HIGH ACCURACY SURGICAL INSTRUMENT DETECTION WITH YOLOV8 MODEL AND REMOTE MONITORING

Duong Ba Tung Le
International School
Vietnam National University
Hanoi, Vietnam
le2407204@gmail.com

Thai Dinh Kim
International School
Vietnam National University
Hanoi, Vietnam
thaikd@vnu.edu.vn

Manh Hung Ha
International School
Vietnam National University
Hanoi, Vietnam
hunghm@vnu.edu.vn

Duc Anh Pham
International school
Vietnam National University
Hanoi, Vietnam
phamducanh086@gmail.com

Ngoc Nam Dao
International school
Vietnam National University
Hanoi, Vietnam
namzzzkb@gmail.com

Minh Duc Nguyen
International school
Vietnam National University
Hanoi, Vietnam

ABSTRACT

Surgical instrument detection is a vital task in analyzing images of laparoscopic surgery. This study introduces a strong method for detecting surgical instruments using the YOLOv8 model. The evaluation results proves that this model exhibits high accuracy and outperforms previous approaches. Additionally, a web interface is developed for convenient remote monitoring of the instruments.

Keywords: *MIS, Object Detection, Surgical Tool Detection, YOLO*

1. INTRODUCTION

Nowadays, laparoscopic surgery is gradually replacing traditional open surgery because it has significant advantages such as less postoperative pain, faster recovery, shorter hospital stay, smaller scars, and lower risk of infection. However, due to the inability to directly see the patient's abdominal cavity and the need for indirect observation through a display screen to perform the operation, laparoscopic surgery is actually much more difficult than traditional open surgery techniques, especially for inexperienced doctors. Especially in Vietnam, the application of computer vision in endoscopic surgery and the research and application of detecting surgical instruments are still quite new and have not yet truly developed. Therefore, we suggest applying object detection to real-time detection of laparoscopic surgical instruments to assist doctors in performing surgeries

The rest of the paper is organized as follows: Section II provides an overview of the architecture of the YOLOv8 [1] model, while Section III presents several results from the dataset. Lastly, in Section IV, conclusions are drawn.

2. YOLOV8 MODEL

YOLO (You Only Look Once) is an advanced object detection algorithm that was first introduced in 2016 by Joseph Redmon et al [2]. It is a deep learning-based approach that predicts bounding boxes and class probabilities for objects in an input image to perform object detection. The YOLO model stands out in terms of processing speed compared to other architectures. Because of this advantage, the model is consistently being enhanced in various versions. These improvements focus on enhancing accuracy, speed, and incorporating additional features like pose detection and instance segmentation. In this study, we utilize the YOLOv8 [1] model, which is the most up-to-date version from the YOLO family.

As shown in Fig. 1, the head of YOLOv8 consists of multiple convolutional layers followed by fully connected layers. These layers are responsible for predicting bounding boxes, objectness scores, and class probabilities for detected objects in an image. An important feature of YOLOv8 is the incorporation of a self-attention mechanism in the network head. This mechanism enables the model to focus on different regions of the image and adjust the importance of various features according to their relevance to the task.

YOLOv8 introduces several new functionalities, such as custom anchor boxes and transfer learning, facilitating easier training and customization for specific tasks. It employs Varifocal Loss (VFL Loss) for classification loss, and DFL Loss (Distribution Focal Loss) + CIOU Loss (Complete IoU Loss) for localization loss. Specifically, VFL Loss is a focal loss variant that adapts to predictions sensitive to quality, while DFL Loss models label distribution as a Dirichlet distribution, and CIOU Loss is a bounding box regression loss that accounts for overlap, aspect ratio, and distance. Regarding optimization, YOLOv8 integrates hyperparameter tuning using Ray Tune, allowing effortless optimization of hyperparameters for the YOLOv8 model. Ray Tune can expedite the tuning process and enhance model performance through advanced search techniques, parallelism, and early stopping mechanisms.

3. RESULTS

3.1 Dataset

In this study, we use the m2cai16-tool-locations dataset, which has been provided in reference [3], as the basis for training and evaluating the effectiveness of our proposed model. The m2cai16-tool-locations dataset is derived from the

m2cai16-tool dataset [4] and has been specifically curated for the purpose of detecting seven commonly used instruments in laparoscopic cholecystectomy. This dataset comprises a total of 2811 labeled images, with each image including the names and coordinates of the bounding boxes surrounding the tips of each instrument.

To train and validate our model, we will split the dataset as follows: 50% for training, 30% for validation, and the remaining 20% for testing the performance of the model.

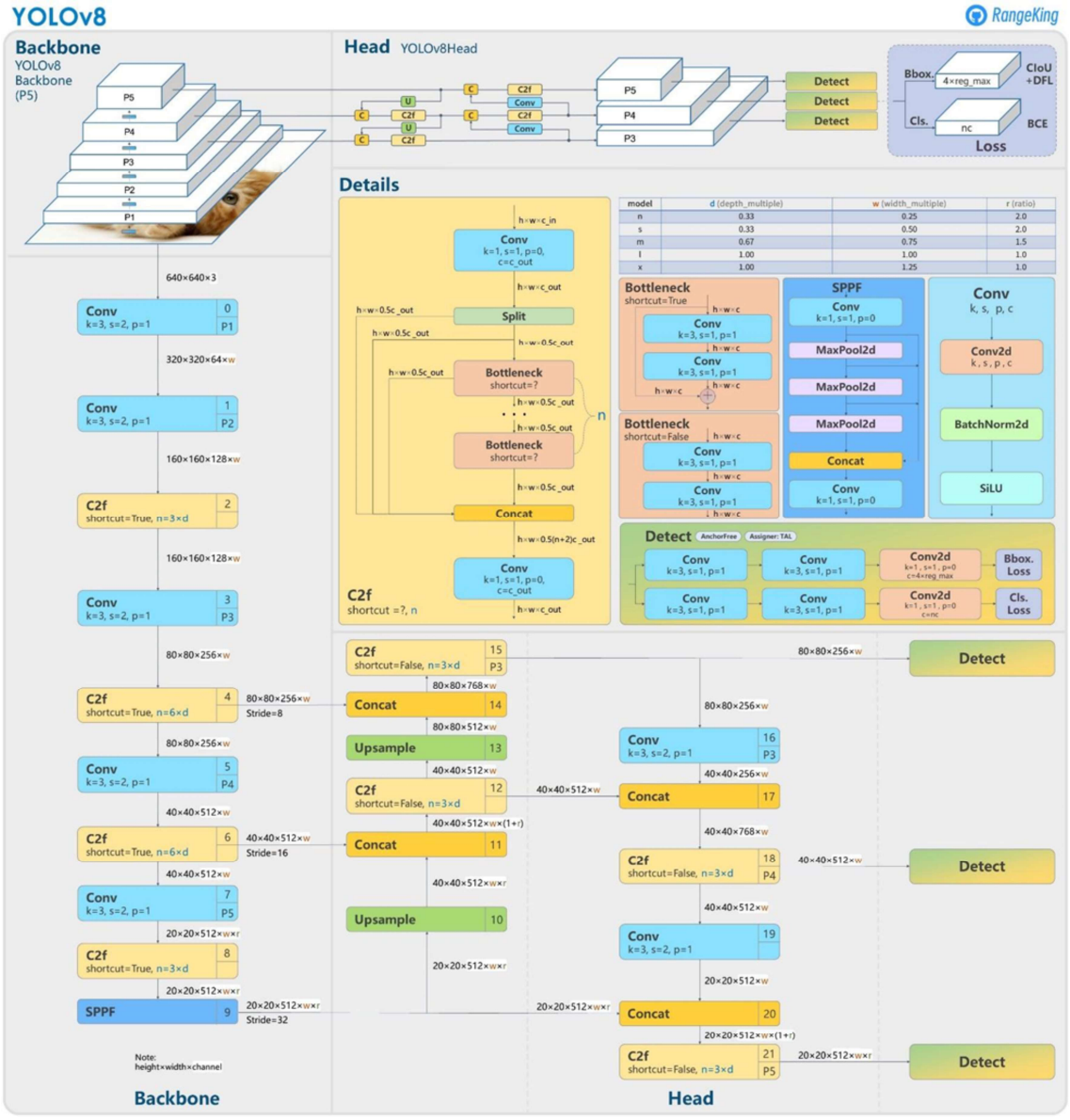

Fig. 1. *The detailed network model architecture of YOLOv8*

Table 1. *The number of instances and images of each class in a train, valid, and test dataset*

Classes	Train	Vaidation	Test
Grasper	707	422	293
Hook	165	79	64
Clipper	220	116	64
Scissors	197	107	81
SpecimenBag	241	139	96
Bipolar	215	140	95
Irrigator	228	173	84
Total images	1045	843	563

Table 1 outlines the number of images, names, and labels for each of the seven surgical tools in the m2cai16-tool-locations dataset.

3.2. Evaluation Metrics

IoU (Intersection over Union) is the basic metric for measuring the performance of object detection algorithms. From Fig. 2., we can explain that it represents the percentage of overlap between two boxes, the ground truth box (G) and the detection box (D). It is calculated using the following Equation:

$$IoU = \frac{Area\ of\ Intersection}{Area\ of\ Union} = \frac{G \cap P}{G \cup P} \quad (1)$$

In this paper, we evaluate the performance of each model by quantitative metrics such as Precision (P), Recall (R), and mean Average Precision (mAP) with the equations as follows:

$$P = \frac{TP}{TP+F} = \frac{TP}{All\ detections} \quad (2)$$

$$R = \frac{TP}{TP+FN} = \frac{TP}{All\ ground\ trut} \quad (3)$$

Specifically, TP (True Positive) is the total number of detections with IoU greater than or equal to 0.5; FP (False Positive) is the total number of detections with IoU less than 0.5; and FN (False Negative) is the total number of undetectable objects in the test set.

$$AP = \sum_{i=1}^{n} p(i)\, \Delta r(i) \quad (4)$$

$$mAP = \frac{1}{K} \sum_{i=1}^{K} AP_i \quad (5)$$

The average precision (AP) is the average accuracy of the model, while the mean average precision (mAP) is the average of AP over all detected classes. K represents the number of categories. In this paper, we use mAP50 and mAP50-95 to evaluate the performance of different models. The term mAP50 refers to the average precision for all classes at an IoU of 0.5. On the other hand, mAP50-95 refers to the average precision for all classes within a range of IoU from 0.5 to 0.95, with a step size of 0.05.

3.3. Experimental results

We trained and tested these models on the Google Colab platform. For the experimental process, we selected the following parameters for the YOLOv8 models: epochs=100, patience=50, batch size=16, image size=640, workers=8, optimizer=SGD, lr=0.01, momentum=0.937, weight decay=0.0005.

Specifically, we trained the models for 100 iterations over the entire dataset and chose a patience of 50 to prevent overfitting and save computational resources. We adjusted the images from the dataset to be squares with a size of 640 ×640 pixels, which is suitable for the training process. We used the stochastic gradient descent (SGD) optimizer to minimize the total loss. The parameters related to the optimizer, such as learning rate and momentum, were kept as default because these values provide a balance between the speed and efficiency of the loss minimizing process. After completing the training process, we analyze the performance of the YOLOv8 model on the valid dataset using quantitative measures. Table 2 shows that the larger models have higher precision. On the other hand, the medium-sized YOLOv8 models, with 25.8 million parameters, exhibit both high precision and high recall. As a result, the YOLOv8m model performs the best overall, achieving a mean average precision (mAP) of 0.956 at a threshold of 50% (mAP50) and a mAP of 0.607 across a range of thresholds from 50% to 95% (mAP50-95).

Table 2. *Performance comparison of different YOLOv8 models*

Model	Params (M)	P	R	mAP50	mAP50-95
YOLOv8n	3.0	0.93	0.929	0.956	0.607
YOLOv8s	11.1	0.936	0.937	0.958	0.629
YOLOv8m	25.8	0.961	0.917	0.958	0.63
YOLOv8l	43.6	0.969	0.910	0.965	0.636

Table 3. *Detailed evaluation results for each class using YOLOv8m model*

Class	Instances	P	R	mAP50	mAP50-95
Total	780	0.930	0.929	0.956	0.607
Grasper	293	0.857	0.863	0.911	0.548
Hook	64	0.974	0.969	0.982	0.736
Clipper	64	0.983	0.905	0.960	0.689
Scissors	84	0.903	0.952	0.964	0.576
Specimen Bag	96	0.914	0.917	0.947	0.630
Bipolar	95	0.953	0.958	0.976	0.582
Irrigator	84	0.929	0.940	0.953	0.491

In Table 3, we provide a detailed breakdown of the results for YOLOv8m, which demonstrate its precise detection of personal protective equipment across four classes, with a mAP50 exceeding 0.97, and nearly 0.92 for one specific class.

Table 4. *Comparing the performance of different models using mAP50*

Class	**YOLOv8n**	**YOLOv8l**	**K. Jo[22]**	**A. Jin[3]**
Total	0.956	0.965	0.847	0.818
Grasper	0.911	0.90	0.921	0.872
Hook	0.982	0.995	0.859	0.953
Clipper	0.96	0.974	0.853	0.884
Scissors	0.964	0.975	0.821	0.708
Specimen Bag	0.947	0.96	0.832	0.821
Bipolar	0.976	0.97	0.823	0.751
Irrigator	0.953	0.979	0.829	0.735

Next, we conducted a comparison between our model and the models proposed in the papers by K. Jo et. al [22] and A. Jin et. al [23], as shown in Table 4. The results of our evaluation show that YOLOv8l outperforms the model described in paper [22] by 2.9% in terms of mAP50 and by 4.6% in terms of mAP50 in comparison to the model in paper [23]. Additionally, YOLOv8l achieves a 1.3% higher mAP50-95 than the results reported in [22] and a 2.1% higher mAP50-95 than the results reported in [23]. These findings demonstrate the superior performance of the YOLOv8l model compared to the previous model.

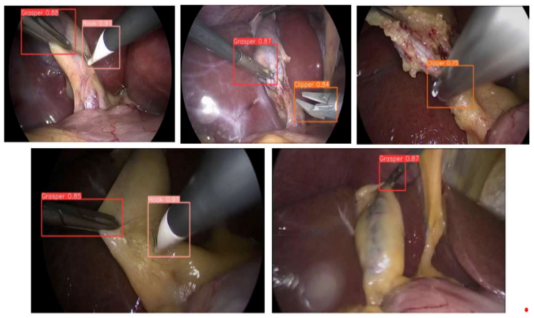

Fig. 2. *Some of the detected object detection results from the proposed model*

Fig. 2 provides some detection results of the YOLOv8n model for various scenarios. These results demonstrate that the model has the capability to accurately detect surgical instruments with different angles and distances

Fig. 3. *A straightforward web interface designed for surgical instrument detection*

Finally, we have developed an efficient web interface that facilitates remote monitoring of surgical instruments. Notably, this interface

prominently presents the results of our algorithm's detection of surgical tools. By uploading an image of a surgical scene, users can rely on the website to accurately identify all the surgical instruments in the image. This tool significantly aids surgeons in quickly and precisely identifying surgical tools, ultimately enhancing patient safety.

CONCLUSION

This paper introduces a powerful method, using the YOLOv8 model, for detecting surgical instruments in laparoscopic surgery. The results demonstrate significantly improved evaluation parameters (P, R, and mAP) compared to previous models. Despite a small dataset of only 2811 labeled images, the mAP50 for all instruments exceeds 95.6% for the YOLOv8 models. Additionally, we have developed a user-friendly web interface for convenient remote monitoring of the instruments. Future research aims to enhance the YOLOv8 model further and explore practical applications like evaluating automatic surgeon skills.

REFERENCES

[1] Jocher, G., Chaurasia, A., and Qiu, J. (2023). YOLO by Ultralytics. https://github.com/ultralytics/ultralytics.

[2] Redmon, J., Divvala, S.K., Girshick, R.B., and Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779-788

[3] Jin, A., Yeung, S., Jopling, J.K., Krause, J., Azagury, D., Milstein, A., and Fei-Fei, L. (2018). Tool Detection and Operative Skill Assessment in Surgical Videos Using Region-Based Convolutional Neural Networks. 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 691-699.

[4] Twinanda, A.P., Mutter, D., Marescaux, J., Mathelin, M.D., and Padoy, N. (2016). Single- and Multi-Task Architectures for Tool Presence Detection Challenge at M2CAI 2016. ArXiv, abs/1610.08851.

[5] J. Redmon et al., "You only look once: Unified, real-time object detection," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779-788, June 2016.

[6] Jo, K., Choi, Y., Choi, J., and Chung, J.W. (2019). Robust RealTime Detection of Laparoscopic Instruments in Robot Surgery Using Convolutional Neural Networks with Motion Vector Prediction. Applied Science.