

YOLO-MED : MULTI-TASK INTERACTION NETWORK FOR BIOMEDICAL IMAGES

Suizhi Huang¹, Shalayiding Sirejiding¹, Yuxiang Lu¹, Yue Ding¹, Leheng Liu², Hui Zhou^{2,*}, Hongtao Lu^{1,*}

¹ Department of Computer Science and Engineering, Shanghai Jiao Tong University

² Department of Gastroenterology, Shanghai General Hospital

ABSTRACT

Object detection and semantic segmentation are pivotal components in biomedical image analysis. Current single-task networks exhibit promising outcomes in both detection and segmentation tasks. Multi-task networks have gained prominence due to their capability to simultaneously tackle segmentation and detection tasks, while also accelerating the segmentation inference. Nevertheless, recent multi-task networks confront distinct limitations such as the difficulty in striking a balance between accuracy and inference speed. Additionally, they often overlook the integration of cross-scale features, which is especially important for biomedical image analysis. In this study, we propose an efficient end-to-end multi-task network capable of concurrently performing object detection and semantic segmentation called YOLO-Med. Our model employs a backbone and a neck for multi-scale feature extraction, complemented by the inclusion of two task-specific decoders. A cross-scale task-interaction module is employed in order to facilitate information fusion between various tasks. Our model exhibits promising results in balancing accuracy and speed when evaluated on the Kvasir-seg dataset and a private biomedical image dataset.

Index Terms— Object Detection, Semantic Segmentation, Multi-Task Learning, Task-interaction, Biomedical Images

1. INTRODUCTION

Accurate detection and segmentation of anatomical structures in biomedical images are critical for numerous clinical applications [1, 2]. Object detection is crucial for identifying abnormalities, like polyps in colonoscopy videos, lesions in retinal fundus images [3, 4, 5]. Meanwhile, segmentation delineates object boundaries, which facilitates quantitative assessment. For instance, it is widely employed in segmenting polyps, tumor regions, and organs in CT scans [3, 4, 6]. Deep learning models have shown immense promise for biomedical image analysis. YOLO series [7, 8] and RetinaNet [9] have become classic network architectures in the field of biomedical object detection, while segmentation networks have showcased impressive performance [10, 11, 12, 13, 14]. To address the simultaneous requirements of detection and segmentation [15] and the need to accelerate inference, multi-task networks for biomedical image detection and segmentation are employed [16, 17]. Nevertheless, existing multi-task networks for biomedical images still have certain limitations, such as hard to strike a balance between accuracy and inference speed and not adequately taking the use of features from different tasks. Representative networks like UOLO incorporates U-Net as its core and connect it with a YOLO detection head [16],

This paper is supported by National Nature Science Foundation of China (62176155), Shanghai Municipal Science and Technology Major Project (2021SHZDZX0102). Hongtao Lu is also with MOE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University.

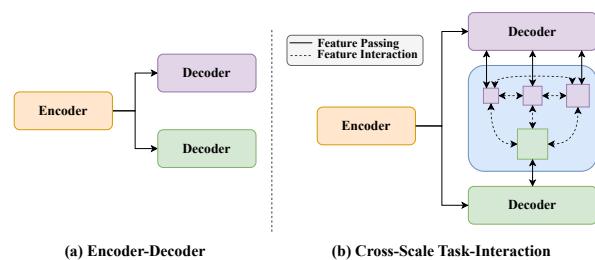


Fig. 1: Comparison between encoder-decoder structure and our cross-scale task-interaction structure.

it remains an encoder-decoder architecture like the structure shown in Fig. 1, and exclusively relies on U-Net [11] for extracting shared features across tasks. MULAN adopts a similar shared encoder, and still faces challenges in effectively fusing detection and segmentation features [17]. Recently, multi-task networks for dense prediction tasks begin to use inter-task information exchange and achieve significant improvements in accuracy [18, 19, 20]. However, these networks are tailored for natural images, which diverge from the unique characteristics of biomedical images. Objects to be detected and segmented in biomedical images usually consist of abnormal cellular tissues that closely resemble the background. Consequently, the incorporation of multi-scale semantic information becomes important in biomedical image analysis. Regrettably, existing networks do not make use of the fusion of cross-scale features.

To address these challenges, we present a novel end-to-end multi-task network for biomedical detection and segmentation. We use a backbone to extract a universal representation of input images, then a neck is used to fuse the multi-scale features generated by the backbone. Two task-specific decoders are used to handle segmentation and detection tasks, where unlike traditional approaches, we split the detection tasks (classification and regression) into different branches to improve the detection accuracy. In order to implement the task-interaction, we combine feature maps from segmentation and detection at different scales through a transformer layer, subsequently delivering the fused results to the respective decoder heads.

In summary, the main contributions of this paper are as follows: 1) We propose YOLO-Med, an efficient end-to-end multi-task network that jointly addresses the tasks of object detection and semantic segmentation in biomedical image analysis. Compared with other multi-task networks for biomedical images, YOLO-Med shows promising results in the trade-off between accuracy and speed. 2) We devise a cross-scale task-interaction module to facilitate interaction between the detection head and segmentation head from multiple scales as shown in Fig. 1. Also a decoupled detection head is adopted, which is first time used in multi-task networks for biomedical image detection and segmentation. 3) We validate YOLO-Med on two datasets, Kvasir-seg [21] and a large private dataset [15]. Our results achieve a promising performance across multiple metrics, confirming the effectiveness of YOLO-Med.

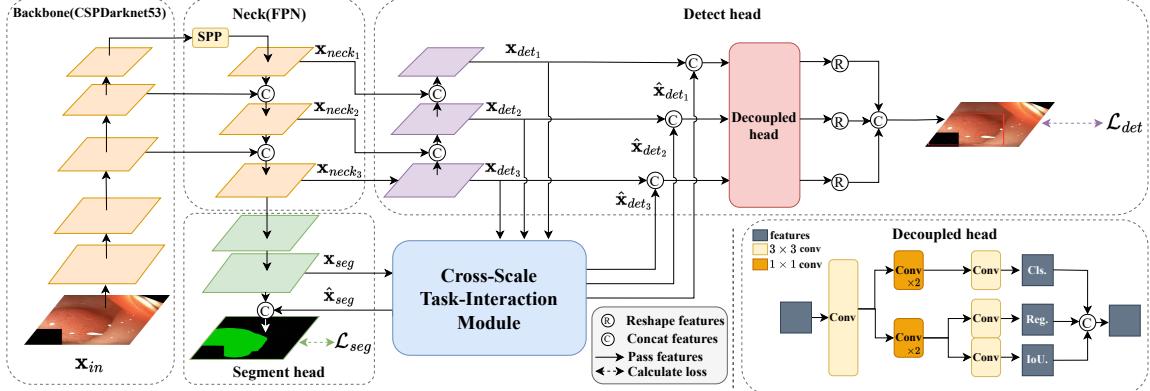


Fig. 2: The architecture of YOLO-Med network. YOLO-Med shares one encoder and combines 2 decoders with a cross-scale task-interaction module to solve different tasks. The encoder consists of a backbone and a neck, and the detection head has a decoupled head module.

2. METHOD

As shown in Fig. 2, YOLO-Med consists of a shared encoder consisting of two main components: the Backbone and the Neck.

Backbone. The backbone network extracts features from input images. We choose CSPDarknet53 [8] as it supports feature propagation and reuse, leading to a significant reduction in parameter and computational overhead during training. This choice guarantees real-time network performance. The initial image data $x_{in} \in \mathbb{R}^{H \times W \times 3}$ is input to the backbone.

Neck. The Neck is responsible for fusing the multi-scale features generated by the backbone. First, we pass the output of the backbone through an SPP (Spatial Pyramid Pooling) network [22] and subsequently feed it into the FPN (Feature Pyramid Network) [23]. The SPP module is utilized for feature generation and fusion across multiple scales, while the FPN module combines features from different semantic levels. This fusion process ensures that the resulting features encompass a rich blend of multi-scale and multi-semantic information. Within this module, we obtain three features with different scales: $x_{neck_i} \in \mathbb{R}^{\frac{H}{s_i} \times \frac{W}{s_i} \times c_i}$ where s_i denotes the scale parameter ranging from 8 to 32, and c_i represents the channel number of each feature map ranging from 128 to 512.

2.2. Decoders

In our network, the two heads are specific decoders for detection and segmentation.

Decoupled heads for detection. First, we construct a Path Aggregation Network (PAN) [24]. PAN operates as a bottom-up pyramid network, aligning with the top-down semantic propagation in FPN.

The diverse scale feature maps $x_{det_i} \in \mathbb{R}^{\frac{H}{s_i} \times \frac{W}{s_i} \times c_i}$ obtained from PAN are subsequently fused with the correspondingly scaled feature maps $\hat{x}_{det_i} \in \mathbb{R}^{\frac{H}{s_i} \times \frac{W}{s_i} \times c_i}$ generated by the cross-scale task-interaction module. These fused features serve as input to the final detection head. For our final detection head component, we opt for the decoupled head architecture [25]. This choice is based on the recognition that traditional coupled heads have demonstrated performance limitations due to inherent conflicts between classification

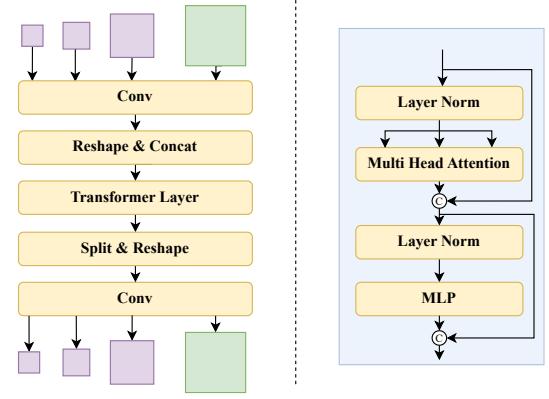


Fig. 3: The architecture of (a) cross-scale task-interaction module and (b) transformer layer

and regression tasks [26]. Therefore, within our multi-task architecture, we introduce decoupled heads to ensure that each task (classification and regression) does not negatively affect the others.

Segment head. For the segmentation head, we devise a straightforward top-down network structure. Initially, we take the feature map from the lowest level of the FPN as input. Following two rounds of feature integration and upsampling, the resulting feature map $x_{seg} \in \mathbb{R}^{\frac{H}{2} \times \frac{W}{2} \times 32}$ is merged with the feature map $\hat{x}_{seg} \in \mathbb{R}^{\frac{H}{2} \times \frac{W}{2} \times 32}$ generated by the cross-scale task-interaction module. This fused feature map is subsequently used as input for the final round of feature integration and upsampling.

2.3. Cross-scale task-interaction module

In this module, we combine features extracted by different decoders. As shown in Fig. 3, we initially merge the outputs of the three different scales from the PAN network in the detection head to obtain a token sequence v_{det} ,

$$v'_{det_i} = \text{Reshape}(\text{Conv}(x_{det_i})), \quad (1)$$

$$v_{det} = \| (v'_{det_i}), \quad (2)$$

where *Conv* is used to standardize the channel number of features x_{det_i} to 64. *Reshape* is applied to flatten the feature $v'_{det_i} \in \mathbb{R}^{\frac{H}{s_i} \times \frac{W}{s_i} \times 64}$ to a sequence $v'_{det_i} \in \mathbb{R}^{n_i \times 64}$ with $n_i = \frac{H}{s_i} \times \frac{W}{s_i}$. Subsequently, we concatenate ($\|$) the three token sequences to ob-

Table 1: Performance comparisons of segmentation (above) and Detection (below) on the Kvasir-seg dataset with different networks. The notation \uparrow : higher is better.

Model type	Model	PA(%) \uparrow	meanIoU(%) \uparrow	Speed(fps) \uparrow
Single-task	U-net [11]	83.37	75.60	11
	Polyp-PVT [10]	91.49	86.40	17
	Single-task baseline	90.95	86.24	41
Multi-task	UOLO [16]	83.41	75.48	9
	MULAN [17]	88.94	82.39	22
	Multi-task Baseline	90.88	85.73	36
	YOLO-Med(Ours)	97.32	88.64	31
Model type	Model	AP50(%) \uparrow	AP95(%) \uparrow	Speed(fps) \uparrow
Single-task	Faster-RCNN [27]	84.18	41.50	18
	RetinaNet [9]	90.95	65.47	17
	YOLOv5s	91.15	72.54	117
	Single-task baseline	91.31	72.66	47
Multi-task	UOLO [16]	75.86	38.73	9
	MULAN [17]	87.49	53.40	22
	Multi-task Baseline	89.73	67.11	36
	YOLO-Med(Ours)	94.72	73.02	31

tain the final token sequence $\mathbf{v}_{det} \in \mathbb{R}^{(n_1+n_2+n_3) \times 64}$.

Similarly, we convert the feature map \mathbf{x}_{seg} from the segment head into a token sequence \mathbf{v}_{seg} .

$$\mathbf{v}_{seg} = \text{Reshape}(\text{Conv}(\mathbf{x}_{seg})), \quad (3)$$

where Conv is used to transform $\mathbf{x}_{seg} \in \mathbb{R}^{\frac{H}{2} \times \frac{W}{2} \times 32}$ to $\mathbf{x}_{seg} \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times 64}$. Reshape is applied to flatten the feature $\mathbf{x}_{seg} \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times 64}$ to a sequence $\mathbf{v}_{seg} \in \mathbb{R}^{n_4 \times 64}$ with $n_4 = \frac{H}{4} \times \frac{W}{4}$.

We concatenate (\parallel) the two token sequences to obtain the final token sequence \mathbf{v} .

$$\mathbf{v} = \|(\mathbf{v}_{det}, \mathbf{v}_{seg}), \quad (4)$$

where $\mathbf{v} \in \mathbb{R}^{n \times 64}$ with $n = \sum_{i=1}^4 n_i$.

Next, we construct a Transformer Layer [28] with multi-head self attention (MHSA) as shown on the right side of Fig. 3.

$$Q = \text{MLP}(\mathbf{v}), K = \text{MLP}(\mathbf{v}), V = \text{MLP}(\mathbf{v}), \quad (5)$$

$$\mathbf{v}' = \text{MHSA}(Q, K, V) + \mathbf{v}, \quad (6)$$

$$\hat{\mathbf{v}} = \text{MLP}(\text{LN}(\mathbf{v}')) + \mathbf{v}', \quad (7)$$

$\hat{\mathbf{v}} \in \mathbb{R}^{n \times 64}$ with $n = \sum_{i=1}^4 n_i$ is the cross-scale task-interaction feature. Here, LN means LayerNorm and MLP is the linear layer.

Conversely, we employ *split* and *reshape* operations to obtain feature maps with sizes consistent with the input features. We then restore the channel number using a convolutional layer.

$$\hat{\mathbf{x}}_{det_i} = \text{Conv}(\text{Reshape}(\text{Split}(\hat{\mathbf{v}}))_i) \in \mathbb{R}^{\frac{H}{s_i} \times \frac{W}{s_i} \times c_i}, \quad (8)$$

$$\hat{\mathbf{x}}_{seg} = \text{Conv}(\text{Reshape}(\text{Split}(\hat{\mathbf{v}}))_4) \in \mathbb{R}^{\frac{H}{2} \times \frac{W}{2} \times 32}, \quad (9)$$

2.4. Loss function

The object detection loss (\mathcal{L}_{det}) comprises a weighted sum of the classification loss (\mathcal{L}_{class}), object loss (\mathcal{L}_{obj}), and bounding box loss (\mathcal{L}_{box}). As for segmentation loss (\mathcal{L}_{seg}), we utilize cross-entropy loss with logits (\mathcal{L}_{ce}). The global loss (\mathcal{L}_{global}) is as follows,

$$\mathcal{L}_{global} = \beta_1(\alpha_1 \mathcal{L}_{class} + \alpha_2 \mathcal{L}_{obj} + \alpha_3 \mathcal{L}_{box}) + \beta_2 \mathcal{L}_{ce}, \quad (10)$$

where $\alpha_1, \alpha_2, \alpha_3$ are uniformly set to $\frac{1}{3}$, and we set the weights of the detection loss and segmentation loss to be the same, with $\beta_1 = \beta_2 = \frac{1}{2}$. Both \mathcal{L}_{class} and \mathcal{L}_{obj} are implemented as focal loss [29]. Additionally, we employ the Localization Complete Intersection over Union (\mathcal{L}_{CIoU}) metric [30] for \mathcal{L}_{box} .

Table 2: Performance comparisons of segmentation (above) and detection (below) on our private dataset with different networks. The notation \uparrow : higher is better.

Model type	Model	PA(%) \uparrow	meanIoU(%) \uparrow	Speed(fps) \uparrow
Single-task	U-net [11]	78.24	57.94	10
	Polyp-PVT [10]	90.60	71.12	16
	Single-task Baseline	86.39	70.64	37
Multi-task	UOLO [16]	80.41	63.28	8
	MULAN [17]	89.76	71.05	19
	Multi-task Baseline	83.24	68.97	33
	YOLO-Med(Ours)	89.96	71.82	29
Model type	Model	AP50(%) \uparrow	AP95(%) \uparrow	Speed(fps) \uparrow
Single model	Faster-RCNN [27]	74.13	20.61	14
	RetinaNet [9]	76.80	29.63	14
	YOLOv5s	76.66	33.85	100
Multi-task model	Single-task Baseline	77.21	33.44	42
	UOLO [16]	60.85	21.26	8
	MULAN [17]	77.40	30.66	19
	Multi-task Baseline	76.30	33.21	33
	YOLO-Med(Ours)	78.56	35.43	29

3. EXPERIMENTS

3.1. Experimental Setting

Datasets. We conduct training using two datasets: Kvasir-seg [21], a publicly available dataset, and a novel private biomedical dataset [15]. Kvasir-seg comprises 1000 gastrointestinal disease images, each meticulously labeled for semantic segmentation and object detection. In addition, in collaboration with *Shanghai General Hospital*, we create a novel private biomedical dataset [15]. This dataset consists of images obtained through magnifying endoscopy with narrow-band imaging (ME-NBI) and includes annotations for both detection bounding boxes and polygon segmentation of gastric neoplastic lesions. It encompasses 3757 images collected from 392 patients, with annotations reviewed and verified by at least two experts. For both datasets, we adopt a split of 70% for training, 15% for validation, and 15% for testing.

Implementation details. Our network is trained using Stochastic Gradient Descent (SGD) optimization algorithm with a learning rate of 1×10^{-2} , weight decay of 5×10^{-4} and momentum of 0.937. Additionally, we employ the Cosine Annealing with Warm Restarts learning rate scheduling strategy, in which the first three epochs serve as warm-up epochs with a reduced learning rate. To initiate the training, we utilize a pre-trained model from the COCO dataset. All experiments are conducted on a single NVIDIA GeForce 2080Ti GPU.

Baseline models and Metrics. We comprehensively evaluate our network by comparing it with various biomedical multi-task networks, as well as networks specialized in either object detection or semantic segmentation tasks. For object detection, we consider high-performing models from recent years in the biomedical image domain, including RetinaNet [9], as well as iconic models like Faster-RCNN [27] and YOLOv5s, representing two-stage and one-stage networks respectively. In addition, we include a comparison with our single-task baseline model, which consists of only the encoder and the detection decoder. We evaluate our model's detection accuracy using mean Average Precision at 50% IoU (mAP50) and mean Average Precision at 95% IoU (mAP95) as metrics. Regarding the semantic segmentation task, our comparisons encompass classic architecture U-net [11], and Polyp-PVT [10] which utilizes transformer modules to enhance accuracy. We also include a comparison with the single-task baseline which comprises only the encoder and the segmentation decoder. We evaluate our model's segmentation accuracy using Pixel Accuracy (PA) and mean Intersection over Union (meanIoU) as metrics. In the realm of multi-task networks, we com-

Table 3: Ablation studies and analysis on Kvasir-seg (left) and our private dataset (right). Decoupled head (DH), Cross-Scale Task-Interaction (CSTI) module are the parts of our model. The notation ↑: higher is better. The w/ indicates "with".

Models	AP50(%)↑	AP95(%)↑	PA(%)↑	meanIoU(%)↑	Speed(fps)↑
Baseline	89.73	67.11	90.88	85.73	36
w/ DH	92.01	72.98	91.59	86.50	34
w/ CSTI	91.75	70.80	94.21	88.43	32
w/ DH+CSTI	93.78	73.02	94.32	88.56	31

pare our approach to the traditional UOLO [16], the latest MULAN [17], and the multi-task baseline of our model which includes only the encoder and two decoders. In addition to these horizontal comparisons with common networks, we conduct ablation experiments to investigate the impact of different modules within YOLO-Med, providing a detailed study of the network’s components.

3.2. Experimental results

Object detection results In the evaluation on the public dataset Kvasir-seg [21], as presented in Table 1, our model outperforms single-task networks such as Faster-RCNN [27], RetinaNet [9], YOLOv5s and our single-task baseline, as well as all three multi-task networks in terms of detection accuracy. Notably, our model demonstrates impressive real-time performance compared to other works, with only YOLOv5s surpassing it. However, it’s important to note that YOLOv5s lacks a segmentation decoder and a cross-scale task-interaction module. In the comparison between YOLO-Med and our single-task multi-task baseline models, all metrics indicate a higher level of object detection accuracy, with only a minimal decrease in inference speed. Similar results are observed on the private dataset as illustrated in Table 2.

Semantic segmentation results In the evaluation on the public dataset Kvasir-seg, as shown in Table 1, our network outperforms all three multi-task networks and three single-task networks. Furthermore, our network exhibit significantly superior real-time performance compared to both single-task or multi-task networks from other works, such as U-net [11], Polyp-PVT [10], UOLO [16] and MULAN [17]. When compared to our baseline models, all metrics surpass them, with an acceptable decrease in inference speed to enhance the segmentation accuracy. Similar results are observed on the large private dataset as shown in Table 2.

As depicted in Fig. 4, we conduct a qualitative performance analysis by comparing our network with UOLO [16] and MULAN [17] on Kvasir-seg [21]. Our network produces more accurate predictions for detection and segmentation, whether it involves multiple small objects (top), single small object (middle) and single huge object (bottom).

3.3. Ablation studies

In this section, we conduct four experiments, starting with a baseline and then introducing the Decoupled Head (DH) and the Cross-Scale Task-Interaction Module (CSTI) separately. We also evaluate a complete version that incorporates both modules. As presented in Table 3, from an accuracy perspective, the CSTI module has the most substantial positive impact on the segmentation task, with the network using only the CSTI module performing nearly as well as the complete version with both CSTI and DH modules. In contrast, the DH module is not able to bring huge improvements. The combined use of CSTI and DH yields the most substantial improvements. Regarding the detection task, the CSTI module alone brings noticeable improvements, while the DH module has a greater impact. Ultimately, the complete version with both CSTI and DH modules achieves the

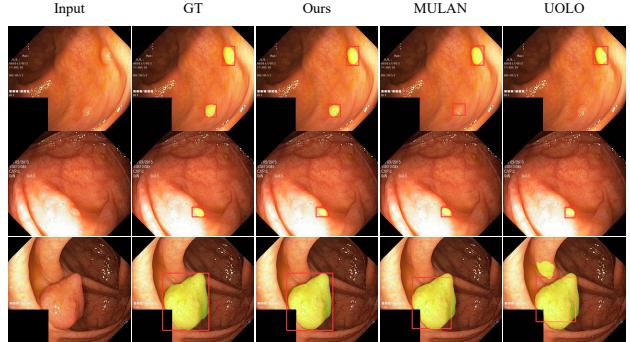


Fig. 4: Qualitative comparison with two multi-task networks MULAN [17] and UOLO [16] on Kvasir-seg [21]. The detection and segmentation results are shown in the same figure.

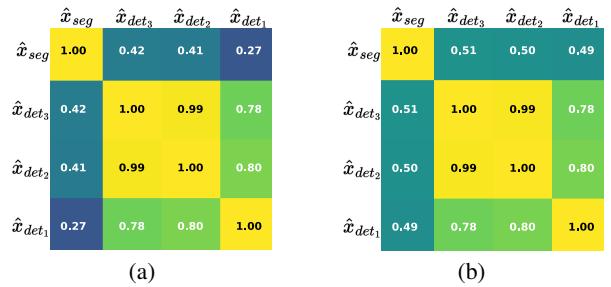


Fig. 5: Example correlation maps for the 4 outputs of the CSTI module. (a) depicts the correlation pattern for detecting and segmenting small objects, while (b) illustrates the scenario for large objects.

highest accuracy improvement. On the other hand, considering inference speed, utilizing the CSTI module alone leads to a 4 fps reduction compared to the baseline model, while employing only the DH module results in a 2 fps decrease due to its fewer parameters.

To enhance the CSTI module’s effectiveness, as depicted in Fig. 5, we conduct an analysis of correlations among its four outputs. Panel (a) presents results for detecting and segmenting small objects, while (b) for large objects. A comparison between them reveals correlations among the outputs of the detection and segmentation tasks. Notably, correlations within the detection task across different scales are consistently stronger than those between detection and segmentation tasks. However, the correlation between detection and segmentation tasks varies with object size. For small objects, the correlation between \hat{x}_{seg} and \hat{x}_{det_1} is only 0.27, whereas for large objects, this value increases to 0.49. These observations suggest that the CSTI module can dynamically adapt task relationships, effectively conveying information and ultimately enhancing overall performance.

4. CONCLUSION

In this paper, we present YOLO-Med, an efficient end-to-end multi-task network specifically designed to address both object detection and semantic segmentation tasks for biomedical image analysis. Our model excels in performance on two datasets: Kavarsir-seg and a private dataset. It not only achieves high accuracy in both tasks but also maintains real-time inference speed. Additionally, we validate the effectiveness of the proposed cross-scale task-interaction module, underscoring the value of cross-scale inter-task information fusion in the biomedical domain. This research carries significant implications for advancing future studies in the field of biomedical multi-task learning.

5. REFERENCES

- [1] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al., “The medical segmentation decathlon,” *Nature communications*, vol. 13, no. 1, pp. 4128, 2022.
- [2] Andre Esteva, Katherine Chou, Serena Yeung, Nikhil Naik, Ali Madani, Ali Mottaghi, Yun Liu, Eric Topol, Jeff Dean, and Richard Socher, “Deep learning-enabled medical computer vision,” *NPJ digital medicine*, vol. 4, no. 1, pp. 5, 2021.
- [3] Ayush Singhal, Manu Phogat, Deepak Kumar, Ajay Kumar, and Mamta Dahiya, “Study of deep learning techniques for medical image analysis: A review,” *Materials Today: Proceedings*, vol. 56, pp. 209–214, 2022.
- [4] Pranav Rajpurkar, Emma Chen, Oishi Banerjee, and Eric J Topol, “Ai in health and medicine,” *Nature medicine*, vol. 28, no. 1, pp. 31–38, 2022.
- [5] Mohammad Shehab, Laith Abualigah, Qusai Shambour, Muhamnad A Abu-Hashem, Mohd Khaled Yousef Shambour, Ahmed Izzat Alsalibi, and Amir H Gandomi, “Machine learning in medical applications: A review of state-of-the-art methods,” *Computers in Biology and Medicine*, vol. 145, 2022.
- [6] S Suganyadevi, V Seethalakshmi, and K Balasamy, “A review on deep learning in medical image analysis,” *International Journal of Multimedia Information Retrieval*, vol. 11, no. 1, pp. 19–38, 2022.
- [7] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, “You only look once: Unified, real-time object detection,” in *CVPR*, 2016.
- [8] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao, “Yolov4: Optimal speed and accuracy of object detection,” *arXiv preprint arXiv:2006.09928*, 2020.
- [9] Paul F Jaeger, Simon AA Kohl, Sebastian Bickelhaupt, Fabian Isensee, Tristan Anselm Kuder, Heinz-Peter Schlemmer, and Klaus H Maier-Hein, “Retina u-net: Embarrassingly simple exploitation of segmentation supervision for medical object detection,” in *Machine Learning for Health Workshop*, 2020, pp. 171–183.
- [10] Bo Dong, Wenhui Wang, Deng-Ping Fan, Jinpeng Li, Huazhu Fu, and Ling Shao, “Polyp-pvt: Polyp segmentation with pyramid vision transformers,” *arXiv preprint arXiv:2106.03234*, 2021.
- [11] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *MICCAI*, 2015, pp. 234–241.
- [12] Deng-Ping Fan, Ge-Peng Ji, Tao Zhou, Geng Chen, Huazhu Fu, Jianbing Shen, and Ling Shao, “Pranet: Parallel reverse attention network for polyp segmentation,” in *MICCAI*, 2020, pp. 263–273.
- [13] Ge-Peng Ji, Yu-Cheng Chou, Deng-Ping Fan, Geng Chen, Huazhu Fu, Debesh Jha, and Ling Shao, “Progressively normalized self-attention network for video polyp segmentation,” in *MICCAI*, 2021.
- [14] Tao Zhou, Yi Zhou, Kelei He, Chen Gong, Jian Yang, Huazhu Fu, and Dinggang Shen, “Cross-level feature aggregation network for polyp segmentation,” *Pattern Recognition*, vol. 140, pp. 109555, 2023.
- [15] Leheng Liu, Zhixia Dong, Jinnian Cheng, Xiongzhu Bu, Kaili Qiu, Chuan Yang, Jing Wang, Wenlu Niu, Xiaowan Wu, Jingxiang Xu, et al., “Diagnosis and segmentation effect of the mnb1-based deep learning model on gastric neoplasms in patients with suspected superficial lesions-a multicenter study,” *Frontiers in Oncology*, vol. 12, 2023.
- [16] Teresa Araújo, Guilherme Aresta, Adrian Galdran, Pedro Costa, Ana Maria Mendonça, and Aurélio Campilho, “Uolo-automatic object detection and segmentation in biomedical images,” in *MICCAI Workshop*, 2018, pp. 165–173.
- [17] Ke Yan, Youbao Tang, Yifan Peng, Veit Sandfort, Mohammad-hadi Bagheri, Zhiyong Lu, and Ronald M Summers, “Mulan: multitask universal lesion analysis network for joint lesion detection, tagging, and segmentation,” in *MICCAI*, 2019, pp. 194–202.
- [18] Yangyang Xu, Yibo Yang, and Lefei Zhang, “Demt: Deformable mixer transformer for multi-task learning of dense prediction,” in *AAAI*, 2023.
- [19] Shalayiding Sirejiding, Yuxiang Lu, Hongtao Lu, and Yue Ding, “Scale-aware task message transferring for multi-task learning,” in *ICME*, 2023, pp. 1859–1864.
- [20] Yuxiang Lu, Shalayiding Sirejiding, Yue Ding, Chunlin Wang, and Hongtao Lu, “Prompt guided transformer for multi-task dense prediction,” *arXiv preprint arXiv:2307.15362*, 2023.
- [21] Debesh Jha, Pia H Smedsrød, Michael A Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D Johansen, “Kvasir-seg: A segmented polyp dataset,” in *MMM*, 2020, pp. 451–462.
- [22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” *IEEE TPAMI*, vol. 37, no. 9, pp. 1904–1916, 2015.
- [23] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, “Feature pyramid networks for object detection,” in *CVPR*, 2017.
- [24] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia, “Path aggregation network for instance segmentation,” in *CVPR*, 2018, pp. 8759–8768.
- [25] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun, “Yolox: Exceeding yolo series in 2021,” *arXiv preprint arXiv:2107.08430*, 2021.
- [26] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar, “Rethinking the faster r-cnn architecture for temporal action localization,” in *CVPR*, 2018.
- [27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” *NeurIPS*, vol. 28, 2015.
- [28] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, and Neil Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *ICLR*, 2021.
- [29] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár, “Focal loss for dense object detection,” *TPAMI*, 2018.
- [30] Zhaohui Zheng, Ping Wang, Wei Liu, Jinze Li, Rongguang Ye, and Dongwei Ren, “Distance-iou loss: Faster and better learning for bounding box regression,” in *AAAI*, 2020.