Heliyon 10 (2024) e32678

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Enhanced object detection in pediatric bronchoscopy images using YOLO-based algorithms with CBAM attention mechanism

Jianqi Yan [a,c], Yifan Zeng [c], Junhong Lin [e], Zhiyuan Pei [a,c], Jinrui Fan [d], Chuanyu Fang [c], Yong Cai [b,*]

[a] Faculty of Innovation Engineering, Macau University of Science and Technology, Avenida Wai Long, Taipa, 999078, Macau
[b] Advanced Institute of Natural Sciences, Beijing Normal University, Jinfeng Road, Xiangzhou District, Zhuhai, 519087, China
[c] R&D Department, Quanbao Technologies Co. Ltd, Hagongda Road, Xiangzhou District, Zhuhai, 519087, China
[d] General Surgery, Zhuhai People's Hospital, Kangning Road, Xiangzhou District, Zhuhai, 519000, China
[e] Pediatric Respiratory Department, M-Healtcare, Zhujiang New Town Clinic 2/F, No. 11 Xiancun Road, Tianhe District, Guangzhou, 510623, China

ARTICLE INFO

Keywords:
Bronchoscopy
Object detection
CBAM attention mechanism
Deep learning
Medical imaging

ABSTRACT

Background and Objective: Bronchoscopy is a widely used diagnostic and therapeutic procedure for respiratory disorders such as infections and tumors. However, visualizing the bronchial tubes and lungs can be challenging due to the presence of various objects, such as mucus, blood, and foreign bodies. Accurately identifying the anatomical location of the bronchi can be quite challenging, especially for medical professionals who are new to the field. Deep learning-based object detection algorithms can assist doctors in analyzing images or videos of the bronchial tubes to identify key features such as the epiglottis, vocal cord, and right basal bronchus. This study aims to improve the accuracy of object detection in bronchoscopy images by integrating a YOLO-based algorithm with a CBAM attention mechanism.

Methods: The CBAM attention module is implemented in the YOLO-V7 and YOLO-V8 object detection models to improve their object identification and classification capabilities in bronchoscopy images. Various YOLO-based object detection algorithms, such as YOLO-V5, YOLO-V7, and YOLO-V8 are compared on this dataset. Experiments are conducted to evaluate the performance of the proposed method and different algorithms.

Results: The proposed method significantly improves the accuracy and reliability of object detection for bronchoscopy images. This approach demonstrates the potential benefits of incorporating an attention mechanism in medical imaging and the benefits of utilizing object detection algorithms in bronchoscopy. In the experiments, the YOLO-V8-based model achieved a mean Average Precision (mAP) of 87.09% on the given dataset with an Intersection over Union (IoU) threshold of 0.5. After incorporating the Convolutional Block Attention Module (CBAM) into the YOLO-V8 architecture, the proposed method achieved a significantly enhanced $mAP_{0.5}$ and $mAP_{0.5:0.95}$ of 88.27% and 55.39%, respectively.

Conclusions: Our findings indicate that by incorporating a CBAM attention mechanism with a YOLO-based algorithm, there is a noticeable improvement in object detection performance in bronchoscopy images. This study provides valuable insights into enhancing the performance of attention mechanisms for object detection in medical imaging.

* Corresponding author.
E-mail addresses: yanjianqi.top@gmail.com (J. Yan), caiyong@bnu.edu.cn (Y. Cai).

https://doi.org/10.1016/j.heliyon.2024.e32678
Received 6 June 2023; Received in revised form 5 June 2024; Accepted 6 June 2024

Available online 17 June 2024

1. Introduction

The first clinically applicable flexible fiber bronchoscope, developed by [35], rapidly became an essential diagnostic and therapeutic tool in adult respiratory medicine. The development of flexible bronchoscopy for children was initially limited by material constraints. A report on its diagnostic use in children was first published in 1978 [44]. Flexible bronchoscopy has become an essential tool in pediatric respiratory medicine due to technological advancements [12] over the past decades. Currently, flexible bronchoscopy is primarily utilized in the diagnosis of pediatric respiratory diseases, conducting mucosal biopsies, collecting secretions, assisting with tracheal intubation, and performing various interventional surgeries [6].

A flexible bronchoscopy examination is a fundamental procedure for diagnosing and treating bronchial conditions. This examination technique is quite complex and can have some risks associated with it. It is important for operators to receive thorough training and have a deep understanding of the respiratory tract and its common variations [31]. However, the available evidence on the necessary training duration for a skilled pediatric flexible bronchoscopy operator to achieve competence is limited [25]. The trachea and bronchus have a maze-like structure that can make it challenging to navigate during a bronchoscopy examination. This can result in the possibility of missing or repeating examination of certain anatomical parts. Even with adequate training, there is a significant difference in the accuracy of junior resident doctors compared to senior resident doctors ($p = 0.001$) and experienced bronchoscopy specialists ($p < 0.001$) when it comes to identifying anatomy during flexible bronchoscopy examinations [16]. The risk of complications is higher for inexperienced bronchoscopy examiners [23].

With the rapid advancement of artificial intelligence (AI) technology in the medical field, clinicians are now able to enhance their diagnostic capabilities and improve efficiency through its integration with medical imaging, pathology, and endoscopy. [46] utilized deep learning to analyze bronchoscope videos. Their research focused on developing an AI system capable of identifying bronchial carinas and determining the precise anatomical locations of the left and right main bronchi. [22] utilized a dataset of 775 laryngoscope and bronchoscope videos to train three convolutional neural networks (CNNs). This enabled the AI system to accurately distinguish the vocal cords and trachea from videos in real time. [18] utilized 28,441 qualified photos from 324 bronchoscope examination videos to train a CNN. The trained AI achieved a recognition accuracy of 54.30% for static images. Recognition of the lumen anatomy by senior and junior doctors was significantly higher when assisted by AI compared to doctors without AI assistance ($p < 0.001$). Considering the distinct requirements of pediatric bronchoscopy, it is clear that adult bronchoscope operators lack the expertise required to properly treat children, who cannot be treated as miniature adults [5]. As a result, it is important to develop separate CNN models for recognizing children's bronchoscope videos and adult videos in the field of AI model training.

Deep learning has been extensively studied in various medical domains due to its effectiveness as a tool for medical image analysis. This includes image segmentation in lung nodules [45], image registration in computerized tomography (CT) [7], and object detection in bronchoscopy videos [22], among other tasks. These techniques, which use artificial neural networks for learning and adaptation, have proven to be highly effective in a wide range of medical applications. Computer vision, based on deep learning, is currently a popular research field with a growing number of applications in the area of bronchoscopy navigation [38,3,33]. Bronchoscopy navigation is a specialized application of computer vision that provides important visual feedback to assist in endoscopic surgery. In order to facilitate accurate and effective endoscopic surgery, it is essential to incorporate deep learning with endoscopic vision.

Endoscopic physicians typically require specialized training and extensive experience to perform bronchoscopies accurately. There is a significant variation in bronchoscopy proficiency across different regions and hospitals, which can be attributed to variations in physicians' expertise. Furthermore, in many practical situations, endoscopic physicians face a significant challenge in identifying crucial bronchial features when they are only present in a few frames among a large array of images. Several object detection algorithms were used in our study to improve the efficiency and accuracy of bronchoscopy procedures while also reducing the workload of physicians.

In this study, a novel approach is proposed to enhance the effectiveness of object detection algorithms in bronchoscopy imaging. We conducted a benchmark of various models, including YOLO-V3 [29], YOLO-V4 [2], YOLO-V5 [15], YOLO-V7 [40], and YOLO-V8 [14] on the bronchoscopy dataset (referenced in Section 2). Several architectures of these models are thoroughly assessed and compared. This comparative analysis enables us to evaluate the variations and biases in each model, aiding us in selecting the most appropriate architecture. In order to improve YOLO-V7 and YOLO-V8, we have integrated the Convolutional Block Attention Module (CBAM) [43] into different parts of the model. After analyzing the performance of various components integrated with CBAM in YOLO-V8 and YOLO-V7, we have concluded that applying CBAM to the head and backbone of YOLO-V8-large yields the most optimal performance.

Furthermore, we conduct a comparative analysis of various data augmentation methods using the bronchoscopy dataset. In particular, we evaluate two significant data augmentation techniques, image scaling and mosaic, by assessing their mean Average Precision (mAP) on three benchmarks: YOLO-V8, YOLO-V7, and YOLO-V5. The intensity of each data augmentation method is controlled by setting different probability thresholds. We examine the different levels of intensity of these data augmentation methods and provide valuable insights into the results, thereby establishing a solid foundation for future research. The experimental results demonstrate that mosaic and scale data augmentation methods improve the performance of YOLO-based object detection models.

Our innovations can be summarized as follows: The main innovation of our study lies in its significant practical engineering value. It can be used as a helpful tool during surgical procedures to identify the anatomical positions of the bronchi. It can also be used as a medical training tool to improve the operational skills of newly hired doctors. This technology is crucial for achieving anatomical navigation in endoscopy. Our second innovation is the high accuracy of our method. Our method outperforms the work of [22] in terms of accuracy, the range of categories considered, and recognition speed. Our method achieves impressive results, accurately

Table 1
Number of instances in each category in the bronchoscopy dataset.

Category	Number of Instances
epiglottis	292
vocal cord	235
right basal bronchus	356
left main bronchus	302
left upper lobar bronchus	314
left division bronchus	289
left lingular bronchus	413
left lower bronchus	281
left superior segment	419
left basal bronchus	346
trachea	420
carina	575
right main bronchus	330
intermediate bronchus	485
right upper lobar bronchus	416
right middle lobar bronchus	397
right lower lobar bronchus	368
right superior segment bronchus	422

identifying 18 categories at a rapid speed of almost 100 frames per second (FPS). The YOLO-V8 benchmark model achieves an average precision of 87.09% ($mAP_{0.5}$) on the test set. The third key innovation is that the proposed YOLO-V8 CBAM model demonstrates a 1.18% improvement over the baseline in terms of $mAP_{0.5}$, demonstrating the effectiveness of our method for endoscopic-type data. In the future, our enhanced object detection method can easily and accurately be applied to different types of endoscopes, such as gastrointestinal endoscopes, cystoscopes, and others.

Similarly, we performed experiments on different YOLO-based benchmarks and compared two distinct data augmentation methods. These experiments provide valuable insights into the field of endoscopic navigation. In addition, to highlight our contribution to the community, we have made our model inference functionality accessible as open-source on a website (https://huggingface.co/spaces/EasonYan/yolo-bronchoscopy), enabling individuals to utilize our model for real-time inference online.

The structure of the rest of this paper is as follows: In Section 2, we provide a description of the bronchoscopy dataset and discuss the data preprocessing methods used in our study. In Section 3, we provide a detailed description of the methodology employed in our experiments. This includes the problem formulation (Section 3.1), the object detection algorithms (Section 3.2), the attention modules (Section 3.3), and the evaluation metrics (Section 3.4). The results of our experiments are presented in Section 4. In Section 5, we discuss the implications of our findings, propose potential directions for future research, and provide a summary of the conclusions and main contributions of this paper.

2. Bronchoscopy dataset

The dataset for bronchoscopy was collected from a database of a pediatric hospital. The dataset consists of 2419 bronchoscopy images, which are screenshots of bronchoscopic videos without any accompanying labels. Every patient willingly signed an informed consent form, and all data were carefully de-identified to ensure the utmost protection of patient privacy. The hospital has acquired consent to utilize bronchoscopy data for research and publication purposes. The images are classified into different categories and labeled manually. Labeling images is the initial and most critical step in an object detection task. To ensure the accuracy, completeness, and availability of the manual labeling process, each bronchoscopy image is annotated by both a medical specialist and a programmer at the same time. A medical specialist, based on experience and intuitive observation, provides information about one or more accurate categories and their respective locations. Subsequently, a bronchoscopy image is overlaid with one or more bounding boxes, and the appropriate categories are chosen using the labeling tool (known as LabelImg [36]).

There are 18 classes in the bronchoscopy dataset, which include: `epiglottis`, `vocal cord`, `trachea`, `carina`, `right main bronchus`, `intermediate bronchus`, `right upper lobar bronchus`, `right middle lobar bronchus`, `right lower lobar bronchus`, `right superior segment bronchus`, `right basal bronchus`, `left main bronchus`, `left upper lobar bronchus`, `left division bronchus`, `left lingular bronchus`, `left lower bronchus`, `left superior segment` and `left basal bronchus`. The specific number of instances in each category is provided in Table 1. In Table 1, this dataset does not exhibit data imbalance or small sample problems. Therefore, there is no need to utilize a specialized deep neural network for object detection to address the issues of data imbalance and small sample size.

2.1. Data preprocessing

Our dataset contains multiple redundant areas in an original bronchoscopy image acquired from a video bronchoscopy machine (Olympus BF-XP260F), such as textual areas and black backgrounds. In theory, when the image resolution increases, the object

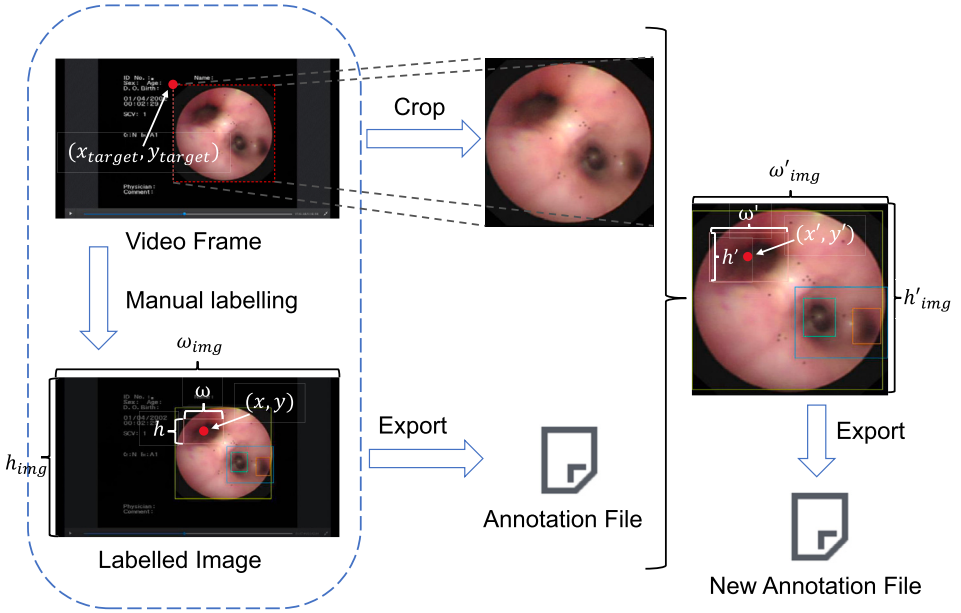

Fig. 1. Illustration of the cropping process and the coordinate transformation process to eliminate redundant areas in the original bronchoscopy image.

detection algorithm's time and space complexities also increase. Redundant areas in an image can increase the computational, training, and inference times of the object detection model.

To address the issue of redundant areas, we crop the original image to eliminate them. In addition, we ensure that the original image information and annotation coordinates are maintained. Since bronchoscopes come in different models with varying imaging sizes, it is necessary to crop the original image to match the actual imaging area in the bronchoscopy image. First, we manually identify the coordinates of the circular imaging area in the image. Then, the original image is cropped to a rectangle and resized to a uniform size (e.g., 320×320).

In this study, we utilized YOLO-based object detection algorithms to identify and categorize objects in the bronchoscopy dataset. The standardized YOLO annotation format, consisting of an integer and four numbers between 0 and 1 expressed to four decimal places, is essential to maintain consistency in the annotated data. The first integer in this format corresponds to the ID number of each category, while the last four numbers represent the center coordinates (x, y) of the bounding box, as well as the width w and height h of the bounding box after normalization. A coordinate transformation method is developed that adjusts each label to its correct position after cropping the image. Fig. 1 illustrates the cropping process and the coordinate transformation process.

Suppose the coordinates of the upper left corner of the rectangle imaging area are $(x_{\text{target}}, y_{\text{target}})$, and the width and height of the original image are w_{img} and h_{img}, respectively. In addition, it is assumed that the dimensions of the transformed image are represented by w'_{img} for width and h'_{img} for height. The transformed coordinate (x', y'), and the converted width w' and height h' of a bounding box can be represented as follows:

$$\begin{cases} x' = (x - \dfrac{x_{\text{target}}}{w_{\text{img}}}) \times \dfrac{w'_{\text{img}}}{w_{\text{img}}} \\ y' = (y - \dfrac{y_{\text{target}}}{h_{\text{img}}}) \times \dfrac{h'_{\text{img}}}{h_{\text{img}}} \\ w' = w \times \dfrac{w'_{\text{img}}}{w_{\text{img}}} \\ h' = h \times \dfrac{h'_{\text{img}}}{h_{\text{img}}} \end{cases} \tag{1}$$

In Fig. 1, a bronchoscopy image shows various bounding boxes labeled in a video frame, and the results are then exported to an annotation file. Afterwards, the rectangular imaging area is cropped from the original video frame and utilized as input for our coordinate transformation method to generate a new annotation file.

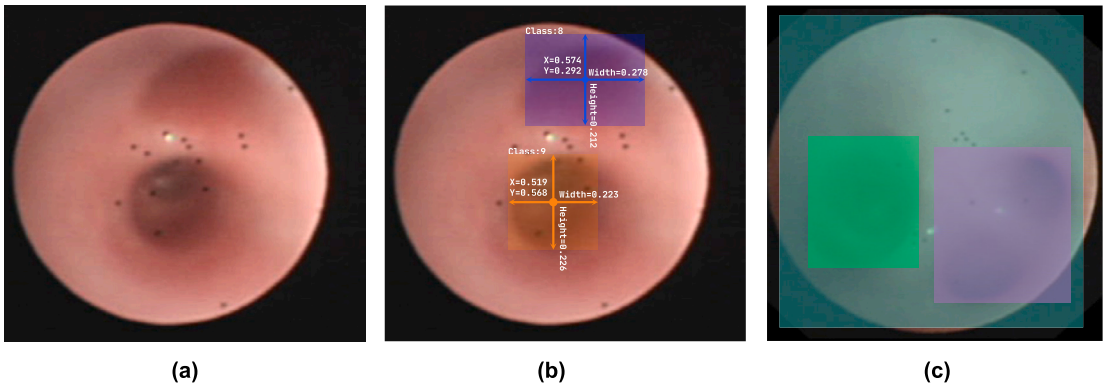

Fig. 2. (a) The cropped image. (b) The class labels and bounding boxes information in the cropped image. (c) Two small bounding boxes enclosed within a larger bounding box in the cropped image.

3. Methodology

3.1. Problem formulation

To precisely categorize and locate the position of the bronchoscope during its operation, a series of video frames captured by the bronchoscope's camera will serve as input for the object detection model. The object detection problem in this work can be defined as the bounding boxes regression problem and the multi-label classification problem. Each labeled position in the image is associated with a bounding box and a class label. Fig. 2 (a) shows the original cropped image in its original form. There are two bounding boxes in Fig. 2 (b). The position of a single bounding box is determined by four values: X, Y, W, and H. The variables X and Y represent the coordinates of the center point of the bounding box, while W and H represent the lengths in the X- and Y-axis directions, respectively. These four values are obtained by normalizing the length and width of the image so that their range falls between 0 and 1. Additionally, each bounding box contains category information, represented by an integer number in this case.

In this study, the one-hot encoding method is employed to convert 18 classes into a feature vector. The objective of object detection is to compute four numerical values for each bounding box in a single image and generate a final bounding box using Non-Maximum Suppression (NMS) [30]. During the training process, the object detection algorithm will predict a category for each bounding box. Fig. 2 (c) depicts two small bounding boxes enclosed within a larger bounding box in a bronchoscopy image. The size of the bounding boxes varies, with some being larger to represent the camera's location on the bronchoscope, and others being smaller to indicate the predicted category and position. Training samples that have a small bounding box enclosed by a larger bounding box can be quite challenging in object detection. In this study, we discovered that by combining YOLO-based object detection algorithms with appropriate data augmentation methods, the scaling issue mentioned earlier can be effectively resolved. These algorithms are introduced in Section 3.2.

3.2. Object detection algorithms

You Only Look Once (YOLO) is a CNN-based object detection algorithm that approaches object detection as a regression problem. Its objective is to spatially separate the bounding boxes and determine the class probabilities. In recent years, numerous object detection algorithms based on YOLO have been proposed, such as YOLO-V1 [27], YOLO-V2 [28], YOLO-V3 [29], YOLO-V4 [2], YOLO-V5 [15], YOLO-V7 [40], YOLOX [8], YOLOR [42], YOLO-V8 [14], etc. Recently, various object detection algorithms, including YOLO-based ones, have been utilized to address a range of practical issues. For instance, [32] employed YOLO-V5 to detect fundus lesions, while [9] used a YOLO-based model to detect and localize lung nodules. [1] utilized YOLO-V3 to detect and classify breast masses. In this study, we analyze and compare the object detection performance and inference speed of various YOLO-based object detection algorithms. In the following section, we will introduce these algorithms and explain why they are suitable for our purposes.

YOLO-V1 [27] is based on the one-state object detection mode, which requires only a single pass through the model and simultaneously predicts all bounding boxes with classes. Building upon YOLO-V1, YOLO-V2 focuses on improving recall and localization to enhance object detection performance. The YOLO-V2 model [28] utilizes anchor boxes to predict bounding boxes. The sizes of these anchor boxes are determined by clustering the bounding boxes of instances in the training set using K-means [21]. The Average Precision (AP) is significantly improved in YOLO-V3 [29] compared to YOLO-V2. YOLO-V3 utilizes logistic regression to estimate the objectness score for each bounding box. This model utilizes the Darknet-53 architecture [26] as the backbone network. In addition, YOLO-V3 is capable of predicting bounding boxes at three distinct scales. Therefore, for our experiments, we selected YOLO-V3 as it has demonstrated superior performance compared to YOLO-V1 and YOLO-V2, which are no longer included in our study.

The network architecture of YOLO-V4 [2] combines several components to achieve its functionality. These include the Cross-Stage Partial (CSP) Darknet53 architecture [41] as the backbone, the additional Spatial Pyramid Pooling (SPP) module [13], the PANet path-aggregation [19] as the neck and the YOLO-V3 anchor-based architecture as the head. The backbone is composed of several convolution blocks featuring the CSP bottleneck. The design of CSP can help address the issue of duplicate gradient information

in network optimization, resulting in a decrease in inference computations. At the end of the backbone, there is an SPP layer that can be used to increase the receptive field. The neck part of YOLO-V4 uses PANet path-aggregation to combine parameters from different backbone levels. The anchor-based approach utilizes anchor boxes for the prediction of bounding boxes. Deep learning techniques are also extensively used in the training of YOLO-V4. In summary, when comparing YOLO-V3 to YOLO-V4, there are significant improvement in the network architecture that result in improved performance and faster processing. These improvements are achieved by implementing various training techniques.

The network architecture of YOLO-V5 [15] closely resembles that of YOLO-V4. It is implemented in PyTorch [24], which is a popular deep learning framework. The architecture consists of three main components: the backbone for extracting image features, the neck for collecting feature maps, and the head for predicting bounding boxes and classes, similar to YOLO-V4. YOLO-V5 proposes several network architectures with different input sizes, such as YOLO-V5-small, YOLO-V5-middle, YOLO-V5-large, and others, to accommodate various application scenarios. YOLO-V5 has the advantage of being deployable on a wide range of hardware platforms.

YOLO-V7 [40] builds on the foundation laid by its predecessors with innovative architectural improvement. It incorporates methods like compound model scaling, a trainable bag of freebies, re-parameterization, and specialized loss functions to improve its learning capability and detection accuracy. YOLO-V7's core architecture is based on the Efficient Layer Aggregation Network (ELAN), which aims to design an efficient network by controlling the shortest and longest gradients, allowing deeper networks to converge and learn effectively. The architecture is further improved by the introduction of the Extended Efficient Layer Aggregation Network (E-ELAN), which serves as the computational block in the YOLO-V7 backbone. YOLO-V8 [14] is the latest state-of-the-art model in the YOLO series, which is designed for object detection, image classification, and instance segmentation tasks. It is an integration of most improvements from previous YOLO versions. YOLO-V8 is significant as an out-of-the-box framework for different kinds of image tasks. The backbone of the YOLO-V8 is based on E-ELAN, which is a modified version of the CSPDarknet53 architecture and the speed of training and inference is improved by the efficient PyTorch implementation.

3.3. Attention modules

There have been recent reports suggesting that object detection performance can be improved by attention modules [37,4,20,39]. The fundamental idea behind attention modules is to allow the model to focus on essential information in the image and the relationships between various pieces of information. The attention module was initially developed for constructing models for natural language processing, and it has gained popularity in image processing for its exceptional performance. In this paper, we examine the attention module of the Convolutional Block Attention Module (CBAM) based on channel and spatial axes.

CBAM is a form of attention mechanism that can be used to CNNs to improve the performance of image recognition. The CBAM architecture includes two consecutive branches, one for channel attention and the other for spatial attention. The channel attention branch utilizes a fully-connected layer, pooling layers, and a sigmoid activation function to evaluate the significance of each channel in the input feature maps. The output of the channel attention branch consists of weights that indicate the relative importance of each channel. The weights are then multiplied element-wise with the input feature maps to generate the new input feature maps for the spatial attention branch. The spatial attention branch utilizes a convolutional layer, pooling layers, and a sigmoid activation function to assess the significance of each spatial location within the feature maps. The spatial attention branch output is multiplied element-wise with the input feature maps to generate the output feature maps.

Mathematically, the channel attention branch can be expressed as follows:

$$M_c(F) = \sigma(FC(AvgPool(F)) + FC(MaxPool(F))), \tag{2}$$

where $F \in \mathbb{R}^{C\times H\times W}$ represents the input feature maps, C represents the number of channels, and H and W represent the spatial dimensions, $AvgPool$ and $MaxPool$ represent two types of pooling operations, FC is the fully-connected layer, and σ is the sigmoid activation function. The spatial attention branch can be represented as follows:

$$M_s = \sigma(Conv([AvgPool(F'); MaxPool(F')]), \tag{3}$$

where $Conv$ represents the convolutional layer, and the concatenated pooling output serves as the input to this layer. Finally, the output feature maps are obtained as follows:

$$\begin{cases} F' = M_c(F) \otimes F, \\ F'' = M_s(F') \otimes F'. \end{cases} \tag{4}$$

We incorporate CBAM into both the backbone and head of YOLO-V8. Integrating CBAM into the backbone and head significantly improves the ability of YOLO-V8 to extract features from images and make the most of the extracted features. In this experiment, we discovered that incorporating CBAM into the final layer before the detector module in the head leads to enhanced detection performance. The improved architecture achieved by incorporating CBAM into YOLO-V7 and YOLO-V8 are shown in Fig. 3 (a) and (b), respectively. In YOLO-V7 experiment, the CBAM integration is implemented based on a standard version of YOLO-V7. A CBAM block is placed after each ELAN block in the backbone part. Three CBAM blocks are used after the three scales of feature extraction in the head part. The CBAM block we designed includes three convolutional layers that handle the feature maps prior to the CBAM operation. YOLO-V8 employs a unified architecture skeleton. For each YOLO-V8 configuration, the CBAM block is placed after the first C2f and SPPF blocks in the backbone. Similar to the YOLO-V7 case, CBAM blocks in the head are placed before the detection

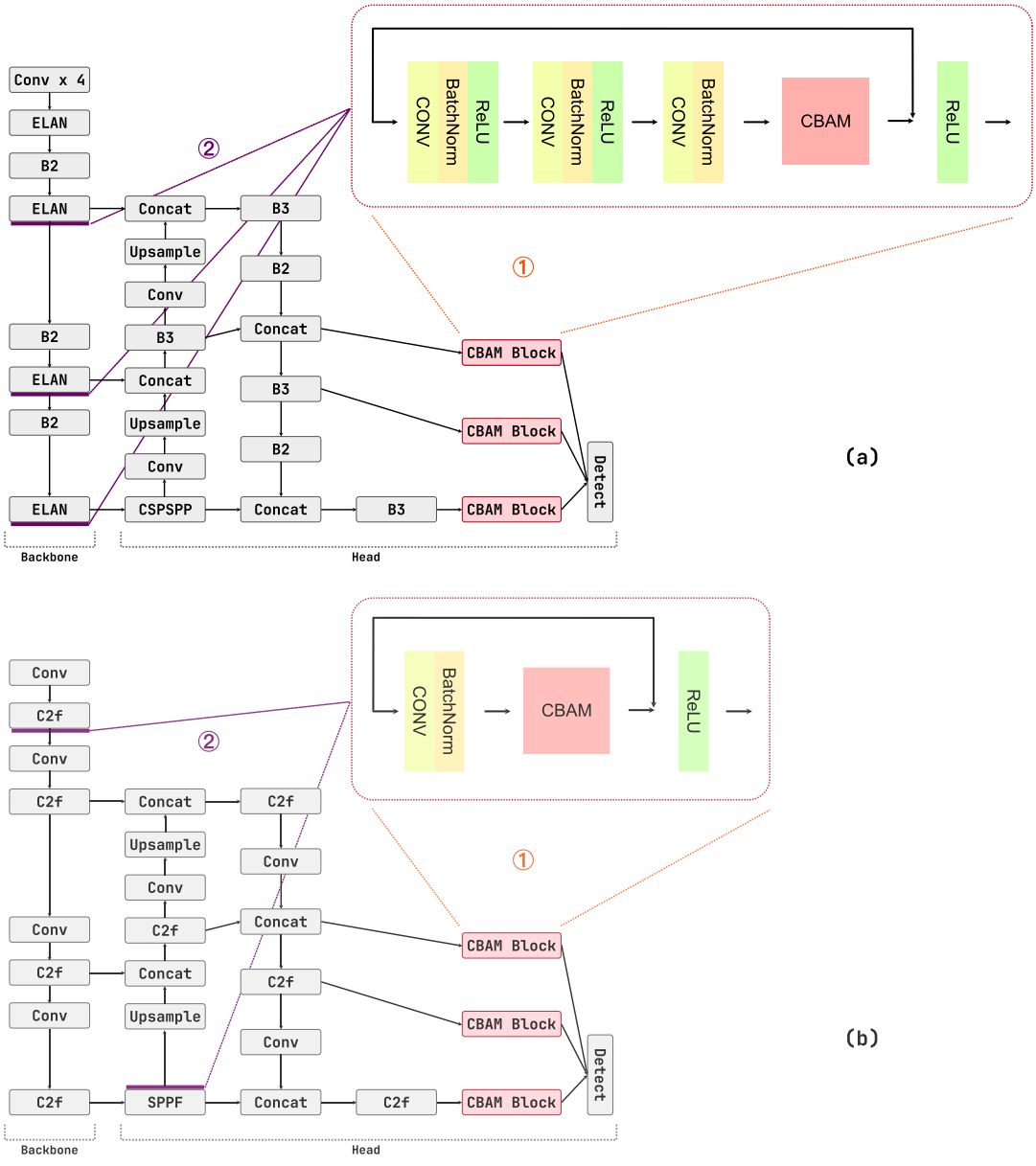

Fig. 3. (a) The improved YOLO-V7 architecture with CBAM. (b) The improved YOLO-V8 architecture with CBAM.

block. We reduce the convolutional layers in the CBAM block for the YOLO-V8. The attention operation in the CBAM block is used to ensure that the model can prioritize important information.

3.4. Evaluation metrics

In all experiments, precision and recall are commonly used as the first step to evaluate performance. Precision is a measure of the model's true positive predictions relative to all positive predictions. Precision is calculated as follows:

$$precision = \frac{TP}{TP + FP}, \tag{5}$$

where TP is the true positive and FP is the false positive. Recall is a metric that quantifies the accuracy of the model by determining the proportion of positive instances that were correctly predicted. Recall is formulated as follows:

$$recall = \frac{TP}{TP + FN}, \tag{6}$$

where FN is the false negative.

Table 2
Table of the benchmark results with different architectures of each YOLO-based object detection model.

Algorithm	Architecture	Precision (%)	Recall (%)	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)	FPS
YOLO-V8	YOLO-V8-tiny	82.66	83.56	86.31	53.36	127.22
	YOLO-V8-small	83.54	83.26	86.94	53.53	127.86
	YOLO-V8-middle	83.34	83.05	86.22	53.46	100.35
	YOLO-V8-large	**85.26**	81.83	**87.09**	**54.30**	82.93
	YOLO-V8-extra-large	85.09	83.34	86.45	54.10	82.36
YOLO-V7	YOLO-V7-base	82.18	**84.17**	85.67	49.36	91.01
	YOLO-V7-tiny	83.73	83.14	85.78	49.73	126.22
	YOLO-V7-extra-large	81.47	82.76	84.88	49.77	80.50
YOLO-V5	YOLO-V5-tiny	84.90	77.18	82.77	45.76	181.82
	YOLO-V5-small	84.53	79.45	83.89	46.56	185.19
	YOLO-V5-middle	85.25	78.66	83.73	46.71	135.14
	YOLO-V5-large	84.05	79.94	84.06	46.45	102.04
	YOLO-V5-extra-large	83.28	79.81	84.72	46.99	81.30
YOLO-V4	YOLO-V4-base	–	–	80.35	37.94	95.80
	YOLO-V4-tiny	–	–	80.61	37.06	**479.00**
YOLO-V3	YOLO-V3-base	–	–	80.98	38.21	159.67

Mean Average Precision (mAP) is a widely used metric for assessing the performance of a machine learning model in object detection tasks. The calculation of mAP involves determining the Average Precision (AP) for each class. This evaluation metric quantifies the precision of a model at various classification thresholds. It is determined by measuring the area under the precision-recall curve for a specific class. AP is then calculated by averaging over the number of classes to determine mAP:

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i, \tag{7}$$

where N represents the number of classes. In our experiments, we computed the $mAP_{0.5}$ and $mAP_{0.5:0.95}$ for all YOLO-based object detection algorithms. The former is the mAP that is calculated using an Intersection over Union (IoU) threshold of 0.5, while the latter is determined using a range of IoU thresholds, from 0.5 to 0.95. Using this range, mAP can be determined at different levels of overlap between the predicted bounding boxes and the ground truth bounding boxes.

We typically use Frames Per Second (FPS) to measure the rate of image output. In this case, FPS evaluates the number of frames output by the object detection algorithm in one second.

4. Experiments

The experiments are conducted on a computer equipped with an NVIDIA TITAN RTX GPU, operating on the Ubuntu 20.04 system. We utilize the PyTorch framework to implement our proposed method. Our models are trained and evaluated using the SGD optimization algorithm, with a learning rate of 0.01 and a batch size of 32. The input image size of the model is 320×320. We train each model for 300 epochs and select the best validation results as the final outcome. We evaluate the models' performance by utilizing metrics such as mAP, precision, and recall.

The YOLO-based object detection models tested in this paper include YOLO-V8, YOLO-V7, YOLO-V5, YOLO-V4 and YOLO-V3. We do not consider YOLO-V1 and YOLO-V2 as YOLO-V3 is the first version that has gained widespread use in various applications. The different architectures of each model are listed in Table 2, along with the corresponding performance metrics such as precision, recall, mAP at different IoU thresholds, and FPS. Since the source code of YOLO-V4 and YOLO-V3 does not provide a direct precision, recall calculation method, in order to ensure the fairness of the comparison, we can not calculate their values directly. YOLO-based object detection algorithms generally provide pre-trained models of varying sizes to accommodate different computational requirements and applications. For example, YOLO-V5 provides models in different sizes, such as tiny, small, medium, large, and extra large. As a result, we compare network architectures of different sizes in the same YOLO-based algorithm in Table 2. The second column of Table 2 shows the network architectures of various sizes for the same model.

The results indicate that the YOLO-V8 models generally outperform the other algorithms in terms of precision, $mAP_{0.5}$, and $mAP_{0.5:0.95}$. The YOLO-V8-large baseline model achieves the highest $mAP_{0.5}$ at 87.09% and $mAP_{0.5:0.95}$ at 54.30%. All YOLO-V8 models demonstrate a mAP of over 50% across the threshold range of 0.5 to 0.95, surpassing the top performances of YOLO-V7 and YOLO-V5 by 4.53% and 7.31% respectively. These results indicate that the enhancements made to the network architecture of YOLO-V8 have significantly improved its ability to capture features in bronchoscopy images, leading to a noticeable boost in performance. Furthermore, it was observed that in YOLO-V5, YOLO-V7, and YOLO-V8, larger models generally achieve higher $mAP_{0.5:0.95}$ in comparison to smaller models. This indicates that models with a substantial number of parameters are capable of learning the feature representation of our dataset without encountering underfitting issues.

Although the YOLO-V4 models do provide impressive FPS values, it is worth noting that the mAP performance of YOLO-V4 is relatively lower. In addition, the speed of YOLO-V4 is mainly due to the utilization of the Darknet framework, which is implemented

Table 3
Table of results with three selected YOLO-based object detection architectures using two data augmentation methods with different magnitudes.

Architecture	Data augmentation method with different magnitude	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)
YOLO-V8-large	Mosaic 0 + Scale 0.0	82.36	50.93
	Mosaic 1 + Scale 0.0	**85.50**	**51.56**
	Mosaic 0 + Scale 0.1	82.55	51.89
	Mosaic 0 + Scale 0.4	83.98	51.93
	Mosaic 0 + Scale 0.7	84.89	**53.24**
	Mosaic 0 + Scale 0.9	**86.22**	52.95
YOLO-V7-base	Mosaic 0 + Scale 0.0	60.17	27.58
	Mosaic 1 + Scale 0.0	**80.13**	**43.69**
	Mosaic 0 + Scale 0.1	66.53	30.24
	Mosaic 0 + Scale 0.4	75.63	33.76
	Mosaic 0 + Scale 0.7	75.77	35.29
	Mosaic 0 + Scale 0.9	**76.94**	**38.05**
YOLO-V5-middle	Mosaic 0 + Scale 0.0	60.13	29.50
	Mosaic 1 + Scale 0.0	**75.92**	**40.64**
	Mosaic 0 + Scale 0.1	72.42	33.37
	Mosaic 0 + Scale 0.4	78.05	34.49
	Mosaic 0 + Scale 0.7	**78.06**	**35.24**
	Mosaic 0 + Scale 0.9	77.70	34.65

in C++. It is important to also take into account other networks such as YOLO-V5, YOLO-V7, and YOLO-V8, which are more advanced and have higher mAP values. Although their FPS values may not reach the levels of the YOLO-V4 models, their superior object detection capabilities are an important aspect to consider.

As indicated in Table 2, it is evident that the model's architecture significantly impacts the object detection model's performance. As the model parameters increase, the FPS decreases, as can be seen in the YOLO-V8 architecture results. When choosing a model for a specific use case, it is important to consider the balance between accuracy and speed.

Fig. 4 shows a comparison of the ground truth (in Column a) and the object detection results achieved by the YOLO-V5 (in Column d), YOLO-V7 (in Column c), and YOLO-V8 (in Column b) architectures, using the best model from each type, as determined in Table 2, was used for this comparison. The figure shows the detection of multiple objects in a bronchoscopy image, with the predicted bounding boxes overlaid on the original image. It has been noted that those three models exhibit accurate object detection and localization. However, there are some differences in the number of false positives and false negatives. Based on the visual comparison in Fig. 4, it is evident that the YOLO-V5 and YOLO-V7 models have similar detection accuracy. However, the YOLO-V7 model exhibits fewer false positives compared to the YOLO-V5 model. The architecture of the YOLO-V8 model closely resembles that of YOLO-V7. Hence, it is evident from Fig. 4 that both YOLO-V7 and YOLO-V8 produce fewer unnecessary bounding boxes compared to YOLO-V5, while maintaining the same IoU threshold. In the context of bronchoscopy navigation, false positives in object detection may lead to misidentification of important elements like bronchial branches, causing navigation errors and potentially prolonging procedure times. Therefore, the YOLO-V8 model might be better suited for clinical applications because of its superior precision.

The results of a study examining the effects of various data augmentation methods on the performance of three YOLO-based object detection models (YOLO-V8-large, YOLO-V7-base, and YOLO-V5-middle) are presented in Table 3. The data augmentation methods used in the study include mosaic and scale, which are applied to the training dataset with different magnitudes. The mosaic method randomly pastes patches of other images onto the original image, whereas the scale method resizes it by a predetermined percentage. The table provides information on the data augmentation method and magnitudes used for each experiment, along with the mAP values at various IoU thresholds.

The results show that using the mosaic data augmentation method with a magnitude of 1.0 results in significantly higher mAP values for those three models than not using any data augmentation. After employing the mosaic data augmentation method, the $mAP_{0.5}$ for YOLO-V5-middle and YOLO-V7-base improved by 15.79% and 19.96%, respectively. This behavior suggests that the mosaic method can effectively improve the performance of these models, especially at 0.5 IoU thresholds. In addition, higher scale magnitudes generally lead to higher mAP values, which indicates that random resizing of images in a wide range can improve the performance of the object detection algorithms. However, it is worth noting that the optimal scale magnitude may vary depending on the specific architecture. For example, the YOLO-V7-base architecture performs best when a scale magnitude of 0.9 is used, whereas the YOLO-V5-middle and YOLO-V8-large architecture perform best when a scale magnitude of 0.7 is applied. As described in Section 2, we utilize large bounding boxes to represent the camera's location on the bronchoscope and smaller bounding boxes to indicate the predicted category and position. A greater proportion of bronchoscopy images with scaling augmentation can generate more samples to characterize the relationship between these two types of bounding boxes. Furthermore, scaling bronchoscope images is more consistent with the real-world practices of physicians using endoscopes. In summary, mosaic and scale data augmentation methods can improve the performance of YOLO-based object detection models.

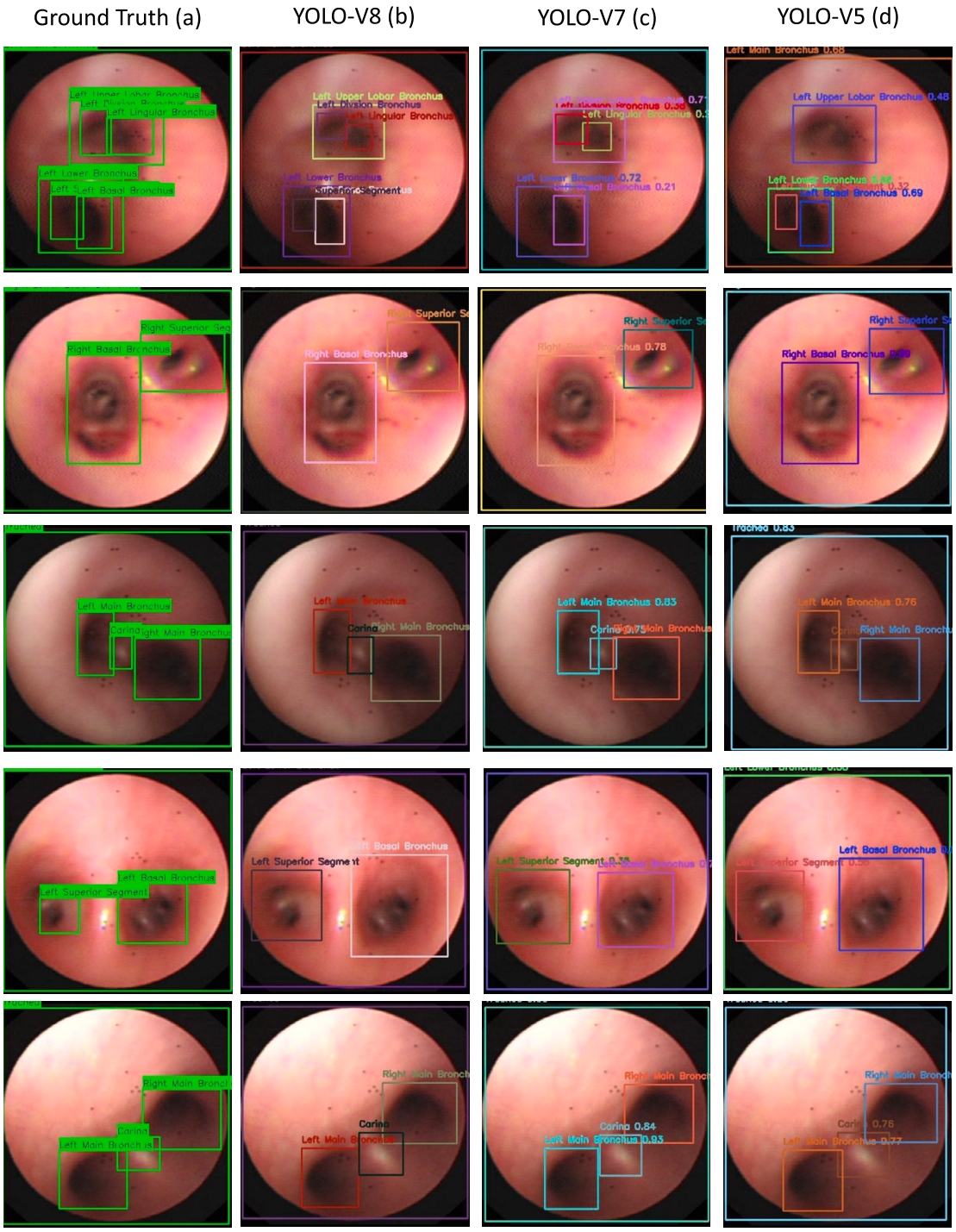

Fig. 4. Comparison of the ground truth (in Column a) and the object detection results of YOLO-V5 (in Column d), YOLO-V7 (in Column c) and YOLO-V8 (in Column b) on bronchoscopy images.

Table 4 presents the results of integrating the CBAM attention module with various layers of two YOLO-based object detection architectures: YOLO-V7 and YOLO-V8. In each architecture, the CBAM module is integrated at different input sizes to evaluate its impact on performance. The CBAM module is applied to either the head (i.e., the output layer) or the backbone (i.e., the intermediate layers) of each architecture. In Table 4, results from different configurations of YOLO-V7 and YOLO-V8, as outlined in Table 2, are used as the benchmarks. The results obtained from the corresponding models are compared to the baseline models without CBAM. At different IoU thresholds, performance is evaluated using a variety of metrics such as precision, recall, and mAP.

Table 4
Table of the results with CBAM combinations in different layers of two YOLO-based object detection architectures.

Architecture	CBAM	Position	Precision	Recall	$mAP_{0.5}$ (%)	$mAP_{0.5:0.95}$ (%)
YOLO-V7-base	×	–	82.18	84.17	85.67	49.36
YOLO-V7-base-cbam-head	✓	head	84.75	83.52	87.12	51.35
YOLO-V7-base-cbam-all	✓	head & backbone	85.56	84.72	87.81	52.10
YOLO-V7-tiny	×	–	83.73	83.14	85.78	49.73
YOLO-V7-tiny-cbam-head	✓	head	83.01	85.98	87.01	51.81
YOLO-V7-tiny-cbam-all	✓	head & backbone	84.15	85.42	87.52	51.86
YOLO-V7-extra-large	×	–	81.47	82.76	84.88	49.77
YOLO-V7-extra-large-cbam-head	✓	head	81.96	82.89	85.47	49.99
YOLO-V7-extra-large-cbam-all	✓	head & backbone	81.74	82.97	85.55	49.86
YOLO-V8-tiny	×	–	82.66	83.56	86.31	53.36
YOLO-V8-tiny-cbam-head	✓	head	85.17	85.93	87.48	54.69
YOLO-V8-tiny-cbam-all	✓	head & backbone	85.44	**87.22**	88.15	55.26
YOLO-V8-small	×	–	83.54	83.26	86.94	53.53
YOLO-V8-small-cbam-head	✓	head	86.34	85.43	87.83	54.81
YOLO-V8-small-cbam-all	✓	head & backbone	86.30	86.29	88.06	55.12
YOLO-V8-middle	×	–	83.34	83.05	86.22	53.46
YOLO-V8-middle-cbam-head	✓	head	85.04	83.89	86.80	54.62
YOLO-V8-middle-cbam-all	✓	head & backbone	85.43	83.85	87.21	54.98
YOLO-V8-large	×	–	85.26	81.83	87.09	54.30
YOLO-V8-large-cbam-head	✓	head	86.13	85.57	87.91	55.02
YOLO-V8-large-cbam-all	✓	head & backbone	**86.42**	86.40	**88.27**	**55.39**
YOLO-V8-extra-large	×	–	85.09	83.34	86.45	54.10
YOLO-V8-extra-large-cbam-head	✓	head	85.05	85.01	87.08	54.68
YOLO-V8-extra-large-cbam-all	✓	head & backbone	85.75	86.03	87.93	55.09

The results demonstrate that the CBAM module consistently improves the performance of all architectures in terms of mAP. As shown in Table 4, the results indicate that integrating the CBAM module into any architecture consistently results in significant improvements in both $mAP_{0.5}$ and $mAP_{0.5:0.95}$ compared to the original baselines. The largest improvements in $mAP_{0.5}$ and $mAP_{0.5:0.95}$ are observed in the YOLO-V7-base model, with increases of 2.14% and 2.74%, respectively. The highest mAP values are observed when CBAM is applied to both the head and the backbone. For example, the YOLO-V8-large-cbam-all model achieves an mAP of 88.27%, which is higher than those obtained from the baseline model (87.09%) and the model with CBAM in the head only (87.91%). Similarly, the YOLO-V7-base-cbam-all model achieves an mAP of 87.81%, which is higher than that of the baseline model (85.67%) and the model with CBAM in the head only (87.12%). In the YOLO-V8-large-cbam-all model, we achieved the highest $mAP_{0.5}$ and $mAP_{0.5:0.95}$ of 88.27% and 55.39%, respectively. Compared to the results of YOLO-V7 in Table 2, YOLO-V8 exhibits significant improvements. The inclusion of CBAM further increases the performance gap between YOLO-V8 and YOLO-V7. Thus, it is evident that CBAM contributes to significant enhancements in our bronchoscopy dataset.

However, it is important to note that mAP improvement does not always result in improved precision or recall. For example, the YOLO-V7-tiny-cbam-head model outperforms the YOLO-V7-tiny model in terms of mAP but falls short in precision. Surprisingly, this phenomenon is not found in the YOLO-V8. This implies that the CBAM module may have varying effects on different performance metrics, depending on the architecture and the layer to which it is applied.

Overall, these results indicate that the CBAM module has the potential to significantly improve the performance of object detection algorithms, particularly when applied to both the architecture's head and backbone. However, the specific impact on different performance metrics may vary depending on the architecture and layer where the CBAM module is applied.

5. Conclusion and discussion

This study investigated the use of object detection techniques on bronchoscopy images. Our study found that YOLO-based architectures are highly accurate in detecting bronchoscopy instruments, with mAP values ranging from 80.35% to 87.09%. With the YOLO-V8-large-cbam-all model, we achieve the highest $mAP_{0.5}$ and $mAP_{0.5:0.95}$ of 88.27% and 55.39%, respectively. The results further demonstrated that including CBAM attention mechanisms in detection architectures improved overall performance, especially when added to both the head and the backbone layers.

Our findings highlighted the potential of object detection techniques to aid in the diagnosis and treatment of lung diseases, by enabling automated detection and localization of bronchoscopy instruments. In particular, the use of CBAM attention mechanisms could improve these systems' accuracy and robustness.

One limitation of our proposed method was the post-processing of the predicted results. Currently, the predicted bounding boxes were processed independently of one another, without considering the sequential nature of bronchoscopy images. This could result in inconsistencies in predicted object locations across sequential frames, reducing the overall performance of the object detection

system. mAP could be further improved by developing post-processing techniques that consider the temporal relationships between sequential frames. Such techniques could include methods for tracking objects across frames and using temporal information to refine the predicted bounding boxes. Additionally, the proposed method could be expanded to include a video-based approach, in which the entire video sequence is processed as a whole rather than individual frames. This could help to improve the accuracy of the object detection system in bronchoscopy images.

Future research on deep learning models can focus in a variety of directions to improve object detection performance on bronchoscopy images. A promising approach is to investigate the use of self-attention mechanisms in the Transformer architecture [37]. The effectiveness of self-attention has been demonstrated for capturing long-range dependencies and contextual information in natural language processing tasks. It can also be adapted to image analysis by modeling the interactions between different parts of an image. The incorporation of self-attention into the object detection pipeline can improve the detection of small, low-contrast objects in bronchoscopy images.

Semi-supervised learning methods [34] are another promising area for further research. As previously stated, the limited availability of annotated data poses a significant challenge in bronchoscopy image analysis. Semi-supervised learning can improve the performance of object detection models by combining labeled and unlabeled data. One approach is to use generative models, such as generative adversarial networks (GANs) [11], to create more training data to fine-tune the detection model. Another approach involves self-supervised learning [17], where a model is trained to predict a particular property of an image (e.g., image rotation, mask recovery) without the need for manual annotations. Finally, the learned representations can be used to improve the detection model's performance on labeled data.

Future research could explore the use of two-stage detection models, such as Faster R-CNN [10]. Unlike YOLO-based models, which perform object detection and classification in a single pass, two-stage models generate region proposals before refining them with a separate classifier. This approach has been shown to improve object detection accuracy across a variety of datasets. It may be beneficial to investigate the adaptability of this approach to bronchoscopy images in order to improve the detection of difficult targets.

Ethics statement

All participants/patients (or their proxies/legal guardians) provided informed consent to participate in the study.

CRediT authorship contribution statement

Jianqi Yan: Writing – review & editing, Writing – original draft, Validation, Project administration, Methodology, Investigation. **Yifan Zeng:** Writing – original draft, Software, Data curation. **Junhong Lin:** Validation, Investigation, Data curation. **Zhiyuan Pei:** Writing – review & editing, Validation, Methodology. **Jinrui Fan:** Writing – review & editing, Formal analysis, Conceptualization. **Chuanyu Fang:** Software, Formal analysis. **Yong Cai:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Investigation, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data for bronchoscopy were collected from a database of a pediatric hospital. The company has a confidentiality agreement with the pediatric hospital that the authors do not have permission to share data.

Funding

This study received funding from the Zhuhai Industry-University-Research Cooperation Project. (ZH22017001210047PWC and 2220004002437-03).

References

[1] Ghada Hamed Aly, Mohammed Marey, Safaa Amin El-Sayed, Mohamed Fahmy Tolba, Yolo based breast masses detection and classification in full-field digital mammograms, Comput. Methods Programs Biomed. 200 (2021) 105823.

[2] Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao, Yolov4: optimal speed and accuracy of object detection, arXiv preprint arXiv:2004.10934, 2020.

[3] Juan Borrego-Carazo, Carles Sanchez, David Castells-Rufas, Jordi Carrabina, Débora Gil, Bronchopose: an analysis of data and model configuration for vision-based bronchoscopy pose estimation, Comput. Methods Programs Biomed. 228 (2023) 107241.

[4] Jie Chen, Li Wan, Jingru Zhu, Gang Xu, Min Deng, Multi-scale spatial and channel-wise attention for improving object detection in remote sensing imagery, IEEE Geosci. Remote Sens. Lett. 17 (4) (2019) 681–685.

[5] Ernst Eber, Juan L. Antón-Pacheco, Jacques de Blic, Iolo Doull, Al Faro, Raffaella Nenna, Thomas Nicolai, Petr Pohunek, Kostas N. Priftis, Paola Serio, et al., Ers statement: interventional bronchoscopy in children, Eur. Respir. J. 50 (6) (2017).

[6] Albert Faro, Robert E. Wood, Michael S. Schechter, Albin B. Leong, Eric Wittkugel, Kathy Abode, James F. Chmiel, Cori Daines, Stephanie Davis, Ernst Eber, et al., Official American thoracic society technical standards: flexible airway endoscopy in children, Am. J. Respir. Crit. Care Med. 191 (9) (2015) 1066–1080.
[7] Yabo Fu, Yang Lei, Tonghe Wang, Kristin Higgins, Jeffrey D. Bradley, Walter J. Curran, Tian Liu, Xiaofeng Yang, Lungregnet: an unsupervised deformable image registration method for 4d-ct lung, Med. Phys. 47 (4) (2020) 1763–1774.
[8] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, Jian Sun, Yolox: exceeding yolo series in 2021, arXiv preprint arXiv:2107.08430, 2021.
[9] Jose George, Shibon Skaria, V.V. Varun, et al., Using yolo based deep learning network for real time detection and localization of lung nodules from low dose ct scans, in: Medical Imaging 2018: Computer-Aided Diagnosis, volume 10575, SPIE, 2018, pp. 347–355.
[10] Ross Girshick, Fast r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.
[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, Generative adversarial networks, Commun. ACM 63 (11) (2020) 139–144.
[12] P. Goussard, P. Pohunek, E. Eber, F. Midulla, G. Di Mattia, M. Merven, J.T. Janson, Pediatric bronchoscopy: recent advances and clinical challenges, Expert Rev. Respir. Med. 15 (4) (2021) 453–475.
[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Spatial pyramid pooling in deep convolutional networks for visual recognition, IEEE Trans. Pattern Anal. Mach. Intell. 37 (9) (2015) 1904–1916.
[14] Glenn Jocher, Ayush Chaurasia, Jing Qiu, Ultralytics YOLO, January 2023.
[15] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, NanoCode012, Yonghye Kwon, TaoXie, Kalen Michael, Jiacong Fang, imyhxy, Lorna, Colin Wong, Zeng Yifu, Abhiram V, Diego Montes, Zhiqiang Wang, Cristi Fati, Jebastin Nadar, Laughing, UnglvKitDe, tkianai, yxNONG, Piotr Skalski, Adam Hogan, Max Strobel, Mrinal Jain, Lorenzo Mammana, xylieong, ultralytics/yolov5: v6.2 - YOLOv5 Classification Models, Apple M1, Reproducibility, ClearML and Deci.ai integrations, August 2022.
[16] Lars Konge, Klaus Richter Larsen, Paul Clementsen, Henrik Arendrup, Christian Von Buchwald, Charlotte Ringsted, Reliable and valid assessment of clinical bronchoscopy performance, Respiration 83 (1) (2012) 53–60.
[17] Yandong Li, Di Huang, Danfeng Qin, Liqiang Wang, Boqing Gong, Improving object detection with selective self-supervised self-training, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIX, Springer, 2020, pp. 589–607.
[18] Ying Li, Xiaoxuan Zheng, Fangfang Xie, Lin Ye, Elena Bignami, Yasmeen K. Tandon, María Rodríguez, Yun Gu, Jiayuan Sun, Development and validation of the artificial intelligence (ai)-based diagnostic model for bronchial lumen identification, Transl. Lung Cancer Res. 11 (11) (2022) 2261.
[19] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, Jiaya Jia, Path aggregation network for instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8759–8768.
[20] Shuai Liu, Lu Zhang, Huchuan Lu, You He, Center-boundary dual attention for oriented object detection in remote sensing images, IEEE Trans. Geosci. Remote Sens. 60 (2021) 1–14.
[21] Stuart Lloyd, Least squares quantization in pcm, IEEE Trans. Inf. Theory 28 (2) (1982) 129–137.
[22] Clyde Matava, Evelina Pankiv, Sam Raisbeck, Monica Caldeira, Fahad Alam, A convolutional neural network for real time classification, identification, and labelling of vocal cord and tracheal using laryngoscopy and bronchoscopy video, J. Med. Syst. 44 (2) (2020) 1–10.
[23] Daniel R. Ouellette, The safety of bronchoscopy in a pulmonary fellowship program, Chest 130 (4) (2006) 1185–1190.
[24] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, Adam Lerer, Automatic differentiation in pytorch, 2017.
[25] K.N. Priftis, M.B. Anthracopoulos, E. Eber, A.C. Koumbourlis, R.E. Wood, Paediatric Bronchoscopy, in: Bollinger C.T. (Ed.), Progress in Respiratory Research, vol. 38, 2010.
[26] Joseph Redmon, Darknet: open source neural networks in c, http://pjreddie.com/darknet/, 2013–2016.
[27] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[28] Joseph Redmon, Ali Farhadi, Yolo9000: better, faster, stronger, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 7263–7271.
[29] Joseph Redmon, Ali Farhadi, Yolov3: an incremental improvement, arXiv preprint arXiv:1804.02767, 2018.
[30] Rasmus Rothe, Matthieu Guillaumin, Luc Van Gool, Non-maximum suppression for object detection by passing messages between windows, in: Asian Conference on Computer Vision, Springer, 2014, pp. 290–306.
[31] Brienne Ryan, Keerti Yendamuri, Sai Yendamuri, Anatomical considerations in bronchoscopy, J. Thorac. Dis. 9 (Suppl 10) (2017) S1123.
[32] Carlos Santos, Marilton Aguiar, Daniel Welfer, Bruno Belloni, A new approach for detecting fundus lesions using image processing and deep neural network architecture based on yolo model, Sensors 22 (17) (2022) 6441.
[33] Mali Shen, Yun Gu, Ning Liu, Guang-Zhong Yang, Context-aware depth and pose estimation for bronchoscopic navigation, IEEE Robot. Autom. Lett. 4 (2) (2019) 732–739.
[34] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, Tomas Pfister, A simple semi-supervised learning framework for object detection, arXiv preprint arXiv:2005.04757, 2020.
[35] Gabriel F. Tucker, Arthur M. Olsen, Albert H. Andrews, John L. Pool, The flexible fiberscope in bronchoscopic perspective, Chest 64 (2) (1973) 149–150.
[36] D. Tzutalin Labelimg, GitHub Repository 6 (2015).
[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).
[38] Marco Visentini-Scarzanella, Takamasa Sugiura, Toshimitsu Kaneko, Shinichiro Koto, Deep monocular 3d reconstruction for assisted navigation in bronchoscopy, Int. J. Comput. Assisted Radiol. Surg. 12 (7) (2017) 1089–1099.
[39] Chen Wang, Xiao Bai, Shuai Wang, Jun Zhou, Peng Ren, Multiscale visual attention networks for object detection in vhr remote sensing images, IEEE Geosci. Remote Sens. Lett. 16 (2) (2018) 310–314.
[40] Chien-Yao Wang, Alexey Bochkovskiy, Hong-Yuan Mark Liao, Yolov7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, arXiv preprint arXiv:2207.02696, 2022.
[41] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, I-Hau Yeh, Cspnet: a new backbone that can enhance learning capability of cnn, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 390–391.
[42] Chien-Yao Wang, I-Hau Yeh, Hong-Yuan Mark Liao, You only learn one representation: unified network for multiple tasks, arXiv preprint arXiv:2105.04206, 2021.
[43] Sanghyun Woo, Jongchan Park, Joon-Young Lee, In So Kweon, Cbam: convolutional block attention module, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3–19.
[44] Robert E. Wood, Robert J. Fink, Applications of flexible fiberoptic bronchoscopes in infants and children, Chest 73 (5) (1978) 737–740.
[45] Zhitao Xiao, Bowen Liu, Lei Geng, Fang Zhang, Yanbei Liu, Segmentation of lung nodules using improved 3d-unet neural network, Symmetry 12 (11) (2020) 1787.
[46] Ji Young Yoo, Se Yoon Kang, Jong Sun Park, Young-Jae Cho, Sung Yong Park, Ho Il Yoon, Sang Jun Park, Han-Gil Jeong, Tackeun Kim, Deep learning for anatomical interpretation of video bronchoscopy images, Sci. Rep. 11 (1) (2021) 23765.