruction operations.

Initially, trainable parameters are computed to derive the information weight $\mathbf{W_1}$ and non-information weight $\mathbf{W_2}$. Multiplying the input feature **X** by W_1 and W_2 yields two weighted features: $\mathbf{X_1}$, with higher information content and $\mathbf{X_2}$, with lower information content. Subsequently, through the cross-reconstruction operation, these distinct weighted information features are merged to produce a feature with enhanced information content, as depicted in Equation (16), as follows:

$$\begin{cases} X_1 = X \bigotimes W_1, \\ X_2 = X \bigotimes W_2, \\ X^w = (X_{11} \oplus X_{22}) \bigcup (X_{21} \bigoplus X_{12}). \end{cases} \tag{16}$$

The spatial fine feature mapping $\mathbf{X^w}$ retains redundancy in the channel dimension, which is a challenge addressed by CRU.

CRU divides the input spatially refined features into two parts, X_{up} and X_{low}, based on channels αC and (1- α)C. Then, it performs a transformation operation using the segmented features and performs weighted summation on the two parts to obtain the final output. The calculation process is shown in Equation (17), as follows, in which β_1 and β_2, respectively, represent the weight vectors of two features:

$$X_{output} = \beta_1 \cdot \left(GWC\left(X_{up}\right) + PWC\left(X_{up}\right)\right) + \beta_2 \cdot \left(PWC\left(X_{low}\right) + X_{low}\right) \tag{17}$$

Integrating SCConv into the model enhances performance by reducing redundant features, leading to decreased complexity and computational costs while maintaining performance. However, spatial information transmission to channels during image conversion may lead to some loss of semantic information due to spatial dimension compression and channel dimension expansion.

The GSCConv module, by merging standard Conv and SCConv, combines their advantages. By integrating it into the neck, feature redundancy is minimized with SCConv while preserving inter-channel correlations through regular convolution. This integration enhances detection efficiency without increasing computational complexity, thereby meeting real-time human detection requirements effectively.

4. Experimental Evaluation

4.1. HCIE Dataset

To address the absence of a dedicated dataset for indoor human body detection in existing publicly available object detection datasets, this study introduced a new dataset named the Human in Complex Indoor Environments Dataset (HCIE dataset).

The HCIE dataset is sourced from two main channels: integrating indoor human images from existing open-source target detection datasets and collecting public indoor images from the internet. The dataset's image distribution was carefully planned from various viewpoints to ensure that models trained on it exhibit a degree of generalization.

Following the construction principles of established object detection datasets [42,43], the HCIE dataset was chosen based on the following specific criteria:

- Diversity: For data diversity, the dataset encompasses diverse lighting conditions, shooting angles, backgrounds, human body sizes, and other variations within indoor scenes. This approach enhances the model's generalization capability.
- Class Balance: While focusing solely on the human class, the dataset incorporates a range of factors like diverse ages, genders, body types, and poses of human subjects in indoor settings (standing, sitting, lying down, etc.). This approach enables the model to learn features from a spectrum of categories.
- Similar targets: The dataset includes both the human body to be detected and objects that are partially similar to the human body that are not intended to be detected, but when labeled, only the target to be detected is labeled.
- Practicality: Ensure the quality of the collected data, preferably with a size close to the actual usage scenario.
- Data integrity: To ensure consistency and completeness between images and annotations in the dataset, and to avoid missing or inconsistent data, LabelImg is used to label all data in this dataset.

The final dataset contains five thousand images, which are divided into training, validation, and testing subsets in an 8:1:1 distribution. The examples of some images in Figure 8 demonstrate that the dataset includes different postures and lighting conditions of human bodies of different age groups in indoor scenes.

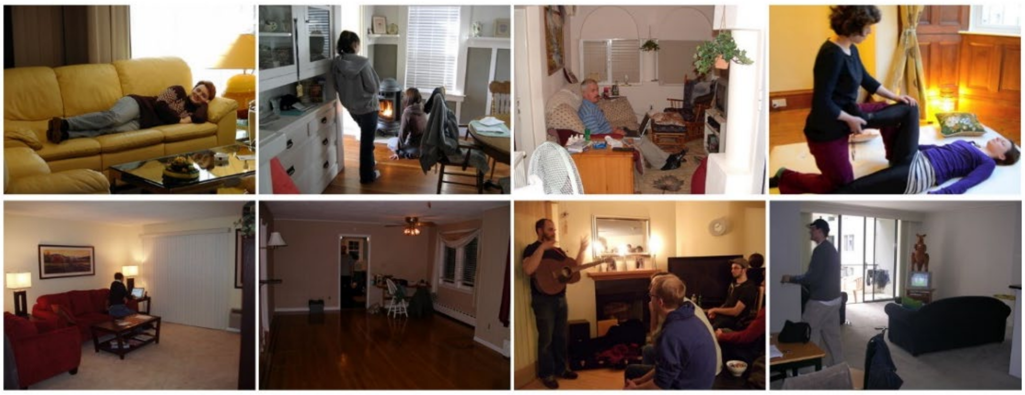

Figure 8. Example of indoor human detection images in HCIE dataset.

Figure 9 shows the label distribution of the HCIE dataset. In Figure 9a, the x and y coordinates denote the ratio of the bounding box center point's coordinates to the image's length and width, respectively. This description illustrates the distribution of bounding box center points within the image. In Figure 9b, the x and y coordinates indicate the ratio of the detection target's horizontal and vertical length within the bounding box to the image's horizontal and vertical length. This representation highlights the distribution of target aspect ratios in the training set.

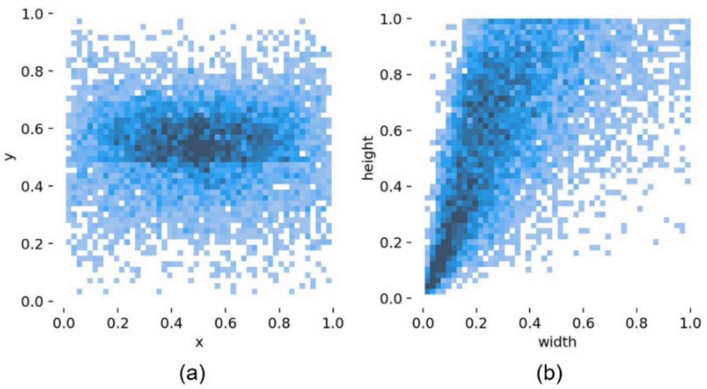

Figure 9. Dataset label distribution. (**a**) The position of the bounding box center point relative to the entire image; (**b**) The aspect ratio of the target in the image relative to the entire image.

4.2. Experimental Process

In this study, the model was trained and assessed on the HCIE dataset with specific training parameters. The number of epochs is set at 100, the batch size is set at 20, and the SGD optimizer is utilized with an initial learning rate of 0.02. Selecting suitable training parameters is essential before initiating model training. After thorough adjustments, the finalized model training parameters are outlined in Table 1.

Table 1. Model training parameters.

Parameter	Value
Initial Learning Rate	0.02
Epochs	100
Batch Size	20
Imgsz	640
Optimizer	SGD
Weight Decay	0.0005
Momentum	0.937

The experimental setup includes an AMD Ryzen7 5800H CPU (AMD, Santa Clara, CA, USA) and NVIDIA GeForce RTX 3060 GPU (Nvidia, Santa Clara, CA, USA) running on Windows 11. The method and experiments in this article are implemented using a PyTorch 2.2.1-based deep learning framework. The development software comprises PyCharm 2023.3.4 and Python 3.11.

4.3. Evaluation Criteria

In this study, key evaluation metrics include mAP at IoU 0.5 (mAP50), mAP50-95, model parameter count, and GFLOPS. The mAP provides a comprehensive assessment of model performance across different precision and recall scenarios that are commonly utilized in object detection evaluations. Parameters (Params) and floating-point operations (FLOP) assess algorithm or model complexity. Equations (18)–(20) define precision (P), recall (R), and mAP.

$$P = \frac{TP}{TP + FP} \times 100\% \tag{18}$$

$$R = \frac{TP}{TP + FN} \times 100\% \tag{19}$$

$$mAP = \frac{\sum_{i=1}^{K} \int_0^1 P(R) dR \times 100\%}{K} \tag{20}$$

Among them, TP represents the number of correctly predicted positive samples, TN represents the number of correctly predicted negative samples, FP represents the number of negative samples classified as positive, FN represents the number of positive samples classified as negative, and K represents the number of categories.

In the evaluation criteria, mAP50 refers to the average precision when the intersection over union (IoU) is above 50%. IoU is used to measure the degree of overlap between the area detected by the model and the actual target area. Its calculation formula is shown in Equation (21), where S_A and S_B refer to the bounding box and ground truth, respectively.

$$IoU = \frac{|S_A \cap S_B|}{|S_A \cup S_B|} \tag{21}$$

Similarly, mAP50-95 represents the average mAP at different IoU thresholds (ranging from 0.5 to 0.95, with a step size of 0.05).

4.4. Experimental Results and Analysis

This study conducted ablation and comparative experiments to assess the CIHD-YOLO algorithm. The ablation experiments compared the original YOLOv8s model with three enhancements, focusing on changes in accuracy indicators to gauge the effectiveness of each improvement. Additionally, CIHD-YOLO was compared with popular object detection algorithms to evaluate its performance and accuracy.

Figures 10 and 11 illustrate the training process of the CIHD-YOLO model, using box loss to quantify the error between predicted and annotated boxes. The loss curves in Figure 10 show a rapid decrease during training, followed by a gradual stabilization. The curves for training and validation losses align closely, demonstrating a smooth descent without oscillations or increases. This suggests effective training with no signs of underfitting or overfitting.

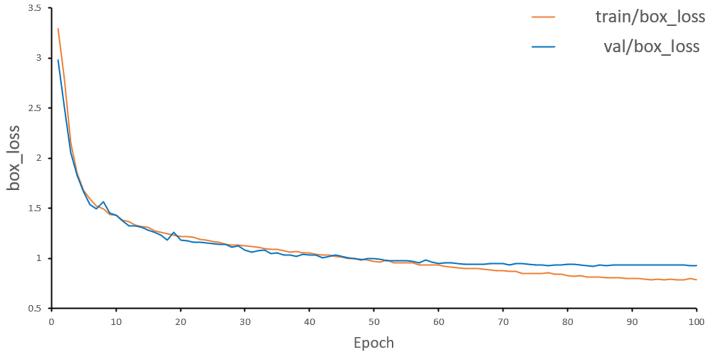

Figure 10. Box loss curve for model training.

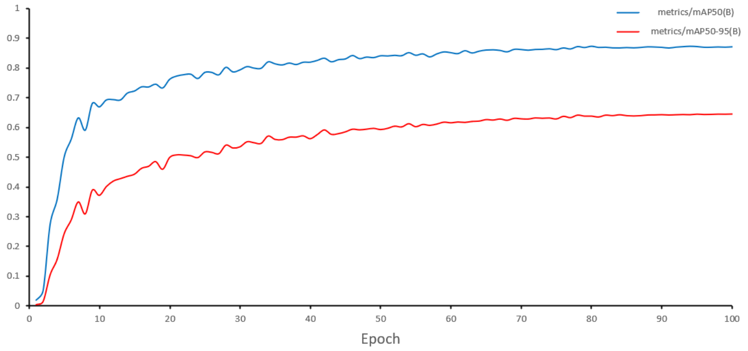

Figure 11. Curves of mAP50 and mAP50-95 during the training process.

Figure 11 shows the changes in mAP50 and mAP50-95 over the training epoch. In the initial training stage, the detection accuracy rapidly improves and gradually slows down as the training epoch progresses. At around the 40th epoch, the accuracy curve reaches the plateau period, which indicates that the model has reached the optimal accuracy benchmark.

4.4.1. Ablation Experiment

Ablation experiments were carried out to assess the impact of different modifications on the original network, ensuring consistency in the experimental environment for accuracy and performance. The results, which are presented in Table 2, showcase the effectiveness of these modifications. Improved models 1–3 integrate SPPEPSA, GSEAM, and GSCConv individually into the YOLOv8s network, while improved models 4–6 combine two of these enhancements in the original network.

Table 2. Ablation experiments with different improvement strategies.

Models	SPPEPSA	GSEAM	GSCConv	mAP50	mAP50-95	Params (M)	FLOPs (G)
YOLOv8s	×	×	×	85.77	61.52	11.17	28.8
YOLOv8s_1	√	×	×	86.15	62.57	12.16	29.6
YOLOv8s_2	×	√	×	86.23	63.05	10.17	26.9
YOLOv8s_3	×	×	√	86.14	63.07	11.25	28.9
YOLOv8s_4	√	√	×	86.42	63.38	11.16	27.7
YOLOv8s_5	√	×	√	86.39	63.50	12.24	29.7
YOLOv8s_6	×	√	√	86.52	63.77	10.25	27.0
CIHD-YOLO	√	√	√	86.76	64.19	11.24	27.8

Table 2 indicates that incorporating the SPPEPSA module into the YOLOv8 backbone network boosted mAP50 by 0.38% and mAP50-95 by 1.05%, mainly due to replacing the SPPF module. Despite a slight increase in Params and FLOPs, this adjustment enhanced the model's multi-scale detection ability.

Replacing the C2f setup with GSEAM in the YOLOv8s neck network reduced Params and FLOPs by 1M and 1.9G, respectively, while improving mAP50 by 0.46% and mAP50-95 by 1.53%. This enhancement better captured feature relationships for more accurate bounding boxes.

Integrating GSCConv into the neck network moderately raised mAP50-95 by 1.55%, with negligible changes in Params and FLOPs. Combining all three enhancements led to a 0.99% increase in mAP50, a substantial 2.67% rise in mAP50-95, almost no change in Params, and a 1.0G FLOP reduction compared to the base YOLOv8s network.

This study demonstrates that the proposed network enhances feature extraction, aiding in capturing intricate details in complex and low-light settings. This optimization notably benefits high-level IoU thresholds, thereby enhancing model performance in indoor scenarios. The improved YOLOv8s exhibit superior human detection efficiency in challenging environments without increasing model complexity while also reducing FLOPs.

4.4.2. Comparison Experiment

To evaluate the effectiveness of various state-of-the-art models in indoor human detection, we selected nine representative lightweight network models for comparison: YOLOv5s, YOLOv5sp6, YOLOv6s, YOLOv8s, YOLOv9s, YOLOv10s, CenterNet [44], EfficientDet [45], and RT-DETR-L [46]. In order to maintain consistency in evaluation, all models were trained and tested using the HCIE dataset proposed in this paper. In addition to the four performance indicators used in Table 2, a new indicator called model size has been added. Table 3 summarizes the comparative results of the tests.

The CIHD-YOLO model, with 11.24 million parameters and 27.8 GFLOPs, strikes a balance between complexity and performance, achieving an impressive 86.76% mAP50 accuracy. It outperforms other models in precision, especially at IoU thresholds of 50–95, with a mAP of 64.19%, surpassing similar models. This means that the model can detect as many targets as possible, avoiding missed detections as much as possible, even if the targets are small or in complex backgrounds.

Table 3. Comparative experiment of human detection results using different lightweight models.

Models	mAP50	mAP50-95	Model Size (MB)	Params (M)	FLOPs (G)
YOLOv5s	82.23	55.89	14.4	7.01	15.8
YOLOv5sp6	84.59	59.06	25.1	12.32	16.3
YOLOv6s	86.26	63.15	32.8	16.30	44.0
YOLOv8s	85.77	61.52	22.5	11.17	28.8
YOLOv9s	84.27	60.89	15.2	7.29	27.4
YOLOv10s	84.10	61.46	16.5	8.04	24.4
CenterNet	72.20	44.00	124.9	32.67	70.2
EfficientDet	59.90	38.40	15.1	3.87	5.2
RT-DETR-L	82.65	59.26	66.2	32.81	108.0
CIHD-YOLO	86.76	64.19	22.8	11.24	27.8

Among various comparison algorithms, YOLOv6s has the closest detection accuracy to CIHD-YOLO, but its computational requirements and model size are significantly higher. Although YOLOv9s and YOLOv10s models have faster computational performance on model size, params, and FLOPs, their mAP50s are 2.49% and 2.66% lower than that of CIHD-YOLO, respectively, and their mAP50-95s are 3.3% and 2.73% lower, respectively, indicating a certain gap in accuracy. As a transformer-based model, RT-DETR-L has strong detection capabilities, but in the reproduction results of this study, its accuracy in detecting indoor human bodies under low-light conditions is poor. Although the transformer model performs well in handling long-distance dependencies, it may be difficult for the model to accurately capture subtle features and relationships between target objects in low-light environments. This may lead to the poor detection accuracy of RT-DETR in human detection tasks in low-light indoor environments.

In summary, these results indicate that the enhanced CIHD-YOLO model not only ensures excellent detection accuracy but also maintains a low computational load, significantly enhancing the network's capabilities.

4.5. Visual Results

This study performed visual testing to compare the original YOLOv8s model with the proposed CIHD-YOLO model in detail. Random samples from the HCIE dataset were used, focusing on images in low-light conditions and with smaller detection targets, as depicted in Figures 12 and 13. The test sets in these figures include original real images, detection outcomes of YOLOv8s, and detection outcomes of CIHD-YOLO.

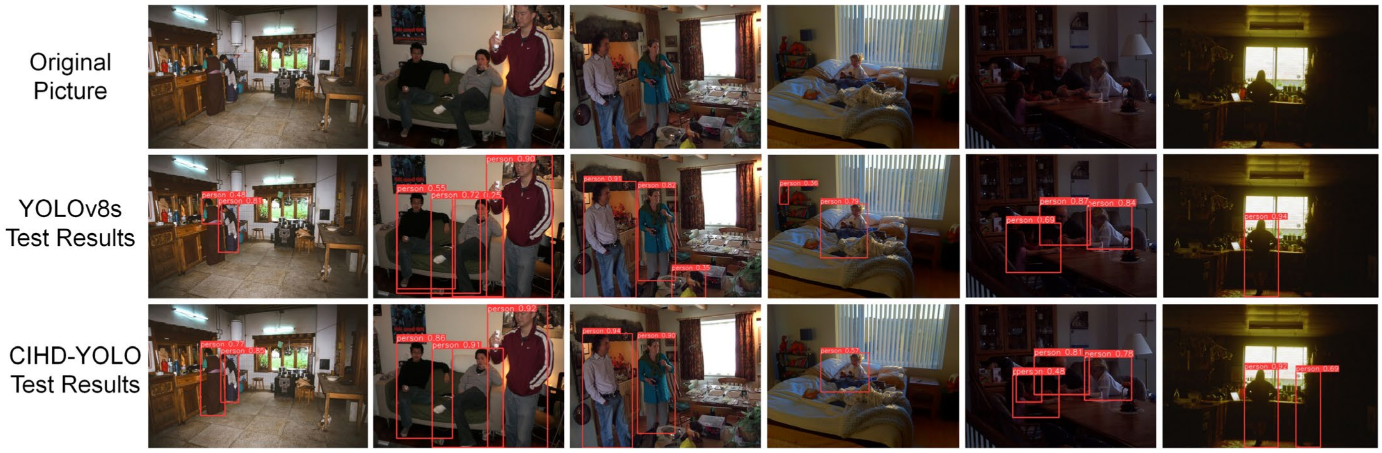

Figure 12. Test image set under low illumination.

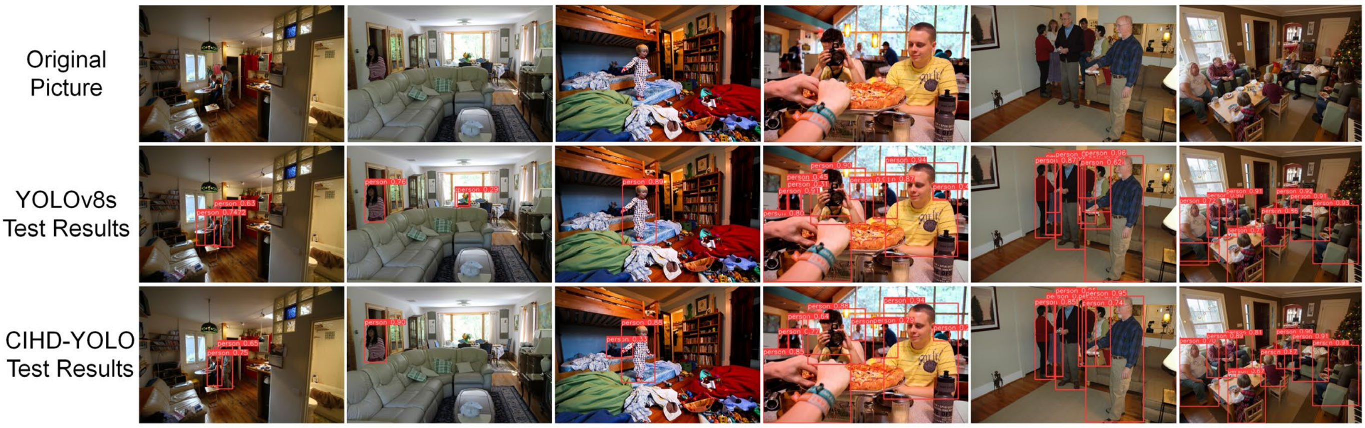

Figure 13. Test image set of small-scale human body.

In Figure 12, the original YOLOv8s model struggles to accurately detect human bodies in low-light indoor settings, often resulting in false positives. In such conditions, color features and complex backgrounds make human recognition challenging for the model. Due to unclear color features, similar-shaped objects are sometimes misidentified as targets. In contrast, CIHD-YOLO excels in precisely localizing human bodies, even in low-light scenarios. While YOLOv8s relies heavily on shape features when color features are vague, CIHD-YOLO's enhanced neck network effectively recognizes features, even when colors are indistinct, leading to precise bounding box localization. The improved detection accuracy and visual results in Figure 12 highlight CIHD-YOLO's superiority in object detection within poorly illuminated indoor environments.

In Figure 13, it is evident that the original model struggles to distinguish small-scale true positive targets effectively. Small objects occupying minimal pixels provide limited information for accurate feature extraction, posing a challenge for the model. The original YOLOv8s model's receptive field may not cover the entire small target, impeding its ability to capture comprehensive global and contextual information. On the contrary, CIHD-YOLO's pyramid pooling incorporates partial self-attention, facilitating the capture of long-range dependencies and establishing global feature correlations, which is especially beneficial for small objects. This approach enables effective fusion and attention to multi-scale features, enhancing detection even in crowded scenes. CIHD-YOLO excels in detecting small-scale human bodies in indoor environments, showcasing improved effectiveness compared to the original model.

In order to investigate how camera angle affects detection accuracy, some test set images were rotated by 30° for further detection, as shown in Figure 14. The results indicate that the rotation of the angle has a significant impact on the test results. The original YOLOv8s model experienced serious missed detections and may even be unable to detect a target from the rotated image. Although CIHD-YOLO also experienced some missed detections, the situation is far better than the original model.

The integration of the three enhanced networks proposed in this study notably boosts the model's performance. The CIHD-YOLO algorithm proves highly effective across diverse complex indoor environments, enhancing both human detection accuracy and precise bounding box localization significantly.

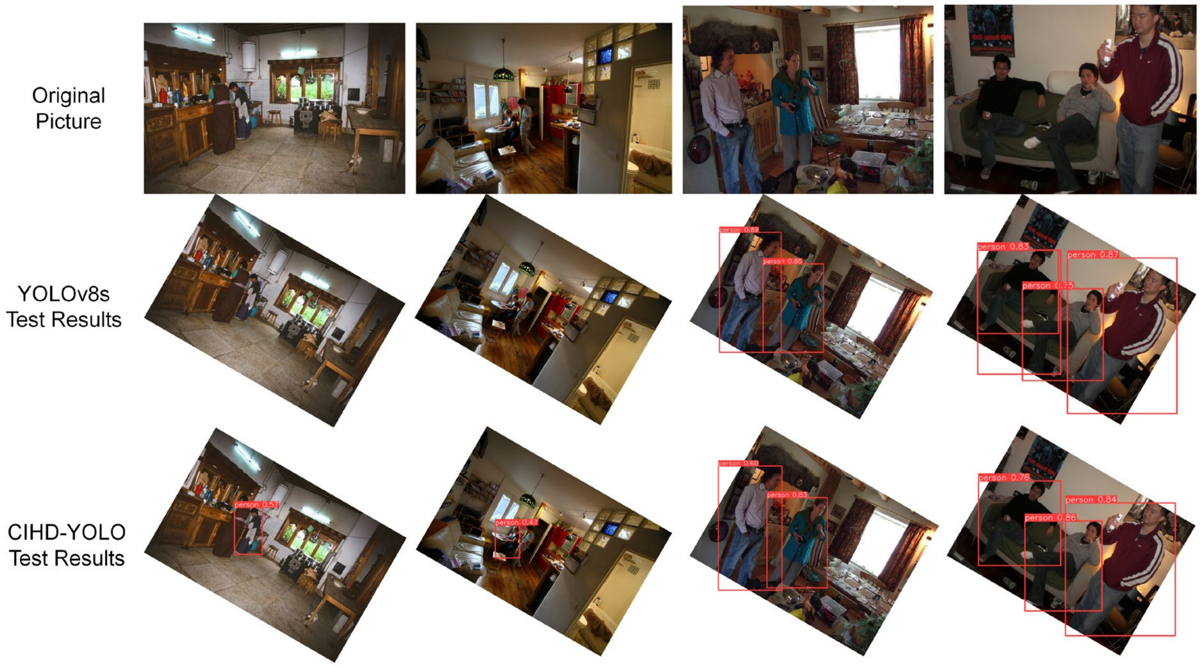

Figure 14. Test image set rotated by 30° angle.

5. Conclusions

The CIHD-YOLO method introduced in this study significantly enhances indoor human detection accuracy by addressing challenges faced by traditional YOLOv8 models in indoor settings. By incorporating specific enhancements like spatial pyramid pooling with effective partial self-attention, a generalized separated and enhancement aggregation network, and global spatial and channel reconstruction convolution, this model offers a balanced solution for low-light indoor environments and multi-scale human detection.

Moreover, a specialized dataset, the HCIE dataset, was curated for training and testing the indoor human detection model. Experimental results indicate that CIHD-YOLO, with a model size of 22.8MB, 11.24M parameters, and 27.8G FLOPs, achieves an mAP50 of 86.76% and an mAP50-95 of 64.19%, showcasing a 2.67% enhancement over the original algorithm on the HCIE dataset and superior detection efficiency compared to existing models.

CIHD-YOLO represents a significant advancement in real-time indoor human detection, with potential applications in various object detection domains like industrial production and smart buildings. Future enhancements could include integrating a low-light image enhancement network for clearer input images, exploring 3D visual data for occlusion handling and background reconstruction, optimizing parameter quantity and computational demands for improved real-time performance, and mitigating camera angle impacts on detection tasks for enhanced human detection capabilities.

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

Funding: This study was funded by Project No. 7430125 of the Cross Research Support Program at Huazhong University of Science and Technology, which is named Development and Application Research of a Big Data Platform for Active Health Monitoring of Elderly People at Home Based on Artificial Intelligence. General Facility Dimension Layout Detection and Evaluation Method in Research and Application of Key Technical Standards in Ergonomics for Aging-Friendly Community, a Key Special Project of the National Quality Infrastructure System. Funding Number: 2022YFF0607004-01.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: This study analyzed a combination of publicly available datasets and data collected by the authors. The available public datasets can be accessed at the following websites: COCO [https://cocodataset.org/#home] (accessed on 3 March 2024); VOC [http://host.robots.ox.ac.uk/pascal/VOC/] (accessed on 10 March 2024); and MPII [http://human-pose.mpi-inf.mpg.de/] (accessed on 12 March 2024). However, due to the sensitive nature of the research, the dataset collected by the authors is not publicly available.

Acknowledgments: The authors express their gratitude to Qinming Feng, Qiang Zhang and Shenglin Liu from Hubei Engineering Research Center of Medical Testing and Active Health Intelligent Equipment for their contributions to the software and technical support. They also extend sincere thanks to Yan Zhao, Jing Zhou, Yao Xiao and Jie Han from Rehabilitation Department of Hubei Provincial Hospital of TCM for their support with the data processing and resources.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Vijayan, R.; Mareeswari, V.; Pople, V. Public Social Distance Monitoring System Using Object Detection YOLO Deep Learning Algorithm. *SN Comput. Sci.* **2023**, *4*, 718.
2. Ganagavalli, K.; Santhi, V. YOLO-Based Anomaly Activity Detection System for Human Behavior Analysis and Crime Mitigation. *Signal Image Video Process.* **2024**, *18* (Suppl. 1), 417–427. [CrossRef]
3. Dalal, S.; Lilhore, U.K.; Sharma, N.; Arora, S.; Simaiya, S.; Ayadi, M.; Almujally, N.A.; Ksibi, A. Improving Smart Home Surveillance through YOLO Model with Transfer Learning and Quantization for Enhanced Accuracy and Efficiency. *PeerJ Comput. Sci.* **2024**, *10*, e1939. [CrossRef]
4. Zhi-Xian, Z.; Zhang, F. Image Real-Time Detection Using LSE-Yolo Neural Network in Artificial Intelligence-Based Internet of Things for Smart Cities and Smart Homes. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 1–8. [CrossRef]
5. Chua, S.N.D.; Chin, K.Y.R.; Lim, S.F.; Jain, P. Hand Gesture Control for Human–Computer Interaction with Deep Learning. *J. Electr. Eng. Technol.* **2022**, *17*, 1961–1970. [CrossRef]
6. Alruwaili, M.; Siddiqi, M.H.; Atta, M.N.; Arif, M. Deep Learning and Ubiquitous Systems for Disabled People Detection Using YOLO Models. *Comput. Hum. Behav.* **2024**, *154*, 108150. [CrossRef]
7. Inturi, A.R.; Manikandan, V.M.; Garrapally, V. A Novel Vision-Based Fall Detection Scheme Using Keypoints of Human Skeleton with Long Short-Term Memory Network. *Arab. J. Sci. Eng.* **2022**, *48*, 1143–1155. [CrossRef]
8. Lafuente-Arroyo, S.; Martín-Martín, P.; Iglesias-Iglesias, C.; Maldonado-Bascón, S.; Acevedo-Rodríguez, F.J. RGB Camera-Based Fallen Person Detection System Embedded on a Mobile Platform. *Expert Syst. Appl.* **2022**, *197*, 116715. [CrossRef]
9. Yu, Z.; Huang, H.; Chen, W.; Su, Y.; Liu, Y.; Wang, X. YOLO-FaceV2: A Scale and Occlusion Aware Face Detector. *Pattern Recognit.* **2024**, *155*, 110714. [CrossRef]
10. Zi, X.; Chaturvedi, K.; Braytee, A.; Li, J.; Prasad, M. Detecting Human Falls in Poor Lighting: Object Detection and Tracking Approach for Indoor Safety. *Electronics* **2023**, *12*, 1259. [CrossRef]
11. Zou, Z.; Chen, K.; Shi, Z.; Guo, Y.; Ye, J. Object Detection in 20 Years: A Survey. *Proc. IEEE* **2023**, *111*, 257–276. [CrossRef]
12. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2012**, *60*, 84–90. [CrossRef]
13. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
14. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Computer Vision—ECCV 2016*; Springer: Cham, Switzerland, 2016; Volume 9905, pp. 21–37.
15. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [CrossRef] [PubMed]
16. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-To-End Object Detection with Transformers. In *Computer Vision—ECCV 2020*; Springer: Cham, Switzerland, 2020; Volume 12346, pp. 213–229.
17. Kaur, R.; Singh, S. A Comprehensive Review of Object Detection with Deep Learning. *Digit. Signal Process.* **2022**, *132*, 103812. [CrossRef]
18. Lezzar, F.; Benmerzoug, D.; Kitouni, I. Camera-Based Fall Detection System for the Elderly with Occlusion Recognition. *Appl. Med. Inform.* **2020**, *42*, 169–179.
19. Aslan, M.F.; Durdu, A.; Sabanci, K.; Mutluer, M.A. CNN and HOG Based Comparison Study for Complete Occlusion Handling in Human Tracking. *Measurement* **2020**, *158*, 107704. [CrossRef]
20. Manakitsa, N.; Maraslidis, G.S.; Moysis, L.; Fragulis, G.F. A Review of Machine Learning and Deep Learning for Object Detection, Semantic Segmentation, and Human Action Recognition in Machine and Robotic Vision. *Technologies* **2024**, *12*, 15. [CrossRef]
21. Schwartz, W.R.; Kembhavi, A.; Harwood, D.; Davis, L.S. Human detection using partial least squares analysis. In Proceedings of the IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 November 2009; pp. 24–31.

22. Ahmed, I.; Ahmad, M.; Adnan, A.; Ahmad, A.; Khan, M. Person Detector for Different Overhead Views Using Machine Learning. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 2657–2668. [CrossRef]
23. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
24. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
25. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
26. Li, C.; Li, L.; Jiang, H.; Weng, K.; Geng, Y.; Li, L.; Ke, Z.; Li, Q.; Cheng, M.; Nie, W.; et al. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. *arXiv* **2022**, arXiv:2209.02976.
27. Wang, C.-Y.; Bochkovskiy, A.; Liao, H.-Y.M. 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, Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
28. Wang, C.-Y.; Yeh, I.-H.; Liao, H.-Y.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
29. Wang, A.; Chen, H.; Liu, L.; Chen, K.; Lin, Z.; Han, J.; Ding, G. YOLOv10: Real-Time End-to-End Object Detection. *arXiv* **2024**, arXiv:2405.14458.
30. Fu, C.-Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. DSSD: Deconvolutional Single Shot Detector. *arXiv* **2017**, arXiv:1701.06659.
31. Aoki, Y.; Kobayashi, N.; Okoshi, T.; Nakazawa, J. Demo: Image-Based Indoor Localization Using Object Detection and LSTM. In Proceedings of the 22nd Annual International Conference on Mobile Systems, Applications and Services, Tokyo, Japan, 3–7 June 2024; pp. 596–597.
32. Safaldin, M.; Zaghden, N.; Mejdoub, M. An Improved YOLOv8 to Detect Moving Objects. *IEEE Access* **2024**, *12*, 59782–59806. [CrossRef]
33. Han, L.; Feng, H.; Liu, G.; Zhang, A.; Han, T. A Real-Time Intelligent Monitoring Method for Indoor Evacuee Distribution Based on Deep Learning and Spatial Division. *J. Build. Eng.* **2024**, *92*, 109764. [CrossRef]
34. Kan, X.; Zhu, S.; Zhang, Y.; Qian, C. A Lightweight Human Fall Detection Network. *Sensors* **2024**, *24*, 922. [CrossRef] [PubMed]
35. Cao, Y.; Li, C.; Peng, Y.; Ru, H. MCS-YOLO: A Multiscale Object Detection Method for Autonomous Driving Road Environment Recognition. *IEEE Access* **2023**, *11*, 22342–22354. [CrossRef]
36. Li, J.; Wen, Y.; He, L. SCConv: Spatial and Channel Reconstruction Convolution for Feature Redundancy. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 6153–6162.
37. Li, H.; Li, J.; Wei, H.; Liu, Z.; Zhan, Z.; Ren, Q. Slim-Neck by GSConv: A Better Design Paradigm of Detector Architectures for Autonomous Vehicles. *J. Real-Time Image Process.* **2024**, *21*, 62. [CrossRef]
38. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
39. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef]
40. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.
41. Chollet, F. Xception: Deep Learning with Depthwise Separable Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
42. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Proceedings of the Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; Volume 8693, pp. 740–755.
43. Everingham, M.; van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *Int. J. Comput. Vis.* **2009**, *88*, 303–338. [CrossRef]
44. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6569–6578.
45. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 10781–10790.
46. Zhao, Y.; Lv, W.; Xu, S.; Wei, J.; Wang, G.; Dang, Q.; Liu, Y.; Chen, J. DETRs Beat YOLOs on Real-Time Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 17–21 June 2024; pp. 16965–16974.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.