IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 6, 2022, accepted April 8, 2022, date of publication April 18, 2022, date of current version April 28, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3168693

BlazeNeo: Blazing Fast Polyp Segmentation and Neoplasm Detection

NGUYEN S. AN [1], PHAN N. LAN [1], DAO V. HANG [2,3], DAO V. LONG [2,3], TRAN Q. TRUNG [4], NGUYEN T. THUY [5], AND DINH V. SANG [1]

[1] School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi 10000, Vietnam
[2] Internal Medicine Faculty, Hanoi Medical University, Hanoi 11520, Vietnam
[3] Institute of Gastroenterology and Hepatology, Hanoi 11522, Vietnam
[4] Department of Internal Medicine,University of Medicine and Pharmacy, Hue University, Hue 53000, Vietnam
[5] Faculty of Information Technology, Vietnam National University of Agriculture, Hanoi 12406, Vietnam

Corresponding author: Dinh V. Sang (sangdv@soict.hust.edu.vn)

This work was supported by the Vingroup Innovation Foundation (VINIF) under Project VINIF.2020.DA17.

ABSTRACT In recent years, computer-aided automatic polyp segmentation and neoplasm detection have been an emerging topic in medical image analysis, providing valuable support to colonoscopy procedures. Attentions have been paid to improving the accuracy of polyp detection and segmentation. However, not much focus has been given to latency and throughput for performing these tasks on dedicated devices, which can be crucial for practical applications. This paper introduces a novel deep neural network architecture called BlazeNeo, for the task of polyp segmentation and neoplasm detection with an emphasis on compactness and speed while maintaining high accuracy. The model leverages the highly efficient HarDNet backbone alongside lightweight Receptive Field Blocks and a feature aggregation mechanism for computational efficiency. An auxiliary training strategy is proposed to take full advantage of the training data for the segmentation quality. Our experiments on a challenging dataset show that BlazeNeo achieves improvements in latency and model size while maintaining comparable accuracy against state-of-the-art methods. We obtain over 155 fps while outperforming all compared models in terms of accuracy in INT8 precision when deploying on a dedicated edge device with a conventional configuration.

INDEX TERMS Semantic segmentation, polyp segmentation, deep learning, colonoscopy.

I. INTRODUCTION

Colorectal polyps, especially adenomas with high-grade dysplasia, carry high risks of progressing into colorectal cancer (CRC) [10], which claims over 640,000 lives each year [3]. There are available procedures to screen and detect high-risk polyps in an early stage, increasing the chances of successful treatment. Polyp detection and removal in colonoscopy are the most effective method to prevent colorectal cancer [14].

In practice, factors such as overloading healthcare systems, low-quality endoscopy equipment, or personnel's lack of experience [2], [21] can severely limit the effectiveness of colonoscopy. A review by Leufkens *et al.* [22] pointed out that $20 - 47\%$ of polyps might have been missed during colonoscopies. Several types of image-enhanced endoscopies and accessories have been proposed to alleviate these, yet they can be prohibitively expensive for practical applications, especially in medical clinics with poorly equipped facilities. On the other hand, computer-aided systems for colonoscopy have shown a lot of promise and have attracted many researchers in recent years. Several works have achieved very good performance on benchmark datasets [9], [12], [39].

Polyp segmentation is a subset of medical image analysis that has gained much attention recently. Traditional machine learning methods for solving the problem are mostly based on hand-crafted features [15], [35] to extract image information such as color, shape, and textures. Since polyps have very high intra-class diversity and low inter-class variation, such approaches are often limited in representing and detecting polyps. Deep neural networks, and especially U-Net [32] have been the state-of-the-art methods for polyp segmentation in the last few years. These networks can learn highly abstract and complex features, allowing them to achieve good performances. At the same time, deep neural networks also

The associate editor coordinating the review of this manuscript and approving it for publication was Marco Anisetti.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

come with a complexity trade-off, as models can be very large (up to several hundred million parameters) and cause high latency during inference.

Most of the existing research in endoscopy image analysis has focused on the polyp segmentation problem, in which lesion regions or polyps are segmented from the background pixels. Those works attempted to improve the learning models to provide accurate segmentation of polyps. However, the segmentation of polyps only does not provide information about the type of polyps, i.e., benign (non-cancerous) or malignancy (presence of cancerous cells). In our seminal work [20], we have defined the problem of Polyp Segmentation and Neoplasm Detection (PSND), aiming at fine-grained segmentation of polyps. This can be considered an extension of the polyp segmentation problem, providing richer semantic information for the segmented regions. Particularly, besides classifying image pixels as polyp or background, the proposed formulation further identifies each polyp pixel as non-neoplastic, neoplastic, or undefined. In general, non-neoplastic polyps are typically benign, while neoplastic polyps have a risk of developing cancer. Our newly developed UNet-based neural network architecture, NeoUNet, has obtained state-of-the-art performance in solving this problem in terms of accuracy. However, attention has not yet been paid to the model size and speed, which is challenging for practical deployment.

In this paper, we further improve on our previous works on Polyp Segmentation and Neoplasm Detection with the proposal of BlazeNeo, a novel deep neural network architecture with an efficient learning mechanism. Our main contributions are:

- A new deep neural network architecture called BlazeNeo designed with a lightweight encoder-decoder and an efficient feature aggregation for polyp segmentation and neoplasm detection. The design aims at reducing the model size and therefore improving inference speed;
- An auxiliary training strategy to fully exploit informative features in the training data for maintaining high accuracy while reducing model size;
- Extensive experiments on the newly collected NeoPolyp dataset and comparisons to existing models. Moreover, we measure model latency and throughput on dedicated hardware in a setting similar to real-life deployments of polyp segmentation and neoplasm detection.

The rest of the paper is organized as follows. We provide a brief review of related works in Section II, including a brief description of the polyp segmentation and neoplasm detection (PSND) problem originally formulated in [20]. The BlazeNeo architecture is presented in Section III. Section IV showcases our experimental studies. Finally, we conclude the paper and highlight future works in Section VI.

II. RELATED WORK

In recent years, many computer vision tasks have seen massive improvements through the advancements of convolutional neural networks (CNNs). AlexNet [19] and VGG [36] are among the first successful CNNs for image classification problems. However, these early models still suffer from degradation when increasing network depth. Many works have attempted to modify the network architectures to improve learning capability aiming at improving network performance. Skip connections, first introduced in ResNet [11] in 2016, helped alleviate the degradation and smoothed out the loss landscape. ResNeXt [43] combined the idea of skip connections with a multi-branch design first proposed by the authors of GoogLeNet [37]. More recently, Tan and Le [38] employed neural architecture search to produce EfficientNet, a family of neural networks with varying levels of the trade-off between accuracy and latency. Meanwhile, HarDNet [4] is a model highly focused on optimizing inference latency and memory traffic.

Many CNN architectures have been designed for the semantic image segmentation task, especially in medical images. Among the earliest was the work by Long *et al.* [27], who adopted several well-known architectures using transfer learning. In the same year, U-Net [32] became a breakthrough model in medical imaging, achieving highly promising results for medical image segmentation. Later works such as UNet++ [47], DoubleUNet [17] and Coupled U-Net [39] further improved and alleviated limitations in U-Net. ColonSegNet [16] was a lightweight encoder-decoder architecture that uses residual connections with squeeze-and-excitation networks as the main components. ColonSegNet achieved a high inference speed but with significant sacrificing accuracy. DDANet [40] was another encoder-decoder design that leverages the strength of residual connection and squeeze-and-excitation modules. DDANet incorporated a single encoder followed by two dual decoders. The first decoder is used for the segmentation mask, while the second one acts as an autoencoder model that reconstructs the grayscale image and helps strengthen the feature representation of the encoder. Attention-UNet [29] proposed attention gates as a filter mechanism for selecting useful salient features. Despite the dominance of UNet-based approaches, the development of non-UNet models in medical segmentation has also remained active. DeepLabV3 [6] is a prominent architecture that utilizes atrous convolutions for dense feature extraction. Fan *et al.* [9] enhanced an FCN-like model with parallel partial decoder and reverse attention to form PraNet, a network that achieved state-of-the-art performance on many benchmark datasets. HarDNet-MSEG [12] employed an encoder-decoder structure with HarDNet as the encoder backbone, achieving good performance on the Kvasir-SEG dataset and very high inference speed. Meanwhile, TransFuse [46] combined a CNN with the Transformer architecture using a fusion module called BiFusion.

Many deep learning methods are also specially designed for the polyp segmentation and detection problem. Qadir *et al.* [30] proposed a framework that incorporates a CNN architecture for labeling segmentation masks for polyps. The framework allows medical doctors to receive

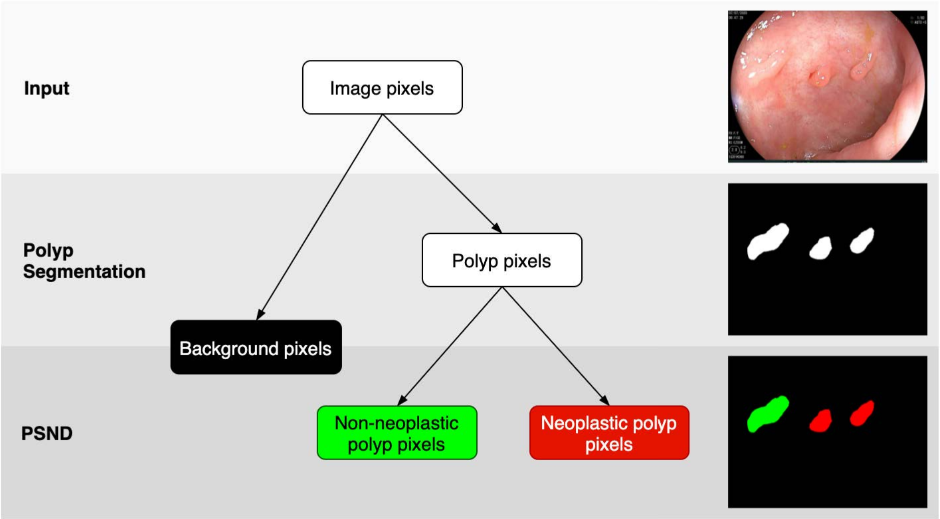


FIGURE 1. PSND [20] as an extension of polyp segmentation, which further discriminates whether a polyp is neoplastic or non-neoplastic. For a given input image shown on the top right corner, expected outputs for polyp segmentation and PSND are depicted in the middle and the bottom images on the right, respectively. The black color denotes background pixels, white color denotes polyp regions; green and red colors denote non-neoplastic and neoplastic polyps, respectively.

pre-annotations from a model trained in a semi-supervised manner. In [33], the authors proposed a model with an Inception-ResNet backbone combined with several post-learning methods to enhance polyp detection accuracy. Shin *et al.* [34] used conditional adversarial networks to generate abnormal samples for training polyp detection models. Liu *et al.* [24] applied different CNN backbones including InceptionV3 [37], ResNet50 [11] and VGG16 [36] to the SSD framework, whose accuracy was much higher than other one-stage object detectors and comparable to the two-stage Faster-RCNN. In [23], Li *et al.* compared the performance of eight state-of-the-art deep learning object detectors and demonstrated promising results in colonoscopic image analysis.

There have been few attempts in the past for the problem of polyp neoplasm detection [31]). It was only recently that neoplasm detection was incorporated into polyp segmentation. Lan *et al.* [20] formally described the problem of polyp segmentation and neoplasm detection (denoted as PSND) and proposed NeoUNet, a UNet-based architecture that established the baseline for the PSND.

Polyp Segmentation and Neoplasm Detection (PSND) has been formulated as a type of fine-grained polyp segmentation problem. Besides segmentation of polyps, this formulation further classifies a polyp pixel into two classes: non-neoplastic or neoplastic. In medical image analysis, non-neoplastic polyps are considered benign, while neoplastic polyps may progress with a risk of cancer. During a colonoscopy procedure, the doctor must decide immediately on the types of polyps, neoplasm or non-neoplastic, to consider an optimal management strategy, i.e., removal or resection during the endoscopy procedure or biopsy then operation. This requires low latency from the detection model. While NeoUNet has obtained promising accuracy, it has not yet addressed the inference speed to ensure smooth operation in real applications. Moreover, practical deployments would have the neural network run on lightweight systems embedded in endoscopic devices. This paper focuses on building a light network architecture for fast inference while improving polyp segmentation and neoplasm detection accuracy.

III. PROPOSED METHOD

Figures 1 depicts the PSND problem as a semantic extension of the polyp segmentation problem. In practice, missed detection of polyps is often associated with small and flat regions. This makes the problem challenging. Besides, in practice, an "undefined" class is labeled if there is not enough information from the endoscopic image to categorize the risk of neoplasm. This undefined subgroup is not a specific class we want to predict since such predictions would not bring any insight to the endoscopist. Therefore, the model only needs to learn to discriminate between neoplastic and non-neoplastic polyps. However, the undefined class still gives supplementary information about polyp regions that can be exploited to help the model gain more representation power in the training phase.

We propose BlazeNeo model with three versions for processing the outputs, as depicted in Figure 2. Inspired by the lightweight model HarDNet-MSEG [12], our models are developed with several improvements tailored for the PSND problem. First, we use a simplified version of RFB [25] with smaller kernel sizes and apply different feature aggregation schemes. For the encoder layer, HarDNet-68 [4] is used as the backbone.

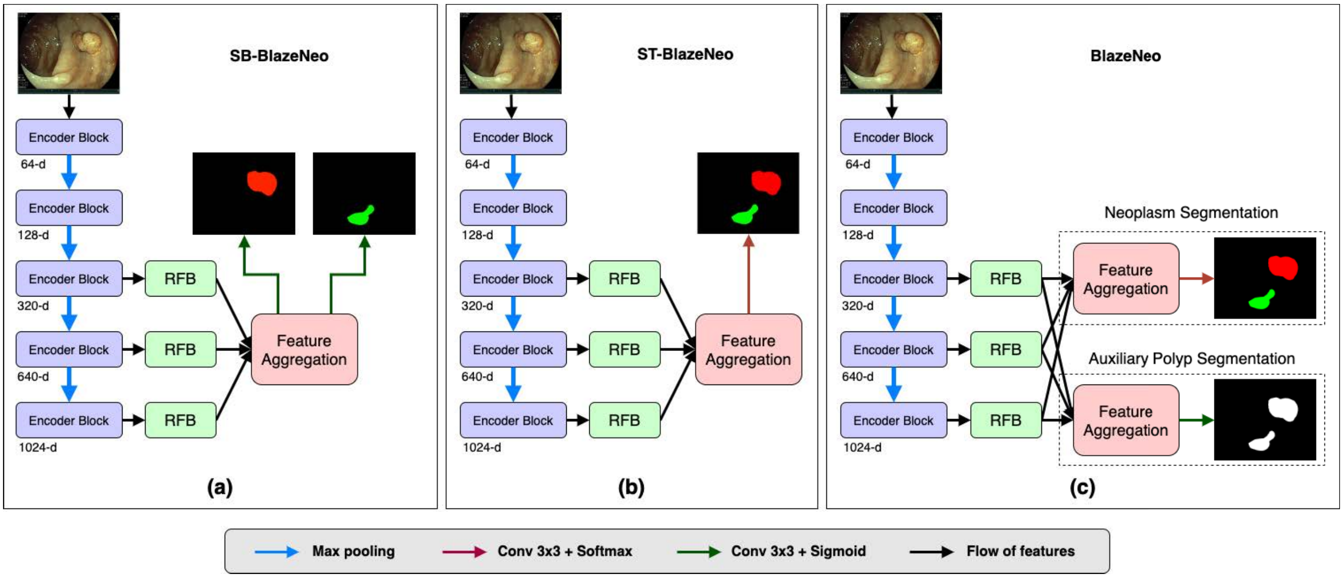


FIGURE 2. Proposed architectures of our BlazeNeo: (a) Single-headed Binary BlazeNeo (SB-BlazeNeo) has one output branch that produces two binary segmentation maps corresponding to neoplastic and non-neoplastic classes; (b) Single-headed Trinary BlazeNeo (ST-BlazeNeo) also has one output branch that directly predicts a trinary segmentation map; (c) Multi-headed BlazeNeo (or BlazeNeo for short) contains two output branches that are responsible for the two tasks: neoplasm segmentation treated as the main task, and polyp segmentation treated as the auxiliary task. Both branches share the same architecture of the feature aggregation module, but they are trained separately without sharing their parameters.

The three BlazeNeo variants differ in how outputs from the model are generated and processed. Inspired by NeoUNet [20], Single-headed Binary BlazeNeo (SB-BlazeNeo) solves two binary segmentation tasks corresponding to neoplastic and non-neoplastic classes. The second variant, Single-headed Trinary BlazeNeo (ST-BlazeNeo), predicts a trinary map for three classes in the neoplasm segmentation task. Finally, Multi-headed BlazeNeo (or BlazeNeo for short) uses two output branches. The main output branch is used for the neoplasm segmentation task, and the auxiliary branch is for the polyp segmentation task. The following subsections describe our design, model's architecture, and training strategies in detail.

A. LIGHTWEIGHT ENCODER: HarDNet

HarDNet (Harmonic Densely Connected Network) [4] is an improvement over DenseNet [13]. The primary goal of HarDNet's design is to lower latency by reducing memory traffic. The authors argued that the connection pattern of Dense Blocks, in which each layer has skip connections toward every proceeding layer in the block, causes ineffective memory access during runtime that severely hinders performance. HarDNet reduces the number of skip connections to form a pattern similar to the harmonic wave function, as well as scaling channel width according to a layer's influence level. We illustrate the difference between the two architectures in Figure 3.

Inside a HarDNet block, each layer is indexed from the input layer 0. A layer l receives a skip connection from layer $l - 2^n$ if 2^n divides l ($n \geq 0, l - 2^n \geq 0$). Given the initial growth rate k and a compression factor m, layer l's channel width is equal to $k \times m^x$, where $x = max\{v \mid l \vdots 2^v\}$.

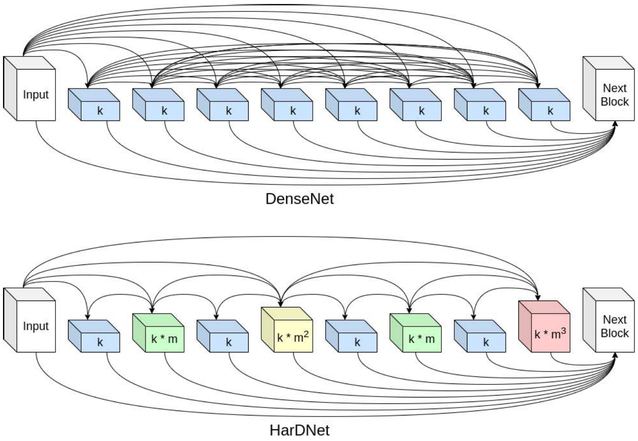


FIGURE 3. Illustrations for DenseNet block and Harmonic DenseNet (HarDNet) block. Each of the layers is a 3 × 3 convolution. The value on each layer denotes the number of output channels.

Chao *et al.* [4] further proposed HarDNet-68 for detection of small objects. While most CNNs focus on stride-16 to enhance classification ability, HarDNet-68 distributes most of the layers on stride-8 to aid small-scale object detection, as shown in Figure 4. Experiments in [4] show that HarDNet-68 is not only 30% faster than ResNet-50 [11] but also more accurate than ResNet-101 [11] when used as backbone for SSD [26] in the object detection problem.

Our BlazeNeo also uses HarDNet-68 as the encoder backbone. For an input colonoscopy image I with size $h \times w$, five levels of features $\{f_i, i = 1, 2, 3, 4, 5\}$ with resolution $[h/2^{i-1}, w/2^{i-1}]$ are produced from the encoder. Wu *et al.* [41] showed that low-level features (corresponding to f_1 and f_2) contribute less to the performance while being

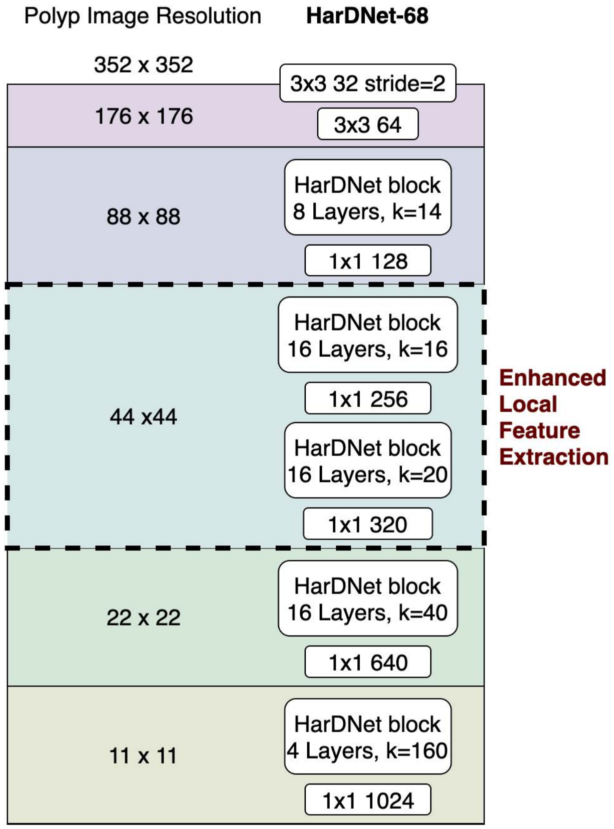

Polyp Image Resolution
HarDNet-68
352 x 352
3x3 32 stride=2
176 x 176
3x3 64
88 x 88
HarDNet block 8 Layers, k=14
1x1 128
44 x44
HarDNet block 16 Layers, k=16
1x1 256
HarDNet block 16 Layers, k=20
1x1 320
Enhanced Local Feature Extraction
22 x 22
HarDNet block 16 Layers, k=40
1x1 640
11 x 11
HarDNet block 4 Layers, k=160
1x1 1024

FIGURE 4. An overview of HarDNet-68 architecture. Following each HarDNet block is a transitional Conv 1 × 1 layer.

computationally expensive due to their size. Therefore, our BlazeNeo discards f_1 and f_2, using only the last three feature maps for the decoder module to accelerate its inference speed.

B. PARALLEL PARTIAL DECODER

In all three variants of our BlazeNeo, the decoder module consists of a Receptive Field Block (RFB) series. Three last feature maps of the encoder are independently passed through the RFB blocks and then fused by the feature aggregation blocks.

1) RECEPTIVE FIELD BLOCK

Polyps can appear in various scales on endoscopic images depending on their actual size, their distance to the colonoscopy camera, or the angle between them and the camera. This is a challenge for CNN architectures, in which the receptive field size is often fixed. Several studies have suggested different mechanisms to create more robust receptive fields, including the Inception block [37], ASPP block [5], and Deformable Convolution block [8]. Conceptually, these proposals are similar in that they all use multiple convolutional branches with different kernel sizes, merging the outputs to form adaptive receptive fields. However, they also have their own limitations. The Inception block samples all the kernels of each branch at their center, ignoring crucial edge details due to small sampling coverage. Meanwhile, the ASPP and Deformable Convolution blocks do not differentiate between different pixel positions, making it difficult for the model to focus on segmentation targets.

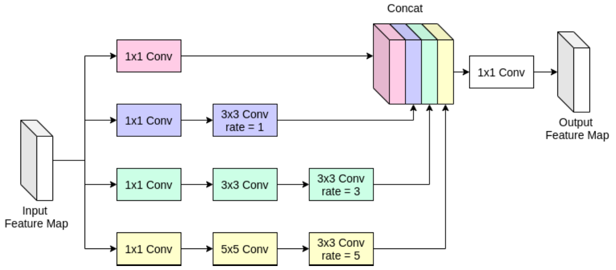


FIGURE 5. Structure of the Receptive Field Block (RFB).

The Receptive Field Block (RFB) [25] has been proposed to address these limitations. RFB also uses the multi-branch convolution approach, with improvements inspired by the human visual cortex. In addition, it highlights the importance of the region nearer to the center and elevates the insensitivity to small spatial shifts.

The RFB module used in PraNet [9] and HarDNet-MSEG [12] is modified with a larger kernel size and larger dilation rate, which makes it more computationally expensive. Our BlazeNeo instead uses a simplified version of RFB (see Figure 5) as proposed in [25] for faster inference.

2) FEATURE AGGREGATION

A high polyp miss rate is often associated with small and flat polyps (whose perimeters are below 10*mm*) [18]. In order to detect these polyps, it is important to obtain high-resolution features from multiple image scales. Feature fusion (or aggregation) is a well-studied technique to achieve this capability for CNNs, in which feature maps from different scales are fused to form a multi-scale feature map. Figure 6 demonstrates four feature aggregation schemes, in order of increasing complexity [45].

Long Skip Connection (LSC) illustrated in Figure 6a is an early aggregation scheme used by segmentation networks such as UNet [32], UNet++ [47], and Attention UNet [29]. Given feature maps X_1, X_2, X_3, higher-level feature maps are upsampled and fused with their adjacent low-level features by a long skip connection, gradually restoring the spatial information. Each fusion module includes a concatenation layer and a convolutional layer with kernel size 3 × 3.

LSC is a well-tested and straightforward technique, but it is not without limitations. Zhang [45] proposed three alterations to produce higher-quality features, as described below.

Iterative Deep Aggregation (IDA) depicted in Figure 6b produces finer feature maps by using multiple iterative convolutions for a single scale.

Inspired by DenseNet, Dense Iterative Aggregation (DIA) introduces dense skip connections to the iterative convolutions in IDA (see Figure 6c). This addition ensures maximum information flow and reduces overfitting.

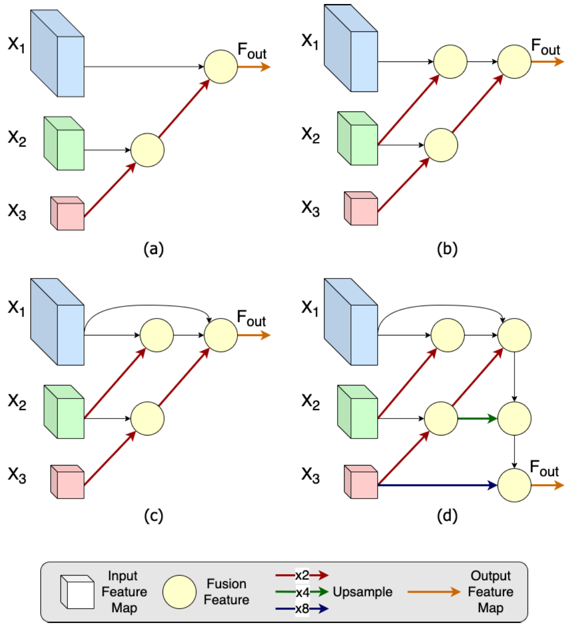


FIGURE 6. Different feature aggregation schemes as shown in [45]. (a) Long skip connection (LSC); (b) Iterative Deep Aggregation (IDA); (c) Dense Iterative Aggregation (DIA); (d) Dense Hierarchical Aggregation (DHA).

Dense Hierarchical Aggregation (DHA) further enhances semantic information by re-combining the output with high-level feature maps, as shown in Figure 6d.

From these observations, in our BlazeNeo, we propose to apply feature aggregation to the outputs of RFB modules at different scales. We examine variants with each of the aforementioned aggregation schemes, namely BlazeNeo-LSC, BlazeNeo-IDA, BlazeNeo-DIA, and BlazeNeo-DHA, in section IV.

C. LOSS FUNCTION AND AUXILIARY TRAINING

We aim to exploit the information from data with the undefined labels for training in a way similar to [20]. The intuition is that while these data do not provide information for deciding on neoplasm class, they still can provide some semantic meaning from the data for the segmentation.

To training BlazeNeo, we propose the loss function $\mathcal{L}_{total}$ consisting of two components: a main loss $\mathcal{L}_{main}$ associated with the main task of neoplasm segmentation, and an auxiliary loss $\mathcal{L}_{aux}$ associated with the auxiliary task of polyp segmentation. The total loss can be expressed as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{main} + \mathcal{L}_{aux} \tag{1}$$

The main loss $\mathcal{L}_{main}$ drives the model toward making accurate class-specific segmentation. The pixels with undefined labels are excluded when calculating $\mathcal{L}_{main}$. The auxiliary loss $\mathcal{L}_{aux}$ drives the model toward making accurate foreground-background segmentation, in which all pixels are used for training. Note that the auxiliary branch can be omitted during inference, so it will not require additional computing overhead.

To investigate this training strategy, we iterate through three different ways to incorporate it into BlazeNeo, which creates three different variants: Single-headed Binary BlazeNeo (SB-BlazeNeo), Single-headed Trinary BlazeNeo (ST-BlazeNeo), and Multi-headed BlazeNeo (the final version of BlazeNeo).

1) SINGLE-HEADED BINARY BlazeNeo

Our first variant, SB-BlazeNeo (Figure 2a), uses the same loss formulation as the NeoUNet model proposed in [20]. The final output layer produces two binary segmentation maps, one for the neoplastic class and another for the non-neoplastic class. Losses are calculated separately for each map and then averaged. The main loss is a combination of Binary Cross Entropy and Focal Tversky loss [1] as follows:

$$\begin{aligned}\mathcal{L}_{main} &= \mathcal{L}_{main}^{neo} + \mathcal{L}_{main}^{non}\\ &= \mathcal{L}_{BCE}(P_{main}^{neo}, G_{main}^{neo}) + \mathcal{L}_{FT}(P_{main}^{neo}, G_{main}^{neo})\\ &\quad + \mathcal{L}_{BCE}(P_{main}^{non}, G_{main}^{non}) + \mathcal{L}_{FT}(P_{main}^{non}, G_{main}^{non})\end{aligned} \tag{2}$$

where P_{main}^{neo} and P_{main}^{non} denote the prediction maps for the neoplastic and non-neoplastic classes, respectively; G_{main}^{neo} and G_{main}^{non} are ground truths; L_{BCE} and L_{FT} denote Binary Cross Entropy and Focal Tversky losses, respectively.

We choose the Focal Tversky loss for $\mathcal{L}_{main}$ to alleviate class imbalance due to the small amount of non-neoplastic polyp pixels, as shown later in Figure 8.

The auxiliary loss is a combination of Binary Cross Entropy and Tversky loss, which uses an auxiliary polyp segmentation map inferred from the two binary class-specific maps:

$$\mathcal{L}_{aux} = \mathcal{L}_{BCE}(P_{aux}^{polyp}, G_{aux}^{polyp}) + \mathcal{L}_{T}(P_{aux}^{polyp}, G_{aux}^{polyp}) \tag{3}$$

where P_{aux}^{polyp} and G_{aux}^{polyp} denote the auxiliary polyp prediction map and the corresponding ground truth, L_{FT} is Tversky loss.

The auxiliary polyp prediction map P_{aux}^{polyp} is inferred using element-wise max:

$$P_{aux}^{polyp} = \max(P_{main}^{neo}, P_{main}^{non}) \tag{4}$$

2) SINGLE-HEADED TRINARY BlazeNeo

While the method using binary map [20] yielded promising classification results, we found that a lighter model can benefit from imposing an additional constraint on the outputs. Specifically, when the model outputs two separate class-specific maps for one image, it may make both maps have high prediction values for the same pixel. Therefore, a class constraint is needed to ensure that one class is chosen for each pixel. In our second variant, ST-BlazeNeo (Figure 2b), we use a 3-channel output map $P_{main}^{trinary}$, denoting the probabilities for the neoplastic, non-neoplastic, and background class, respectively. A softmax activation is used on the channel dimension, meaning each pixel may only belong to one class.

The main loss is a combination of Categorical Cross Entropy and Focal Tversky loss as follows:

$$\begin{aligned}\mathcal{L}_{main} = & \ \mathcal{L}_{CCE}(P^{trinary}_{main}, G^{trinary}_{main}) \\ & + \mathcal{L}_{FT}(P^{trinary}_{main}, G^{trinary}_{main})\end{aligned} \quad (5)$$

where $G^{trinary}_{main}$ is the trinary ground truth.

Similarly, the auxiliary loss is a combination of Categorical Cross Entropy and Tversky loss:

$$\mathcal{L}_{aux} = \mathcal{L}_{CCE}(P^{polyp}_{aux}, G^{polyp}_{aux}) + \mathcal{L}_{T}(P^{polyp}_{aux}, G^{polyp}_{aux}) \quad (6)$$

Here we apply element-wise max to the two channels of the map $P^{trinary}_{main}$, which correspond to the neoplastic and non-neoplastic classes, to produce the auxiliary polyp segmentation map P^{polyp}_{aux}.

3) MULTI-HEADED BlazeNeo

We further evolve the use of auxiliary loss by adding an auxiliary segmentation branch. Auxiliary training is the process of jointly learning a *side* or *auxiliary* task to enhance the main task's performance [7], [44]. This idea is similar to multi-task learning, except the auxiliary branch is not activated in the inference phase.

The auxiliary branch uses an identical (with separate weights) feature aggregation module but outputs a binary segmentation map instead of a 3-channel multi-class map. The main loss and auxiliary loss are calculated exactly as in Eq. (5) and Eq. (6), respectively. However, there is no conversion between multi-class and polyp segmentation maps as in Eq. (4). Instead, the main loss is calculated with the 3-channel multi-class map, and the auxiliary loss is calculated with the output map from the auxiliary branch. Intuitively, we believe the model will benefit from adding an auxiliary network branch since it strengthens the supervised signal and betters the optimization process during training the model. Figure 2c describes how BlazeNeo incorporates the auxiliary branch for training in our BlazeNeo.

IV. EXPERIMENTS

A. BENCHMARK DATASET

We use the NeoPolyp dataset as introduced in [20] to train and benchmark the proposed BlazeNeo. The dataset consists of 7,466 annotated endoscopic images captured directly during endoscopic recording and includes all four lighting modes: WLI (White Light Imaging), FICE (Flexible spectral Imaging Color Enhancement), BLI (Blue Light Imaging), and LCI (Linked Color Imaging). NeoPolyp is split into a training set of 5,966 images and a test set of 1,500 images. Some examples of the NeoPolyp dataset are shown in Figure 7. For comparison with baseline models, we also use the NeoPolyp-Clean dataset, which does not contain any polyps with undefined class labels. This dataset consists of 5,277 training images and 1,353 test images.

In practice, most non-neoplastic polyps are small, and the endoscopists can immediately remove them without the need to capture images or to take a biopsy for lesions less than 5mm for post-checking. Due to that reason, neoplastic polyps take up a majority of the polyps present in NeoPolyp (see Figure 8). The number of neoplastic, non-neoplastic, and undefined polyps are 5113, 3185, and 1031, respectively. However, if we look at the pixel-wise level as shown in Figure 8b, we can observe a strong data imbalance between the three classes. The number of neoplastic polyp pixels takes up to 80% of all polyp pixels in the dataset. Meanwhile, these numbers for non-neoplastic and undefined classes are 13% and 7%, respectively. This data imbalance, combined with the inherent challenges of PSND, makes it a difficult benchmark for models to overcome.

B. EXPERIMENT SETUP

Our experiments include several ablation studies to verify the effectiveness of each component in BlazeNeo, a comparison against several polyp segmentation methods, and benchmarks on the NVIDIA Jetson AGX Xavier developer kit,[1] which closely resembles real deployments. The Jetson device is configured to run at MAXN power mode.

We oversample non-neoplastic polyps to account for class imbalance in the NeoPolyp dataset, as addressed by [20]. Images containing non-neoplastic polyps are duplicated such that $P_{non} \approx P_{neo}$, where P_{non} and P_{neo} are the number of pixels containing in non-neoplastic and neoplastic polyps, respectively.

The models are trained using Stochastic Gradient Descent (SGD) with Nesterov momentum and an initial learning rate of 0.001. The learning rate is adjusted according to a combination of linear warmup and a cosine annealing schedule.

Images used for training are at 3 different scales: 256×256, 352 × 352 and 512 × 512.

During training, augmentations including random scaling, rotation, horizontal/vertical flip, motion blur, and color jittering are added to improve generality. These augmentations are performed on-the-fly with a probability of 0.5.

BlazeNeo and other baseline models are implemented in Python 3.7 using the PyTorch framework.

C. EVALUATION METRICS

Commonly used metrics Dice score and IoU score are employed to measure the model's output quality. They are evaluated in three classes: neoplastic polyp, non-neoplastic polyp, and generic polyp (same as polyp segmentation). Dice and IoU are calculated pixel-wise on the entire test set (micro-averaged). Equations (7) and (8) describe how these metrics are calculated.

$$Dice_c = \frac{2 \times \sum_{i \in I} u^c_i v^c_i}{\sum_{i \in I} u^c_i + \sum_{i \in I} v^c_i} \quad (7)$$

$$IoU_c = \frac{\sum_{i \in I} u^c_i v^c_i}{\sum_{i \in I} u^c_i + \sum_{i \in I} v^c_i - \sum_{i \in I} u^c_i v^c_i} \quad (8)$$

[1] https://developer.nvidia.com/embedded/jetson-agx-xavier-developer-kit

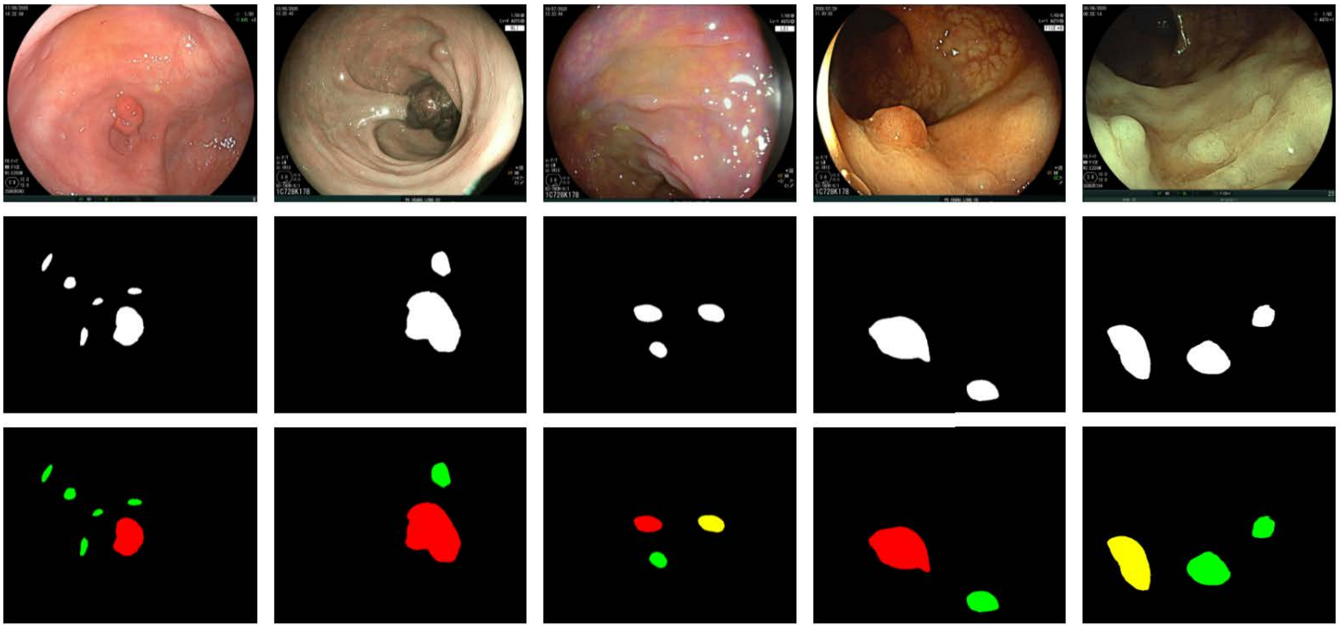
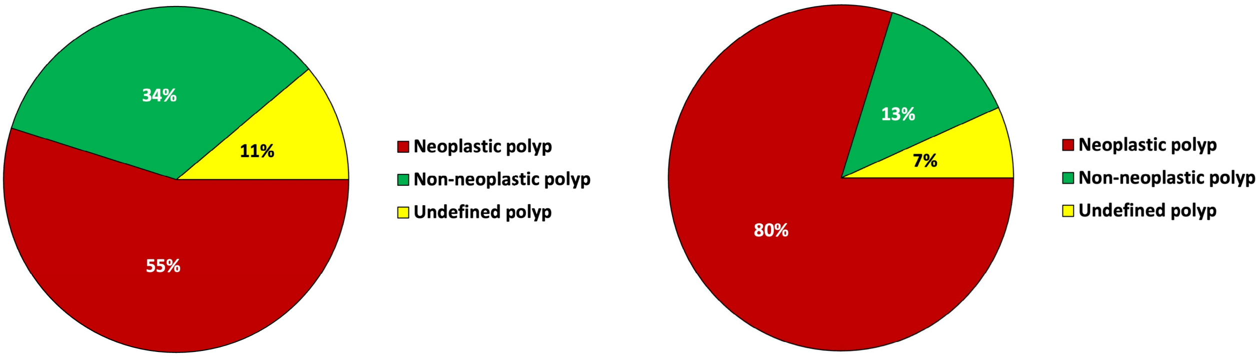


FIGURE 7. Some examples from the NeoPolyp dataset. The first row displays original images from the dataset. The second row shows the ground truths for polyp segmentation. The last row shows the ground truths for neoplasm segmentation, where some polyps are undefined and marked by yellow color. From left to right, the color modes are WLI, BLI, LCI, FICE, and FICE, respectively.



(a) Polyp-wise distribution

(b) Pixel-wise distribution

FIGURE 8. Data distribution of polyp class labels in the NeoPolyp dataset. In the pixel-wise distribution on the right, percentages are calculated on polyp pixels only (not including background pixels.).

where $i \in I$ denotes a prediction pixel within the entire test set. $u_i^c = 1$ if the model predicts pixel i to have class c, and 0 otherwise. Similarly, $v_i^c = 1$ if the ground truth map states that pixel i has class c, and 0 otherwise. For neoplastic and non-neoplastic class evaluation, I does not include undefined neoplasm pixels.

We also evaluate each model's inference speed using the number of processed frames per second (FPS). This metric is measured by running each model with a batch size of 1 on 100 colonoscopy images. When not specified otherwise, FPS is measured on a Google Colaboratory instance with an NVIDIA Tesla V100 GPU.

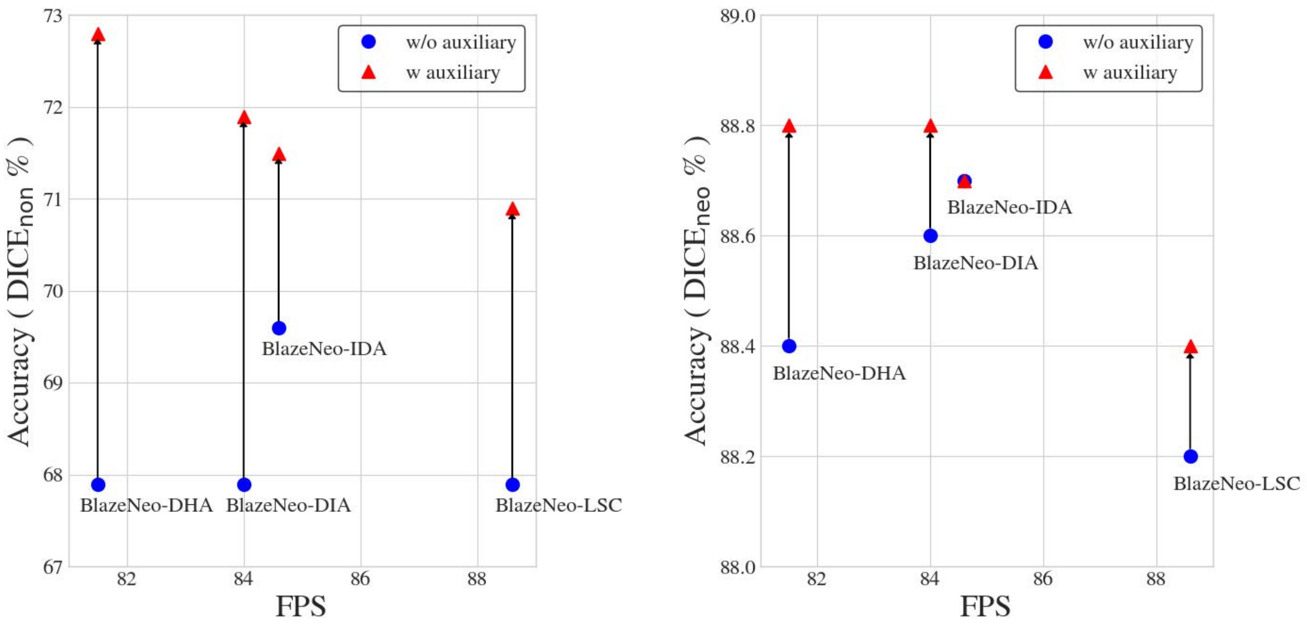
Finally, we log each model's number of parameters and floating-point operations (measured in GFLOPs) to evaluate their size and complexity.

V. RESULTS AND DISCUSSION

A. ABLATION STUDY

1) THE EFFECTIVENESS OF DIFFERENT OUTPUT ARCHITECTURES

Firstly, we compare our three BlazeNeo variants shown in Figure 2 to evaluate the effectiveness of different output architectures. Table 1 shows performance metrics for each variant on the NeoPolyp test set. Here we use the same DHA feature aggregation scheme for all three of BlazeNeo. The final BlazeNeo model with multi heads achieves the best results on all metrics. Notably, this final variant outperforms the other two variants by over 6% in IoU score for the non-neoplastic class. This shows the effectiveness of the auxiliary branch, which helps anchor segmentation performance and makes use of undefined labels. We also see that ST-BlazeNeo



(a) Non-neoplastic class

(b) Neoplastic class

FIGURE 9. Dice scores and FPS for different BlazeNeo variations, with and without auxiliary training. Red triangles denote results with auxiliary training, while blue circles are those without auxiliary training.

TABLE 1. Performance metrics on the NeoPolyp test set for the three variants of BlazeNeo using the same DHA feature aggregation scheme.

Model	Dice$_{seg}$	IoU$_{seg}$	Dice$_{non}$	IoU$_{non}$	Dice$_{neo}$	IoU$_{neo}$
SB-BlazeNeo	0.866	0.764	0.683	0.518	0.862	0.758
ST-BlazeNeo	0.874	0.777	0.671	0.505	0.865	0.762
BlazeNeo	**0.901**	**0.820**	**0.728**	**0.572**	**0.888**	**0.800**

outperforms SB-BlazeNeo, which justifies our use of trinary output in the final variant.

2) THE EFFECTIVENESS OF DIFFERENT FEATURE AGGREGATION SCHEMES

This experiment investigates the use of four different feature aggregation schemes in BlazeNeo: Long Skip Connection (LSC), Iterative Deep Aggregation (IDA), Dense Iterative Aggregation (DIA), and Dense Hierarchical Aggregation (DHA).

Table 2 shows performance metrics for each model variation on the NeoPolyp dataset. We can see that BlazeNeo-DHA produces the highest Dice and IoU scores for all classes. This is what we expected, as DHA is the most complex aggregation mechanism that preserves a lot of high-resolution features. However, this improvement comes at a cost as BlazeNeo-DHA is also the slowest variation at only 81.5 FPS, compared to the fastest BlazeNeo-LSC at 88.6 FPS. Interestingly, the nested IDA aggregation scheme performs worse than basic LSC in the segmentation task. This is alleviated in DIA and DHA with the use of skip connections.

3) THE EFFECTIVENESS OF AUXILIARY TRAINING

This experiment looks into the effectiveness of enabling and disabling auxiliary training for each variation of BlazeNeo, namely BlazeNeo-LSC, BlazeNeo-IDA, BlazeNeo-DIA, and BlazeNeo-DHA.

Figure 9 shows that auxiliary training generally improves output quality. In fact, without auxiliary segmentation learning, BlazeNeo-DIA and BlazeNeo-DHA achieve even lower accuracy than BlazeNeo-IDA. The non-neoplastic class benefits the most from auxiliary training, improving BlazeNeo-LSC, BlazeNeo-IDA, BlazeNeo-DIA, and BlazeNeo-DHA by 3%, 1.9%, 4%, and 4.9%, respectively.

4) THE EFFECTIVENESS OF INCLUDING UNDEFINED POLYPS

This experiment examines the effectiveness of using undefined neoplasm pixels via the auxiliary module. Table 3 shows performance metrics for BlazeNeo-DHA, the best-performing BlazeNeo model, when trained on the NeoPolyp dataset (which contains undefined polyps) and the NeoPolyp-Clean dataset (which does not contain undefined polyps). Results show that the addition of these undefined pixels leads to improvements across the board, especially for the non-neoplastic class.

TABLE 2. Performance metrics on the NeoPolyp test set for BlazeNeo with different feature aggregation schemes.

Method	Dice$_{seg}$	IoU$_{seg}$	Dice$_{non}$	IoU$_{non}$	Dice$_{neo}$	IoU$_{neo}$	FPS
BlazeNeo-LSC	0.897	0.814	0.709	0.550	0.884	0.792	**88.6**
BlazeNeo-IDA	0.890	0.803	0.715	0.557	0.887	0.798	84.6
BlazeNeo-DIA	0.897	0.814	0.719	0.562	**0.888**	**0.800**	84.0
BlazeNeo-DHA	**0.901**	**0.820**	**0.728**	**0.572**	**0.888**	**0.800**	81.5

TABLE 3. Performance metrics for BlazeNeo-DHA when training on NeoPolyp and NeoPolyp-Clean, measured on the NeoPolyp test set.

Training dataset	Dice$_{seg}$	IoU$_{seg}$	Dice$_{non}$	IoU$_{non}$	Dice$_{neo}$	IoU$_{neo}$
NeoPolyp-Clean	0.900	0.818	0.714	0.555	0.884	0.792
NeoPolyp	**0.901**	**0.820**	**0.728**	**0.572**	**0.888**	**0.799**

B. COMPARISON WITH STATE-OF-THE-ART MODELS

1) QUANTITATIVE COMPARISON

We compare the performance of BlazeNeo-DHA, the best-performing BlazeNeo model, with seven state-of-the-art models for the polyp segmentation and PSND problem: U-Net [32], ColonSegNet [16], DDANet [40], DoubleUNet [17], HarDNet-MSEG [12], PraNet [9], and NeoUNet [20]. We keep all default training settings of these models as reported by the authors. Except for NeoUNet, the models mentioned above do not handle undefined neoplasm pixels during training. Thus, in the interest of a fair comparison, we use the NeoPolyp-Clean dataset for this experiment, which does not contain undefined polyps. Results are shown in Table 4.

We can see that while NeoUNet remains the most accurate model, BlazeNeo is a close second, with a difference of less than 1% on most accuracy metrics. At the same time, BlazeNeo is a much more lightweight and faster model. Compared to NeoUNet, BlazeNeo achieves higher FPS (81.5 versus 68.3), has half as many parameters (17, 143, 324 versus 38, 288, 397), and lower GFLOPs (11.06 versus 39.88). Notably, BlazeNeo-DHA is faster and more accurate than HarDNet-MSEG, while NeoUNet is slower than HarDNet-MSEG and U-Net. We attribute the speed improvement of BlazeNeo over HarDNet-MSEG to the different decoder designs, especially the use of the small RFB module.

Although achieving a high overall dice score, our BlazeNeo is not without limitations. The dice and IoU scores of BlazeNeo are still low for the non-neoplastic class, and both of them are degraded when the model is converted into the INT8 precision. The reason may be due to the small size of non-neoplastic polyps. When the weights are converted into integers ranging from -128 to 127 in the INT8 precision, the model seems to lose the capacity of representing detailed information and, therefore, is easier to miss small objects in the input images.

2) QUALITATIVE COMPARISON

Figure 10 shows output examples of BlazeNeo-DHA and other baseline models. Overall, BlazeNeo-DHA produces the most accurate segmentation and classification results for different types of polyps.

The first five rows of Figure 10 contain "easier" polyp examples. BlazeNeo-DHA and NeoUNet perform quite well in these examples, with similarly high accuracy. Meanwhile, PraNet, HarDNet-MSEG, DoubleU-Net, DDANet, U-Net, and ColonSegNet produce predictions with less uniformity, with some polyps containing both neoplastic and non-neoplastic regions. For larger polyps such as the 5th row, PraNet cannot fully segment the area.

The last five rows in Figure 10 represent more challenging examples, in which all models struggle to provide accurate segmentation masks. This is because non-neoplastic polyps are usually small in size and easier to be miss-detected. BlazeNeo-DHA and NeoUNet produce fewer false positives in these situations, while U-Net and ColonSegNet create the most inaccurate masks.

The last two rows show two non-neoplastic polyps in BLI and FICE modes. Contributing factors for these struggles in enhanced color modes include their smaller proportions in the training dataset. Furthermore, the endoscopist put the camera scope close to the mucosa in these cases, making the polyps' surface look very clear and sharp such that one can even observe small dots that are glandular holes on the surface. Therefore, these polyps are easily confused with angiogenesis in neoplastic lesions, which is why all models failed in classifying them.

C. BENCHMARK FOR EMBEDDED DEVICE

In this experiment, we apply model compression techniques via the NVIDIA TensorRT 7.1 toolkit [28] to BlazeNeo and the baseline models. The compressed models are then benchmarked on the NVIDIA Jetson AGX Xavier developer kit, an embedded computation unit with an NVIDIA GPU specialized for edge AI deployments. This setup is more in-line with deployment scenarios for polyp segmentation and PSND models, i.e., embedded on-site into colonoscopy devices.

Three available precision modes are tested for compressing each model: FP32, FP16, and INT8. Each mode can be seen as a different trade-off level between accuracy and speed. FP32 precision applies techniques such as layer/tensor fusion while

TABLE 4. Performance metrics of different models on the NeoPolyp-Clean test set.

Method	Dice_{seg}	IoU_{seg}	Dice_{non}	IoU_{non}	Dice_{neo}	IoU_{neo}	FPS	Parameters	GFLOPs
ColonSegNet [16]	0.738	0.585	0.505	0.338	0.732	0.577	44.9	5,010,000	64.84
U-Net [32]	0.785	0.646	0.525	0.356	0.773	0.631	69.6	31,043,651	103.59
DDANet [40]	0.813	0.684	0.578	0.406	0.802	0.670	46.2	6,840,000	31.45
DoubleU-Net [17]	0.837	0.720	0.621	0.450	0.832	0.712	43.2	18,836,804	83.62
HarDNet-MSEG [12]	0.883	0.791	0.659	0.492	0.869	0.769	77.1	17,424,031	11.38
PraNet [9]	0.895	0.811	0.705	0.544	0.873	0.775	55.6	30,501,341	13.11
NeoUNet [20]	**0.911**	**0.837**	**0.720**	**0.563**	**0.889**	**0.800**	68.3	38,288,397	39.88
BlazeNeo-DHA (Ours)	0.904	0.825	0.717	0.559	0.885	0.792	**81.5**	**17,143,324**	**11.06**

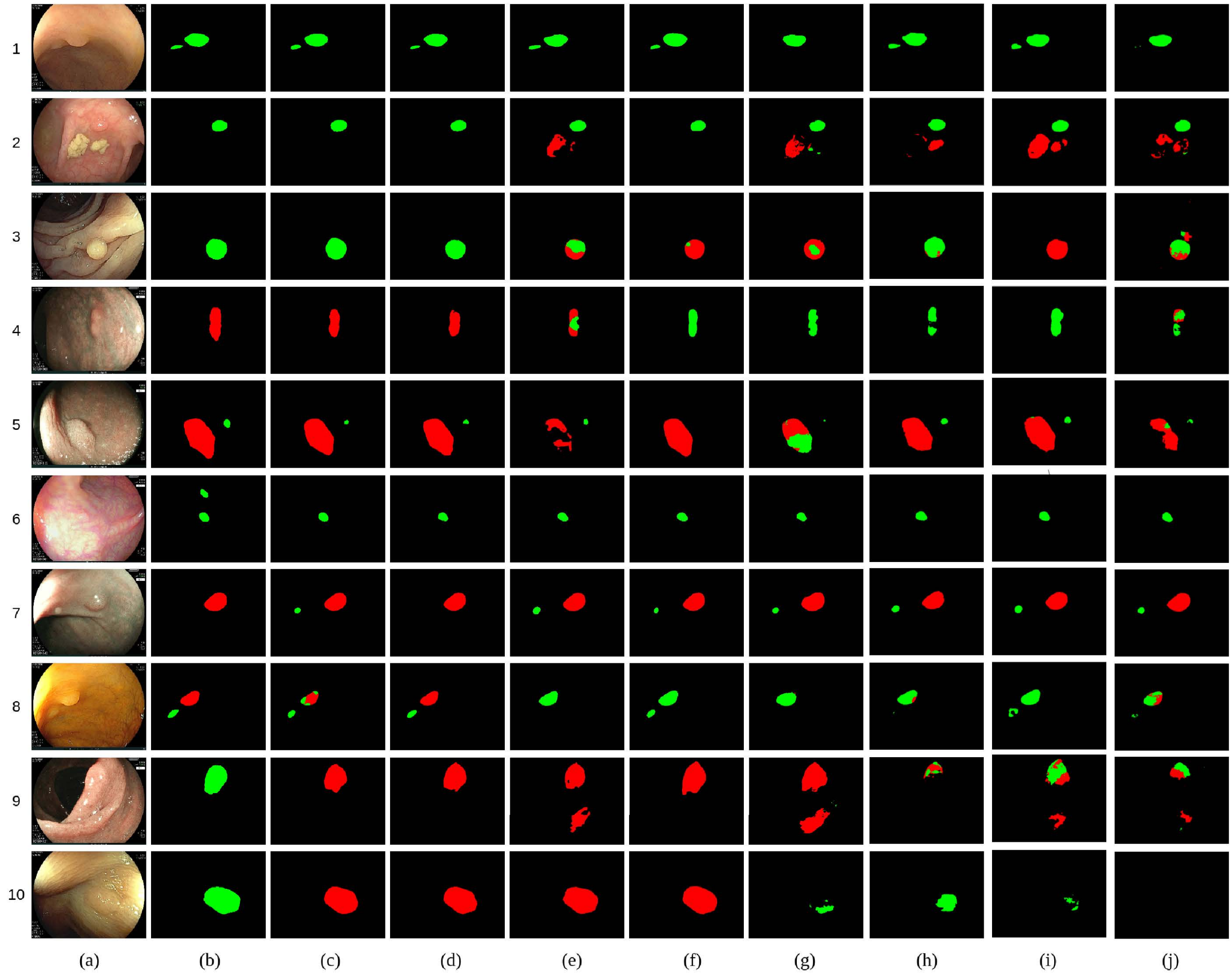


FIGURE 10. Qualitative comparison of the proposed method with other baseline methods: (a) image, (b) ground truth, (c) BlazeNeo (Ours), (d) NeoUNet, (e) PraNet, (f) HarDNet-MSEG, (g) UNet, (h) DoubleU-Net, (i) DDANet, and (j) ColonSegNet.

keeping parameters as 32-bit floating-point numbers. Hence, this mode provides some speed-up while minimizing accuracy degradation. FP16 precision converts suitable parameters to 16-bit floating-point numbers, greatly reducing model size and latency but is subject to more degradation. Finally, INT8 precision mode quantizes model parameters to 8-bit integers. This precision mode requires an additional calibration procedure to maintain model integrity, which attempts to replicate the original model's output on a small calibration dataset. Despite calibration, INT8 precision is susceptible to a lot more degradation. For this experiment, INT8 calibration is done on a randomized set of images.

Table 5 shows performance metrics in different precisions for U-Net, PraNet, HarDNet-MSEG, NeoUNet, and BlazeNeo-DHA. We can see a stark difference in FPS when running models on the Jetson AGX Xavier compared to the

TABLE 5. Performance metrics of state-of-the-art models in FP16, FP32 and INT8 precision levels on NVIDIA Jetson AGX Xavier.

Model	Precision	Dice_{seg}	IoU_{seg}	Dice_{non}	IoU_{non}	Dice_{neo}	IoU_{neo}	FPS
ColonSegNet@FP32	FP32	0.745	0.594	0.528	0.359	0.728	0.572	9.4
UNet@FP32	FP32	0.800	0.667	0.537	0.367	0.781	0.641	10.7
DDANet@FP32	FP32	0.799	0.665	0.557	0.386	0.776	0.635	16.6
DoubleU-Net@FP32	FP32	0.840	0.725	0.627	0.456	0.835	0.717	10.6
HarDNet-MSEG@FP32	FP32	0.891	0.804	0.685	0.521	0.871	0.771	52.2
PraNet@FP32	FP32	0.849	0.738	0.571	0.400	0.844	0.730	37.0
NeoUNet@FP32	FP32	**0.909**	**0.832**	**0.725**	**0.568**	**0.893**	**0.806**	25.4
BlazeNeo@FP32 (Ours)	FP32	0.906	0.828	0.721	0.563	0.887	0.796	**53.3**
ColonSegNet@FP16	FP16	0.746	0.594	0.528	0.359	0.728	0.572	26.3
UNet@FP16	FP16	0.800	0.666	0.537	0.367	0.781	0.641	35.5
DDANet@FP16	FP16	0.799	0.665	0.557	0.386	0.776	0.635	31.8
DoubleU-Net@FP16	FP16	0.840	0.725	0.627	0.456	0.835	0.717	22.9
HarDNet-MSEG@FP16	FP16	0.891	0.804	0.685	0.521	0.871	0.771	118.7
PraNet@FP16	FP16	0.850	0.740	0.572	0.401	0.845	0.731	95.1
NeoUNet@FP16	FP16	**0.908**	**0.832**	**0.724**	**0.568**	**0.893**	**0.806**	73.4
BlazeNeo@FP16 (Ours)	FP16	0.906	0.828	0.721	0.563	0.887	0.796	**121.9**
ColonSegNet@INT8	INT8	0.672	0.507	0.456	0.295	0.633	0.463	41.6
UNet@INT8	INT8	0.764	0.618	0.501	0.334	0.753	0.604	44.8
DDANet@INT8	INT8	0.770	0.626	0.492	0.326	0.748	0.598	46.4
DoubleU-Net@INT8	INT8	0.835	0.717	0.562	0.391	0.830	0.709	46.7
HarDNet-MSEG@INT8	INT8	0.810	0.680	0.594	0.422	0.799	0.665	143.7
PraNet@INT8	INT8	0.817	0.691	0.575	0.404	0.815	0.688	116.7
NeoUNet@INT8	INT8	0.848	0.736	0.638	0.468	0.848	0.737	100.3
BlazeNeo@INT8 (Ours)	INT8	**0.870**	**0.770**	**0.678**	**0.513**	**0.857**	**0.750**	**155.6**

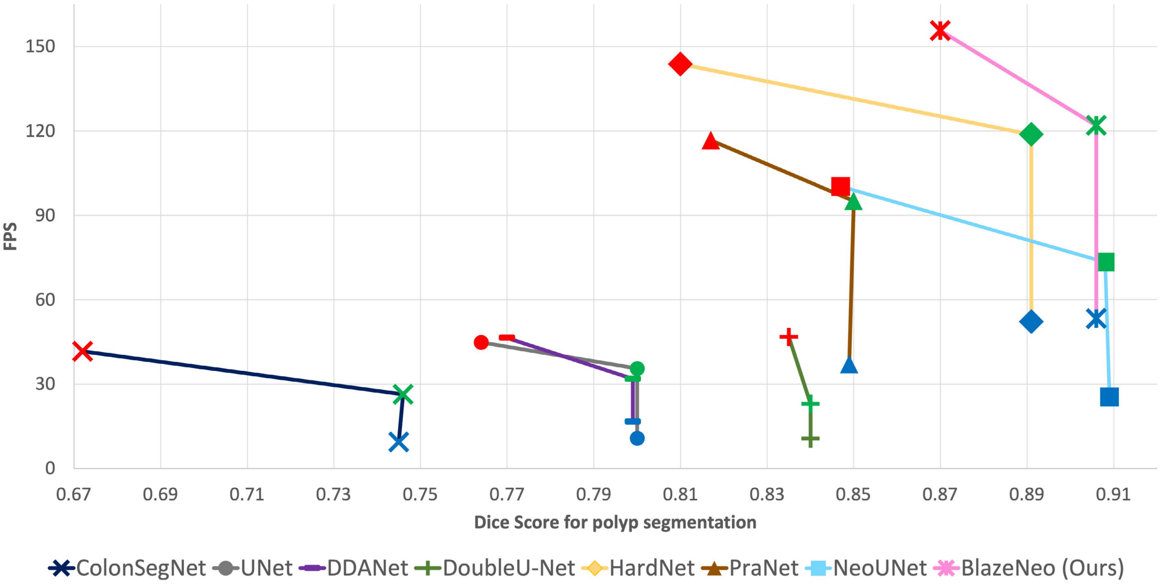


FIGURE 11. **Comparison of models' performance in different precisions. Red, green and blue markers denote INT8, FP16 and FP32 precisions, respectively.**

Google Colab environment. At FP32 precision, the fastest model (BlazeNeo) achieves only 53.3 FPS on the device, while the slowest model in the Colab environment (PraNet) without any compression still achieves 55.6 FPS. On the other hand, Dice and IoU measures are hardly affected at this precision level. In fact, U-Net and HarDNet-MSEG see improvements up to 1.5% across all metrics after compression. DoubleU-Net and ColonSegNet achieve a slight improvement of about $0.5 - 1\%$ after compression. DDANet suffers a little from compression, dropping by about 1.5% in segmentation metrics. Finally, PraNet is most affected after compression, dropping by 4.6% in segmentation metrics.

At FP16 precision, latency for all models are vastly improved, the fastest being BlazeNeo (121.9 FPS) and HarDNet-MSEG (118.7 FPS). In addition, accuracy metrics for all models are within 0.01% of their FP32 counterparts. This further shows that large neural networks do not require high float precision to remain effective.

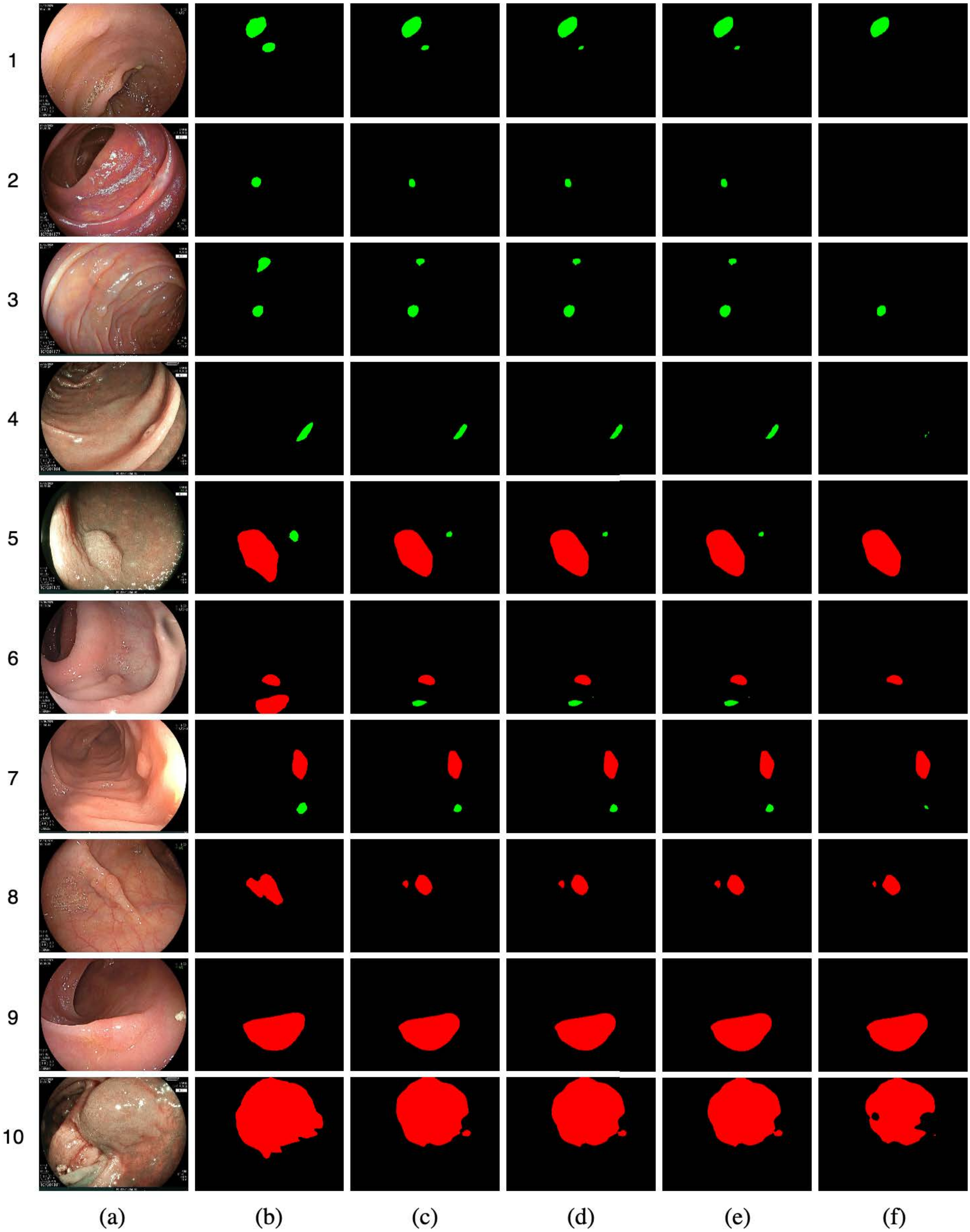


FIGURE 12. Qualitative comparison of BlazeNeo-DHA models in different precisions: (a) image, (b) ground truth, (c) Pytorch, (d) TensorRT FP32, (e) TensorRT FP16, (f) TensorRT INT8.

TABLE 6. Accuracy metrics on the NeoPolyp-Clean test set for BlazeNeo-DHA models in different precisions.

Model	Dice_{seg}	IoU_{seg}	Dice_{non}	IoU_{non}	Dice_{neo}	IoU_{neo}
PyTorch (w/o compression)	0.904	0.825	0.717	0.559	0.885	0.792
TensorRT FP32	0.906	0.828	0.721	0.563	0.887	0.796
TensorRT FP16	0.906	0.828	0.721	0.563	0.887	0.796
TensorRT INT8	0.870	0.770	0.678	0.513	0.857	0.750

TABLE 7. Latency metrics for BlazeNeo-DHA models in different precisions. The latency is measured on Jetson Xavier AGX with power mode MAXN.

Precision	Host Latency (ms)				GPU Compute (ms)			
	min	max	mean	median	min	max	mean	median
TensorRT FP32	18.50	22.04	18.67	18.55	18.40	21.94	18.57	18.45
TensorRT FP16	8.10	10.24	8.20	8.15	7.99	10.13	8.09	8.04
TensorRT INT8	**6.35**	**6.51**	**6.42**	**6.42**	**6.25**	**6.42**	**6.32**	**6.32**

INT8 precision gives the most speed gain out of the three modes, but at great expense in terms of accuracy. BlazeNeo runs at 155.6 FPS in this mode, whereas HarDNet-MSEG is the second fastest model at 143.7 FPS. However, HarDNet-MSEG also suffers the largest drop in accuracy at about 8.1% on all metrics. This drop is a bit lower for PraNet ($\approx 3.3\%$), NeoUNet ($\approx 6\%$) and BlazeNeo ($\approx 3.6\%$).

In every precision mode, BlazeNeo is consistently the fastest model while being a close second in terms of accuracy behind NeoUNet. BlazeNeo also displays its robustness to compression techniques, incurring significantly less degradation compared to models such as PraNet. Its small size also gives the proposed model advantage in long-term deployment, as energy usage and equipment wear become factors. In Figure 11, we visualize the performance of BlazeNeo in terms of speed (FPS) and accuracy (dice score on polyp segmentation task) compared to other existing methods.

It should be emphasized that high FPS is essential in real scenarios because it helps endoscopists operate smoother and detect lesions easier. According to endoscopists' experience, the ideal speed for colonoscopy should be at least 60 FPS. In addition, the fast inference speed gives us the potential to deploy the model on even more low-cost devices with less computational power, such as NVIDIA Jetson TX1/TX2, while still satisfying the minimal required FPS. This is important because it allows us to deploy the application on a large scale to many medical facilities, especially the poorly equipped ones in developing countries.

For further discussion, we compare the performance of the models quantitatively and qualitatively in different precisions.

Table 6 shows the accuracy of our BlazeNeo-DHA models in different precisions. One can observe that the models with FP32 and FP16 precisions give the same result, which is lower than the original model by a mean margin of 2.32%. The TensorRT INT8 model yields the worst accuracy, which is lower than the original model by a mean margin of 5.75%.

With lower precision, a model can reduce latency, throughput and increase power efficiency. To compare the speed performance of the models in different precisions, we present detailed benchmarking results of them for 100 iterations on the Jetson AGX Xavier with two metrics: host latency and GPU compute time. Host latency is measured as the end-to-end execution time from the CPU point of view, while GPU computing time is the actual working time for GPU calculation.

As shown in Table 7, the model in INT8 precision has the shortest latency compared to all other models, less than one-third of the model in FP32 precision. However, we must consider the trade-off between accuracy and speed for deep learning inference. Therefore, the model in FP16 precision is the best choice, which gives the best result in all precisions and has a median latency.

For a more intuitive understanding of the loss in accuracy of each compressed model, Figure 12 illustrates the sample results of BlazeNeo-DHA models in different precisions. As we can observe, the prediction result of models in FP32 and FP16 precisions are almost the same. In general, the INT8 compressed model loses the ability to segment some small polyps compared to other modes. For segmented polyps, the neoplasm prediction remains constant for every compression mode in all examples. This demonstrates that BlazeNeo can be quite robust to model compression and can be deployed in high compression modes such as FP16 with confidence.

VI. CONCLUSION

This paper has proposed BlazeNeo, a novel neural network architecture for the polyp segmentation and neoplasm detection problem, with an emphasis on speed and deployability. BlazeNeo is an extremely lightweight and fast neural network, thanks to the use of an efficient HarDNet backbone, a multi-level feature aggregation structure, and an auxiliary training module to take advantage of undefined labels. Our experiments show that BlazeNeo outperforms all other state-of-the-art models in terms of inference latency while providing competitive accuracy. We also show that BlazeNeo can be robust against degradation caused by compression techniques. In general, the proposed model is highly suitable for lightweight deployments with real-time requirements. Our source code is available at https://github.com/tofuai/blazeneo.

In future works, we will investigate recent advancements in Transformer-based architectures to improve the performance of the models. Especially, we will focus on lightweight architectures such as SegFormer [42] since they ensure both high accuracy and fast inference speed which are crucial factors for a real computer-aided system in colonoscopy.

REFERENCES

[1] N. Abraham and N. M. Khan, "A novel focal Tversky loss function with improved attention U-Net for lesion segmentation," in *Proc. IEEE 16th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2019, pp. 683–687.

[2] M. A. Armin, V. H. De, G. Chetty, C. Dumas, D. Conlan, F. Grimpen, and O. Salvado, "Visibility map: A new method in evaluation quality of optical colonoscopy," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 396–404.

[3] J. Bernal *et al.*, "Comparative validation of polyp detection methods in video colonoscopy: Results from the MICCAI 2015 endoscopic vision challenge," *IEEE Trans. Med. Imag.*, vol. 36, no. 6, pp. 1231–1249, Jun. 2017, doi: 10.1109/TMI.2017.2664042.

[4] P. Chao, C.-Y. Kao, Y. Ruan, C.-H. Huang, and Y.-L. Lin, "HarDNet: A low memory traffic network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3552–3561.

[5] 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.

[6] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*.

[7] S. Chennupati, G. Sistu, S. Yogamani, and S. Rawashdeh, "AuxNet: Auxiliary tasks enhanced semantic segmentation for automated driving," 2019, *arXiv:1901.05808*.

[8] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.

[9] D. P. Fan, G. P. Ji, T. Zhou, G. Chen, H. Fu, J. Shen, and L. Shao, "PraNet: Parallel reverse attention network for polyp segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2020, pp. 263–273.

[10] M. Gschwantler, S. Kriwanek, E. Langner, B. Göritzer, C. Schrutka-Kölbl, E. Brownstone, H. Feichtinger, and W. Weiss, "High-grade dysplasia and invasive carcinoma in colorectal adenomas: A multivariate analysis of the impact of adenoma and patient characteristics," in *Proc. Eur. J. Gastroenterol. Hepatol.*, vol. 14, no. 2, pp. 183–188, 2002.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[12] C.-H. Huang, H.-Y. Wu, and Y.-L. Lin, "HarDNet-MSEG: A simple encoder-decoder polyp segmentation neural network that achieves over 0.9 mean dice and 86 FPS," 2021, *arXiv:2101.07172*.

[13] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[14] I. A. Issa and M. Noureddine, "Colorectal cancer screening: An updated review of the available options," *World J. Gastroenterol.*, vol. 23, no. 28, p. 5086, 2017.

[15] Y. Iwahori, T. Shinohara, A. Hattori, R. J. Woodham, S. Fukui, M. K. Bhuyan, and K. Kasugai, "Automatic polyp detection in endoscope images using a Hessian filter," in *Proc. MVA*, 2013, pp. 21–24.

[16] D. Jha, S. Ali, N. K. Tomar, H. D. Johansen, D. Johansen, J. Rittscher, M. A. Riegler, and P. Halvorsen, "Real-time polyp detection, localization and segmentation in colonoscopy using deep learning," *IEEE Access*, vol. 9, pp. 40496–40510, 2021.

[17] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, "DoubleU-Net: A deep convolutional neural network for medical image segmentation," in *Proc. 33rd IEEE Int. Symp. Comput.-Based Med. Syst. (CBMS)*, Rochester, MN, USA, A. G. S. de Herrera, A. R. González, K. C. Santosh, Z. Temesgen, B. Kane, and P. Soda, Eds., Jul. 2020, pp. 558–564, doi: 10.1109/CBMS49503.2020.00111.

[18] N. H. Kim, Y. S. Jung, W. S. Jeong, H.-J. Yang, S.-K. Park, K. Choi, and D. I. Park, "Miss rate of colorectal neoplastic polyps and risk factors for missed polyps in consecutive colonoscopies," *Intestinal Res.*, vol. 15, no. 3, p. 411, 2017.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[20] P. N. Lan, N. S. An, D. V. Hang, D. V. Long, T. Q. Trung, N. T. Thuy, and D. V. Sang, "NeoUNet: Towards accurate colon polyp segmentation and neoplasm detection," in *Proc. 16th Int. Symp. Adv. Vis. Comput. (ISVC)*, in Lecture Notes in Computer Science, vol. 13018. Cham, Switzerland: Springer, Oct. 2021, pp. 15–28.

[21] S.-H. Lee, I.-K. Chung, S.-J. Kim, J.-O. Kim, B.-M. Ko, Y. Hwangbo, W. H. Kim, D. H. Park, S. K. Lee, C. H. Park, I.-H. Baek, D. I. Park, S.-J. Park, J.-S. Ji, B.-I. Jang, Y.-T. Jeen, J. E. Shin, J.-S. Byeon, C.-S. Eun, and D. S. Han, "An adequate level of training for technical competence in screening and diagnostic colonoscopy: A prospective multicenter evaluation of the learning curve," *Gastrointestinal Endoscopy*, vol. 67, no. 4, pp. 683–689, Apr. 2008.

[22] A. Leufkens, M. van Oijen, F. Vleggaar, and P. Siersema, "Factors influencing the miss rate of polyps in a back-to-back colonoscopy study," *Endoscopy*, vol. 44, no. 5, pp. 470–475, May 2012.

[23] K. Li, M. I. Fathan, K. Patel, T. Zhang, C. Zhong, A. Bansal, A. Rastogi, J. S. Wang, and G. Wang, "Colonoscopy polyp detection and classification: Dataset creation and comparative evaluations," *PLoS ONE*, vol. 16, no. 8, Aug. 2021, Art. no. e0255809.

[24] M. Liu, J. Jiang, and Z. Wang, "Colonic polyp detection in endoscopic videos with single shot detection based deep convolutional neural network," *IEEE Access*, vol. 7, pp. 75058–75066, 2019.

[25] S. Liu and D. Huang, "Receptive field block net for accurate and fast object detection," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 385–400.

[26] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[27] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[28] (2018). *NVIDIA: NVIDIA Tensorrt*. Accessed: Aug. 3, 2021. [Online]. Available: https://developer.nvidia.com/tensorrt

[29] O. Oktay, J. Schlemper, L. L. Folgoc, M. C. H. Lee, M. P. Heinrich, K. Misawa, K. Mori, S. G. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, and D. Rueckert, "Attention U-Net: Learning where to look for the pancreas," 2018, *arXiv:1804.03999*.

[30] 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. 169537–169547, 2019.

[31] E. Ribeiro, A. Uhl, G. Wimmer, and M. Häfner, "Exploring deep learning and transfer learning for colonic polyp classification," *Comput. Math. Methods Med.*, vol. 2016, pp. 1–16, Oct. 2016.

[32] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[33] Y. Shin, H. A. Qadir, L. Aabakken, J. Bergsland, and I. Balasingham, "Automatic colon polyp detection using region based deep CNN and post learning approaches," *IEEE Access*, vol. 6, pp. 40950–40962, 2018.

[34] Y. Shin, H. A. Qadir, and I. Balasingham, "Abnormal colon polyp image synthesis using conditional adversarial networks for improved detection performance," *IEEE Access*, vol. 6, pp. 56007–56017, 2018.

[35] J. Silva, A. Histace, O. Romain, X. Dray, and B. Granado, "Toward embedded detection of polyps in WCE images for early diagnosis of colorectal cancer," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 9, no. 2, pp. 283–293, Mar. 2014.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, Y. Bengio and Y. LeCun, Eds., May 2015. [Online]. Available: https://dblp.org/db/conf/iclr/iclr2015.html

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[38] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[39] Z. Tang, X. Peng, S. Geng, Y. Zhu, and D. N. Metaxas, "CU-Net: Coupled U-Nets," in *Proc. 29th Brit. Mach. Vis. Conf.*, 2019, p. 305.

[40] N. K. Tomar, D. Jha, S. Ali, H. D. Johansen, D. Johansen, M. A. Riegler, and P. Halvorsen, "DDANet: Dual decoder attention network for automatic polyp segmentation," in *Pattern Recognition. ICPR International Workshops and Challenges* (Lecture Notes in Computer Science), vol. 12668. Cham, Switzerland: Springer, Jan. 2021, pp. 307–314.

[41] Z. Wu, L. Su, and Q. Huang, "Cascaded partial decoder for fast and accurate salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3907–3916.

[42] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and efficient design for semantic segmentation with transformers," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 12077–12090.

[43] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.

[44] L. Zhang, M. Yu, T. Chen, Z. Shi, C. Bao, and K. Ma, "Auxiliary training: Towards accurate and robust models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 372–381.

[45] Y. Zhang, "Multi-scale object detection model with anchor free approach and center of gravity prediction," in *Proc. IEEE 5th Inf. Technol. Mechatronics Eng. Conf. (ITOEC)*, Jun. 2020, pp. 38–45.

[46] Y. Zhang, H. Liu, and Q. Hu, "TransFuse: Fusing transformers and CNNs for medical image segmentation," 2021, *arXiv:2102.08005*.

[47] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.

NGUYEN S. AN received the graduate degree (Global Engineering Program) in information and communication technology from the Hanoi University of Science and Technology (HUST). In 2018, he received the ERAMUS+ Scholarship at the Tampere University of Technology (TUT), Finland. He was selected as one of the two Vietnamese students for participating in the Asia-Oceania Top University League (AOTULE) Summer Program 2019 at the Bandung Institute of Technology.

PHAN N. LAN received the degree (Eng.) in information and communication technology from HUST, in 2019, where he is currently pursuing the graduate degree in data science and artificial intelligence. He is a Student Member of the International Research Center for Artificial Intelligence (BK.AI). His research interests include machine learning, computer vision, and optimization techniques.

DAO V. HANG is a Gastroenterologist and a Clinical Lecturer with Hanoi Medical University and the Institute of Gastroenterology and Hepatology. She also assumes different roles in the Vietnam Association of Gastroenterology (VNAGE), including organizing endoscopy training courses. Her current research interest includes the application of innovative technologies, such as image-enhanced endoscopy, smart apps, and artificial intelligence (AI) in endoscopy, microbiome, and GI motility. She believes technological solutions, among which AI is promising and feasible, can help solve the problems in limited-resources settings. She has experience participating in collaborative AI projects, where she led her team to recruit, label data, validate the product, and implement clinical studies.

DAO V. LONG is a Professor in gastroenterology and hepatology and is currently the Vice President of VNAGE. He was the Vice Rector and the Director of Hanoi Medical University Hospital and the Head of the Gastroenterology Department, Bach Mai Hospital. He has good collaborations with foreign specialists and has organized many workshops and training courses for Vietnamese GI doctors and endoscopists. His research interests include interventional endoscopy and GI tract diseases. During clinical practice, he has realized the importance of technology, particularly artificial intelligence in GI endoscopy and has become interested in researching its application to facilitate diagnostic accuracy and reduce human labor in the practice of clinical physicians and endoscopists.

TRAN Q. TRUNG is currently pursuing the MD.PhD program at Greifswald University of Medicine, Germany, and a lecturer at the Department of Internal Medicine, University of Medicine and Pharmacy, Hue University, Vietnam. He is one of the young scientists who was selected globally to meet 70 Nobel Laureates at the 70th Lindau Nobel Meeting, where he can join a multidisciplinary network. He has been passionate about GI endoscopy, since 2010, when he became a Medical Doctor. With experiences gained from advanced training in GI endoscopy in Vietnam, Japan, and Germany, and from a variety of international conferences in endoscopy, he is doing some novel and fruitful works for patients in Vietnam. His research interest includes applying AI to the field and ultimately bringing benefit to patients.

NGUYEN T. THUY received the Ph.D. degree in computer science from the Graz University of Technology, Austria, in 2009. She is currently an Associate Professor and the Head of the Department of Computer Science, Faculty of Information Technology, Vietnam National University of Agriculture (VNUA), Vietnam. She has more than ten years of research experience in her research areas. She is an author and coauthor of more than 70 research papers and patents. She is a principal investigator and key member of a number of research projects in computer vision, machine learning, and applications. Her research interests include computer vision, machine learning, and pattern recognition.

DINH V. SANG received the Ph.D. degree in computer science from the Dorodnitsyn Computing Centre of the Russian Academy of Sciences (CCRAS), in 2013. He is working with the Faculty of Computer Science, School of Information and Communication Technology (SoICT), Hanoi University of Science and Technology (HUST), Vietnam, where he is currently the Deputy Managing Director of the International Research Center for Artificial Intelligence (BK.AI). He has more than ten years of research experience in computer vision and machine learning and has published about 50 publications. His research interests include computer vision, machine learning, and deep learning. He is the first NVIDIA Deep Learning Institute (DLI) Ambassador in Vietnam.