Learn to Threshold: ThresholdNet with Confidence-Guided Manifold Mixup for Polyp Segmentation

Xiaoqing Guo, Chen Yang, Yajie Liu, and Yixuan Yuan

***Abstract*— The automatic segmentation of polyp in endoscopy images is crucial for early diagnosis and cure of colorectal cancer. Existing deep learning-based methods for polyp segmentation, however, are inadequate due to the limited annotated dataset and the class imbalance problems. Moreover, these methods obtained the final polyp segmentation results by simply thresholding the likelihood maps at an eclectic and equivalent value (often set to 0.5). In this paper, we propose a novel ThresholdNet with a confidence-guided manifold mixup (CGMMix) data augmentation method, mainly for addressing the aforementioned issues in polyp segmentation. The CGMMix conducts manifold mixup at the image and feature levels, and adaptively lures the decision boundary away from the under-represented polyp class with the confidence guidance to alleviate the limited training dataset and the class imbalance problems. Two consistency regularizations, mixup feature map consistency (MFMC) loss and mixup confidence map consistency (MCMC) loss, are devised to exploit the consistent constraints in the training of the augmented mixup data. We then propose a two-branch approach, termed ThresholdNet, to collaborate the segmentation and threshold learning in an alternative training strategy. The threshold map supervision generator (TMSG) is embedded to provide supervision for the threshold map, thereby inducing better optimization of the threshold branch. As a consequence, ThresholdNet is able to calibrate the segmentation result with the learned threshold map. We illustrate the effectiveness of the proposed method on two polyp segmentation datasets, and our methods achieved the state-of-the-art result with 87.307% and 87.879% dice score on the EndoScene dataset and the WCE polyp dataset. The source code is available at** `https://github.com/Guo-Xiaoqing/ThresholdNet`.

***Index Terms*— Polyp segmentation, CGMMix data augmentation, Consistency regularization, ThresholdNet, TMSG module.**

I. Introduction

COLORECTAL cancer (CRC) is the third most commonly diagnosed cancer and the second most common cause of cancer deaths in the United States, with 147,950 new cases being estimated and 53,200 deaths being caused by CRC in 2020 [1]. Though the survival rate is low if cancer has spread outside the colorectum, early diagnosis and proper treatment can lead to a high cure rate with a favorable survival rate of 90% [1]. CRC usually arises from adenomatous polyps, and a polyp may take 10 to 15 years to develop into cancer if left untreated [2]. Hence, detecting and removing polyps before they become malignant can significantly reduce mortality rates. Regular screening is a widely used procedure in hospital to identify the adenomatous polyps and prevent CRC [2]. But such screenings are manually performed by clinicians, therefore affected by human factors such as experience, leading to subjective diagnosis. A possible solution is to design automatic polyp segmentation models with great accuracy and sensitivity, which could aid clinicians during the screening procedure.

In recent years, many deep learning approaches have shown prominent performance for the polyp segmentation [3]–[22]. However, it remains an unsolved challenge with two main limitations. Firstly, annotated data in the medical domain is limited, especially in polyp segmentation that requires pixel-wise annotations. Such manual segmentation requires professional medical knowledge as well as a high degree of concentration, and even skilled clinicians may fail to reach a consensus on the manual segmentation results. The limited annotated data in polyp segmentation thus leads to overfitting problems and becomes a bottleneck to deep learning-based methods. Secondly, existing methods [6]–[22] simply acquire the segmentation result by thresholding the predicted likelihood map with 0.5 during the testing phase, ignoring the fact that different thresholds lead to varying results. Intuitively, Fig. 1 illustrates segmentation results obtained with different thresholds, and the best results of (a), (b), (c), (d) images are achieved when the threshold equals to 0.1, 0.9, 0.1 and 0.9, respectively. Hence, thresholding the likelihood maps with an arbitrary constant is insufficient.

To tackle the aforementioned challenges, we delicately design a ThresholdNet with the CGMMix data augmentation method. For the first challenge, to mitigate the overfitting problem and enhance the generalization of the trained model, one dominant solution is data augmentation. Mixup is a recently-proposed data augmentation method to *generate extra training samples by applying linear combinations of training images and labels* [23]–[31]. However, mixup encourages the model to center the decision boundary between classes, ignoring the

This work was supported by Hong Kong Research Grants Council (RGC) Early Career Scheme grant 21207420, Hong Kong RGC Collaborative Research Fund grant C4063-18GF, and National Natural Science Foundation of China (62001410). *(Corresponding author: Yixuan Yuan)*

X. Guo, C. Yang and Y. Yuan are with the Department of Electrical Engineering, City University of Hong Kong, Hong Kong SAR, China (e-mail: xqguo.ee, cyang.ee@my.cityu.edu.hk; yxyuan.ee@cityu.edu.hk).

Y. Liu is with the Department of Radiation Oncology, Peking University Shenzhen Hospital, Shenzhen, China (e-mail: liuyajie@pkuszh.com).

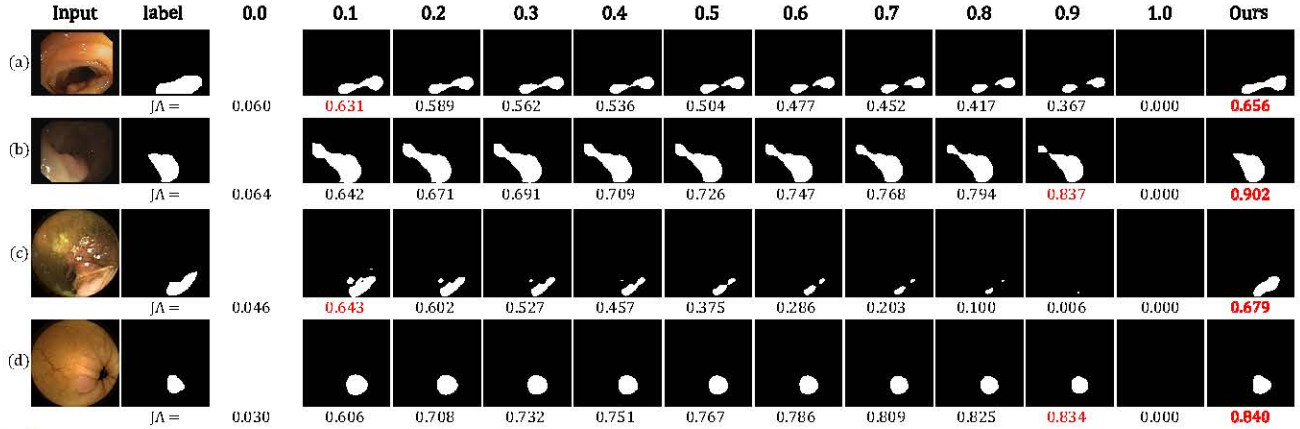

Fig. 1: Illustration of segmentation results obtained through different thresholds (constant threshold from 0.0 to 1.0 and our learned threshold map). If the likelihood of polyp class exceeds the threshold, then this pixel is categorized into the polyp class. Images in rows (a-b) are sampled from the EndoScene dataset, while images in rows (c-d) are from WCE polyp dataset.

class imbalance problem existing in the polyp segmentation task. It is reported that only around 5% pixels belong to the polyp class in the collected endoscopy images, and this class imbalance tends to group unseen foreground samples towards the background class [29]. To remedy this drawback of mixup, we propose confidence-guided manifold mixup (CGMMix) to asymmetrically and adaptively adjust the decision boundary close to the background class and stay away from the polyp class. Specifically, different from the original mixup method, which directly implements linear behavior in the label space, our CGMMix integrates the clinical consideration and confidence map to adaptively threshold the soft mixup label, further obtaining the ground truth of mixup images. Moreover, CGMMix jointly conducts linear combination on image and feature levels to enhance the smooth behavior between training samples. In addition, two regularization losses, a mixup feature map consistency (MFMC) loss and a mixup confidence map consistency (MCMC) loss, are proposed to ensure the deep supervision of the mixup sample at multiple feature levels.

To better threshold the predicted likelihood map for accurate segmentation, a feasible way is to *automatically learn the threshold map that indicates the threshold value for each position*. Only one work [32] investigated this problem by introducing a threshold loss. To explicitly learn the correct threshold map, we propose a ThresholdNet to adaptively adjust the threshold value of the corresponding likelihood map for each pixel. Specifically, we design a two-branch network, where a threshold branch is introduced to be parallel to the segmentation branch. The threshold branch utilizes semantic information extracted from the base network to decode and obtain the threshold map, while the segmentation branch decodes the same semantic information to predict the likelihood map. These two branches are updated with a strategy of alternative optimization, guaranteeing global convergence. Moreover, we propose a threshold map supervision generator (TMSG) to obtain the ground truth of threshold map, thereby inducing better supervision of the threshold branch. Adaptively adjusting threshold for each position, ThresholdNet can explicitly calibrate the final segmentation results by dynamically thresholding the predicted likelihood map, thus final predictions are more aligned with ground truths.

In this paper, we propose a two-branch ThresholdNet with the CGMMix for the polyp segmentation. The main contributions of this paper can be summarized into three aspects:

- In order to prevent overfitting and tackle the class imbalance problem, we propose a novel CGMMix method to augment limited training data with multi-level confidence guidance, especially to enrich polyp information. Moreover, a MFMC loss and a MCMC loss are proposed to enhance the supervision for mixup data, thus promoting segmentation performance.
- We develop a novel two-branch ThresholdNet, which simultaneously predicts the likelihood map and the threshold map. Different from existing methods that simply acquire the segmentation result by thresholding the predicted likelihood map with 0.5, ThresholdNet is devised to predict the threshold value for each position, further to adaptively adjust the predicted likelihood map and rectify final segmentation result.
- We validate the effectiveness of our approach and conduct ablation studies to investigate the proposed ThresholdNet with CGMMix on two polyp segmentation datasets. Extensive experiments demonstrate that the proposed method shows superiority to state-of-the-art polyp segmentation methods.

II. Related Work

A. Deep Learning for Polyp Segmentation

Deep convolutional neural networks (CNN), demonstrating superior feature representation capability, are widely utilized in the automatic polyp detection and segmentation [3]–[22]. Vázquez et al. [6] straightforwardly introduced the fully convolutional network to polyp segmentation task, which is the first attempt to apply deep learning method on the polyp segmentation and serves as a benchmark. Qadir *et al.* [12] adopted Mask R-CNN to joint polyp detection and segmentation learning. To better segment polyps with various sizes and features in image space, connections and aggregations between high and low-level features were considered for accurate prediction [7], [9], [17], [21], [22]. Wickstrøm *et al.* [17] utilized the pooling indices computed in the encoder

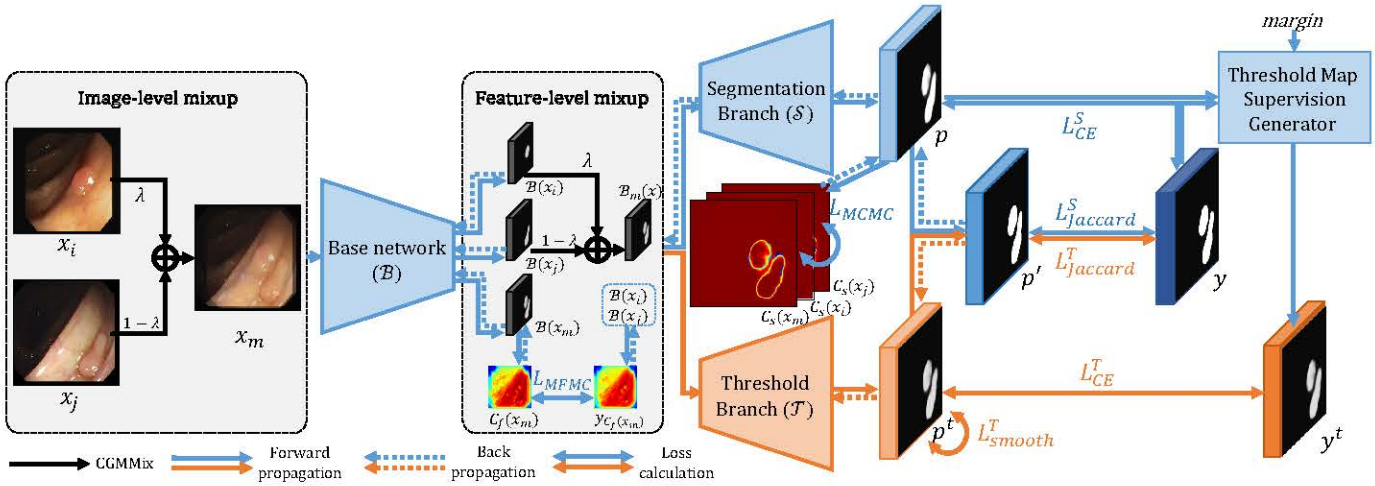

Fig. 2: Illustration of the proposed ThresholdNet with CGMMix. ThresholdNet is comprised of a base network $\mathcal{B}$, a segmentation branch $\mathcal{S}$ and a threshold branch $\mathcal{T}$, which is optimized with the original data and augmented data attained by CGMMix.

to perform non-linear upsampling in the decoder. To further reduce the semantic gap between the encoder and decoder, Zhou *et al.* [7] proposed UNet++ that introduced a series of nested and dense skip connections between encoder and decoder to enable deep supervision. Fang *et al.* [9] followed the UNet++ to introduce up-concatenations and proposed a selective kernel module for multi-scale feature aggregation. Then the proposed network was trained with the joint guidance of polyp area and boundary label, making the learned model more sensitive to the prediction around polyp boundary.

Despite the significant progress of these models for polyp segmentation, they may suffer unsatisfactory performance for clinical application due to the limited training data. In addition, the aforementioned segmentation algorithms are still prone to make errors at uncertain regions, such as polyp boundaries and fuzzy regions, because they simply threshold the predicted likelihood map with 0.5 to obtain final polyp regions.

B. Mixup

Data augmentation increases the variety of training samples and can greatly improve the generalization capability of deep learning models [23]–[31], [33]. The traditional data augmentation methods simply enrich datasets by distorting image space, such as random translation, rotation, cropping, flipping and adding noises [25], [33]. However, the aforementioned augmentation methods can only produce new images that closely resemble original images, and the improvement may not be satisfactory. Moreover, the resultant new images share the same class of the original images, thus the vicinal distribution between different classes has not been considered.

The recently proposed mixup augments data by producing an element-wise linear combination of training images and labels [23]. The newly generated images are noticeably different than original images, and the linear combination reduces the undesirable oscillations while predicting and guarantees a robust model behavior. Increasing interests have been attracted to modify and apply mixup in various tasks due to its excellent properties [24]–[31]. Li *et al.* [29] proposed an asymmetric mixup for brain tumor core segmentation. The asymmetric mixup can keep the decision boundary close to the background class and increase the area of foreground logits, thus alleviating the class imbalance problem. Wang *et al.* [26] employed an adversarial training strategy and applied mixup between labeled and unlabeled data to reduce the empirical distribution mismatch in semi-supervised learning. Xu *et al.* [24] applied mixup to domain adaptation and proposed a domain mixup to fully utilize the inter-domain information, which guaranteed domain invariance in a continuous latent space.

Though mixup has been widely applied to natural image classification problems, no work investigates its application in the polyp segmentation. Moreover, the mixup method ignored the data augmentation at feature level and the class imbalance problem in the polyp segmentation task.

III. Methodology

The overall framework of the proposed method is illustrated in Fig. 2. Image-level mixup is first implemented to obtain x_m by pixel-wise convex combinations of two randomly selected images x_i and x_j in the training dataset. Both original images and mixup images are then fed into ThresholdNet. A base network ($\mathcal{B}$) maps input images to feature space. At feature level, the extracted features $\mathcal{B}(x_i)$ and $\mathcal{B}(x_j)$ are also linearly mixed to produce mixup feature $\mathcal{B}_m(x)$. After that, the framework is split into two branches, a segmentation branch ($\mathcal{S}$) and a threshold branch ($\mathcal{T}$). Through the segmentation branch, $\mathcal{B}(x_i)$, $\mathcal{B}(x_m)$ and $\mathcal{B}_m(x)$ are separately passed to obtain the likelihood map p. The threshold branch is parallel to the original segmentation branch and has an identical structure to that of the segmentation branch. $\mathcal{B}(x_i)$, $\mathcal{B}(x_m)$ and $\mathcal{B}_m(x)$ are individually processed through it to obtain the prediction of threshold map p^t. Then the predicted likelihood map is subtracted by the corresponding threshold map to obtain the final segmentation prediction p', which should be similar to the segmentation ground truth y. To explicitly learn the threshold map, the TMSG module is proposed to obtain the ground truth of threshold map y^t, leading to better supervision of the threshold branch. With the proposed confidence-guided mixup

label, the mixup image and feature can also obtain the corresponding y^t through the TMSG module for threshold learning, and adaptively adjust the decision boundary to alleviate the class imbalance problem. Through alternative optimization of the proposed segmentation and threshold losses, these two branches are mutually constrained to extract individual information and predict separate maps. Details are illuminated in the following parts.

A. Confidence-Guided Manifold Mixup (CGMMix)

High-quality annotations are rare in the polyp segmentation task, since the acquisition of expert annotations is time-consuming and requires professional domain knowledge. The limited pixel-wise annotated data unavoidably leads to overfitting problem during training. The recently proposed mixup augments data by producing an element-wise linear combination of training images and labels [23], which is an effective data augmentation method to improve the generalization of whole-image classification model. Given two randomly sampled training data (x_i, y_i) and (x_j, y_j), the augmented mixup image and its corresponding label are calculated by $\widetilde{x_i} = \lambda x_i + (1-\lambda)x_j$ and $\widetilde{y_i} = \lambda y_i + (1-\lambda)y_j$, where λ is randomly sampled from a beta distribution $Beta(\alpha, \beta)$, and α, β are empirically set as 0.4 [23].

However, mixup only implements interpolations at the input image space, which could not guarantee the linear behavior at the level of hidden representations. Moreover, directly applying the mixup, which is developed for the whole-image classification task, to the polyp image segmentation model is problematic. The reasons can be summarized into two aspects. On the one hand, the original mixup encourages the model to center the decision boundary between classes [29], ignoring the class imbalance problem in polyp segmentation task. On the other hand, considering that characteristics of polyps vary significantly for different degrees of CRC, although feature representations of some conspicuous polyp regions are weakened by mixing with normal tissues, the corresponding regions (just like polyps at their early stages) should still be diagnosed as the polyp category in clinical practice.

To remedy these drawbacks of mixup, we propose CGMMix data augmentation method at both image and feature levels, and the CGMMix can adaptively adjust the decision boundary by introducing confidence-guided threshold to the soft mixup labels. In this section, we first describe the manifold mixup, then introduce the mixup labels with confidence guidance and finally present two consistency regularizations.

1) Image and feature-level mixup: Two randomly sampled images x_i and x_j from training dataset are first linearly mixed to produce mixup images x_m. Inputs of x_i and x_j are then embedded to $\mathcal{B}(x_i)$ and $\mathcal{B}(x_j)$ in the feature space by a base network. In order to yield a more smooth and linear feature distribution, two feature embeddings are also linearly interpolated to produce the mixup feature. The image and feature-level mixups can be formulated as

$$\begin{aligned} x_m &= \lambda x_i + (1-\lambda)x_j; \\ \mathcal{B}_m(x) &= \lambda \mathcal{B}(x_i) + (1-\lambda)\mathcal{B}(x_j). \end{aligned} \tag{1}$$

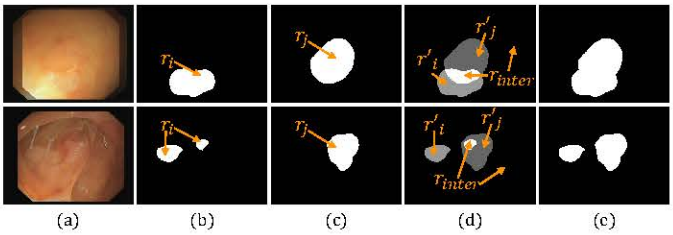

Fig. 3: Illustration of confidence-guided mixup label: (a) the mixed sample derived from two original images (x_i and x_j); (b, c) confidence- guided polyp regions (r_i and r_j); (d) the mixup label; (e) the confidence-guided mixup label.

With the linear interpolations at image space and hidden representation, the ThreshodNet is able to behave linearly between training images and features, thereby improving the robustness and facilitating the generalization ability.

2) Confidence-guided mixup label: A threshold coefficient t was introduced to threshold the original soft mixup label in [29], and the modified mixup label is denoted as $y_m = \delta(\lambda y_i + (1-\lambda)y_j > t)$, where $\delta(\cdot) = 1$ if condition is satisfied, and otherwise $\delta(\cdot) = 0$. This asymmetric mixup label can increase the area of foreground logit distribution, thereby alleviating class imbalance problem. Considering the various appearances and characteristics of polyps for different degrees of CRC, applying a constant threshold coefficient of t on all polyp samples is insufficient. Therefore, it may be beneficial to adaptively assign the threshold for different polyp regions, and we propose the confidence-guided mixup label. Specifically, we first calculate confidence maps $C_s(x_i)$ and $C_s(x_j)$ in the segmentation branch by $C_s(x) = p[:,:,1] \times y + p[:,:,0] \times (1-y)$, where $p[:,:,1]$ indicates the probability of the polyp category. These calculated confidence maps in polyp regions can reveal the severity degree of the polyp, and they are integrated to obtain the correct mixup label, which can be formulated as

$$y_m = \delta(\lambda y_i \cdot C_s(x_i) + (1-\lambda)y_j \cdot C_s(x_j) > t), \tag{2}$$

where $y_m \in \{0,1\}^{W \times H}$, and y_m has the same spatial resolution of $W \times H$ as input images. Assuming $r_i = \delta(\lambda y_i \cdot C_s(x_i) > t)$ and $r_j = \delta((1-\lambda)y_j \cdot C_s(x_j) > t)$ are confidence-guided polyp regions of x_i and x_j, then $r_i' = r_i - r_i \times r_j$ and $r_j' = r_j - r_i \times r_j$ are the non-overlapping polyp regions, and $r_{inter} = r_i \times r_j + (1-r_i) \times (1-r_j)$ denotes the resting regions, as in Fig. 3. Then the confidence-guided mixup label can be reformulated as $y_m = y_i \cdot r_i' + y_j \cdot r_j' + [\lambda y_i + (1-\lambda)y_j] \cdot r_{inter}$.

Intuitively, if those non-obvious polyps with the ambiguous boundary and relatively low contrast (especially at their early stages) are mixed with normal tissues, i.e., the characteristics of polyp are overwhelmed, then its corresponding confidence score should be small and the condition would not be satisfied, thereby belonging to the normal class. Only if the mixup polyp region, with a soft label above the adjusted threshold, still belongs to the polyp class. Considering the confidence maps may not be able to indicate the degree of the polyp accurately at the early stages of training, the augmented images by CGMMix are included after 50 epochs.

3) Consistency regularization: To enhance the supervision for mixup data, we propose a MFMC loss and a MCMC loss to constrain the hidden feature representation and the likelihood map of the mixed images.

For the MFMC loss, it regularizes the feature map of the mixed image $\mathcal{B}(x_m)$ to be similar to the combination of the feature maps ($\mathcal{B}(x_i)$ and $\mathcal{B}(x_j)$) derived from two original images. Since the element-wise consistency of feature maps may lead to unstable training, we propose to regularize the consistency on a defined cosine similarity map to avoid the negative effects caused by the feature noises. Take $\mathcal{B}(x_i)$ for example, the feature cosine similarity map is calculated by

$$C_f(x_i) = \cos(f_p, \mathcal{B}(x_i)) = \frac{f_p^\top \mathcal{B}(x_i)}{\|f_p\|_2 \cdot \|\mathcal{B}(x_i)\|_2}, \tag{3}$$

where $f_p = \frac{\sum_{i=1}^{wh} \widetilde{y_i} \cdot \mathcal{B}(x_i)}{\sum_{i=1}^{wh} \widetilde{y_i}}$ is the average of foreground features, and $\widetilde{y_i} \in \{0,1\}^{w \times h}$ is down-sampled from y_i with the size $w \times h$. To induce the decision boundary far away from the under-represented polyp class at hidden features, we follow the spirit of confidence-guided mixup label in Eq. (2) to derive the ground truth in MFMC loss calculation. We first calculate confidence-guided polyp regions of x_i and x_j, which are denoted by $r_i = \delta(\lambda \widetilde{y_i} \cdot \widetilde{C_s(x_i)} > t)$ and $r_j = \delta((1-\lambda)\widetilde{y_j} \cdot \widetilde{C_s(x_j)} > t)$. Note that $\widetilde{C_s(x_i)}, \widetilde{C_s(x_i)}$ are bilinearly down-sampled from $C_s(x_i), C_s(x_i)$ with size of $w \times h$ and $r_i, r_j \in \{0,1\}^{w \times h}$. Then the intersecting polyp and normal regions can be represented by $r_{inter} = r_i \times r_j + (1-r_i) \times (1-r_j)$, and the non-overlapping polyp regions are $r_i' = r_i - r_i \times r_j$ and $r_j' = r_j - r_i \times r_j$. Hence, the ground truth of $C_f(x_m)$ is formulated by

$$\begin{aligned} y_{C_f(x_m)} = {} & C_f(x_i) \cdot r_i' + C_f(x_j) \cdot r_j' \\ & + [\lambda C_f(x_i) + (1-\lambda) C_f(x_j)] \cdot r_{inter}. \end{aligned} \tag{4}$$

Notably, $y_{C_f(x_m)}$ mainly modifies the mixed feature at non-overlapping polyp regions, which is different from the mixup method that directly conducts linear combination. This modification prevents the feature of polyp regions being perturbed and overwhelmed by the normal features in the mixed data, thereby, explicitly adjusting the decision boundary at hidden features. Then the MFMC loss can be computed by

$$\mathcal{L}_{MFMC} = \left\| y_{C_f(x_m)} - C_f(x_m) \right\|_2. \tag{5}$$

For the MCMC loss, it minimizes the dissimilarity between the confidence map calculated by mixup sample $C_s(x_m)$ and the mixed confidence maps ($C_s(x_i)$ and $C_s(x_j)$) of two original samples. We use the mean squared error to measure the discrepancy:

$$\mathcal{L}_{MCMC} = \|\lambda C_s(x_i) + (1-\lambda) C_s(x_j) - C_s(x_m)\|_2. \tag{6}$$

Through optimizing with MFMC and MCMC regularization losses, the ThresholdNet thus can be deeply supervised with the mixup samples at both feature and segmentation levels.

B. ThresholdNet

In this paper, we propose a ThresholdNet for polyp segmentation to joint segmentation and threshold learning in a robust way, as in Fig. 2. The following subsections present the architecture and loss functions of the ThresholdNet in detail.

1) Network architecture: Our model is composed of a base network ($\mathcal{B}$), a segmentation ($\mathcal{S}$) and a threshold ($\mathcal{T}$) branches. The base network is constructed for feature extraction, mapping input image x to feature maps $\mathcal{B}(x)$. We utilize the pre-trained parameters of ResNet-101 on ImageNet dataset to initialize the base network, in order to alleviate the overfitting problem caused by the limited training data. The subsequent segmentation branch is a decoder to obtain the likelihood map $p = \mathcal{S}(\mathcal{B}(x)), p \in [0,1]^{W \times H \times 2}$. In particular, the base network and the segmentation branch constitute the extraordinary DeepLabv3+ [34] backbone.

Even though existing deep learning-based methods [6]–[20] can segment polyp regions with high accuracy, these methods ignore the fact that different thresholds lead to varying results. Therefore, the further post-processing of the likelihood map is necessary for attaining better performance [32]. To adaptively learn a threshold map for each likelihood map, a threshold branch, with the identical structure of the segmentation branch, is introduced to be parallel to the segmentation branch. The feature maps extracted from the base network thus can be mapped to the threshold map $p^t = \mathcal{T}(\mathcal{B}(x)), p^t \in [0,1]^{W \times H \times 2}$ through the threshold branch. Then the predicted likelihood map is subtracted by the threshold map to obtain the final segmentation result, which can be represented by $p' = p - p^t = \mathcal{S}(\mathcal{B}(x)) - \mathcal{T}(\mathcal{B}(x))$. Hence, the generated threshold map provides the pixel-level threshold value to calibrate the final segmentation result, leading to a more accurate prediction at regions with less confidence values.

The base network with segmentation branch ($\mathcal{B} \cup \mathcal{S}$) and the threshold branch ($\mathcal{T}$) are constrained and optimized alternatively by minimizing their individual segmentation and threshold losses.

2) Segmentation loss: The segmentation loss $\mathcal{L}^{\mathcal{S}}$ is comprised of two commonly used loss functions, i.e., the binary cross-entropy loss function and the Jaccard loss function:

$$\begin{aligned} \mathcal{L}^{\mathcal{S}} &= \mathcal{L}^{\mathcal{S}}_{CE} + \mathcal{L}^{\mathcal{S}}_{Jaccard} \\ &= -\sum_i y_i \log p_i + \left(1 - \frac{\sum_i y_i M(p')_i}{\sum_i y_i + \sum_i M(p')_i - \sum_i y_i M(p')_i}\right), \end{aligned} \tag{7}$$

where p_i indicates the probability of i^{th} pixel being categorized as polyp, and $y \in \{0,1\}^{W \times H}$ denotes the corresponding segmentation ground-truth. $M(p')$ is a masking function to transfer the continuous segmentation prediction to a soft mask, and we utilize the sigmoid function as an approximation to make it derivable, as in the following formulation

$$M(p') = \frac{1}{1+e^{-\omega p'}} = \frac{1}{1+e^{-\omega(p-p^t)}}, \tag{8}$$

where ω is a scale parameter ensuring $M(p')_i$ approximately equals to 1 when p_i is larger than p_i^t, or to 0 otherwise. We set ω as 50 and keep it consistent for all experiments.

3) Threshold loss: In order to optimize the threshold branch, we comprehensively design a threshold loss, which is comprised of the cross-entropy loss function, the Jaccard loss function and an edge-aware smooth regularization. Since there is no ground truth of the threshold map provided in the training

dataset, a TMSG module is devised to produce the supervision for the threshold map learning.

Threshold Map Supervision Generator (TMSG). The TMSG module integrates the likelihood map and the segmentation label to attain the ground truth of threshold map y^t. Assuming the one-hot encoding form of segmentation label is $O(y) \in \{0,1\}^{W\times H\times 2}$, TMSG module can be formulated as

$$y^t = \begin{cases} p-m, & O(y)=1; \\ p+m, & \text{otherwise}, \end{cases} \tag{9}$$

where $y^t \in [0,1]^{W\times H\times 2}$ and $m \in (0, 0.5)$ is a margin coefficient to ensure the discrepancy of the threshold map and the likelihood map. The intuition behind TMSG is to ensure that the soft mask generated by Eq. 8 is the same as segmentation label $O(y)$ if substituting y^t into p^t. Hence, with the supervision of the ground truth y^t, the learned threshold map can provide precise pixel-wise threshold values for discerning polyp versus normal class. The threshold loss $\mathcal{L}^{\mathcal{T}}$ can be formulated as

$$\begin{aligned} \mathcal{L}^{\mathcal{T}} &= \mathcal{L}^{\mathcal{T}}_{CE} + \mathcal{L}^{\mathcal{T}}_{Jaccard} + \mathcal{L}^{\mathcal{T}}_{Smooth} \\ &= -\sum_i y_i^t \log p_i^t + (1 - \frac{\sum\limits_i y_i M(p')_i}{\sum\limits_i y_i + \sum\limits_i M(p')_i - \sum\limits_i y_i M(p')_i}) \\ &\quad + (\partial_h p_i^t \cdot e^{-\|\partial_h x_i\|} + \partial_v p_i^t \cdot e^{-\|\partial_v x_i\|}), \end{aligned} \tag{10}$$

where ∂_h and ∂_v denote partial derivatives along with the horizontal and vertical directions, respectively. The first item is the cross-entropy loss, while the second item is the Jaccard loss. In addition, we add the third item to regularize the smoothness of the learned threshold map since the discontinuities often occur at edges. With the optimization of the threshold loss, the threshold branch can adaptively threshold the likelihood map, thus polishing the final segmentation prediction.

C. Optimization

The solution to the ThresholdNet can be approximately attained via the alternative search strategy, which commutatively optimizes the involved parameters $\mathcal{B}$, $\mathcal{S}$ and $\mathcal{T}$ as described in Algorithm 1. More specifically, denote $\mathcal{B}^k$, $\mathcal{S}^k$ and $\mathcal{T}^k$ as the optimization variables involved in the ThresholdNet at iteration k ($k = 0, 1, 2, ...,$), respectively, then our optimization strategy in each iteration contains the following steps:

Update $\mathcal{B}^k$ and $\mathcal{S}^k$ with fixed $\mathcal{T}^k$: This step aims to update the base network and the segmentation branch. For original input images, the likelihood map p is obtained with the parameters $\mathcal{B}^k$ and $\mathcal{S}^k$, and the soft mask $M(p')$ is derived from p and $\mathcal{T}^k$. Both p and $M(p')$ are constrained to be similar with the ground truth y, and the corresponding objective function is computed by Eq. (7). In this case, the optimization procedure can be represented by the following form:

$$\mathcal{B}^{k+1} \cap \mathcal{S}^{k+1} = \min_{\mathcal{B}\cap\mathcal{S}} \mathcal{L}^{\mathcal{S}}(\mathcal{B}^k, \mathcal{S}^k, \mathcal{T}^k, x, y). \tag{11}$$

For mixup images, apart from the segmentation loss, two regularization losses in Eq. (5) and Eq. (6) should also be involved for deep supervision. Hence, the optimization for mixup images can be formulated as

$$\begin{aligned} \mathcal{B}^{k+1} \cap \mathcal{S}^{k+1} = \min_{\mathcal{B}\cap\mathcal{S}} [&\mathcal{L}^{\mathcal{S}}(\mathcal{B}^k, \mathcal{S}^k, \mathcal{T}^k, x_m, y_m) \\ &+\zeta \cdot \mathcal{L}_{MFMC}(\mathcal{B}^k, \mathcal{S}^k, x_m, y_m) \\ &+\xi \cdot \mathcal{L}_{MCMC}(\mathcal{B}^k, \mathcal{S}^k, x_m, y_m)], \end{aligned} \tag{12}$$

where ζ and ξ are trade-off parameters controlling the contribution of MFMC and MCMC losses.

For mixed features, it passes through $\mathcal{S}^k$ and $\mathcal{T}^k$ to attain p and $M(p')$, and the optimization process is defined as:

$$\mathcal{B}^{k+1} \cap \mathcal{S}^{k+1} = \min_{\mathcal{B}\cap\mathcal{S}} \eta \cdot \mathcal{L}^{\mathcal{S}}(\mathcal{S}^k, \mathcal{T}^k, \mathcal{B}_m(x), y_m), \tag{13}$$

where the weighting factor η is a hyper-parameter that adjusts the contribution of mixed features.

Update $\mathcal{T}^k$ with fixed $\mathcal{B}^{k+1}$ and $\mathcal{S}^{k+1}$: The goal of this step is to update the threshold branch with $\mathcal{B}^{k+1}$ and $\mathcal{S}^{k+1}$ updated in last step. The optimization method for original input images and mixup images are the same, which can be formulated as

$$\mathcal{T}^{k+1} = \min_{\mathcal{T}} \mathcal{L}^{\mathcal{T}}(\mathcal{B}^{k+1}, \mathcal{S}^{k+1}, \mathcal{T}^k, x, y). \tag{14}$$

For mixed features, $\mathcal{T}^k$ can be updated by the following formulation:

$$\mathcal{T}^{k+1} = \min_{\mathcal{T}} \eta \cdot \mathcal{L}^{\mathcal{T}}(\mathcal{S}^{k+1}, \mathcal{T}^k, \mathcal{B}_m(x), y_m). \tag{15}$$

Ultimately, the whole alternative search process of the proposed method can be summarized as in Algorithm 1.

Algorithm 1 : Optimization

Input: Dataset (x, y)
Output: Likelihood map p, threshold map p^t, segmentation results p'

```
Initialize B with pre-trained parameters and randomly initialize S and T
for k = 1:3:iter do
    Conduct image-level mixup on randomly selected images (x_i, x_j) to obtain mixup data x_m by Eq. (1)
    Calculate the deep features B(x_i), B(x_j) and B(x_m)
    Conduct feature-level mixup on B(x_i) and B(x_j) to obtain mixup feature B_m(x) by Eq. (1)
    for input = [x_i, x_m, B_m(x)] do
        if input = x_i then
            Calculate the confidence-guided mixup label y_m by Eq. (2)
            Update parameters B and S by Eq. (11)
            Update parameters T by Eq. (14)
        else if input = x_m then
            Update parameters B and S by Eq. (12)
            Update parameters T by Eq. (14)
        else if input = B_m(x) then
            Update parameters S by Eq. (13)
            Update parameters T by Eq. (15)
        end if
    end for
end for
```

TABLE I: Quantitative comparison of segmentation results for EndoScene and WCE polyp dataset. The small P-values ($P < 0.001$) calculated between our method vs. state-of-the-art methods in terms of *Dice* indicate the improvements are significant.

Datasets	Methods	*Dice (%)*	*Jac (%)*	*Sen (%)*	*Spe (%)*	*Acc (%)*	*F2 (%)*	*p*-value
EndoScene	DeepLabv3+ [34]	82.523	74.927	82.299	99.306	96.444	82.023	1.21e-12
	Vázquez *et al.* [6]	80.099	72.320	78.986	99.439	96.296	79.061	3.05e-11
	Zhou *et al.* [7]	79.842	71.756	82.492	98.622	95.885	80.252	2.71e-9
	Fang *et al.* [9]	81.987	73.654	82.630	99.306	96.438	81.859	1.24e-9
	Qadir *et al.* [12]	84.145	77.369	84.575	99.328	96.877	83.433	5.71e-6
	Wickstrøm *et al.* [17]	81.867	74.542	82.130	99.286	96.643	81.731	1.36e-7
	Ours	**87.307**	**80.570**	**87.973**	**99.466**	**97.213**	**87.278**	–
WCE	DeepLabv3+ [34]	79.110±2.123	69.280±2.113	81.832±2.136	98.613±0.290	98.250±0.081	80.217±2.120	4.86e-8
	Vázquez *et al.* [6]	73.422±1.304	64.113±0.808	73.562±2.199	99.010±0.321	97.969±0.355	73.048±1.644	5.30e-11
	Zhou *et al.* [7]	80.811±1.791	72.326±1.426	81.273±1.916	99.165±0.170	98.265±0.337	80.808±1.850	5.40e-5
	Fang *et al.* [9]	75.106±0.662	65.130±0.881	74.960±0.877	**99.306±0.132**	97.118±0.574	74.556±0.626	1.10e-11
	Qadir *et al.* [12]	82.927±1.342	74.096±1.513	85.847±1.689	98.580±0.304	98.188±0.165	84.031±1.461	3.23e-4
	Wickstrøm *et al.* [17]	78.103±2.778	69.502±2.289	78.576±2.103	99.162±0.094	98.248±0.243	78.033±2.389	3.77e-6
	Ours	**87.879±1.038**	**79.807±1.214**	**88.878±0.868**	99.087±0.300	**98.794±0.112**	**88.267±0.951**	–

IV. Experiments

We introduced the utilized datasets in subsection IV-A and our experimental setup in IV-B. Then we compared our method with the state-of-the-art methods on EndoScene and WCE polyp datasets in IV-C. To clarify the validity of CGMMix and ThresholdNet, we further conducted ablation experiments on EndoScene dataset, as in subsection IV-D and IV-E.

A. Datasets

Two polyp image datasets were utilized in this study, and two samples from different datasets are shown in Fig. 1.

EndoScene dataset: This dataset includes 912 images and the corresponding pixel-wise labels, obtained from 44 videos of 36 patients. We followed the standard setup for polyp segmentation with 547 training, 183 validation, and 182 testing colonoscopy images [6] based on the constraint that one patient can not be in different sets.

WCE polyp dataset: Our WCE polyp dataset comprises 541 images, collected from the Prince of Wales Hospital with Medtronics Pillcam wireless capsule endoscope (WCE). The ground truths of the polyp regions were annotated by two experts. This dataset was randomly split for fourfold cross-validation.

B. Experiment Setup

Implementation details: Our method was implemented with the PyTorch library. SGD was chosen for optimization with a batch size of 16. We adopted polynomial learning rate scheduling with the initial learning rate of 0.001, the power of 0.9 and the maximum epoch number of 500. The threshold coefficient t in CGMMix was set to 0.3, and the margin coefficient m in TMSG module was also set to 0.3. To further enlarge the training dataset, we employed the online data augmentation, including adding perturbation in HSV color space, random horizontal flip, rotation from 0 to 180 degrees, scale and crop. The augmented patches were then resized to 256×256 for training.

Evaluation metrics: The performance of polyp segmentation was evaluated by six commonly-used metrics, including Dice similarity coefficient ($Dice$), Jaccard index (Jac), Sensitivity (Sen), Specificity (Spe), Accuracy (Acc) and F2-score ($F2$). For these evaluation metrics, a higher value indicates a better segmentation result.

C. Segmentation Performance

Since the base network and the segmentation branch constitute DeepLabv3+ [34], we first compared the proposed method with the backbone DeepLabv3+. To verify the effectiveness of our approach, we further compared it with state-of-the-art polyp segmentation models [6], [7], [9], [12], [17]. Especially, the DeepLabv3+, UNet++ [7] and method in [12] were initialized with the corresponding pre-trained parameters, and then the whole segmentation networks were fine-tuned on the two polyp datasets. For a fair comparison, we implemented the network architectures of these methods and used the same online data augmentation method for data preprocessing. The quantitative performance of different methods on the two polyp datasets are listed in Table I, and the qualitative comparison results are illustrated in Fig. 4.

1) Results on EndoScene dataset: The weighting factors ζ, ξ in Eq. (12) and η in Eq. (13, 15) are empirically set as 0.5, 0.2 and 0.5 for the training of EndoScene dataset. As in Table I, our method shows promising performance with a $Dice$ of 87.307% and a Sen of 87.973% and exhibits a significant improvement ($P < 0.001$) of 4.784%, 5.674% in $Dice$ and Sen compared with DeepLabv3+. This is because that the threshold branch in the ThresholdNet provides pixel-level threshold value to amend the predicted results in error-prone regions. In addition, the CGMMix enriches the limited training samples and enhances the linear behavior of the ThresholdNet. Notably, the considerable improvement in Sen reveals that CGMMix can greatly alleviate the class imbalance problem by adaptively luring the decision boundary away from the under-represented polyp class. Hence, it can be concluded that the proposed ThresholdNet together with CGMMix contributes to the favorable performance promotion. Moreover, it is worth noting that the proposed approach achieves better performance than state-of-the-art methods [6], [7], [9], [12], [17]. Specifically, the proposed method possesses

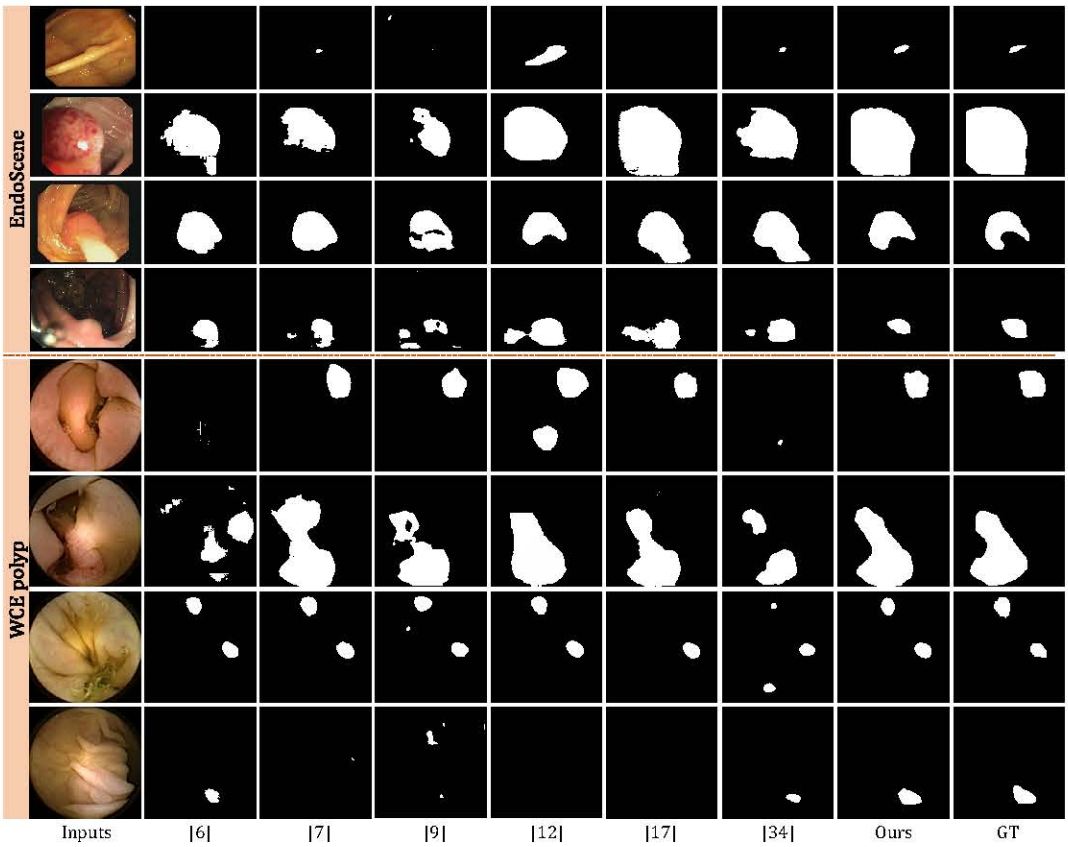

Fig. 4: Qualitative comparison of segmentation results by different methods for EndoScene and WCE polyp dataset. From left to right are the test images (1^{st} col), results of state-of-the-art methods [6], [7], [9], [12], [17] ($2^{nd} - 6^{th}$ col), results of our backbone DeepLabv3+ [34] (7^{th} col), results of our method (8^{th} col), and ground truth (9^{th} col).

superior capability for polyp segmentation with increments of 8.250%, 8.814%, 6.916%, 3.201%, 6.028% in Jac and 8.987%, 5.481%, 5.343%, 3.398%, 5.843% in Sen compared with methods [6], [7], [9], [12], [17], respectively.

The $1^{st} - 4^{th}$ $rows$ in Fig. 4 visualizes four polyp images and the corresponding segmentation results derived from methods in [6], [7], [9], [12], [17], [34] and our method. It can be observed that compared with other baselines, the predictions of our method have much fewer false negatives. For example, other methods tend to ignore fine structures (1^{st} row) and can not tackle the huge variances exhibited in a polyp ($2^{nd} - 3^{rd}$ $rows$), while our approach can automatically pinpoint these error-prone regions. This may be ascribed to the effect of CGMMix, which can adaptively threshold the soft mixup label and induce the adjustment of the decision boundary. Indeed, this adaptive adjustment can prevent unseen under-presented polyp samples from shifting across the decision boundary. In addition, other methods tend to category specular reflections as the polyp regions as in 4^{th} row due to the visual similarity, while our approach can automatically remove these over-segmented regions. We conjecture the reasons are: the introduced threshold branch can rectify the segmentation results, and the enriched information by CGMMix enhances the feature representation capability of the ThresholdNet.

2) Results on WCE polyp dataset: The hyper-parameters ζ, ξ and η are all set as 0.1 for WCE polyp segmentation. We presented the results on WCE polyp dataset using the mean and the standard deviation of evaluation metrics in Table I. Due to the relatively lower resolution, polyp regions in WCE images exhibit ambiguous boundary (5^{th} row) and high degree of visual similarity among polyp and normal tissues ($7^{th} - 8^{th}$ row) as in Fig. 4. Our method can still perform well with a $Dice$ of 87.879% and a Sen of 88.878% in such dilemma. Moreover, our method demonstrates superior segmentation performance in comparison to methods [34], [6], [7], [9], [12], [17] with statistically significant increments ($P < 0.001$) of 8.769%, 14.457%, 7.068%, 12.773%, 4.952%, 9.776% in $Dice$ score. The 5^{th} to 8^{th} $rows$ in Fig. 4 presents typical segmentation results of WCE polyp images, for a quantitative comparison of these different methods.

D. Ablation Analysis on CGMMix

1) Effectiveness of CGMMix: For in-depth analysis of CGMMix, we designed the following four experimental settings.

- DeepLabv3+ [34]: it serves as our backbone network, and this network architecture is separately trained with different data augmentation methods as follows, for a fair comparison of different methods.
- w/ Mixup [23]: it conducts linear combination on input images and labels.
- w/ Asymmetric mixup [29]: the linear combination is only implemented on input images, and the corresponding labels are derived with a threshold of $t = 0.3$.

TABLE II: Comparison results of polyp segmentation under different data augmentation techniques.

Methods	*Dice*	*Jac*	*Sen*	*Spe*	*Acc*	*F2*
DeepLabv3+ [34]	82.52	74.93	82.30	99.31	96.44	82.02
w/ Mixup [23] †	83.55	76.51	83.60	99.39	96.64	83.08
w/ Asymmetric mixup [29] ‡	84.69	77.44	85.42	99.33	96.73	84.63
w/ CGMMix	85.43	78.41	85.79	99.45	96.89	85.24
Ours	**87.31**	**80.57**	**87.97**	**99.47**	**97.21**	**87.28**

† P-value (CGMMix vs. Mixup): 0.005;
‡ P-value (CGMMix vs. Asymmetric mixup): 0.028.

- w/ CGMMix: our proposed data augmentation method with a threshold of $t = 0.3$.

The comparison results in Table II show that the proposed CGMMix method performs favorably against the mixup and asymmetric mixup with significant improvements ($P < 0.05$), demonstrating the good capability of CGMMix to enrich limited training dataset. Among these metrics, Sen represents the percentage of polyp pixels that are correctly classified, which is more clinically relevant and can accelerate the screening examination [32]. Since the area of polyp is smaller than that of normal tissues, existing methods [6]–[22] usually exhibit a poor sensitivity in polyp segmentation. The CGMMix was confirmed to have an inherent capability of dealing with this class imbalance problem and achieved a prominent Sen of 85.79%, which shows increments of 3.49%, 2.19% in comparison to the DeepLabv3+, mixup, respectively. The proposed CGMMix was verified to be superior than asymmetric mixup [29] with an increment of 0.97% in Jac. The improvement is due to that the proposed confidence-guided mixup label is more precise than the asymmetric label that simply thresholds original mixup label with a certain constant. In addition, the feature-level mixup and two consistent regularizations enable the comprehensive and sufficient utilization of mixup data.

2) Visualization of data distribution: To demonstrate the effectiveness of CGMMix, we then visualized the logit distribution of DeepLabv3+ [34], mixup [23], Asymmetric mixup [29] and our CGMMix for comparison, as in Fig. 5. The 2-dimensional logit distribution can reflect the feature distribution in high dimensional space since they are linearly mapped from features. From the logit visualization of DeepLabv3+ and mixup, we can observe that the logit activations of normal pixels from the training and testing sets display an obvious gap in terms of the logit center, which reveals the overfitting problem exists in polyp segmentation. In addition, many logit activations of pixels in polyp regions shift significantly towards or even across the decision boundary in DeepLabv3+ and mixup methods. This shift can cause false negatives and result in the under-segmentation of polyps. Notably, as in the logit distribution of CGMMix, the logit centers of training and testing sets in the polyp class both stay away from the boundary. This observation indicates that CGMMix can asymmetrically and adaptively adjust the decision boundary in high dimensional feature space, which significantly reduces logit shift of unseen polyp pixels and leads to the large improvement in sensitivity. Moreover, it is clear that our logit activation centers from the training and testing sets tend to be similar, demonstrating the generalization ability of the proposed CGMMix.

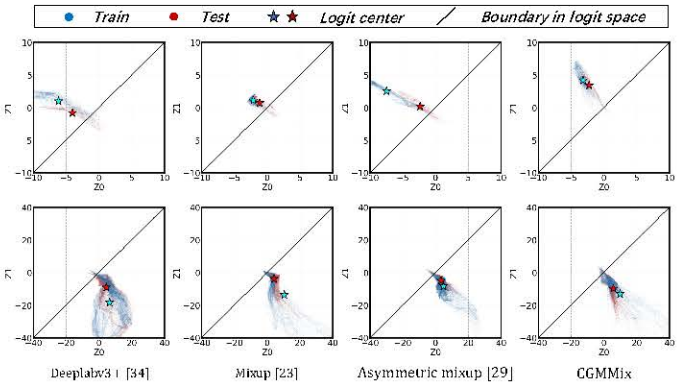

Fig. 5: Activations of the classification layer (z_0 for normal logit, z_1 for polyp logit) when processing polyp (top) and normal (bottom) pixels in EndoScene dataset.

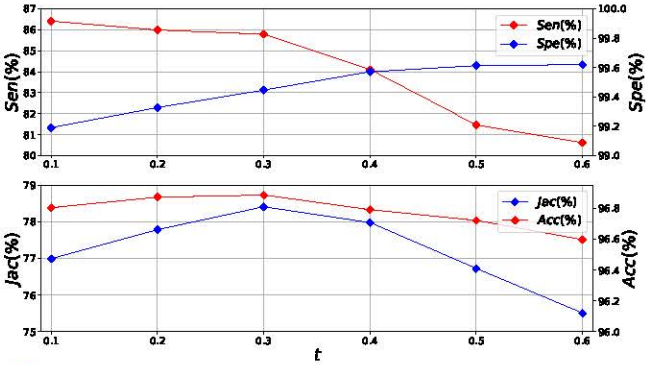

Fig. 6: CGMMix with threshold varying from 0.1 to 0.6.

3) Analysis on the threshold t: A lower threshold t of CGMMix in Eq. (2) often results in a larger number of pixels belonging to polyp class, further leading to a relatively balanced class distribution. We conducted experiments to evaluate the performance of CGMMix with different t, and the corresponding Acc, Jac, Sen and Spe curves are shown in Fig. 6. Obviously, aggravating the class imbalanced distribution ($t = 0.6$) further deteriorates the sensitivity of the model. In the meanwhile, choosing a lower $t = 0.1$ achieves a higher sensitivity, since a lower t can lure the decision boundary away from the polyp class and increase the corresponding logits area. Acc and Jac are comprehensive metrics that measure the overall performance of CGMMix without being affected by the class imbalance problem. From the below subfigure in Fig. 6, the performance of CGMMix increases first and then decreases with t varying from 0.1 to 0.6. Properly choosing the value of t can reach a trade-off between sensitivity and specificity and further promote the segmentation performance.

4) Ablation study for each component in CGMMix: To evaluate the contributions of different components devised in the CGMMix, we first quantified the contribution of image and feature-level mixups by seriatim ablating the multiple mixups, as shown in Table III. In particular, we studied different variants of CGMMix, including: 1) "Baseline" corresponds to the baseline method without using CGMMix; 2) "w/ image-level mixup" embraces in the image-level mixed samples for training; 3) "w/ feature-level mixup" incorporates mixed feature maps for the optimization of segmentation branch.

TABLE III: Comparison results of image and feature-level mixups in CGMMix.

Methods	*Dice*	*Jac*	*Sen*	*Spe*	*Acc*	*F2*
Baseline	82.52	74.93	82.30	99.31	96.44	82.02
w/ image-level mixup	84.78	77.53	84.42	99.42	96.88	84.23
w/ feature-level mixup	84.84	77.70	84.38	**99.45**	96.82	84.17
CGMMix	**85.43**	**78.41**	**85.79**	**99.45**	**96.89**	**85.24**

TABLE IV: Comparison results of MFMC loss and MCMC loss in the image-level mixup of CGMMix.

Methods	*Dice*	*Jac*	*Sen*	*Spe*	*Acc*	*F2*
w/o MFMC	85.14	77.86	**85.91**	99.34	96.80	85.05
w/o MCMC	85.09	78.08	84.97	99.43	99.43	84.54
CGMMix	**85.43**	**78.41**	85.79	**99.45**	**96.89**	**85.24**

The image-level mixup was demonstrated to be effective in improving the generalization of baseline model with increments of 2.60%, 2.12% in Jac and Sen, respectively. We also verified the capability of the proposed feature-level mixup (4^{th} row) with reference to the result of baseline (2^{nd} row). The comparison results show that the feature-level mixup method promotes the performance of baseline model, with a relative improvement of 2.32% from 82.52% to 84.84% in $Dice$ score for polyp segmentation.

In the proposed CGMMix, two consistency regularizations, MFMC and MCMC losses, were developed to optimize the image-level mixup. Therefore, we analyzed the individual components of objective functions in CGMMix. MFMC loss in Eq. (5) regularizes the feature map of the mixed image to be similar to the combination of the feature maps derived from two original images, and MCMC loss in Eq. (6) minimizes the dissimilarity between the confidence map calculated by mixed image and the mixed confidence maps of two original samples. The corresponding ablated results were listed in Table IV. It is evident that both MFMC and MCMC losses contribute to the performance boosts of CGMMix. In particular, discarding MFMC loss led to a worse performance with a relative reduction of 0.55% in Jac score, and the elimination of MCMC resulted in a degradation of 0.82% Jac score.

5) Visualization of confidence map: To qualitatively verify the proposed CGMMix data augmentation method, we visualized the learned confidence map $C_s(x_m)$, likelihood map p of CGMMix images and their corresponding labels, as in Fig. 7. Though the feature representation of polyp is weakened by normal tissues in the mixed images ((c) col), the corresponding region should be diagnosed as a polyp in clinical practice, and the generated confidence-guided mixup label ((g) col) is consistent with the diagnosis from clinicians. From the confidence maps ((d), (e) col), it is clear to see that DeepLabv3+ tends to make the wrong decision around the boundaries of polyp, i.e., error-prone regions. Therefore, MCMC loss emphasizes the constraints on these error-prone regions, thereby boosting the performance of CGMMix method.

E. Ablation Analysis on ThresholdNet

1) Effectiveness of Threshold: To quantify the performance of ThresholdNet, we designed the ablation experiment, and the corresponding results are listed in Table V. We first

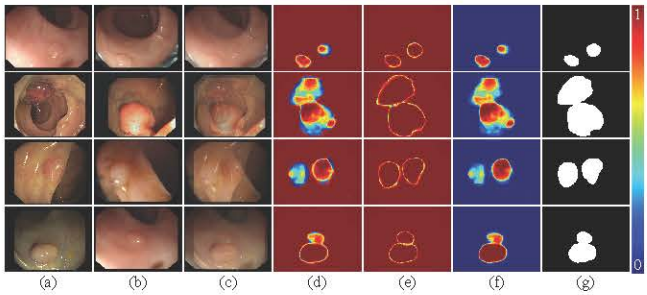

Fig. 7: Illustration of CGMMix: (a, b) original images to be mixed; (c) the CGMMix image x_m generated under the setting $\lambda = 0.5$; (d) the confidence map $C_s(x_m)$; (e) the ground truth of confidence map; (f) the likelihood map p; (g) the confidence-guided mixup label.

TABLE V: Ablation study on the testing set of EndoScene dataset in terms of margin (m) parameter in ThresholdNet.

Methods	*Dice*	*Jac*	*Sen*	*Spe*	*Acc*	*F2*
DeepLabv3+ [34]	82.52	74.93	82.30	99.31	96.44	82.02
w/ TMSG	84.41	77.53	84.74	99.39	96.74	84.06
w/ $\mathcal{L}^{\mathcal{S}}_{Jaccard}$ & $\mathcal{L}^{\mathcal{T}}_{Jaccard}$	86.08	79.04	86.43	99.46	96.96	85.85
ThresholdNet (m=0.1)	86.66	79.81	86.45	99.53	97.03	86.22
ThresholdNet (m=0.2)	86.59	79.80	87.06	99.50	97.10	86.38
ThresholdNet (m=0.3)	86.85	79.97	86.14	**99.56**	97.11	86.13
ThresholdNet (m=0.4)	86.64	79.79	86.09	99.54	97.05	86.03
Ours	**87.31**	**80.57**	**87.97**	99.47	**97.21**	**87.28**

investigated the effect of the threshold branch and the TMSG module (3^{rd} row), and all evaluation metrics increased in comparison to the baseline DeepLabv3+ [34] (2^{nd} row). This result proves the advantage of the threshold branch and the TMSG module, which can automatically learn the correct threshold map to help the segmentation branch discerning polyps versus normal ones. Then the reciprocal constraints, i.e., $\mathcal{L}^{\mathcal{S}}_{Jaccard}$ and $\mathcal{L}^{\mathcal{T}}_{Jaccard}$, were added to demonstrate the effectiveness of introducing interaction between two branches (4^{th} row), which shows increments of 3.56% in $Dice$ and 4.11% in Jac compared with DeepLabv3+. Through these bidirectional constraints, the threshold branch is still able to learn the adaptive threshold map dynamically by referring to the likelihood map in $\mathcal{L}^{\mathcal{T}}_{Jaccard}$ calculation, even though no direct supervision is provided. In conjunction with these two approaches and choosing a margin of $m = 0.3$, the polyp segmentation performance is boosted to overall $Dice$ of 86.85% (6^{th} row), far outstripping our baseline DeepLabv3+ with an increment of 4.33% in $Dice$. This promising improvement demonstrates the good capability of ThresholdNet to rectify final segmentation results.

2) Visualization of threshold map with varying margin m*:* We conducted comparison experiments to explore the sensitiveness of the hyper-parameter m in the TMSG module, and the corresponding results were listed in $5^{th}-8^{th}$ $rows$ of Table V. It is apparent that the results are not very sensitive to the value of margin m, demonstrating the stability and robustness of the ThresholdNet. To further evaluate the proposed ThresholdNet, we visualized the learned threshold maps with m varying from 0.1 to 0.4, as in Fig. 8. Obviously, a larger value of m results in a more discrepant threshold map compared with the corresponding likelihood map.

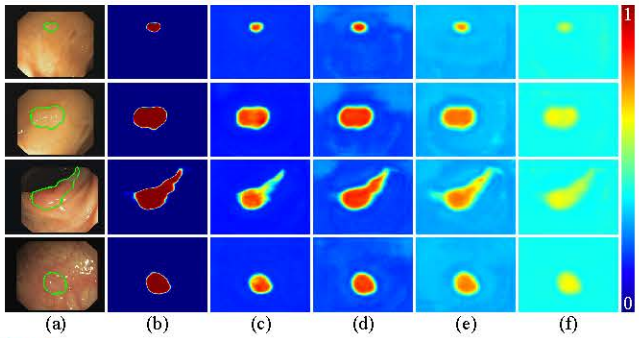

Fig. 8: Illustration of threshold maps: (a) the input image with green contours outlining the polyp regions; (b) the likelihood map generated under the setting $m = 0.3$; (c-g) predicted threshold maps with margin m equals to 0.1, 0.2, 0.3, 0.4.

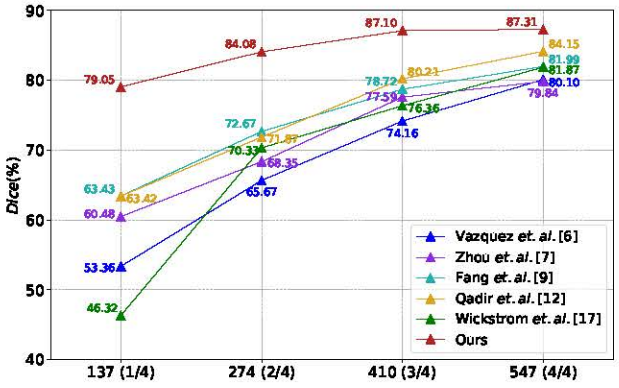

Fig. 9: Comparison results with different numbers of training images. "274 (2/4)" indicates 274 images (2/4 proportion of EndoScene training set) are involved for optimization.

V. Discussion

A. Different Numbers of Training Images

Adequate training images are essential to the optimization of deep neural networks, which can mitigate the overfitting problem of the learned model. To discuss the generalization of our method, we first assessed the performance of state-of-the-art polyp segmentation models [6], [7], [9], [12], [17] and the proposed model trained with different numbers of training images. The training samples were varied from 1/4 to 4/4 of the total training set (547 images) in increments of 1/4, and we drew the $Dice$ curve for comparison, as shown in Fig. 9. In general, our method shows a relatively stable performance with different training data settings, demonstrating the robustness of our approach. Moreover, it is clear that the proposed method consistently performs better than state-of-the-art methods in different number of training data settings. When the number of training images is small (137 images), other competed methods exhibit unsatisfactory performance as a result of overfitting problem. On the contrary, our method obtains a promising performance and possesses superior generalization ability for polyp segmentation with 25.69%, 18.57%, 15.62%, 15.63%, 32.73% increments in $Dice$ compared to methods [6], [7], [9], [12], [17]. This observation demonstrates that the proposed method can help prevent overfitting effectively. Notably, our approach is highly promising in medical image segmentation where pixel-level segmentation annotations are limited.

TABLE VI: Performance of our method on different segmentation baseline networks.

Methods	*Dice*	*Jac*	*Sen*	*Spe*	*Acc*	*F2*
UNet [35]	76.07	67.12	75.76	99.05	95.63	75.21
Ours (UNet)	**83.34**	**75.59**	**83.57**	**99.31**	**96.56**	**83.05**
P-value	<0.001	<0.001	<0.001	0.001	<0.001	<0.001
SegNet [36]	81.87	74.54	82.13	99.29	96.64	81.73
Ours (SegNet)	**84.51**	**77.31**	**84.05**	**99.34**	**96.77**	**83.89**
P-value	<0.001	0.001	0.041	0.004	0.048	0.017
OCNet [37]	82.25	74.09	84.12	99.11	96.36	82.36
Ours (OCNet)	**85.27**	**78.14**	**85.64**	**99.39**	**96.77**	**84.86**
P-value	<0.001	<0.001	0.026	<0.001	<0.001	<0.001
DenseASPP [38]	85.60	78.73	85.10	99.46	96.96	84.70
Ours (DenseASPP)	**87.80**	**81.05**	**88.26**	**99.49**	**97.31**	**87.79**
P-value	<0.001	<0.001	0.011	0.046	0.001	<0.001
DeepLabv3+ [34]	82.52	74.93	82.30	99.31	96.44	82.02
Ours (DeepLabv3+)	**87.31**	**80.57**	**87.97**	**99.47**	**97.21**	**87.28**
P-value	<0.001	<0.001	<0.001	0.001	<0.001	<0.001

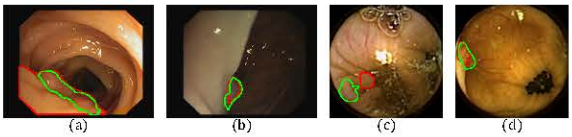

Fig. 10: Illustration of failure cases in EndoScene (a, b) and WCE polyp (c, d) datasets. Red and green contours outline the ground truth and our prediction of polyp boundary.

B. Comparisons with Different Baseline Networks

The proposed ThresholdNet with CGMMix data augmentation method was performed on the baseline of DeepLabv3+ [34] in afore experiments. To verify the effectiveness and generalization of the proposed approach, we further integrated the proposed method to other segmentation baselines: UNet [35], SegNet [36], OCNet [37] and DenseASPP [38]. Note that OCNet and DenseASPP were with the backbones of ResNet50 and DenseNet121, and these two backbones were initialized with pertained parameters. Table VI summarizes the overall comparison results on different baseline models. We can observe that our approach can significantly improve the results over different baselines, UNet, SegNet, OCNet, DenseASPP, DeepLabv3+, with increments of 8.47%, 2.77%, 4.05%, 2.32%, 5.64% in Jac and 7.84%, 2.16%, 2.50%, 3.09%, 5.26% in $F2$ scores, respectively. The statistical significance P-values derived by the paired t-test reveal that the proposed approach is an efficient strategy to boost the performance of existing segmentation networks. Moreover, compared with the baseline model DeepLabv3+, our method only requires 1.23 times floating-point operations (FLOPs) and 1.02 times parameters (i.e., 2.74×10^{10} FLOPs, 6.05×10^{7} parameters), demonstrating the computational efficiency.

C. Failure Cases

Although the proposed method shows promising performance, it does make erroneous predictions in certain cases, as in Fig. 10. Sometimes polyp and normal tissues share high degree similarity of appearance, yielding wrong predictions, such as Fig. 10 (a, c). The proposed method may give biased predictions at shadow regions (Fig. 10 (b)) or ambiguous boundaries (Fig. 10 (d)). Auxiliary angular contrastive constraint [39] will be further explored to enhance the discriminative ability and improve the polyp segmentation performance.

VI. Conclusion

Automatic polyp segmentation is a challenging task due to the limited pixel-wise annotated dataset and the class imbalanced data distribution. Moreover, thresholding the likelihood map with an eclectic constant to obtain final segmentation results is problematic. In this paper, we propose a ThresholdNet with CGMMix data augmentation method to tackle the aforementioned issues. CGMMix conducts manifold mixup at multiple levels for data augmentation and is able to reach a trade-off between sensitivity and specificity with the confidence guidance. The MFMC and MCMC losses are designed to ensure the robust training of the mixup data. The proposed ThresholdNet collaborates the segmentation and threshold learning in a robust way, i.e., these two branches are reciprocally propagated and constrained throughout the whole learning process. Extensive experiments on two polyp segmentation datasets demonstrate the superiority of our method compared with state-of-the-art methods. In addition, although our model is built upon the specific application of polyp segmentation, the proposed approach is a generic and general strategy that could be flexibly applied to extensive medical image segmentation tasks.

References

[1] R. L. Siegel, K. D. Miller, and A. Jemal, "Cancer statistics, 2020," *CA: Cancer J. Clin.*, vol. 70, no. 1, pp. 7–30, 2020.

[2] R. A. Smith, K. S. Andrews, D. Brooks, S. A. Fedewa, D. Manassaram-Baptiste, D. Saslow, O. W. Brawley, and R. C. Wender, "Cancer screening in the united states, 2018: a review of current american cancer society guidelines and current issues in cancer screening," *CA: Cancer J. Clin.*, vol. 68, no. 4, pp. 297–316, 2018.

[3] Y. Yuan, B. Li, and M. Q.-H. Meng, "WCE abnormality detection based on saliency and adaptive locality-constrained linear coding," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 1, pp. 149–159, 2016.

[4] Y. Yuan, D. Li, and M. Q.-H. Meng, "Automatic polyp detection via a novel unified bottom-up and top-down saliency approach," *IEEE J. Biomed. Health Inform.*, vol. 22, no. 4, pp. 1250–1260, 2017.

[5] X. Guo and Y. Yuan, "Triple ANet: Adaptive abnormal-aware attention network for WCE image classification," in *MICCAI*, 2019, pp. 293–301.

[6] D. Vázquez, J. Bernal, F. J. Sánchez, G. Fernández-Esparrach, A. M. López, A. Romero, M. Drozdzal, and A. Courville, "A benchmark for endoluminal scene segmentation of colonoscopy images," *J Healthc Eng.*, vol. 2017, pp. 1–9, 2017.

[7] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *DLMIA*. Springer, 2018, pp. 3–11.

[8] M. Akbari, M. Mohrekesh, E. Nasr-Esfahani, S. R. Soroushmehr, N. Karimi, S. Samavi, and K. Najarian, "Polyp segmentation in colonoscopy images using fully convolutional network," in *IEEE Eng. Med. Biol. Soc.*, 2018, pp. 69–72.

[9] Y. Fang, C. Chen, Y. Yuan, and K.-y. Tong, "Selective feature aggregation network with area-boundary constraints for polyp segmentation," in *MICCAI*, 2019, pp. 302–310.

[10] J. Poomeshwaran, K. S. Santhosh, K. Ram, J. Joseph, and M. Sivaprakasam, "Polyp segmentation using generative adversarial network," in *EMBC*, 2019, pp. 7201–7204.

[11] J. Kang and J. Gwak, "Ensemble of instance segmentation models for polyp segmentation in colonoscopy images," *IEEE Access*, vol. 7, pp. 26 440–26 447, 2019.

[12] H. A. Qadir, Y. Shin, J. Solhusvik, J. Bergsland, L. Aabakken, and I. Balasingham, "Polyp detection and segmentation using mask R-CNN: Does a deeper feature extractor cnn always perform better?" in *ISMICT*, 2019, pp. 1–6.

[13] N.-Q. Nguyen and S.-W. Lee, "Robust boundary segmentation in medical images using a consecutive deep encoder-decoder network," *IEEE Access*, vol. 7, pp. 33 795–33 808, 2019.

[14] L. Wang, R. Chen, S. Wang, N. Zeng, X. Huang, and C. Liu, "Nested dilation network (NDN) for multi-task medical image segmentation," *IEEE Access*, vol. 7, pp. 44 676–44 685, 2019.

[15] B. Murugesan, K. Sarveswaran, S. M. Shankaranarayana, K. Ram, J. Joseph, and M. Sivaprakasam, "Psi-Net: Shape and boundary aware joint multi-task deep network for medical image segmentation," in *EMBC*, 2019, pp. 7223–7226.

[16] H. A. Qadir, J. Solhusvik, J. Bergsland, L. Aabakken, and I. Balasingham, "A framework with a fully convolutional neural network for semi-automatic colon polyp annotation," *IEEE Access*, vol. 7, pp. 169 537–169 547, 2019.

[17] K. Wickstrøm, M. Kampffmeyer, and R. Jenssen, "Uncertainty and interpretability in convolutional neural networks for semantic segmentation of colorectal polyps," *Med Image Anal*, vol. 60, p. 101619, 2020.

[18] M. Bagheri, M. Mohrekesh, M. Tehrani, K. Najarian, N. Karimi, S. Samavi, and S. R. Soroushmehr, "Deep neural network based polyp segmentation in colonoscopy images using a combination of color spaces," in *EMBC*, 2019, pp. 6742–6745.

[19] X. Jia, X. Xing, Y. Yuan, L. Xing, and M. Q.-H. Meng, "Wireless capsule endoscopy: A new tool for cancer screening in the colon with deep-learning-based polyp recognition," *Proc. IEEE*, vol. 108, no. 1, pp. 178–197, 2019.

[20] X. Jia, X. Mai, Y. Cui, Y. Yuan, X. Xing, H. Seo, L. Xing, and M. Q.-H. Meng, "Automatic polyp recognition in colonoscopy images using deep learning and two-stage pyramidal feature prediction," *IEEE Trans. Autom. Sci. Eng.*, 2020.

[21] D. Jha, P. H. Smedsrud, M. A. Riegler, D. Johansen, T. De Lange, P. Halvorsen, and H. D. Johansen, "ResUNet++: An advanced architecture for medical image segmentation," in *ISM*, 2019, pp. 225–2255.

[22] N. Ibtehaz and M. S. Rahman, "MultiResUNet: Rethinking the u-net architecture for multimodal biomedical image segmentation," *Neural Networks*, vol. 121, pp. 74–87, 2020.

[23] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *ICLR*, 2018. [Online]. Available: https://openreview.net/forum?id=r1Ddp1-Rb

[24] M. Xu, J. Zhang, B. Ni, T. Li, C. Wang, Q. Tian, and W. Zhang, "Adversarial domain adaptation with domain mixup," *arXiv preprint arXiv:1912.01805*, 2019.

[25] K. Chaitanya, N. Karani, C. F. Baumgartner, A. Becker, O. Donati, and E. Konukoglu, "Semi-supervised and task-driven data augmentation," in *IPMI*, 2019, pp. 29–41.

[26] Q. Wang, W. Li, and L. V. Gool, "Semi-supervised learning by augmented distribution alignment," in *ICCV*, 2019, pp. 1466–1475.

[27] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *NeurIPS*, 2019, pp. 5050–5060.

[28] E. Panfilov, A. Tiulpin, S. Klein, M. T. Nieminen, and S. Saarakkala, "Improving robustness of deep learning based knee mri segmentation: Mixup and adversarial domain adaptation," in *ICCV Workshops*, 2019, pp. 0–0.

[29] Z. Li, K. Kamnitsas, and B. Glocker, "Overfitting of neural nets under class imbalance: Analysis and improvements for segmentation," in *MICCAI*, 2019, pp. 402–410.

[30] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "AugMix: A simple data processing method to improve robustness and uncertainty," *arXiv preprint arXiv:1912.02781*, 2019.

[31] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, A. Courville, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," *arXiv preprint arXiv:1806.05236*, 2018.

[32] Y. Wang, N. Wang, M. Xu, J. Yu, C. Qin, X. Luo, X. Yang, T. Wang, A. Li, and D. Ni, "Deeply-supervised networks with threshold loss for cancer detection in automated breast ultrasound," *IEEE Trans. Med. Imag.*, 2019.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NeurIPS*, 2012, pp. 1097–1105.

[34] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018, pp. 801–818.

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015, pp. 234–241.

[36] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, 2017.

[37] Y. Yuan and J. Wang, "Ocnet: Object context network for scene parsing," *arXiv preprint arXiv:1809.00916*, 2018.

[38] M. Yang, K. Yu, C. Zhang, Z. Li, and K. Yang, "Denseaspp for semantic segmentation in street scenes," in *CVPR*, 2018, pp. 3684–3692.

[39] X. Guo and Y. Yuan, "Semi-supervised WCE image classification with adaptive aggregated attention," *Med. Image Anal.*, p. 101733, 2020.