

Semi-Supervised Multi-task Learning with Cross Pseudo Supervision for Gastrointestinal Lesion Analysis

Huda Dham Bader ^{1,*}, Ali Rakan Hasan Al-Jader ², and Mohammed Sabah Jarjees ²

¹ Department of Medical Physiology, College of Medicine, University of Mosul, Mosul, Iraq

² Department of Medical Instrumentation Techniques Engineering, Technical Engineering College of Mosul, Northern Technical University, Mosul, Iraq

Email: huda.badr@uomosul.edu.iq (H.D.B.); ali.rakan@ntu.edu.iq (A.R.H.A.-J.);

mohammed.s.jarjees@ntu.edu.iq (M.S.J.)

*Corresponding author

Abstract—This paper proposes a unified semi-supervised multi-task framework, Dual-Framework Cross Pseudo Supervision (DFCPS), that simultaneously performs segmentation and classification within a single shared-backbone network. Our method uses Cross Pseudo Supervision (CPS) with a confidence threshold of 0.95, updating every three training steps, and applies class-weighted losses to handle a severe 229:1 class imbalance. This design cuts computational cost by half compared to using separate models. On the Hyper-Kvasir dataset (11,662 images, 23 disease classes), our model achieves 68.37% mean Intersection over Union (mIoU) and 79.11% Dice for segmentation, and 75.49% accuracy with 57.49% balanced accuracy for classification. Our ablation studies show that removing Atrous Spatial Pyramid Pooling (ASPP) drops mIoU by 2.27%, removing class weights widens the imbalance gap by 1.33%, and running CPS at every step reduces mIoU by 5.26% while increasing training time by 62%. These results show our framework works well, but also show that class imbalance remains a tough open problem.

Keywords—endoscopic imaging, gastrointestinal lesion analysis, semi-supervised learning, multi-task learning, cross pseudo supervision, class imbalance

I. INTRODUCTION

Endoscopic imaging plays a central role in the diagnosis and assessment of gastrointestinal diseases in clinical practice [1]. Medical practitioners are the ones who analyze these images by hand to identify the abnormalities of internal organs, and it is a very time-consuming process, which can be subject to fatigue, thereby compromising the accuracy of the diagnosis [2]. More frequent has been the use of medical image segmentation as a technique of automatic lesion detection and clinical efficiency [3]. Segmentation of medical images is more difficult than natural images due to a variety of reasons; These are low Signal-to-Noise Ratio (SNR), low contrast, organ

deformation, and patient variation [4]. In addition, their sensitive nature, such as patient anatomy and diagnostic data, limits their distribution [5]. Thus, only small datasets can be found. Models in traditional methods take into account the localization and classification separately [6]. It consists of one network that determines the lesion location [7], and another that classifies the disease type [8]. Operating numerous models is expensive, restricts real-time operation, and does not allow the sharing of learned representations [9].

Medical data show the presence of a class imbalance, where certain diseases are significantly overrepresented compared to others, making them more common than others and influencing a model [10]. The dataset was unbalanced with a 229:1 ratio between classes, which forced the models to favor popular diseases over uncommon ones. Low-contrast tumors require enhancement [11]. Medical imaging requires fine-tuned convolutional filters [12]. Capsule endoscopy requires automated analysis [13]. Although there are many unlabeled images, an expert has to spend 30–60 min annotating each one of them. We solve those problems with a unified multi-task framework, and segmentation and classification are carried out simultaneously [9].

We adopt a combination of Cross Pseudo Supervision (CPS) [14], to utilize unlabeled data. Semi-supervised learning, which leverages both labeled and unlabeled data, is essential in medical imaging as it: (1) lowers annotation costs, (2) enhances generalization by learning from the underlying data distribution, (3) allows training effective models with few labeled samples, and (4) tackles data scarcity [15]. CPS uses the mutual supervision between network predictions, which differs from FixMatch [16], that uses a single teacher model. The idea is to use pseudo-labels that are generated by consistency regularization between two types of augmentation: a weak augmentation and a strong augmentation. Our results show that, due to the high-class imbalance, class-weighted loss

functions are used; however, this remains an open research problem.

Recently, Iqbal *et al.* [17] classified Gastrointestinal (GI) abnormalities with an accuracy of 97.75% by using a deep Convolutional Neural Network (CNN) known as HGANet. At the same time, a thorough survey by Zhang and Yang [18] indicated that multi-task learning can benefit the performance by transferring the knowledge to some related tasks, which is our idea. This work makes three main contributions. First, we propose a single-network design with a shared backbone that reduces computational cost by $2\times$ compared to separate models. Second, we integrate CPS with a confidence threshold $\tau = 0.95$ to exploit unlabeled images. Third, we analyze the successes and limitations of our approach, showing that segmentation generalizes well while classification struggles with rare classes due to extreme imbalance (229:1 ratio).

II. MATERIALS AND METHODS

A. Overview of the Model

Fig. 1 shows an overview of the proposed multi-task framework. A single ResNet-50 backbone [8], used as the shared feature extractor in a multi-task learning setting [19]

and pretrained on ImageNet [20], processes both labeled and unlabeled gastrointestinal images using shared weights. Task-specific heads perform segmentation and classification. CPS exploits unlabeled data using weak-strong augmentation consistency with confidence thresholding ($\tau = 0.95$). CPS is applied independently within each task to enhance representation learning. For unlabeled data, weak and strong augmentations are applied to the same sample. Weak augmentation includes horizontal flip and small rotation. Strong augmentation is color jitter, vertical flip, larger rotation, gaussian blur and random grayscale. Using pseudo-labels by confidence thresholding [16], the weakly augmented image is used to generate pseudo-labels. These pseudo-labels are used to supervise predictions on the strongly augmented version [14].

In our model, segmentation and classification work together through the shared ResNet-50 backbone. Early layers learn basic features like edges and textures that help both tasks, while deeper layers focus more on classifying the disease. By training both tasks at the same time, the backbone learns better representations than it would from either task alone. Also, the segmentation branch helps the classifier focus on the right areas of the image, acting like a built-in attention mechanism.

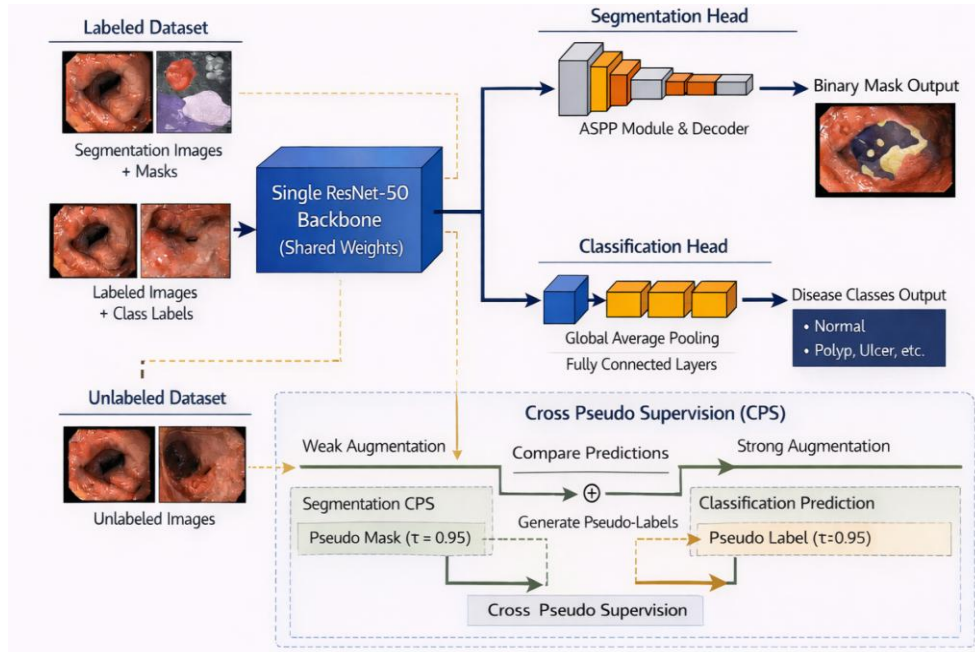


Fig. 1. CPS-based multi-task framework for GI image segmentation and classification.

B. Network Architecture

The segmentation branch maps the backbone features to pixel-level predictions [7]. Segmentation is treated as a binary problem that distinguishes the lesion areas from the background. Segmentation has three parts: an Atrous Spatial Pyramid Pooling (ASPP) module, decoder blocks, and an output head. The ASPP module employs atrous convolution at dilation rates of 6, 12, and 18 for multi-scale context information acquisition [21]. A global average pooling branch captures statistic level of the

image. The outputs are concatenated and fused through 1×1 convolution. The decoder upscales the features of the ASPP to the original resolution by means of bilinear interpolation. Feature processing occurs through two convolutional blocks with the addition of batch normalization and Rectified Linear Unit (ReLU) activation functions. Dropout layers with dropout rates of 0.5 and 0.1 are used for regularization. The model concludes with a 1×1 convolution operation to generate the binary segmentation logits.

The classification branch of the model follows the same features as the backbone. Global average pooling is applied to reduce the spatial dimensionality of the features, resulting in a 2048-dimensional feature vector. For the classification stage, the model uses three fully connected layers. The size of the data shrinks as it moves through the network: it starts at 2048, drops to 512, then to 256, and ends up at 23. Each of the layers is accompanied by the addition of batch normalization, ReLU activation, and dropout functions. The dropout rates used are 0.5 and 0.4, which are higher than normal to counter the effects of overfitting on minority classes [10]. The complete

architecture: shared ResNet-50 backbone ($7 \times 7 \times 2048$), ASPP with dilation rates $\{6, 12, 18\}$, decoder with bilinear upsampling ($\times 32$) to $224 \times 224 \times 1$ mask. Classification: global pooling to 2048-D, then FC layers $2048 \rightarrow 512 \rightarrow 256 \rightarrow 23$ with dropout (0.5, 0.4). CPS uses $\tau = 0.95$ with weak (flip, $\pm 10^\circ$ rotation) and strong (color jitter, blur) augmentation. Losses: BCE + Dice (seg) and weighted CE (Cls). The entire architecture of the model is outlined in Table I, which includes the specifications of the layers, their output sizes, and the functionality of the model.

TABLE I. SUMMARY OF MULTI-TASK DFCPS NETWORK COMPONENTS

Model	Component	Key Operation	Output Shape	Purpose
Shared Encoder (ResNet-50)	Backbone	Conv1→Layer4	$7 \times 7 \times 2048$	Feature extraction
Segmentation Head	ASPP Module	Atrous conv (6,12,18)	$7 \times 7 \times 256$	Multi-scale context
	Decoder	Bilinear upsample $\times 32$	$224 \times 224 \times 1$	Binary mask output
Classification Head	Global Pooling	AdaptiveAvgPool2d	$1 \times 1 \times 2048$	Spatial compression
	FC Layers	$2048 \rightarrow 512 \rightarrow 256 \rightarrow 23$	23	Disease classification
Cross Pseudo Supervision (CPS)	Weak Aug.	Flip, rotation 10°	$224 \times 224 \times 3$	Reliable pseudo-labels
	Strong Aug.	Flip, color jitter, blur	$224 \times 224 \times 3$	Robustness training
	Pseudo-Label	Confidence $\tau = 0.95$	Various	High-quality targets
Loss Functions	Segmentation	BCE + Dice Loss	Scalar	Mask accuracy
	Classification	Weighted CrossEntropy	Scalar	Class imbalance
Optimization	Optimizer	Adam, lr = 1×10^{-4}	-	Adaptive learning
	Scheduler	CosineAnnealingLR	-	LR decay to 1×10^{-6}

C. Loss Function

Three loss functions are used in the training. To segment, we will use a combination of Binary Cross-Entropy and Dice Loss that will solve the issue of the imbalance in the number of classes (foreground and background). Dice loss is calculated as in the case of the Dice loss given in Eq. (1):

$$Dice = 1 - \frac{2 \sum \sigma(\hat{y}) \cdot y + \varepsilon}{\sum \sigma(\hat{y}) + \sum y + \varepsilon} \quad (1)$$

where σ is the sigmoid and $\varepsilon = 1$ is a smoothing parameter [7].

To classify, weighted cross-entropy is used to deal with extreme class imbalance [22]. Weighting along with computing class weights, is done through balanced weighting (Eq. (2)) [22]:

$$w_c = \frac{N}{C \cdot n_c} \quad (2)$$

where N = the total number of samples, C = the total number of classes and n_c = the number of samples in class c . This makes sure that during the training; the minority classes are given importance. In the case of unlabeled images, the CPS loss can be determined as Cross Pseudo Supervision Loss (L_{CPS}) [14]. In this loss, pseudo-labels of weakly augmented samples are used as targets of strongly augmented samples. The confidence level τ will be 0.95 [10]. Those that are below this threshold are dropped since they are not confident. CPS is only run every three

iterations as opposed to every single step to be efficient.

The total loss is computed as Eq. (3):

$$Loss = L_{seg} + \lambda_{cls} \cdot L_{cls} + \lambda_{cps} \cdot L_{CPS} \quad (3)$$

where $\lambda_{cls} = 1.0$ and $\lambda_{cps} = 0.5$ are balancing hyperparameters [14].

III. EXPERIMENTAL SETUP

A. Datasets

We used the Hyper-Kvasir dataset [23], a large collection of gastrointestinal images containing 11,662 annotated images spanning 23 different disease classes. Segmented images include 1000 samples with pixel-level masks, and classification-only images include 10,662 samples with disease labels. An additional 9329 unlabeled images are available. Disease categories include Barrett's esophagus, polyps, ulcerative colitis, and anatomical landmarks. The distribution is heavily imbalanced where common categories contain over 1000 samples, but rare diseases have fewer than 10 samples. The most common class has 229 times more samples than the rarest one. We employed an 80/10/10 split for training, validation, and testing. The training phase uses 9329 labeled images and an additional 9329 unlabeled images for CPS. The validation set contains 1166 images. The test set contains 1167 images. The unlabeled set of 9329 images was used exclusively during training for CPS.

B. Experimental Details

The experimental server is equipped with NVIDIA Graphics Processing Units (GPUs). We employed

automatic mixed precision using a PyTorch implementation, which reduces memory consumption by 40% and accelerates training by 2×. The optimizer is Adam with learning rate $\eta = 10^{-4}$ and a cosine annealing scheduler. The minimum learning rate is set to 10^{-6} . Batch size is set to 20. The number of workers for data loading is 4. Pin memory is enabled for faster data transfer. All experiments use the same random seed of 42 for reproducibility. Early stopping with a patience of 10 epochs prevents overfitting [24]. Training terminates if validation performance does not improve for 10 consecutive epochs. The maximum number of epochs is set to 50 but actual training may stop earlier.

C. Evaluation Metrics

For segmentation, we evaluate using mean Intersection over Union (mIoU) and Dice Coefficient [10]. The mIoU (Eq. (4)) is computed as the average of IoU across all samples:

$$mIoU = \frac{1}{N} \sum_{i=1}^N \frac{TP_i}{TP_i + FP_i + FN_i} \quad (4)$$

The Dice coefficient (Eq. (5)) is computed as [10]:

$$Dice = \frac{2TP}{2TP + FP + FN} \quad (5)$$

where TP , FP , and FN are the True Positives, False Positives, and False Negatives, respectively. To classify them, we employ accuracy, balanced accuracy, and weighted F1-Score. The overall percentage of correct predictions is known as accuracy. Balanced Accuracy is the mean of per-class accuracies, referred to as Eq. (6) [10]:

$$Balanced\ Accuracy = \frac{1}{C} \sum_{c=1}^C \frac{TP_c}{TP_c + FN_c} \quad (6)$$

where C is specified as the number of classes and TP_c and FN_c are the true positives and false negatives of the class c . This measure is significant with skewed data. We present the Gap measure (Eq. (7)), which is the difference between the standard accuracy and balanced accuracy [10]:

$$Gap = Accuracy - Balanced\ Accuracy \quad (7)$$

A wide gap implies that the model works well in typical classes, but there is poor performance in rare classes. This is a measure of model bias in the majority classes.

IV. RESULTS AND DISCUSSION

Results on validation and test sets are presented in Table II. The model was able to reach mIoU 68.34% and Dice of 79.03% on validation data. These measures were consistent on the test set with a 68.37 % mIoU and 79.11 %

Dice. The gap in mIoU and Dice is just 0.03 and 0.08 percent, respectively, which shows that it does not overfit in generalization.

We noticed a clear sign of overfitting. The balanced accuracy dropped from 67.12% on validation (at epoch 14) to 57.49% on the test set, a loss of 9.63%. This happens because our training set has very few examples of rare diseases (some have fewer than 10 images). The class weights helped a lot with validation, but they didn't generalize well with new test data. This shows that simply using class weights is not enough when the minority classes have extremely few samples.

To classify, standard accuracy was on the test set at 75.49%. The balanced accuracy was, however, significantly lower at 57.49 %. This shows an issue of disparity in classes. The 18 % gap shows that the model works well on the common diseases but fails with the rare ones. The segmentation branch has been shown to perform on both the validation set and the test set. The low variance in validation and test measures proves that the model acquired features that are generalizable. The Dice coefficient of 79.11% is comparable to the state of the art [7].

TABLE II. OVERALL MODEL PERFORMANCE ON VALIDATION AND TEST SETS

Model Task	Metric	Validation	Test	Difference
Segmentation	mIoU (%)	68.34	68.37	+0.03
	Dice (%)	79.03	79.11	+0.08
	Loss	0.4923	0.4112	-0.0811
Classification	Accuracy (%)	77.53	75.49	-2.04
	Balanced Acc (%)	67.12	57.49	-9.63
	F1-Score (%)	77.33	75.73	-1.60
	Gap (%)	10.41	18.00	+7.59

The ASPP module has been demonstrated to be effective in the generation of multi-scale features that is critical to the processing of lesions with different sizes [21]. The model, however, is sometimes not good with small lesions that occupy less than 5% of the image area. Fuzzy lesions are also difficult. These results of classification present a more intricate picture. The test set score of 75.49% overall accuracy seems reasonable, but a test set balance of 57.49% shows a lot of weaknesses. Our class weighting scheme assigned weights computed using balanced weighting, which produced values ranging from 0.442 for the most common class to 101.402 for the rarest, which is a ratio of 229:1 [24].

TABLE III. CLASSIFICATION METRICS PROGRESSION DURING TRAINING

Epoch	Accuracy (%)	Balanced Acc (%)	Gap (%)
1	75.73	57.72	18.01
5	79.33	63.21	16.12
14 (Best)	77.53	67.12	10.41
24 (Final)	76.07	61.73	14.34

Classification performance as training progressed can be seen in Table III. Balanced accuracy peaked at 67.12% at the 14th epoch, then fell to 61.73% at the 24th epoch.

This indicates that the classification branch is overfitting the training data. Additionally, the gap increased from 10.41% in validation to 18% in the test set. This means the few rare-class examples in training weren't diverse enough to help the model generalize to new data [21].

However, the gap widened considerably on the test set to 18%, which indicates that improvements in balanced accuracy during training did not fully generalize to unseen data. The gap between standard accuracy and balanced accuracy fluctuated between 10% and 18% throughout training. This persistent gap indicates that class weights alone were insufficient to fully address the 229:1 imbalance ratio. The model consistently achieved higher standard accuracy, suggesting strong performance in common classes but limited learning on rare conditions. The widening gap from 10.41% on validation to 18% on test data suggests that the few rare class samples in the training set lacked sufficient diversity for generalization [25]. We analyzed the behavior of Cross Pseudo Supervision during training by examining the CPS loss trajectory and its relationship to model performance. Table IV indicates the CPS loss over time over major training periods. CPS was used upon every 3rd iteration with a lower weight of $\lambda_{cps} = 0.5$ to strike the right balance between stability and regularization.

TABLE IV. CPS LOSS PROGRESSION DURING TRAINING

Epoch	CPS Loss	Total Train Loss
1	0.287	2.629
8	0.529	1.160
14 (Best)	0.616	0.941

The CPS loss increased and reached 0.287 at epoch 1, and 0.616 at epoch 14, indicating that during training, the network generated more accurate pseudo labels. This increase in CPS loss is an indication of increasing disagreement between weak and strong augmentation predictions that yields a stronger regularization signal. The overall supervised loss has steadily decreased from 2.629 to 0.941, showing successful learning. We used about the same number of labeled and unlabeled samples (9329 each). CPS usually benefits from 5–10× the amount of unlabeled data [26]. Only high-confidence pseudo-labels (confidence threshold of 0.95) were used for training. The use of CPS at every 3rd iteration instead of every iteration reduced the amount of computational overhead without

sacrificing the regularization benefit. The small CPS weight allowed the supervised signal not to be overwhelmed by pseudo-label noise [14].

The training progression was studied to see the evolution of the metrics of segmentation and classification over epochs. Table V shows some important moments throughout the 24 epochs of training until early stopping was activated [24].

The segmentation metrics steadily increased during training, reaching a maximum of 69.80% mIoU at epoch 24. However, the classification balanced accuracy reached its highest level of 67.12% at epoch 14 and dropped to 61.73% at epoch 24, which suggests overfitting on the classification task. The decision to stop early was motivated by the discrepancy between the improving segmentation and decreasing classification after epoch 14. The metric of mIoU and balanced accuracy was combined, reflecting both tasks simultaneously, for model selection. According to Tarvainen and Valpola [24], the computation time of 26 epochs of early stopping at epoch 24 was saved, which is around 4.2 h.

TABLE V. TRAINING PROGRESSION AT KEY MILESTONES (VALIDATION SET)

Epoch	mIoU (%)	Dice (%)	Balanced Acc (%)	F1-Score (%)
1	53.07	66.46	57.72	74.15
5	64.63	76.34	63.21	78.09
14 (Best)	68.34	79.03	67.12	77.33
24 (Final)	69.80	80.05	61.73	75.16

We performed a large number of ablation studies on different core components, CPS hyperparameters, loss functions, and noise robustness to systematically investigate our design decisions.

A. Ablation of Core Components

We conducted an ablation study to measure the contribution of each component by removing one component at a time. The results are shown in Table VI: the mIoU for the ablation baseline (A1: 67.26%) is close to the values seen in the main results (68.37%), as can be expected ($\pm 1.5\%$) because of the unpredictability of using different GPUs [27].

TABLE VI. ABLATION STUDY OF CORE COMPONENTS (TEST SET)

Experiment	Configuration	mIoU (%)	Dice (%)	BalAcc (%)	Gap (%)
Full Model (Main)	CPS + CW + ASPP + Shared	68.37	79.11	57.49	18.00
A1: Full Model (Ablation)	CPS + CW + ASPP + Shared	67.26	78.27	60.34	17.81
A2: No CPS	Supervised only	68.96	79.80	61.93	17.94
A3: No Class Weights	Uniform weights	65.01	-	61.06	19.14
A4: No ASPP	Simple decoder	66.10	77.53	61.88	18.75
A5: No Shared Backbone	Separate models	64.71	75.81	60.07	18.25

It seems removing CPS (A2) provided better results, as seen in Table VI, mIoU increased from 67.26% to 68.96% and balanced accuracy from 60.34% to 61.93%. This indicates that there is no major effect of CPS in the case of rare diseases. The mIoU was decreased by 2.25% without

class weights (A3), and the balanced accuracy remained roughly the same, indicating that weights are more useful for segmentation than classification. For lesions of varying sizes, we saw a 1.16% drop in mIoU when we removed ASPP (A4). Shared backbone (A5) reduced the mIoU by

2.55% and quadrupled the training time, which demonstrates the advantage of sharing features across tasks.

B. Effect of Confidence Threshold

We explored different confidence thresholds for the CPS, τ ranges between 0.8 and 0.95 (Table VII). Lower thresholds increase pseudo-labels, leading to noise.

TABLE VII. EFFECT OF CONFIDENCE THRESHOLD τ (TEST SET)

τ	mIoU (%)	Dice (%)	BalAcc (%)	Gap (%)
0.8	63.91	74.75	64.08	12.10
0.9	61.72	73.36	63.22	14.25
0.95	68.37	79.11	57.49	18.00

The performance with respect to the confidence threshold τ is shown in Table VII. Lower thresholds like 0.8 or 0.9 let in more pseudo-labels but add noise, dropping mIoU by 4–6% (63.91% and 61.72% vs. 68.37%). The best segmentation is achieved by the highest threshold ($\tau = 0.95$), but the classification gap increases to 18%. This is because pseudo-labels with high confidence mostly are derived from common diseases which can help segmentation, but not too much for the rare classes.

C. Effect of CPS Frequency

The frequency (freq) of CPS updates on performance is indicated in Table VIII. If CPS is run every iteration (freq = 1), that is detrimental for segmentation (mIoU is reduced to 63.11% from 68.37%) and slows training 62% (1024 s vs. 634 s). Why? Frequent updating will make the model learn from noisy pseudo labels before it is ready, which will cause the shared backbone to be confused. Using CPS every 5 iterations (freq = 5) reduces this time to 544 s/epoch but the regularization is too weak, which results in a low mIoU value of 64.71%. Every 3 iterations is the best option, considering speed and accuracy.

TABLE VIII. EFFECT OF CPS FREQUENCY

CPS Frequency	Time Per Epoch	mIoU (%)	BalAcc (%)
Every 1 iteration	1024s	63.11	59.56
Every 3 iterations	634s	68.37	57.49
Every 5 iterations	544s	64.71	60.07

D. Alternative Loss Functions for Class Imbalance

We tested four loss functions to deal with the class imbalance: weighted cross-entropy (baseline), Focal Loss ($\gamma = 2$ and $\gamma = 0.5$), and label smoothing. Results are shown in Table IX.

TABLE IX. COMPARISON OF LOSS FUNCTIONS

Loss Function	BalAcc (%)	Gap (%)
Weighted CE (Ours)	57.49	18.00
Focal Loss ($\gamma = 2$)	57.37	10.58
Focal Loss ($\gamma = 0.5$)	62.46	13.29
Label Smoothing	62.69	17.43

For classification, Focal Loss with $\gamma = 0.5$ obtains the highest balanced accuracy (62.46%) and smallest gap (13.29%) among all the methods. However, the weighted CE is better for segmentation (68.37% mIoU). Even the

ongoing disparity among the various methods renders it clear that loss-based approaches by themselves cannot resolve the imbalance of 229:1.

E. Robustness to Pseudo-Label Noise

To test robustness, 5% of random noise was added to pseudo-labels. Table X shows the model barely flinched: mIoU dropped only 0.20% (from 68.37% to 68.17%), and Dice dropped 0.11% (79.11% to 79.00%). Interestingly, the balanced accuracy also increased marginally (57.49% to 60.08%), likely the noise served as a form of regularization.

TABLE X. EFFECT OF PSEUDO-LABEL NOISE

Noise Level	mIoU (%)	Dice (%)	Bal Acc (%)
0% (Ours)	68.37	79.11	57.49
5% Noise	68.17	79.00	60.08

F. FixMatch Baseline Implementation

As a baseline, FixMatch was also trained with the same configuration: ResNet-50 backbone, confidence threshold $\tau = 0.95$, Adam optimizer ($\text{lr} = 1 \times 10^{-4}$), cosine annealing and early stopping. Training ended prematurely at epoch 8, resulting in a test balanced accuracy of 57.77% and an F1-Score of 73.31%. These numbers are very close to the classification branch of this model (57.49% Bal Acc), demonstrating that the multi-task design did not hurt the classification performance and at the same time also provides segmentation for free.

G. Comparison with State-of-the-Art

Comparing with previous methods is a bit difficult due to different datasets and tasks. HyGe-Net [17] and GISEgNet [28] achieves results on Kvasir with only 8 disease classes, while Hyper-Kvasir used here has 23 classes with a severe imbalance of 229:1. Single task problems (classification or segmentation), rather than both, are handled in Mean Teacher [24] and Uncertainty-Aware Mean Teacher (UA-MT) [29]. This method is qualitatively compared to other methods in the literature in Table XI.

TABLE XI. COMPARISON WITH STATE-OF-THE-ART METHODS

Method	Dataset	Task	Metric	Score
Mean Teacher [24]	SVHN	Cls.	Error%	4.35
UA-MT [29]	Left Atrium	Seg.	Dice%	88.88
FixMatch [16]	CIFAR-10	Cls.	Error%	5.07
HGANet [17]	Kvasir (8-class)	Cls.	Acc%	97.75
GISEgNet [28]	Kvasir	Seg. + Cls.	mIoU/Acc	75.13/93.16
FixMatch (ours)	Hyper-Kvasir	Cls.	BalAcc	57.77
Ours	Hyper-Kvasir	Seg. + Cls.	mIoU/BalAcc	68.37/57.49

V. CONCLUSION

In this paper, a multi-task framework for GI lesion segmentation and classification was proposed that

leverages the shared ResNet-50 backbone, CPS for unlabeled data, and class-weighted losses. The model achieves 68.37% mIoU and 79.11% Dice for segmentation, but classification lags with 57.49% balanced accuracy and an 18% gap, highlighting the challenge of rare disease classification. Ablation results show that the shared backbone cuts computational cost by half. Removing CPS improved balanced accuracy, suggesting CPS reinforces common class bias. Class weights overfit to rare classes (67.12% validation vs 57.49% test). Focal Loss ($\gamma = 0.5$) gives better balanced accuracy (62.46%) than weighted CE. The optimal CPS uses $\tau = 0.95$ every 3 iterations. Limitations include cross-dataset validation, generalization on external datasets, and direct comparison with GISegNet on the same dataset. This remains as future work.

CONFLICT OF INTEREST

The authors declare no competing interests.

AUTHOR CONTRIBUTIONS

H.D.B. conceived and designed the study, developed the model, and wrote the manuscript. A.R.H.A.-J. performed data analysis, result interpretation, and validation. M.S.J. supervised the work, reviewed the manuscript, and provided critical revision. All authors reviewed and approved the final manuscript.

FUNDING

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ACKNOWLEDGMENT

The authors acknowledge the use of the Hyper-Kvasir dataset and the computational resources provided.

REFERENCES

- [1] M. Barreiro-de Acosta, A. Gutierrez, I. Rodríguez-Lago *et al.*, "Recommendations of the Spanish Working Group on Crohn's disease and Ulcerative Colitis (GETECCU) on dysplasia screening in inflammatory bowel disease patients," *Gastroenterol. Hepatol.*, vol. 42, no. 9, pp. 568–578, 2019. doi: 10.1016/j.gastre.2019.10.002
- [2] X. Liu, L. Song, S. Liu, and Y. Zhang, "A review of deep-learning-based medical image segmentation methods," *Sustainability*, vol. 13, no. 3, 1224, 2021.
- [3] Y. Xu, R. Quan, W. Xu, Y. Huang, X. Chen, and F. Liu, "Advances in medical image segmentation: A comprehensive review of traditional, deep learning and hybrid approaches," *Bioengineering*, vol. 11, no. 10, 1034, 2024.
- [4] A. S. Panayides *et al.*, "AI in medical imaging informatics: Current challenges and future directions," *IEEE J. Biomed. Health Inform.*, vol. 24, no. 7, pp. 1837–1857, Jul. 2020. doi: 10.1109/JBHI.2020.2991043
- [5] G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, "Secure, privacy-preserving and federated machine learning in medical imaging," *Nat. Mach. Intell.*, vol. 2, no. 6, pp. 305–311, Jun. 2020. doi: 10.1038/s42256-020-0186-1
- [6] J. Bai *et al.*, "Localizing from classification: Self-directed weakly supervised object localization for remote sensing images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 12, pp. 17935–17949, Dec. 2024. doi: 10.1109/TNNLS.2023.3309889
- [7] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.
- [8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90
- [9] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 27, no. 1, pp. 41–75, Apr. 1997. doi: 10.1023/A:1007379606734
- [10] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Netw.*, vol. 106, pp. 249–259, Oct. 2018. doi: 10.1016/j.neunet.2018.07.011
- [11] H. D. Bader and M. S. Jarjees, "Hybrid CNN-transformer architecture for robust liver tumor segmentation in 2D CT slices," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 8, no. 3, pp. 983–999, Mar. 2026.
- [12] Z. Farej, A. F. Sheet, and N. M. Sheet, "On the classification accuracy of wireless capsule endoscopy images for small intestine bleeding detection using convolutional neural network," *J. Image Graph.*, vol. 12, no. 4, pp. 450–457, Dec. 2024. doi: 10.18178/joig.12.4.450-457
- [13] Z. K. Farej, A. F. Sheet, and N. M. Sheet, "Review on wireless capsule endoscopy system issues, challenges, and technologies," *Przegląd Elektrotechniczny*, vol. 99, no. 12, pp. 130–137, Dec. 2023.
- [14] X. Chen, Y. Yuan, G. Zeng, and J. Wang, "Semi-supervised semantic segmentation with cross pseudo supervision," in *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2613–2622.
- [15] X. Zhang, J. Wang, J. Wei, X. Yuan, and M. Wu, "A review of non-fully supervised deep learning for medical image segmentation," *Information*, vol. 16, no. 6, 433, Jun. 2025.
- [16] K. Sohn *et al.*, "FixMatch: Simplifying semi-supervised learning with consistency and confidence," *Advances in Neural Information Processing Systems*, vol. 33, 596–608, 2020.
- [17] I. Iqbal, K. Walayat, M. U. Kakar, and J. Ma, "Automated identification of human gastrointestinal tract abnormalities based on deep convolutional neural network with endoscopic images," *Intell. Syst. Appl.*, vol. 16, 200149, Dec. 2022. doi: 10.1016/j.iswa.2022.200149
- [18] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 12, pp. 5586–5609, Dec. 2022. doi: 10.1109/TKDE.2021.3070203
- [19] S. Ruder, "An overview of multi-task learning in deep neural networks," *arXiv Preprint*, arXiv:1706.05098, 2017. doi: 10.48550/arXiv.1706.05098
- [20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Miami, FL, USA, Jun. 2009, pp. 248–255. doi: 10.1109/CVPR.2009.5206848
- [21] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018. doi: 10.1109/TPAMI.2017.2699184
- [22] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 9260–9269, doi: 10.1109/CVPR.2019.00949
- [23] H. Borgli *et al.*, "HyperKvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy," *Sci. Data*, vol. 7, no. 1, 283, Aug. 2020. doi: 10.1038/s41597-020-00622-y
- [24] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Long Beach, CA, USA, Dec. 2017, vol. 30, pp. 1195–1204.
- [25] J. M. Johnson and T. M. Khoshgohar, "Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, 27, Dec. 2019. doi: 10.1186/s40537-019-0192-5

- [26] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Data Min. Knowl. Disc.*, vol. 34, no. 2, pp. 373–440, Mar. 2020. doi: 10.1007/s10618-019-00655-7
- [27] E. Badash, D. Boneh, I. Komargodski, and M. Srivastava, "Hawkeye: Reproducing GPU-level non-determinism," arXiv Preprint, arXiv:2603.20421, 2026.
- [28] A. Şener and B. Ergen, "Automatic detection of gastrointestinal system abnormalities using deep learning-based segmentation and classification methods," *Health Inf. Sci. Syst.*, vol. 13, no. 1, 25, Dec. 2025. doi: 10.1007/s13755-025-00354-6.
- [29] L. Yu, S. Wang, X. Li, C.-W. Fu, and P.-A. Heng, "Uncertainty-aware self-ensembling model for semi-supervised 3D left atrium segmentation," in *Proc. Medical Image Computing and Computer Assisted Intervention – MICCAI*, 2019, vol. 11765, 2019, pp. 605–613. doi: 10.1007/978-3-030-32245-8_67

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](#)).