IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 28, 2021, accepted November 14, 2021, date of publication November 16, 2021,
date of current version November 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3128607

AEDCN-Net: Accurate and Efficient Deep Convolutional Neural Network Model for Medical Image Segmentation

BEKHZOD OLIMOV, SEOK-JOO KOH, AND JEONGHONG KIM
Computer Science and Engineering Department, Kyungpook National University, Daegu 41566, South Korea

Corresponding author: Jeonghong Kim (jhk@knu.ac.kr)

This work was supported in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education under Grant 2021R1I1A3043970; and in part by the BK21 FOUR Project (AI-driven Convergence Software Education Research Program) funded by the Ministry of Education, School of Computer Science and Engineering, Kyungpook National University, South Korea, under Grant 4199990214394.

ABSTRACT Image segmentation was significantly enhanced after the emergence of deep learning (DL) methods. In particular, deep convolutional neural networks (DCNNs) have assisted DL-based segmentation models to achieve state-of-the-art performance in fields critical to human beings, such as medicine. However, the existing state-of-the-art methods often use computationally expensive operations to achieve high accuracy and lightweight networks often lack a precise medical image segmentation. Therefore, this study proposes an accurate and efficient DCNN model (AEDCN-Net) based on an elaborate preprocessing step and a resourceful model architecture. The AEDCN-Net exploits bottleneck, atrous, and asymmetric convolution-based residual skip connections in the encoding path that reduce the number of trainable parameters and floating point operations (FLOPs) to learn feature representations with a larger receptive field. The decoding path employs the nearest-neighbor based upsampling method instead of a computationally resourceful transpose convolution operation that requires an extensive number of trainable parameters. The proposed method attains a superior performance in both computational time and accuracy compared to the existing state-of-the-art methods. The results of benchmarking using four real-life medical image datasets specifically illustrate that the AEDCN-Net has a faster convergence compared to the computationally expensive state-of-the-art models while using significantly fewer trainable parameters and FLOPs that result in a considerable speed-up during inference. Moreover, the proposed method obtains a better accuracy in several evaluation metrics compared with the existing lightweight and efficient methods.

INDEX TERMS Computational efficiency, deep convolutional neural networks, medical image segmentation.

I. INTRODUCTION

Image segmentation is a computer vision task that specializes in categorizing an input image or a video frame into a pre-defined number of classes by generating non-intersecting and easily-interpretable sections of the input beneficial for further processing. The image segmentation task is considerably complex compared to other computer vision tasks, such as image classification because image classification categorizes an input by processing the entire image [1], whereas image segmentation generates an output for every single image pixel.

Image segmentation has numerous real-life applications, including video surveillance [2], augmented reality [3], and driverless cars [4]. The most beneficial and noteworthy image segmentation application is in the field of medicine, where it provides a detailed illustration of the human body for the anatomy analysis, detects illnesses, and identifies the severity level of a disease, to name a few [5]. Medical image segmentation is directly associated with a person's health and life; hence, it must be very accurate to prevent a disease or cure an illness [6], [7], [9], [10].

Based on the input specifications, the image segmentation task can broadly be divided into two distinct groups: binary and multiclass image segmentation. The binary image has two available categories, namely background and foreground.

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

Some of the applications of the medical image segmentation belongs to this group [9], [10]. On the other hand, the multiclass segmentation may have more than two countable classes, including semantic segmentation in autonomous driving applications [11].

Considering the notable performance of deep learning (DL) methods', artificial intelligence (AI) systems have been shown to outperform humans in image classification tasks [13], [14]. While a person can compete with an AI system in the image classification task, it is impossible in image segmentation due to the significantly complex nature of the task. Because pixel-by-pixel classification is prohibitively tedious and not feasible given the enormous quantity of data in modern medical images. Therefore, generating precisely segmented medical images using AI techniques is becoming a research hotspot [16].

Due to the criticality of the DL methods for the medical image segmentation, extensive research has so far been made in this domain. The most popular DL model architecture in this field is U-Net [17]. After its introduction in 2015, researchers have proposed DL-based networks that achieve a state-of-the-art performance [9], [16], [18]–[22]. However, some of these models [16], [18]–[20], [22] perform complex computations, which make them unusable in machines with limited computation resources. In addition, these computationally expensive models require an extremely long training time for DL-based medical image segmentation models. Some efficient models [9], [21] cannot attain a state-of-the-art performance and cannot generate an accurate medical image segmentation. The aforementioned problems should be addressed to ensure further progress in medical image segmentation. Considering the existing shortcomings, we propose herein an accurate and efficient deep convolutional neural network (DCNN) model, called AEDCN-Net, to alleviate the current issues by reducing the number of trainable parameters and training/ inference time as well as improving the medical image segmentation accuracy. The contributions of this study are fourfold:

- The AEDCN-Net benefits from bottleneck, atrous, and asymmetric convolution-based skip connections in the encoding path and nearest-neighbor interpolation method in the decoding path, which significantly reduces the number of trainable model parameters.
- Due to the carefully designed architecture of AEDCN-Net, on average, it is 40% faster than the existing computationally expensive methods that achieve a state-of-the-art performance in medical image segmentation.
- Although AEDCN-Net demands fewer trainable parameters and less training time, it has a superior performance in terms of accuracy and generates more precise segmented medical images compared with its counterparts.
- To the best of our knowledge, no proposed model has yet outperformed the existing methods in both computational efficiency and segmentation accuracy so far. Therefore, the proposed model can be used as a benchmark for further studies in the medical image segmentation domain.

The rest of this paper is structured as follows: Section II reveals detailed information on the existing methods in medical image segmentation; Section III provides a meticulous explanation of the proposed methodology; Section IV presents the experimental details; Section V discusses results of the experiments and qualitative comparison of the considered models; and finally, Section VI concludes this study and presents future study directions.

II. RELATED WORK

This section summarizes the currently available methods used in medical image segmentation. Based on the techniques characteristics, they can broadly be categorized into computationally expensive and powerful, as well as lightweight and efficient models.

A. COMPUTATIONALLY EXPENSIVE AND POWERFUL MODELS FOR SEMANTIC SEGMENTATION

After the introduction of the convolutional neural network (CNN) models in computer vision tasks, considerable progress has been observed in the medical image segmentation accuracy. The most notable DL-based network is a fully CNN encoder-decoder architechture-based model for biomedical image segmentation, called U-Net [17]. The existing DL-based methods attaining a state-of-the-art performance in medical image segmentation have a similar model architecture to the U-Net [23]. They are precisely enhanced U-Net variants. For example, Zhou *et al.* proposed a novel encoder-decoder architecture that uses blocks of nested, dense skip connections [20]. These pathways reduce the semantic gap between the feature maps of the encoder and decoder sub-networks that assisted to significantly outperform the existing methods. Isensee *et al.* developed a robust and self-adapting framework on the basis of the original U-Net architecture [19]. The network benefits from the leaky rectified linear unit activation function and instance normalization to achieve a performance better than that of the original U-Net. Li *et al.* improved the U-Net architecture with residual connections by increasing the network depth and adding strong dropouts to extract finer features that allow state-of-the-art performance in fundus image segmentation [18]. Similarly, Jha *et al.* developed a ResUNet++ model architecture using a conditional random field and a test-time augmentation that achieved a superior performance compared with the existing DL-based networks on various polyp segmentation datasets [22]. Although these models exhibit a superior performance in terms of accuracy and precision in medical segmentation, they require an enormous number of trainable parameters; therefore, they are computationally expensive.

B. EFFICIENT AND LIGHTWEIGHT MODELS FOR MEDICAL IMAGE SEGMENTATION

To devise efficient DL-based models, Mehta *et al.* introduced a lightweight network that employs group point-wise

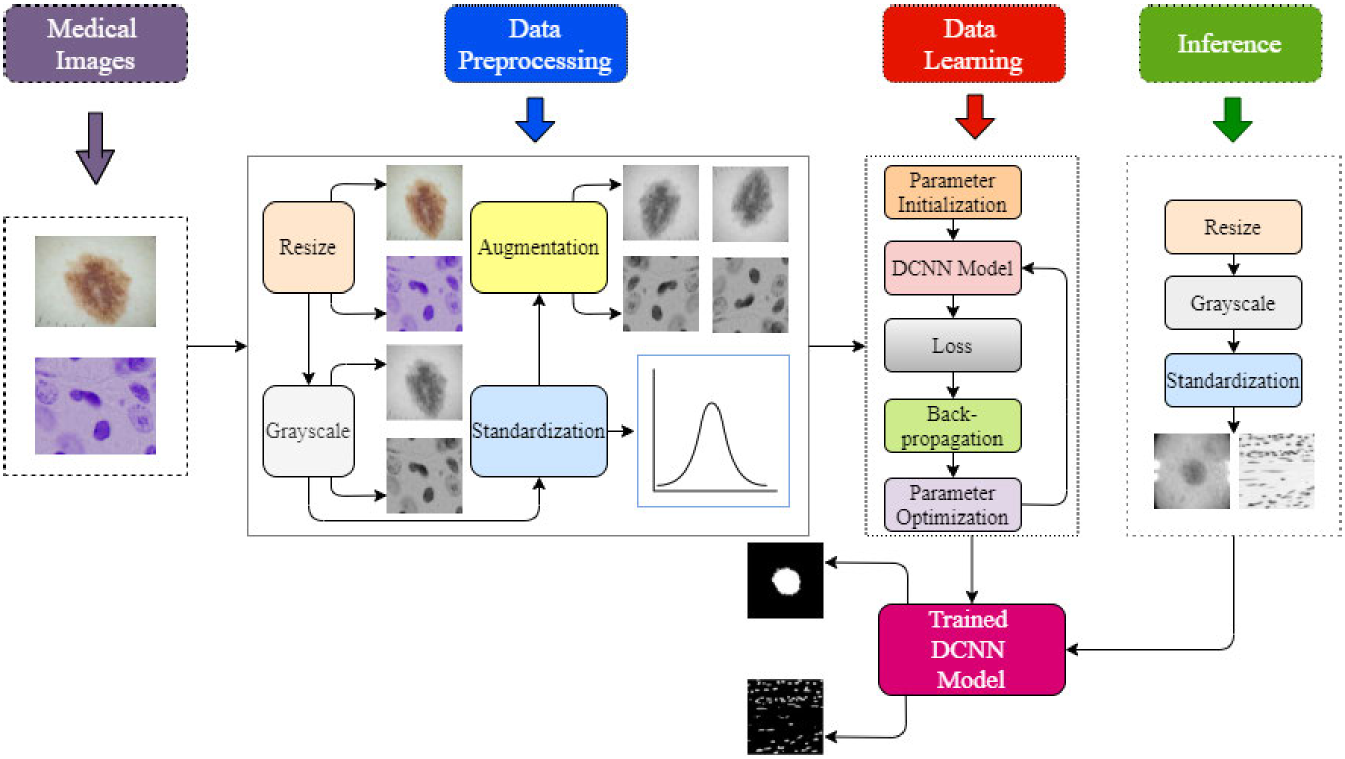

FIGURE 1. Graphical illustration of the proposed methodology.

and depth-wise dilated separable convolutions to achieve a state-of-the-art performance in semantic segmentation [21]. Similarly, [24] and [25] used compressing techniques, such as vector quantization to increase the speed of semantic segmentation models. Punn *et al.* also presented an inception U-Net architecture [26] inspired by [27]. This network illustrates the model perception of target segmentation images using activation maximization and filter map visualization techniques and attained a superior performance in terms of accuracy. Gadosey *et al.* developed a modified version of U-Net for devices with a low computational power based on bottleneck layers [28]. They used depth-wise separable convolutions in the entire network. In addition, the model benefited from a weight standardization algorithm with the group normalization method. The modifications allowed the model to be computationally efficient and lightweight. Similarly, Olimov *et al.* presented a fast U-Net (FU-Net) model relying on the bottleneck convolution layers in the encoding and decoding paths of the model, which allowed medical image segmentation on the devices with limited computational power and memory [9]. Although these models address the problem of efficient computation, they do not provide highly-accurate segmented images.

III. PROPOSED METHODOLOGY

This section presents AEDCN-Net in detail. Figure 1 shows an overview of the proposed methodology. AEDCN-Net has three distinct stages: data preprocessing, data learning, and inference.

A. DATA PREPROCESSING

In data preprocessing, raw medical images are prepared for training using the DCNN model. First, the images are resized to match the network input size. The images are resized to be 256 × 256. Moreover, the image ranges are preserved, and the outside boundary pixels are infilled with a constant value of 0 [9]. After obtaining same size images, their colors are transformed from three channels (i.e., red, green, and blue) to a single-channel grayscale mode. This process is useful in reducing the computational complexity of the DCNN model with almost no impact on its accuracy. Grayscale images are used for training; thus the number of trainable parameters in the first convolutional layer is reduced by thrice. After obtaining the grayscale images, we standardize the data by making them follow the standard normal distribution. For this purpose, we employ the following equation:

$$X_{std} = \frac{X - \frac{1}{M}\sum_{i=1}^{M} x_i}{\sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(x_i - \frac{1}{M}\sum_{i=1}^{M} x_i\right)^2}} \tag{1}$$

In (1), X and X_{std} are the original and standardized data, respectively, while i and M are the particular data point and the total number of instances, respectively.

Most medical image databases suffer from data scarcity problems [9]. To alleviate this issue, we applied data augmentation based on the characteristics of the medical image data

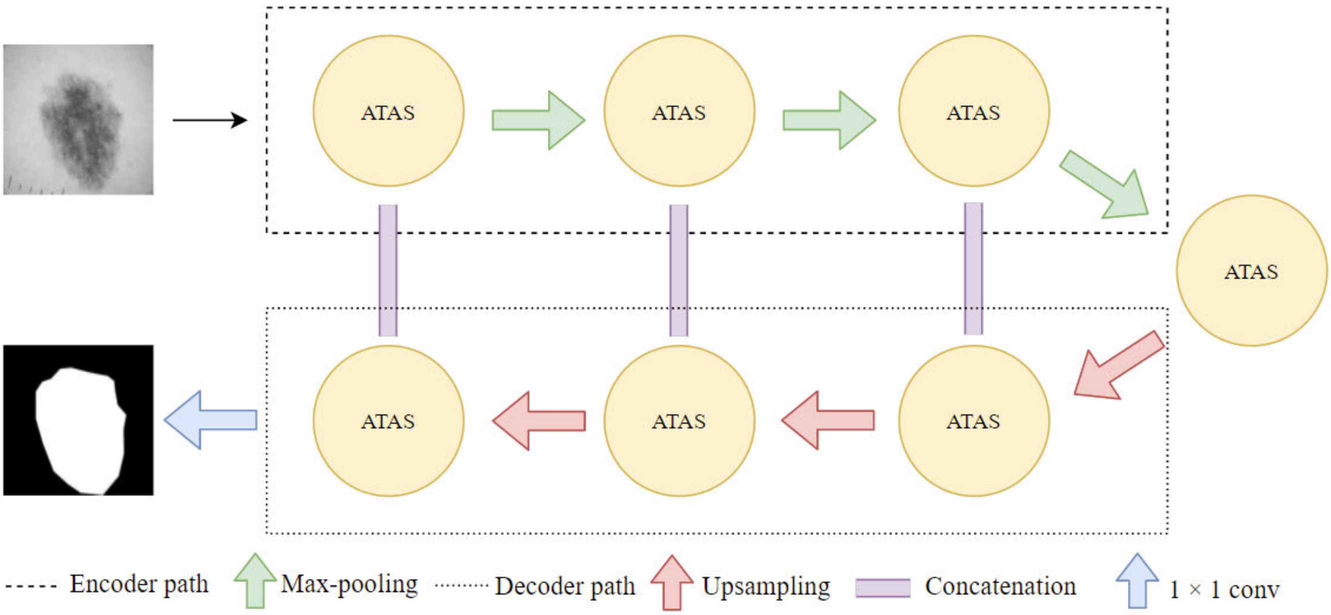

FIGURE 2. DCNN model architecture of the proposed method containing atrous-asymmetric convolution (ATAS) blocks for encoding and decoding paths.

after completing the data standardization process. The data augmentation techniques should be chosen carefully based on the dataset image characteristics; otherwise, they can result in a low performance of the DCNN model in the data learning stage. The data augmentation is a part of pre-processing stage and pre-computed before starting the data learning stage. The data augmentation is conducted only once before training stage and every epoch in the learning phase used the same augmented images. We used the following data augmentation techniques:

- Horizontally flipping the images;
- Randomly shifting the image dimensions in the range of integer value x;
- Zooming the images in the range of random integer value x;
- Randomly changing the angle of images by an integer value of y.

In the proposed method, we used x values ranging from -10% to 10%, and y values ranging from -5% to 5% because they resulted in the best performance of AEDCN-Net in the conducted experiments. Since we used four augmentation techniques, the proposed model uses four times more images per epoch in comparison to the original number of images in the datasets. Each epoch in the training process uses slightly different versions of original images in the dataset, which results in better generalizability of the model.

B. DATA LEARNING

After obtaining the preprocessed medical images from the first stage of the proposed methodology, we trained them using a DCNN model. Figure 2 shows the AEDCN-Net model architecture, which was similar to the original U-Net.

TABLE 1. Detailed description of the ATAS blocks: ks, a, p, and s stand for kernel size, atrous convolution factor, padding, and stride, respectively. Each block employs batch normalization (BN) and weight initialization-based rectified linear unit (WIB-ReLU) activation function [29].

Input image size: 256 × 256 × 1	
Main branch	*Secondary branch*
Conv (ks=1 × 1, a=1, p=1, s=1) BN WIB-ReLU Conv (ks=3 × 3, a=1, p=1, s=1) BN WIB-ReLU Conv(ks=3 × 1, a=1, p=1, s=1) BN WIB-ReLU Conv (ks=1 × 3, a=1, p=1, s=1) BN	Conv (ks=1 × 1, p=0, s=1) BN
Addition	
WIB-ReLU	

However, several modifications ensured the enhancement of the performance of the proposed model architecture. Specifically, it comprised atrous-asymmetric convolution (ATAS) blocks, max-pooling, concatenation, and upsampling operations. The ATAS blocks are responsible for learning useful features from the preprocessed medical images. Table 1 present details of the ATAS blocks.

Table 1 shows two branches in the ATAS blocks, namely the main and secondary branches. First, a raw medical image was input into the main branch by passing through bottleneck, atrous, and asymmetric convolution operations.

1) BOTTLENECK CONVOLUTION

The bottleneck convolutional layer is based on exploiting fewer convolution filters than the input image, each of which measures 1 × 1. This reduces the computational complexity due to the decrease in the input image channels. Specifically, given a medical input image I ($I \in \mathbb{R}^{H \times W \times C}$, where H, W, and C are the image height, width, and channels, respectively) and a convolutional filter F ($F \in \mathbb{R}^{T \times X \times Y \times C}$, where T, X, Y, and C are the total number of output filters, the filter height, filter width, and number of input filters, respectively), the number of required trainable weights and floating point operations (FLOPs) for a certain original and bottleneck convolution layer can be computed as follows:

$$\begin{aligned} W_{conv} &= c^{l-1} \times x \times y \times c^{l} \\ W_{bnck} &= c^{l-1} \times \frac{c^{l}}{b} + \frac{c^{l-1}}{b} \times x \times y \times c^{l} \\ FLOPs_{conv} &= H \times W \times c^{l-1} \times x \times y \times c^{l} \\ FLOPs_{bnck} &= H \times W \times \\ &\quad \times \left(c^{l-1} \times \frac{c^{l}}{b} + \frac{c^{l-1}}{b} \times x \times y \times c^{l} \right) \end{aligned} \tag{2}$$

In (2), l and b are the l^{th} convolutional layer of the network and the bottleneck convolution parameter, respectively. For the proposed model, we set b to 4 because it provided the best results in ablation studies (refer to Section V-C). The bottleneck convolution layer significantly reduces the number of trainable parameters and FLOPS and results in nearly two times of reduction in the aforementioned aspects.

2) ATROUS CONVOLUTION

The atrous convolution uses an atrous factor of a and is defined as follows:

$$(A *_a F)(p) = \sum_{c+a\,b=p} A(c) f(b) \tag{3}$$

In (3), A and f are the function and the convolution filter, respectively. The atrous convolution allows the increase of the receptive field of the convolution kernel without any additional memory space and computational power. Moreover, it ensures that the receptive field decoding does not negatively affect the image resolution and has no loss of its coverage. Considering these advantages of the atrous convolution, we exploited this technique in all convolution operations in AEDCN-Net to obtain a computationally and memory efficient model.

3) ASYMMETRIC CONVOLUTION

Equation (4) represents the regular convolution operation between an image I and a convolutional filter F_t.

$$\begin{aligned} \Psi_{conv} &= I * F_t \\ &= \sum_{x=1}^{X} \sum_{y=1}^{Y} \sum_{c=1}^{C} I\,(h-x, w-y, c)\, F_t\,(x, y, c) \end{aligned} \tag{4}$$

In (4), h, w, and c are in range of $1, 2, \ldots, H-1, H$, $1, 2, \ldots, W-1, W$, and $1, \ldots, C$, respectively. However, this convolution operation is significantly expensive because it requires $T \times X \times Y \times C^{l} \times H \times W \times C^{l-1}$ FLOPs. This study aims to develop an accurate and efficient network. Specifically, the expensive cost of the convolution operation can be alleviated by introducing an asymmetric convolution operation as follows:

$$\begin{aligned} \Psi_{th} &= I * F_{th} \\ &= \sum_{x=1}^{X} \sum_{c=1}^{C} I\,(h-x, y, c)\, F_{th}\,(x, 1, c) \\ \Psi_{tw} &= \Psi_{th} * F_{tw} \\ &= \sum_{y=1}^{Y} \sum_{tw=1}^{TW} \Psi_{tw}\,(x, w-y)\, F_{tw}\,(1, y, tw) \\ \hat{\Psi}_{tac} &= ((I * F_{th}) * F_{tw}) \end{aligned} \tag{5}$$

In (5), Ψ_{th} and Ψ_{tw} define the asymmetric convolution filters convolving with the height and the width of an input image, respectively, whereas $\hat{\Psi}_{tac}$ represents the output of the asymmetric convolution operation. With the usage of this convolution type, the trainable parameters were reduced to $(X \times C \times TH + Y \times TW \times T)$ and the FLOPs decreased to $(H \times W) \times (X \times C \times TH + Y \times TW \times T)$. Moreover, the asymmetric convolution conducted two convolution operations using various filters; thus, it could learn many non-linear functions and extract more useful features from the input images.

4) MODEL ARCHITECTURE

We progressively increased the number of filters in the encoding path. The first convolution layer contained 64 filters that have a size of 3 × 1, with atrous factor, padding, and a stride of 1. In every subsequent ATAS block, the number of convolution filters and the atrous convolution factor increased by 2 in the encoding and decreased by the same ratio in the decoding path. Each convolution operation was followed by batch normalization [30] and WIB-ReLU activation function [29]. Regarding the secondary branch, the input data passed through a regular convolution operation with a kernel size of 1 × 1, padding, and a stride of 1, followed by a batch normalization layer. The output of the considered branches were then added and passed through the WIB-ReLU activation function. Inspired by [13], we used the skip connections in the ATAS block to alleviate the vanishing gradient problem. These skip connections ensured that the information from the earlier layers is connected with the subsequent layers, allowing a more effective training of the DCNN model.

Moreover, the max-pooling operation decreased the spatial dimension of the images by a factor of two, ensuring a computational complexity reduction. The upsampling operation also recovered the image original size as the training progressed by increasing the output of the ATAS block in the decoding path by a factor of two. In the proposed model architecture, we used the nearest-neighbor interpolation method to recover the original image size, as in [9]. We chose this operation because it does not have trainable parameters and ensures a reduction in the number of parameters to train, which is consistent with our objective of developing an accurate and efficient DCNN model. Finally, the concatenation operation connected the output of the ATAS blocks in the encoding path to the corresponding output of the upsampling operation in the decoding path. The concatenation helped alleviate the problem of feature loss resulting from the max-pooling and upsampling operations.

In the end, the output of the ATAS blocks passed through a 1 × 1 convolution operation with a sigmoid activation function to generate a segmented image with an object in the foreground and black pixels in the background.

5) LOSS FUNCTION

We used the sum of two loss functions, namely cross entropy loss and dice loss, as a value for minimization. The loss function is formulated as follows:

$$L_f = \left(\frac{1}{M}\sum_{i=1}^{M} -y_i log(\hat{y}_i)\right) + \left(-\frac{2}{N}\sum_{n=1}^{N}\frac{\sum_{p=1}^{P} y_p^k \hat{y}_p^k}{\sum_{p=1}^{P} y_p^k + \sum_{p=1}^{P} \hat{y}_p^k}\right) \quad (6)$$

In (6), M, N, and P are the total number of images, classes, and pixels, respectively, and y and $\hat{y}$ are the ground truth and the predicted masks for the segmentation, respectively.

C. INFERENCE

After completing the data learning stage and obtaining a trained DCNN model, we can now employ this model to generate segmented medical images in an inference stage. In this step, the raw data should pass through the same preprocessing operations, as in the training stage, except for data augmentation. A test set of a dataset or real-life medical images was precisely resized, transformed into grayscale, and standardized using (1). For standardization, X must be the training data, i.e., the same data that was used in training and validation stages, to ensure that data in inference stage follow the same distribution. The images are then input into the trained model, which consequently generates segmented medical images.

IV. EXPERIMENTS AND RESULTS

This section describes the conducted experiments and their results and presents a comparison of the performances of the proposed method and the existing state-of-the-art models.

A. EXPERIMENT DATASETS

For the experiments, we employed four publicly available and widely used medical image datasets, namely the 2018 Liver Tumor Segmentation challenge dataset containing abdominal computed tomography (CT) scans [31], 2018 Data Science Bowl (DSB) challenge dataset containing a large number of segmented nuclei images [32], Kvasir-SEG dataset containing polyp images [33], and International Skin Imaging Collaboration (ISIC) 2018: Skin Lesion Analysis Toward Melanoma Detection challenge dataset containing dermoscopic images [34]. Real-life medical image datasets often experience a problem of limited data for training and validation [35], [36]; therefore, we used various datasets that have limited (2018 LiTS: 331) and ample (ISIC 2018: 2594) training images to test the performance of the proposed method from different angles. Table 2 presents the details of these datasets.

TABLE 2. Detailed description of the experimental datasets.

Dataset name	Image type	Image size	Number of images		
			Train	Validation	Test
2018 LiTS	Liver	512×512	201	65	65
2018 DSB	Cell nuclei	Various	603	67	65
Kvasir-SEG	Polyp	Various	800	100	100
ISIC 2018	Skin lesion	Various	2,075	260	259

B. BASELINE MODELS

We selected five recent medical image segmentation DCNN models that attain state-of-the-art performance to compare the results of the proposed method: FU-Net [9], nnU-Net [19], UNet++ [20], ESPNetv2 [21], and ResUNet++ [22]. We have provided a detailed summary of these models in the Section II; hence, we do not mention their specifications here.

C. TRAINING SETUP

We formulated the baseline and proposed methods using Python version 3.6.9 and TensorFlow Library version 2.4.0, respectively. We initialized the weight parameters based on a standard normal distribution with a mean and a standard deviation of 0 and 1, respectively, to follow the standards of the WIB-ReLU activation function [29]. We did not use bias parameters because they are canceled out while the batch normalization method is used. We used combined cross entropy and dice loss functions as the function for minimization (refer to Section III-B5) and an Adam optimizer [37] with learning rate $\eta = 3e^{-3}$, the exponential decay rate for the first moment $\beta_1 = 9e^{-1}$, and the exponential decay rate for the second moment $\beta_2 = 9e^{-3}$ to update the trainable parameters. The experiments were conducted using a 32 GB NVIDIA Tesla V100-SXM2 GPU with CUDA 10.0 with a mini-batch size of 4 for 2018 LiTS, 16 for 2018 DSB and Kvasir-SEG, and 32 for the ISIC 2018 datasets. The models required approximately 100 epochs to converge; therefore, we trained them only for this number of epochs because further training did not improve their performance.

D. EVALUATION METRICS

We assessed the performance of the baseline and proposed methods using several evaluation metrics, including pixel accuracy (PA), dice coefficient (DC), and mean intersection over union (mIoU). The formulas of these evaluation metrics

TABLE 3. Comparison of the baseline and proposed models in terms of accuracy and speed*.

Datasets	Models	PA	DC	mIoU	Training time per epoch** (s)	Inference time (ms)
2018 LiTS	AEDCN-Net	**0.969 ± 0.002**	**0.902 ± 0.005**	**0.802 ± 0.003**	**1.69 ± 0.09**	**3.42 ± 0.04**
	ESPNetv2	*0.967 ± 0.003*	*0.898 ± 0.004*	*0.799 ± 0.005*	2.71 ± 0.10	5.18 ± 0.04
	FU-Net	0.966 ± 0.003	0.891 ± 0.006	0.790 ± 0.005	*1.97 ± 0.12*	*3.98 ± 0.03*
	UNet++	0.967 ± 0.002	0.889 ± 0.003	0.786 ± 0.003	2.88 ± 0.13	5.87 ± 0.06
	nnU-Net	0.959 ± 0.004	0.864 ± 0.005	0.771 ± 0.004	3.63 ± 0.17	6.91 ± 0.05
	ResUNet++	0.962 ± 0.005	0.857 ± 0.006	0.764 ± 0.003	4.14 ± 0.13	8.03 ± 0.07
2018 DSB	AEDCN-Net	**0.980 ± 0.001**	**0.926 ± 0.004**	**0.851 ± 0.003**	**5.13 ± 0.09**	**10.96 ± 0.05**
	ESPNetv2	0.977 ± 0.002	**0.926 ± 0.003**	*0.849 ± 0.005*	8.37 ± 0.13	17.26 ± 0.06
	UNet++	**0.980 ± 0.001**	*0.924 ± 0.006*	0.847 ± 0.004	9.22 ± 0.15	20.19 ± 0.05
	nnU-Net	*0.979 ± 0.002*	0.920 ± 0.005	0.839 ± 0.003	11.40 ± 0.14	23.06 ± 0.10
	ResUNet++	0.979 ± 0.001	0.917 ± 0.006	0.834 ± 0.007	12.41 ± 0.17	23.27 ± 0.09
	FU-Net	0.966 ± 0.005	0.903 ± 0.004	0.812 ± 0.006	*5.91 ± 0.08*	*11.32 ± 0.06*
Kvasir-SEG	AEDCN-Net	0.975 ± 0.002	**0.912 ± 0.004**	**0.835 ± 0.003**	**6.97 ± 0.08**	**13.94 ± 0.04**
	ResUNet++	**0.977 ± 0.001**	*0.909 ± 0.002*	*0.833 ± 0.002*	16.48 ± 0.16	30.72 ± 0.09
	UNet++	*0.976 ± 0.001*	0.904 ± 0.003	0.832 ± 0.004	12.68 ± 0.11	23.63 ± 0.06
	nnU-Net	0.974 ± 0.002	0.905 ± 0.005	0.831 ± 0.005	15.23 ± 0.12	28.17 ± 0.09
	ESPNetv2	0.972 ± 0.001	0.893 ± 0.002	0.797 ± 0.003	11.32 ± 0.11	22.38 ± 0.10
	FU-Net	0.964 ± 0.002	0.885 ± 0.003	0.792 ± 0.002	*7.91 ± 0.10*	*15.04 ± 0.07*
ISIC 2018	ResUNet++	**0.969 ± 0.002**	**0.906 ± 0.002**	**0.807 ± 0.003**	43.01 ± 0.17	82.13 ± 0.09
	AEDCN-Net	**0.969 ± 0.002**	*0.903 ± 0.003*	*0.805 ± 0.002*	**15.37 ± 0.07**	**36.48 ± 0.05**
	nnU-Net	*0.967 ± 0.002*	0.901 ± 0.005	0.804 ± 0.004	39.80 ± 0.21	82.98 ± 0.14
	UNet++	0.963 ± 0.003	0.898 ± 0.004	0.799 ± 0.006	32.37 ± 0.12	63.27 ± 0.06
	ESPNetv2	0.962 ± 0.003	0.894 ± 0.005	0.792 ± 0.003	28.77 ± 0.09	60.04 ± 0.07
	FU-Net	0.959 ± 0.003	0.887 ± 0.005	0.782 ± 0.006	*18.93 ± 0.12*	38.76 ± 0.06

*This information is based on experiments using 32 GB NVIDIA Tesla V100-SXM2 GPU.

**Data augmentation time is included in training time per epoch.

are as follows:

$$
\begin{aligned}
PA &= \frac{1}{M}\sum_{i=1}^{M}\frac{\sum_{p}^{P}\hat{y}_p == y_p}{\sum_{p}^{P} y_p}\\
DC &= \frac{2\times TP}{2\times TP + FP + FN}\\
mIoU &= \frac{TP}{TP + FP + FN}
\end{aligned}
\tag{7}
$$

Equation (7) shows the computation methods of the considered evaluation metrics, where $\hat{y}$ and y are the predicted, and target values, respectively; P and M are the total number of pixels in an image and the total number of instances, respectively; and TP, TN, FP, and FN stand for true positive, true negative, false positive, and false negative, respectively.

V. DISCUSSION

This section discusses the results of the conducted experiments in terms of computational and memory efficiency and shares the results of ablation studies. Moreover, it exhibits qualitative comparison of the baseline and proposed methods and enumerates limitations of the proposed method.

A. EXPERIMENT RESULTS

Table 3 summarizes the experimental results of the considered models on the test sets of the aforementioned datasets. From the table, the proposed model enjoyed high speed for training and inference and significantly outperformed the existing computationally expensive models, such as ResUNet++ and nnU-Net by achieving nearly 3× of speed-up. As regards the lightweight and efficient models, AEDCN-Net attained a performance faster than those of ESPNetv2 and FU-Net, too. The proposed model was approximately 38% and 15% quicker in training (data augmentation process time is included in training time per epoch) and inference than the ESPNetv2 and FU-Net models, respectively.

In the case of the accuracy-related metrics, the proposed model considerably outperformed the baseline networks in the datasets with a limited number of medical images, like 2018 LiTS and 2018 DSB primarily because the computationally expensive models with a large number of trainable parameters experienced overfitting and could not generalize well to the unseen test data. However, in the experiments on datasets with 1000 and more images, such as Kvasir-SEG and ISIC 2018, the nn-UNet and ResUNet++ models attained better performances than the lightweight models due to a great number of computations and parameters. AEDCN-Net still could largely outperform the lightweight models and achieve at least a second best result in terms of the PA, DC, and mIoU metrics on the considered datasets.

B. COMPUTATIONAL AND MEMORY EFFICIENCY

We also compared the considered models in terms of trainable parameters, model size, and FLOPs. Table 4 presents the evaluation results.

In Table 4, AEDCN-Net required nearly seven and 15 times fewer trainable parameters in comparison with the lightweight and computationally expensive models, respectively. Moreover, the size of the proposed model was considerably smaller than the baseline networks. Finally, AEDCN-Net was efficient in terms of computation by requiring the

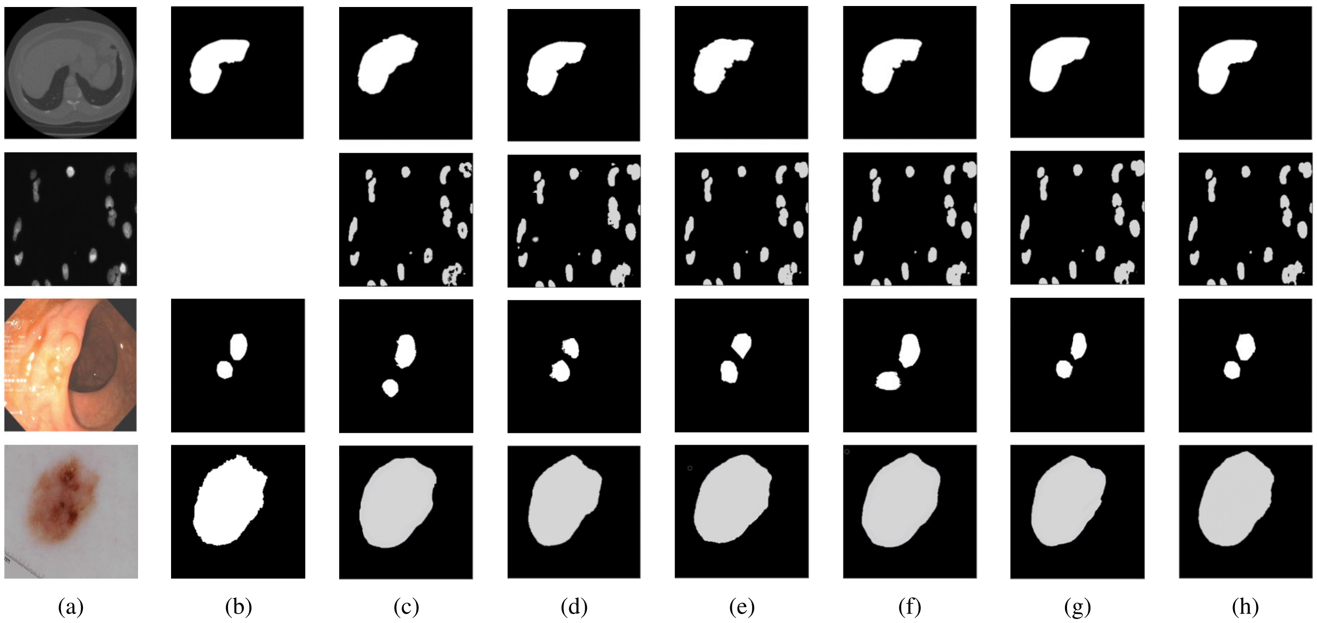

FIGURE 3. Comparison of the segmentation results: (a) input images; (b) ground truth masks; and the corresponding segmented masks using (c) FU-Net, (d) nnU-Net, (e) UNet++, (f) ESPNetv2, (g) ResUNet++, and (h) AEDCN-Net. The test set of the 2018 DSB dataset had no ground truth mask; therefore, there is no image in the second row and the second column of the figure.

TABLE 4. Comparison of the baseline and proposed models in terms of memory and computational efficiency.

Models	Params (m)	Size (mb)	FLOPs (b)
AEDCN-Net	**0.92**	**11.16**	**1.5**
FU-Net	*1.79*	*21.34*	1.8
ESPNetv2	7.27	43.28	*1.6*
UNet++	9.04	54.77	2.9
nnU-Net	13.39	79.14	3.6
ResUNet++	16.23	95.84	4.0

lowest number of FLOPs to produce the medical image segmentation.

C. ABLATION STUDIES

Table 5 analyzes the effect of different components in the proposed method on the accuracy-related evaluation metrics and number of trainable parameters. We selected the datasets with the fewest and the largest number of images to conduct ablation studies to reduce the computational cost for the experiments.

As shown in Table 5, the asymmetric convolution operation with the kernel sizes of 3 × 1, 1 × 3 always performed better than that with 5 × 1, 1 × 5 in both datasets. Moreover, the progressive increase of the atrous factor followed by a progressive decrease (2, 4, 8, 4, 2) resulted in the highest scores in the evaluation metrics when compared with the other options. In addition, the AEDCN-Net with seven blocks worked better in the dataset with limited number of images, while a more complex network with nine blocks performed well in ISIC 2018 with a large number of trainable images. Although AEDCN-Net attained the most accurate medical image segmentation, it increased the number of trainable parameters by nearly six times and resulted in a longer training and inference time; therefore, we employed AEDCN-Net with seven blocks by default.

TABLE 5. Effect of different components in the ATAS blocks, where $\uparrow a$ is a progressive increase, $\downarrow a$ is a progressive decrease; and a^* is an increase followed by a decrease of the atrous factor.

Dataset name	Convolution type / No. of blocks	PA	DC	mIoU	Params (m)
2018 LiTS	$\uparrow a$; 3×1, 1×3	0.962	0.888	0.786	0.92
	$\downarrow a$; 3×1, 1×3	0.953	0.883	0.779	0.92
	a^*; 3×1, 1×3	**0.969**	**0.902**	**0.802**	**0.92**
	$\uparrow a$; 5×1, 1×5	0.963	0.881	0.777	1.36
	$\downarrow a$; 5×1, 1×5	0.959	0.872	0.769	1.36
	a^*; 5×1, 1×5	0.960	0.874	0.764	1.36
ISIC 2018	$\uparrow a$; 3×1, 1×3	0.953	0.873	0.774	0.92
	$\downarrow a$; 3×1, 1×3	0.942	0.863	0.759	0.92
	a^*; 3×1, 1×3	**0.969**	**0.903**	**0.805**	**0.92**
	$\uparrow a$; 5×1, 1×5	0.949	0.864	0.774	1.36
	$\downarrow a$; 5×1, 1×5	0.937	0.849	0.767	1.36
	a^*; 5×1, 1×5	0.942	0.863	0.782	1.36
2018 LiTs	7 blocks	**0.969**	**0.902**	**0.802**	**0.92**
	9 blocks	0.958	0.883	0.785	5.70
ISIC 2018	7 blocks	0.969	0.903	0.805	**0.92**
	9 blocks	**0.971**	**0.907**	**0.810**	5.70

D. QUALITATIVE COMPARISON OF THE CONSIDERED MODELS

After finishing the training and evaluating the model performance on the considered datasets, we show herein the generated segmented images using the baseline and proposed methods. Figure 3 depicts the input medical images, ground truth masks, and generated segmentation masks by the considered methods. The most efficient baseline model, FU-Net, failed to generalize well on the test images. Particularly, the model's inferior performance was noticeable in the segmented images from the 2018 DSB dataset. In addition, nnU-Net produced lower-quality segmentation masks in the Kvasir-SEG and ISIC 2018 datasets. Notably, the proposed method could produce more detailed and precise segmented medical images than baseline methods in all considered datasets.

E. LIMITATIONS OF THE PROPOSED METHOD

The results of the conducted experiments using four medical image datasets and comparison of the performance with the existing state-of-the-art models showed that the proposed AEDCN-Net outperformed the baseline models in terms of speed, memory, efficiency, and accuracy. However, the proposed method have several limitations. First, some datasets used in the experiments have limited number of training set, which cannot fully demonstrate a performance difference between the proposed method and the more powerful and computationally expensive networks. Second, the considered datasets in the experiments exhibit only binary (foreground and background) output. Although, the proposed method can easily be employed for multiple output segmentation by slightly altering its activation function in the final output layer, this operation can lead to increase in computational complexity.

VI. CONCLUSION AND FUTURE WORK

This study investigated the medical image segmentation using DL-based techniques. Based on the extensive literature review, we found that the currently available state-of-the-art methods in this field are computationally inefficient and slow. Moreover, the lightweight and efficient models cannot generate precise segmented images. Therefore, we proposed the AEDCN-Net model that benefits from the carefully designed preprocessing and the computationally efficient DCNN model using skip connection-based bottleneck, atrous and asymmetric convolution operations in the encoding path and nearest-neighbor interpolation upsampling technique in the decoding path. In the conducted experiments using four open-source medical image datasets, the proposed method showed a superior performance in terms of computational efficiency, memory, and accuracy compared with the counterpart models. Moreover, the AEDCN-Net significantly outperformed the efficient models by achieving greater results when assessed using several evaluation metrics.

For the future directions of AEDCN-Net enhancement, we will work on increasing the accuracy of the proposed model and attempt to interpret the predicted segmented medical images based on the severity level of illness.

REFERENCES

[1] L. Schmarje, M. Santarossa, S.-M. Schroder, and R. Koch, "A survey on semi-, self-and unsupervised learning for image classification," *IEEE Access*, vol. 9, pp. 82146–82168, 2021.

[2] D. Zeng, X. Chen, M. Zhu, M. Goesele, and A. Kuijper, "Background subtraction with real-time semantic segmentation," *IEEE Access*, vol. 7, pp. 153869–153884, 2019.

[3] A. Valenzuela, C. Arellano, and J. E. Tapia, "Towards an efficient segmentation algorithm for near-infrared eyes images," *IEEE Access*, vol. 8, pp. 171598–171607, 2020.

[4] M. Kim, B. Park, and S. Chi, "Accelerator-aware fast spatial feature network for real-time semantic segmentation," *IEEE Access*, vol. 8, pp. 226524–226537, 2020.

[5] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Feb. 7, 2021, doi: 10.1109/TPAMI.2021.3059968.

[6] X. Wang, L. Gu, and Z. Wang, "Computer medical image segmentation based on neural network," *IEEE Access*, vol. 8, pp. 158778–158786, 2020.

[7] M. M. Rahaman, C. Li, X. Wu, Y. Yao, Z. Hu, T. Jiang, X. Li, and S. Qi, "A survey for cervical cytopathology image analysis using deep learning," *IEEE Access*, vol. 8, pp. 61687–61710, 2020.

[8] B. Olimov, B. Subramanian, and J. Kim, "Deepmednet: Deep learning based medical image segmentation model," in *Proc. Korean Inf. Sci. Soc. Conf.*, 2021, pp. 576–578.

[9] B. Olimov, K. Sanjar, S. Din, A. Ahmad, A. Paul, and J. Kim, "FU-Net: Fast biomedical image segmentation model based on bottleneck convolution layers," *Multimedia Syst.*, vol. 2021, pp. 1–14, Jan. 2021.

[10] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-net and its variants for medical image segmentation: A review of theory and applications," *IEEE Access*, vol. 9, pp. 82031–82057, 2021.

[11] B. Olimov, J. Kim, and A. Paul, "REF-Net: Robust, efficient, and fast network for semantic segmentation applications using devices with limited computational resources," *IEEE Access*, vol. 9, pp. 15084–15098, 2021.

[12] B. Olimov, J. Kim, A. Paul, and B. Subramanian, "An efficient deep convolutional neural network for semantic segmentation," in *Proc. 8th Int. Conf. Orange Technol. (ICOT)*, 2020, pp. 1–9.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1026–1034.

[14] B. Olimov, J. Kim, and A. Paul, "DCBT-Net: Training deep convolutional neural networks with extremely noisy labels," *IEEE Access*, vol. 8, pp. 220482–220495, 2020.

[15] B. Olimov and J. Kim, "DeepCleanNet: Training deep convolutional neural network with extremely noisy labels," *J. Korea Multimedia Soc.*, vol. 23, no. 11, pp. 1349–1360, 2020.

[16] K. Sanjar, O. Bekhzod, J. Kim, J. Kim, A. Paul, and J. Kim, "Improved U-Net: Fully convolutional network model for skin-lesion segmentation," *Appl. Sci.*, vol. 10, no. 10, p. 3658, May 2020.

[17] 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.* Springer, 2015, pp. 234–241.

[18] D. Li, D. A. Dharmawan, B. P. Ng, and S. Rahardja, "Residual U-Net for retinal vessel segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1425–1429.

[19] F. Isensee, J. Petersen, A. Klein, and D. Zimmerer, "nnU-Net: Self-adapting framework for U-Net-based medical image segmentation," in *Bildverarbeitung Die Medizin*. Springer, 2019, pp. 22–29.

[20] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer, 2018, pp. 3–11.

[21] S. Mehta, M. Rastegari, L. Shapiro, and H. Hajishirzi, "ESPNetV2: A light-weight, power efficient, and general purpose convolutional neural network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9190–9200.

[22] D. Jha, P. H. Smedsrud, D. Johansen, T. de Lange, H. D. Johansen, P. Halvorsen, and M. A. Riegler, "A comprehensive study on colorectal polyp segmentation with ResUNet++, conditional random field and test-time augmentation," *IEEE J. Biomed. Health Informat.*, vol. 25, no. 6, pp. 2029–2040, Jun. 2021.

[23] L. Jiao and J. Zhao, "A survey on the new generation of deep learning in image processing," *IEEE Access*, vol. 7, pp. 172231–172263, 2019.

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[25] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2018, pp. 6848–6856.

[26] N. S. Punn and S. Agarwal, "Inception U-Net architecture for semantic segmentation to identify nuclei in microscopy cell images," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 1, pp. 1–15, Apr. 2020.

[27] 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.

[28] P. K. Gadosey, Y. Li, E. A. Agyekum, T. Zhang, Z. Liu, P. T. Yamak, and F. Essaf, "SD-UNet: Stripping down U-Net for segmentation of biomedical images on platforms with low computational budgets," *Diagnostics*, vol. 10, no. 2, p. 110, Feb. 2020.

[29] B. Olimov, S. Karshiev, E. Jang, S. Din, A. Paul, and J. Kim, "Weight initialization based-rectified linear unit activation function to improve the performance of a convolutional neural network model," *Concurrency Comput., Pract. Exper.*, vol. 33, no. 22, p. e6143, Nov. 2021.

[30] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[31] P. Bilic, P. F. Christ, E. Vorontsov, and G. Chlebus, "The liver tumor segmentation benchmark (LiTS)," 2019, *arXiv:1901.04056*.

[32] J. C. Caicedo, A. Goodman, K. W. Karhohs, and B. A. Cimini, "Nucleus segmentation across imaging experiments: The 2018 Data Science Bowl," *Nature methods*, vol. 16, no. 12, pp. 1247–1253, 2019.

[33] D. Jha, P. H. Smedsrud, M. A. Riegler, P. Halvorsen, T. de Lange, D. Johansen, and H. D. Johansen, "Kvasir-SEG: A segmented polyp dataset," in *Proc. Int. Conf. Multimedia Modeling*. Springer, 2020, pp. 451–462.

[34] N. Codella, V. Rotemberg, P. Tschandl, M. Emre Celebi, S. Dusza, D. Gutman, B. Helba, A. Kalloo, K. Liopyris, M. Marchetti, H. Kittler, and A. Halpern, "Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (ISIC)," 2019, *arXiv:1902.03368*.

[35] K. C. Wong, T. Syeda-Mahmood, and M. Moradi, "Building medical image classifiers with very limited data using segmentation networks," *Med. image Anal.*, vol. 49, pp. 105–116, 2018.

[36] W. Chi, L. Ma, J. Wu, M. Chen, W. Lu, and X. Gu, "Deep learning-based medical image segmentation with limited labels," *Phys. Med. Biol.*, vol. 65, no. 23, Dec. 2020, Art. no. 235001.

[37] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

BEKHZOD OLIMOV received the B.S. degree in economics from the Fergana Polytechnic Institute, Uzbekistan, in 2014, and the M.S. degree from Yeungnam University, South Korea, in 2018. He is currently pursuing the Ph.D. degree with the Computer Science and Engineering Department, Kyungpook National University, South Korea.

His research interests include computer vision and pattern recognition using deep learning techniques. He received the best paper award in Korea Multimedia Society conference, in 2020 and acted as a Session Chair in IEEE ICOT 2020 international conference. He serves as a Reviewer for various IEEE journals, including IEEE Transactions on Industrial Informatics, IEEE Transactions on Medical Imaging, IEEE Access and IEEE Sensors Journal.

SEOK-JOO KOH received the B.S., M.S., and Ph.D. degrees from the Korean Advanced Institute of Science and Technology, Daejon, South Korea, in 1992, 1994, and 1998, respectively. He worked as a Senior Researcher with the Electronics and Telecommunications Research Institute, from 1998 to 2004.

He is currently working as a Professor with the School of Computer Science and Engineering, Kyungpook National University, South Korea. His research interests include data communications and the Internet of Things.

JEONGHONG KIM received the B.S. and M.S. degrees from Kyungpook National University, Daegu, South Korea, in 1986, and the Ph.D. degree from Chungnam National University, Daejeon, in 2001.

He worked as a Senior Researcher at the Electronics and Telecommunications Research Institute from 1988 to 1996. He worked as a Professor at the Sangju National University from 1996 to 2008. He is currently working as a Professor with the School of Computer Science and Engineering, Kyungpook National University, South Korea. His research interests include bio signal processing and pattern recognition using deep learning techniques.